Міністерство освіти і науки України
Чорноморський національний університет імені Петра Могили

Болюбаш Н. М.

ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ

Навчальний посібник

ЧНУ
ВИДАВНИЦТВО

Миколаїв – 2023

УДК 004.8+004.62
Б79

Рекомендовано до друку вченою радою Чорноморського національного університету імені Петра Могили (протокол №10 від 22.12.2022 р.)

Рецензенти:

Атаманюк І. П. – д-р техн. наук, професор, завідувач кафедри вищої та прикладної математики Миколаївського національного аграрного університету.

Волосюк Ю. В. – канд. техн. наук, доцент, завідувач кафедри інформаційних систем та технологій обліково-фінансового факультету Миколаївського національного аграрного університету.

Б79 **Болюбаш Н. М.** Інтелектуальний аналіз даних : навч. посіб. / Н. М. Болюбаш. – Миколаїв : Вид-во ЧНУ ім. Петра Могили, 2023. – 320 с.

ISBN 978-966-336-443-8

У навчальному посібнику розглянуто базові питання, необхідні для теоретичної та практичної підготовки майбутніх фахівців з інформаційних технологій у галузі інтелектуального аналізу даних. За змістом та структурою видання відповідає робочій програмі дисципліни «Інтелектуальний аналіз даних» для бакалаврів спеціальності 122 «Комп'ютерні науки».

Посібник спрямовано на формування практичних умінь і навичок використання алгоритмів і методів Data Mining на етапі попередньої обробки даних та при розв'язанні задач кластерного аналізу, класифікації, пошуку асоціативних правил і послідовностей, виявлення зв'язків та закономірностей, прогнозування. Він містить розділи, які включають систематизований стисло викладений теоретичний матеріал, необхідний для актуалізації знань студентів при підготовці до виконання лабораторних робіт, методичні рекомендації й інструкції з виконання практичних завдань, завдання для самостійного виконання за індивідуальними варіантами, перелік питань для підготовки до захисту лабораторних робіт і контролю набутих знань. Подано рекомендації щодо оформлення звітів із виконання лабораторних робіт та сформульовано критерії їх оцінювання.

Навчальний посібник призначений для студентів та викладачів факультету комп'ютерних наук.

УДК 004.8+004.62

ISBN 978-966-336-443-8

ЗМІСТ

ВСТУП

Відповідальним етапом у професійному становленні майбутнього фахівця з інформаційних технологій є отримання теоретичної та практичної підготовки в області інтелектуального аналізу даних (англ. *Data Mining*), яка здійснюється у процесі вивчення дисципліни «Інтелектуальний аналіз даних».

Основними завданнями дисципліни «Інтелектуальний аналіз даних» є:

1) надання студентам цілісного уявлення про наукові основи, сучасну методологію та особливості застосування інтелектуального аналізу даних;

2) ознайомлення майбутніх фахівців із теоретичними основами інформаційних технологій, орієнтованими на застосування стандартів Data Mining, практичною значимістю методів і засобів інтелектуального аналізу даних, їх застосуванням при розв'язанні найрізноманітніших гуманітарних, технічних, економічних і наукових проблем;

3) формування умінь та навичок реалізації алгоритмів та методів Data Mining із використанням програмних засобів підтримки технологій інтелектуального аналізу даних;

4) формування умінь та навичок, необхідних для програмування окремих елементів технологій Data Mining різного призначення і різної проблемної орієнтації;

5) виховання у майбутніх фахівців творчого підходу до розв'язання задач, пов'язаних із аналітичним дослідженням великих масивів інформації з метою виявлення нових, раніше невідомих, практично корисних знань і закономірностей, необхідних для прийняття рішень;

6) розвиток здатності і відчуття необхідності до постійної самоосвіти і самовдосконалення, підготовка студентів до самостійної роботи з вирішення задач засобами інтелектуального аналізу даних і розробки інтелектуальних систем, формування елементів інформаційної культури, самостійного дослідницького характеру пошуку нових знань.

Вивчення дисципліни «Інтелектуальний аналіз даних» базується на знанні основ вищої та дискретної математики, має логічний і змістовно-методичний зв'язок із іншими навчальними дисциплінами та безпосередньо спирається на теорію ймовірностей і математичну статистику, теорію алгоритмів, організацію баз даних та знань, математичні методи дослідження операцій, що знаходить відображення у міжпредметних зв'язках цих дисциплін зі змістом курсу «Інтелектуальний аналіз даних».

Навчальний посібник «Інтелектуальний аналіз даних» розкриває основні вимоги до виконання лабораторних робіт відповідно до робочої програми дисципліни для студентів спеціальності 122 «Комп'ютерні науки» першого рівня вищої освіти «бакалавр». У процесі виконання лабораторних робіт здійснюється закріплення знань та формування практичних умінь і навичок застосування алгоритмів та методів Data Mining у процесі розв'язання задач, дослідження великих за обсягом цифрових наборів даних із метою виявлення знань, необхідних для прийняття рішень у певній предметній області.

Відповідно до ОПП вивчення дисципліни «Інтелектуальний аналіз даних» спрямоване на формування у студентів наступних компетентностей:

1) здатність до абстрактного мислення, аналізу та синтезу;

2) здатність застосовувати знання у практичних ситуаціях;

3) здатність до пошуку, оброблення та аналізу інформації з різних джерел;

4) здатність до інтелектуального аналізу даних на основі методів обчислювального інтелекту, включно з великими та погано структурованими даними, їхньої оперативної обробки та візуалізації результатів аналізу у процесі розв'язування прикладних задач.

Вивчення дисципліни спрямоване на формування наступних програмних результатів навчання:

1) застосування знання основних форм і законів абстрактно-логічного мислення, основ методології форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук;

2) використовувати знання закономірностей випадкових явищ, їх властивостей та операцій над ними, моделей випадкових процесів та сучасних програмних середовищ для розв'язування задач статистичної обробки даних і побудови прогнозних моделей;

3) застосовувати методи та алгоритми обчислювального інтелекту та інтелектуального аналізу даних у задачах класифікації, прогнозування, кластерного аналізу, пошуку асоціативних правил із використанням програмних інструментів підтримки багатовимірного аналізу даних на основі технологій Data Mining.

У посібнику приділена увага кібернетичним методам Data Mining та самостійній роботі студентів. Виконання лабораторних робіт передбачає формування практичних умінь та навичок використання програмних засобів із вбудованими алгоритмами Data Mining: SPSS, MatLab, MathCAD, MS Excel.

У першому розділі навчального посібника розкрито базові поняття Data Mining та описано етапи підготовки до лабораторної роботи, її виконання й захисту, сформульовано вимоги до оформлення звіту виконаної роботи та критерії її оцінювання. Наступні розділи містять лабораторні роботи, які розкривають базові поняття Data Mining й охоплюють попередню обробку даних, основні алгоритми та методи розв'язання задач кластеризації, класифікації, пошуку асоціативних правил та послідовностей, виявлення зв'язків та закономірностей, прогнозування. Кожна лабораторна робота містить систематизований стислий виклад теоретичного матеріалу, завдання для самостійного виконання за індивідуальними варіантами та рекомендації з їх виконання і список питань для підготовки до захисту.

З огляду на високі темпи розвитку сфери Data Mining, її міждисциплінарний характер та яскраво виражену прикладну направленість, спрямовану на високопродуктивну інтелектуальну аналітичну обробку значних за обсягом масивів накопичених цифрових даних із метою оперативного отримання цінних знань для підтримки управлінської діяльності, існує певний дефіцит систематизованих уявлень у цій області серед ІТ-фахівців, на усунення яких націлений цей навчальний посібник.

ОСНОВИ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ

Базові поняття Data Mining

Виникнення та розвиток інтелектуального аналізу даних як перспективного напряму у сфері інформаційних технологій обумовлене бурхливими темпами інформатизації сучасного суспільства, які супроводжуються накопиченням великих обсягів даних, представлених у цифровому форматі та постійним вдосконаленням апаратного, програмного забезпечення і технологій збереження, передачі й обробки інформації.

Великі масиви структурованих, слабоструктурованих і неструктурованих даних різних форматів містять ***приховані знання*** (англ. *Hidden Knowledge*), які аналітик без застосування алгоримів та методів інтелектуального аналізу даних дослідити не має можливості.

В основу інтелектуального аналізу даних покладено концепцію ***шаблонів*** (англ. Patterns), що є закономірностями, які властиві наборам даних і можуть бути подані у формі, зрозумілій людині.

Інтелектуальний аналіз даних (англ. Data Mining) застосовується для виявлення у необроблених великих за обсягом даних нових, раніше невідомих нетривіальних, практично корисних і доступних інтерпретацій знань, необхідних для прийняття рішень у різних сферах життєдіяльності.

Термін «Data Mining» походить від двох англомовних понять: «data» – дані та «mining» – видобуток і однозначного перекладу українською мовою не має. Використовуються такі трактування: видобуток даних, розвідка даних, глибинний аналіз даних, просіювання інформації, вилучення даних, інтелектуальний аналіз даних. Деякі дослідники вважають невдалими більшість варіантів перекладу й оперують англомовними термінами. Ми будемо дотримуватися термінів *Інтелектуальний аналіз даних* та *Data Mining*.

Інтелектуальний аналіз даних є процесом збору, очищення, опису та моделювання даних для виявлення нових закономірностей: кореляцій, тенденцій, шаблонів, зв'язків, категорій. Для подання отриманих знань у Data Mining служать моделі.

Сучасна термінологія, яку фахівці використовують при описі предметної області інтелектуального аналізу даних, в умовах інтенсивного розвитку галузі не є остаточно встановленою. Часто використовують різні терміни, які описують одну і ту ж сферу чи явище або один термін, який може трактуватися по-різному. Поширеними є терміни «Data Science» та «Data Scientist». Data Scientist є фахівцем – аналітиком даних, який працює у сфері Data Science. Між Data Science та Data Mining досить важко найти відмінності, а інколи їх розглядають як синоніми, оскільки вони використовують одні й ті ж методи та алгоритми.

Останнім часом IT-фахівці, які займаються аналізом даних, почали розрізняти напрями діяльності Data Science та Data Mining. Data Mining асоціюють більше зі збором, попередньою підготовкою, перетворенням та візуалізацією в основному структурованих даних, а Data Science розглядають як більш широку сферу діяльності, яка працює з усіма видами даних і орієнтована на аналіз тенденцій, побудову прогностичних моделей. Однак єдина точка зору стосовно їх розмежування серед аналітиків відсутня.

Ми будемо виходити з того, що Data Science є наукою про дані і більше зосереджується на їх дослідженні, а Data Mining є процесом, який націлений на отримання практичного результату і включає повний цикл перетворення даних у знання, необхідні для прийняття рішень у певній предметній області.

Інтелектуальний аналіз даних є ***міждисциплінарною областю дослідження***, яка виникла і розвивається на базі теорії ймовірностей, математичної статистики, теорії інформації і математичної логіки, розпізнавання образів, штучного інтелекту, алгоритмізації і програмування, теорії баз даних, машинного навчання, візуалізації даних та багатьох інших дисциплін. У діючих системах Data Mining реалізовано велику кількість методів та алгоритмів, які інтегрують у собі відразу кілька міждисциплінарних підходів.

До ***основних задач*** Data Mining відносять:

1) *класифікацію* (англ. Classification) – найбільш просту та розповсюджену задачу встановлення функціональної залежності між вхідними і дискретними вихідними змінними, яка дозволяє віднести об'єкти до одного із заздалегідь відомих класів;

2) *кластеризацію* (англ. Clustering) – групування об'єктів на основі даних, що описують їх сутність, результатом якого є поділ набору даних на *кластери* – групи споріднених об'єктів;

3) *пошук асоціацій* (англ. Associations) – знаходження закономірностей між пов'язаними подіями, які відбуваються одночасно;

4) *пошук послідовностей* (англ. Sequence), або *пошук послідовних асоціацій* (англ. Sequential Association) – встановлення закономірностей між подіями, пов'язаними за часом;

5) *прогнозування* (англ. Forecasting) – визначення тенденцій динаміки значень показників об'єктів набору даних на основі їхніх характеристик;

6) *виявлення відхилень* (англ. Deviation Detection) – виявлення й аналіз нехарактерних шаблонів даних, які найбільше відрізняються від загальної сукупності об'єктів набору даних;

7) *аналіз зв'язків* (англ. Link Analysis) – задачу знаходження залежностей у наборі даних;

8) *візуалізацію* (англ. Visualization, Graph Mining) – створення графічного образу аналізованих даних із використанням методів, які наочно показують наявність закономірностей у даних;

9) *підбивання підсумків* (англ. Summarization) – опис конкретних груп об'єктів, обраних із аналізованого набору даних.

За призначенням описані вище задачі поділяються на ***дескриптивні*** (описові) та ***предикативні*** (передбачаючі). Дескриптивні задачі поліпшують розуміння даних, до них відносять кластеризацію, пошук асоціативних правил. Предикативні задачі будують модель на основі набору даних, яка використовується для прогнозу результатів з новими даними. До предикативних задач відносять класифікацію, прогнозування, виявлення зв'язків.

За способами розв'язання задачі поділяються на такі, що вирішують за допомогою вчителя та без його допомоги:

a) ***навчання з учителем*** (англ. Supervised Learning) – передбачає побудову моделі аналізованих даних на наборі даних із відомим характеристиками та використання цієї моделі для подальшого аналізу нових даних (класифікація, прогнозування);

b) ***навчання без учителя*** (англ. Unsupervised Learning) – передбачає побудову моделі без будь-яких попередніх знань про дані з метою виявлення закономірностей та структур даних (кластеризація, пошук асоціативних правил).

Повний цикл розв'язання задачі Data Mining включає:

1) постановку задачі;
2) первинний аналіз даних;
3) підготовку даних до аналізу;
4) інтелектуальний аналіз даних із використанням обраного методу;
5) інтерпретацію отриманих результатів;
6) використання отриманих знань для прийняття рішень.

Для розв'язання задач Data Mining існує багато методів та алгоритмів.

Метод – це систематизована сукупність дій для вирішення поставленої задачі.

Алгоритм – конкретна послідовність дій, точний набір інструкцій для перетворення вхідних даних у шуканий результат.

Опис методу дає більш повне уявлення про ідеї, реалізовані в алгоритмі. Однак досить часто ці поняття розглядають як синоніми.

Схема перетворення даних із застосуванням технології Data Mining для виявлення знань включає наступні *етапи* (рис. 1.1):

1) відбір: отримання цільових даних (англ. Target data);
2) попередня обробка даних (англ. Preproccessing data);
3) перетворення даних (англ. Transformation data);
4) інтелектуальний аналіз даних: пошук закономірностей та побудова моделей (англ. Patterns / Model);
5) інтерпретація отриманих результатів: отримання нових знань (англ. Knowledge).

Data Mining як процес виявлення в загальних масивах даних раніше невідомих практично корисних знань, необхідних для прийняття рішень, має нічим не обмежені сфери застосування у різних галузях людської діяльності: банківській справі, маркетингу, страхових, податкових та кредитних компаніях, роздрібній торгівлі, охороні здоров'я та фармацевтичних компаніях, Web-аналітиці, рекомендаційних системах тощо.

Головною особливістю Data Mining є поєднання широкого математичного інструментарію (від класичного статистичного аналізу до нових кібернетичних методів) і останніх досягнень у сфері інформаційних технологій.

Існує багато програмних засобів із вбудованим алгоритмами інтелектуального аналізу даних, використання яких значно прискорює роботу аналітиків: Weka, IBM SPSS, MatLab, MathCAD та ін. При розробці власних програмних продуктів для розв'язання задач Data Mining програмісти можуть використовувати широкий спект спеціалізованих бібліотек із реалізованими алгоритмами інтелектуального аналізу даних. Проте використання таких засобів потребує високої кваліфікації фахівців у сфері Data Mining, оскільки технологія не може повністю замінити аналітика.

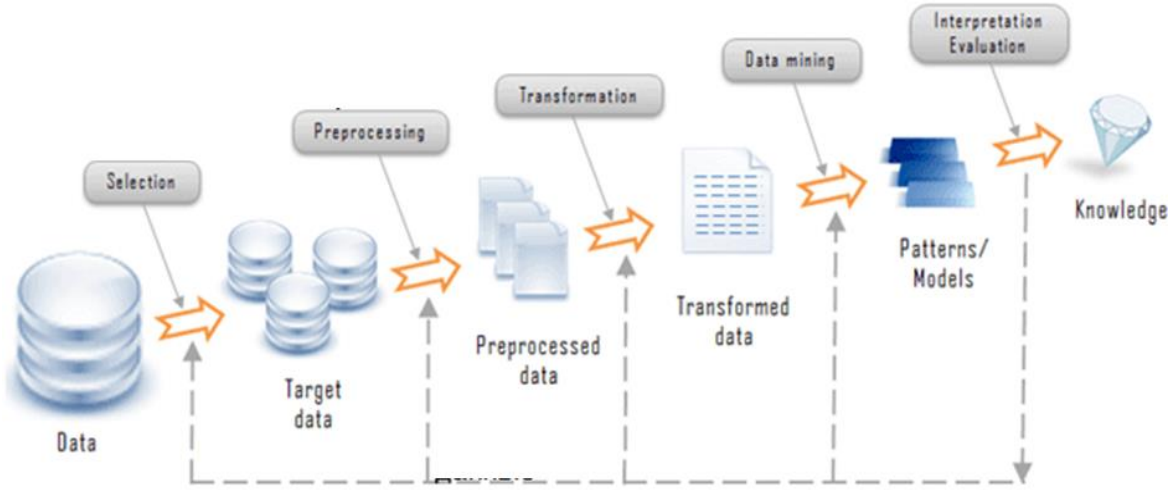


Рис. 1.1. Схема перетворення даних із застосування технології Data Mining

Порядок виконання лабораторних робіт з Data Mining

Лабораторні роботи мають важливе значення у структурі дисципліни «Інтелектуальний аналіз даних», оскільки вони спрямовані на закріплення знань та формування практичних умінь і навичок застосування алгоритмів та методів Data Mining у процесі розв'язання задач інтелектуального аналізу даних.

Навчальний посібник містить лабораторні роботи відповідно до робочої програми дисципліни. Кожна ***лабораторна робота містить***:

1) стислий виклад теоретичного матеріалу, яким необхідно володіти для виконання завдань лабораторної роботи;
2) приклади розв'язання поставлених задач із застосуванням алгоритмів та методів, засвоєння яких є метою лабораторної роботи;
3) практичні завдання, частина яких може бути завданнями, які необхідно виконати за зразком, а частина – завданнями для самостійного виконання за індивідуальним варіантом;
4) список контрольних питань із теми лабораторної роботи.

Виконання кожної лабораторної роботи включає як аудиторну, так і позааудиторну роботу студента над матеріалом окремої теми й включає такі ***етапи***.

1. *Підготовка до виконання лабораторної роботи*: передбачає вивчення теоретичного матеріалу та розбір прикладів з теми лабораторної роботи у позааудиторний час із використанням методичних матеріалів, наданих викладачем.

2. *Виконання лабораторної роботи*: здійснюється у комп'ютерному класі під керівництвом викладача й включає роботу студента з виконання практичних завдань із використанням програмних засобів, які передбачено використовувати для реалізації методів та алгоритмів в даній лабораторній роботі.

3. *Оформлення звіту з лабораторної роботи*. Виконання лабораторної роботи необхідно оформити у вигляді звіту, який здається в електронному вигляді й містить:

1) титульну сторінку, оформлену відповідно до вимог (додаток *А*);
2) назву лабораторної роботи;
3) мету роботи;
4) короткі теоретичні відомості з теми лабораторної роботи;
5) поставлені завдання та опис етапів їх розв'язання з детальним поясненням та інтерпретацією отриманих результатів.

В електронному вигляді необхідно також здати файли, які містять результати виконання поставлених завдань.

4. *Захист лабораторної роботи*. Лабораторну роботу після оформлення та здачі звіту необхідно захистити при безпосередньому спілкуванні з викладачем. При підготовці до захисту необхідно скористатися списком контрольних питань, призначених для актуалізації знань з теми лабораторної роботи.

За результатами захисту кожної лабораторної роботи студент отримує певну кількість балів відповідно до рейтингової системи оцінювання, яка доводиться до відома студентів на початку вивчення дисципліни. При оцінюванні лабораторної роботи максимальна кількість балів буде нарахована у випадку, якщо студент на захисті продемонстрував знання теоретичного матеріалу, необхідні для виконання завдань, засвоїв основні алгоритми аналізу даних, необхідні для їх розв'язання та виконав у повному обсязі усі поставлені завдання лабораторної роботи.

1. ОБРОБКА ДАНИХ У СЕРЕДОВИЩІ MATLAB. НОРМАЛІЗАЦІЯ, СТАНДАРТИЗАЦІЯ ДАНИХ

Лабораторна робота № 1

Мета: формування та закріплення знань про призначення та основи інтерфейсу MatLab. Набуття навичок здійснення обробки та аналізу даних у MatLab. Формування умінь здійснення нормалізації та стандартизації даних.

Теоретичні знання: призначення MatLab, структура вікна. Змінні, обчислення математичних виразів. Створення mat-файлів. Робота з матрицями. Файли-функції та файли-сценарії. Візуалізація даних – створення графіків, теплових карт. Програмування основних алгоритмічних елементів. Основні формули нормалізації та стандартизації даних.

1.1. ОСНОВИ РОБОТИ У СИСТЕМІ MATLAB

1.1.1. Призначення та особливості інтегрованого програмного комплексу MatLab

MatLab є інтегрованим програмним комплексом, інтерактивним середовищем для виконання різноманітних наукових та інженерних розрахунків. Основою розрахунків є матричні розрахунки, назва походить від Matrix Laboratory – матрична лабораторія.

Програмний пакет містить велику бібліотеку програм по чисельним методам, використовує двовимірну та тривимірну графіку та формати мов високого рівня, що дозволяє виконувати, модифікувати та створювати програми реалізації алгоритмів інтелектуального аналізу даних.

Існує можливість написання власних функцій та сценаріїв за допомогою мови програмування MatLab.

Файли, з якими працює програма, мають розширення:

– *.mat – бінарний файл – документ програми, містить введені змінні та їх значення;

– *.m – файл, що містить програми мовою програмування MatLab (функції та сценарії);

– *.asv – резервна копія m-файлу, створюється при його збереженні.

Мова програмування MatLab є об’єктно-орієнтованою мовою високого рівня інтерпретуючого типу, підтримує процедурне та візуально-орієнтоване програмування. Кожна інструкція MatLab одразу ж розпізнається та виконується, етап компіляції відсутній. Кінцеві програми у вигляді файлів програм, що виконуються, відсутні, програми існують як m-файли.

Інтерпретуючий характер мови програмування MatLab означає, що з перших же кроків спілкування з програмою буде описуватися її мова програмування.

Сесія – сеанс роботи з MatLab, є поточним документом, який відображає роботу користувача з MatLab.

1.1.2. Знайомство з інтерфейсом, структурою вікна програми MatLab

Завдання 1. Ознайомитися з основами роботи у середовищі MatLab.

Основними елементами робочого середовища MatLab є (рис. 1.2):

1. *Стрічка* із вкладками, що містять групи споріднених команд.

2. Вікно *Workspace* – робоча область, яка містить усі змінні та функції поточної сесії роботи з MatLab.

2. Вікно *Current Folder* – робочий каталог з m-файлами та mat-файлами, які використовуються у поточній сесії роботи з MatLab.

3. Вікно *Command Window* – є аналогом командного рядка, призначене для введення команд і виведення результатів. У вікні *Command Window* вводяться команди, виводяться результати, повідомлення про помилки.

Одразу після завантаження MatLab користувач потрапляє в основне вікно *Command Window*. По замовчуванню система готова до обчислень у ***командному режимі***.

У цьому режимі робота з програмою має діалоговий характер, основні команди управління вікном наведені у додатку *Б*.

1.1.3. Виконання розрахунків із обчисленням математичних виразів

Для обчислення математичного виразу необхідно ввести його у командному рядку і натиснути клавішу *Enter*. Арифметичні вирази можуть містити знаки операцій, константи, змінні та виклики функцій (додаток *В*, додаток *Г*, додаток *Д*).

Змінна створюється при першому присвоюванні їй значення. Тип змінної та розмірність масиву для вектора чи матриці визначаються автоматично.

Ім'я змінної може складатися з довільної комбінації букв та цифр, але не більше 19 знаків, та починатися з букви. Дані можуть бути занесені в робочу область у скалярній та матричній формі. MatLab не вимагає декларації типу даних чи їх розміру.

Текстові коментарі можна вводити у робочій області, після символу %.

У випадку занадто довгого математичного виразу, для якого не вистачає рядка, його можна перенести на нижній рядок з допомогою знака **…** у кінці.

Ctrl + C – одночасне натискання цих клавіш дозволяє у випадку помилок переривати обчислення, що зациклилися.

Якщо змінній не привласнити ім'я, їй автоматично надається ім'я *ans*.

Виконайте за зразком обчислення математичних виразів, наведених у прикладах, ввівши відповідні команди у вікні *Command Window*.

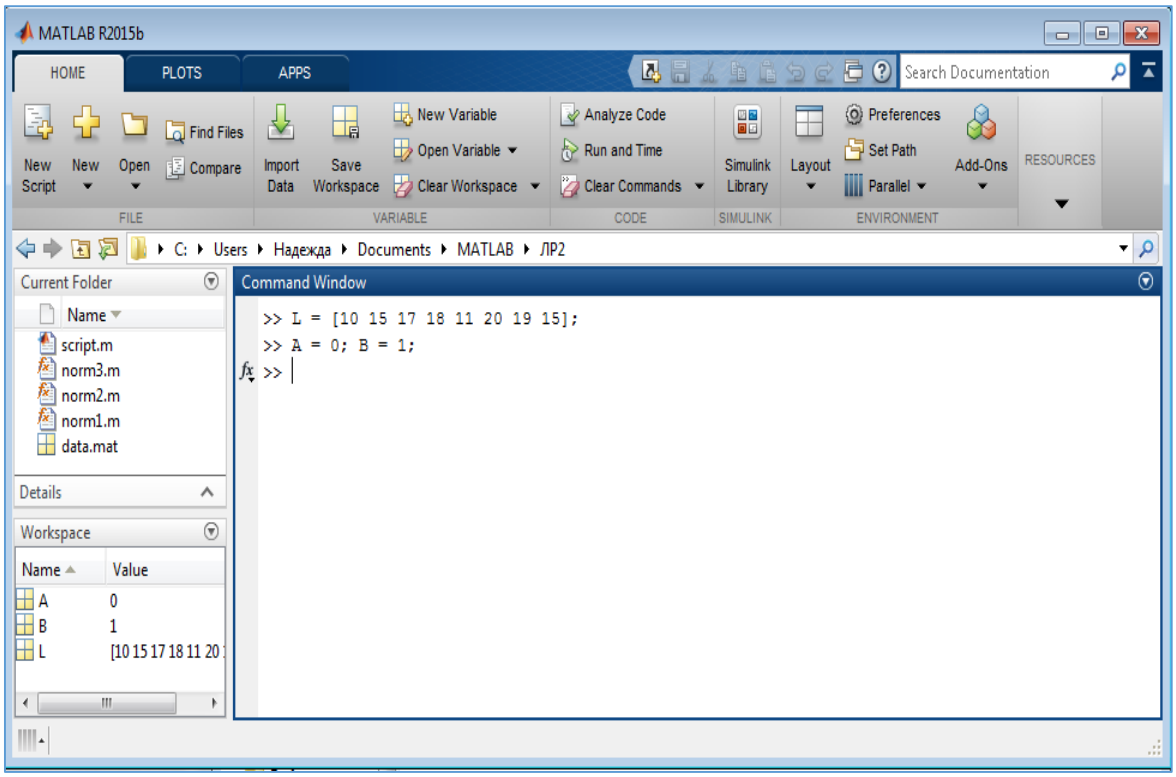


Рис. 1.2. Вікно програми MatLab

Приклад 1. Введіть оператор присвоювання:

```
>>A = 10*sin(0.35*pi) + 12^(3/2)
```

Після виконання цього оператора буде створено змінну *A* і поміщено в неї результат обчислення виразу. Ця ж змінна *A* з'явиться в області *Workspace*. Щоб вивести на екран поточне значення змінної, необхідно ввести її ім'я у командному рядку та натиснути клавішу *Enter*.

Приклад 2. Виведіть на екран поточне значення змінної A:

```
>> A
A =
50.4793
```

Приклад 3. Введіть значення змінної без привласнення їй імені:

```
 >> 45
ans =
      45
```

Якщо двічі клацнути мишею змінну A у вікні *Workspace*, вона буде відображена у вікні *Variables A* як матриця розміром 1x1 (рис. 1.3).

Значення змінної A можна змінити, ввівши у комірці вікна *Variables A* нове значення.

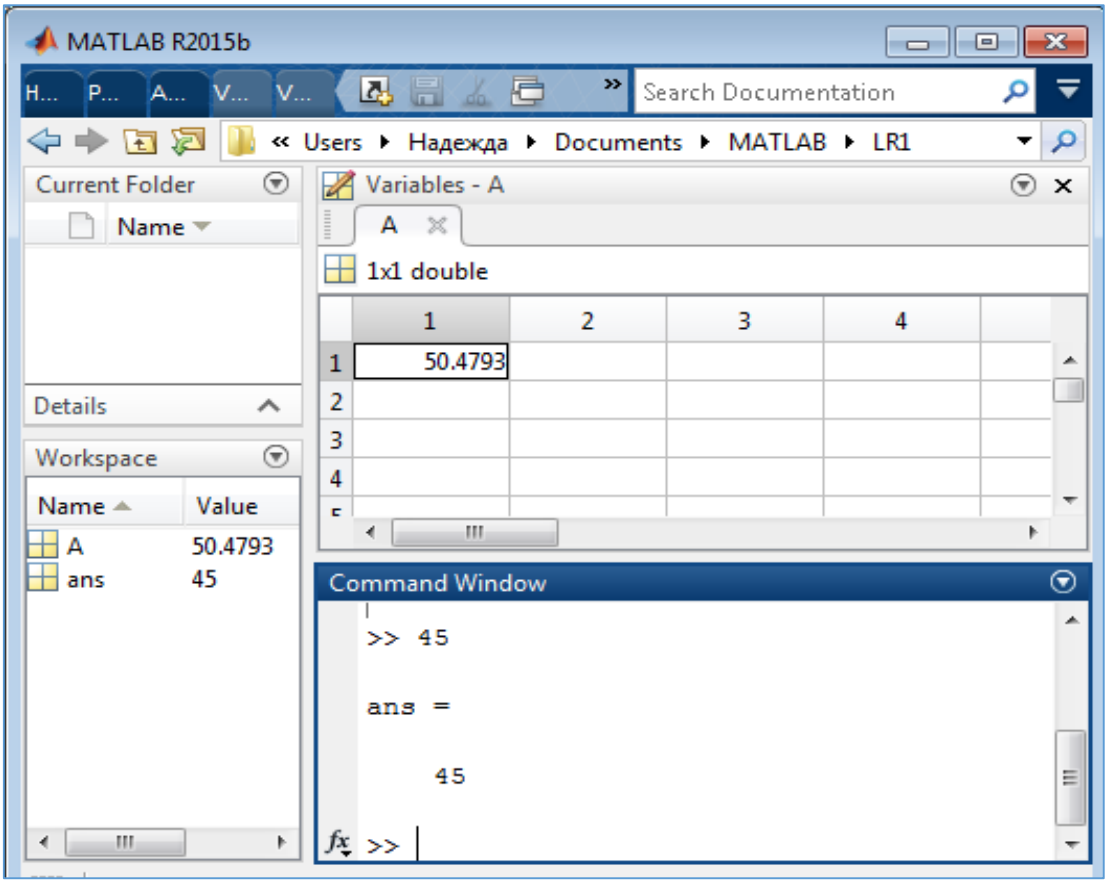


Рис. 1.3. Відкриття вікна Variables A у середовищі MatLab

Приклад 4. Змініть значення змінної A у вікні Variables A на 25 та введіть команду у вікні Command Window для виведення зміненого значення змінної A:

```
>> A
A =
25
```

Змінну A у вікні *Variables A* можна перетворити на матрицю, ввівши наприклад у комірці (1, 2) значення 6. У вікні *Workspace* A буде представлена як матриця.

Приклад 5. У вікні Command Window вводимо команду для виведення значення змінної A у вигляді матриці:

```
>> A
A =
25   6
```

Для очищення вікна *Command Window* необхідно ввести команду:

```
>> clc
```

Приклад 6. Для повторення раніше введеної команди у вікні Command Window необхідно скористатися клавішами управління курсором. При натисненні клавіші ↑ відкриється вікно, у якому необхідно обрати введену раніше команду клацанням лівої кнопки миші або за допомогою клавіш ↑ і ↓ та натиснення клавіші *Enter* (рис. 1.4).

Використання *Історії команд* / *Command History* прискорює введення команд у вікні *Command Window*, оскільки є можливість відобразити раніше введену команду та відредагувати її.

1.1.4. Збереження змінних поточної сесії у mat-файлі та їх повторне використання

Значення змінних та функцій, розміщених у робочій області, можна зберегти як файл з розширенням *.mat*:

1) обрати вкладку *HOME* – групу *VARIABLE* – кнопку *Save Workspace*;

2) у вікні, що відкриється, вказують ім'я файлу та місце його збереження;

3) у вікні *Current Folder* / *Активна папка* MatLab з'являється створений файл;

4) збережені у mat-файлі змінні та функції можуть бути знову завантажені у робочу область пам'яті у наступній сесії або після їх видалення, для цього mat-файл необхідно відобразити у вікні *Current Folder / Активна папка* та двічі клацнути лівою кнопкою миші або перетягнути цей файл у вікно *Command Window*, ухопившись за нього лівою кнопкою миші.

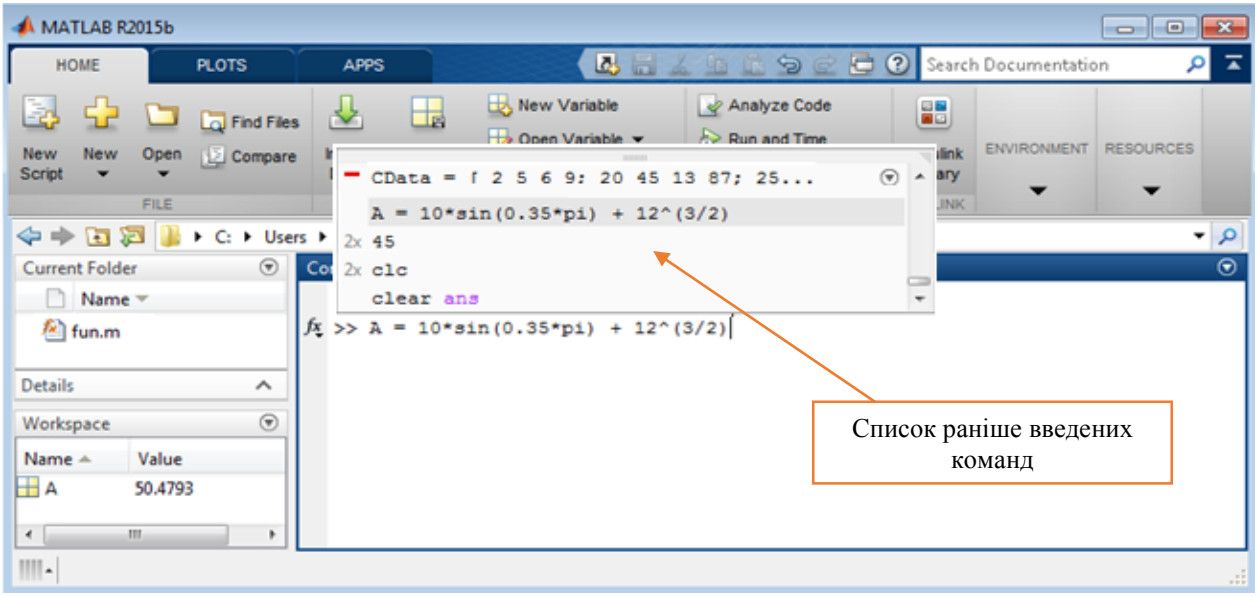


Рис. 1.4. Вікно програми MatLab з відображенням раніше введених команд

Ці ж операції можна виконати за допомогою команд MatLab.
Для збереження змінних робочої області та їх значень необхідно ввести команду:

>> save FILENAME / або save('FILENAME'), де

FILENAME – ім'я файлу, в якому будуть збережені змінні.
Для збереження значення окремих змінних після імені файлу необхідно подати їх перелік. Наприклад:

>> save C:\Users\mia\FILENAME.mat X Y Z

Для завантаження збережених змінних у робочу область необхідно виконати команду:

>> load FILENAME / або load('FILENAME')

Для завантаження окремих змінних X, Y:

>> load FILENAME X Y

Приклад 7. Виконайте збереження створених у поточному сеансі роботи змінних у файлі з іменем *Ваше_прізвище.mat* у папці Data Mining\MatLab.

Приклад 8. У вікні Command Window введіть команду знищення змінних у робочій області:

>>clear

Якщо ввести команду >> *clear A*, то буде знищено тільки вказану змінну *A*.

Приклад 9. Завантажте у робочу область змінні зі збереженого вами файлу Ваше_прізвище.mat.

Якщо при завантаженні файлу виникають проблеми, необхідно каталог, у кому було збережено файл, додати до списку пошуку за допомогою команди:

>> addpath('шлях до файлу');

1.1.5. Робота з матрицями

Усі змінні MatLab інтерпретуються як матриці. Матриці та вектори записуються у квадратних дужках, елементи матриці в одному рядку розділяються пробілом, рядки відділяються знаком «;» (крапка з комою). Розмірність матриць за необхідності можна змінювати у процесі обчислень.

Приклад 10. Введіть команду для створення матриці та виведіть її на екран:

```
>> M  = [10 15 17; 9 6 12; 3 44 25]
```

На екран буде виведено результат:

```
M  =
   10    15    17
    9     6    12
    3    44    25
```

При роботі з матрицями зручно користуватися оператором діапазону ":" що записується таким чином: *Початок_Діапазону:Крок:Кінець_Діапазону*

Приклад 11. Введіть запис:

```
>> V = 1:0.5:3
```

Він дасть результат:

```
V =
   1.0000    1.5000    2.0000    2.5000    3.0000
```

Крок можна не задавати, тоді його значення приймається рівним одиниці.

Приклад 12. Введіть запис:

```
>> V1= 1:3
```

Він дасть результат:

```
V1 =
   1    2    3
```

Для отримання доступу до конкретного елемента матриці необхідно задати його індекс у круглих дужках.

Приклад 13. Введіть команду для виведення на екран окремого елемента матриці:

```
>> M(2,3)
ans =     12
```

Приклад 14. Введіть команду для зміни окремого елемента матриці та переконайтеся, що зміну здійснено:

```
>> M(2,3) = 19;
>> M
M =
   10    15    17
    9     6    19
    3    44    25
```

Знак «;» (крапка з комою) у кінці команди, що вводиться, відміняє виведення інформації про виконанні команди на екран.

В якості індексів можна застосовувати діапазони.

Приклад 15. Введіть команду для виведення діапазону елементів матриці, розміщених у першому та другому рядках першого стовпця матриці:

```
>> M(1:2,1)
ans =
   10
    9
```

Знак «:» без крайніх значень діапазону означає звертання до всього рядка чи стовпця.

Приклад 16. Введіть команду для виведення елементів другого рядка матриці:

```
>> M(2,:)
ans =
    9     6    19
```

Над окремими елементами матриці можна застосовувати операції:

«.*» – множення, « ./» – ділення, «.^» – піднесення до степеня.

Приклад 17. Введіть наступні команди, які демонструють виконання арифметичних операцій над елементами матриці:

```
>> T=[4 56 8; 2 9 5]
T =
    4    56     8
    2     9     5
>> Y=T*2
Y =
    8   112    16
    4    18    10
>> M = Y/5
M =
   1.6000   22.4000    3.2000
   0.8000    3.6000    2.0000
>> M.^2
ans =
   2.5600  501.7600   10.2400
   0.6400   12.9600    4.0000
```

Для вилучення елементів матриці використовуються порожні квадратні дужки [ ] та оператор **:** (двокрапка).

Приклад 18. Введіть команди для створення матриці M та вилучення у ній другого стовпця і другого рядка:

```
>> M=[1 2 3; 4 5 6; 7 8 9]
M =
1 2 3
4 5 6
7 8 9
```

Вилучимо другий стовпець:

```
>> M(:,2)=[]
M =
1 3
4 6
7 9
```

Вилучимо другий рядок:

```
>> M(2,:)=[ ]
M =
1 3
7 9
```

Деякі корисні функції для роботи з матрицями наведені у додатку *E*.

1.2. ПРОГРАМУВАННЯ У СЕРЕДОВИЩІ MATLAB

1.2.1. Пакетний режим роботи: файли-сценарії та файли функцій

Система MatLab підтримує ***пакетний режим*** роботи, в якому можна розробляти програми, що складаються з послідовності команд користувача та зберігаються на диску у вигляді окремого m-файлу.

Файл-сценарій, або *Script-файл*, є найпростішою програмою із записом серії команд без параметрів. Файл-сценарій не має вхідних і вихідних аргументів, використовує тільки глобальні змінні з робочої області, у процесі виконання не компілюється. Файл-сценарій має наступну структуру: основний коментар, додатковий коментар, тіло файлу з будь-якими виразами.

У MatLab користувач може визначити функцію, написавши власний ***m-файл-функцію***, що буде засобом розширення системи. При виявленні файлу-функції він компілюється і потім виконується, а створені машинні коди зберігаються в робочій області системи MatLab. Усі змінні, наявні у тілі файлу-функції, є локальними, тобто діють тільки у межах тіла функції.

М-файли функції створюються, редагуються і налагоджуються в спеціальному редакторі MatLab *Editor*. Цей редактор можна відкрити, обравши вкладку *HOME* – групу *FILE – New – Function*.

Структура m-файлу функції з одним вихідним параметром є наступною:

```
function var=fname(список_параметрів)
% коментар
```

```
% тіло файлу з будь-якими виразами
var=вираз % var – ім'я змінної вихідного параметра
end
```

де *var=вираз* — конструкція, яка вводиться, якщо необхідно, щоб функція повертала результат обчислень.

Функція повертає своє значення і може використовуватися в математичних виразах у виді: *fname(список_параметрів)*.

Структура m-файлу функції з двома вихідними параметрами *var1* та *var2* більше схожа на процедуру й повертає не один результат, а декілька (у виразах застосовувати складніше):

```
function [var1, var2]=fname(список_параметрів)
% тіло файлу з будь-якими виразами
var1=вираз % var – ім'я змінної 1-го вихідного параметра
var2=вираз % var – ім'я змінної 2-го вихідного параметра
end
```

Якщо така функція використовується у вигляді *fname(список_параметрів)*, то повертається значення тільки першого вихідного параметра – змінної *var1*.

Завдання 2. Ознайомитися з можливостями MatLab зі створення файлу-сценарію, візуалізації даних за допомогою графіків та теплових карт, визначенню власних функцій.

1.2.2. Побудова графіків

1. Для побудови графіків у MatLab можна скористатися функцією *plot()*, яка може мати різні аргументи. Введіть команди для побудови синусоїди зеленого кольору, задавши діапазон виведення:

```
>> y=0:0.05:4;  plot(y,sin(y),'g');   grid on
```

2. Побудований графік буде виведено у окремому вікні (рис. 1.5).

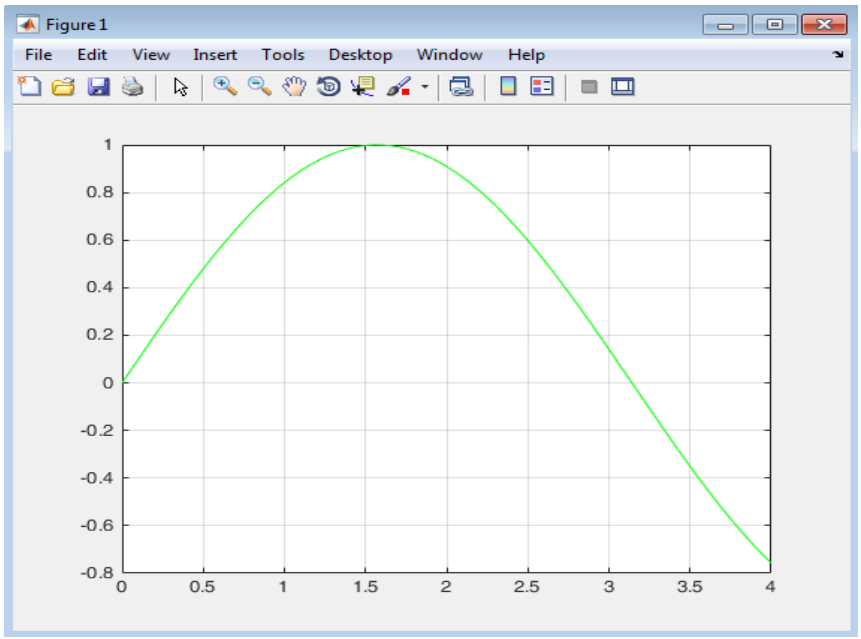


Рис. 1.5. Вікно з відображенням побудованого в MatLab графіка

1.2.3. Створення файлу-сценарію

1.Script-файли створюються, редагуються і налагоджуються в спеціальному редакторі MatLab *Editor*. Цей редактор можна відкрити, обравши вкладку *HOME* – групу *FILE – New – Script*.

2. У вікні редактора MatLab *Editor*, яке буде відкрито, вводимо команди сценарію:

```
% plot with color red
% будує графік косинусоїди лінією червоного кольору
% з виведеною масштабною сіткою в інтервалі [0, 12]
x=0: 0.1: 12;
```

```
plot(x, cos(x),'r')
grid on
```

3. Для збереження файлу-сценарію необхідно обрати: вкладка *Editor* – група *File* – команда *Save* та вказати місце збереження й ім'я m-файлу. Виконайте збереження файлу-сценарію у файлі з іменем *ScriptGraph.m* на диску *N:* у папці *DataMining\MatLab*.

4. У вікні *Command Window* введіть команди знищення змінних у робочій області та очищення екрану:

```
>> clear
>> clc
```

5. Скористайтесь створеним Script-файлом. Для виконання команд цього Script-файлу необхідно перетягнути при натисненій лівій кнопці миші ім'я цього файлу з вікна *Current Folder* у вікно *Command Window* або ввести у вікні *Command Window* команду:

```
>> ScriptGraph
```

Будуть виконані усі команди, остання – побудова графіка, який буде виведено у окремому вікні.

6. Двічі клацніть лівою кнопкою миші ім'я Script-файлу *ScriptGraph.m* у вікні *Current Folder* для його відкриття у вікні редактора, та відредагуйте файл, змінивши червоний колір косинусоїди на синій. Для цього необхідно змінити параметр функції побудови графіка таким чином:

```
>> plot(x, cos(x),'b')
```

7. Збережіть внесені зміни, натиснувши у групі *File* вкладки *Editor* кнопку 

, та виконайте команди зміненого скрипта, натиснувши кнопку 

у групі *Run* вкладки *Editor*. У окремому вікні буде виведено косинусоїду синього кольору відповідно до зроблених налаштувань (рис. 1.6).

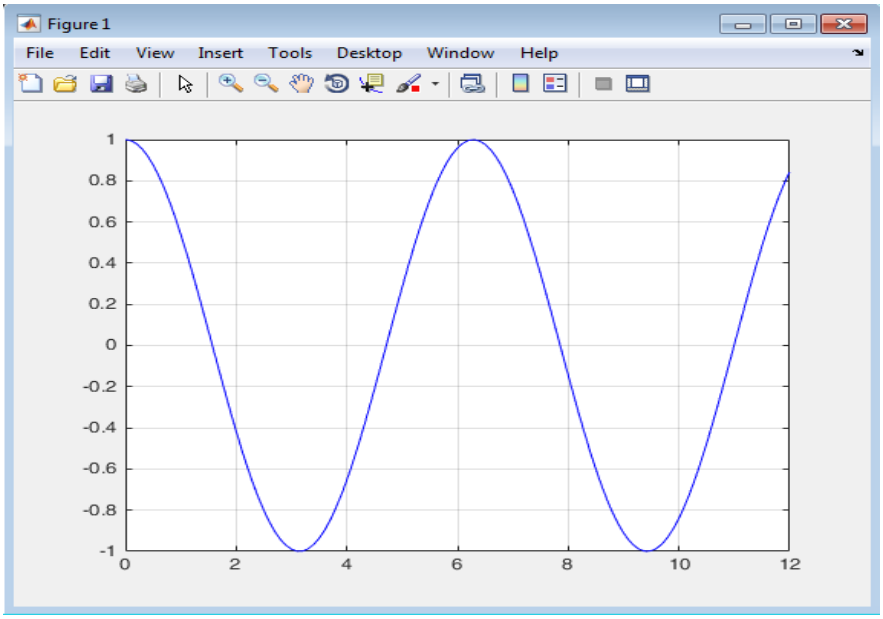


Рис. 1.6. Вікно з відображенням графіка, відредагованого у Script-файлі

1.2.4. Визначення власних функцій користувача

1. У m-файлі з назвою *fun.m* визначимо функцію fun(x) = 1 + x - $x^2/4$, ввівши у вікні редактора вираз для обчислення її значень (рис. 1.7).

2. Збережемо функцію на диску *N:* у папці *Data Mining\MatLab* з іменем *fun.m*. Ім'я файлу повинно співпадати з іменем функції, яка у файлі записується.

3. Перевіримо роботу функції, ввівши у вікні *Command Window* команди:

```
>> fun(3)
```

```
ans =
   1.7500
>> k = 25+6*fun(5);
>> k
k =
   23.5000
```

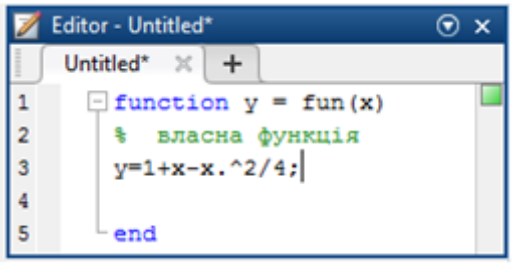


Рис. 1.7. Вікно редактора MatLab Editor з функцією fun.m

1.2.5. Побудова теплових карт

Теплова карта (англ. heatmap) є графічним представленням даних із допомогою кольорової палітри.

Теплові карти знаходять широке застосування під час аналізу даних різних предметних сфер: у Web-аналітиці при вивченні поведінки користувачів, у бізнесі при вивченні попиту на певні товари, у геоаналітиці тощо.

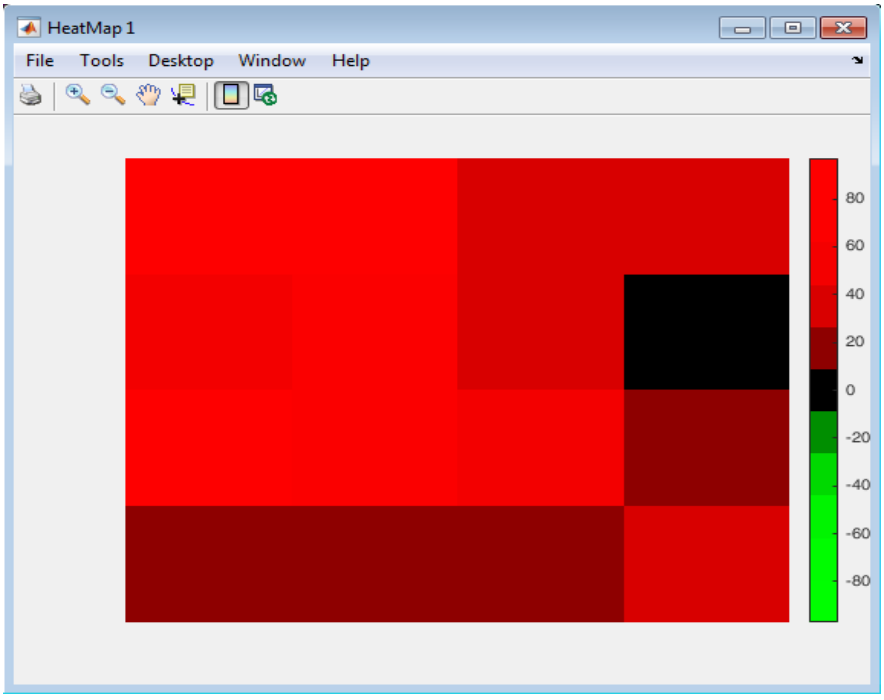
Для візуалізації даних за допомогою теплових карт у MatLab можна скористатися функцією *HeatMap()*, яка може мати різні аргументи.

Приклад 19. Побудувати теплову карту в середовищі MatLab.

1. Введіть команди для створення теплової карти даних, представлених у вигляді матриці:

```
>> CData = [10 15 25 40; 87 64 55 12; 45 66 31 5; 97 86 42 30];
>> h = HeatMap(CData);
```

2. У окремому вікні буде виведено теплову карту з палітрою кольорів, заданою по замовчуванню. Для відображення шкали кольорів у вікні із зображенням теплової карти необхідно обрати меню *Tools – Insert Colorbar* (рис. 1.8).



(*палітра кольорів задана по замовчуванню*)

Рис. 1.8. Графічне зображення теплової карти

4. Змінимо кольорову палітру теплової карти, скориставшись параметром *Colormap* та відобразимо числові значення у комірках теплової карти, ввівши у вікні *Command Window* команди (рис. 1.9):

```
>> h.Colormap = winter;
>> h.Annotate = true;
```

5. Введіть команди для побудови теплових карт з різними кольоровими палітрами:

```
>> h1 = HeatMap(CData,'Colormap',summer);
>> h2 = HeatMap(CData,'Colormap',autumn);
>> h3 = HeatMap(CData,'Colormap',redbluecmap);
>> h4 = HeatMap(CData,'Colormap',parula);
```

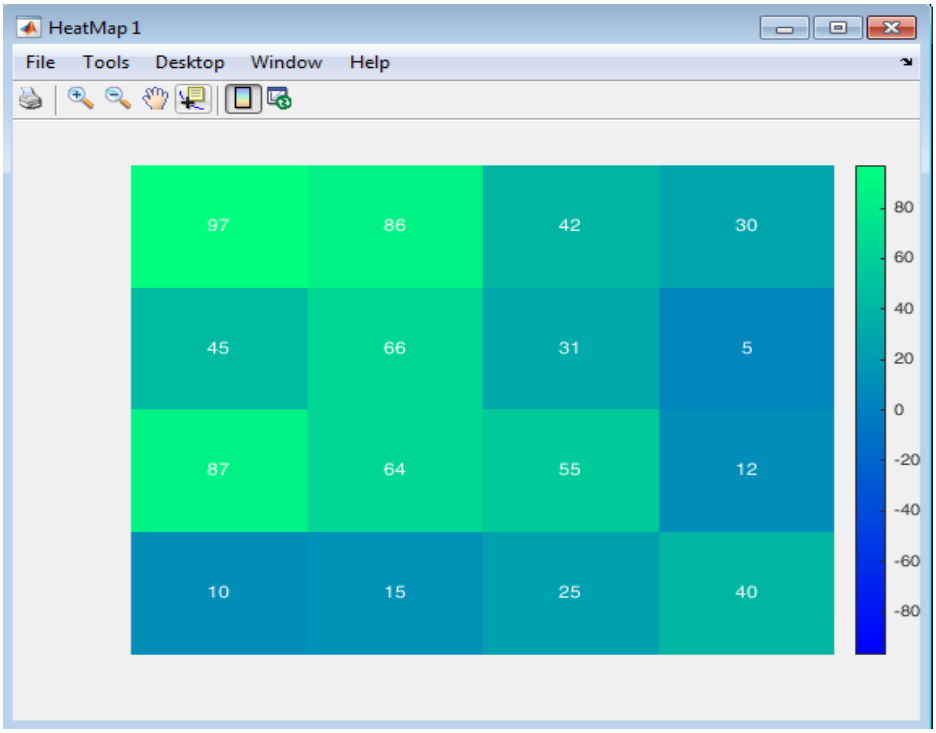


Рис. 1.9. Графічне зображення теплової карти зі зміненою палітрою

Завдання 3. Ознайомитися з можливостями MatLab з програмування основних алгоритмічних елементів.

1.2.6. Умовне виконання інструкцій

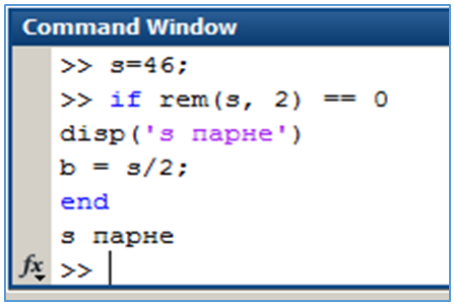
Оператор умовного виконання може бути представлений наступними варіантами:

```
if Логічний_вираз
   Виконувані оператори
end
```

```
if Логічний_вираз
  Виконувані оператори
else
  Виконувані оператори
end
```

```
if Логічний_вираз
  Виконувані оператори
elseif Логічний_вираз
  Виконувані оператори
else
  Виконувані оператори
end
```

На рисунку 1.10 наведено приклади застосування інструкцій умовного виконання команд. Виконання команд на рисунку 1.10 *а* буде виводити на екран фразу 's парне' у випадку, якщо s буде парним числом. Виконання команд на рисунку 1.10 *б* буде присвоювати значення матриці A в залежності від значення змінної k.

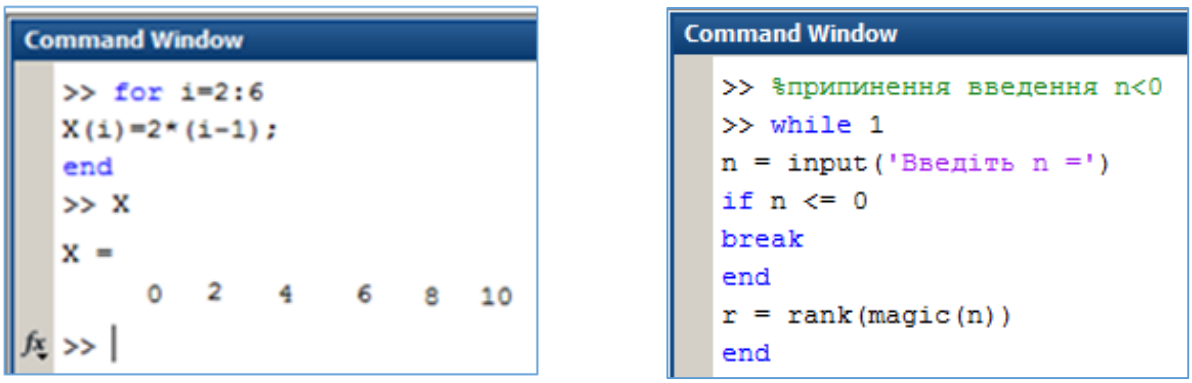


а) *б)*

Рис. 1.10. Вікно Command Window з прикладами застосування операторів умовного виконання інструкцій

1.2.7. Цикли

Оператори ***for*** і ***while*** установлені в MatLab аналогічно їх прототипам в інших мовах програмування (рис. 1.11, рис. 1.12, рис. 1.13).



а) оператор for *б) оператор while*

Рис. 1.11. Використання операторів циклу

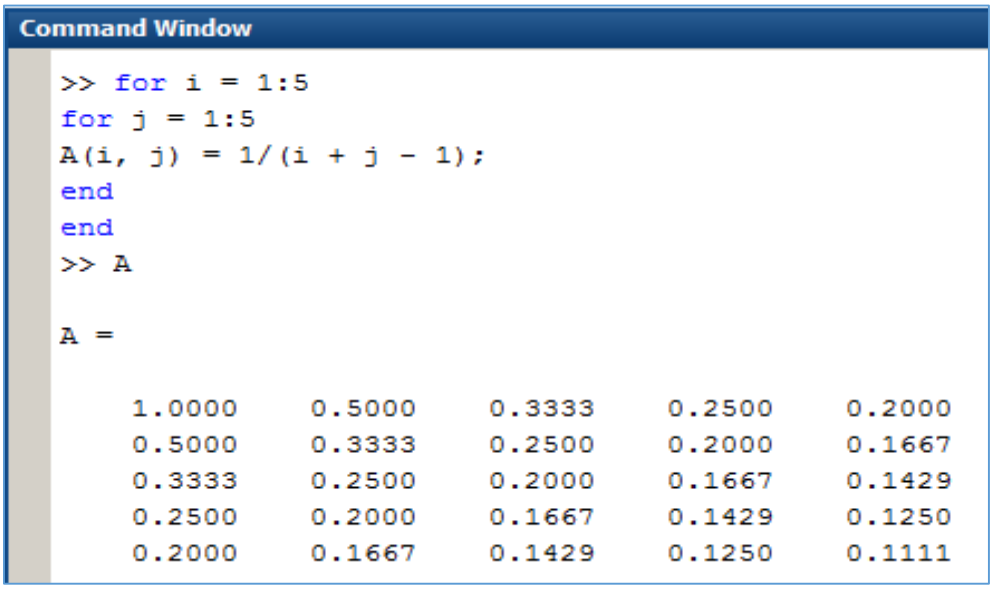


Рис. 1.12. Використання вкладених операторів циклу for

```
Command Window
>> r=1;
>> % Обчислення довжини кола з діалоговим
while r >= 0,
r=input('Введіть радіус кола r=');
if r>= 0
disp('Довжина кола l=');
disp(2*pi*r);
end
end
Введіть радіус кола r=5
Довжина кола l=
   31.4159
Введіть радіус кола r=-1
>>
```

Рис. 1.13. Використання вкладених операторів while та if

Для здійснення множинного вибору (або розгалуження) використовується конструкція з перемикачем типу ***switch*** (рис. 1.14).

```
Command Window
>> var=5;
>> switch var
case {1,2,3}
disp('Перший квартал')
case {4,5,6}
disp('Другий квартал')
case {7,8,9}
disp('Третій квартал')
case {10,11,12}
disp('Четвертий квартал')
otherwise
disp('Помилка в завданні')
end
Другий квартал
>>
```

Рис. 1.14. Використання конструкції з перемикачем типу switch

1.3. ОБРОБКА ДАНИХ ЗАСОБАМИ MATLAB: НОРМАЛІЗАЦІЯ, СТАНДАРТИЗАЦІЯ

При розв’язанні задач аналізу даних доводиться мати справу з перетворенням даних у процесі їх підготовки для подальшого аналізу.

Попередня обробка й перетворення даних часто здійснюється з використанням статистичних показників, які їх характеризують. Розрахунок показників описової статистики дає можливість отримати узагальнені характеристики набору даних за певною змінною – міри центральної тенденції, та міри мінливості. Наприклад, для змінної x: $x_1, x_2, \ldots, x_n$, де n – кількість об’єктів, ці показники наведені у таблиці 1.1.

Нормалізація даних дозволяє перетворювати один діапазон зміни значень числової ознаки в інший діапазон, більш зручний для застосування до даних алгоритмів інтелектуального аналізу (наприклад, від 0 до 1), а також для погодження діапазонів змін різних ознак.

Стандартизація даних передбачає таке перетворення даних, після якого кожна ознака має середнє рівне 0 та дисперсію рівну 1.

Існує багато підходів до нормалізації даних. Одним із них є лінійна *min-max* нормалізація з використанням формули:

$$x_i^{'} = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}},$$

де x_i' – нормалізоване значення змінної x_i, x_{max}, x_{min} – максимальне та мінімальне значення по усій множині значень змінної x: $x_1, x_2, \ldots, x_n$, де n – кількість об'єктів. Дана формула розміщує усі значення в інтервалі [0, 1].

Таблиця 1.1

Деякі статистичні показники змінної x набору даних

№ з/п	Показники		Визначення показників
1	Міри центральної тенденції (характеристики положення)	Середнє значення (середнє арифметичне, чутливе до екстремальних значень змінної)	$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$
2		Медіана (не чутлива до екстремальних значень змінної)	Значення x_i, зліва та справа від якого у ранжованому ряді значень змінної x знаходиться однакова кількість елементів
3		Мода (чутлива не до самих екстремальних значень змінної, а до їх кількості)	Значення змінної x, яке зустрічається найчастіше
4	Міри мінливості (розкиду)	Дисперсія (міра відхилення значень змінної від її середнього значення)	$D_x = \sigma_x^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$
5		Стандартне (середньоквадратичне) відхилення (корінь квадратний із дисперсії)	$\sigma_x = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2}$
6		x_{min}	Мінімальне значення змінної x
7		x_{max}	Максимальне значення змінної x

Різні види стандартизації та нормалізації передбачають перетворення даних по-різному: зі значень змінних вираховується їх середнє і ці значення діляться на стандартне відхилення (z-стандартизація), лінійним перетворенням добиваються розкиду змінних в інтервалі [-1, 1], [0, 1], значення змінних ділять на середнє, максимум або стандартне відхилення (додаток *Є*).

Перетворення даних у MatLab виконують із використанням функцій статистичної обробки даних (додаток *Ж*) та функцій для роботи з матрицями (додаток *Е*).

Приклад 20. Здійснити обчислення середніх значень матриці (рис. 1.15).

```
Command Window
>> M
M =
     1     2     3
     4     5     6
     7     8     9
>> Y=mean(M)
Y =
     4     5     6
>> N=mean(M,2)
N =
     2
     5
     8
fx >>
```

Рис. 1.15. Приклад обчислення середніх значень матриці

Приклад 21. Здійснити нормалізацію вектора з використанням функції *norm()* (обчислює норму вектора), ввівши наведені нижче команди:

```
>> M=[1 2 3; 4 2 1; 2 8 3];
>> M
M =
   1   2   3
   4   2   1
   2   8   3
>> Y=M/norm(M)
Y =
  0.1010   0.2020   0.3030
  0.4040   0.2020   0.1010
  0.2020   0.8081   0.3030
```

Приклад 22. Здійснити нормалізацію вектора за формулою лінійної *min-max* нормалізації:

```
>> L=[2 3 8 4 2 3 5]
>> K=(L-min(L))/(max(L)-min(L))
K =
      0   0.1667   1.0000   0.3333   0   0.1667   0.5000
```

Приклад 23. Здійснити Z-стандартизацію вектора:

```
>> L=[2 3 8 4 2 3 5]
>> D=(L-mean(L))/std(L)
D =
  -0.8778  -0.4051   1.9581   0.0675  -0.8778  -0.4051   0.5402
```

1.4. ЗАВДАННЯ ДЛЯ САМОСТІЙНОЇ РОБОТИ

Завдання 4. У відповідності до отриманого варіанта обрати значення набору даних із таблиці 1.2 та виконати наведені нижче завдання.

1. Створити власну функцію для здійснення min-max нормалізації даних за формулою 1 (додаток *Є*), яка в якості аргументів повинна мати межі діапазону [A, B], в якому повинні бути представлені нормалізовані дані. Функція повинна завантажувати з mat-файлу дані та зберігати у mat-файлі вектор нормалізованих значень.
2. Створити функцію для здійснення z-стандартизації даних (формула 3 додатку *Є*), яка зберігає у mat-файлі вектор нормалізованих значень.
3. Створити файл-сценарій здійснення нормалізації та стандартизації даних з використанням створених функцій та формул 2, 4, 5 (додаток *Є*). Порівняти значення, отримані за допомогою різних формул.
4. Здійснити візуалізацію нормалізованих та стандартизованих значень, представивши їх різними символами на одному полі (функція plot(), детальніше див. додаток *З*).
5. Здійснити візуалізацію вхідного набору даних у вигляді теплової карти, кольорову палітру якої налаштувати за самостійним вибором.
6. При виконанні завдань повинні бути використані управляючі структури мови програмування MatLab (умовний оператор, цикли).

КОНТРОЛЬНІ ПИТАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ №1

1. Призначення MatLab, основні елементи робочого середовища.
2. Розрахунки з обчислення математичних виразів, оператори MatLab та їх пріоритет.
3. Створення та використання mat-файлів.
4. Робота з матрицями в середовищі MatLab.
5. Створення та використання m-файлів-сценаріїв та m-файлів-функцій.
6. Побудова графіків, теплових карт, управляючі структури мови програмування MatLab.
7. Нормалізація та стандартизація даних у MatLab.

Таблиця 1.2

Варіанти значень набору даних для виконання завдання 4

№ варіанта	Вхідні дані																			
1	18	51	43	13	30	25	39	12	41	20	43	42	23	16	24	18	29	10	30	27
2	62	76	52	58	46	51	77	72	44	65	70	56	54	60	62	21	64	27	43	70
3	41	45	44	43	48	50	46	49	44	45	50	44	46	46	44	44	43	46	47	49
4	48	68	57	30	47	37	74	37	64	48	29	32	65	65	67	52	66	47	70	35
5	26	23	32	32	34	17	19	29	18	21	21	28	34	31	18	30	18	15	35	31
6	83	94	27	42	65	91	92	98	48	19	57	79	18	93	73	20	97	28	33	73
7	84	94	84	86	83	81	91	90	89	92	86	84	84	92	91	92	92	86	85	85
8	65	44	53	63	67	46	42	62	67	63	49	59	65	53	60	58	61	59	65	65
9	96	82	83	81	87	97	87	94	90	88	94	94	91	95	86	90	87	86	81	91
10	12	27	13	22	11	13	12	23	30	11	21	10	14	14	29	14	16	30	21	15
11	36	33	38	43	25	25	31	41	20	37	36	21	37	34	42	33	44	41	28	25
12	45	27	56	42	46	16	18	31	51	41	20	16	18	43	59	58	30	51	40	48
13	44	44	36	75	72	70	56	64	43	57	64	25	77	29	73	26	55	79	22	23
14	44	43	48	43	41	43	48	48	49	47	46	42	49	45	40	41	45	42	41	43
15	29	31	35	67	54	72	63	52	46	69	38	75	30	55	58	37	48	58	67	45
16	20	27	29	17	17	29	30	34	30	29	16	34	16	15	30	28	18	33	21	30
17	64	65	66	80	46	91	72	64	70	70	51	96	75	66	62	62	62	91	66	89
18	84	85	87	88	93	82	88	84	81	89	84	86	80	90	89	84	89	82	89	84
19	61	55	66	58	66	44	67	64	52	59	54	67	40	57	42	43	43	69	48	41
20	97	82	94	87	94	95	84	86	86	95	97	91	87	91	85	92	98	84	95	95
21	28	19	16	28	23	12	26	23	26	25	27	20	12	16	14	29	26	15	27	20
22	21	24	39	39	26	41	21	43	23	28	25	40	26	21	34	28	33	36	45	42
23	87	95	89	72	94	71	78	63	91	76	65	51	77	65	50	98	77	57	92	89
24	83	82	89	83	85	80	87	92	88	85	81	91	85	94	94	84	86	85	83	88
25	59	52	61	64	63	42	53	53	43	57	45	55	64	51	45	51	50	44	61	50

2. НАБОРИ ДАНИХ. ШКАЛИ. ПОПЕРЕДНЯ ОБРОБКА ДАНИХ

Лабораторна робота № 2

Мета: закріплення знань про проведення попередньої обробки даних. Формування умінь та навичок очищення набору даних, кодування числових та якісних ознак, здійснення перетворення даних та генерації ознак.

Теоретичні знання: набори даних та їх атрибути. Типи даних. Шкали категоріальних та числових даних. Data Preparation: основні завдання попередньої обробки даних. Очищення даних. Способи обробки дублікатів, суперечливих, помилкових та екстремальних даних. Перетворення даних: нормалізація та стандартизація, дискретизація, агрегація, dummy-кодування.

2.1. НАБОРИ ДАНИХ ТА ШКАЛИ

Набір даних (англ. *Data Set*) є необробленим матеріалом і повинен бути представлений у формі, придатній для зберігання, передачі, обробки та подальшого аналізу з використанням методів Data Mining.

Набір даних зручно представити у вигляді двомірної таблиці, у якій по горизонталі розміщуються атрибути об'єкта – його ознаки, а по вертикалі – об'єкти (табл. 2.1).

Об'єкт – набір атрибутів, представлений як рядок таблиці.

Атрибут – властивість, що характеризує об'єкт. Його також називають *ознакою*, *характеристикою* об'єкта.

Таблиця 2.1

Набір даних, представлений у вигляді двомірної таблиці «об'єкт – атрибут»

Об'єкти	Атрибути				
	Код клієнта	Вік	Сімейний стан	Дохід	Клас
	1	18	Одинокий	125	1
	2	22	Одружений	100	1
	3	30	Одинокий	70	1
	4	32	Одружений	120	1
	5	24	Вдівець	95	2
	6	25	Одружений	60	1
	7	32	Вдівець	220	1
	8	19	Одинокий	85	2
	9	22	Одружений	75	1
	10	40	Одинокий	90	2

Переходячи від конкретних до загальних величин, отримуємо *набір змінних*, який характеризує об'єкти досліджуваного набору даних.

Змінна – ознака або характеристика, загальна для усіх об'єктів, що вивчаються, прояв якої змінюється від об'єкта до об'єкта.

Значення змінної є проявом ознаки (характеристики). Значення змінних можуть бути ***числовими*** або ***категоріальними*** (*якісними*). Числові значення можуть бути неперервними та дискретними.

Терміни «атрибут» (англ. *attribute*), «вимір» (англ. *dimension*), «ознака» (англ. *feature*), «змінна» (англ. *variable*) у літературі часто використовують як взаємозамінні. Фахівці з Data Mining зазвичай використовують термін «атрибут» або «ознака».

У процесі підготовки даних здійснюється вимірювання характеристик об'єктів, що досліджуються, у певній шкалі.

Шкала – правило, відповідно до якого характеристикам об'єктів привласнюються певні значення. Типи шкал представлено у таблиці 2.2.

Окремо виділяють також ***дихотомічну шкалу*** – номінальну, яка містить тільки дві категорії, і відповідні їм бінарні атрибути: 0 або 1, «так» або «ні», «істинне» або «хибне». ***Бінарні атрибути*** можуть бути:

a) *симетричними* – для них рівноцінним є кожне значення (наприклад, стать: чоловіча та жіноча);

b) *асиметричними* – для них важливість станів є різною (наприклад, результати медичного тесту: позитивний тест є більш значимим, ніж негативний).

Таблиця 2.2

Типи шкал

Тип шкали		Допустимі операції (математичні, логічні)				Приклади	Особливості
		= ≠	> <	+ -	* /		
Категоріальні	Номінальна	✓	–	–	–	Стать, національність, клінічний діагноз	Дані не можуть упорядковуватися
	Порядкова	✓	✓	–	–	Оцінка, посада, військовий ранг	Порівняння не числове: визначають порядок слідування
Числові (метричні)	Інтервальна	✓	✓	✓	–	Календарний час, температура за Цельсієм	Нуль не означає відсутність ознаки
	Відношень	✓	✓	✓	✓	Вага, ціна, ріст, температура за Кельвіном	Нуль означає відсутність ознаки

У таблиці 2.3 наведено приклади використання різних шкал.

Таблиця 2.3

Приклад використання різних шкал для множини ознак різних об’єктів

№ об’єкта	Професія (*номінальна шкала*)	Середній бал (*інтервальна шкала*)	Освіта (*порядкова шкала*)
1	кур’єр	22	середня
2	вчений	55	вища
3	учитель	47	вища

2.2. DATA PREPARATION

2.2.1. Основні завдання Data Preparation

Підготовка даних (англ. *Data Preparation*) включає етапи попередньої обробки даних, які є виключно важливими в аналізі даних та займають до 80% усіх витрат ресурсів і часу в життєвому циклі при розробці проєкту Data Mining.

Реальні дані можуть бути:

a) неповними: якщо відсутні деякі атрибути чи їх значення;

b) неузгодженими: містити невідповідності в іменах, кодах чи значеннях;

c) містити шуми та викиди.

Викиди – аномальні значення, які різко відрізняються від загальної вибірки даних.

Шум – відхилення від середніх значень даних, яке не несе ніякої корисної інформації, у тому числі прихованої.

Data Preparation включає наступні завдання обробки вхідних («сирих») даних:

1. ***Вибірка даних*** – відбір об’єктів та їх ознак (*features* – предикторів, атрибутів) для цілей Data Mining.
2. ***Інтеграція даних*** – злиття даних з різних джерел.
3. ***Очищення даних*** – відновлення цілісності, логічних зв’язків, унікальності даних.
4. ***Перетворення даних*** – нормалізація, стандартизація, дискретизація, масштабування, бінаризація.
5. ***Генерація ознак*** – створення похідних ознак і їх перетворення у числові вектори, придатні для застосування алгоритмів Data Mining.
6. ***Скорочення даних*** – скорочення простору ознак шляхом відсікання непотрібних ознак та їх об’єднання.
7. ***Форматування даних*** – синтаксичні зміни, які змінюють не значення даних, а їх представлення (сортування, видалення непотрібних знаків, округлення чисел).

2.2.2. Очищення даних

Очищення даних (англ. *Data cleaning*) полягає у перевірці зібраних даних, виявлені та видаленні помилок з метою покращення якості даних.

Перевірка даних на стадії очищення дозволяє виявляти:

1. *Недопустимі дані*, що виходять за межі певного діапазону: їх виправляють.
2. Логічно непослідовні, *суперечливі, помилкові дані*. Способи їх обробки: 1) виключити з аналізу; 2) виправити.
3. Екстремальні значення, *шуми та викиди*. Способи їх обробки: 1) для відсікання шуму використовують спектральний аналіз, ауторегресійні методи при аналізі часових рядів; 2) аномальні значення можна відсікати або міняти на найближчі граничні; 3) для виявлення викидів, їх розраховують за певними алгоритмами.
4. *Пропуски в даних*. Способи їх обробки: 1) пропущені значення можна ігнорувати; 2) розрахувати нові значення; 3) виключити з аналізу; 4) замінити на можливі значення, визначені шляхом апроксимації чи розрахунку найбільш ймовірного значення.
5. *Дублювання даних*. Способи їх обробки: 1) видаляється уся група даних як недостовірна; 2) група дублікатів замінюється на один унікальний запис.

Процедура очищення передбачає:

1) проведення аналізу даних;
2) визначення порядку та правил перетворення даних;
3) підтвердження правильності обраних правил перетворення;
4) заміну даних на очищені.

2.2.3. Обробка викидів

Як шукати викиди? Як установити, наприклад, що дохід у 900 000 – викид і його слід відфільтрувати. Для цього слід установити наступне:

1. *Перший квартиль* Q_1: таке значення змінної, що рівно 25% об'єктів мають значення, які лежать лівіше від нього у ранжованому ряду значень цієї змінної.
2. *Третій квартиль* Q_3: таке значення змінної, що рівно 75% об'єктів мають значення, які лежать лівіше від нього у ранжованому ряду значень цієї змінної.
3. *Інтерквартильний розмах* $IQR = Q_3 - Q_1$ – це є міра розкиду значень змінної, чимось схожа на дисперсію.

Викиди лежать за межами інтервалу:

$$[Q_1 - 1.5 * IQR;\ Q_3 + 1.5 * IQR]. \qquad (2.1)$$

Такий підхід є евристикою, обґрунтувань якісних немає, але його часто використовують на практиці.

2.3. ПРИКЛАДИ ПОПЕРЕДНЬОЇ ОБРОБКИ ТА ПЕРЕТВОРЕННЯ ДАНИХ

2.3.1. Виявлення та обробка проблемних даних

Приклад 1. Маємо набір даних з інформацією про суб'єкти кредитування (табл. 2.4). Необхідно виявити проблемні дані.

Таблиця 2.4

Задача кредитного скорінгу

Стать	Вік	Місто	Дохід	Освіта	Повернув?
Ч	25	Київ	15000	A	1
Ж	31	Київ	9000000	C	1
Ч	10	Київ	8760	C	0
A	28	Миколаїв	7800	B	0
Ч	23	?	6550	A	1
Ж	30	Одеса	16782	B	1

Здійснивши аналіз набору даних, представленого у вигляді таблиці, можна виявити помилкові та суперечливі дані:

1. A, 10 – помилки статі та віку (статі A немає, кредит не могли дати неповнолітньому).
2. Відсутня інформація про місто проживання одного об'єкта.
3. У стовпці «Дохід» два значення викликають питання – дуже велике (або помилка, або викид – реальне значення, дуже відрізняється від інших) та занадто детально вказане (усі інші округлені).
4. Не зрозуміло значення у стовпці освіта (їх потрібно дізнатися).

Приклад 2. Маємо набір даних з інформацією про суб'єкти кредитування, який містить пропуски (табл. 2.5). Необхідно обрати спосіб обробки пропущених значень.

Таблиця 2.5

Задача кредитного скорінгу

Стать	Вік	Місто	Дохід	Освіта	Повернув?
Ч	25	Київ	15000	A	1
Ж	31	Київ	?	C	1
Ч	20	?	8760	C	0
?	28	Миколаїв	7800	B	0
Ч	23	?	6550	A	1
Ж	30	Одеса	16782	B	1

Пропуски у наборі даних можна замінити на:

1) середній дохід по усій вибірці даних: 10978,4;

2) ознаки «Стать» та «Місто» категоріальні, для них використовують інший підхід: обирають місто та стать, які найбільш часто зустрічаються: Київ та Ч (чоловіча).

2.3.2. Здійснення нормалізації та стандартизації даних

Дані для аналізу можуть бути різнотипними, тому виникає необхідність їх перетворення шляхом приведення до однієї шкали, одного числового діапазону для подальшого аналізу. Вирішити поставлену задачу дозволяє нормалізація та стандартизація даних.

Основні формули для здійснення нормалізації та стандартизації (додаток Є) було розглянуто у лабораторній роботі № 1.

Приклад 3. Маємо набір даних з інформацією про розмір заробітної плати. Змінна «зарплата» приймає наступні значення (грн): 3000, 3600, 4700, 5000, 5200, 5200, 5600, 6000, 6300, 7000, 7020, 11000. Необхідно здійснити min-max нормалізацію.

$x_{\min} = 3000$, $x_{\max} = 11000$, A = 0, B = 1

$$x_i' = \frac{x_i - 3000}{11000 - 3000} = \frac{x_i - 3000}{8000}, \quad x_3' = \frac{4700 - 3000}{8000} = 0{,}213.$$

Приклад 4. Маємо набір даних з інформацією про розмір заробітної плати. Змінна «зарплата» приймає наступні значення (грн): 3000, 3600, 4700, 5000, 5200, 5200, 5600, 6000, 6300, 7000, 7020, 11000. Необхідно здійснити z-нормалізацію.

$x_{\min} = 3000$, $x_{\max} = 11000$, $\bar{x} = 5427$, $\sigma_x \approx 2438$

$$x_i' = \frac{x_i - 5427}{2438}, \quad x_3' = \frac{4700 - 5427}{2438} \approx -0{,}298.$$

Приклад 5. Маємо набір даних з інформацією про розмір заробітної плати. Змінна «зарплата» приймає наступні значення (грн): 3000, 3600, 4700, 5000, 5200, 5200, 5600, 6000, 6300, 7000, 7020, 11000. Необхідно провести масштабування таким чином, щоб значення зміннної були на проміжку [-1, 1**]**.

$$\lambda = 10^{-p}, \quad \max_{i=1..n}(|x_i|) = 11000 \Rightarrow \frac{\max_{i=1...n}(|x_i|)}{10^5} = 0{,}11 < 1 \Rightarrow p = 5$$

$$x_i' = \frac{x_i}{10^5}, \qquad x_3' = \tau(4700) = \frac{4700}{10^5} = 0{,}047.$$

2.3.3. Дискретизація даних

Дискретизація числової ознаки – заміна початкових значень на інтервальні або концептуальні мітки. Дозволяє неперервні дані перетворити у дискретні. У процесі дискретизації здійснюється перетворення неперервного числового значення змінної у категоріальне значення – якісне, вимірюване у порядковій шкалі.

Приклад 6. Необхідно здійснити дискретизацію значень змінної «Вік».

Значення ознаки «Вік» можна замінити на:

a) *інтервальні мітки*: 0-10, 11-20, …;

b) *концептуальні мітки*: *молодий, дорослий, старий*;

c) мітки можуть бути об'єднані в поняття більш високого рівня, визначаючи ієрархію понять числового атрибуту (рис. 2.1).

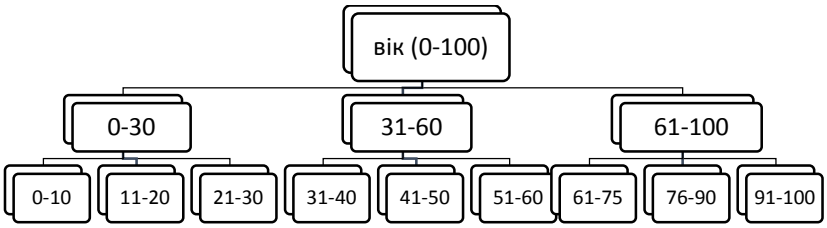


Рис. 2.1. Приклад дискредидації значень змінної «Вік»

2.3.4. Dummy-кодування

Одним із популярних підходів при роботі з категоріальними ознаками є ***Dummy-кодування***, для якого характерно наступне:

1) ознака x категоріальна й приймає значення з множини $U = \{u_1, u_2, ..., u_m\}$;

2) створимо m нових бінарних ознак-індикаторів, які поставимо у відповідність елементам множини U таким чином, що кожному u_j буде відповідати вектор $(u_{j1}, .. u_{jk}., u_{jm})$, у якого усі елементи будуть рівними нулю, крім елемента u_{jk}, який буде рівним одиниці й відповідатиме значенню u_k із множини U значень змінної x.

При цьому k-й індикатор показує, чи дорівнює ця ознака у даного об'єкта значенню u_k.

Приклад 7. Необхідно здійснити dummy-кодування змінної, яка кодує міста. Нехай при цьому множина значень змінної містить усього три міста: U = {Київ, Харків, Одеса}.

Тоді їх можна закодувати трьома бінарними ознаками:

Київ → (1,0,0) (буде закодовано вектором (1 0 0))

Харків → (0,1,0)

Одеса → (0,0,1)

Процес перетворення категоріального атрибуту у набір двійкових атрибутів називається ***бінаризацією***. Часто виконують перетворення неперервного атрибуту у категоріальний, а потім здійснюють його бінаризацію.

Приклади подібних перетворень:

a) колір очей: перетворюють у значення {голубі, чорні, коричневі, сірі}, а потім бінаризують;

b) зріст перетворюють у значення {високий, середній, низький}, а потім бінаризують.

Розглянемо проблеми dummy-кодування.

Приклад 8. Маємо набір даних із змінною «*Місто*». У наборі значень цієї змінної одне місто зустрічається тільки один раз, значить одна з кодуючих ознак прийме значення 1 тільки на одному об'єкті – ця ознака не має змісту.

Для вирішення цієї проблеми значення ознак, що рідко зустрічаються, можна під час dummy-кодування об'єднувати у одну категорію: усі міста, які зустрічаються рідко, оголошують одним містом.

Приклад 9. Потрібно передбачити, чи клікне користувач на рекламний банер. Ця задача має 4 ознаки:

a) ідентифікатор користувача;

b) ідентифікатор банера;

c) ідентифікатор сайту, на якому показано банер;

d) ідентифікатор категорії банера.

Це категоріальні ознаки, які можуть приймати багато різних значень. При використанні dummy-кодування буде отримано мільйон ознак-індикаторів. Це буде дуже велика вибірка, з якою буде важко працювати у подальшому при застосуванні алгоритмів Data Mining.

Для вирішення проблеми можна скористатися *лічильниками*. Ідея їх застосування полягає у наступному:

1) нехай по банеру u_1 клікають частіше, ніж по банеру u_2;

2) замінимо категорії на ймовірності кліків.

Описаний вище спосіб можна формалізувати. У випадку, якщо поставлена задача класифікації, оцінюють ймовірність одного класу за умови певного значення змінної.

Приклад 10. Необхідно здійснити кодування ознаки, яка кодує міста.

Нехай є 7 об'єктів, кожен з яких відноситься до класу 0 або класу 1 (табл. 2.6). Та одна категоріальна ознака – «місто».

Оцінимо ймовірності значень категоріальної змінної «місто» для класу 1:

$p(y = 1|\text{Київ}) = 2/3 = 0{,}67 \quad p(y = 1|\text{Харків}) = 0/2 = 0 \quad p(y = 1|\text{Одеса}) = 2/2 = 1$

Таблиця 2.6

Міста та класи, до яких вони відносяться

Ознака	Об'єкти з набору даних						
	x_1	x_2	x_3	x_4	x_5	x_6	x_7
Місто	Київ	Одеса	Київ	Харків	Харків	Київ	Одеса
Клас	1	1	0	0	0	1	1

Далі робимо заміну кожного міста на зроблені оцінки ймовірностей (табл. 2.7).

Таблиця 2.7

Кодування ознаки, що кодує міста

Ознака	Об'єкти з набору даних						
	x_1	x_2	x_3	x_4	x_5	x_6	x_7
Місто	0,67	1	0,33	0	0	0,67	1
Клас	1	1	0	0	0	1	1

2.3.5. Побудова ієрархії понять, агрегація даних

Номінальні дані можуть приймати скінчену кількість значень без відношень порядку (*наприклад:* вулиця).

Здійснивши узагальнення атрибутів більш загальними поняттями, можна реалізувати ***побудову ієрархії*** понять (*наприклад:* місто – країна). Дані при цьому перетворюються та виражаються у термінах більш високопоставлених понять, що супроводжується їх стисненням. Об'єднання пов'язаних понять у рамках загальної гілки ієрархії дозволяє реалізувати агрегацію даних.

Агрегація даних є способом їх згортання, підсумовування або групування.

Для числових даних агрегація може здійснюватися шляхом приведення деталізованих даних до більш узагальненого їх подання з використанням функцій агрегації: підсумовування, знаходження середнього, мінімального або максимального значення тощо.

2.4. ЗАВДАННЯ ДЛЯ САМОСТІЙНОЇ РОБОТИ

Завдання 1. Розробити власну програму, яка здійснює попередню обробку набору даних для їх подальшого аналізу з використанням алгоритмів Data Mining.

Предметну область, із якої відібрано дані, кількість ознак, їх типи кожен студент визначає самостійно. Кількість ознак повинна бути не менша 5. У наборі даних повинні бути числові та категоріальні змінні, представлені у різних шкалах виміру.

Програма повинна виконувати наступне:

1) здійснювати аналіз сирих даних із метою виявлення проблемних значень;

2) проводити очистку даних, обробляючи пропуски даних, викиди, суперечливі та помилкові значення змінних;

3) здійснювати перетворення та кодування даних різних типів: нормалізацію, dummy-кодування, у разі потреби – дискретизацію, інтеграцію та агрегацію;

4) формувати очищений підготовлений для подальшого аналізу в Data Mining набір даних, у якому здійснено необхідні перетворення даних.

Мову програмування та середовище для розробки кожен студент обирає за власним вибором самостійно.

Набір даних для аналізу можна сформувати самостійно або скористатися готовими Data Set, які можна знайти за адресами:

1. DataSets платформи Kaggle – спільноти спеціалістів Data Science [Електронний ресурс]. – Режим доступу: https://www.kaggle.com/datasets
2. Джерела даних [Електронний ресурс]. – Режим доступу: https://medium.com/@kpi.vision.hack/kpi-vision-hack-datasets-5cec04a5ece2
3. Data Science Central [Електронний ресурс]. – Режим доступу: https://www.datasciencecentral.com/profiles/blogs/great-github-list-of-public-data-sets
4. 52 Data Set для тренировочных проектов [Електронний ресурс]. – Режим доступу: https://habr.com/ru/company/edison/blog/480408/
5. Где взять данные для машинного обучения: 3 способа собрать Data Set из открытых источников [Електронний ресурс]. – Режим доступу: https://chernobrovov.ru/articles/gde-vzyat-dannye-dlya-mashinnogo-obucheniya-3-sposoba-sobrat-dataset-iz-otkrytyh-istochnikov.html

Зауваження. У наведених вище наборах дуже велика кількість даних, можна взяти частину з них. Для демонстрації роботи програми у наборі необхідно створити проблемні дані (пропущені, помилкові тощо).

КОНТРОЛЬНІ ПИТАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ №2

1. Набори даних та їх атрибути. Типи даних.
2. Шкали категоріальних та числових даних.
3. Data Preparation: основні завдання обробки даних.
4. Відбір та генерація ознак, інтеграція даних.
5. Очищення даних. Способи обробки дублікатів, суперечливих, помилкових та екстремальних даних.
6. Перетворення даних: нормалізація та стандартизація, дискретизація, агрегація, dummy-кодування.

3. ПЕРВИННИЙ СТАТИСТИЧНИЙ АНАЛІЗ ДАНИХ

Лабораторна робота № 3

Мета: закріплення знань про первинний статистичний аналіз даних, описову статистику, перевірку статистичних гіпотез, оцінку параметрів розподілу. Формування умінь та навичок проведення статистичного аналізу, побудови варіаційного ряду, оцінки характеристик випадкової величини та параметрів розподілу за допомогою інструментальних засобів MS Excel.

Теоретичні знання: первинний статистичний аналіз даних. Вибірка, варіаційний ряд та його характеристики. Графічне представлення вибірки. Точкова та інтервальна оцінки параметрів генеральної сукупності, довірчі інтервали. Статистичні гіпотези та критерії. Закони розподілу. Статистичні функції, пакет аналізу MS Excel.

3.1. ПЕРВИННИЙ СТАТИСТИЧНИЙ АНАЛІЗ ДАНИХ В DATA MINING

3.1.1. Поняття вибірки, варіаційного ряду. Емпіричний розподіл вибіркових даних

Статистичні методи Data Mining базуються на первинному статистичному аналізі даних, який включає описову статистику, візуалізацію характеристик набору даних, перевірку статистичних гіпотез, оцінку параметрів розподілу та ідентифікацію його типу.

Проведення первинного статистичного аналізу дозволяє отримати узагальнені характеристики набору даних.

Здійснюючи статистичний аналіз, зазвичай мають справу з вибіркою даних, яка є частиною сукупності усіх даних – генеральної сукупності. Висновки, зроблені на основі аналізу вибіркових даних, розповсюджують на всю генеральну сукупність. Для цього вибірка повинна бути репрезентативною – відображати усі властивості генеральної сукупності.

Для кожної ознаки, яка характеризує об'єкти набору даних, можна побудувати одновимірну ***вибірку***, представлену значеннями цієї ознаки, розміщеними у порядку їх отримання. Це не завжди є зручним для подальшого аналізу, оскільки значення вибірки не є упорядкованими і можуть повторюватися. Тому на основі вибірки будують варіаційний ряд.

Варіаційний ряд – упорядкована послідовність значень ознаки x вибірки, елементи якої розміщено у порядку їх зростання, де кожне значення ознаки представлене тільки один раз.

Кожне значення варіаційного ряду називається ***варіантою*** – це таке значення ознаки вибірки, що не повторюється.

Частота варіанти – кількість елементів вибірки зі значенням ознаки, рівним цій варіанті.

Розмах варіаційного ряду R дорівнює різниці мінімального та максимального значень ознаки у вибірці:

$$R = x_{\max} - x_{\min}\,. \tag{3.1}$$

Відносна частота варіанти – відношення частоти варіанти до обсягу вибірки.

Для формування варіаційного ряду вибірку необхідно ***ранжувати*** – розмістити у порядку зростання та обчислити частоти і відносні частоти варіант:

$$\begin{array}{llllll} x_1, & x_2, & \ldots & x_i, & \ldots & x_k \\ n_1, & n_2, & \ldots & n_i, & \ldots & n_k \\ p_1, & p_2, & \ldots & p_i, & \ldots & p_k \end{array}, \tag{3.2}$$

де x_i – варіанта ($x_i < x_j$, якщо $i < j$);

n_i – частота варіанти x_i: $\sum_{i=1}^{k} n_i = n$;

k – кількість варіант; n – обсяг вибірки даних;

$p_i = \dfrac{n_i}{n}$ – відносна частота варіанти x_i: $\sum_{i=1}^{k} p_i = 1$.

Залежно від шкали, в якій вимірюють ознаку, представлену у вибірці, варіаційний ряд може бути дискретним та інтервальним.

Дискретний варіаційний ряд утворений дискретними числовими ознаками вибіркових даних, кількість значень яких є скінченою.

Інтервальний варіаційний ряд утворений числовими ознаками, які приймають неперервні значення. Значення інтервального варіаційного ряду задаються на інтервалах.

Для побудови інтервального варіаційного ряду необхідно визначити кількість інтервалів та їх ширину. З цією метою застосовують різні підходи, одним із яких є визначення ***кількості інтервалів*** за формулою Стерджеса з округленням отриманого значення до більшого цілого числа:

$$m = 1 + 3{,}322 \cdot \lg n. \tag{3.3}$$

Для вибірок невеликого обсягу кількість інтервалів може бути встановлена аналітиком без формул.

Визначивши кількість інтервалів m, розраховують ***крок розбиття*** за формулою:

$$h = \frac{x_{\max} - x_{\min}}{m}, \tag{3.4}$$

де x_{min} та x_{max} – мінімальне та максимальне значення ознаки у вибірці.

Після визначення кроку формують послідовність інтервалів інтервального варіаційного ряду від $[x_{\min}; x_{\min} + h)$ до інтервалу, який буде містити x_{max}, та розраховують частоти попадання значень у кожен інтервал і відповідні відносні частоти.

У випадку, коли дискретний варіаційний ряд містить багато різних значень, область вибіркових значень ознаки x також розбивають на інтервали, застосовуючи групування даних і формуючи інтервальний варіаційний ряд.

Статистичний аналіз даних спирається на теорію ймовірностей. Значення, представлені у варіаційному ряді, та його числові характеристики є випадковими величинами, оскільки вони базуються на вибіркових даних. Тому у процесі визначення характеристик необхідно враховувати їх ймовірнісний характер.

Побудова варіаційного ряду за даними вибірки дозволяє задати емпіричний закон розподілу значень аналізованої ознаки.

Закон розподілу випадкової величини – це співвідношення між значеннями випадкової величини та її ймовірностями.

Закон розподілу може бути заданий у вигляді таблиці, графіка, формули. Емпіричний закон розподілу у вигляді таблиці буде містити варіанти варіаційного ряду та відповідні їм відносні частоти.

Відносні частоти є ***емпіричними ймовірностями*** відповідних значень ознаки x вибіркових даних, а побудована з них таблиця – ***емпіричним розподілом вибірки***.

3.1.2. Побудова дискретного та інтервального варіаційного ряду

Побудова варіаційного ряду за даними вибірки у середовищі MS Excel здійснюється із використанням наступних функцій:

- МАКС()/MAX() – визначення максимального значення ознаки у вибірці;
- МИН()/MIN() – визначення мінімального значення ознаки у вибірці;
- ЧАСТОТА(*виб_знач*; *межі_інтерв*)/FREQUENCY(*виб_знач*; *межі_інтерв*) – визначення частот інтервального ряду вибірки, аргументи функції:

вибір_знач – масив даних вибірки;

межі_інтерв – діапазон меж інтервалів.

Зауваження. Функція ЧАСТОТА() є функцією массиву, тому при її введенні необхідно виділити масив комірок, які будуть містити виведені значення частот, ввести формулу з функцією та необхідними аргументами, натиснути клавішу F2 а потім комбінацію клавіш Ctrl + Shift + Enter.

Приклад 1. Побудувати дискретний варіаційний ряд для вибірки з даними про кількість дітей у 20 родинах: 0, 1, 2, 3, 1, 2, 1, 2, 1, 0, 4, 3, 2, 1, 1, 1, 0, 1, 0, 2.

1. Кількість різних значень ознаки рівна п’яти, розмістимо їх у порядку зростання: $x_1 = 0$, $x_2 = 1$, $x_3 = 2$, $x_4 = 3$, $x_5 = 4$. Ми отримали варіанти варіаційного ряду.

2. Для кожної варіанти визначимо частоти – кількість родин із відповідним значенням варіанти: для значення варіанти $x_1 = 0$ частота $n_1 = 4$, для значення варіанти $x_2 = 1$ частота $n_2 = 8$, для значення варіанти $x_3 = 2$ частота $n_3 = 5$, для значення варіанти $x_4 = 3$ частота $n_4 = 2$, для значення варіанти $x_5 = 4$ частота $n_5 = 1$. Ми визначили частоти варіаційного ряду.

3. Для визначення відносних частот розділимо кожну частоту на обсяг вибірки $n = 20$. Маємо:

$$p_1 = \frac{4}{20} = 0{,}2, \quad p_2 = \frac{8}{20} = 0{,}4, \quad p_3 = \frac{5}{20} = 0{,}25, \quad p_4 = \frac{2}{20} = 0{,}1, \quad p_5 = \frac{1}{20} = 0{,}05.$$

4. Занесемо розраховані значення у таблицю 3.1, отримаємо дискретний варіаційний ряд, заданий у табличній формі – емпіричний розподіл вибірки.

Таблиця 3.1

Дискретний варіаційний ряд

Варіанти, x_i	0	1	2	3	4	Всього
Частоти, n_i	4	8	5	2	1	20
Відносні частоти, p_i	0,2	0,4	0,25	0,1	0,05	1

Приклад 2. Побудувати інтервальний варіаційний ряд для вибірки, яка містить 50 значень ознаки, представлених у таблиці 3.2.

Таблиця 3.2

Вибірка даних

-1,75	-1,25	-0,75	-0,25	0,26	0,75	1,26	-1,55	-1,22	-0,74
-0,24	0,3	0,9	1,4	-1,42	-1,2	-0,73	-0,22	0,33	1,24
1,5	-1,3	-1	-0,6	-0,19	0,4	1,74	-1,26	-0,99	-0,59
-0,1	0,58	-0,81	-0,4	-0,05	0,6	-0,81	-0,33	0	0,71
-0,76	-0,3	0,1	0,73	-0,26	0,12	0,74	0,18	0,2	0,24

1. Для побудови інтервального варіаційного ряду обсягом $n = 50$ розрахуємо за формулою 3.3 кількість інтервалів:

$$m = 1 + 3{,}322 \cdot \lg n = 1 + 3{,}322 \cdot \lg 50 = 1 + 3{,}322 \cdot 1{,}699 = 6{,}64.$$

Отримане значення округлимо, кількість інтервалів буде рівна 7: $m = 7$.

2. Внесемо дані вибірки у комірки електронної таблиці MS Excel і за формулою 3.4 розрахуємо крок розбиття о області визначення ознаки на інтервали (рис. 3.1).

	A	B	C	D	E	F	G	H	I	J	K
2											
3		-1,75	-1,25	-0,75	-0,25	0,26	0,75	1,26	-1,55	-1,22	-0,74
4		-0,24	0,3	0,9	1,4	-1,42	-1,2	-0,73	-0,22	0,33	1,24
5		1,5	-1,3	-1	-0,6	-0,19	0,4	1,74	-1,26	-0,99	-0,59
6		-0,1	0,58	-0,81	-0,4	-0,05	0,6	-0,81	-0,33	0	0,71
7		-0,76	-0,3	0,1	0,73	-0,26	0,12	0,74	0,18	0,2	0,24
8											
9		Формули для розрахунку									
10		=СЧЁТ(B3:K7)		N	50	Кількість елементів вибірки даних					
11		=1+3,322*LOG10(E10)		M	7	Кількість інтервалів					
12		=МИН(B3:K7)		x_{min}	-1,75	Мінімальне значення					
13		=МАКС(B3:K7)		x_{max}	1,74	Максимальне значення					
14		=(E13-E12)/E11		h	0,5	Крок інтервалу					

Рис. 3.1. Розрахунок кроку розбиття на інтервали

3. Сформуємо інтервали з кроком $h = 0{,}5$ від x_{min} до x_{max} та порахуємо частоти – кількість елементів вибірки, які потрапляють у кожен із інтервалів. Для розрахунку першого інтервалу необхідно до x_{min} додати крок $h: -1{,}75 + 0{,}5 = -1{,}25$, отримуємо інтервал [-1,75; -1,25) і т.д. до x_{max}. Останнім буде інтервал [1,25; 1,75).

4. Сформуємо таблицю, яка буде містити стовпці із межами визначених інтервалів та стовпець для частот, які будуть розраховані. Для обчислення частот інтервального варіаційного ряду необхідно виділити діапазон комірок D18:D24 та у рядку формул ввести: =ЧАСТОТА(B3:K7;B19:B25) (рис. 3.2).

5. Після введення формули необхідно натиснути клавішу F2, а потім комбінацію клавіш Ctrl + Shift + Enter. У діапазоні комірок D18:D24 з'являться частоти – кількість значень із вибірки, які знаходяться у кожному інтервалі.

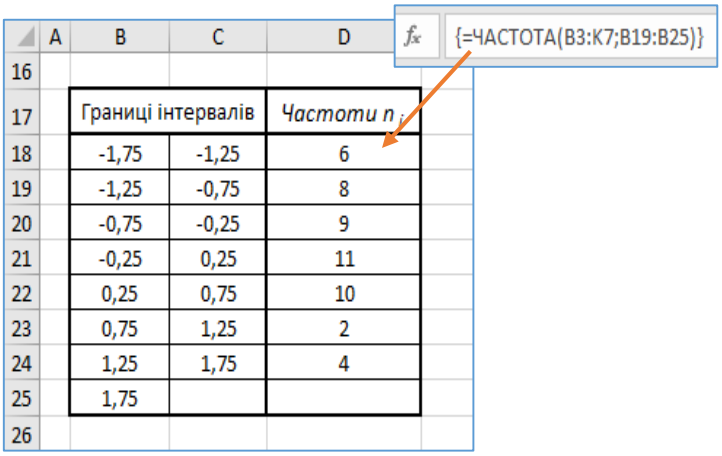


Рис. 3.2. Визначення частот варіаційного ряду

4. Для кожного інтервалу визначимо відносні частоти, розділивши відповідну частоту на обсяг вибірки.

5. Проведені розрахунки дозволяють побудувати інтервальний варіаційний ряд, заданий у табличній формі – емпіричний розподіл вибірки (рис. 3.3).

	S	T	U	V	W	X	Y	Z	AA	AB
3	Інтервальний варіаційний ряд									Сумма
4	x_i		[-1,75; -1,25)	[-1,25; -0,75)	[-0,75; -0,25)	[-0,25; 0,25)	[0,25; 0,75)	[0,75; 1,25)	[1,25; 1,75)	
5	n_i		6	8	9	11	10	2	4	50
6	p_i		0,12	0,16	0,18	0,22	0,20	0,04	0,08	1

Рис. 3.3. Інтервальний варіаційний ряд

3.1.3. Графічне представлення вибірки: полігон, гістограма, емпірична функція розподілу та щільність ймовірності

Побудова емпіричного розподілу вибірки дозволяє здійснити візуалізацію вибіркових даних і визначити емпіричну функцію розподілу та емпіричну щільність ймовірності. Це дає можливість оцінити теоретичну функцію розподілу та щільність ймовірності значень ознаки генеральної сукупності даних.

Функція розподілу (***інтервальна функція розподілу***) – це функція $F(x)$, яка для кожного значення випадкової величини $x = x_i$ визначає ймовірність того, що x прийме значення, менше за x_i: $F(x) = p(x < x_i)$.

Щільність ймовірності (***диференціальна функція розподілу***), функція розподілу щільності ймовірностей) – це функція $f(x)$, яка характеризує щільність, із якою розподіляється випадкової величина x у даній точці, та яка дорівнює першій похідній від функції розподілу: $f(x) = F'(x)$.

Вибірку, для якої побудовано дискретний варіаційний ряд, графічно представляють у вигляді ***полігону*** відносних частот – по осі абсцис відкладають можливі значення ознаки x_i, а по осі ординат – їх відносні частоти p_i, одержані точки з'єднують відрізками прямих (рис. 3.4). Полігон можна побудувати і для інтервального варіаційного ряду. Для цього по горизонтальній осі відкладають середини інтервалів, а по вертикальній – відповідні їм відносні частоти p_i.

Побудований полігон є графічним способом подання емпіричної щільності ймовірності випадкової величини.

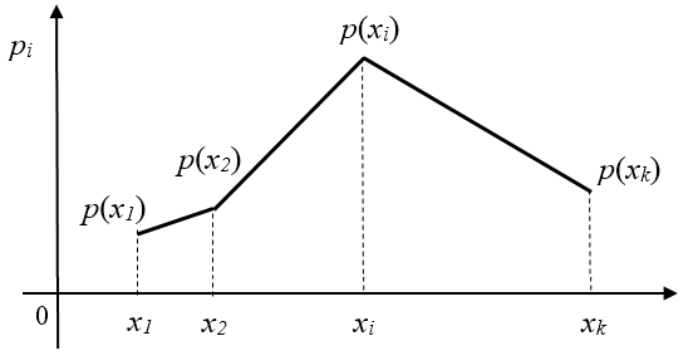


Рис. 3.4. Полігон відносних частот дискретного варіаційного ряду

Для вибірки, на підставі даних якої побудовано інтервальний варіаційний ряд, будують ***гістограму*** відносних частот – графічне представлення даних у вигляді прямокутників, основами яких є інтервали групування, а висоти дорівнюють відносним частотам p_i (рис. 3.5).

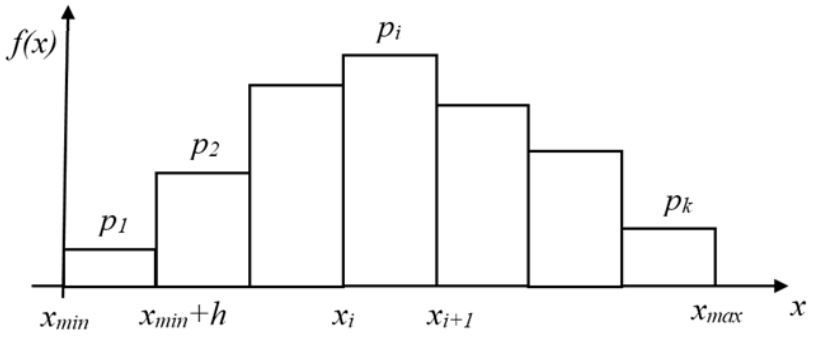


Рис. 3.5. Гістограма відносних частот інтервального варіаційного ряду

Побудована гістограма є графічним зображенням ***емпіричної функції щільності ймовірності***. При збільшенні обсягу вибірки вона прямує до теоретичної щільності ймовірності.

Для кожної варіанти варіаційного ряду можна обчислити накопичені відносні частоти w_i:

$$w_1 = p_1, \quad w_i = w_{i-1} + p_i, \tag{3.5}$$

де p_i – відносна частота варіанти, $i \in \{1, 2, \ldots, k\}$, k – кількість варіант (інтервалів).

Визначення накопичених відносних частот варіаційного ряду дозволяє задати емпіричний закон розподілу у вигляді графіка.

Побудувавши ***кумуляту*** – графік накопичених відносних частот, отримуємо графічне зображення ***емпіричної функції розподілу*** – функції $F(x)$, значення якої для кожного $x = x_i$, дорівнює ймовірності того, що ознака x, яка є випадковою величиною, приймає значення, менші за x_i. Значення емпіричної функції розподілу знаходяться в інтервалі [0; 1].

Аналітичне подання емпіричної функції розподілу дозволяє задати емпіричний закон розподілу у вигляді формули. Зі збільшенням обсягу вибірки емпірична функція розподілу наближається до теоретичної.

Функція розподілу ознаки, заданої дискретним варіаційним рядом, розривна і зростає стрибками при переході через кожне значення вибіркової ознаки даних (рис. 3.6, *а*).

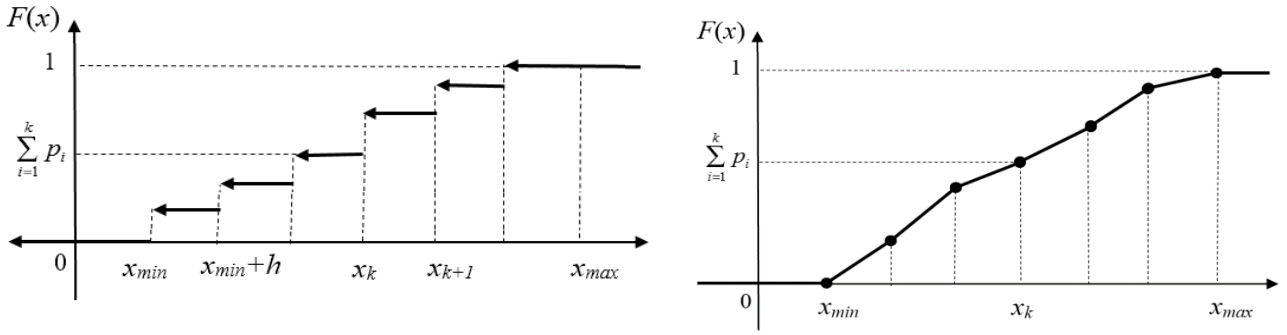
Функція розподілу ознаки, заданої інтервальним варіаційним рядом, має вигляд ламаної лінії, яка зростає на кожному інтервалі, оскільки за основу береться припущення, що ознака на кожному інтервалі має рівномірну щільність ймовірностей (рис. 3.6, *б*).

Приклад 3. Для вибірки, заданої дискретним варіаційним рядом, побудованим у прикладі 1 (табл. 3.1):

1) визначити накопичені відносні частоти;

2) побудувати полігон відносних частот;

3) побудувати емпіричну функцію розподілу ознаки, представленої у вибірці, та її графік.



а) дискретного варіаційного ряду *б*) інтервального варіаційного ряду

Рис. 3.6. Емпірична функція розподілу

1. Внесемо дані таблиці 3.1 у комірки електронної таблиці MS Excel і за формулою 3.5 розрахуємо накопичені відносні частоти w_i для кожної варіанти дискретного варіаційного ряду (рис. 3.7).

	A	B	C	D	E	F	G
2	Характеристики	Дискретний варіаційний ряд					Сумма
3	*Варіанта, x_i*	0	1	2	3	4	
4	*Частота, n_i*	4	8	5	2	1	20
5	*Відносна частота, p_i*	0,20	0,40	0,25	0,10	0,05	1
6	*Накопичена відносна частота, w_i*	0,20	0,60	0,85	0,95	1,00	

Рис. 3.7. Розрахунок накопичених відносних частот дискретного варіаційного ряду

2. Для заданих значень дискретного варіаційного ряду засобами MS Excel будуємо полігон відносних частот, обравши точкову діаграму (рис. 3.8).

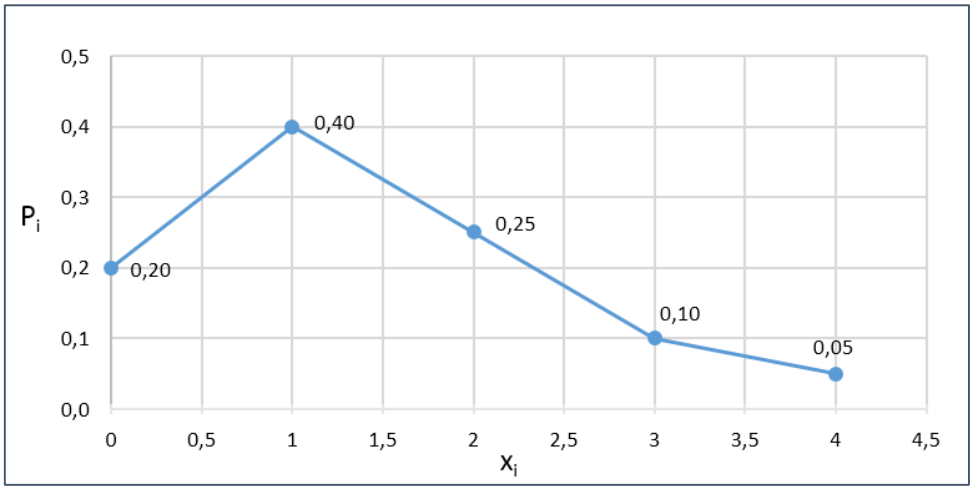


Рис. 3.8. Полігон дискретного варіаційного ряду вибірки

3. Засобами MS Excel для діапазону комірок, які містять значення варіант (B3:F3) та накопичених відносних частот (B6:F6), будуємо аналітичне задання емпіричної функції розподілу $F(x)$ та її графік (рис. 3.9):

$$F(x)=\begin{cases}0{,}00 & x\le 0\\ 0{,}20 & 0<x\le 1\\ 0{,}60 & 1<x\le 2\\ 0{,}85 & 2<x\le 3\\ 0{,}95 & 3<x\le 4\\ 1{,}00 & x>4\end{cases}.$$

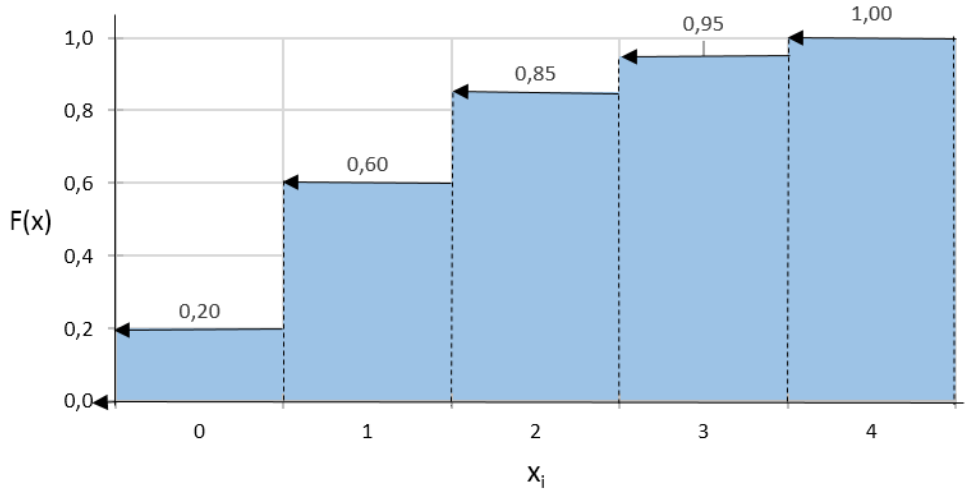


Рис. 3.9. Емпірична функція розподілу дискретного варіаційного ряду вибірки

Приклад 4. Для вибірки, заданої інтервальним варіаційним рядом, побудованим у прикладі 2 (рис. 3.3):
1) визначити накопичені відносні частоти;
2) побудувати гістограму відносних частот;
3) побудувати графік емпіричної функції розподілу ознаки, представленої у вибірці.

1. Внесемо дані інтервального варіаційного ряду у комірки електронної таблиці MS Excel і за формулою 3.5 розрахуємо накопичені відносні частоти w_i для кожного інтервалу заданого варіаційного ряду (рис. 3.10). Для побудови графіків необхідно визначити та внести у таблицю також значення a_i, які є границями інтервалів.

	S	T	U	V	W	X	Y	Z	AA	AB
3	Інтервальний варіаційний ряд									Сумма
4	x_i		[-1,75; -1,25)	[-1,25; -0,75)	[-0,75; -0,25)	[-0,25; 0,25)	[0,25; 0,75)	[0,75; 1,25)	[1,25; 1,75)	
5	n_i		6	8	9	11	10	2	4	50
6	p_i		0,12	0,16	0,18	0,22	0,20	0,04	0,08	1
7	w_i	0	0,12	0,28	0,46	0,68	0,88	0,92	1,00	
8	a_i	-1,75	-1,25	-0,75	-0,25	0,25	0,75	1,25	1,75	

Рис. 3.10. Розрахунок накопичених відносних частот інтервального варіаційного ряду вибірки

2. Користуючись визначеними значеннями відносних частот та границь інтервалів для вибірки, заданої інтервальним варіаційним рядом засобами MS Excel, будуємо гістограму (рис. 3.11).

3. Для побудови графіка накопичених відносних частот, використовуючи розраховані границі інтервалів a_i, засобами MS Excel будуємо графік емпіричної функції розподілу $F(x)$ (рис. 3.12).

3.1.4. Числові характеристики вибірки

Серед основних числових характеристик вибіркових значень ознаки x розрізняють характеристики положення та характеристики розсіювання.

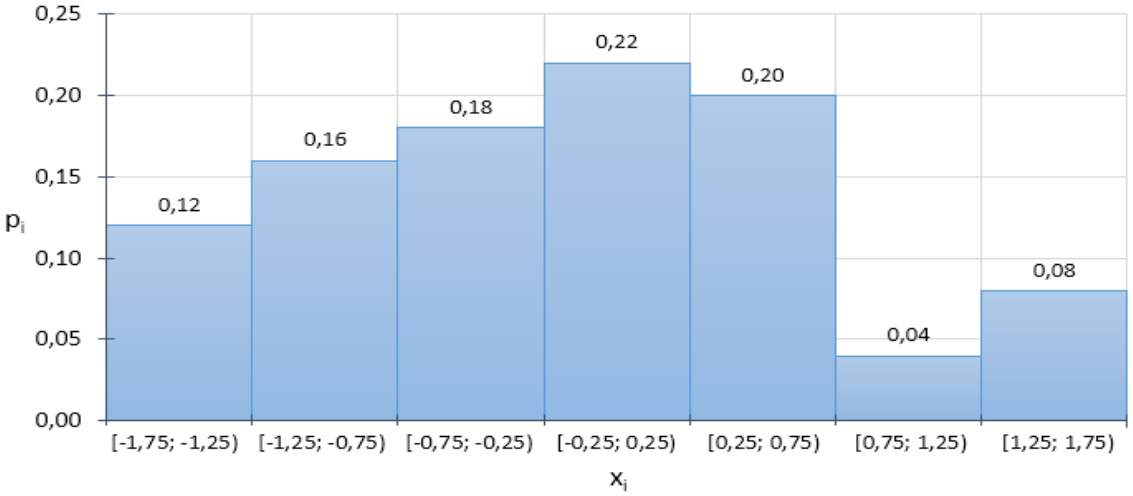


Рис. 3.11. Гістограма відносних частот інтервального варіаційного ряду

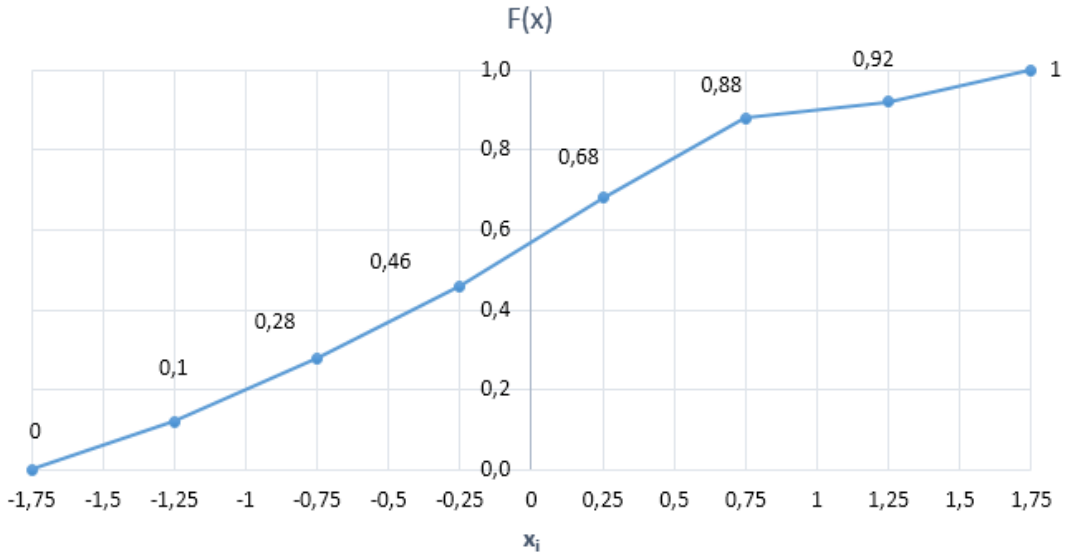


Рис. 3.12. Емпірична функція розподілу інтервального варіаційного ряду

До числових ***характеристик положення*** вибірки відносять середнє арифметичне, моду і медіану.

1. ***Середнє арифметичне*** значення ознаки $\overline{x}$, представленої у вибірці, обчислюють за формулами:

a) за даними вибірки:

$$\overline{x} = \frac{\sum_{i=1}^{n} x_i}{n}, \tag{3.6}$$

де x_i – i -те значення ознаки у вибірці даних, n – обсяг вибірки;

b) за даними варіаційного ряду вибірки:

$$\overline{x} = \frac{\sum_{i=1}^{k} x_i \cdot n_i}{n}, \tag{3.7}$$

де k – кількість варіант (інтервалів) варіаційного ряду, n – обсяг вибірки,

x_i – i -та варіанта (середина i -го інтервалу),

n_i – частота i -ї варіанти (i -го інтервалу).

2. ***Мода*** – значення ознаки, ймовірність якого максимальна: має найбільшу частоту у варіаційному ряді вибірки (якщо всі варіанти у варіаційному ряді однакові – мода відсутня).

3. ***Медіана*** – середина розподілу, таке значення ознаки у вибірці, що ділить її на дві рівні за обсягом частини.

До числових ***характеристик розсіювання*** – мір розкиду вибірки, відносять дисперсію та середнє квадратичне відхилення.

1. ***Дисперсія*** $\sigma_в^2$ – міра відхилень значень вибіркових даних від середнього арифметичного значення ознаки, представленої у вибірці. Обчислюють за формулами:

a) за даними вибірки:

$$\sigma_в^2 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n}, \text{ або } \sigma_в^2 = \frac{\sum_{i=1}^{n} x_i^2}{n} - \bar{x}^2 \tag{3.8}$$

де x_i – i -те значення ознаки у вибірці даних, n – обсяг вибірки,

$\bar{x}$ – середнє арифметичне значення ознаки у вибірці даних;

b) за даними варіаційного ряду вибірки:

$$\sigma_в^2 = \frac{\sum_{i=1}^{k}(x_i - \bar{x})^2 \cdot n_i}{n}, \text{ або } \sigma_в^2 = \frac{\sum_{i=1}^{k} x_i^2 \cdot n_i}{n} - \bar{x}^2 \tag{3.9}$$

де x_i – i -та варіанта (середина i -го інтервалу),

n_i – частота i -ї варіанти (i -го інтервалу),

$\bar{x}$ – середнє арифметичне значення ознаки у вибірці даних, k – кількість варіант (інтервалів).

2. ***Середнє квадратичне (стандартне) відхилення*** $\sigma_в$ – додатній корінь квадратний із дисперсії:

$$\sigma_в = \sqrt{\sigma_в^2}\ . \tag{3.10}$$

3.1.5. Точкова та інтервальна оцінка параметрів генеральної сукупності. Статистичні гіпотези та критерії

Числові характеристики, розраховані за даними вибірки, є наближеними значеннями цих характеристик, оскільки при їх визначенні не розглядалися усі об'єкти генеральної сукупності. Тому визначені за вибірковими даними числові значення параметрів називають їх оцінками. Оцінки параметрів бувають:

1) ***точкові:*** визначаються одним числом;

2) ***інтервальні:*** визначаються межами інтервалу, до якого із заданою ймовірністю потрапляє оцінюваний параметр.

Точковою оцінкою параметра називається його наближене значення, отримане за вибіркою.

Вибіркове середнє арифметичне значення ознаки $\bar{x}$, представленої у вибірці, є незміщеною точковою оцінкою ***математичного сподівання*** генеральної сукупності. Це значить, що вибіркове середнє арифметичне наближено рівне математичному сподіванню.

Вибіркова дисперсія $\sigma_в^2$ є зміщеною точковою оцінкою дисперсії генеральної сукупності σ^2, тобто $\sigma^2 \neq \sigma_в^2$. Незміщеною точковою оцінкою дисперсії є ***виправлена вибіркова дисперсія***:

$$\sigma^2 = \frac{n}{n-1}\sigma_в^2. \tag{3.11}$$

Вибіркове середнє квадратичне (стандартне) відхилення $\sigma_в$ є зміщеною точковою оцінкою середнього квадратичного відхилення генеральної сукупності σ, тобто $\sigma \neq \sigma_в$. Незміщеною оцінкою середнього квадратичного відхилення є ***виправлене середнє квадратичне (стандартне) відхилення***:

$$\sigma = \sqrt{\frac{n}{n-1}} \cdot \sigma_в. \tag{3.12}$$

При невеликих обсягах вибірки незміщені точкові оцінки параметрів можуть мати значне розходження з істинними значеннями параметрів генеральної сукупності. Однак зі збільшенням обсягу вибірки це розходження зменшується і стає несуттєвим. Чим більшим є обсяг вибірки n, тим точнішими будуть оцінки параметрів генеральної сукупності.

Для того щоб перевірити, чи є дійсними результати, отримані за даними вибірки, формулюють статистичні гіпотези.

Статистичною гіпотезою називається будь-яке припущення про властивості досліджуваної величини, висунуте на основі вибіркових даних.

За змістом розрізняють такі види статистичних гіпотез:

1) про числові характеристики досліджуваної величини;

2) про рівність числових характеристик досліджуваної величини;

3) про вид закону розподілу досліджуваної величини;

4) про належність досліджуваних величин до однієї генеральної сукупності;

5) про вид моделі, яка описує взаємозв'язок між досліджуваними величинами;

6) про належність досліджуваних величин до одного класу тощо.

Основну гіпотезу, яку перевіряють, називають ***нульовою гіпотезою*** H_0.

Гіпотезу, що суперечить нульовій, називають ***альтернативною гіпотезою*** (конкуруючою). Альтернативних гіпотез може бути декілька: H_1, H_2, і т.д..

Прийняття гіпотези основної або однієї з альтернативних здійснюється на основі дослідження вибіркових даних за певним ***критерієм***, який обирається відповідно до змісту гіпотези та типу даних.

Існує цілий ряд критеріїв, які застосовують для перевірки статистичних гіпотез. На практиці як критерії використовують певні модельні розподіли, які приблизно відповідають розподілу досліджуваного параметра. Найчастіше це критерій Пірсона (χ^2– хі-квадрат), критерії Фішера, Стьюдента та інші.

Перевірка статистичної гіпотези передбачає розрахунок емпіричного значення критерію за даними вибірки та порівняння його з критичним значенням критерію для перевірки потрапляння до області прийняття нульової гіпотези чи до критичної області, де нульову гіпотезу слід відхилити.

Критичні значення критеріїв, які визначають межі критичної області, задають у спеціальних таблицях, а програмні пакети MS Excel, MatLab та інші мають засоби для їх розрахунку.

Перевірка статистичних гіпотез дозволяє на основі вибіркових даних за допомогою теорії ймовірності зробити науково обгрунтований висновок про прийняття нульової або альтернативної гіпотези. Критичні значення критеріїв задаються на певному рівні значущості, який вказують при формулюванні гіпотез.

Рівень значущості α – це ймовірність відкидання нульової гіпотези, якщо вона справедлива: ймовірність помилки. Зазвичай це значення, близькі до нуля: 0,001, 0,01, 0,05 та 0,1.

Довірчий рівень $p = 1 - \alpha$ – це ймовірність прийняття справедливої нульової гіпотези: ймовірність прийняття правильного рішення. Він відповідно до значень α частіше усього приймає значення, близькі до одиниці: 0,999, 0,99, 0,95 та 0,9.

Інтервальні оцінки параметрів генеральної сукупності дозволяють установити точність і надійність точкових оцінок за даними вибірки на заданому рівні значущості α.

Точність оцінки – це таке число δ, на яке з ймовірністю $p = 1 - \alpha$ може відрізнятися точкова оцінка параметру від його точного значення:

$$\left|a_в - a\right| < \delta, \tag{3.13}$$

де a – точне значення параметра, $a_в$ – його точкова оцінка за вибіркою.

Надійністю оцінки називають рівень довіри p – ймовірність, із якою виконується нерівність 3.13, яка задає точність оцінки.

Інтервальною оцінкою параметра називається інтервал, у якому із заданою точністю δ та надійністю p знаходиться визначена за вибіркою точкова оцінка параметра $a_в$ – ***довірчий інтервал***:

$$a_в \pm \delta, \text{ або } (a_в - \delta;\ a_в + \delta). \tag{3.14}$$

Надійність оцінки (або рівень значущості) при визначені довірчого інтервалу задають наперед.

3.1.6. Розрахунок точкових та інтервальних оцінок числових характеристик генеральної сукупності за вибірковими даними

При здійсненні точкової оцінки параметрів генеральної сукупності за вибірковими даними та для перевірки статистичних гіпотез із метою інтервальної оцінки параметрів у середовищі MS Excel використовують наступні функції:

1) СРЗНАЧ(*вибір_знач*)/AVERAGE(*вибір_знач*) – визначення вибіркового середнього;

2) МЕДИАНА(*вибір_знач*)/MEDIAN(*вибір_знач*) – визначення медіани;

3) МОДА(*вибір_знач*)/MODE(*вибір_знач*) – визначення моди;

4) ДИСП.Г(*вибір_знач*)/VAR.P(*вибір_знач*) – визначення вибіркової дисперсії;

5) ДИСП.В(*вибір_знач*)/VAR.S(*вибір_знач*) – визначення виправленої вибіркової дисперсії;

6) СТАНДОТКЛОН.Г(*вибір_знач*)/STDEV.P(*вибір_знач*) – визначення вибіркового стандартного відхилення;

7) СТАНДОТКЛОН.В(*вибір_знач*)/STDEV.B(*вибір_знач*) – визначення виправленого вибіркового стандартного відхилення;

8) ДОВЕРИТ.СТЬЮДЕНТ(α, σ, n)/CONFIDENCE.T(α, σ, n) – визначення точності оцінки вибіркового середнього (α – рівень значущості, σ – стандартне відхилення, n – обсяг вибірки).

Аргумент вказаних функцій – *вибір_знач*, є масивом даних вибірки.

Зауваження 1. При обсязі вибірки $n > 30$ функції для розрахунку вибіркових і виправлених значень дисперсії та середнього квадратичного (стандартного) відхилення дають практично однакові результати.

Зауваження 2. При визначенні точності оцінки вибіркового середнього функція ДОВЕРИТ.СТЬЮДЕНТ()/CONFIDENCE.T() використовує t-критерій Стьюдента для перевірки гіпотези про рівність середнього, розподіленого у генеральній сукупності за нормальним законом.

Довірчий інтервал для вибіркового середнього $\overline{x}$ при заданому значенні стандартного відхилення генеральної сукупності σ із використанням t-критерію Стьюдента буде рівним:

$$\left(\overline{x} - \frac{\sigma \cdot t}{\sqrt{n}}; \overline{x} + \frac{\sigma \cdot t}{\sqrt{n}}\right), \tag{3.15}$$

де n – обсяг вибірки, t – критерій Стьюдента, заданий на рівні значущості α, який знаходиться з рівності:

$$2\Phi(t) = 1 - \alpha.$$

$\Phi(t)$ – функція Лапласа, за значенням якої зі спеціальних таблиць або за допомогою програмних засобів визначають t-критерій Стьюдента та за формулою 3.15 розраховують межі довірчого інтервалу.

Приклад 5. Обчислити точкові оцінки числових характеристик генеральної сукупності значень ознаки, представленої вибіркою невеликого обсягу (рис. 3.13) із використанням функцій MS Excel.

	A	B	C	D	E	F	G	H	I	J	K	L
1												
2		Вибірка даних										
3		5	4	4	2	15	9	5	5	3	2	
4												

Рис. 3.13. Вибіркові значення ознаки

1. Із використанням статистичних функцій MS Excel здійснюємо розрахунок основних числових характеристик заданої вибірки, які є точковими оцінками відповідних характеристик генеральної сукупності (рис. 3.14).

	N	O	P
2			
3		=МАКС(B3:K3)	15
4		=МИН(B3:K3)	2
5		=МЕДИАНА(B3:K3)	4,5
6		=МОДА(B3:K3)	5
7		=СРЗНАЧ(B3:K3)	5,4
8		=ДИСП.В(B3:K3)	15,38
9		=ДИСП.Г(B3:K3)	13,84
10		=СТАНДОТКЛОН.В(B3:K3)	3,92
11		=СТАНДОТКЛОН.Г(B3:K3)	3,72
12			

Рис. 3.14. Обчислення точкових оцінок числових характеристик

Приклад 6. Є вибірка обсягом 100 місячних заробітних плат працівників фірми. Було визначено, що середнє значення величини місячної заробітної плати рівне 14550 грн, а стандартне відхилення генеральної сукупності становить 1324 грн. Припускаючи, що в генеральній сукупності даний параметр розподілений за нормальним законом, необхідно визначити довірчий інтервал для середнього значення $\overline{x}$ заробітної плати із заданою надійністю $p = 0{,}95$ (рівень значущості: $\alpha = 1 - 0{,}95 = 0{,}05$).

1. Для розрахунку довірчого інтервалу в MS Excel необхідно ввести формулу:

= ДОВЕРИТ.СТЬЮДЕНТ(0,05;1324;100)

2. Буде отримано значення 262,7. Отже, середня місячна заробітна плата працівників фірми з ймовірністю 95% знаходиться в межах: $14550 \pm 262{,}7$ грн.

3.1.7. Ідентифікація закону розподілу з використанням критерію Пірсона

У прикладних задачах аналізу за даними вибіркових значень ознаки x, яка є випадковою величиною, часто необхідно визначити відповідність генеральної сукупності даних певному теоретичному закону розподілу.

Серед великої кількості законів розподілу найбільш поширеними є нормальний, рівномірний та експоненціальний (показниковий) розподіли для неперервної випадкової величини і пуасонівський та біноміальний розподіли для дискретної випадкової величини (додаток *К*). Особливу роль відіграє нормальний розподіл, оскільки багато характеристик є нормально розподіленими. Виділяють також стандартний нормальний розподіл, який має рівне нулю математичне сподівання та рівне одиниці стандартне відхилення.

Використовуючи графічне представлення вибіркових значень досліджуваної ознаки, можна приблизно візуально оцінити тип розподілу генеральної сукупності (рис. 3.15).

Порівняння параметрів, оцінених за даними вибірки, із параметрами теоретичних законів розподілу, також дає можливість висловити припущення про тип розподілу генеральної сукупності значень досліджуваної величини. Зокрема, враховують наступне (додаток *К*):

1) закон розподілу Пуассона має рівні за значенням математичне сподівання й дисперсію та надто асиметричний розподіл частот – його використовують для опису подій, які рідко відбуваються;

2) експоненціальний (показниковий) закон розподілу також має рівні за значенням математичне сподівання й дисперсію;

3) для рівномірного закону розподілу характерним є рівномірно розподілені на деякому інтервалі значення величини;

3) нормальний закон розподілу має рівні за значенням математичне сподівання, моду і медіану та симетричний відносно них розподіл частот.

Для достовірного обґрунтування припущення про тип розподілу необхідно сформулювати статистичну гіпотезу про вид закону розподілу генеральної сукупності значень ознаки вибіркових даних та підтвердити її прийняття чи відхилення з використанням статистичного критерію згоди.

Існує багато критеріїв згоди, найбільш поширеним із них є ***критерій Пірсона*** (χ^2 – хі-квадрат). Критерій згоди Пірсона є універсальним, оскільки його можна застосовувати для перевірки на відповідність будь-якому закону розподілу.

Емпіричне значення критерію Пірсона розраховують на підставі вибіркових даних:

$$\chi^2_{емп} = \sum_{i=1}^{k} \frac{(n_i - n_i')^2}{n_i'}, \tag{3.16}$$

де n_i – емпіричні частоти, n_i' – теоретичні частоти, k – кількість варіант (інтервалів).

Теоретичні частоти розраховують за формулою:

$$n_i' = n \cdot p_i, \tag{3.17}$$

де p_i – ймовірність, розрахована за передбачуваним законом розподілу, n – обсяг вибірки.

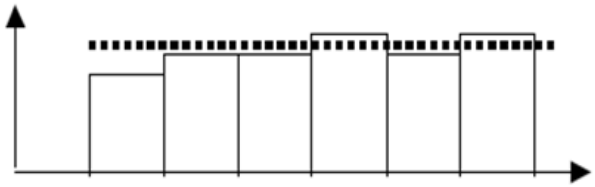
Критичне значення критерію Пірсона $\chi^2_{\alpha,l}$ визначають зі спеціальних таблиць або з використанням відповідних функцій програмних засобів, вказавши рівень значущості α та ступінь свободи l.

Якщо розраховане значення критерію Пірсона менше за критичне: $\chi^2_{\text{емп}} \le \chi^2_{\alpha,l}$ – нульова гіпотеза про відповідність передбачуваному закону розподілу приймається з ймовірністю $p = 1 - \alpha$. У протилежному випадку приймається альтернативна гіпотеза: випадкова величина не розподілена за цим законом розподілу.

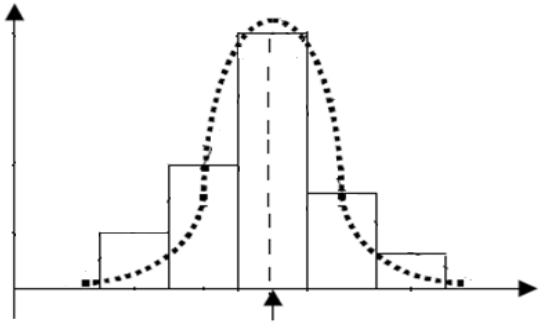
Ступінь свободи розраховують за формулою:

$$l = k - r - 1, \tag{3.18}$$

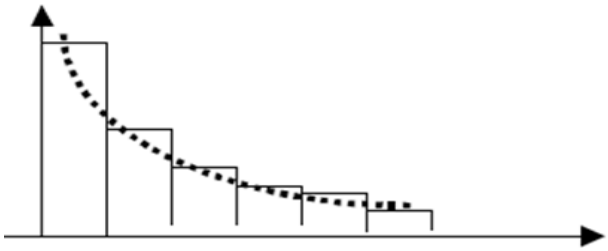
де k – кількість варіант (інтервалів), r – кількість оцінюваних параметрів теоретичного закону розподілу.



а) гістограма, близька до рівномірного розподілу



б) гістограма, близька до нормального розподілу



в) гістограма, близька до експоненціального розподілу

Рис. 3.15. Візуальна оцінка типу розподілу генеральної сукупності

Для перевірки статистичних гіпотез та індентифікації закону розподілу за даними вибірки використовують такі функції MS Excel:

1) НОРМ.СТ.РАСП(x, m)/NORM.S.DIST(x, m) – визначення теоретичної функції стандартного нормального розподілу при $m = 1$ або теоретичної щільності ймовірності стандартного нормального розподілу при $m = 0$ (x – значення з інтервалу області визначення ознаки вибіркових даних);

2) ХИ2.ОБР.ПХ(α, l)/CHISQ.INV.RT(α, l) – визначення критичного значення критерію згоди Пірсона χ^2 (α – рівень значущості, $l = k - 1$ – ступінь свободи, k – кількість варіант (інтервалів));

3) ХИ2.ТЕСТ(n, n')/CHISQ.TEST(n, n') – перевірка узгодженності розподілу емпіричних і теоретичних частот із використанням критерію Пірсона χ^2 (n – емпіричні частоти, n' – теоретичні частоти).

Приклад 7. За даними інтервального варіаційного ряду вибірки знайти закон розподілу генеральної сукупності значень досліджуваної ознаки (табл. 3.3).

Таблиця 3.3

Інтервальний варіаційний ряд

Інтервали $[x_i; x_{i+1})$	$[-2; -1{,}2)$	$[-1{,}2; -0{,}4)$	$[-0{,}4; 0{,}4)$	$[0{,}4; 1{,}2)$	$[1{,}2; 2)$
Частоти, n_i	6	11	21	7	5

1. Внесемо дані інтервального варіаційного ряду у комірки електронної таблиці MS Excel. Користуючись заданими значеннями частот та границь інтервалів інтервального варіаційного ряду вибірки будуємо гістограму частот (рис. 3.16).

2. Аналіз графічного представлення вибіркових значень дозволяє висловити припущення, що значення генеральної сукупності досліджуваної ознаки розподілені за нормальним законом. Формулюємо нульову та альтернативну гіпотези:

H_0: досліджувана величина розподілена за нормальним законом розподілу;

H_1: досліджувана величина не розподілена за нормальним законом розподілу.

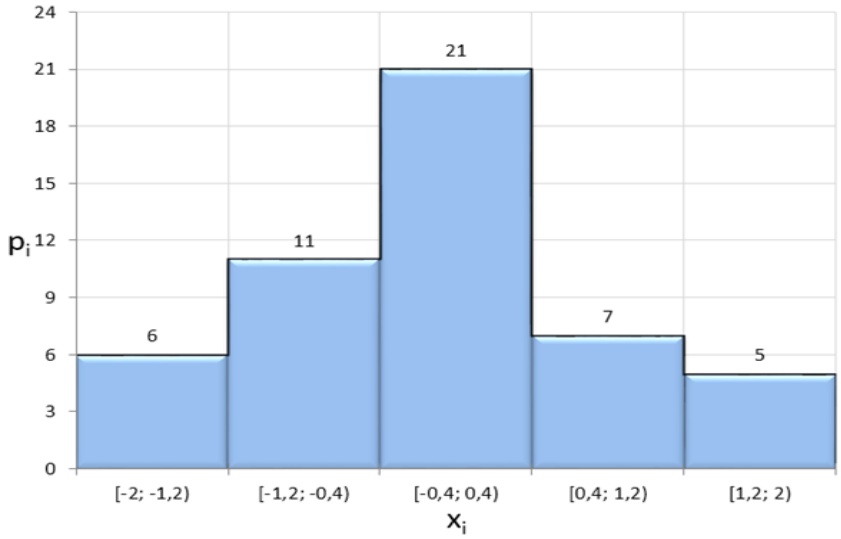


Рис. 3.16. Гістограма частот інтервального варіаційного ряду

3. У MS Excel будуємо таблицю, яка буде містити необхідні для перевірки гіпотези розрахунки (рис. 3.17). Перші три стовпці таблиці заповнюємо даними з таблиці 3.3, вводячи границі інтервалів, інтервали та частоти варіаційного ряду.

	Y	Z	AA	AB	AC	AD	AE
9	Границі інтервалів	Інтервали $[x_i; x_{i+1})$	Емпіричні частоти n_i	Емпіричний розподіл, p_i	Теоретичні частоти $n'_i=np'_i$	Стандартний нормальний розподіл, p'_i	Розрахунок критерію Пірсона $\chi^2_{емп}$
10	-2	менше за -2	0	0,00	1,14	0,023	1,138
11	-1,2	[-2; -1,2)	6	0,12	4,62	0,092	0,415
12	-0,4	[-1,2; -0,4)	11	0,22	11,48	0,230	0,020
13	0,4	[-0,4; 0,4)	21	0,42	15,54	0,311	1,917
14	1,2	[0,4; 1,2)	7	0,14	11,48	0,230	1,745
15	2	[1,2; 2)	5	0,10	4,62	0,092	0,032
16	2,8	більше ніж 2	0	0,00	1,01	0,020	1,010
17		Всього	50	1	50	1	6,28

Рис. 3.17. Обчислення емпіричного значення критерію Пірсона

4. Стовпець *Емпіричний розподіл* (діапазон комірок AB10:AB16 на рисунку 3.17) містить відносні частоти p_i заданого варіаційного ряду – емпіричну ймовірність того, що значення досліджуваної ознаки буде належати *i*-му інтервалу $[x_i; x_{i+1})$. Для їх визначення необхідно розділити відповідну частоту n_i на обсяг вибірки $n = 50$.

5. У стовпці *Стандартний нормальний розподіл* (діапазон комірок AD10:AD16 на рисунку 3.17) із використанням функції MS Excel НОРМ.СТ.РАСП(*x*;1)/NORM.S.DIST(*x*;1) розраховуємо теоретичні відносні частоти p'_i – інтегральну ймовірність того, що значення ознаки, розподіленої за нормальним стандартним розподілом, буде належати *i*-му інтервалу $[x_i; x_{i+1})$. Для цього необхідно ввести наступні формули:

– у комірку AD10: =НОРМ.СТ.РАСП(Y10;1);
– у комірку AD11: =НОРМ.СТ.РАСП(Y11;1)-НОРМ.СТ.РАСП(Y10;1);
– у комірку AD12: =НОРМ.СТ.РАСП(Y12;1)-НОРМ.СТ.РАСП(Y11;1);
– далі аналогічно до комірки AD16;

– у комірку AD16: =НОРМ.СТ.РАСП(Y16;1)-НОРМ.СТ.РАСП(Y15;1).

6. У стовпці *Теоретичні частоти* (діапазон комірок AC10:AC16 на рисунку 3.17) розраховуємо за формулою 3.17 теоретичні частоти $n_i^{'} = n \cdot p_i^{'}$ для кожного інтервалу. Для цього необхідно ввести наступні формули: у комірку AC10: =AD10*50, у комірку AC11: =AD11*50 і т.д., до комірки AC16: =AD16*50.

7. У стовпці *Розрахунок критерію Пірсона* (діапазон комірок AE10:AE16 на рисунку 3.17) з метою визначення за формулою 3.16 емпіричного значення критерію Пірсона $\chi^2_{\text{емп}}$ для кожного інтервалу варіаційного ряду розраховуємо значення $\frac{(n_i - n_i')^2}{n_i'}$: формула для комірки AE10: =((AA10-AC10)^2)/AC10 і т.д. до комірки AE16: =((AA16-AC16)^2)/AC16).

У комірці AE17 знаходимо суму: =СУММ(AE10:AE16) та отримуємо $\chi^2_{емп} = 6{,}28$.

8. Критичне значення критерію Пірсона на рівні значущості $\alpha = 0{,}05$ розраховуємо із використанням функції MS Excel ХИ2.ОБР.ПХ(α;*l*)/ CHISQ.INV.RT(α;*l*) із числом ступенів свободи $l = k - 1$, де $k = 7$ – кількість інтервалів. У комірку робочого аркуша вводимо формулу:

=ХИ2.ОБР.ПХ(0,05;6).

Отримуємо критичне значення критерію Пірсона $\chi^2_{\alpha,l} = \chi^2_{0{,}05;6} = 12{,}59$.

9. Порівнюємо емпіричне та критичне значення критерію згоди Пірсона, маємо $\chi^2_{\text{емп}} < \chi^2_{\alpha,l}$ – емпіричне значення критерію є меншим за критичне: $6{,}28 < 12{,}59$. Отже, нульова гіпотеза приймається, досліджувана величина розподілена за нормальним законом розподілу з ймовірністю 95%.

10. Використовуючи функцію ХИ2.ТЕСТ(n, n')/CHISQ.TEST(n, n'), перевіримо ймовірність узгодженості емпіричних n і теоретичних $n^{'}$ значень частот із використанням критерію Пірсона. Першим аргументом функції є масив емпіричних частот розподілу, що підлягають порівнянню, а другим – теоретичні частоти стандартного нормального розподілу (зі середнім рівним 0 і стандартним відхиленням рівним 1). Вводимо формулу:

=ХИ2.ТЕСТ(AA10:AA16;AC10:AC16).

Отримуємо ймовірність рівну 0,393, яка є більшою за рівень значущості 0,05. Це свідчить про те, що суттєва різниця між емпіричними та теретичними частотами відсутня. Таким чином, ми отримали іще одне підтвердження відповідності розподілу генеральної сукупності значень досліджуваної ознаки нормальному закону розподілу.

3.1.8. Пакет аналізу даних MS Excel: описова статистика, діаграма Парето

Пакет аналізу MS Excel *Аналіз даних/Data Analysis* включає засіб для аналізу одномірних наборів даних – *Описова статистика/Descriptive Statistics*. ***Описова статистика*** є поширеним методом аналізу числових даних, за допомогою якого обчислюються статистичні оцінки характеристик набору даних.

Ці інструменти входять до складу надбудов MS Excel та по замовчуванню можуть бути не установлені. Для того, щоб ними можна було скористатися, необхідно установити та активувати надбудову *Пакет аналізу/Analysis ToolPak*.

Надбудови – це компоненти MS Excel, що надають доступ до додаткових функцій і команд, пов'язаних з аналізом даних.

Серед показників ***описової статистики*** є уже розглянуті нами – середнє арифметичне, медіана, мода, вибіркова дисперсія, розмах варіації, мінімальне та максимальне значення, сума значень вибірки, кількість елементів вибірки та:

a) *стандартна помилка* – обчислюється як відношення стандартного відхилення до кореня квадратного з числа елементів вибірки;

b) *ексцес* – форма розподілу випадкової величини;

c) *асиметричність* – слугує для оцінювання симетричності розподілу випадкової величини щодо середньої.

Діаграма Парето або відсортована гістограма – це стовпчикова діаграма, що показує частоту повторюваності значень. Така гістограма демонструє кількісні співвідношення різних показників у порядку їх убування по частоті. Для її побудови використовують Майстер діаграм із *Пакету аналізу/Analysis ToolPak*.

Діаграма Парето графічно відображає ***закон Парето***, який стверджує: 20% зусиль дають 80% результату, а 80% зусиль, що залишилися, дають 20% результату. Цей закон дозволяє при плануванні оптимізації процесів виявити фактори, які є суттєвими для ефективності проєкту й зосередити увагу саме на них.

Приклад 8. Здійснення установки та активації надбудови *Пакет аналізу/Analysis ToolPak* в MS Excel.

1. Перевірте, чи є група *Аналіз/Analysis* у вкладці *Дані/Data* програми MS Excel. Якщо такої групи немає, її потрібно установити та активувати.

2. Для установки та активації Пакету аналізу необхідно відкрити вкладку *Файл/File*, обрати *Параметри/Options – Надбудови/Add-ins.*

3. Упевнившись, що у нижній частині вікна *Параметри Excel/Excel Options* в полі *Управління/Manage* обрано елемент *Надбудови Excel/Excel Add-ins*, необхідно натиснути кнопку *Перехід/Go.*

4. У діалоговому вікні *Надбудови/Add-ins* необхідно установити прапорці *Пакет аналізу/Analysis ToolPak* та *Пошук рішення/ Solver Add-in* й натиснути кнопку *ОК* (рис. 3.18). Якщо виведеться пропозиція установити надбудову – натисніть *Да/Yes* для її установки.

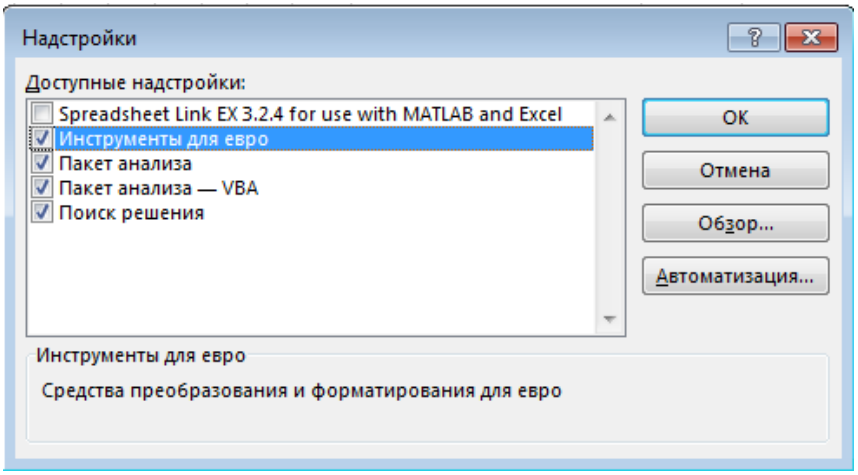


Рис. 3.18. Вікно Надбудов /Add-ins

5. Після цього необхідно відкрити вкладку MS Excel *Дані/Data* й упевнитися, що там є група *Аналіз/Analysis* (рис. 3.19). Вона містить кнопки команд для надбудов *Аналіз даних* та *Пошук рішення/ Solver Add-in.*

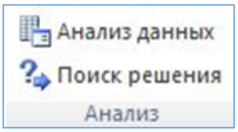


Рис. 3.19. Група *Аналіз/Analysis*

6. Після установки та активації *Пакету аналізу/Analysis ToolPak* в MS Excel стануть доступними інструментальні засоби для здійснення статистичного аналізу даних: описова статистика, діаграми Парето (відсортована гістограма), генерація значень випадкової величини за певним законом розподілу, кореляційний, регресійний, дисперсійний аналізи.

Приклад 9. Визначити методом описової статистики узагальнювальні характеристики набору даних про роботу менеджерів із залучення клієнтів – кількість прийнятих ними протягом дня клієнтів, із якими було оформлено договір на надання послуг, які реалізує фірма (рис. 3.20).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Кількість клієнтів, залучених менеджерами														
2	№ п/пр менеджера	1	2	3	4	5	6	7	8	9	10	11	12	13	14
3	Кількість клієнтів	12	11	12	10	12	10	12	14	11	12	9	13	10	14

Рис. 3.20. Дані про кількість залучених менеджерами клієнтів

1. У MS Excel за рисунком 3.20 сформуємо електронну таблицю з даними про кількість залучених менеджерами клієнтів.

2. Для здійснення аналізу даних методом описової статистики необхідно відкрити вкладку *Дані/Data* й у групі *Аналіз/Analysis* обрати команду *Аналіз даних/Data Analysis.*

3. У вікні *Аналіз даних/Data Analysis*, що відкриється (рис. 3.21), необхідно обрати *Описова статистика/Descriptive Statistics* та натиснути кнопку *ОК.*

4. У діалоговому вікні, що з'явиться (рис. 3.22), в області *Вхідні дані/Input Range* необхідно вказати вхідний інтервал, виділяючи діапазон даних із кількістю прийнятих клієнтів. Параметр *Групування/Grouped By* потрібно обрати по рядкам та установити прапорець *Мітки в першому стовпці/Labels in first column*, тому що ця комірка містить підпис.

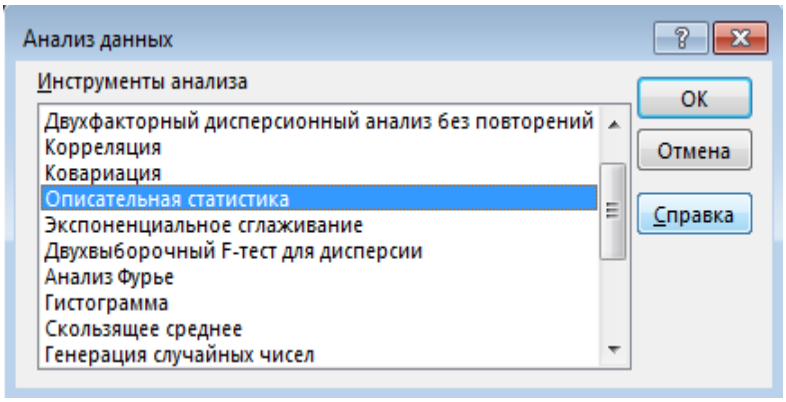


Рис. 3.21. Вікно Аналіз даних/Data Analysis

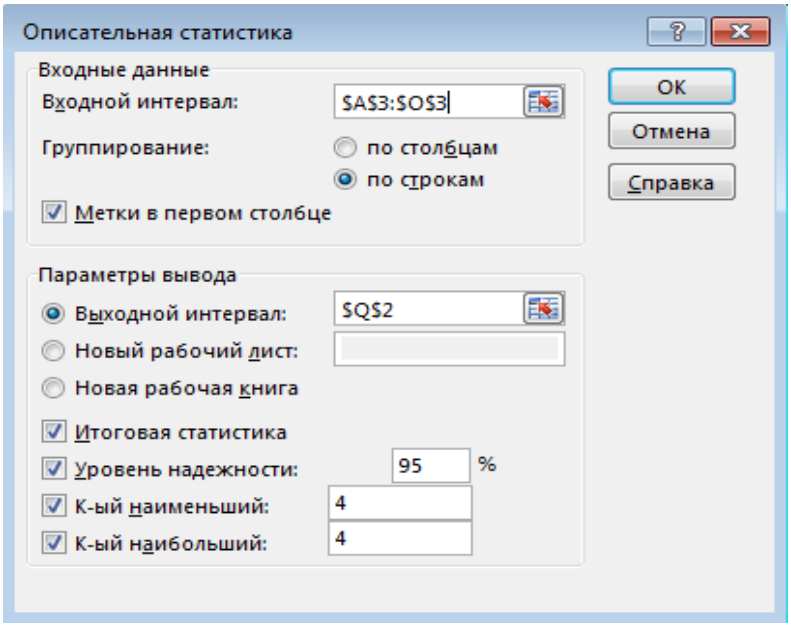


Рис. 3.22. Вікно Описова статистика /Descriptive Statistics

5. В області *Параметри виводу/Output Range* необхідно включити параметр *Вихідний інтервал/Output Range.* Для вказівки місця виведення результату на аркуші Excel спочатку потрібно клацнути у текстовому полі параметра *Вихідний інтервал/Output Range*, а потім виділити комірку (наприклад, *Q2*), що вказує адресу лівого верхнього кута області виведення даних. Далі необхідно установити такі прапорці:

A. *Підсумкова статистика/Summary statistics*: дана опція управляє виведенням вихідних даних.

B. *Рівень надійності/Confidence Level for Mean*: дана опція обчислює половину довжини довірчого інтервалу для середнього із заданою значимістю (у %). Встановимо надійність рівною 95%. Це означає, що ймовірність того, що середнє генеральної сукупності даних перебуває в межах довірчого інтервалу, дорівнює 0,95.

6. Далі необхідно натиснути кнопку *ОК*. Excel обчислить узагальнювальні показники й розмістить їх у вигляді таблиці у двох стовпцях вказаного вихідного інтервалу (рис. 3.23).

	P	Q	R
1			
2		*Кількість клієнтів*	
3			
4		Среднее	11,57143
5		Стандартная ошибка	0,402114
6		Медиана	12
7		Мода	12
8		Стандартное отклонение	1,504572
9		Дисперсия выборки	2,263736
10		Эксцесс	-0,589907
11		Асимметричность	0,077434
12		Интервал	5
13		Минимум	9
14		Максимум	14
15		Сумма	162
16		Счет	14
17		Наибольший(2)	14
18		Наименьший(2)	10
19		Уровень надежности(95,0%)	0,868714

Рис. 3.23. Результати аналізу, отриманого за методом описової статистики

7. Інтерпретація отриманих результатів: вихідні дані містять такі узагальнювальні характеристики вибіркових даних.

A. Загальна кількість клієнтів, з якими 14-ма менеджерами було оформлено договір на надання послуг, які реалізує фірма, становить 162 особи.

B. Менеджери у середньому оформляють договір із 12-ма клієнтами в день (середнє 11,57 було округлено до цілих).

C. Медіана та мода рівні також 12, що свідчить про симетричність емпіричного розподілу та те, що робоче навантаження рівномірно розподілене між менеджерами.

D. Стандартне відхилення є незначним і рівним 1,5 – це є розкид навколо середнього значення. Отже, щоденне навантаження менеджера в день становить 10-13 клієнтів, що свідчить про майже рівномірний розподіл продуктивності праці серед менеджерів.

E. Невеликий розмах варіаційного ряду для даної вибірки – 5: різниця між максимальним (рівним 14) та мінімальним (рівним 9) значеннями кількості залучених в день клієнтів також свідчить про рівномірний розподіл продуктивності праці менеджерів.

F. Точність оцінки середнього, розрахованого за вибірковими даними, становить 0,87, рівень надійності рівний 95%. Отже, довірчий інтервал для розрахованого середнього значення визначається як $11{,}57 \pm 0{,}87$. У результаті можемо стверджувати, що розраховане середнє значення кількості оформлених договорів на надання послуг, які реалізує фірма, із ймовірністю 0,95 буде знаходитися у інтервалі (10,7; 12,4). У даний інтервал попадає одне ціле значення, рівне 12, що говорить про достатню точність оцінки середнього значення за даними вибірки.

Приклад 10. За даними вибірки попереднього прикладу 9 (рис. 3.20) побудувати відсортовану гістограму – діаграму Парето, та визначити менеджерів, які працювали більш продуктивно й забезпечили для фірми більший прибуток.

1. Для створення діаграми Парето в MS Excel скористаємося майстром діаграм із *Пакету аналізу*. Для побудови відсортованої гістограми використовують два стовпці даних – один із даними про кількість залучених клієнтів, які необхідно проаналізувати (у прикладі 9 діапазон B3:O3), і один із числами, які позначають інтервали, для яких необхідно розрахувати частоти.

Інтервали утворюють рівномірним розбиттям значень вибірки від мінімального до максимального значення. Якщо числа не ввести, програма створить їх автоматично. Ми введемо наступі значення інтервалів – кармани: 9, 10, 11, 12, 13, 14.

3. Далі потрібно відкрити вкладку *Данні/Data* у групі *Аналіз/Analysis* і обрати команду *Аналіз даних/Analysis ToolPak*. У вікні, що відкриється, необхідно обрати *Гістограма/Histogram* та натиснути кнопку *ОК*.

4. У діалоговому вікні, що з'явиться, в області *Вхідні дані/Input Range* потрібно вказати вхідний інтервал, виділяючи діапазон даних із кількістю залучених менеджерами клієнтів (B3:O3) і діапазон даних із карманами – інтервалами для визначення частот та установити прапорець *Мітки/Labels*.

5. В області *Параметри виводу/ Output Range* необхідно включити параметр *Вихідний інтервал/Output Range* та виділити комірку (наприклад, T2), що вказує адресу лівого верхнього кута області виведення даних. Необхідно установити наступні прапорці:

A. *Парето/Pareto*: дана опція вказує про створення відсортованої діаграми.

B. Інтегральний процент.

C. *Виведення графіка/Chart Output*: відображає гістограму.

6. Далі необхідно натиснути *OK*. MS Excel обчислить необхідні показники та виведе гістограму із вказаними частотами (рис. 3.24, рис. 3.25).

	S	T	U	V	W	X	Y
1							
2		*Карман*	*Частота*	*Интегральный %*	*Карман*	*Частота*	*Интегральный %*
3		9	1	7,14%	12	5	35,71%
4		10	3	28,57%	10	3	57,14%
5		11	2	42,86%	11	2	71,43%
6		12	5	78,57%	14	2	85,71%
7		13	1	85,71%	9	1	92,86%
8		14	2	100,00%	13	1	100,00%
9		Еще	0	100,00%	Еще	0	100,00%

Рис. 3.24. Результати з розрахунками до діаграми Парето

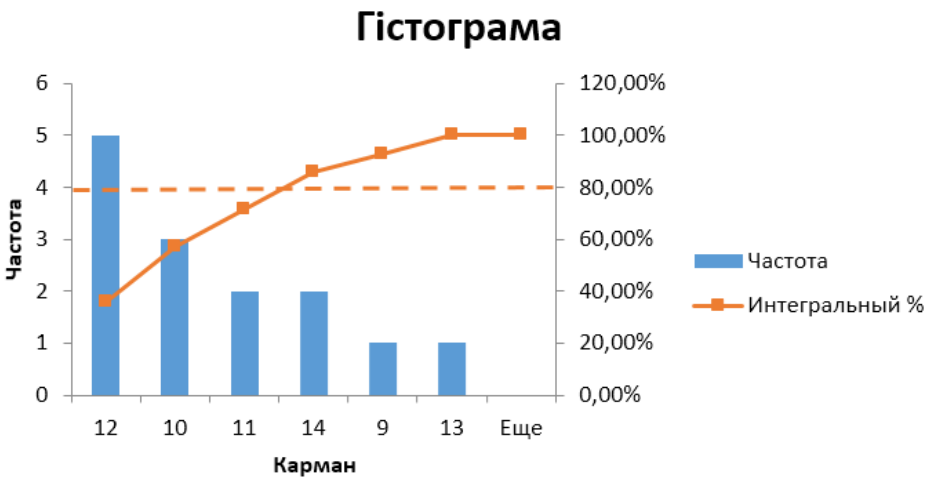


Рис. 3.25. Діаграма Парето

7. Зробимо аналіз отриманих результатів. Для цього на графіку діаграми Парето проведемо горизонтальну пунктирну лінію на рівні 80%. Це дає змогу побачити, що 80% робочого навантаження припадає на менеджерів, які залучали в день 10, 11 та 12 клієнтів – ці значення частот лежать ліворуч від точки перетину горизонтальної лінії на рівні 80% із графіком інтервального %. Інші менеджери виконали 20% робочого навантаження. Це дає можливість визначити, хто з працівників працював більш продуктивно, та враховувати це, здійснюючи оптимізацію розподілу обов'язків на фірмі та планування робочого навантаження.

3.1.9. Генерація випадкових значень за певним законом розподілу

Приклад 11. Здійснити генерацію випадкових значень за певним законом розподілу з використанням засобів Пакета аналізу MS Excel.

1. Щоб здійснити генерацію випадкових значень, потрібно в надбудові *Аналіз даних* обрати функцію генерації випадкових значень та у вікні генерації випадкових чисел вказати кількість змінних, кількість випадкових чисел, тип розподілу та його параметри (рис. 3.26).

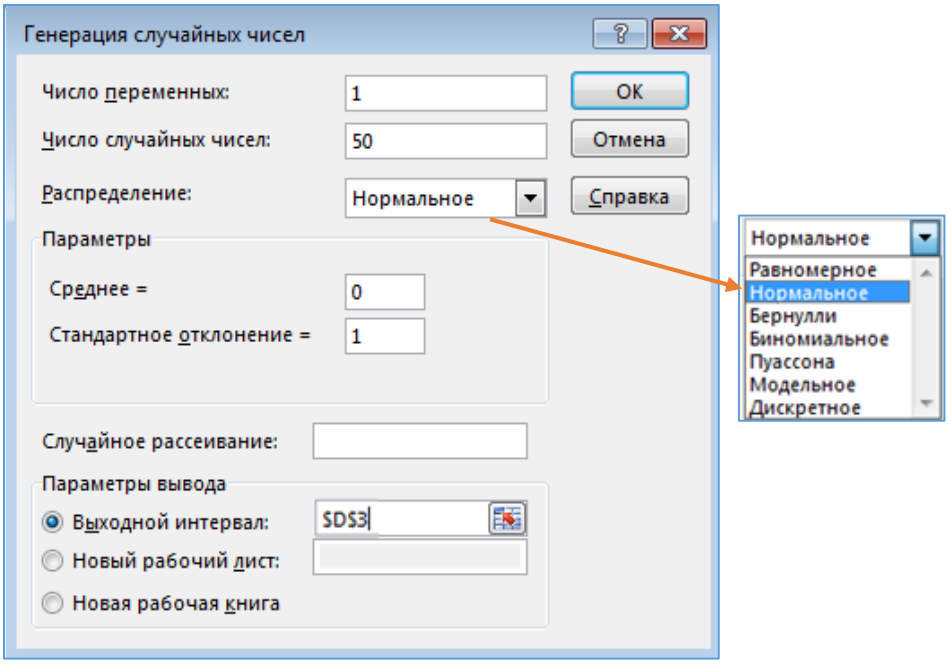


Рис. 3.26. Вікно вибору опцій генерації випадкових чисел

2. У списку, що розкривається (*Розподіл* вікна *Генерація випадкових чисел*), необхідно обрати один із законів розподілу (наприклад, Пуассона), налаштувати його параметри та згенерувати необхідну кількість випадкових чисел. Згенеровані числа з'являться на робочому аркуші MS Excel.

3.2. ЗАВДАННЯ ДЛЯ САМОСТІЙНОЇ РОБОТИ

Використовуючи можливості інструментальних засобів MS Excel для здійснення первинного статистичного аналізу даних, виконати такі завдання.

Завдання 1. Для набору даних, отриманому відповідно до варіанта (табл. 3.4):

a) побудувати інтервальний або дискретний варіаційний ряд, враховуючи тип та кількість вибіркових значень ознаки;

b) обчислити розмах варіаційного ряду, мінімальне та максимальне значення, середнє арифметичне, моду, медіану;

c) обчислити вибіркову та виправлену дисперсію, вибіркове та виправлене стандартне відхилення;

d) побудувати гістограму (полігон), емпіричну функцію розподілу, діаграму Парето;

e) провести аналіз та здійснити інтерпретацію отриманих результатів;

f) за даними побудованого варіаційного ряду вибірки перевірити відповідність розподілу генеральної сукупності значень досліджуваної ознаки нормальному закону розподілу.

Завдання 2. Згенерувати вибірки обсягом $n = 100$ та $n = 150$ таких типів: 1) нормально розподілених випадкових величин із параметрами: середнє значення $\overline{x} = M$, стандартне відхилення $\sigma = 1$; 2) рівномірно розподілених випадкових величин у діапазоні $[0, M]$ (де M – порядковий номер студента у списку журналу). Для кожної згенерованої вибірки:

a) обчислити точкові оцінки параметрів генеральної сукупності: математичного сподівання, дисперсії, середнього квадратичного відхилення, скориставшись методом описової статистики із Пакета аналізу MS Excel.

b) розрахувати інтервальні оцінки математичного сподівання.

КОНТРОЛЬНІ ПИТАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ № 3

1. Сутність первинного статистичного аналізу даних.
2. Поняття вибірки, варіаційного ряду.
3. Побудова дискретного та інтервального варіаційних рядів.
4. Графічне представлення вибірки: полігон, гістограма, емпірична функція розподілу та щільність ймовірності.
5. Характеристики випадкової величини: медіана, мода, математичне сподівання, дисперсія, стандартне відхилення.
6. Числові характеристики вибірки.
7. Точкова та інтервальна оцінка параметрів генеральної сукупності.
8. Статистичні гіпотези та критерії.
9. Основні закони розподілу неперервної та дискретної випадкової величини.
10. Ідентифікація закону розподілу за даними вибірки. Критерій згоди Пірсона.
11. Пакет аналізу даних MS Excel, описова статистика, діаграма Парето.
12. Генерація випадкових значень за певним законом розподілу засобами MS Excel.
13. Основні статистичні функції MS Excel, які застосовують при проведенні первинного статистичного аналізу даних.

Таблиця 3.4

Індивідуальні варіанти з даними для виконання завдання 1

№ варіанта	Набір даних із вибірковими значеннями досліджуваної ознаки												
1	Дані про час розв'язання завдань контрольної роботи (хв)												
	38	60	41	51	33	42	45	21	53	60	68	52	47
	49	14	57	54	41	11	47	28	48	58	32	42	58
2	Дані про тривалість роботи електронних ламп одного типу (год)												
	13,4	17,7	15,2	11,7	13,0	21,9	14,0	17,9	15,1	16,5			
	14,2	16,3	17,2	15,1	16,4	15,1	17,6	14,1	18,8	11,6			
	18,0	12,4	14,6	14,0	16,3	13,7	15,5	16,2	8,4	14,7			
3	Дані виміру ємності транзисторів (кОм)												
	1,9	3,1	1,3	0,7	3,2	1,1	2,9	1,6	2,7	4,0			
	1,7	3,2	0,9	0,8	3,1	1,2	2,6	2,7	2,3	3,2			
	4,1	1,3	2,4	4,5	2,3	0,9	1,4	1,9	2,2	3,1			
	1,5	1,1	2,3	4,3	2,1	0,7	1,2	1,5	1,8	2,9			
4	Дані про час відновлення діодів з однієї партії (наносекунди)												
	69	73	70	68	61	73	70	72	67	70	66		
	68	71	71	68	70	64	65	72	70	70	76		
	70	77	69	71	74	72	72	72	68	70	67		
5	Дані про час реакції (секунди)												
	5,3	7,1	6,0	6,2	5,3	4,4	6,0	5,8	5,4	8,2	6,9	6,5	
	8,5	6,0	6,7	6,4	2,9	7,7	6,0	5,8	6,1	5,6	4,7	5,6	
	3,8	7,4	6,7	5,6	6,1	4,5	6,8	6,5	5,8	4,2	6,1	5,4	
6	Дані про вибірку мас сталевих заготівок (гр)												
	41,6	41,7	41,8	42,2	41,2	40,9	41,3	41,5	41,7				
	41,3	41,4	41,1	41,1	41,5	42,0	42,3	41,6	42,0				
	41,8	41,4	41,3	41,2	41,1	41,6	41,9	41,2	41,5				
7	Дані про зміну меж міцності на розрив сталевого листа (МПа)												
	51,8	51,7	51,8	52,2	51,2	50,9	51,3	51,5	51,7	51,6			
	51,4	51,1	51,4	51,5	51,6	51,7	51,5	52,0	52,3	51,8			
	51,5	51,3	51,4	51,3	51,2	51,1	51,6	51,9	51,2	52,0			
8	Дані про глибину шару дифузії партії мікросхем (мкм)												
	9,8	9,8	8,6	8,6	9,2	9,4	9,8	9,0	10,0	10,0	10,1		
	9,4	9,0	11,2	10,8	9,2	9,2	9,3	10,1	9,1	8,8	9,5		

№ варіанта	Набір даних із вибірковими значеннями досліджуваної ознаки											
9	Дані про спадання напруги на діодах (вольти)											
	0,917	0,918	0,921	0,909	0,919	0,917	0,918	0,909				
	0,916	0,917	0,918	0,919	0,919	0,916	0,917	0,923				
	0,920	0,916	0,917	0,922	0,915	0,917	0,916	0,912				
10	Дані виміру маси речовини, отриманої у результаті хімічної реакції (гр)											
	14,5	17	15	14,8	17	21	22	17,5	9,2	23,3	14	14,2
	15,0	14	16	20,0	15	24	14	15,1	14,1	5,5	16	4,8
11	Дані про продуктивність роботи цеху (в умовних одиницях)											
	13,0	13,1	13,0	12,5	12,8	12,3	12,1					
	12,2	12,1	12,7	12,0	12,6	12,8	12,5					
	13,1	13,2	12,6	12,4	13,0	12,9	13,2					
12	Дані про відхилення контрольного розміру деталей від номінального значення (мкм)											
	-11	10	-6	5	-4	-5	15	-9	21	7	-3	6
	4	5	4	-10	-15	2	1	-3	2	-2	3	-4
13	Дані про питомий опір у партії мікросхем для легування полікремнію (Ом·м)											
	52	33	76	32	49	32	191	112				
	32	71	33	69	92	48	16	50				
14	Дані про середню кількість членів родини											
	5	5	6	3	2	5	6	5				
	6	6	4	3	3	5	7	3				
	5	4	5	6	4	4	4	4				
	4	3	5	3	7	4	6	5				
	4	4	6	7	6	3	3	6				
15	Дані про врожайність пшениці (ц/га)											
	27,1	18,2	16,3	22,0	24,3	24,8	33,0	27,3				
	28,5	18,0	19,5	28,1	29,5	26,7	28,4	29,6				
	25,3	15,1	31,0	19,8	25,1	23,5	20,2	25,1				
	23,7	27,0	20,4	24,0	26,0	22,9	19,9	27,0				
	22,8	23,9	24,5	23,1	21,1	22,6	25,8	23,8				
16	Дані про тарифні розряди робітників цеху заводу											
	3	5	6	3	2	4	3	5	5	6		
	4	3	2	3	4	5	4	2	4	6		
	5	3	4	5	4	3	3	6	2	3		
	2	6	3	4	5	3	4	4	5	4		
17	Дані про обсяги виробництва за категоріями (2019/2020, %)											
	101,8	110	104,1	98,7	105	106,4	102	102	101,8	102,5		
	100,4	101,8	103,0	106,1	102,6	101,7	99,7	100	98,8	104		
18	Дані про процентні ставки по кредитах (ломбардний кредит, %)											
	54,1	64,4	31,4	54,4	61,1							
	70	62,3	52,7	60,9	30,6							
	52,7	50	51,8	59,7	44							
19	Дані про середній зріст населення (см)											
	174	163	170	175	182	173	152	158				
	152	163	181	159	148	146	155	178				
	168	163	177	179	164	161	180	172				
	198	153	167	155	179	152	156	159				
	151	169	184	181	163	168	159	157				
20	Дані про важливість результатів за 10-бальною шкалою											
	8	1	5	3	3	7	7	3	2	4		
	4	1	1	5	1	7	3	1	8	5		

№ варіанта	Набір даних із вибірковими значеннями досліджуваної ознаки									
21	Вклад галузей у приріст ВВП (2019-2020 рр., у % до підсумку)									
	5,3	5,2	5,1	4,8	5,0	5,2				
	5,5	4,9	5,04	5,1	5,41	5,5				
	5,61	5,7	5,06	5,8	5,8	6				
22	Дані про випуск продукції підприємством (тис. од.)									
	30,5	33,2	22,2	30	30,2	25,8	25	25,2	28,6	28,8
	33,4	35,2	31,5	27,8	29,5	29	33,6	28,4	32,1	26,8
23	Дані про вартість основних фондів підприємства (млн. грн)									
	9,4	6,3	10,0	15,0	8,2	7,3	9,2	5,8	8,7	8,0
	5,2	13,2	8,1	7,5	11,8	14,6	8,5	7,8	10,5	6,0
	5,1	6,8	8,3	7,7	7,9	9,0	10,1	8,0	12,0	14,0
	8,2	9,8	13,5	12,4	5,5	7,9	9,2	10,8	12,1	12,4
24	Дані про динаміку електроспоживання 2019-2020 рр. (ГВт/год)									
	270	191	181	177	172	174	177	181	185	
	185,5	186	190	190	192	193	201	244	244	
25	Коефіцієнти мотивації праці (умовні одиниці)									
	0,52	0,48	0,45	0,50	0,49	0,61	0,56	0,47		
	0,62	0,59	0,75	0,69	0,45	0,57	0,63	0,66		

4. ВИЗНАЧЕННЯ МІР БЛИЗЬКОСТІ МІЖ ОБ'ЄКТАМИ НАБОРУ ДАНИХ

Лабораторна робота № 4

Мета: формування знань про сутність кластерного аналізу та класифікацію даних, формування та закріплення умінь визначення мір близькості – подібності та несхожості/відстані для даних різних типів, навичок побудови матриць близькості, таблиць спряженості.

Теоретичні знання: поняття класифікації, кластерного аналізу даних. Матриця даних, поняття близькості – подібності та несхожості об'єктів. Матриці близькості, таблиці спряженості. Міри близькості для простих типів даних. Визначення мір близькості для числових та категоріальних даних. Міри близькості для бінарних атрибутів. Визначення близькості між об'єктами, представленими різними типами атрибутів та розрідженими векторами даних.

4.1. ПОНЯТТЯ БЛИЗЬКОСТІ МІЖ ОБ'ЄКТАМИ

4.1.1. Сутність класифікації та кластеризації даних

Кластерний аналіз та класифікацію даних відносять до найбільш поширених задач Data Mining.

Класифікація (англ. *Classification*) – це задача розбиття множини об'єктів (спостережень, подій) на апріорно задані групи – ***класи***, відповідно до значень змінних, які характеризують властивості цих об'єктів.

За допомогою класифікації встановлюється функціональна залежність між вхідними і дискретними вихідними змінними та здійснюється віднесення нових об'єктів до одного зі заздалегідь відомих класів. Використовують також термін *розпізнавання образів* (англ. pattern recognition), який можна вважати синонімом.

Кластеризація (англ. *Clustering*) – це задача групування множини об'єктів (спостережень, подій) на групи – ***кластери***, на основі даних, що описують сутність цих об'єктів – значень змінних, які характеризують їх властивості.

Кожен кластер або клас містить схожі об'єкти, які мають приблизно однакові властивості та суттєво відрізняються від об'єктів інших кластерів чи, відповідно, класів.

Головна відмінність кластеризації від класифікації полягає в тому, що перелік груп чітко не заданий і визначається у процесі роботи алгоритму. Задача кластеризації є логічним продовженням ідеї класифікації й застосовується часто на початкових етапах дослідження для виявлення внутрішньої структури даних, після чого застосовують інші методи та алгоритми Data Mining.

Для розв'язання задач класифікації та кластеризації існує багато методів, в основі яких лежать поняття мір близькості, відстані між об'єктами. У цій лабораторній роботі більш детально ознайомимося з цими поняттями.

4.1.2. Матриця даних, поняття подібності та несхожості об'єктів

Сукупність об'єктів набору даних та змінних, які містять значення їх характеристик, зручно представити у вигляді таблиці, кожен рядок якої відповідає одному об'єкту, а кожен стовпець – певній змінній, яка міняється від об'єкта до об'єкта (табл. 4.1).

Таблиця 4.1

Набір вхідних даних, представлений у вигляді двомірної таблиці

	Характеристики			
Об'єкти	ПІБ клієнта	Вік	Сімейний стан	Дохід
	Шевченко А. Д.	24	Одинокий	12500
	Новицька Л. І.	22	Одружена	10520
	Іванов М. С.	35	Одружений	7580
	Савицька Л. М.	45	Вдова	6900

Така таблиця містить дані для формування ***матриці даних*** (англ. *Data matrix*), в якій рядкам відповідають об'єкти, а стовпцям – змінні, що характеризують ці об'єкти:

$$X = \begin{vmatrix} x_{11} & \dots & x_{1m} \\ \dots & \dots & \dots \\ x_{n1} & \dots & x_{nm} \end{vmatrix}, \quad (4.1)$$

де x_{ij}– значення j -ї змінної для i -го об'єкта, $i \in \{1, \dots, n\}$, $j \in \{1, \dots, m\}$, n – кількість об'єктів, m – кількість змінних.

Для оцінки близькості об'єктів один до одного розраховують міри близькості між ними. ***Міри близькості*** (англ. *Proximity measures*) є математичними методами обчислення подібності чи несхожості об'єктів набору даних шляхом визначення ***мір подібності*** (англ. *Similarity measures*) або ***мір несхожості*** (англ. *Dissimilarity measures*).

Поняття подібності є протилежним до поняття несхожості у багатомірному просторі ознак, оскільки меншій мірі несхожості між об'єктами відповідає більше значення міри подібності.

Якщо об'єкти не схожі, міра подібності повертає значення, близькі до нуля або 0, а міра несхожості повертає значення, близькі до одиниці або 1. Для схожих об'єктів навпаки, міра подібності повертає значення, близькі до одиниці або 1 (якщо об'єкти однакові), а міра несхожості повертає значення, близькі до нуля або 0 (якщо об'єкти однакові).

4.1.3. Матриці близькості, таблиці спряженості

Для набору даних, атрибути яких представлені у числових (метричних) шкалах, використовують ***матрицю відстаней*** (англ. *Distance matrix*), стовпці та рядки якої відповідають об'єктам вхідної множини даних. Елементами матриці відстаней D є значення відстані d_{ij} між об'єктами, які розташовані у рядку i та стовпці j. Матриця відстаней є симетричною відносно головної діагоналі, яка містить нулі:

$$D = \begin{pmatrix} 0 & d_{12} & \dots & d_{1n} \\ d_{21} & 0 & \dots & d_{2n} \\ \dots & \dots & \dots & \dots \\ d_{n1} & d_{n2} & \dots & 0 \end{pmatrix} \quad (4.2)$$

Матриця відстані є ***матрицею несхожості*** (англ. *Dissimilarity matrix*) розмірністю $n \times n$ (де n – кількість об'єктів набору даних), оскільки її елементи – відстані між об'єктами, є їх мірами несхожості.

Є ряд задач аналізу даних, у яких при проведенні класифікації чи кластеризації будують ***матрицю подібності*** (англ. *Similarity matrix*), яка також є симетричною відносно головної діагоналі і має розмірність $n \times n$, а її головна діагональ містить одиниці:

$$S = \begin{pmatrix} 1 & s_{12} & \dots & s_{1n} \\ s_{21} & 1 & \dots & s_{2n} \\ \dots & \dots & \dots & \dots \\ s_{n1} & s_{n2} & \dots & 1 \end{pmatrix} \quad (4.3)$$

Елементами матриці подібності S є значення мір подібності s_{ij} між об'єктами, які розташовані у рядку i та стовпці j.

Спосіб визначення мір близькості між об'єктами залежить від типу атрибутів набору даних та шкал, у яких їх вимірюють. У разі визначення за певним методом міри несхожості $d_{ij} = d(x_i, x_j)$ між двома об'єктами x_i і x_j, міра подібності цих об'єктів $s_{ij} = s(x_i, x_j)$ може бути розрахована таким чином: $s_{ij} = 1 - d_{ij}$. Наведена формула справедлива для мір несхожості, які знаходяться у діапазоні [0, 1].

Матриця несхожості та матриця подібності є різними видами *матриці близькості* (англ. *Proximity matrix*), елементами якої є міри близькості між об'єктами – міри несхожості (відстані) або міри подібності. Матрицю близькості використовують при реалізації алгоритмів класифікації та кластеризації даних.

Якщо у алгоритмі використовують відстань між об'єктами або інші міри несхожості, то в основі віднесення об'єктів до одного класу чи кластеру лежить знаходження мінімуму відстаней (мір несхожості) між ними. У разі використання в алгоритмі мір подібності, основою об'єднання об'єктів в одному класі чи кластері є знаходження максимуму мір подібності між ними.

Для визначення близькості між об'єктами, ознаки яких вимірюються у категоріальних шкалах, використовують ***таблицю спряженості*** (англ. *Contingency table*). Кожен рядок таблиці спряженості відповідає можливим значенням однієї змінної, а стовпець – іншої. У комірках таблиці *містяться частоти* – число об'єктів, у яких виявлено відповідне поєднання значень змінних.

Приклад таблиці спряженості, побудованої при проведенні дослідження ставлення юнаків та дівчат до певної телевізійної програми, наведено у таблиці 4.2.

Таблиця 4.2

Приклад таблиці спряженості для дихотомічних шкал

Показник	Подобається, $j = 1$	Не подобається, $j = 2$	Разом
Юнаки, $i = 1$	35	25	60
Дівчата, $i = 2$	14	26	40
Разом	49	51	100

4.2. МІРИ БЛИЗЬКОСТІ ДЛЯ РІЗНИХ ТИПІВ ДАНИХ

4.2.1. Міри близькості для простих типів даних

Міри близькості двох об'єктів x_i і x_j відносного одного простого категоріального або числового атрибута можуть бути розраховані за формулами, представленими у таблиці 4.3.

Таблиця 4.3

Міри близькості для простих атрибутів

Тип шкали (атрибуту)	Міра несхожості	Міра подібності
Номінальна шкала	$d(x_i, x_j) = \begin{cases} 0, & x_i = x_j \\ 1, & x_i \neq x_j \end{cases}$	$s(x_i, x_j) = \begin{cases} 1, & x_i = x_j \\ 0, & x_i \neq x_j \end{cases}$
Порядкова шкала	$d(x_i, x_j) = \dfrac{\lvert x_i - x_j \rvert}{n-1}$ (значення атрибуту представлені цілими числами з проміжку $\{0, \ldots, n-1\}$, де n – кількість значень атрибуту)	$s(x_i, x_j) = 1 - d(x_i, x_j)$
Числова (інтервальна та шкала відношень)	$d(x_i, x_j) = \lvert x_i - x_j \rvert$	$s(x_i, x_j) = \dfrac{1}{1 + d(x_i, x_j)}$, $s(x_i, x_j) = 1 - \dfrac{d(x_i, x_j) - d_{\min}}{d_{\max} - d_{\min}}$

Приклад 1. Для 3-х об'єктів (осіб) визначити міри близькості за номінальною ознакою «стать», яка може приймати два значення: «чоловіча», «жіноча» (табл. 4.4) та побудувати матриці близькості (несхожості і подібності) для заданого набору даних.

Таблиця 4.4

Значення однієї номінальної ознаки для 3-х об'єктів

Об'єкти	Ознака x_i – «стать»
1	чоловіча
2	жіноча
3	чоловіча

1. Знаходимо міри несхожості між об'єктами: $d_{1,1} = 0$ (значення змінної для одного й того ж об'єкта співпадає: $x_1 = x_1 =$ "чоловіча", тому несхожість є мінімальною), $d_{1,2} = d_{2,1} = 1$ (значення змінної для двох об'єктів

різне: $x_1 \neq x_2$, оскільки $x_1 =$ "чоловіча", а $x_2 =$ "жіноча", тому несхожість є максимальною). Аналогічно знаходимо: $d_{1,3} = d_{3,1} = 0$, $d_{2,2} = 0$, $d_{2,3} = d_{3,2} = 1$, $d_{3,3} = 0$, та будуємо матрицю несхожості між об'єктами за однією номінальною ознакою:

$$D = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}.$$

2. Враховуючи, що $s(x_i, x_j) = 1 - d(x_i, x_j)$, розраховуємо міри подібності між об'єктами:

$s_{1,1} = 1$, $s_{1,2} = s_{2,1} = 0$, $s_{1,3} = s_{3,1} = 1$, $s_{2,2} = 1$, $s_{2,3} = s_{3,2} = 0$, $s_{3,3} = 1$. Отримуємо матрицю подібності за однією номінальною ознакою:

$$S = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}.$$

Приклад 2. Для 4-х об'єктів (студентів) визначити міри близькості за порядковою ознакою „оцінка", яка приймає три значення: «задовільно», «добре», «відмінно» (табл. 4.5) та побудувати матриці близькості (несхожості і подібності) для заданого набору даних.

Таблиця 4.5

Значення однієї порядкової ознаки для 4-х об'єктів

Об'єкти	Ознака x_i – «оцінка»
1	добре
2	задовільно
3	добре
4	відмінно

1. Поставимо у відповідність значенням порядкового атрибуту цілі числа:

«задовільно» → 0, «добре» → 1, «відмінно» → 2.

2. Знаходимо міри несхожості між парами об'єктів відповідно до формули, наведеної у таблиці 3.3 для порядкових атрибутів, враховуючи, що кількість значень атрибуту $n = 3$:

$d_{1,1} = \frac{|1-1|}{3-1} = \frac{0}{2} = 0$, для 1-го об'єкта значення змінної: «добре» → 1;

$d_{2,2} = \frac{|0-0|}{2} = 0$, для 2-го об'єкта значення змінної: «задовільно» → 0;

$d_{3,3} = \frac{|1-1|}{2} = 0$, для 3-го об'єкта значення змінної: «добре» → 1;

$d_{4,4} = \frac{|2-2|}{2} = 0$, для 4-го об'єкта значення змінної: «відмінно» → 2;

$d_{1,2} = d_{2,1} = \frac{|1-0|}{2} = 0{,}5$, оскільки для 1-го об'єкта значення змінної: «добре» → 1, а для 2-го об'єкта значення змінної: «задовільно» → 0.

Аналогічно маємо: $d_{1,3} = d_{3,1} = \frac{|1-1|}{2} = 0$, $d_{1,4} = d_{4,1} = \frac{|1-2|}{2} = 0{,}5$,

$d_{2,3} = d_{3,2} = \frac{|0-1|}{2} = 0{,}5$, $d_{2,4} = d_{4,2} = \frac{|0-2|}{2} = 1$, $d_{3,4} = d_{4,3} = \frac{|1-2|}{2} = 0{,}5$.

3. Розраховані значення дають можливість побудувати матрицю несхожості між об'єктами за однією порядковою ознакою:

$$D=\begin{pmatrix} 0 & 0{,}5 & 0 & 0{,}5 \\ 0{,}5 & 0 & 0{,}5 & 1 \\ 0 & 0{,}5 & 0 & 0{,}5 \\ 0{,}5 & 1 & 0{,}5 & 0 \end{pmatrix}.$$

4. Враховуючи, що $s(x_i, x_j) = 1 - d(x_i, x_j)$, матриця подібності між об'єктами за однією порядковою ознакою буде мати такий вигляд:

$$S=\begin{pmatrix} 1 & 0{,}5 & 1 & 0{,}5 \\ 0{,}5 & 1 & 0{,}5 & 0 \\ 1 & 0{,}5 & 1 & 0{,}5 \\ 0{,}5 & 0 & 0{,}5 & 1 \end{pmatrix}.$$

Приклад 3. Для 3-х об'єктів визначити міри близькості за числовою ознакою «вага», яка приймає два значення: 65 кг і 55 кг (табл. 4.6) та побудувати матриці близькості (несхожості і подібності) для заданого набору даних.

Таблиця 4.6

Значення однієї числової ознаки для 3-х об'єктів

Об'єкти	Ознака x_i – „вага"
1	65
2	55
3	65

1. Знаходимо міри несхожості між парами об'єктів відповідно до формули, наведеної у таблиці 4.3 для числових атрибутів:

$d_{1,1} = |65 - 65| = 0$, для 1-го об'єкта значення змінної «вага» $= 65$;

$d_{2,2} = |55 - 55| = 0$, для 2-го об'єкта значення змінної «вага» $= 55$;

$d_{3,3} = |65 - 65| = 0$, для 3-го об'єкта значення змінної «вага» $= 65$;

$d_{1,2} = d_{2,1} = |65 - 55| = 10$, $d_{1,3} = d_{3,1} = |65 - 65| = 0$, $d_{2,3} = d_{3,2} = |55 - 65| = 10$.

2. Розраховані значення дають можливість побудувати матрицю несхожості між 3-ма об'єктами за однією числовою ознакою:

$$D=\begin{pmatrix} 0 & 10 & 0 \\ 10 & 0 & 10 \\ 0 & 10 & 0 \end{pmatrix}.$$

3. Міри подібності для заданих числових даних будемо розраховувати за формулою:

$$s(x_i, x_j) = 1 - \frac{d(x_i, x_j) - d_{\min}}{d_{\max} - d_{\min}}.$$

Враховуючи, що d_{min}, а d_{max}, маємо матрицю подібності між об'єктами за однією числовою ознакою:

$$S=\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}.$$

4.2.2. Визначення мір близькості для числових даних

Для числових даних мірою близькості між об'єктами – мірою їх несхожості – є ***відстань,*** яка розраховується за формулами, що містять числові атрибути – характеристиками цих об'єктів.

При визначенні відстані між об'єктами кожен об'єкт розглядається як точка в багатомірному просторі, кожному виміру якого відповідає деяка змінна, яка характеризує об'єкт, а відстань є функцією від значень даних змінних.

Перед тим, як розраховувати відстань між об'єктами, потрібно впевнитися у тому, що значення різних змінних змінюються в одному діапазоні значень. Якщо це не так, то значення всіх змінних необхідно нормалізувати або стандартизувати, перетворивши їх до одного числового діапазону.

Основні формули розрахунку відстаней між об'єктами для метричних шкал представлено у таблиці 4.7. Зазначимо, що для наведених у даній таблиці формул:

x_i та x_j – i-й та j-й об'єкти набору даних, m – кількість змінних, n – кількість об'єктів набору даних,

X_i та X_j – вектори-стовпці ознак i-го та j-го об'єктів, s_{kl} – елементи коваріаційної матриці S,

$\overline{x}_k$ та $\overline{x}_l$ – середні значення ознак k та l.

Таблиця 4.7

Міри відстаней між об'єктами для метричних шкал

Метрика	Формула
Відстань Евкліда	$d_E(x_i, x_j) = \sqrt{\sum_{t=1}^{m}(x_{it} - x_{jt})^2}$
Квадрат відстані Евкліда	$d_E^2(x_i, x_j) = \sum_{t=1}^{m}(x_{it} - x_{jt})^2$
Манхеттенська відстань (відстань міських кварталів)	$d_H(x_i, x_j) = \sum_{t=1}^{m}\lvert x_{it} - x_{jt}\rvert$
Відстань Чебишева	$d_\infty(x_i, x_j) = \max_{1\le t\le m}\lvert x_{it} - x_{jt}\rvert$
Відстань Мінковського	$d_{Minc}(x_i, x_j) = \sqrt[p]{\sum_{t=1}^{m}\lvert x_{it} - x_{jt}\rvert^p}$
Степенева відстань	$d_{cm}(x_i, x_j) = \sqrt[r]{\sum_{t=1}^{m}\lvert x_{it} - x_{jt}\rvert^p}$
Відстань Махаланобіса	$d_M(x_i, x_j) = \sqrt{(X_i - X_j)^T S^{-1}(X_i - X_j)}$ $S = (s_{kl}); k \in \{1,\dots,m\}, l \in \{1,\dots,m\}$ $s_{kl} = \frac{1}{n}\sum_{t=1}^{n}(x_{tk} - \overline{x}_k)(x_{tl} - \overline{x}_l)$
Пікова відстань	$d_L(x_i, x_j) = \frac{1}{m}\sum_{t=1}^{m}\frac{\lvert x_{it} - x_{jt}\rvert}{x_{it} + x_{jt}}$
Відстань Джеффріса–Матусіти	$d_{DM}(x_i, x_j) = \sqrt{\sum_{t=1}^{m}(\sqrt{x_{it}} - \sqrt{x_{jt}})^2}$

Наведені у таблиці 4.7 відстані є ***метриками***, оскільки для них виконуються умови:

1) відстань між двома об'єктами є додатною та симетричною: $d(x_i, x_j) = d(x_j, x_i) \ge 0$;

2) нетотожні об'єкти розрізняються, відстань між об'єктами дорівнює нулю тільки тоді, коли об'єкти співпадають: $d(x_i, x_j) = 0$ при $x_i = x_j$, а якщо $d(x_i, x_j) \ne 0$, то $x_i \ne x_j$;

3) відстань між двома об'єктами не більша за суму відстаней від кожного з них до третього об'єкта, задовольняється *нерівність трикутника*: $d(x_i, x_j) \le d(x_i, x_k) + d(x_j, x_k)$.

Розглянемо особливості застосування різних формул для визначення відстані між об'єктами.

Відстань Евкліда (англ. *Euclidean distance*) порівнює близькість двох об'єктів по великому числу ознак, є найбільш поширеною й обчислюється за формулою обчислення геометричної відстані у багатомірному просторі. З геометричної точки зору евклідова відстань буде безглуздою, якщо ознаки визначені у різних одиницях. Для коригування ситуації вдаються до нормалізації кожної ознаки.

Квадрат відстані Евкліда (англ. *Squared Euclidean distance*) використовують для надання ваги більш віддаленим один від одного об'єктам.

Манхеттенська відстань (англ. *Manhattan distance*) або відстань міських кварталів (англ. *City Block distance*) дає ті ж результати, що й евклідова відстань, проте дозволяє зменшувати вплив окремих викидів (вони не підносяться до квадрату).

Відстань Чебишева (англ. *Chebyshev distance*) є корисною у випадку, коли бажають визначити два об'єкти як різні, якщо вони відрізняються по одній координаті (одному виміру), є грубою мірою, частина інформації губиться.

Зазначимо, що ***відстань Мінковського*** (англ. *Minkowski distance*) є узагальненням манхеттенської відстані та відстані Евкліда: при $p = 1$ – це манхетенська відстань, при $p = 2$ – відстань Евкліда, а при $p \to \infty$ – відстань Чебишева.

Степенева відстань є узагальненням Евклідової відстані, використовується для збільшення або зменшення ваги, яка відноситься до розмірності, для якої відповідні об'єкти сильно відрізняються. При $r = p = 2$ степенева відстань збігається з відстанню Евкліда, де r та p – параметри, які визначає користувач. Параметр p відповідає за поступове зважування різниць за окремими координатами, параметр r відповідальний за прогресивне зважування більших відстаней між об'єктами.

Відстань Махаланобіса (англ. *Mahalanobis distance*) узагальнює поняття відстані Евкліда, враховуючи кореляції між змінними набору даних. Її обчислюють у матричному вигляді, де об'єкти x_i та x_j є векторами у просторі ознак розмірністю m, а S – коваріаційна матриця розмірністю $m \times m$ (m – кількість ознак об'єкта, відібрана для аналізу). Цю формулу не можна застосовувати, якщо дисперсія хоча б однієї змінної дорівнює нулю.

Пікова відстань передбачає незалежність між випадковими змінними й є відстанню в ортогональному просторі.

Крім описаних вище відстаней – *мір несхожості*, близькість об'єктів може бути установлена також шляхом розрахунку *мір подібності*, які визначають подібність об'єктів (табл. 3.8). Для наведених у цій таблиці формул: $s(x_i, x_j)$ – міра подібності *i*-го та *j*-го об'єктів набору даних, m – кількість змінних.

Кореляція Пірсона (англ. *Pearson correlation*) є мірою подібності, яка дозволяє знайти схожість двох об'єктів за допомогою формули кореляції Пірсона, що оцінює близькість між ними за тіснотою зв'язку двох або більше ознак і може приймати значення з діапазону [-1, 1].

Косинус подібності (англ. *Cosine similarity*) є мірою подібності, яка обчислюється як косинус кута між двома об'єктами у багатомірному просторі ознак і також може приймати значення з інтервалу [-1, 1].

Ці міри подібності мають перевагу завдяки низькій складності обчислень, особливо для розрідженого простору ознак. Міри подібності часто використовують під час класифікації текстів, у рекомендаційних системах – для прогнозу намірів користувачів.

Використовують також міри несхожості, які відповідають наведеним таблиці 4.8 мірам подібності: відстань косинус та відстань кореляції. Спосіб розрахунку цих мір несхожості: одиниця мінус косинус подібності та одиниця мінус кореляція Пірсона відповідно, дає можливість використовувати їх у додатному просторі ознак.

Таблиця 4.8

Міри подібності

Міра	Формула
Кореляція Пірсона	$s_{corr}(x_i, x_j) = \dfrac{\sum_{t=1}^{m}(x_{it} - \overline{x_t})(x_{jt} - \overline{x_t})}{\sqrt{\sum_{t=1}^{m}(x_{it} - \overline{x_t})^2} \cdot \sqrt{\sum_{t=1}^{m}(x_{jt} - \overline{x_t})^2}}$
Косинус подібності	$s_{\cos}(x_i, x_j) = \dfrac{\sum_{t=1}^{m}(x_{it} \cdot x_{jt})}{\sqrt{\sum_{t=1}^{m}(x_{it}^2)} \cdot \sqrt{\sum_{t=1}^{m}(x_{jt}^2)}}$

Відстань косинус (англ. *Cosine distance*) обчислюють за формулою:

$$d_{\cos}(x_i, x_j) = 1 - s_{\cos}(x_i, x_j) = 1 - \frac{\sum_{t=1}^{m}(x_{it} \cdot x_{jt})}{\sqrt{\sum_{t=1}^{m}(x_{it}^2)} \cdot \sqrt{\sum_{t=1}^{m}(x_{jt}^2)}}. \quad (4.4)$$

Відстань кореляції (англ. *Correlation distance*) з врахуванням того, що $d_{corr}(x_i, x_j) = 1 - s_{corr}(x_i, x_j)$, обчислюють за формулою:

$$d_{corr}(x_i, x_j) = 1 - \frac{\sum_{t=1}^{m}(x_{it} - \bar{x}_t)(x_{jt} - \bar{x}_t)}{\sqrt{\sum_{t=1}^{m}(x_{it} - \bar{x}_t)^2} \cdot \sqrt{\sum_{t=1}^{m}(x_{jt} - \bar{x}_t)^2}}. \quad (4.5)$$

Розраховані таким чином міри несхожості між об'єктами є елементами *матриці несхожості D*, симетричної відносно головної діагоналі, яка містить нулі:

$$D = \begin{pmatrix} 0 & d_{12} & \dots & d_{1m} \\ d_{21} & 0 & \dots & d_{2m} \\ \dots & \dots & \dots & \dots \\ d_{n1} & d_{n2} & \dots & 0 \end{pmatrix},$$

де $d_{ij} = d(x_{i.}, x_j)$ є мірою несхожості між об'єктами, розташованими у рядку i та стовпці j.

Така матриця несхожості не буде матрицею відстаней, оскільки відстань косинус та відстань кореляції не є метриками – для них не виконується нерівність трикутника. Однак є ряд задач, у яких застосування таких мір несхожості має переваги порівняно з метричними мірами відстаней. Наприклад, під час аналізу текстових документів, де кожен вимір співпадає з окремим ***термом*** – словом чи словосполучення, оцінка подібності двох текстів різних обсягів у термінах теми з використанням метричних мір несхожості – відстаней, має більш високу розрахункову складність.

Приклад 4. Для 6-ти об'єктів, кожен з яких характеризується двома числовими ознаками (табл. 4.9), здійснити розрахунок:

1) відстаней між об'єктами 2 і 3: відстані Евкліда, квадрата відстані Евкліда, манхетенської відстані, відстані Чебишева, Пікової відстані, відстані косинус;

2) мір подібності між об'єктами 3 і 5: кореляцію Пірсона та косинус подібності.

Таблиця 4.9

Значення двох числових ознак для 6-ти об'єктів

Ознаки об'єктів	Об'єкти $x_i, i \in \{1, 2, \dots, 6\}$					
	1	2	3	4	5	6
x_{i1}	2	4	5	12	14	15
x_{i2}	8	10	7	6	6	4

1. Знаходимо відстань Евкліда: $d_{23} = \sqrt{(4-5)^2 + (10-7)^2} = \sqrt{10} = 3{,}16$.

2. Знаходимо квадрат відстані Евкліда: $d_{23} = (4-5)^2 + (10-7)^2 = 10$.

4. Знаходимо відстань Чебишева: $d_{23} = \max(|4-5|, |10-7|) = 3$.

5. Знаходимо Пікову відстань: $d_{23} = \frac{1}{2}\left(\frac{|4-5|}{4+5} + \frac{|10-7|}{10+7}\right) = \frac{1}{2}\left(\frac{1}{9} + \frac{3}{17}\right) = 0{,}144$.

6. Знаходимо відстань косинус: $d_{23} = 1 - \frac{(4 \cdot 5 + 10 \cdot 7)}{\sqrt{4^2 + 10^2} \cdot \sqrt{5^2 + 7^2}} = 1 - \frac{90}{\sqrt{116} \cdot \sqrt{74}} = 0{,}029$.

7. Знаходимо косинус подібності: $s_{35} = \dfrac{(5\cdot 14 + 7\cdot 6)}{\sqrt{5^2+7^2}\cdot\sqrt{14^2+6^2}} = \dfrac{112}{\sqrt{74}\cdot\sqrt{232}} = 0{,}855$.

8. Знаходимо міру подібності – кореляцію Пірсона. Спочатку визначимо $\bar{x}_1 = \left(\dfrac{2+4+5+12+14+15}{6}\right) = 8{,}67$ та $\bar{x}_2 = \left(\dfrac{8+10+7+6+6+4}{6}\right) = 6{,}83$, далі маємо:

$$s_{35} = \frac{(5-8{,}67)(14-8{,}67)+(7-6{,}83)(6-6{,}83)}{\sqrt{(5-8{,}67)^2+(7-6{,}83)^2}\cdot\sqrt{(14-8{,}67)^2+(6-6{,}83)^2}} = 0{,}855.$$

4.2.3. Визначення мір близькості для категоріальних даних

Для даних, представлених категоріальними атрибутами, як *міру несхожості* об'єктів x_i і x_j використовують ***відстань Хеммінга*** (англ. *Hamming distance*), яка дорівнює кількості атрибутів, значення яких для об'єктів відрізняються і є метрикою.

Для даних, представлених категоріальними атрибутами, як *міру несхожості* використовують також ***відсоток незгоди***:

$$d(x_i, x_j) = \frac{l}{m}, \tag{4.6}$$

де m – загальна кількість категоріальних ознак, l – кількість не співпадаючих ознак об'єктів x_i і x_j, для яких $x_{ik} \neq x_{jk}, k \in \{1,.2,..,m\}$.

Міра подібності об'єктів x_i і x_j у цьому випадку визначається за формулою:

$$s(x_i, x_j) = 1 - d(x_i, x_j) = 1 - \frac{l}{m} = \frac{m-l}{m}. \tag{4.7}$$

Розрахунок мір близькості об'єктів за одним порядковим категоріальним атрибутом здійснюється за формулами, наведеними у таблиці 4.3. Порядковий атрибут має ***ранг*** – порядок, це дає можливість кожному значенню атрибуту присвоїти номер, який відповідає його порядку (див. приклад 2). Для різних порядкових ознак кількість рангів може відрізнятися, тому для розрахунку мір близькості значення порядкових атрибутів необхідно нормалізувати до діапазону [0, 1]. Це можна зробити за формулою:

$$x'_{if} = \frac{r_{if} - 1}{m_f - 1}, \tag{4.8}$$

де x'_{if} – нормалізоване значення порядкової ознаки f для об'єкта x_i, m_f – максимальне число рангів ознаки f, r_{if} – ранг ознаки f для об'єкта x_i.

Для оцінки близькості об'єктів за двома категоріальними ознаками, представленими у таблиці спряженості, використовують ***частотні міри***: χ^2 та ϕ^2. У загальному вигляді таблиця спряженості наведена у таблиці 4.10.

Таблиця 4.10

Таблиця спряженості двох категоріальних ознак

Рівні ознаки 1	Рівні ознаки 2				Разом
	1	2	…	r	
1	f_{11}	f_{12}	…	f_{1r}	n_1
2	f_{21}	f_{22}	…	f_{2r}	n_2
…	…	…	…	…	…
c	f_{c1}	f_{c2}	…	f_{cr}	n_c
Разом	m_1	m_2	…	m_r	S

У таблиці 4.10 ознака 1 може приймати c значень, ознака 2 – r значень, а f_{ij} є частотою поєднання i-го значення ознаки 1 та j-го значення ознаки 2.

Міра χ^2 розраховує близькість між об'єктами за двома ознаками таблиці спряженості з використанням формули Пірсона. Як міра несхожості використовується квадратний корінь зі значення коефіцієнта кореляції Пірсона χ^2:

$$d_{\chi^2} = \sqrt{\chi^2} = \sqrt{\sum_{i=1}^{c}\sum_{j=1}^{r}\frac{(f_{ij} - f_{ij}^t)^2}{f_{ij}^t}}, \quad (4.9)$$

де f_{ij} – спостережувана частота в комірці таблиці спряженості на перетині *i*-го рядка та *j*-го стовпця,

$f_{ij}^t = \frac{n_i m_j}{S}$ – очікувана (теоретична) частота в цій комірці.

Міра ϕ^2 є модифікацією формули Пірсона – при розрахунку близькості між об'єктами за двома ознаками таблиці спряженості проводиться нормалізація: перед добуванням квадратного кореня зі значення коефіцієнта кореляції Пірсона χ^2 він ділиться на загальну суму спостережуваних частот S:

$$d_{\phi^2} = \sqrt{\frac{\chi^2}{S}} = \sqrt{\frac{\sum_{i=1}^{c}\sum_{j=1}^{r}\frac{(f_{ij} - f_{ij}^t)^2}{f_{ij}^t}}{S}}. \quad (4.10)$$

Міра ϕ^2 змінюється від 0 до 1.

Приклад 5. Для 3-х об'єктів, кожен з яких характеризується трьома категоріальними ознаками (табл. 4.11), здійснити розрахунок мір близькості – несхожості та подібності та побудувати матрицю несхожості і матрицю подібності.

Таблиця 4.11

Значення трьох категоріальних ознак для 3-х об'єктів

Об'єкти (*студенти*)	Ознаки		
	«Рейтинг» (*бали*)	«Стипендія» (*нарахована чи ні*)	«Проживання у гуртожитку» (*так чи ні*)
1	96	Так	Ні
2	87	Так	Так
3	83	Ні	Ні

1. Знаходимо міри несхожості між парами об'єктів відповідно до формули 4.6, враховуючи, що загальна кількість категоріальних ознак $m = 3$:

$$d_{1,1} = \frac{0}{3} = 0,\ d_{2,2} = \frac{0}{3} = 0,\ d_{3,3} = \frac{0}{3} = 0,$$

$$d_{1,2} = d_{2,1} = \frac{2}{3} = 0{,}67,\ d_{1,3} = d_{3,1} = \frac{2}{3} = 0{,}67,\ d_{2,3} = d_{3,2} = \frac{3}{3} = 1.$$

3. Розраховані значення дають можливість побудувати матрицю несхожості D: $D = \begin{pmatrix} 0 & 0{,}67 & 0{,}67 \\ 0{,}67 & 0 & 1 \\ 0{,}67 & 1 & 0 \end{pmatrix}$.

4. Враховуючи, що $s(x_i, x_j) = 1 - d(x_i, x_j)$, матриця подібності буде мати такий вигляд: $S = \begin{pmatrix} 1 & 0{,}33 & 0{,}33 \\ 0{,}33 & 1 & 0 \\ 0{,}33 & 0 & 1 \end{pmatrix}$.

Приклад 6. Для 25 об'єктів – студентів однієї групи, кожен з яких характеризується двома категоріальними ознаками, представленими у таблиці спряженості (табл. 4.12), визначити частотні міри χ^2 та ϕ^2.

Таблиця 4.12

Таблиця спряженості двох категоріальних ознак для 25 об'єктів

Рівні ознаки «Оцінка»	Рівні ознаки «Проживання у гуртожитку»		Разом
	Так	Ні	
Задовільно	4	3	7
Добре	3	7	10
Відмінно	3	6	9
Разом	10	15	25

1. Розрахуємо теоретичні частоти у комірках таблиці спряженості, враховуючи, що загальна сума спостережуваних частот $S = 25$:

$$f_{11}^t = \frac{10 \cdot 7}{25} = 2{,}8,\ f_{12}^t = \frac{15 \cdot 7}{25} = 4{,}2,$$

$$f_{21}^t = \frac{10 \cdot 10}{25} = 4,\ f_{22}^t = \frac{15 \cdot 10}{25} = 6,$$

$$f_{31}^t = \frac{10 \cdot 9}{25} = 3{,}6,\ f_{32}^t = \frac{15 \cdot 9}{25} = 5{,}4.$$

2. Розрахуємо міру χ^2:

$$d_{\chi^2} = \sqrt{\chi^2} = \sqrt{\frac{(4-2{,}8)^2}{2{,}8} + \frac{(3-4{,}2)^2}{4{,}2} + \frac{(3-4)^2}{4} + \frac{(7-6)^2}{6} + \frac{(3-3{,}6)^2}{3{,}6} + \frac{(6-5{,}4)^2}{5{,}4}} = \sqrt{1{,}44} = 1{,}2 \cdot$$

2. Розрахуємо міру ϕ^2: $d_{\phi^2} = \sqrt{\frac{\chi^2}{S}} = \sqrt{\frac{1{,}44}{25}} = 0{,}24.$

4.2.4. Міри близькості для бінарних атрибутів

Бінарними атрибутами зазвичай представляють дані, які вимірюються у дихотомічній шкалі. Будь-який категоріальний атрибут можна також бінарізувати – перетворити у набір двійкових атрибутів, здійснивши *dummy-кодування*. Детально це було розглянуто у параграфі 2.3.4 лабораторної роботи №2. Номінальні атрибути можна кодувати за допомогою асиметричних двійкових атрибутів шляхом створення нового двійкового атрибута для кожного зі значень номінального атрибуту.

Для визначення близькості двох об'єктів x_i та x_j, описаних набором m бінарних ознак, які можуть приймати двійкові значення 0 або 1, будують таблицю спряженості, представлену у та. блиці 4.13

Таблиця 4.13

Таблиця спряженості для бінарних атрибутів

Можливі значення бінарного атрибута об'єкта x_i	Можливі значення бінарного атрибута об'єкта x_j		Разом
	1	0	
1	q	r	$q+r$
0	p	t	$p+t$
Разом	$q+p$	$r+t$	m

Для визначення міри подібності двох об'єктів із *симетричними* бінарними атрибутами використовують ***коефіцієнт простої відповідності*** (англ. *SMC – Simple matching coefficient*):

$$s_{SMC}(x_i, x_j) = \frac{q+t}{q+p+r+t} = \frac{q+t}{m}, \quad (4.11)$$

де q – кількість бінарних ознак, які рівні 1 для обох об'єктів;

t – кількість бінарних ознак, які рівні 0 для обох об'єктів;

p – кількість бінарних ознак, які для об'єкта x_i рівні 0, а для об'єкта x_j рівні 1;

r – кількість бінарних ознак, які об'єкта x_i рівні 1, а для об'єкта x_j рівні 0.

Для визначення відповідної міри несхожості (англ. *SMD – Simple matching distance*) двох об'єктів із *симетричними бінарними атрибутами* загальною кількістю m із врахуванням того, що $d_{SMD}(x_i, x_j) = 1 - s_{SMC}(x_i, x_j)$, розраховують за формулою:

$$d_{SMD}(x_i, x_j) = \frac{r+p}{q+p+r+t} = \frac{r+p}{m}. \quad (4.12)$$

Для *асиметричних двійкових атрибутів*, у яких бінарні значення, рівні одиниці є більш важливими (наприклад, результат тесту на хворобу), кількість збігів бінарних ознак, які рівні 0 для обох об'єктів x_i і x_j є маловажливою і нею можна знехтувати. Тому для асиметричних бінарних атрибутів *міра несхожості* між двома об'єктами може бути розрахована за формулою:

$$d_{JD}(x_i, x_j) = \frac{r+p}{q+p+r}. \quad (4.13)$$

Міра подібності двох об'єктів із асиметричними бінарними атрибутами може бути визначена відповідно таким чином:

$$s_{JC}(x_i, x_j) = 1 - d_{JD}(x_i, x_j) = 1 - \frac{r+p}{q+p+r} = \frac{q}{q+p+r}. \quad (4.14)$$

Остання формула досить часто застосовується й має назву – ***коефіцієнт Жаккара*** (англ. *Jaccard coefficient*).

Приклад 7. Знайти міри близькості двох об'єктів x_i і x_j, описаних набором 10 симетричних бінарних атрибутів:

$$x_i = \{1,0,0,0,0,1,0,0,0,1\}, \quad x_j = \{0,0,0,0,0,1,0,1,0,0\}.$$

1. Будуємо таблицю спряженості (табл. 3.14). Маємо:

$q = 1$ – кількість бінарних ознак, які рівні 1 для обох об'єктів;

$t = 6$ – кількість бінарних ознак, які рівні 0 для обох об'єктів;

$p = 1$ – кількість бінарних ознак, які для об'єкта x_i рівні 0, а для об'єкта x_j рівні 1;

$r = 2$ – кількість бінарних ознак, які для об'єкта x_i рівні 1, а для об'єкта x_j рівні 0.

Таблиця 4.14

Таблиця спряженості для 10 симетричних бінарних атрибутів 2-х об'єктів

	$x_j = 1$	$x_j = 0$	Разом
$x_i = 1$	1	2	3
$x_i = 0$	1	6	7
Разом	2	8	10

2. Розраховуємо коефіцієнт простої відповідності (міру подібності) об'єктів x_i і x_j:

$$s_{SMC}(x_i, x_j) = \frac{q+t}{m} = \frac{1+6}{10} = 0{,}7.$$

3. Розраховуємо відповідну міру несхожості об'єктів x_i і x_j:

$$d_{SMD}(x_i, x_j) = 1 - s_{SMC}(x_i, x_j) = 1 - 0{,}7 = 0{,}3.$$

Приклад 8. Знайти міри близькості об'єктів x_i і x_j, описаних набором 10 асиметричних бінарних атрибутів:

$$x_i = \{1,0,0,0,0,0,0,0,0,0\}, \quad x_j = \{0,0,0,0,0,0,1,0,0,1\}.$$

1. Будуємо таблицю спряженості (табл. 4.15).

2. Маємо: $q = 0, t = 7, p = 2, r = 7$.

3. Розраховуємо коефіцієнт Жаккара (міру подібності) об'єктів x_i і x_j:

$$s_{JC}(x_i, x_j) = \frac{q}{q+p+r} = \frac{0}{0+1+2} = 0.$$

4. Розраховуємо міру несхожості об'єктів x_i і x_j (відстань між ними):

$$d_{JD}(x_i, x_j) = 1 - s_{JC}(x_i, x_j) = 1 - 0 = 1.$$

Таблиця 4.15

Таблиця спряженості для 10 асиметричних бінарних атрибутів 2-х об'єктів

	$x_j = 1$	$x_j = 0$	Разом
$x_i = 1$	0	1	1
$x_i = 0$	2	7	9
Разом	2	8	10

4.2.5. Міри близькості об'єктів, представлених різними типами атрибутів

У разі, якщо набір даних представлено об'єктами з різними типами атрибутів, існує декілька підходів до їх обробки.

Перший підхід передбачає виконання окремого інтелектуального аналізу даних (наприклад, кластеризації) для однотипних наборів атрибутів. Такий підхід має сенс, якщо аналізи для різних типів атрибутів будуть давати сумісні результати. Однак для реальних даних малоймовірно, щоб окремі аналізи за типом атрибутів дали сумісні результати.

Тому кращим підходом є обробка даних, представлених різними типами атрибутів, виконуючи єдиний аналіз. У цьому випадку всі значущі атрибути повинні бути представлені в інтервалі [0, 1], а далі необхідно розрахувати за певним методом міри близькості для кожної пари об'єктів та побудувати матрицю близькості.

Міра несхожості між двома об'єктами x_i та x_j – $d_{ij} = d(x_i, x_j)$, може бути визначена за формулою:

$$d_{ij} = d(x_i, x_j) = \frac{\sum_{f=1}^{m} \delta_{ij}^f d_{ij}^f}{\sum_{f=1}^{m} \delta_{ij}^f}, \quad (4.15)$$

де m – кількість атрибутів (ознак), f – номер атрибута, $f \in \{0,1,\ldots,m\}$;

δ_{ij}^f– індикатор атрибута f для об'єктів x_i і x_j, який може приймати значення $\delta_{ij}^f = 0$ у випадку, якщо відсутні x_{if} чи x_{jf} (немає виміру атрибута для об'єкта x_i та x_j) або $x_{if} = x_{jf} = 0$ (атрибут f є асиметричним бінарним), інакше $\delta_{ij}^f = 1$;

d_{ij}^f – внесок атрибута f у несхожість об'єктів x_i і x_j, який розраховується залежно від його типу, за формулами несхожості для числових, номінальних, порядкових чи бінарних атрибутів, які були розглянуті нами раніше у цій лабораторній роботі.

Після визначення парних мір несхожості між усіма об'єктами набору даних будують матрицю несхожості, яка використовується у подальшому аналізі даних.

4.2.6. Міри близькості об'єктів, представлених розрідженими векторами даних

Векторна модель лежить в основі аналізу текстових документів – пошуку документів певної тематики за запитом, класифікації чи кластеризації документів.

Текстовий документ може бути представлений тисячами атрибутів, кожен з яких містить частоту певного ***терму*** – слова (наприклад, ключового) або словосполучення в документі. Це дає можливість текстовий документ розглядати як об'єкт, який є вектором частот у багатомірному просторі ознак-термів:

$$v_j = (v_{j1}, v_{j2}, \ldots, v_{jm}), \quad (4.16)$$

де v_j – векторне представлення j -го документа, v_{ji} – частота i -го терма в j -му документі,

m – загальна кількість термів для колекції документів.

Наприклад, у таблиці 4.16 представлено частоти 10 термів для 4-х різних документів. Так, у документі 1 слово «команда» зустрічається 5 разів, слово «хокей» – 3 рази, а слово «тренер» відсутнє. Виходячи з цих даних наявні документи можна представити векторами наступним чином:

$$v_1 = (5,0,3,0,2,0,0,2,0,0), \qquad v_2 = (3,0,2,0,1,1,0,1,0,1),$$
$$v_3 = (0,7,0,2,1,0,0,3,0,0), \qquad v_4 = (0,1,0,0,1,2,2,0,3,0).$$

Як правило, вектори частот є дуже довгими та розрідженими (мають багато частот, рівних 0). Для таких розріджених даних міри близькості, які були розглянуті нами раніше, не підходять. Наприклад, вектори частот термів двох документів можуть містити багато спільних значень, рівних нулю, але це не робить їх подібними, оскільки означає, що документи мають не багато спільних слів. Міра подібності векторів частот термів повинна ігнорувати нульові збіги, зосереджуючись на аналізі ненульових частот.

Таблиця 4.16

Частоти 10 термів 4-х документів

№ з/п	Терм	Документ			
		v_1 (*документ 1*)	v_2 (*документ 2*)	v_3 (*документ 3*)	v_4 (*документ 4*)
1	«команда»	5	3	0	0
2	«тренер»	0	0	7	1
3	«хокей»	3	2	0	0
4	«баскетбол»	0	0	2	0
5	«футбол»	2	1	1	1
6	«пенальті»	0	1	0	2
7	«забито»	0	0	0	2
8	«перемога»	2	1	3	0
9	«програш»	0	0	0	3
10	«сезон»	0	1	0	0

Для розрахунку мір подібності векторів частот термів використовують косинус подібності, кореляцію Пірсона (див. табл. 4.8) та ***коефіцієнт Танімото***, який можна розрахувати таким чином:

$$s_{\tan}(v_1, v_2) = \frac{\sum_{t=1}^{m}(v_{1t} \cdot v_{2t})}{\sqrt{\sum_{t=1}^{m}(v_{1t}^2)} + \sqrt{\sum_{t=1}^{m}(v_{2t}^2)} - \sum_{t=1}^{m}(v_{1t} \cdot v_{2t})}.$$

Приклад 9. Розрахуємо схожість документів v_1 і v_2 з таблиці 4.16, використовуючи міри – косинус подібності, кореляцію Пірсона, коефіцієнт Танімото.

1. Для розрахунку косинусу подібності скористаємося формулою: $s_{\cos}(v_1, v_2) = \frac{\sum_{t=1}^{10}(v_{1t} \cdot v_{2t})}{\sqrt{\sum_{t=1}^{10}(v_{1t}^2)} \cdot \sqrt{\sum_{t=1}^{10}(v_{2t}^2)}}$.

Маємо:

$$\sqrt{\sum_{t=1}^{10}(x_{1t}^2)} = \sqrt{5^2+0^2+3^2+0^2+2^2+0^2+0^2+2^2+0^2+0^2} = 6{,}48,$$

$$\sqrt{\sum_{t=1}^{10}(x_{2t}^2)} = \sqrt{3^2+0^2+2^2+0^2+1^2+1^2+0^2+1^2+0^2+1^2} = 4{,}12,$$

$$\sum_{t=1}^{10}(x_{1t} \cdot x_{2t}) = 5 \cdot 3 + 0 \cdot 0 + 3 \cdot 2 + 0 \cdot 0 + 2 \cdot 1 + 0 \cdot 1 + 0 \cdot 0 + 1 \cdot 2 + 0 \cdot 1 + 0 \cdot 1 = 25,$$

$$s_{\cos}(v_1, v_2) = \frac{25}{6{,}48 \cdot 4{,}12} = 0{,}94.$$

Таким чином, використання міри подібності косинус при порівнянні двох документів v_1 і v_2 дає результат – ці документи досить подібні.

2. Для розрахунку коефіцієнта Танімото скористаємося формулою:

$$s_{\tan}(v_1, v_2) = \frac{\sum_{t=1}^{10}(v_{1t} \cdot v_{2t})}{\sqrt{\sum_{t=1}^{10}(v_{1t}^2)} + \sqrt{\sum_{t=1}^{10}(v_{2t}^2)} - \sum_{t=1}^{10}(v_{1t} \cdot v_{2t})}.$$

Вище уже було розраховано: $\sqrt{\sum_{t=1}^{10}(x_{1t}^2)} = 6{,}48$, $\sqrt{\sum_{t=1}^{10}(x_{2t}^2)} = 4{,}12$, $\sum_{t=1}^{10}(x_{1t} \cdot x_{2t}) = 25$.

Маємо: $s_{\tan}(v_1, v_2) = \frac{25}{6{,}48 + 4{,}12 + 25} = 0{,}702$.

Отримане значення коефіцієнта Танімото свідчить, що документи є схожими.

3. Знаходимо міру подібності – кореляцію Пірсона: $s_{corr}(v_1, v_2) = \frac{\sum_{t=1}^{10}(v_{1t} - \bar{v}_t)(v_{2t} - \bar{v}_t)}{\sqrt{\sum_{t=1}^{10}(v_{1t} - \bar{v}_t)^2} \cdot \sqrt{\sum_{t=1}^{10}(x_{2t} - \bar{v}_t)^2}}$.

Спочатку визначимо:

$$\bar{v}_1 = \frac{5+3+0+0}{4} = \frac{8}{4} = 2, \quad \bar{v}_2 = \frac{0+0+7+1}{4} = \frac{8}{4} = 2,$$

$$\bar{v}_3 = \frac{3+2+0+0}{4} = \frac{5}{4} = 1{,}25, \quad \bar{v}_4 = \frac{0+0+2+0}{4} = \frac{2}{4} = 0{,}5,$$

$$\bar{v}_5 = \frac{2+1+1+1}{4} = \frac{5}{4} = 1{,}25, \quad \bar{v}_6 = \frac{0+1+0+2}{4} = \frac{3}{4} = 0{,}75,$$

$$\bar{v}_7 = \frac{0+0+0+2}{4} = \frac{2}{4} = 0{,}5, \quad \bar{v}_8 = \frac{2+1+3+0}{4} = \frac{3}{4} = 1{,}5,$$

$$\bar{v}_9 = \frac{0+0+0+3}{4} = \frac{3}{4} = 0{,}75, \quad \bar{v}_{10} = \frac{0+1+0+0}{4} = \frac{1}{4} = 0{,}25.$$

Далі маємо:

$$\sqrt{\sum_{t=1}^{10}(v_{1t} - \bar{v}_t)^2} = \sqrt{18{,}5625} = 4{,}31, \quad \sqrt{\sum_{t=1}^{10}(v_{2t} - \bar{v}_t)^2} = \sqrt{7{,}5625} = 2{,}75,$$

$$\sum_{t=1}^{10}(v_{1t} - \bar{v}_t)(v_{2t} - \bar{v}_t) = (5-2)(3-2) + (0-2)(0-2) + (3-1{,}25)(2-1{,}25) + (0-0{,}5)(0-0{,}5) +$$
$$+ (2-1{,}25)(1-1{,}25) + (0-0{,}75)(1-0{,}75) + (0-0{,}5)(0-0{,}5) + (2-1{,}5)(1-1{,}5) + (0-0{,}75)(0-0{,}75) + \quad ,$$
$$+ (0-0{,}25)(1-0{,}25) = 3 + 4 + 1{,}3125 + 0{,}25 - 0{,}1875 - 0{,}1875 + 0{,}25 - 0{,}25 + 0{,}5625 - 0{,}1875 = 8{,}5625$$

$$s_{corr}(v_1, v_2) = \frac{8{,}5625}{4{,}3084 \cdot 2{,}75} = 0{,}723.$$

Використання міри подібності – кореляції Пірсона при порівнянні документів v_1 і v_2 також дозволяє стверджувати, що ці документи є подібними.

4.3. ЗАВДАННЯ ДЛЯ САМОСТІЙНОЇ РОБОТИ

Завдання 1. Визначення мір близькості та побудова матриць близькості для простих типів даних. Використовуючи набір даних, сформований у лабораторній роботі № 2, визначити міри близькості та побудувати матриці близькості (несхожості і подібності) для простих типів даних:

1) для 3-х об’єктів за однією номінальною ознакою, яка може приймати два значення;

2) для 3-х об’єктів за однією порядковою ознакою, яка приймає два значення;

3) для 3-х об’єктів за однією числовою ознакою, яка приймає два значення.

<u>Завдання 2</u>. Визначення мір близькості для числових даних

2.1. Для 8-ми об'єктів, кожен з яких характеризується двома числовими ознаками (табл. 4.17), здійснити розрахунок між об'єктами n і k (n та k обирають за індивідуальним варіантом із таблиці 4.18):

1) відстаней: відстані Евкліда, квадрата відстані Евкліда, манхеттенської відстані, відстані Чебишева, Пікової відстані, відстані косинус;

2) мір подібності: кореляцію Пірсона та косинус подібності.

Таблиця 4.17

Значення двох числових ознак для 8-ми об'єктів

Ознаки об'єктів	Об'єкти $x_i, i \in \{1,2,\ldots,8\}$							
	1	2	3	4	5	6	7	8
x_{i1}	5	2	9	3	2	8	5	8
x_{i2}	9	1	5	8	4	7	10	7

Таблиця 4.18

Значення n та kдля завдання 2.1

Варіант	n	k	Варіант	n	k	Варіант	n	k
1	5	7	9	3	6	17	4	8
2	2	3	10	1	4	18	1	3
3	1	5	11	5	8	19	2	4
4	4	6	12	2	6	20	5	6
5	1	8	13	3	7	21	1	7
6	2	5	14	1	6	22	4	5
7	4	7	15	2	7	23	2	8
8	5	7	16	3	5	24	3	4

2.2. Із таблиці 4.17 відібрати чотири об'єкти, кожен із яких характеризується двома числовими ознаками та побудувати матрицю відстаней між ними. Номери об'єктів, які необхідно відібрати, та метрика для визначення відстані між об'єктами обираються за індивідуальним варіантом (див. табл. 4.19).

Таблиця 4.19

Значення номерів об'єктів та метрики для виконання завдання 2.2

Варіант	Номери об'єктів	Метрика	Варіант	Номери об'єктів	Метрика
1	1, 2, 3, 8	відстань Евкліда	13	1, 3, 5, 7	відстань Евкліда
2	2, 4, 5, 7	квадрат відстані Евкліда	14	1, 4, 6, 8	квадрат відстані Евкліда
3	2, 3, 5, 6	Манхеттенська відстань	15	1, 2, 4, 7	Манхеттенська відстань
4	1, 5, 6, 7	відстань Чебишева	16	1, 3, 4, 5	відстань Чебишева
5	4, 5, 6, 8	Пікова відстань	17	4, 5, 7, 8	Пікова відстань
6	1, 2, 7, 8	відстань Евкліда	18	1, 3, 5, 7	відстань Евкліда
7	2, 3, 4, 5	квадрат відстані Евкліда	19	1, 2, 5, 7	квадрат відстані Евкліда
8	3, 4, 7, 8	Манхеттенська відстань	20	4, 5, 6, 7	Манхеттенська відстань
9	1, 2, 3, 6	відстань Чебишева	21	1, 3, 7, 8	відстань Чебишева
10	1, 2, 4, 7	Пікова відстань	22	2, 4, 5, 6	Пікова відстань
11	1, 5, 6, 8	квадрат відстані Евкліда	23	1, 3, 4, 6	квадрат відстані Евкліда
12	2, 3, 6, 7	відстань Евкліда	24	1, 4, 6, 8	відстань Евкліда

<u>Завдання 3</u>. Визначення мір близькості та побудова матриць близькості і таблиць спряженості для категоріальних даних. Використовуючи набір даних, сформований у лабораторній роботі № 2, визначити:

1) міри близькості – несхожості та подібності для 3-х об'єктів, кожен з яких характеризується трьома категоріальними ознаками та побудувати матрицю несхожості і матрицю подібності;

2) для 20 об'єктів, кожен із яких характеризується двома категоріальними ознаками, побудувати таблицю спряженості та визначити частотні міри χ^2 та ϕ^2.

<u>Завдання 4</u>. Визначення мір близькості для бінарних атрибутів. Для двох об'єктів x_i і x_j, описаних набором 10 бінарних атрибутів, відповідно до індивідуального варіанта (табл. 4.20):

1) побудувати таблицю спряженості і розрахувати міру подібності – коефіцієнт простої відповідності та відповідну міру несхожості за умови, що бінарні атрибути є симетричними;

2) побудувати таблицю спряженості і розрахувати міру подібності – коефіцієнт Жаккара та відповідну міру несхожості – відстань між ними за умови, що бінарні атрибути є асиметричними.

Таблиця 4.20

Об'єкти x_i і x_j, описані набором бінарних атрибутів

Варіант	x_i	x_j	Варіант	x_i	x_j
1	{1,1,1,0,0,0,0,1}	{1,0,1,0,1,0,1,0}	13	{0,1,1,1,1,0,0,0}	{1,0,0,1,0,1,0,1}
2	{0,1,0,1,1,0,1,0}	{1,0,0,1,0,1,0,1}	14	{0,0,1,1,0,0,1,1}	{1,1,0,1,0,0,1,0}
3	{0,1,1,0,1,1,0,0}	{1,1,0,1,0,0,1,0}	15	{0,0,1,1,0,0,0,1}	{1,0,1,1,1,0,0,0}
4	{1,0,0,0,1,1,1,0}	{1,0,1,1,1,0,0,0}	16	{1,1,0,1,0,0,1,0}	{0,0,0,1,1,0,1,1}
5	{0,0,0,1,1,1,0,1}	{0,0,0,1,1,0,1,1}	17	{1,0,0,0,1,1,0,1}	{1,0,1,0,1,0,1,0}
6	{1,1,0,0,0,0,1,1}	{1,0,1,0,1,0,1,0}	18	{1,1,0,0,0,0,1,1}	{1,1,0,0,1,0,1,0}
7	{0,1,1,1,1,0,0,0}	{1,1,0,0,1,0,1,0}	19	{1,1,1,0,0,0,0,1}	{0,0,0,1,1,1,1,0}
8	{0,0,1,1,0,0,1,1}	{0,0,0,1,1,1,1,0}	20	{0,1,0,1,1,0,1,0}	{0,0,0,1,1,1,1,0}
9	{0,0,1,1,0,0,1,1}	{0,0,0,1,1,1,1,0}	21	{0,1,1,0,1,1,0,0}	{0,1,0,1,1,1,0,0}
10	{1,1,0,1,0,0,1,0}	{0,1,0,1,1,1,0,0}	22	{1,0,0,0,1,1,1,0}	{0,1,1,1,0,1,0,0}
11	{1,0,0,0,1,1,0,1}	{0,1,1,1,0,1,0,0}	23	{0,0,0,1,1,1,0,1}	{1,0,1,0,1,0,1,0}
12	{1,1,0,0,0,0,1,1}	{1,0,1,0,1,0,1,0}	24	{0,1,1,1,1,0,0,0}	{0,0,0,1,1,1,1,0}

КОНТРОЛЬНІ ПИТАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ №4

1. Сутність класифікації та кластеризації даних.
2. У чому полягає головна відмінність кластеризації від класифікації?
3. Що таке міри близькості між об'єктами?
4. Чим відрізняються міри подібності та міри несхожості об'єктів?
5. Що таке матриця близькості – відстані (несхожості) та подібності?
6. Як здійснюється побудова матриці відстані (несхожості)?
7. Як здійснюється побудова матриці подібності?
8. Як будують таблицю спряженості?
9. Як розраховують міри близькості – подібності та несхожості для простих типів даних?
10. Формули для визначення мір відстаней, мір близькості між об'єктами для числових даних.
11. Як розраховують міри несхожості та подібності для категоріальних даних?
12. Коли використовують частотні міри χ^2 і ϕ^2?
13. Як розраховують близькість об'єктів, представлених симетричними та асиметричними бінарними атрибутами?
14. Міри близькості об'єктів, представлених різними типами атрибутів.
15. Міри близькості об'єктів, представлених розрідженими векторами даних.

5. ІЄРАРХІЧНИЙ КЛАСТЕРНИЙ АНАЛІЗ ДАНИХ У ПАКЕТІ SPSS ТА MS EXCEL

Лабораторна робота № 5

Мета: закріплення знань про сутність ієрархічного кластерного аналізу, типи алгоритмів ієрархічної кластеризації, графічне представлення результатів у вигляді дендрограми. Формування умінь проведення ієрархічного кластерного аналізу засобами SPSS та MS Excel із застосуванням різних методів зв'язку кластерів.

Теоретичні знання: ієрархічний агломеративний та дивізимний алгоритми кластерного аналізу. Методи зв'язку кластерів на етапах ієрархічної кластеризації. Проведення ієрархічного кластерного аналізу, аналіз отриманих результатів, налаштування видачі інформації у пакеті SPSS та програмі MS Excel.

5.1. ОСНОВНІ ПОНЯТТЯ ІЄРАХІЧНОГО КЛАСТЕРНОГО АНАЛІЗУ

5.1.1. Типи алгоритмів ієрархічної кластеризації

У процесі здійснення ***ієрархічного кластерного аналізу*** будують систему вкладених розбиттів набору даних на кластери й отримують на виході ***дерево кластерів***, коренем якого є весь набір даних, а листи є найбільш дрібними кластерами – об'єктами набору даних.

Серед алгоритмів ієрархічної кластеризації виділяють два основних типи.

1. ***Агломеративні алгоритми*** (*висхідні*, побудова кластерів здійснюється знизу вгору) – більш поширені, на першому кроці кожен об'єкт вважається окремим кластером, а потім здійснюється об'єднання кластерів у все більші, поки всі об'єкти набору даних не будуть утворювати один кластер.

2. ***Дивізимні алгоритми*** (*спадні*, побудова кластерів здійснюється зверху вниз) – на першому кроці всі об'єкти набору даних відносять до одного кластера, який потім розбивається на все більш дрібні кластери.

Послідовність об'єднання чи розбиття кластерів графічно може бути представлена у вигляді *дендрограми*, яка у вигляді дерева відображає кластери, утворені на кожному етапі ітерації (рис. 5.1). При створенні дендрограми використовують побудовану матрицю близькості, яка містить розраховані за обраним методом попарні міри близькості між об'єктами заданого набору даних.

5.1.2. Етапи ієрархічного кластерного аналізу

Основними етапами ієрархічного кластерного аналізу є:

1. Вибір змінних для кластеризації, які відповідають характеристикам об'єктів набору даних, суттєвим для предметної області, в якій здійснюється аналіз.

2. За необхідності – здійснення нормалізації чи стандартизації значень змінних із метою приведення значень усіх змінних до єдиного числового діапазону для отримання однакового внеску їх у розрахунок мір близькості між об'єктами.

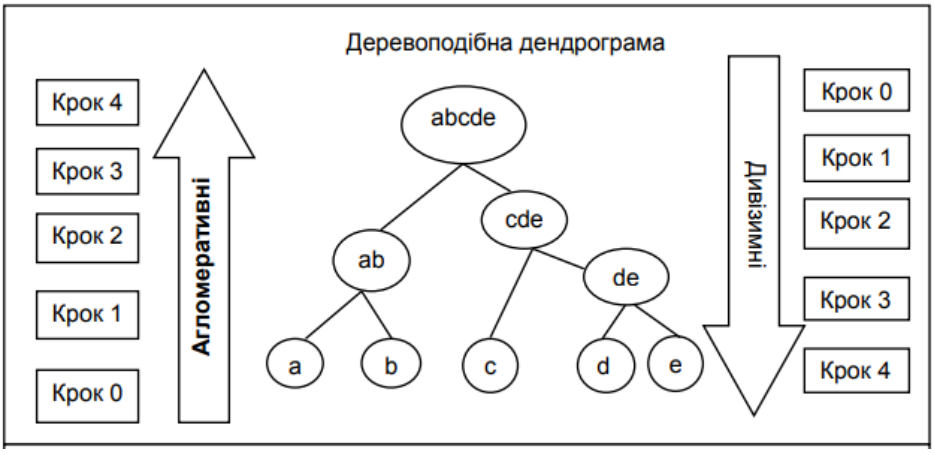


Рис. 5.1. Дендрограма алгоритмів ієрархічної кластеризації

3. Вибір методу розрахунку мір близькості між об'єктами – подібності чи несхожості (відстані).

4. Побудова матриці близькості – несхожості (відстані для метричних даних) чи подібності на основі попарно розрахованих за обраним методом мір близькості між об'єктами набору даних.

5. Вибір методу зв'язку кластерів у процесі їх об'єднання (для агломеративного алгоритму) чи поділу (для дивізимного алгоритму) та формування кластерів, який є ітеративним:

5.1. Для *агломеративного алгоритму*:

a) на першому кроці об'єднують у один кластер пару найближчих кластерів (об'єктів), порівнюючи попарні міри близькості між усіма об'єктами, які містяться у матриці близькості;

b) на наступних кроках здійснюється об'єднання найближчих кластерів за обраним методом їх зв'язку та оновлення матриці близькості доти, поки усі кластери не будуть об'єднані в один.

5.2. Для *дивізимного алгоритму*:

a) на першому кроці знаходять пару найдальших об'єктів, порівнюючи попарні міри близькості між усіма об'єктами, які містяться у матриці близькості;

b) здійснюють поділ об'єктів, що залишилися, на два кластери на основі їх близькості до одного з двох виділених на попередньому етапі об'єктів;

c) два попередні кроки повторюють доти, поки усі об'єкти не будуть поділені на кластери таким чином, що кожен об'єкт буде окремим кластером.

6. Визначення оптимальної кількості кластерів, яка буде відображати внутрішню структуру набору даних. ***Оптимальну кількість кластерів*** розраховують як різницю між кількістю об'єктів набору даних та номером етапу ітерації, на якому відстань між кластерами, які об'єднуються (діляться), змінюється стрибкоподібно.

7. Інтерпретація результатів – виявлення внутрішньої структури набору даних, що передбачає аналіз вмісту кластерів та визначення спільних характеристик об'єктів кожного кластеру. Цей етап є досить складним і залежить від мети дослідника.

5.1.3. Методи зв'язку кластерів

На етапах ітерації ієрархічного кластерного аналізу застосовують різні ***методи зв'язку*** (англ. *Linkage Methods*), які є критеріями під час об'єднання (для агломеративного алгоритму) чи поділу (для дивізимного алгоритму) кластерів.

До ***основних методів зв'язку***, що лежать в основі визначення близькості двох кластерів, відносять наступні.

1. ***Найближчого (ближнього) сусіда*** або *одиночного зв'язку* (англ. *Nearest neighbor, Single Linkage*): за відстань між кластерами приймають відстань між найближчими об'єктами цих кластерів. Результуючі кластери мають тенденцію об'єднуватися у ланцюжки.

2. ***Найбільш віддаленого (дальнього) сусіда*** або *повного зв'язку* (англ. *Furthest neighbor, Complete Linkage*): за відстань між кластерами приймають відстань між найбільш віддаленими об'єктами цих кластерів. Цей метод доцільно застосовувати у випадках, коли кластери мають вигляд сильно віддалених одна від одної скупчень.

3. ***Середнього*** або ***міжгрупового зв'язку*** (англ. *Average Linkage* або *Between-groups Linkage*):

а) *незваженого* (англ. *Unweighted Average Linkage* або *UPGMA – Unweighted Pair-Group Method using Averages*): відстань між двома різними кластерами обчислюється як середня відстань між усіма парами об'єктів у них. Метод ефективний, коли об'єкти формують різні групи, однак він працює добре й у випадках протяжних (ланцюжкового типу) кластерів;

б) *зваженого* (англ. *Weighted Average Linkage* або *WPGMA – Weighted Pair-Group Method using Averages*): відрізняється від попереднього врахуванням вагових коефіцієнтів – розмірів кластерів (кількості об'єктів у них). Цей метод доцільно застосовувати, коли передбачаються нерівні розміри кластерів.

4. ***Центроїдний метод*** (англ. *Centroid Linkage*):

а) *незважений* (англ. *Unweighted Pair-Group centroid* або *UPGMC – unweighted pair-group method using the centroid average*): відстань між двома кластерами визначається як відстань між їхніми *центрами ваги – центроїдами*, які розраховують як точки у просторі ознак набору даних із координатами, рівними середнім значенням змінних усіх об'єктів кластера. У цьому методі більший вклад у розрахунок відстані будуть вносити великі кластери.

б) *зважений* (англ. *Weighted Pair-Group centroid* або *WPGMC – weighted pair-group method using the centroid average*): відрізняється від попереднього врахуванням вагових коефіцієнтів – розмірів кластерів (кількості об'єктів у них), що робить однаковим вклад у розрахунок відстаней кластерів різних розмірів.

6. ***Медіанний метод*** (англ. *Median Linkage*): відстань між будь-яким кластером та новим кластером, отриманим у результаті об'єднання двох кластерів, визначається як відстань між цим кластером та серединою відрізка, що з'єднує об'єднані кластери.

5. ***Метод Уорда*** (англ. *Ward's Linkage*): заснований на мінімізації загальної внутрішньогрупової суми квадратів у результаті об'єднання кластерів, яка визначається як сума квадратів відстаней між усіма об'єктами у

кластері й центроїдом кластера. Метод направлений на об'єднання близько розташованих кластерів і має тенденцію до створення кластерів малого розміру.

Зауваження. У описаних вище методах зв'язку ми використовували поняття «відстані» між об'єктами набору даних та кластерами. Необхідно пам'ятати, що відстань між об'єктами є мірою несхожості – метрикою, обчисленою за однією з формул таблиці 4.7 (див. лабораторна робота № 4). У випадку використання інших мір несхожості для побудови матриці близькості при визначенні близькості кластерів ми використовуємо обрані міри, які уже не будуть відстанями. У разі використання мір подібності ми враховуємо, що найближчі кластери будуть мати найбільші міри подібності, а найдальші кластери будуть мати найменші міри подібності.

5.2. ЗДІЙСНЕННЯ ІЄРАХІЧНОГО АГЛОМЕРАТИВНОГО КЛАСТЕРНОГО АНАЛІЗУ ЗАСОБАМИ MS EXCEL

5.2.1. Побудова матриці відстаней

Приклад 1. Для 6-ти об'єктів, кожен із яких характеризується двома числовими ознаками: x – обсяг продукції, що випускається, та y – середньорічна вартість основних промислово-виробничих фондів (табл. 5.1), побудувати матрицю відстаней за метрикою – відстань Евкліда.

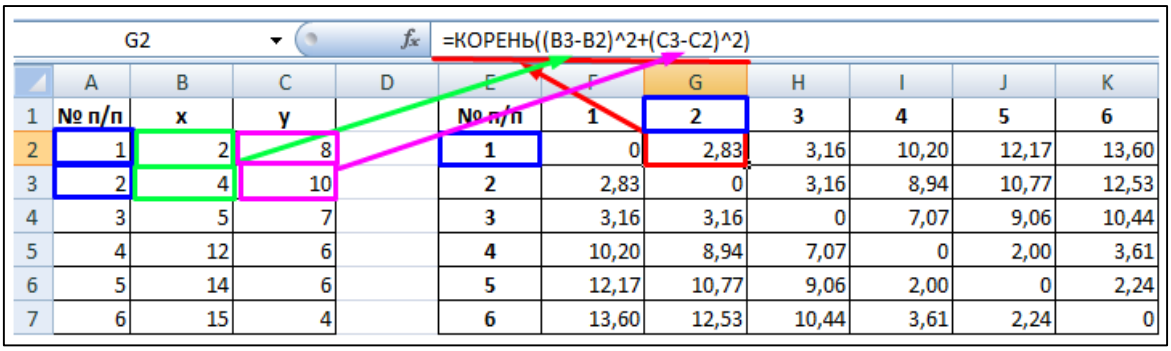
Таблиця 5.1

Значення двох числових ознак для 6-ти об'єктів

Ознаки	Об'єкти					
	1	2	3	4	5	6
x	2	4	5	12	14	15
y	8	10	7	6	6	4

1. У MS Excel у комірках A1:B7 робочого аркуша створюємо таблицю з вихідними даними (на рис. 5.2 діапазон комірок A1:C7).

2. Розраховуємо відстані між об'єктами за формулою Евкліда, розраховані дані розміщуємо у матриці відстаней (на рис. 5.2 діапазон комірок E1:K7).



	A	B	C	D	E	F	G	H	I	J	K
1	№ п/п	x	y		№ п/п	1	2	3	4	5	6
2	1	2	8		1	0	2,83	3,16	10,20	12,17	13,60
3	2	4	10		2	2,83	0	3,16	8,94	10,77	12,53
4	3	5	7		3	3,16	3,16	0	7,07	9,06	10,44
5	4	12	6		4	10,20	8,94	7,07	0	2,00	3,61
6	5	14	6		5	12,17	10,77	9,06	2,00	0	2,24
7	6	15	4		6	13,60	12,53	10,44	3,61	2,24	0

Рис. 5.2. Аркуш MS Excel із вхідними даними та матрицею відстаней

3. У побудованій матриці відстаней найбільш близькими об'єктами будуть об'єкти 4 і 5, оскільки відстань між ними дорівнює 2 й є найменшою у матриці. Тому у один кластер на першому етапі ітерації необхідно об'єднати об'єкти 4 і 5 (рис. 5.3). Подальше визначення відстаней на етапах ітерації ієрархічного алгоритму у процесі об'єднання кластерів буде залежати від обраного методу їх зв'язку.

№ п/пр	1	2	3	4	5	6
1	0	2,83	3,16	10,20	12,17	13,60
2	2,83	0	3,16	8,94	10,77	12,53
3	3,16	3,16	0	7,07	9,06	10,44
4	10,20	8,94	7,07	0	2,00	3,61
5	12,17	10,77	9,06	2,00	0	2,24
6	13,60	12,53	10,44	3,61	2,24	0

Рис. 5.3. Об'єднання кластерів на першому етапі ієрархічної кластеризації

5.2.2. Ієрархічна кластеризація: метод найближчого сусіда

Приклад 2. Використовуючи побудовану у *прикладі 1* матрицю відстаней, здійснити ієрархічну агломеративну кластеризацію з використанням методу найближчого сусіда.

1. При формуванні нової матриці відстаней стовпці 4 і 5 будуть об'єднані у одну групу і рядки 4 і 5 будуть об'єднані у одну групу. У разі використання методу найближчого сусіда при їх об'єднанні з двох значень залишають те, яке є меншим (рис. 5.4). Значення на перетині інших стовпців матриці відстаней залишаємо без змін.

№ п/пр	1	2	3	4	5	6
1	0	2,83	3,16	10,20	12,17	13,60
2	2,83	0	3,16	8,94	10,77	12,53
3	3,16	3,16	0	7,07	9,06	10,44
4	10,20	8,94	7,07	0	2,00	3,61
5	12,17	10,77	9,06	2,00	0	2,24
6	13,60	12,53	10,44	3,61	2,24	0

а)

№ п/пр	1	2	3	4, 5	6
1	0	2,83	3,16	10,20	13,60
2	2,828	0	3,16	8,94	12,53
3	3,162	3,16	0	7,07	10,44
4, 5	10,20	8,94	7,07	0	2,24
6	13,60	12,53	10,44	2,24	0

б)

а) вибір значень при об'єднанні комірок
б) матриця відстаней після об'єднання 4 і 5 об'єктів у один кластер

Рис. 5.4. Перший етап об'єднання кластерів за методом найближчого сусіда

2. У новій матриці відстаней найменше значення рівне 2,24 є відстанню між кластером (4, 5) та об'єктом 6 (рис. 5.4, *б*), тому їх можна об'єднати у один кластер, залишаючи менші значення при об'єднанні відповідних комірок (рис. 5.5, *а*). Значення на перетині інших стовпців матриці відстаней залишаємо без змін. Після об'єднання отримуємо наступну матрицю відстаней меншого розміру (рис. 5.5, *б*).

3. У новій матриці відстаней найменше значення рівне 2,83 і є відстанню між об'єктами 1 та 2 (рис. 5.5, *б*), тому їх можна об'єднати у один кластер, залишаючи менше значення у комірках при об'єднанні (рис. 5.6, *а*). Після об'єднання отримуємо наступну матрицю відстаней (рис. 5.6, *б*).

№ п/пр	1	2	3	4, 5	6
1	0	2,83	3,16	10,20	13,60
2	2,828	0	3,16	8,94	12,53
3	3,162	3,16	0	7,07	10,44
4, 5	10,20	8,94	7,07	0	2,24
6	13,60	12,53	10,44	2,24	0

а)

№ п/пр	1	2	3	4, 5, 6
1	0	2,83	3,16	10,20
2	2,83	0	3,16	8,94
3	3,16	3,16	0	7,07
4, 5, 6	10,20	8,94	7,07	0

б)

а) вибір значень при об'єднанні комірок
б) матриця відстаней після об'єднання 4,5 і 6 об'єктів у один кластер

Рис. 5.5. Другий етап об'єднання кластерів за методом найближчого сусіда

№ п/пр	1	2	3	4, 5, 6
1	0	2,83	3,16	10,20
2	2,83	0	3,16	8,94
3	3,16	3,16	0	7,07
4, 5, 6	10,20	8,94	7,07	0

а)

№ п/пр	1, 2	3	4, 5, 6
1, 2	0	3,16	8,94
3	3,16	0	7,07
4, 5, 6	8,94	7,07	0

б)

а) вибір значень при об'єднанні комірок
б) матриця відстаней після об'єднання 1 і 2 об'єктів у один кластер

Рис. 5.6. Третій етап об'єднання кластерів за методом найближчого сусіда

4. У новій матриці відстаней найменше значення рівне 3,16 і є відстанню між кластером (1, 2) та об'єктом 3 (рис. 5.6, *б*), тому їх можна об'єднати у один кластер (рис. 5.7, *а*). Після об'єднання маємо два кластери (1, 2, 3) та (4, 5, 6), відстань між якими дорівнює 0,7 (рис. 5.7, *б*).

№ п/пр	1, 2	3	4, 5, 6
1, 2	0	3,16	8,94
3	3,16	0	7,07
4, 5, 6	8,94	7,07	0

а)

№ п/пр	1, 2, 3	4, 5, 6
1, 2, 3	0	7,07
4, 5, 6	7,07	0

б)

а) вибір значень при об'єднанні комірок
б) матриця відстаней після об'єднання 1,2 і 3 об'єктів у один кластер

Рис. 5.7. Останній етап об'єднання кластерів за методом найближчого сусіда

5. На останньому етапі кластеризації кластери (1, 2, 3) та (4, 5, 6) об'єднують у один кластер (1, 2, 3, 4, 5, 6), який буде містити усі об'єкти набору даних.

6. Результати проведеної кластеризації можна представити графічно у вигляді дендрограми (рис. 5.8).

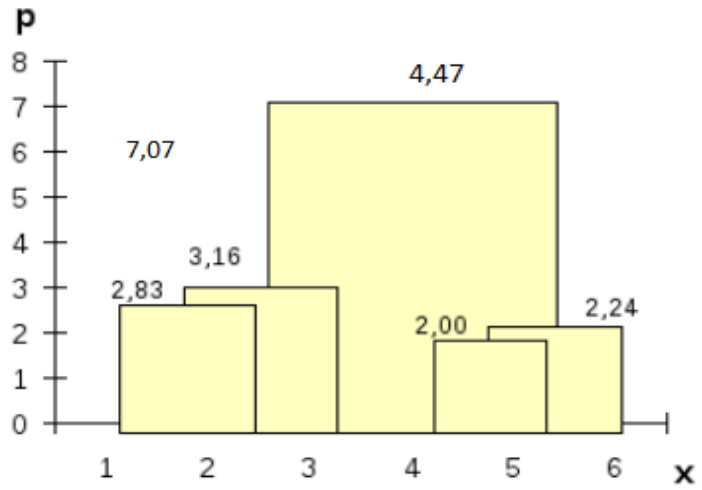


Рис. 5.8. Дендрограма для проведеної ієрархічної кластеризації

7. Аналіз дендрограми показує, що на останньому етапі спостерігається різкий скачок відстані між кластерами, які об'єднуються, порівняно з відстанями об'єднаних кластерів попередніх етапів (7,07 порівняно з 3, 2,24, 2,83, 3,16). Це дає можливість розрахувати оптимальне число кластерів, рівне різниці кількості об'єктів – 6 та номеру етапу, після якого відстань збільшується стрибкоподібно – 4. Тобто оптимальне число кластерів дорівнює 6 – 4 = 2, а склад кластерів визначаємо з даних, отриманих у п. 4 (рис. 5.7): кластер 1 містить об'єкти (1, 2, 3), а кластер 2 містить об'єкти (4, 5, 6).

5.2.3. Ієрархічна кластеризація: метод найдальшого сусіда

Приклад 3. Використовуючи побудовану у *прикладі 1* матрицю відстаней, здійснити ієрархічну агломеративну кластеризацію з використанням методу найдальшого сусіда.

1. Як було з'ясовано раніше (див. п. 2.1), найближчими об'єктами будуть об'єкти 4 і 5, оскільки відстань між ними дорівнює 2 та є найменшою у побудованій матриці відстаней. Тому при формуванні нової матриці відстаней стовпці 4 і 5 будуть об'єднані у одну групу і рядки 4 і 5 будуть об'єднані у одну групу. У разі використання методу найдальшого сусіда, на відміну від попереднього методу, при їх об'єднанні з двох значень залишають те, яке є більшим (рис. 5.9). Значення на перетині інших стовпців матриці відстаней залишаємо без змін.

2. У новій матриці відстаней найменше значення рівне 2,83 є відстанню між об'єктами 1 і 2, тому їх можна об'єднати у один кластер, залишаючи більші значення при об'єднанні відповідних комірок.

3. Наступні етапи ієрархічного алгоритму проводяться аналогічно – знаходиться найменший елемент у матриці відстаней (рис. 5.9). Рядок та стовпець, на перетині яких знаходиться цей елемент, визначають кластери, які необхідно об'єднати. При об'єднанні стовпців та рядків із двох значень залишаємо ті, які є більшими. На останньому етапі ітерації об'єднують кластери (1, 2, 3) та (4, 5, 6), утворивши один кластер (1, 2, 3, 4, 5, 6).

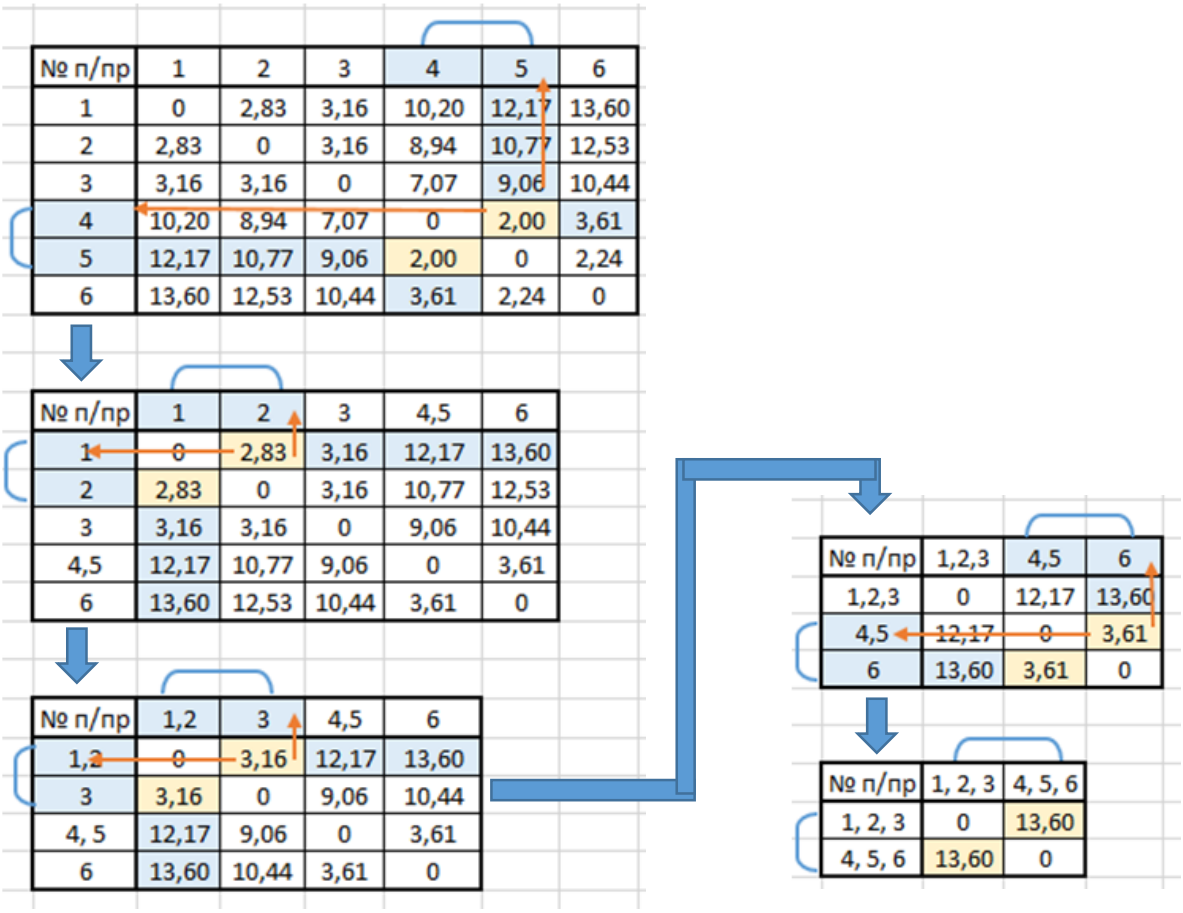


Рис. 5.9. Проведення ієрархічної кластеризації за методом найдальшого сусіда

4. Для визначення оптимального числа кластерів проаналізуємо послідовність значень відстаней між кластерами, які об'єднувалися на кожному етапі ітерації. Різкий скачок відстані між кластерами, які об'єднуються, спостерігається на останньому етапі: 13,6 порівняно з 2, 2,83, 3,16, 3,61. Тому оптимальне число кластерів буде рівним 6 – 4 = 2 (6: кількість об'єктів набору даних, 4: номер етапу, після якого відстань збільшується стрибкоподібно). Кластер 1 містить об'єкти (1, 2, 3), а кластер 2 містить об'єкти (4, 5, 6).

Отримані результати співпадають із розбиттям, яке було виконане у разі використання при об'єднанні кластерів методу найближчого сусіда (див. п. 2.2).

5.3. ЗНАЙОМСТВО З ОСНОВАМИ ЗДІЙСНЕННЯ ІЄРАРХІЧНОГО КЛАСТЕРНОГО АНАЛІЗУ ДАНИХ У СЕРЕДОВИЩІ SPSS

5.3.1. Постановка задачі та здійснення налаштувань

Завдання 1. Провести ієрархічний кластерний аналіз даних про процентний склад робітників у різних галузях промисловості європейських країн із метою знаходження країн зі схожими властивостями.

Дані для виконання лабораторної роботи зберігаються у файлі *lab_spss.sav* і містять відомості про зайнятість у різних галузях промисловості європейських країн у 1979 році (http://www.dm.unibo.it/~simoncin/EuropeanJobs.html). Тому розбиття на кластери країн зі схожими властивостями буде відображати стан економічного розвитку країн для вказаного періоду часу.

1. Відкрийте файл *lab_spss.sav* у програмі SPSS, доступний за посиланням: https://drive.google.com/file/d/1DWLYEgk5dCi_uJS3JPz0aGo4olM2R5sg/view.

2. Перейдіть у вікні *Редактора даних/Data Editor* на вкладку *Представлення Змінні/Variable View* та ознайомтеся з характеристиками країн, які будуть аналізуватися (рис. 5.10).

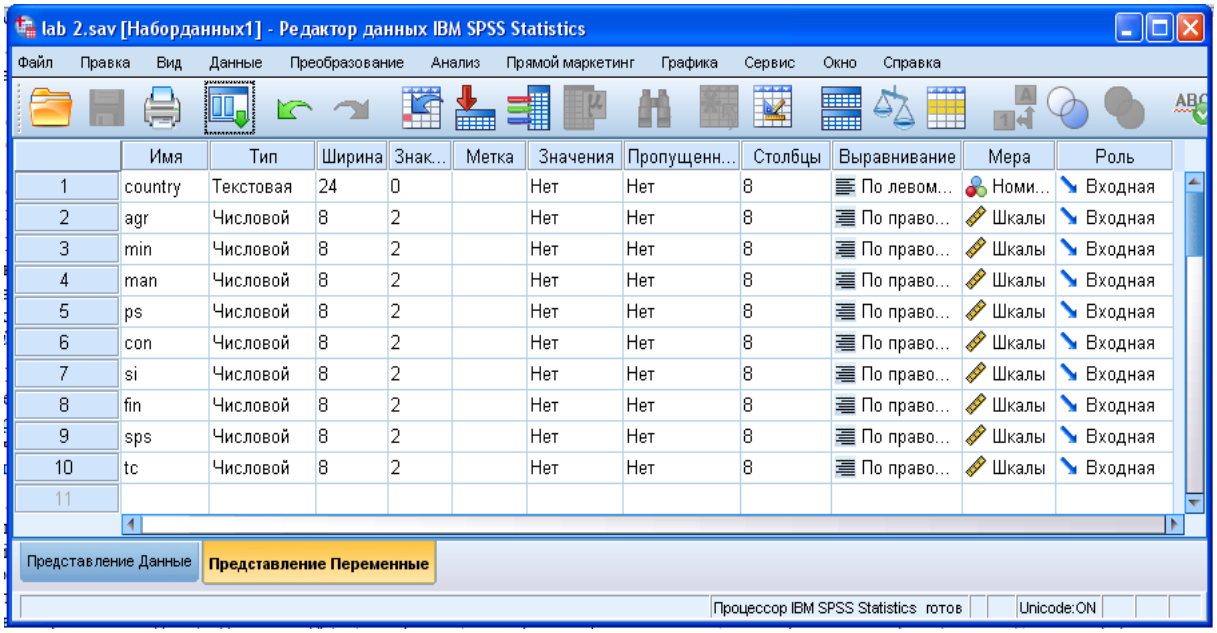


Рис. 5.10. Редактор даних/Data Editor SPSS, вкладка Представлення Змінні/Variable View

Для представлення даних – відсотка робочих категорій окремої галузі – у файлі використано наступні числові змінні:

agr – відсоток зайнятих у сільському господарстві;
min – відсоток зайнятих у гірничодобувній галузі;
man – відсоток зайнятих у обробної промисловості;
ps – відсоток зайнятих у енергетичної галузі;
con – відсоток зайнятих у будівництві;
si – відсоток зайнятих у сфері обслуговування;
fin – відсоток зайнятих у фінансовій сфері;
sps – відсоток зайнятих у соціальних службах;
tc – відсоток зайнятих у транспорті і комунікаціях;
country – назва країни (текстова змінна).

3. Перейдіть у вікні *Редактора даних/Data Editor* SPSS на вкладку *Представлення Дані/Data View* та перегляньте значення змінних, до яких буде застосовано кластерний аналіз (рис. 5.11).

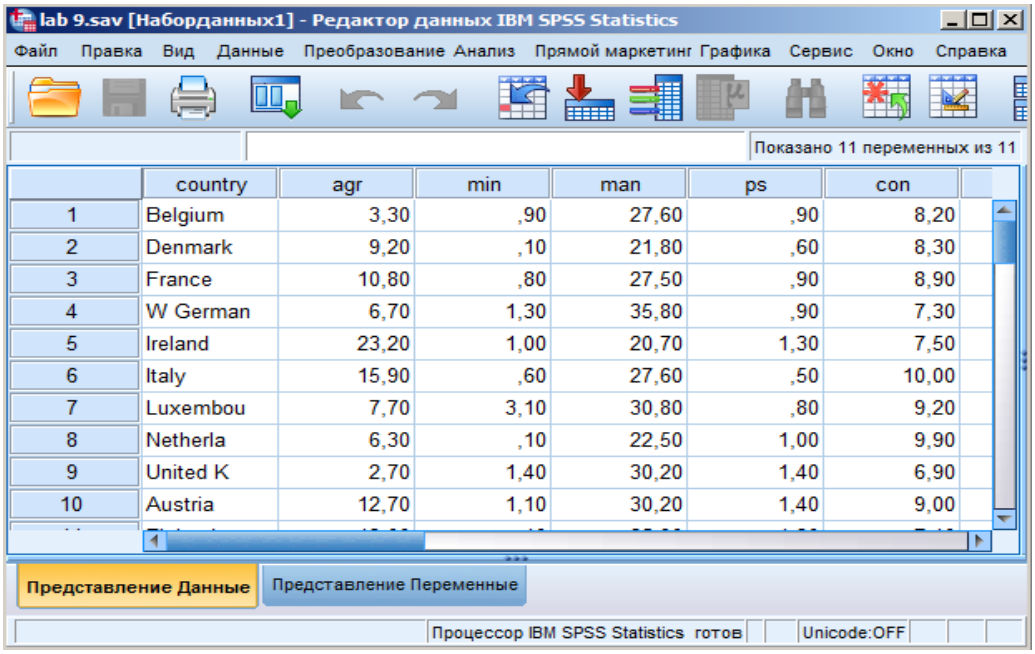


Рис. 5.11. Редактор даних/Data Editor SPSS, вкладка Представлення Дані/Data View

4. Для здійснення ієрархічної класифікації у пункті меню *Аналіз/Analyze* оберіть *Класифікація/Classify – Ієрархічна кластеризація/Hierarchical Cluster*.

5. У діалоговому вікні, що відкриється (*Ієрархічний кластерний аналіз/Hierarchical Cluster Analysis*) (рис. 5.12), перемістіть:

a) числові змінні, що тестуються: *agr-tc* у поле *Змінні/Variable(s)*;

b) текстову змінну *country* (країна) використайте для позначення (маркування) спостережень і перенесіть у поле з ім'ям *Мітить спостереження значеннями/Label cases by.*

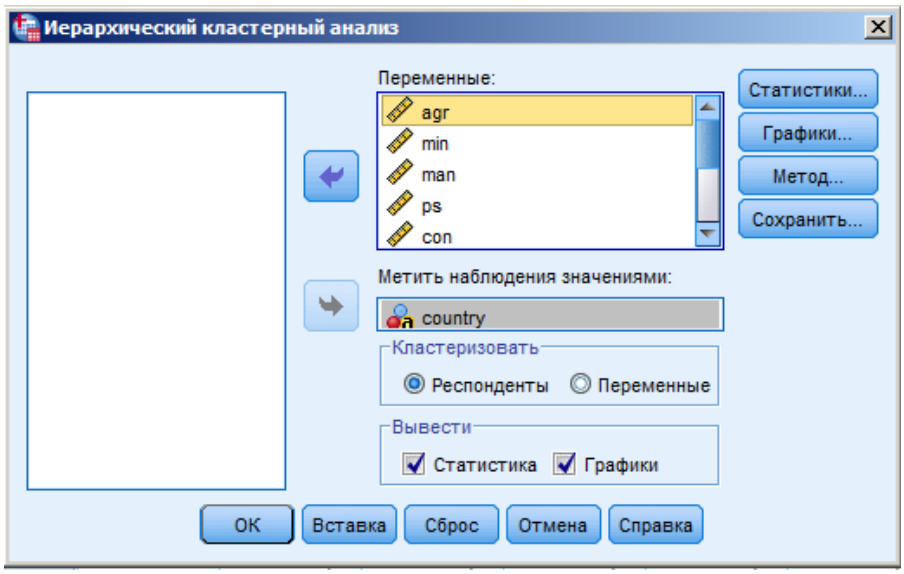


Рис. 5.12. Вікно Ієрархічний кластерний аналіз/Hierarchical Cluster Analysis

6. Натисніть кнопку *Графіки/Plots* та у вікні *Ієрархічний кластерний аналіз: Графіки/Hierarchical Cluster Analysis: Plots* активуйте опцію виводу *Дендрограма/Dendrogram* – деревоподібної діаграми і за допомогою опції *Ні/None* в області *Сосульчата діаграма/Icicle* скасуйте видачу накопичувальної діаграми (рис. 5.13).

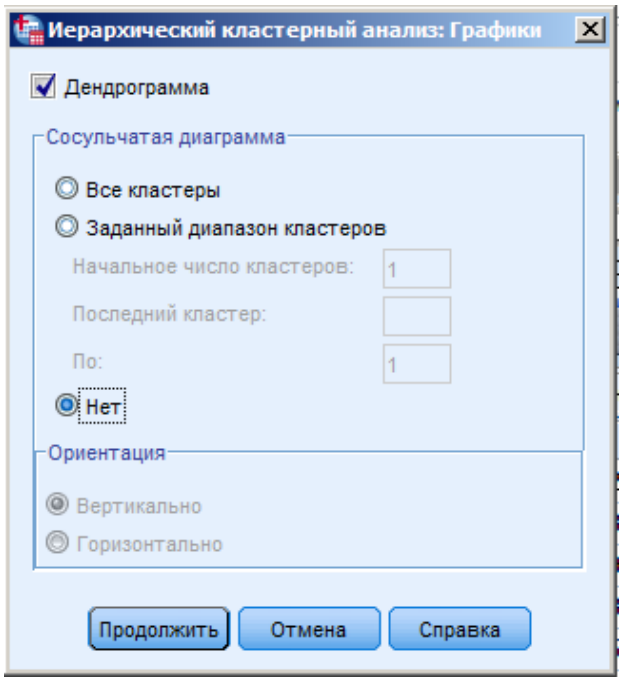


Рис. 5.13. Вікно Ієрархічний кластерний аналіз: Графіки/Hierarchical Cluster Analysis: Plots

7. Натисніть кнопку *Продовжити/Continue* та поверніться у вікно *Ієрархічний кластерний аналіз/Hierarchical Cluster Analysis*.

8. Натисніть кнопку *Метод/Method* та перейдіть до вікна *Ієрархічний кластерний аналіз:Метод/Hierarchical Cluster Analysis:Method* для вибору методу об'єднання кластерів *(Cluster Method)* та способу розрахунку міри відстані *(Measure)* (рис. 5.14).

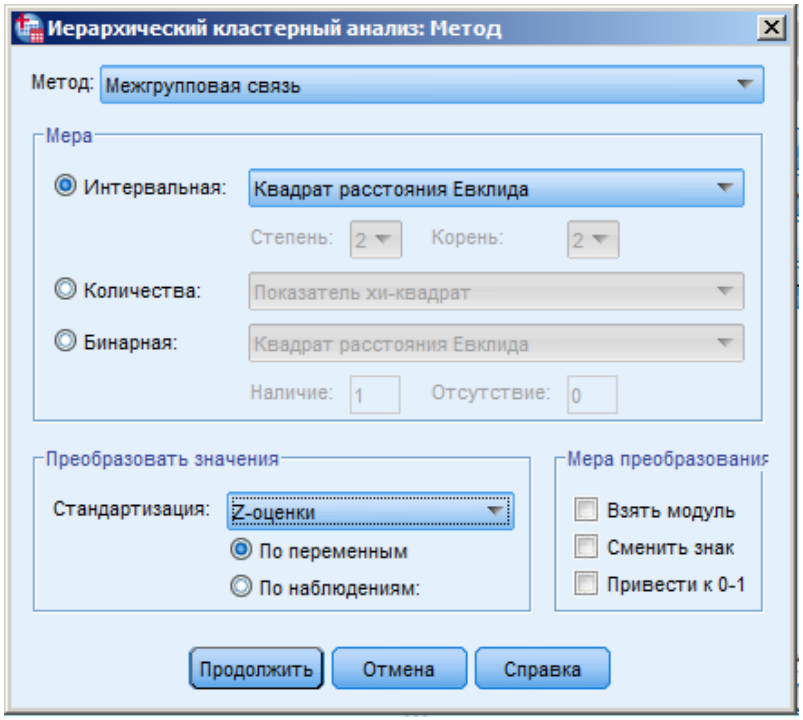


Рис. 5.14. Вікно Ієрархічний кластерний аналіз: Метод/Hierarchical Cluster Analysis: Method

9. Залиште попередні установки у полях *Метод/Cluster Method* та *Міра/Measure*: міжгруповий зв'язок та квадрат відстані Евкліда – вони установлені за замовчуванням.

10. У полі *Перетворити значення/Transform Values* установіть стандартизацію значень змінних: *z-оцінки/z-scores*. Вона потрібна для приведення значень змінних до одного числового діапазону.

11. Натисніть кнопку *Продовжити/Continue*, поверніться у вікно *Ієрархічний кластерний аналіз/Hierarchical Cluster Analysis* і почніть розрахунок, натиснувши *ОК*.

12. У вікні виводу буде представлено результати проведеного кластерного аналізу.

13. Після звичайного статистичного зведення підсумків за спостереженнями (рис. 5.15) буде представлена таблиця, яка відображає порядок агломерації (рис. 5.16) та дендрограма (рис. 5.17).

5.3.2. Аналіз отриманих результатів

Завдання 2. Здійснити аналіз результатів проведеного кластерного аналізу.

1. Зробимо аналіз результатів, представлених у таблиці *Agglomeration Schedule/Порядок агломерації* (рис. 5.16). Дані, представлені у таблиці, дозволяють з'ясувати черговість побудови кластерів, а також їхню оптимальну кількість. У двох стовпчиках, розташованих під загальною шапкою *Cluster Combined/Об'єднаний кластер*, можна побачити, що:

A. На першому кроці були об'єднані об'єкти 2 і 16 (Данія і Швеція): ці країни максимально схожі одна на одну і віддалені одна від одної на дуже малу відстань 1,288. Вони утворюють спільний кластер, який на 9-му етапі кластеризації з'являється з номером 2, а кластер 16 в таблиці більше не з'являється.

B. На наступному кроці відбувається об'єднання об'єктів 1 і 3 (Бельгія і Франція: віддалені на відстань 2,187) – утворений кластер з'являється з номером 1 на 4-му етапі кластеризації (кластер 3 в таблиці більше не з'являється).

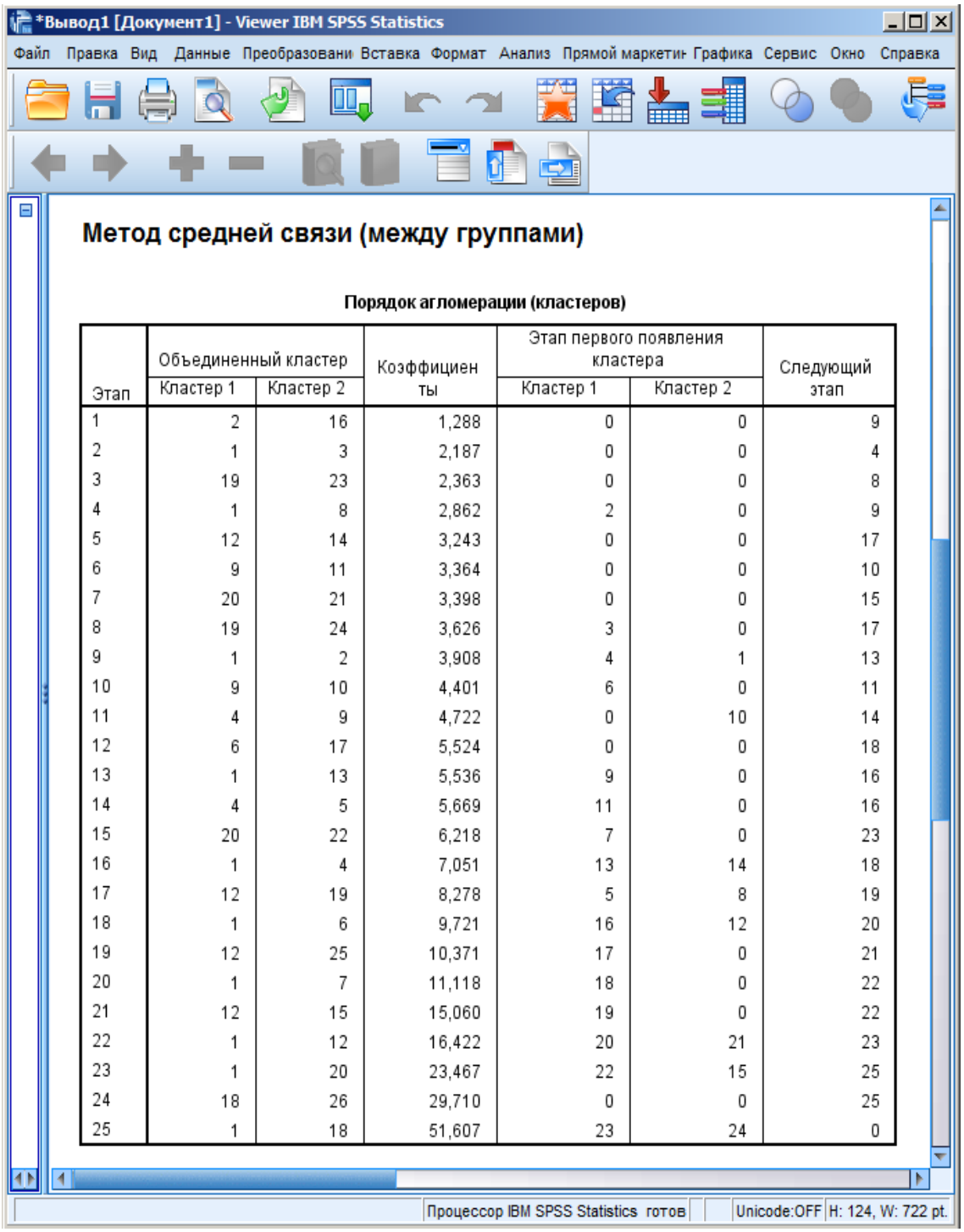
C. Потім об'єднуються об'єкти 19 і 23 (Болгарія і Польща: віддалені на відстань 2,363) – утворений кластер з'являється з номером 19 на 8-му етапі кластеризації (кластер 23 в таблиці більше не з'являється) і т.д.

Сводный отчет по наблюдениям[a]

Респонденты					
Допустимо		Пропущенные		Всего	
N	Проценты	N	Проценты	N	Проценты
26	100,0%	0	0,0%	26	100,0%

a. используемые квадрат евклидова расстояния

Рис. 5.15. Зведені підсумки за спостереженнями



Метод средней связи (между группами)

Порядок агломерации (кластеров)

Этап	Объединенный кластер		Коэффициенты	Этап первого появления кластера		Следующий этап
	Кластер 1	Кластер 2		Кластер 1	Кластер 2	
1	2	16	1,288	0	0	9
2	1	3	2,187	0	0	4
3	19	23	2,363	0	0	8
4	1	8	2,862	2	0	9
5	12	14	3,243	0	0	17
6	9	11	3,364	0	0	10
7	20	21	3,398	0	0	15
8	19	24	3,626	3	0	17
9	1	2	3,908	4	1	13
10	9	10	4,401	6	0	11
11	4	9	4,722	0	10	14
12	6	17	5,524	0	0	18
13	1	13	5,536	9	0	16
14	4	5	5,669	11	0	16
15	20	22	6,218	7	0	23
16	1	4	7,051	13	14	18
17	12	19	8,278	5	8	19
18	1	6	9,721	16	12	20
19	12	25	10,371	17	0	21
20	1	7	11,118	18	0	22
21	12	15	15,060	19	0	22
22	1	12	16,422	20	21	23
23	1	20	23,467	22	15	25
24	18	26	29,710	0	0	25
25	1	18	51,607	23	24	0

Рис. 5.16. Таблиця Agglomeration Schedule/Порядок агломерації

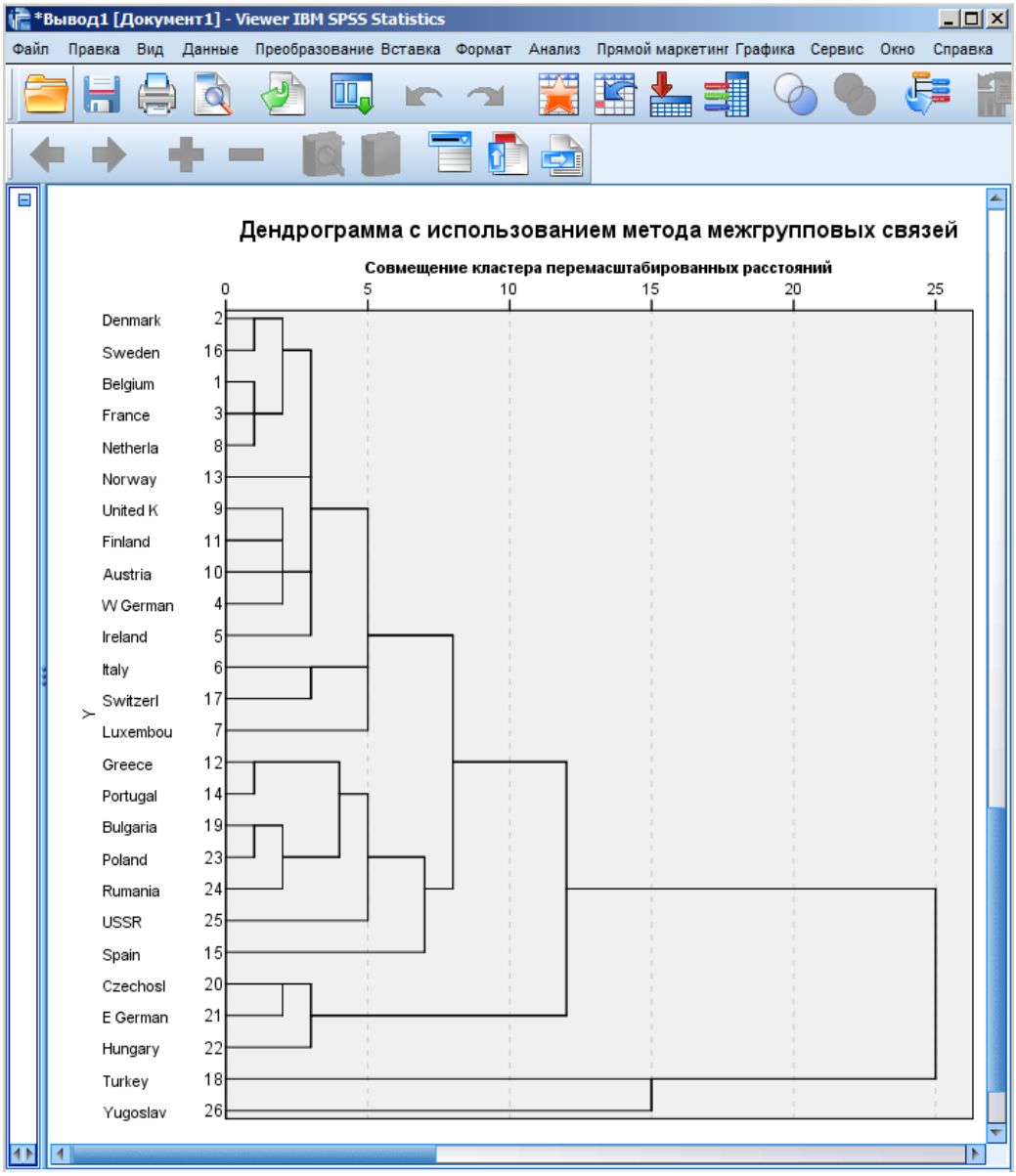


Рис. 5.17. Дендрограма

2. Для визначення *оптимальної кількості кластерів* вирішальне значення має показник *коефіцієнт*. Під цим коефіцієнтом мається на увазі відстань між двома кластерами, визначена на підставі обраної міри відстані. У нашому випадку це квадрат відстані Евкліда, визначений з використанням стандартизованих значень. На етапі, де ця міра відстані між двома кластерами збільшується стрибкоподібно, процес об'єднання в нові кластери необхідно зупинити, тому що у протилежному випадку були б об'єднані вже кластери, що знаходяться на відносно великій відстані один від одного.

Значний стрибок коефіцієнта спостерігається на 21-му кроці. У наведеному прикладі — це стрибок з 11,118 до 15,060. Саме на цьому етапі ми більше не повинні робити ніяких наступних об'єднань, а результат із п'ятьма кластерами є оптимальним. *Оптимальним* вважається число кластерів, рівне різниці кількості спостережень (тут: 26) і кількості кроків, після якого коефіцієнт збільшується стрибкоподібно (тут: 21). Тобто оптимальне число кластерів дорівнює 26 – 21 = 5.

3. Дендрограма дозволяє не тільки прослідкувати процес кластеризації наочно на будь-якому етапі, але і судити про те, яка відстань між кластерами на кожному з етапів (рис. 5.17).

Числа від 0 до 25 є умовною шкалою. Число 0 відповідає найменшій відстані, а 25 – найбільшій відстані на останньому етапі. Будь-який вертикальний перетин, який можна провести на дендрограмі, покаже скільки кластерів отримано на відповідному етапі (кількість перетинів із горизонтальними лініями), а відслідкувавши будь-яку горизонтальну лінію дендрограми вліво, можна дізнатися які країни увійшли до якого з кластерів.

5.3.3. Налаштування виведення результатів проведеного кластерного аналізу

Завдання 3. Здійснити налаштування видачі результатів здійсненого аналізу.

1. Після визначення оптимальної кількості кластерів (5) організуємо для кожного спостереження видачу інформації про належність до кластера.

2. Для налаштування видачі інформації необхідно перейти до вікна *Редактора даних/Data Editor* й обрати пункт меню *Аналіз/Analyze – Класифікація/Classify – Ієрархічна кластеризація/Hierarchical Cluster*.

3. Натисніть кнопку *Статистики/Statistics* та у вікні *Ієрархічний кластерний аналіз: Статистики/Hierarchical Cluster Analysis: Statistics* активуйте опції *Порядок агломерації/Agglomeration schedule* для виведення показника належності до кластера для кожного спостереження та *Матриця близькості* для виведення матриці близькості об'єктів (рис. 5.18).

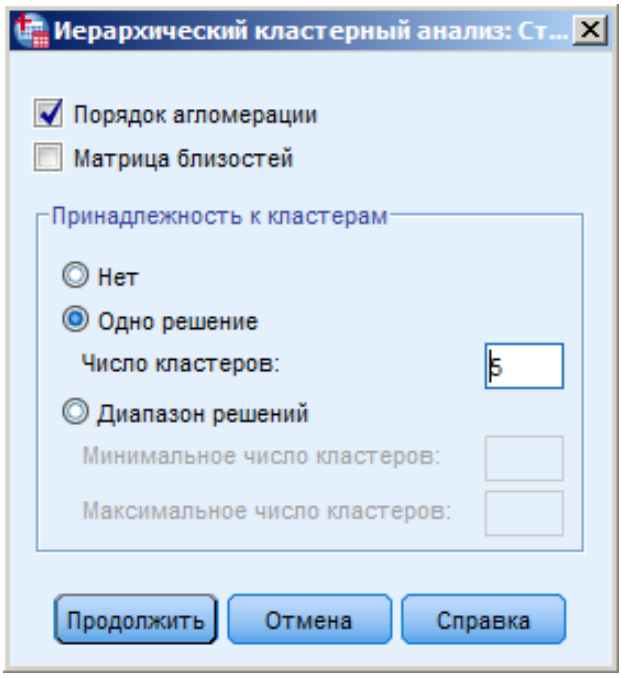


Рис. 5.18. Вікно Ієрархічний кластерний аналіз: Статистики/Hierarchical Cluster Analysis: Statistics

4. В області *Належність до кластерів/Cluster Membership* активуйте опцію *Одне рішення/Single solution*, вкажіть бажану кількість кластерів 5 та натисніть кнопку *Продовжити/Continue*.

5. Інформацію про належність кожного спостереження до визначеного кластера можна зберегти в новій змінній. Для цього натисніть кнопку *Зберегти/Save* й у вікні, що відкриється, оберіть опцію *Одне рішення/Single solution*, вкажіть кількість кластерів 5, та натисніть кнопку *Продовжити/Continue* (рис. 5.19).

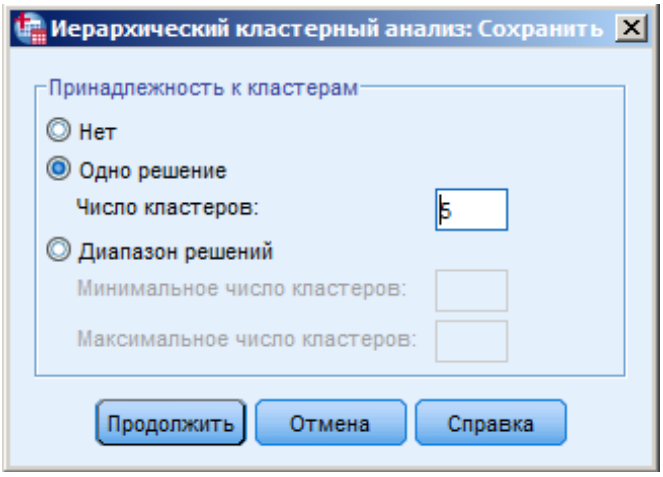


Рис. 5.19. Вікно Ієрархічний кластерний аналіз: Зберегти/Hierarchical Cluster Analysis: Save New Variables

6. У вікні *Ієрархічний кластерний аналіз/Hierarchical Cluster Analysis* натисніть кнопку *ОК*. Тепер крім таблиці порядку агломерації для кожного спостереження буде виводитися й інформація про належність до кластера.

7. У вікні виводу після звичайного статистичного зведення підсумків за спостереженнями, результатів кластеризації та дендрограми буде представлено належність до кластерів (рис. 5.20) та матрицю близькості об'єктів. Зверніть увагу, що під матрицею близькості вказано вид матриці близькості: *«це матриця несхожості»*, відповідно до обраної міри близькості – відстані Евкліда, яка є мірою несхожості.

Принадлежность к кластерам

Наблюдение	Кластеры 5
1:Belgium	1
2:Denmark	1
3:France	1
4:W German	1
5:Ireland	1
6:Italy	1
7:Luxembou	1
8:Netherla	1
9:United K	1
10:Austria	1
11:Finland	1
12:Greece	2
13:Norway	1
14:Portugal	2
15:Spain	2
16:Sweden	1
17:Switzerl	1
18:Turkey	3
19:Bulgaria	2
20:Czechosl	4
21:E German	4
22:Hungary	4
23:Poland	2
24:Rumania	2
25:USSR	2
26:Yugoslav	5

Рис. 5.20. Таблиця з виведенням належності до кластерів

8. З таблиці належності до кластерів можна побачити вміст п'яти кластерів:

a) до першого кластеру увійшли розвинуті капіталістичні країни;

b) до другого – капіталістичні та соціалістичні країни з аграрно орієнтованою економікою (Греція, Португалія, Іспанія, Болгарія, Польща, Румунія, Радянський Союз);

c) до четвертої групи – соціалістичні країни із розвинутою економікою (НДР, Чехословаччина, Угорщина);

d) окремі кластери створили дві країни – Туреччина (кластер №3) та Югославія (кластер № 5), що лише підтвердило їхній особливий шлях у світової економіці на час здійснення аналізу. Югославія впроваджувала специфічний шлях розвитку за рахунок тісних зв'язків з капіталістичними та соціалістичними країнами, а Туреччина мала стиль розвитку, який можна назвати азіатським.

9. У вікні *Редактора даних/Data Editor* відкрийте вкладки *Представлення Змінні/Variable View* та *Представлення Дані/Data View* та зверніть увагу, що додалася змінна *clu5_l*, яка вказує на кластерну належність кожного

спостереження. Змінна *clu5_l* буде використана для визначення *кластерних профілів*. *Кластерні профілі* допомо-жуть розібратися в значенні кластерів. Вони є середніми значеннями змінних, котрі включені в аналіз, розподі-лених за кластерною належністю.

10. У вікні *Редактора даних/Data Editor* оберіть у меню *Аналіз/Analyze – Порівняння середніх/Compare Means – Середні/Means* та у вікні, що відкриється *Середні/Means* змінним *agr-tc* надайте статус залежних змінних (*Dependent List*), а змінній *clu5_1* статус незалежної змінної (*Independent List*) (рис. 5.21).

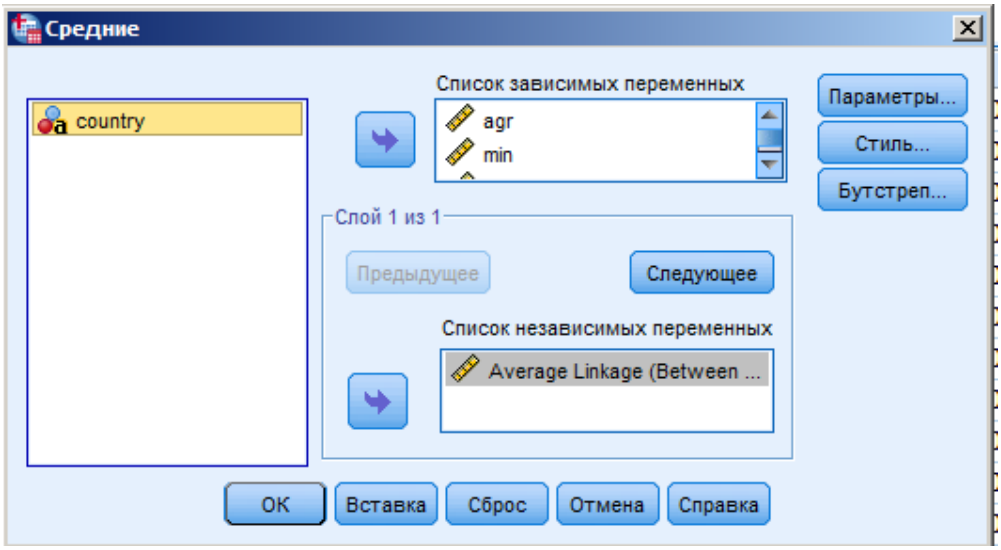


Рис. 5.21. Вікно Середні/Means

11. Натисніть кнопку *Параметри/Options* та задайте у вікні *Середні: Параметри/Means: Options* у області *Властивості в комірках/Cell Statistics* тільки видачу середніх значень (*Mean*) (рис. 5.22).

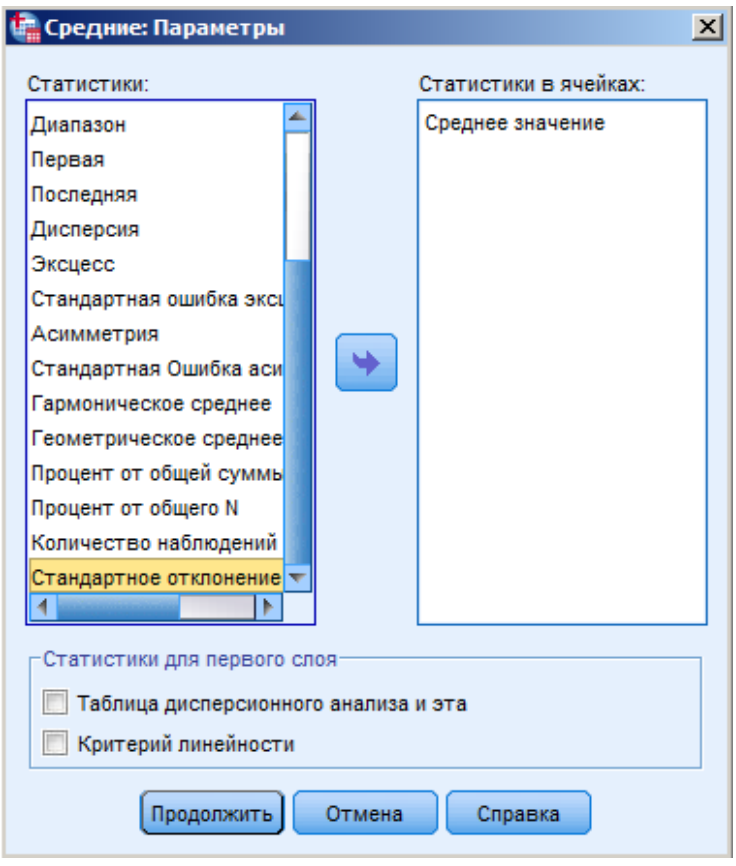


Рис. 5.22. Вікно Середні: Параметри/Means: Options

12. Натисніть кнопку *Продовжити/Continue* та кнопку *ОК* у вікні *Середні/Means* і почніть розрахунок.

13. У результаті проведеного розрахунку у вікні виводу виводяться середні значення підсумків дев'яти змінних *agr-tc* для п'яти кластерів (рис. 5.23).

Отчет

Среднее значение

Average Linkage (Between Groups)	agr	min	man	ps	con	si	fin	sps	tc
1	9,5929	,8500	27,6214	,9571	8,4214	16,6786	5,1143	24,0786	6,7000
2	29,3143	1,3714	26,3571	,6571	8,8857	8,8571	2,4286	15,5857	6,5429
3	66,8000	,7000	7,9000	,1000	2,8000	5,2000	1,1000	11,9000	3,2000
4	14,1333	2,9667	35,4333	1,4667	8,1667	9,9333	1,0000	19,0667	7,8000
5	48,7000	1,5000	16,8000	1,1000	4,9000	6,4000	11,3000	5,3000	4,0000
Всего	19,1308	1,2538	27,0077	,9077	8,1654	12,9577	4,0000	20,0231	6,5462

Рис. 5.23. Звіт за змінними та кластерами

14. Проаналізувавши інформацію звіту, можна зробити *такі висновки*:

a) країни капіталістичного табору з розвинутою економікою (кластер № 1) мали незначну зайнятість у аграрному секторі (agr), добувній галузі (min) і дуже велику зайнятість у фінансовій (fin) та соціальній сфері (sps);

b) капіталістичні та соціалістичні країни, що увійшли до кластера № 2 були країнами з аграрно орієнтованою економікою та розвинутою обробною промисловістю. Країни першої і другої групи мають однакові показники щодо зайнятості у будівництві та транспорті;

c) Туреччина (кластер № 3) мала надзвичайно велику зайнятість у аграрному секторі і, водночас, надзвичайно низьку у енергетичній галузі, тому й була виділена у окремий кластер;

d) Югославія (кластер № 5) так само мала великі особливості у зайнятості населення.

15. Збережіть результат здійсненого кластерного аналізу у вигляді файлу з іменем *Vivod.spv*.

5.4. ЗАВДАННЯ ДЛЯ САМОСТІЙНОЇ РОБОТИ

Завдання 4. Необхідно сформувати власний набір даних, які характеризуються двома числовими ознаками та виконати відповідно до отриманого варіанта (табл. 5.2) ієрархічний агломеративний кластерний аналіз із використанням:

1) засобів MS Excel зі збереженням вхідних даних та результату у файлі з іменем *Lab5_Номер_варіанту.xlsx*;

2) пакету SPSS зі збереженням вхідних даних та результату у файлах із іменами *Lab5_Номер_варіанту.sav* та *Lab5_Номер_варіанту. spv*.

КОНТРОЛЬНІ ПИТАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ № 5

1. У чому полягає сутність ієрархічного кластерного аналізу?
2. Назвіть основні типи алгоритмів ієрархічної кластеризації.
3. У чому полягає сутність ієрархічного агломеративного алгоритму?
4. У чому полягає сутність ієрархічного дивізимного алгоритму?
5. Основні етапи ієрархічного кластерного аналізу.
6. Основні методи зв'язку кластерів при здійсненні ієрархічного кластерного аналізу.
7. Послідовність здійснення ієрархічного кластерного аналізу у програмі SPSS.
8. Редактор даних, вікно виведення результатів програми SPSS.
9. У яких файлах зберігаються дані програми SPSS?
10. У яких файлах зберігаються результати аналізу програми SPSS?
11. Який критерій використовується для визначення оптимальної кількості кластерів?
12. Для чого призначена дендрограма та що вона відображає?

Таблиця 5.2

Міри близькості та методи зв'язку кластерів за варіантами для виконання завдання 4

Варіант	Кластеризація в MS Excel		Кластеризація в SPSS	
	Відстань	Метод зв'язку кластерів	Міра близькості	Метод зв'язку кластерів
1	Відстань Евкліда	Ближнього сусіда	Відстань Евкліда	Міжгрупового зв'язку
2	Квадрат відстані Евкліда	Дальнього сусіда	Квадрат відстані Евкліда	Внутрішньо групового зв'язку
3	Манхеттенська відстань	Ближнього сусіда	Квадрат відстані Евкліда	Ближнього сусіда
4	Відстань Чебишева	Дальнього сусіда	Блок (Манхеттенська відстань)	Дальнього сусіда
5	Пікова відстань	Ближнього сусіда	Відстань Чебишева	Центроїдний метод
6	Відстань Евкліда	Дальнього сусіда	Блок (Манхеттенська відстань)	Метод Уорда
7	Квадрат відстані Евкліда	Ближнього сусіда	Відстань Мінковського	Медіанний метод
8	Манхеттенська відстань	Дальнього сусіда	Квадрат відстані Евкліда	Міжгрупового зв'язку
9	Відстань Чебишева	Ближнього сусіда	Відстань Евкліда	Внутрішньо групового зв'язку
10	Пікова відстань	Дальнього сусіда	Квадрат відстані Евкліда	Ближнього сусіда
11	Квадрат відстані Евкліда	Ближнього сусіда	Відстань Евкліда	Дальнього сусіда
12	Відстань Евкліда	Дальнього сусіда	Відстань Мінковського	Центроїдний метод
13	Відстань Евкліда	Ближнього сусіда	Відстань Чебишева	Метод Уорда
14	Квадрат відстані Евкліда	Дальнього сусіда	Блок (Манхеттенська відстань)	Медіанний метод
15	Манхеттенська відстань	Ближнього сусіда	Відстань Мінковського	Міжгрупового зв'язку
16	Відстань Чебишева	Дальнього сусіда	Квадрат відстані Евкліда	Внутрішньо групового зв'язку
17	Пікова відстань	Ближнього сусіда	Відстань Евкліда	Ближнього сусіда
18	Відстань Евкліда	Дальнього сусіда	Квадрат відстані Евкліда	Дальнього сусіда
19	Квадрат відстані Евкліда	Ближнього сусіда	Відстань Мінковського	Центроїдний метод
20	Манхеттенська відстань	Дальнього сусіда	Квадрат відстані Евкліда	Метод Уорда
21	Відстань Чебишева	Ближнього сусіда	Відстань Чебишева	Медіанний метод
22	Пікова відстань	Дальнього сусіда	Блок (Манхеттенська відстань)	Ближнього сусіда
23	Квадрат відстані Евкліда	Ближнього сусіда	Відстань Мінковського	Дальнього сусіда
24	Відстань Евкліда	Дальнього сусіда	Відстань Евкліда	Міжгрупового зв'язку
25	Квадрат відстані Евкліда	Ближнього сусіда	Відстань Мінковського	Внутрішньо групового зв'язку

6. АЛГОРИТМИ K-MEANS, C-MEANS. КЛАСТЕРНИЙ АНАЛІЗ ДАНИХ У СЕРЕДОВИЩІ MATLAB

Лабораторна робота № 6

Мета: закріплення знань про сутність кластерного аналізу, етапи його проведення та типи алгоритмів кластеризації. Набуття навичок проведення кластерного аналізу в середовищі MatLab із використанням ієрархічного алгоритму та алгоритмів квадратичної похибки – чіткого алгоритму k-means та нечіткого алгоритму c-means.

Теоретичні знання: ієрархічні та неієрархічні алгоритми кластерного аналізу. Алгоритми квадратичної похибки. Етапи чіткого алгоритму кластеризації k-means. Етапи нечіткого алгоритму кластеризації c-means. Модифікації алгоритмів k-means, c-means. Проведення кластерного аналізу засобами середовища MatLab, аналіз отриманих результатів.

6.1. АЛГОРИТМИ K-MEANS, C-MEANS: БАЗОВІ ПОНЯТТЯ

6.1.1. Типи алгоритмів кластерного аналізу

Існує велика кількість алгоритмів кластерного аналізу, серед яких можна виділити наступні основні типи.

1. За способом обробки даних:
   a) ***ієрархічні алгоритми***: будують систему вкладених розбиттів набору даних й отримують на виході дерево кластерів, коренем якого є весь набір даних, а листами – окремі об'єкти набору даних;
   b) ***неієрархічні*** (плоскі) ***алгоритми***: будують одне розбиття об'єктів набору даних на кластери, що не перетинаються.
2. За способом аналізу даних:
   a) ***чіткі алгоритми***: кожному об'єкту набору даних ставлять у відповідність номер кластера таким чином, що кожен об'єкт належить тільки одному кластеру;
   b) ***нечіткі алгоритми***: кожному об'єкту ставлять у відповідність набір значень, які показують ступінь його відношення до кластерів таким чином, що кожен об'єкт належить до кожного кластера з деякою ймовірністю.

6.1.2. Алгоритми квадратичної похибки

Алгоритми квадратичної похибки відносяться до типу плоских алгоритмів, які спочатку були розроблені для числових метричних даних. Задачу кластеризації вони розглядають як побудову оптимального розбиття набору даних на групи споріднених об'єктів. Найчастіше для таких даних при визначенні близькості об'єктів обирають відстань Евкліда. Тоді дія алгоритму спрямована на мінімізацію цільової функції у вигляді суми квадратів відхилень відстаней від об'єктів до центрів ваги кластерів, яким вони належать.

Центр ваги або ***центроїд*** кластера – це точка у багатовимірному просторі ознак із координатами, які є середніми арифметичними значеннями відповідних ознак об'єктів, що входять до кластера.

Оптимальність визначається шляхом мінімізації ***цільової функції J*** – внутрішньокластерної квадратичної похибки розбиття (англ. *SSE – Sum of Squared Errors*):

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n_j} d^2(x_{ij}, c_j), \tag{6.1}$$

де c_j – j -й кластер, k – кількість кластерів,

x_{ij} – i -й об'єкт j -го кластера, n_j – кількість об'єктів у j -му кластері,

$d(x_{ij}, c_j)$ – відстань між i -м об'єктом j -го кластера та його центром ваги.

Найбільш розповсюдженим алгоритмом квадратичної похибки є алгоритм *k-means*.

6.1.3. Етапи алгоритму k-means

Алгоритм ***k-means*** є чітким алгоритмом кластеризації (*Hard k-Means*), процедура якого є ітераційною.

Основні етапи алгоритму k-means є наступними.

1. Обрати характеристики об'єктів набору даних, суттєві для предметної області, в якій здійснюється аналіз, та визначити змінні, які їм відповідають.

2. Здійснити нормалізацію чи стандартизацію значень змінних, обраних для кластеризації, якщо вони належать до різних числових діапазонів.

3. Сформувати матрицю даних X, у якій рядкам відповідають об'єкти, а стовпцям – змінні, що характеризують ці об'єкти: $X = (x_{ij})$, де

x_{ij}– значення j -ї змінної для i -го об'єкта, $i \in \{1,\dots,n\}, j \in \{1,\dots,m\}$,

n – кількість об'єктів, m – кількість змінних.

4. Задати значення k – кількості кластерів, на які необхідно розбити набір даних.

5. Випадково обрати k початкових центрів ваги кластерів – *центроїдів*.

6. Розрахувати відстань від усіх об'єктів до центроїдів.

7. Віднести кожен об'єкт до кластера з найближчим центром ваги.

8. Перерахувати центри ваги кластерів відповідно до їх поточного складу.

9. Повернутися до пункту 6 та повторяти пункти 6, 7 і 8 до досягнення мінімального значення середньоквадратичної похибки – критерію зупинки алгоритму. Роботу алгоритму можна припиняти на тому етапі ітерації, на якому не відбулося переміщення об'єктів із кластера в кластер.

Перевагами алгоритму k-means є його простота та швидкість виконання. До недоліків алгоритму відносять необхідність задання наперед кількості кластерів k. До того ж алгоритм є чутливим до вибору початкових центрів ваги кластерів, наявності викидів та не вирішує задачу у разі, коли близькість об'єкту є однаковою до центрів ваги декількох кластерів.

6.1.4. Етапи алгоритму c-means

Нечіткий алгоритм кластеризації ***c-means*** (*Fuzzy c-Means*) є вдосконаленням алгоритму *k-means*. Відмінність полягає у тому, що алгоритм *c-means* визначає ймовірність віднесення об'єкту до кожного з кластерів, формуючи матрицю нечіткого розбиття.

Матриця нечіткого розбиття n об'єктів на k кластерів – це матриця U розміром $k \times n$, рядки якої відповідають кластерам, а стовпці – об'єктам. Елементи матриці u_{ij} містять ***коефіцієнти належності*** – ймовірність, із якою j - й об'єкт відноситься до i -го кластера. Сума коефіцієнтів належності по стовпцям повинна дорівнювати 1.

Цільова функція J, яка визначає оптимальність нечіткого розбиття набору даних на групи споріднених об'єктів, буде мати вигляд:

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n_j} u_{ji}^{w} d^2(x_{ij}, c_j), \tag{6.2}$$

де u_{ji} – елемент матриці нечіткого розбиття U, w – коефіцієнт нечіткості.

Усі інші компоненти цільової функції, заданої формулою 6.2 аналогічні компонентам цільової функції, заданої формулою 6.1.

Основні етапи алгоритму c-means є наступними:

1-3. Перші три етапи співпадають із етапами алгоритму *k-means* (див. попередній п. 1.3).

4. Здійснити початкову ініціалізацію – задати параметри:

k – кількість кластерів, на які необхідно розбити набір даних;

w – *коефіцієнт нечіткості*: найчастіше $w = 2$ (або 1,5), при наближенні значеньw до 1 алгоритм вироджується у чіткий алгоритм *k-means*;

δ – *параметр зупинки*: якщо δ ближче до 0, то розбиття буде більш розмитим, а якщо δ буде близьким до 1 – більш чітким (результат розбиття буде схожим на роботу алгоритму k-means).

5. Ініціалізувати випадковим чином початкову матрицю U нечіткого розбиття об'єктів набору даних на k кластерів таким чином, щоб сума елементів матриці по стовпцям дорівнювала 1.

6. Обчислити центри ваги кластерів: для розрахунку i -ї координати центру j -го кластера необхідно розділити суму попарних добутків елементів i -го стовпця матриці вхідних даних на елементи j -го рядка матриці нечіткого розбиття у степенях w на суму елементів j -го рядка матриці нечіткого розбиття у степенях w:

$$c_{ji} = \frac{\sum_{l=1}^{n} x_{li} u_{jl}^{w}}{\sum_{l=1}^{n} u_{jl}^{w}}. \tag{6.3}$$

7. Розрахувати відстань від об'єктів до центрів кластерів та перерахувати елементи матриці нечіткого розбиття U за формулою:

$$u_{jk} = \frac{1}{\left(\sum_{i=1}^{m} \frac{d^2(x_k, c_j)}{d^2(x_i, c_j)} \right)^{\frac{1}{w-1}}}, \quad (6.4)$$

де $d(x_i, c_j)$ – відстань від i -го об'єкта до j -го кластера, m – кількість змінних.

8. Перевірити виконання умови зупинки алгоритму: порівняти різницю значень *цільової функції J*, розрахованої за формулою 5.2 для поточного та попереднього етапів ітерації, із параметром зупинки δ. Якщо ця різниця менша за параметр зупинки – це є критерієм зупинки алгоритму кластеризації. Якщо ні – необхідно повернутися до пункту 6 та знову повторити пункти 6, 7 і 8.

Роботу алгоритму можна припиняти на тому етапі, на якому зміна матриці нечіткого розбиття поточного та попереднього етапів ітерації $\left\| U^{(l)} - U^{(l-1)} \right\|$ є меншою за параметр зупинки (тут l – номер поточної ітерації а $\left\| U^{(l)} - U^{(l-1)} \right\|$ – матрична норма).

9. У результаті проведеного кластерного аналізу кожен об'єкт відносять до того кластеру, для якого він у матриці нечіткого розбиття на останньому етапі ітерації має найбільший коефіцієнт належності.

6.1.5. Модифікації алгоритмів k-means та c-means

При розв'язанні задачі кластерного аналізу оцінку близькості об'єктів можна здійснювати шляхом визначення як мір несхожості (відстаней для метричних даних), так і мір подібності. Як уже було зазначено раніше, спосіб визначення мір близькості між об'єктами за певним методом залежить від типу атрибутів набору даних та шкал, у яких їх вимірюють.

Тому існує багато варіантів вибору мір близькості та цільової функції, які можуть бути використані в алгоритмах *k-means* і *c-means*. Що обумовлює наявність модифікацій цих алгоритмів, реалізованих шляхом використання:

1) різних методів визначення мір близькості між об'єктами;
2) різних підходів до розрахунку цільової функції, яку необхідно оптимізувати;
3) різних способів визначення центру ваги кластерів.

Так, при виборі Манхеттенської відстані для визначення близькості об'єктів до цетроїдів дія алгоритму може бути спрямована на мінімізацію цільової функції у вигляді суми відстаней від об'єктів до центрів ваги кластерів, яким вони належать. А центр ваги кожного кластера може бути розрахований із використанням ***медіанного методу*** як точка у багатовимірному просторі ознак із координатами, які є медіанами відповідних ознак об'єктів, що входять до кластера.

Медіана у статистиці є такою величиною, яка розміщена посередині ранжованого ряду значень ознаки набору даних. Наприклад, якщо кластер містить об'єкти зі значеннями ознак (0, 1), (1, 0) та (5, 5), то координати центроїда, визначені за медіанним методом будуть рівні (1, 1). А якщо їх розраховувати як середньоарифметичне значення ознак об'єктів, координати центроїда будуть рівними (2, 2).

У випадку, якщо для оцінки близькості об'єктів до центроїдів розраховують міри подібності (наприклад, косинус подібності), як цільову функцію доцільно обрати суму мір подібності кожного об'єкта кластера до його центроїда. Але у цьому випадку оптимізація цільової функції потребує визначення її максимуму.

6.1.6. Приклад кластерного аналізу за алгоритмом k-means

Наведемо приклад здійснення кластерного аналізу у одновимірному просторі ознак для більшої наочності демонстрації етапів алгоритму за рахунок зменшення громіздкості розрахунків. У випадку метричних даних відстань між i -м об'єктом та центроїдом j -го кластера буде визначатись за формулою:

$$d_{ij} = \left| x_i - c_j \right|,$$

де x_i – значення ознаки i -го об'єкта, c_j – координата центра ваги j -го кластера.

У випадку набору даних із більшою кількістю ознак близькість об'єктів до центроїдів визначають за розглянутими раніше мірами близькості (відстанями для метричних даних).

Приклад 1. Згрупувати відвідувачів вебсайту за їх віком (табл. 6.1).

Таблиця 6.1

Значення віку відвідувачів сайту

Ознака	Відвідувачі сайту (об'єкти)																		
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Вік	15	15	16	19	19	20	20	21	22	28	35	40	41	42	43	44	60	61	65

1. Кластеризацію об'єктів – їх групування на 2 кластери (k = 2) будемо здійснювати у одновимірному просторі ознаки – «вік» із використанням алгоритму k-means.

2. Випадковим чином визначаємо початкові координати центроїдів – центрів ваги кластерів:

$c_1 = 16$ – центроїд першого кластера, $c_2 = 22$ – центроїд другого кластера.

3. Знаходимо відстані від об'єктів до центрів ваги кластерів:

для 1-го об'єкта: $d_{11} = |15 - 16| = 1$, $d_{12} = |15 - 22| = 7$;
для 2-го об'єкта: $d_{21} = |15 - 16| = 1$, $d_{22} = |15 - 22| = 7$;
для 3-го об'єкта: $d_{31} = |16 - 16| = 0$, $d_{32} = |16 - 22| = 6$.

Для наступних об'єктів обчислення робимо аналогічно, отримуємо розраховані відстані від усіх об'єктів до першого та другого центроїда на першому етапі ітерації та визначаємо належність кожного об'єкта до того кластера, до якого він знаходиться ближче (табл. 6.2).

Таблиця 6.2

Визначення відстаней до центроїдів та перерахунок центрів ваги кластерів на першому етапі ітерації

Вік відвідувачів сайту	d_{i1} – відстань до центру ваги 1-го кластера	d_{i2} – відстань до центру ваги 2-го кластера	Визначена належність до кластера	Координати центроїдів
15	1	7	1	$c_1 = 15{,}33$
15	1	7	1	
16	0	6	1	
19	9	3	2	$c_2 = 36{,}25$
19	9	3	2	
20	16	2	2	
20	16	2	2	
21	25	1	2	
22	36	0	2	
28	12	6	2	
35	19	13	2	
40	24	18	2	
41	25	19	2	
42	26	20	2	
43	27	21	2	
44	28	22	2	
60	44	38	2	
61	45	39	2	

4. Перераховуємо центри ваги кластерів із урахуванням їх складу.

Центроїд першого кластера:

$$c_1 = \frac{15+15+16}{3} = 15{,}33.$$

Центроїд другого кластера:

$$c_2 = \frac{19+19+20+20+21+22+28+35+40+41+42+43+44+60+61+65}{16} = 36{,}25 \ .$$

5. Знаходимо відстані від об'єктів до нових центрів ваги кластерів та визначаємо належність кожного об'єкта до того кластера, до якого він знаходиться ближче на другому етапі ітерації (табл. 6.3).

6. Перераховуємо центри ваги кластерів відповідно до їх оновленого складу. Маємо:

$c_1 = 18{,}56$ – центроїд першого кластера, $c_2 = 45{,}9$ – центроїд другого кластера.

7. Знаходимо відстані від об’єктів до нових центрів ваги кластерів та визначаємо належність кожного об’єкта до того кластера, до якого він знаходиться ближче на третьому етапі ітерації (табл. 6.4).

Таблиця 6.3

Визначення відстаней до центроїдів та перерахунок центрів ваги кластерів на другому етапі ітерації

Вік відвідувачів сайту (об’єкти)	d_{i1} – відстань до центру ваги 1-го кластера	d_{i2} – відстань до центру ваги 2-го кластера	Визначена належність до кластера	Координати центроїдів
15	0,33	21,25	1	$c_1 = 18{,}56$
15	0,33	21,25	1	
16	0,67	20,25	1	
19	3,67	17,25	1	
19	3,67	17,25	1	
20	4,67	16,25	1	
20	4,67	16,25	1	
21	5,67	15,25	1	
22	6,67	14,25	1	
28	12,67	8,25	2	$c_2 = 45{,}9$
35	19,67	1,25	2	
40	24,67	3,75	2	
41	25,67	4,75	2	
42	26,67	5,75	2	
43	27,67	6,75	2	
44	28,67	7,75	2	
60	44,67	23,75	2	
61	45,67	24,75	2	
65	49,67	28,75	2	

Таблиця 6.4

Визначення відстаней до центроїдів та перерахунок центрів ваги кластерів на третьому етапі ітерації

Вік відвідувачів сайту	d_{i1} – відстань до центру ваги 1-го кластера	d_{i2} – відстань до центру ваги 2-го кластера	Визначена належність до кластера	Координати центроїдів
15	3,56	30,9	1	$c_1 = 19{,}50$
15	3,56	30,9	1	
16	2,56	29,9	1	
19	0,44	26,9	1	
19	0,44	26,9	1	
20	1,44	25,9	1	
20	1,44	25,9	1	
21	2,44	24,9	1	
22	3,44	23,9	1	
28	9,44	17,9	1	
35	16,44	10,9	2	$c_2 = 47{,}89$
40	21,44	5,9	2	
41	22,44	4,9	2	
42	23,44	3,9	2	
43	24,44	2,9	2	
44	25,44	1,9	2	
60	41,44	14,1	2	
61	42,44	15,1	2	
65	46,44	19,1	2	

8. Перераховуємо центри ваги кластерів відповідно до їх оновленого складу. Маємо:

$c_1 = 19{,}50$ – центроїд першого кластера,
$c_2 = 47{,}89$ – центроїд другого кластера.

9. Знаходимо відстані від об'єктів до нових центрів ваги кластерів та визначаємо належність кожного об'єкта до того кластера, до якого він знаходиться ближче на четвертому етапі ітерації (табл. 6.5).

10. Ми бачимо, що на четвертому етапі ітерації склад кластерів залишився таким же, як і на третьому етапі. Це є критерієм зупинки алгоритму.

11. Проведений кластерний аналіз за алгоритмом *k-means* дозволив виявити дві групи споріднених об'єктів. До першого кластера було віднесено відвідувачів сайту віком від 15 до 28 років. До другого кластера віднесено відвідувачів сайту віком від 35 до 65 років.

Таблиця 6.5

Визначення відстаней до центроїдів та перерахунок центрів ваги кластерів на четвертому етапі ітерації

Вік відвідувачів сайту	d_{i1} – відстань до центру ваги 1-го кластера	d_{i2} – відстань до центру ваги 2-го кластера	Визначена належність до кластера	Координати центроїдів
15	4,50	32,89	1	
15	4,50	32,89	1	
16	3,50	31,89	1	
19	0,50	28,89	1	
19	0,50	28,89	1	
20	0,50	27,89	1	$c_1 = 19{,}50$
20	0,50	27,89	1	
21	1,50	26,89	1	
22	2,50	25,89	1	
28	8,50	19,89	1	
35	15,50	12,89	2	
40	20,50	7,89	2	
41	21,50	6,89	2	
42	22,50	5,89	2	
43	23,50	4,89	2	$c_2 = 47{,}89$
44	24,50	3,89	2	
60	40,50	12,11	2	
61	41,50	13,11	2	
65	45,50	17,11	2	

6.1.7. Приклад кластерного аналізу за алгоритмом c-means

Наведемо приклад здійснення кластерного аналізу у двовимірному просторі ознак. Для зменшення громіздкості розрахунків та більшої наочності демонстрації етапів алгоритму *c-means* розглянемо набір даних, який містить невелику кількість об'єктів.

Приклад 2. Здійснити кластерний аналіз набору даних із 6-ти об'єктів, кожен з яких характеризується двома ознаками за алгоритмом c-means (табл. 6.6).

Таблиця 6.6

Значення ознак об'єктів набору даних

Ознаки	Об'єкти					
	1-*й*	2-*й*	3-*й*	4-*й*	5-*й*	6-*й*
1	1	2	3	6	5	6
2	6	5	8	4	7	9

1. Кластеризацію об'єктів – їх групування, будемо здійснювати у двовимірному просторі ознак із використанням алгоритму *c-means*.

2. Ознаки об’єктів представлені у одному числовому діапазоні, тому необхідності здійснювати їх нормалізацію немає.

3. Сформуємо матрицю даних X, у якій рядкам відповідають об’єкти, а стовпцям – змінні, що характеризують ці об’єкти:

$$X = \begin{vmatrix} 1 & 6 \\ 2 & 5 \\ 3 & 8 \\ 5 & 4 \\ 5 & 7 \\ 6 & 9 \end{vmatrix}.$$

4. Здійснимо початкову ініціалізацію, задавши значення параметрів:

$k = 2$ – кількість кластерів, на які необхідно розбити набір даних;

$w = 2$ – коефіцієнт нечіткості;

$\delta = 0{,}001$ – параметр зупинки.

5. Ініціалізуємо випадковим чином початкову матрицю U нечіткого розбиття об’єктів набору даних на 2 кластери таким чином, щоб сума елементів матриці по стовпцям дорівнювала 1.

$$U = \begin{vmatrix} 0{,}8 & 0{,}9 & 0{,}7 & 0{,}4 & 0{,}5 & 0{,}2 \\ 0{,}2 & 0{,}1 & 0{,}3 & 0{,}6 & 0{,}5 & 0{,}8 \end{vmatrix}.$$

6. Обчислимо центри ваги кластерів за формулою 6.3. Враховуючи, що $w = 2$, розрахуємо координати центроїдів:

$$c_{11} = \frac{1 \cdot 0{,}8^2 + 2 \cdot 0{,}9^2 + 3 \cdot 0{,}7^2 + 6 \cdot 0{,}4^2 + 5 \cdot 0{,}5^2 + 6 \cdot 0{,}2^2}{0{,}8^2 + 0{,}9^2 + 0{,}7^2 + 0{,}4^2 + 0{,}5^2 + 0{,}2^2} = 2{,}586$$

$$c_{12} = \frac{6 \cdot 0{,}8^2 + 5 \cdot 0{,}9^2 + 8 \cdot 0{,}7^2 + 4 \cdot 0{,}4^2 + 7 \cdot 0{,}5^2 + 9 \cdot 0{,}2^2}{0{,}8^2 + 0{,}9^2 + 0{,}7^2 + 0{,}4^2 + 0{,}5^2 + 0{,}2^2} = 6{,}092$$

$$c_{21} = \frac{1 \cdot 0{,}2^2 + 2 \cdot 0{,}1^2 + 3 \cdot 0{,}3^2 + 6 \cdot 0{,}6^2 + 5 \cdot 0{,}5^2 + 6 \cdot 0{,}8^2}{0{,}2^2 + 0{,}1^2 + 0{,}3^2 + 0{,}6^2 + 0{,}5^2 + 0{,}8^2} = 5{,}453$$

$$c_{22} = \frac{6 \cdot 0{,}2^2 + 5 \cdot 0{,}1^2 + 8 \cdot 0{,}3^2 + 4 \cdot 0{,}6^2 + 7 \cdot 0{,}5^2 + 9 \cdot 0{,}8^2}{0{,}2^2 + 0{,}1^2 + 0{,}3^2 + 0{,}6^2 + 0{,}5^2 + 0{,}8^2} = 7{,}165$$

Маємо координати центрів ваги кластерів C_1(2,586; 6,092) та C_2(5,453; 7,165).

7. Розрахуємо відстань від об’єктів до центрів кластерів C_1 і C_2 за формулою Евкліда:

$$d(x_1, c_1) = \sqrt{(1-2{,}586)^2 + (6-6{,}092)^2} = 1{,}588$$

$$d(x_1, c_2) = \sqrt{(1-5{,}453)^2 + (6-7{,}165)^2} = 4{,}603$$

$$d(x_2, c_1) = \sqrt{(2-2{,}586)^2 + (5-6{,}092)^2} = 1{,}239$$

$$d(x_2, c_2) = \sqrt{(2-5{,}453)^2 + (5-7{,}165)^2} = 4{,}076$$

$$d(x_3, c_1) = \sqrt{(3-2{,}586)^2 + (8-6{,}092)^2} = 1{,}952$$

$$d(x_3, c_2) = \sqrt{(3-5{,}453)^2 + (8-7{,}165)^2} = 2{,}591$$

$$d(x_4, c_1) = \sqrt{(6-2{,}586)^2 + (4-6{,}092)^2} = 4{,}004$$

$$d(x_4, c_2) = \sqrt{(6-5{,}453)^2 + (4-7{,}165)^2} = 3{,}212$$

$$d(x_5, c_1) = \sqrt{(5-2{,}586)^2 + (7-6{,}092)^2} = 2{,}579$$

$$d(x_5, c_2) = \sqrt{(5-5{,}453)^2 + (7-7{,}165)^2} = 0{,}482$$

$$d(x_6, c_1) = \sqrt{(6-2{,}586)^2 + (9-6{,}092)^2} = 4{,}485$$

$$d(x_6, c_2) = \sqrt{(6-5{,}453)^2 + (9-7{,}165)^2} = 1{,}914$$

8. Розраховуємо значення цільової функції J на етапі $l = 1$ за формулою 6.2:

$$J_1 = 1{,}588 \cdot 0{,}8^2 + 1{,}239 \cdot 0{,}9^2 + 1{,}952 \cdot 0{,}7^2 + 4{,}004 \cdot 0{,}4^2 + 2{,}59 \cdot 0{,}5^2 + 4{,}485 \cdot 0{,}2^2 + 4{,}603 \cdot 0{,}2^2 +$$
$$+ 4{,}076 \cdot 0{,}1^2 + 2{,}591 \cdot 0{,}3^2 + 3{,}212 \cdot 0{,}6^2 + 0{,}482 \cdot 0{,}5^2 + 1{,}914 \cdot 0{,}8^2 = 7{,}40266.$$

Маємо $J_1 = 7{,}40266$.

9. Перераховуємо елементи матриці нечіткого розбиття U за формулою 6.4:

$$u_{11} = \frac{1}{1{,}588^2 \cdot \left(\frac{1}{1{,}588^2} + \frac{1}{4{,}603^2}\right)^{\frac{1}{(2-1)}}} = 0{,}894 \qquad u_{21} = \frac{1}{4{,}603^2 \cdot \left(\frac{1}{1{,}588^2} + \frac{1}{4{,}603^2}\right)^{\frac{1}{(2-1)}}} = 0{,}106$$

$$u_{12} = \frac{1}{1{,}239^2 \cdot \left(\frac{1}{1{,}239^2} + \frac{1}{4{,}076^2}\right)^{\frac{1}{(2-1)}}} = 0{,}915 \qquad u_{22} = \frac{1}{4{,}076^2 \cdot \left(\frac{1}{1{,}239^2} + \frac{1}{4{,}076^2}\right)^{\frac{1}{(2-1)}}} = 0{,}085$$

$$u_{13} = \frac{1}{1{,}952^2 \cdot \left(\frac{1}{1{,}952^2} + \frac{1}{2{,}591^2}\right)^{\frac{1}{(2-1)}}} = 0{,}638 \qquad u_{23} = \frac{1}{2{,}591^2 \cdot \left(\frac{1}{1{,}952^2} + \frac{1}{2{,}591^2}\right)^{\frac{1}{(2-1)}}} = 0{,}362$$

$$u_{14} = \frac{1}{4{,}004^2 \cdot \left(\frac{1}{4{,}004^2} + \frac{1}{3{,}212^2}\right)^{\frac{1}{(2-1)}}} = 0{,}392 \qquad u_{24} = \frac{1}{3{,}212^2 \cdot \left(\frac{1}{4{,}004^2} + \frac{1}{3{,}212^2}\right)^{\frac{1}{(2-1)}}} = 0{,}608$$

$$u_{15} = \frac{1}{2{,}579^2 \cdot \left(\frac{1}{2{,}579^2} + \frac{1}{0{,}482^2}\right)^{\frac{1}{(2-1)}}} = 0{,}034 \qquad u_{25} = \frac{1}{0{,}482^2 \cdot \left(\frac{1}{2{,}579^2} + \frac{1}{0{,}482^2}\right)^{\frac{1}{(2-1)}}} = 0{,}966$$

$$u_{16} = \frac{1}{4{,}485^2 \cdot \left(\frac{1}{4{,}485^2} + \frac{1}{1{,}914^2}\right)^{\frac{1}{(2-1)}}} = 0{,}154 \qquad u_{26} = \frac{1}{1{,}914^2 \cdot \left(\frac{1}{4{,}485^2} + \frac{1}{1{,}914^2}\right)^{\frac{1}{(2-1)}}} = 0{,}846$$

Маємо оновлену матрицю нечіткого розбиття:

$$U = \begin{vmatrix} 0{,}894 & 0{,}915 & 0{,}638 & 0{,}392 & 0{,}034 & 0{,}154 \\ 0{,}106 & 0{,}085 & 0{,}362 & 0{,}608 & 0{,}966 & 0{,}846 \end{vmatrix}.$$

10. Обчислимо з урахуванням перерахованих значень матриці нечіткого розбиття нові центри ваги кластерів за формулою 6.3. Отримаємо координати центроїдів на етапі ітерації на етапі $l = 2$:

$C_1(2{,}144;\ 5{,}349)$ $C_2(5{,}884;\ 7{,}196)$.

11. За формулою Евкліда розрахуємо відстань від об'єктів до центрів кластерів C_1 і C_2 на другому етапі ітерації. Отримаємо:

$$d(x_1, c_1) = 1{,}316 \qquad d(x_1, c_2) = 5{,}028$$
$$d(x_2, c_1) = 0{,}377 \qquad d(x_2, c_2) = 4{,}462$$
$$d(x_3, c_1) = 2{,}786 \qquad d(x_3, c_2) = 2{,}994$$
$$d(x_4, c_1) = 4{,}085 \qquad d(x_4, c_2) = 3{,}199$$
$$d(x_5, c_1) = 3{,}299 \qquad d(x_5, c_2) = 0{,}905$$
$$d(x_6, c_1) = 5{,}310 \qquad d(x_6, c_2) = 1{,}807$$

12. За формулою 6.2 розраховуємо значення цільової функції J на етапі $l = 2$. Маємо $J_2 = 7{,}25502$.

13. Перевіряємо виконання умови зупинки алгоритму: знаходимо різницю значень цільової функції $|J_1 - J_2| = 7{,}40266 - 7{,}25502 = 0{,}14724$. Це значення більше за параметр зупинки $\delta = 0{,}001$, тому переходимо до наступного етапу ітерації $l = 3$: знову обчислюємо координати центроїдів C_1 і C_2, розраховуємо відстань від об'єктів до центрів кластерів, перераховуємо елементи матриці нечіткого розбиття та перевіряємо умову зупинки алгоритму.

14. У таблиці 6.7 наведено значення цільової функції, координати центроїдів та результати перевірки умови зупинки алгоритму, отримані на наступних етапах ітерації. Обчислення виконані аналогічно описаним вище, детально наводити їх не будемо. Умова зупинки алгоритму була виконана на п'ятому етапі ітерації. Тому роботу алгоритму було припинено.

Таблиця 6.7

Значення ознак об'єктів набору даних

Етап ітерації	Координати центроїдів	Значення цільової функції	Різниця значень цільової функції ($\delta = 0{,}001$)	Виконання умови зупинки алгоритму
$l = 3$	C_1(2,144; 5,349) C_2(5,884; 7,196)	$J_2 = 7{,}11803$	$J_3 - J_2 > \delta$	не виконується
$l = 4$	C_1(2,144; 5,349) C_2(5,884; 7,196)	$J_4 = 6{,}88693$	$J_4 - J_3 > \delta$	не виконується
$l = 5$	C_1(2,144; 5,349) C_2(5,884; 7,196)	$J_5 = 6{,}88693$	$J_5 - J_4 < \delta$	виконується

15. На останньому п'ятому етапі ітерації матриця нечіткого розбиття має вигляд:

$$U = \begin{vmatrix} 0{,}941 & 0{,}994 & 0{,}492 & 0{,}383 & 0{,}048 & 0{,}093 \\ 0{,}059 & 0{,}006 & 0{,}508 & 0{,}617 & 0{,}952 & 0{,}907 \end{vmatrix}.$$

Аналізуючи значення коефіцієнтів належності цієї матриці робимо висновок, що до першого кластера необхідно віднести 1-й та 2-й об'єкти, а до другого кластера – 3-й, 4-й, 5-й і 6-й об'єкти.

6.2. ІЄРАРХІЧНИЙ КЛАСТЕРНИЙ АНАЛІЗ ДАНИХ У СЕРЕДОВИЩІ MATLAB

6.2.1. Функції MatLab для здійснення кластерного аналізу

Функції для здійснення *ієрархічного кластерного аналізу*:

1) *pdist(X,'metric')* – повертає парні міри близькості між об'єктами (по замовчуванню – відстань Евкліда), параметр *'metric'* дозволяє вказувати обрану міру близькості, параметр *X* задає матрицю вхідних даних;

2) *squareform()* – перетворює вектор парних мір близькості між об'єктами, визначений за допомогою функції *pdist(),* у симетричну квадратну матрицю близькості;

3) *linkage()* – повертає ієрархічне дерево кластерів, використовуючи за замовчуванням метод найближчого сусіда як метод зв'язку кластерів, результат використовується рядом інших функцій, зокрема функцією побудови дендрограми;

4) *dendrogram()* – повертає графічне відображення результату виконання функції *linkage()*;

5) *cophenet(Z,Y)* (де *Y* і *Z* – матриці, що повертаються, відповідно, функціями *pdist()* і *linkage()*) – повертає аналог коефіцієнта кореляції, який характеризує якість розбиття об'єктів набору даних на дерево кластерів (чим ближче до 1, тим краще);

6) *cluster()* – здійснює розбивку ієрархічного дерева кластерів на окремі кластери;

7) *clusterdata()* – здійснює групування матриці вихідних даних у кластери, еквівалентне послідовному застосуванню функцій *pdist, linkage* і *cluster*.

Функції для здійснення кластерного аналізу за *алгоритмами квадратичної похибки*:

1) *kmeans()* – кластеризація за алгоритмом чіткого розбиття k-means;

2) *fcm()* – кластеризація за алгоритмом нечіткого розбиття c-means.

Більш детальну інформацію про функції MatLab для здійснення кластерного аналізу та їх параметри наведено у додатках: *clusterdata()* (додаток *Л*), *kmeans()* (додаток *М*), *fcm()* (додаток *Н*).

6.2.2. Етапи ієрархічного кластерного аналізу в MatLab

1. Задаємо матрицю вхідних даних *X*: рядки – об'єкти набору даних, стовпці – ознаки об'єктів.

Примітка. Для задання даних можна скористатися функцією *rand()* (*наприклад, 10*rand(10,2)* – генерує дані у межах від 0 до 10 для матриці розміром 10x2).

2. Визначаємо парні міри близькості між об'єктами вхідного набору даних: *Y = pdist(X)*, якщо потрібно – вказуємо міру близькості як параметр функції *pdist()* (по замовчуванню це відстань Евкліда).

3. Для зручності конвертуємо *Y* у симетричну квадратну матрицю близькості (відстаней – для метричних даних): *S = squareform(Y)*.

4. Будуємо ієрархічне дерево кластерів: *Z = linkage(Y)*.

5. Будуємо графічне відображення результату виконання попередньої функції – дентрограму: *dendrogram(Z)*.

6. Розраховуємо коефіцієнт для визначення якості розбиття кластерів (чим ближче до 1, тим розбиття якісніше): *cophenet(Z,Y)*.

7. Визначаємо оптимальну кількість кластерів *k*, яка може бути розрахована як різниця між кількістю об'єктів набору даних та номером етапу кластеризації, на якому міра несхожості (відстань) між кластерами різко зростає (а міра подібності – різко спадає).

8. Отримуємо вектор *T*, елементи якого є номерами кластерів, до яких віднесені вхідні об'єкти (*k* – оптимальна кількість кластерів): *T = cluster(Z, k)* – розбиття для *k*-кластерів.

6.2.3. Проведення ієрархічного агломеративного кластерного аналізу засобами MatLab

Завдання 1. Розрахувати різними способами міри близькості між об'єктами, які характеризуються двома ознаками.

1. У вікні *Command Window* MatLab задаємо вхідні дані у вигляді матриці, в якій представлено 6 об'єктів із двома характеристиками кожен:

```
>>X = [2 8; 4 10; 5 7; 12 6; 14 6; 15 4]
```

2. Для розрахунку парних мір близькості між об'єктами в MatLab необхідно скористатися функцією *Y=pdist(X,'metric')*, де *X* – матриця вхідних даних, *'metric'* – спосіб визначення міри близькості. Для розрахунку близькості за відповідною мірою послідовно вводимо команди:

```
>>Y1 = pdist(X,'euclidean'); % відстань Евкліда
>>Y2 = pdist(X,'seuclidean'); % стандартизована відстань Евкліда (поділена на середнє квадратичне
% відхилення відповідних змінних)
>>Y3 = pdist(X,'mahalanobis'); % відстань Махаланобіса
>>Y4 = pdist(X,'cityblock'); % Манхеттенська відстань
>>Y5 = pdist(X,'minkowski'); % відстань Мінковського
>>Y6 = pdist(X,'cosine'); % відстань косинус
>>Y7 = pdist(X,'hamming'); % відстань Хеммінга
>>Y8 = pdist(X,'correlation'); % відстань кореляції
```

3. Конвертуємо розраховані міри близькості у симетричні квадратні матриці близькості (відстані – для метричних даних):

```
>>D1 = squareform(Y1);
>>D2 = squareform(Y2);
>>D3 = squareform(Y3);
>>D4 = squareform(Y4);
>>D5 = squareform(Y5);
>>D6 = squareform(Y6);
>>D7 = squareform(Y7);
>>D8 = squareform(Y8);
```

5. Виведіть на екран отримані матриці близькості та порівняйте їх значення, розраховані з використанням різних мір близькості.

Завдання 2. Засобами MatLab провести ієрархічну кластеризацію об'єктів, кожен із яких характеризується двома ознаками. Маємо 6 об'єктів, які характеризуються двома ознаками: x – обсяг продукції, що випускається, та y – середньорічна вартість основних промислово-виробничих фондів (табл. 6.8).

Таблиця 6.8

Значення ознак об’єктів

№ з/п	1	2	3	4	5	6
x_1	2	4	5	12	14	15
x_2	8	10	7	6	6	4

1. Для задання матриці, яка містить вхідні дані, необхідно ввести команду:

>>X = [2 8; 4 10; 5 7; 12 6; 14 6; 15 4]

де X – матриця, що містить дані про 6 об’єктів з 2-ма характеристиками.

2. Для розрахунку парних відстаней між об’єктами в множині X введено у командному вікні команду:

>>Y=pdist(X)

Отримаємо Y – вектор розрахованих попарних відстаней між об’єктами – матрицю відстаней (по замовчуванню – відстань Евкліда).

3. Для перетворення вектору вихідних даних Y функції *pdist()* у симетричну квадратну матрицю відстаней введемо у командному вікні команду:

>>M= squareform(Y)

Отримаємо симетричну квадратну матрицю відстаней M.

4. Для формування ієрархічного дерева кластерів (рис. 6.1) скористаємося функцією *linkage*, ввівши у командному вікні MatLab команду:

>>Z=linkage(Y)

По замовчуванню для об’єднання кластерів буде використано метод зв’язку найближчого сусіда.

5. Розглянемо структуру матриці *Z*, яка містить дані про етапи процесу кластеризації (рис. 6.1):

a) перший рядок матриці Z містить номери об’єктів 4 і 5, які були об’єднані у кластер на першому етапі кластеризації, та 2 – відстань між цими кластерами, яка була на цьому етапі мінімальною (критерій кластеризації). Утвореному кластеру привласнили номер 7.

b) другий рядок матриці Z містить номери об’єкта 6 та кластера 7, які були об’єднані у кластер на другому етапі кластеризації, та 2.2361 – відстань між кластерами на цьому етапі. Утвореному кластеру привласнили номер 8.

c) третій рядок матриці Z містить номери об’єктів 1 і 2, які були об’єднані у кластер на третьому етапі кластеризації, та 2.8284 – відстань між кластерами на цьому етапі. Утвореному кластеру привласнили номер 9.

d) четвертий рядок матриці Z містить номери об’єкта 3 та кластера 9, які були об’єднані у кластер на четвертому етапі, та 3.1623 – відстань між кластерами на цьому етапі. Утвореному кластеру привласнили номер 10.

e) п’ятий рядок матриці Z містить номери кластерів 8 і 10, які були об’єднані у кластер на цьому етапі, та 7.0711 – відстань між кластерами. На цьому етап кластеризації завершується – усі об’єкти об’єднані у один кластер.

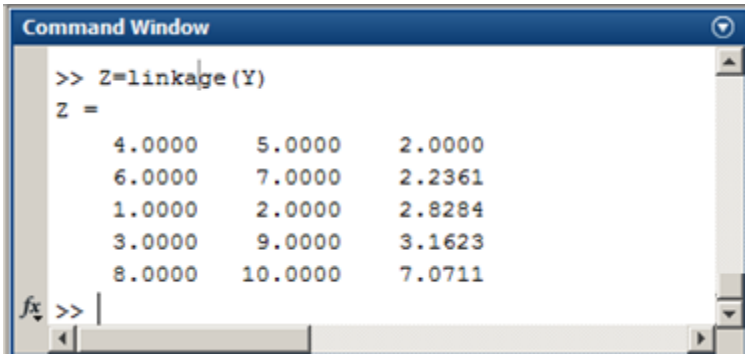


Рис. 6.1. Формування ієрархічного дерева кластерів

6. Представимо побудоване дерево кластерів у вигляді дендрограми. Для цього скористаємося функцією *dendrogram*, ввівши у командному вікні MatLab команду: H = dendrogram(Z). У окремому вікні буде виведена побудована дендрограма (рис. 6.2).

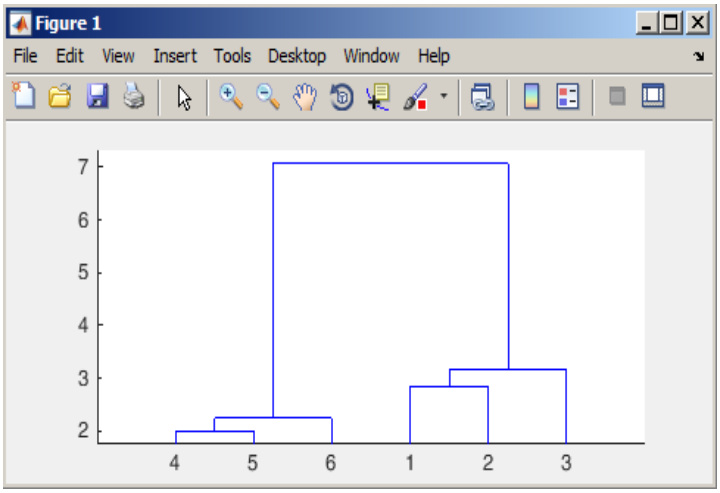


Рис. 6.2. Дендрограма Z, побудована за методом ближнього сусіда

7. Порівняємо результати, отримані при застосуванні різних методів зв'язку кластерів при їх об'єднанні під час ієрархічної кластеризації, ввівши послідовно у командному вікні MatLab команди:

```
>>Z1 = linkage(Y, 'complete') % метод дальнього сусіда
>>dendrogram(Z1)
>>Z2 = linkage(Y, 'average') % метод середнього зв'язку
>>dendrogram(Z2)
>>Z3 = linkage(Y, 'centroid') % центроїдний метод
>>dendrogram(Z3)
>>Z4 = linkage(Y, 'ward') % метод Уорда
>>dendrogram(Z4)
```

Дендрограми, утворені за різними методами зв'язку кластерів, представлено на рисунках 6.3-6.6.

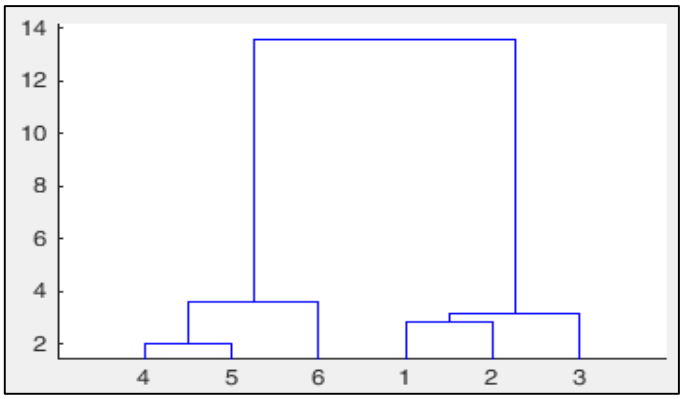


Рис. 6.3. Дендрограма Z1, побудована за методом дальнього сусіда

Як бачимо, різні методи зв'язку дають схожий результат розбиття вхідного набору даних на кластери.

8. Здійснимо аналіз динаміки змін відстаней між кластерами на етапах кластеризації. Візуально це можна зробити, порівнюючи утворені дендрограми, аналітично – порівнюючи значення матриць Z, Z1, Z2, Z3, Z4. Робимо висновок, що оптимальна кількість кластерів, на яку необхідно розбити вхідну множину даних, дорівнює 2 (відстань між кластерами різко збільшилася після 4-го етапу: 6 – 4 = 2). До одного з кластерів необхідно віднести об'єкти 1, 2 і 3, а до іншого – 4, 5, і 6.

9. Оцінимо якість розбиття об'єктів набору даних на два кластери:

```
>> K=cophenet(Z,Y)
```

У вікні *Command Window* отримаємо K = 0,9249. Значення K близьке до одиниці, що свідчить про досить високу якість розбиття набору даних на 2 кластери.

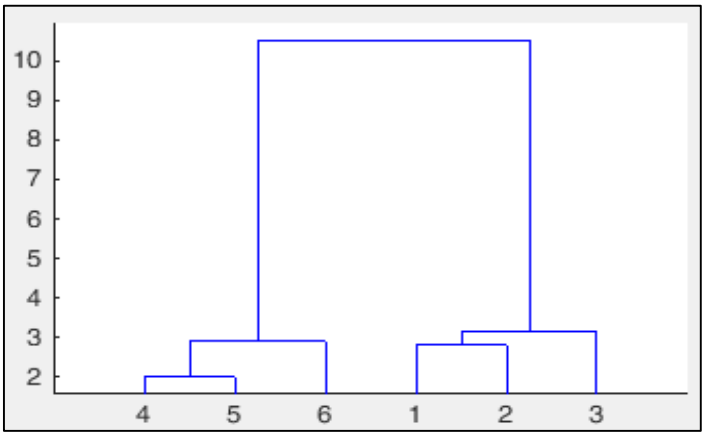


Рис. 6.4. Дендрограма Z2, побудована за методом середнього зв'язку

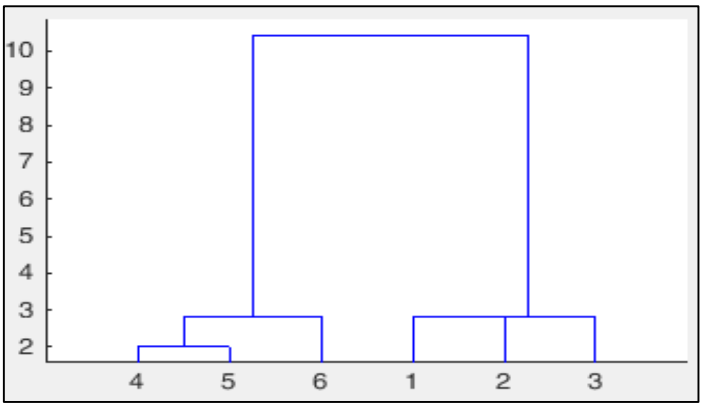


Рис. 6.5. Дендрограма Z3, побудована за центроїдним методом

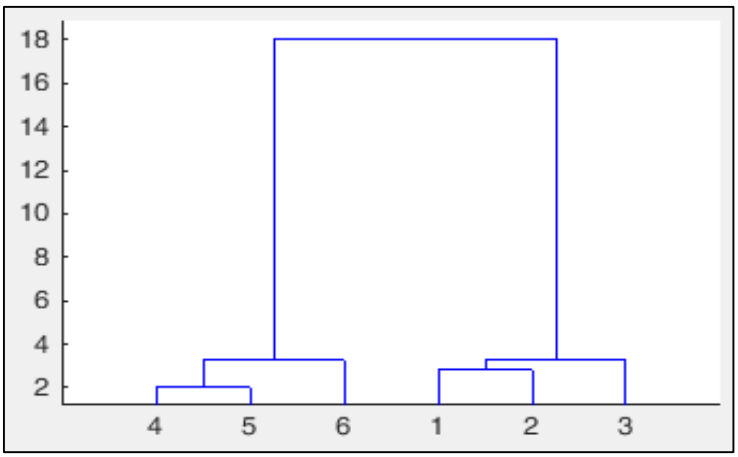


Рис. 6.6. Дендрограма Z4, побудована за методом Уорда

10. Сформуємо матрицю T, яка буде містити розбиття множини даних на два кластери:

```
>> T = cluster(Z,2)
```

11. Здійснимо розбиття набору даних на 2 кластери з використанням функції *clusterdata()*:

```
>> T1 =  clusterdata(X,2)
```

Як бачимо, матриці T та T1 є ідентичними, оскільки застосування функції *clusterdata()* еквівалентне послідовному застосуванню функцій *pdist(), linkage()* і *cluster()*.

12. Відобразимо графічно точки кластерів розбивки T із допомогою функції *scatter()*, ввівши команду:

```
>>scatter(X(:,1),X(:,2),100, T, 'filled')
```

В окремому вікні буде виведено графічне зображення розбиття набору даних на два кластери (рис. 6.7).

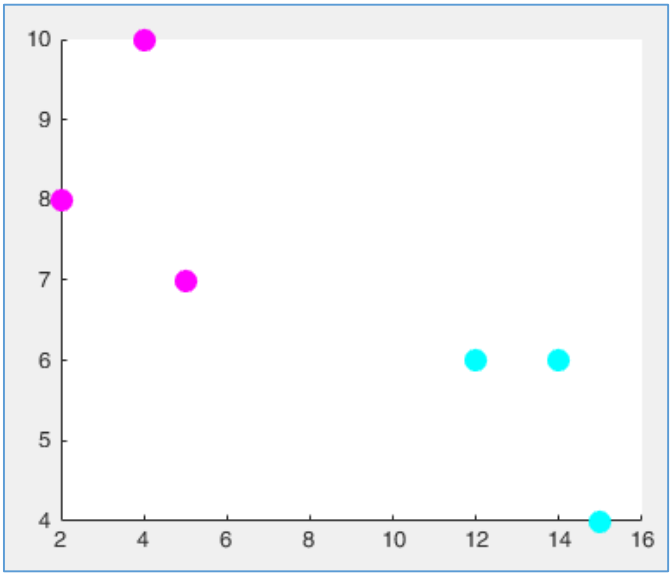


Рис. 6.7. Графічне зображення розбиття T множини X на 2 кластери

13. Створимо та збережемо у m-файлі сценарій для здійснення ієрархічної кластеризації:

```
% ієрархічна кластеризація
X=[2 8; 4 10; 5 7; 12 6; 14 6; 15 4]
Y=pdist(X);
Y
Z=linkage(Y)
H= dendrogram (Z)
K=cophenet(Z,Y)
T = cluster(Z,2)
```

14. Продемонструємо роботу сценарію, ввівши у вікні *Command Window* ім'я створеного файлу-сценарію.

15. Створені змінні Робочої області збережіть у mat-файлі.

6.3. КЛАСТЕРНИЙ АНАЛІЗ ДАНИХ ЗА АЛГОРИТМАМИ КВАДРАТИЧНОЇ ПОХИБКИ У СЕРЕДОВИЩІ MATLAB

6.3.1. Етапи кластерного аналізу в MatLab за алгоритмом чіткої кластеризації k-means

1. Задаємо матрицю вхідних даних *X*: рядки – об'єкти набору даних, стовпці – ознаки об'єктів.
2. Із урахуванням аналізу значень ознак об'єктів набору даних задаємо значення *k* – кількість кластерів.
3. Розділяємо множину об'єктів набору даних на k кластерів із використанням функції *kmeans()*:

$$[IDX,C,sumd,D] = kmeans(X, k);$$

3. Отримуємо результат кластеризації:

a) *IDX*: вектор-стовпець, що вказує належність кожного об'єкта до певного кластера;
b) C: матрицю координат центрів ваги кластерів – центроїдів;
c) *sumd*: вектор сум відстаней від об'єктів кожного кластера до його центроїда;
d) *D*: матрицю відстаней від кожного об'єкта до кожного центроїда – центра кластера.

4. По замовчуванню між об'єктами буде розраховано квадрат відстані Евкліда. У разі необхідності може бути використана інша міра близькості, яку необхідно вказати як додатковий параметр функції *kmeans()* (див. додаток *M*).

6.3.2. Етапи нечіткого кластерного аналізу в MatLab за алгоритмом c-means

1. Задаємо матрицю вхідних даних *X*: рядки – об'єкти набору даних, стовпці – ознаки об'єктів.

2. З урахуванням аналізу значень ознак об'єктів набору даних задаємо значення *k* – кількість нечітких кластерів.

3. У разі необхідності вказуємо значення інших параметрів функції *fcm()*, відмінних від установлених по замовчуванню (див. додаток *H*):

a) мінімальну кількість ітерацій алгоритму (по замовчуванню – 100);

b) показник ступеня належності *w* (коефіцієнт нечіткості, по замовчуванню – 2);

c) мінімальну величину досягнення точності – покращення цільової функції за одну ітерацію (по замовчуванню – 0,00001).

4. Здійснюємо нечітке розбиття множини об'єктів набору даних, заданого матрицею вхідних даних *X* на *k* кластерів із використанням функції *fcm()*: [*C, U, J*] = *fcm*(*X, k*);

5. Отримуємо результуючі дані, які повертає функція *fcm()*:

a) кількість етапів ітерації і значення цільової функції *J* на кожному етапі;

b) матрицю координат центрів ваги кластерів *C* – кожен рядок містить координати центроїдів окремого кластера;

c) кінцеву матрицю *U* нечіткого розбиття об'єктів набору даних на кластери.

6. Результатом проведеного кластерного аналізу буде віднесення кожного об'єкта до того кластера, до якого він у матриці нечіткого розбиття має найбільший коефіцієнт належності.

6.3.3. Проведення кластерного аналізу за алгоритмами k-means і c-means засобами MatLab

Завдання 3. Провести чітку та нечітку кластеризацію об'єктів, кожен з яких характеризується двома ознаками за алгоритмами k-means та c-means.

1. Маємо 6 об'єктів, які характеризуються двома ознаками: x_1 та x_2 (табл. 6.6). У вікні *Command Window* вводимо значення матриці вхідних даних та кількості кластерів:

```
>> X = [2 8; 4 10; 5 7; 12 6; 14 6; 15 4];      k = 2;
```

2. Для здійснення чіткої кластеризації *методом k-means* розділимо вхідну множину об'єктів на 2 кластери з використанням функції *kmeans()*. Основні та додаткові параметри функції наведено у додатку *M*. У вікні *Command Window* введемо команди:

```
>> [IDX,C] = kmeans(X,k);     IDX=IDX';
```

3. Виведемо у вікні *Command Window* значення IDX та C (рис. 6.8).

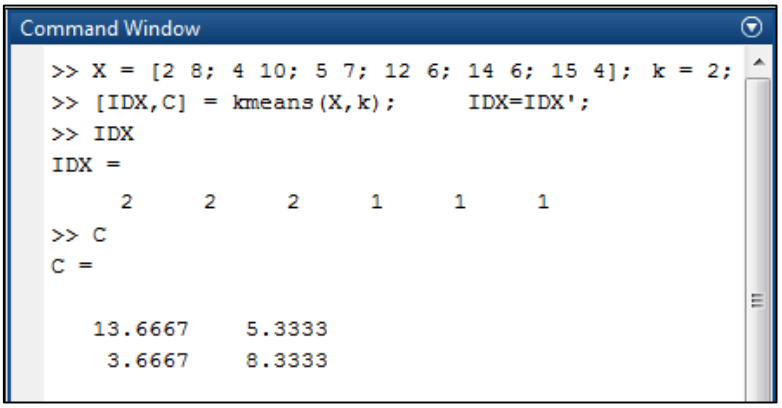


Рис. 6.8. Здійснення чіткої кластеризації методом k-means

Отримаємо результат:

IDX – вектор-стовпець, який вказує належність об'єкта до певного кластера;

C – вектор центроїдів кластерів.

Як бачимо, розбиття набору даних на кластери за алгоритмом k-means співпадає з розбиттям, виконаним у попередньому завданні за ієрархічним агломеративним алгоритмом (вміст векторів T та IDX співпадають).

4. Розрахуємо суми відстаней від об'єктів кожного кластера до його центроїда *sumd* та матрицю відстаней від кожного об'єкта до кожного центроїда – центра кластера *D* (рис. 6.9).

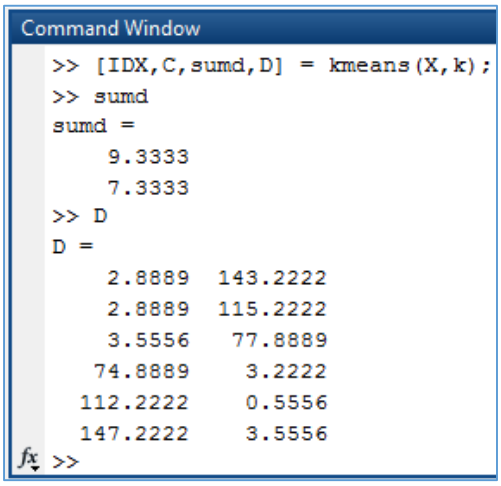


Рис. 6.9. Розрахунок вектора сум відстаней від об'єктів кожного кластера до його центроїда *sumd* та матриці відстаней від кожного об'єкта до центру кожного кластера *D*

5. У вікні *Command Window* введемо команди для здійснення нечіткої кластеризації *методом c-means*, розділивши вхідну множину об'єктів X на 2 кластери з використанням функції *fcm()* (рис. 6.10). Основні та додаткові параметри функції *fcm()* наведено у додатку *H*.

```
>> X = [2 8; 4 10; 5 7; 12 6; 14 6; 15 4];
>> [center,U,objFcn] = fcm(X,2)
Iteration count = 1, obj. fcn = 125.739595
Iteration count = 2, obj. fcn = 28.394927
Iteration count = 3, obj. fcn = 16.237428
Iteration count = 4, obj. fcn = 16.162482
Iteration count = 5, obj. fcn = 16.162146
Iteration count = 6, obj. fcn = 16.162145
center =
    3.6501    8.3486
   13.6821    5.3360
U =
    0.9806    0.9759    0.9555    0.0417    0.0048    0.0233
    0.0194    0.0241    0.0445    0.9583    0.9952    0.9767
objFcn =
  125.7396
   28.3949
   16.2374
   16.1625
   16.1621
   16.1621
```

Рис. 6.10. Здійснення нечіткої кластеризації методом c-means

6. Проведемо аналіз отриманого результату нечіткої кластеризації. Розбиття множини об'єктів на 2 кластери здійснювалося у 6 етапів.

На першому кроці ітерації *count = 1* цільова функція *obj.fcn* мала значення 125,739595.

На 5-му кроці ітерації *count = 5* цільова функція *obj.fcn* мала значення 16,162146.

На 6-му кроці ітерації *count = 6* цільова функція *obj.fcn* мала значення 16,162145, зміна її значення порівняно з попереднім етапом становила 0,000001. Така різниця задовольняє параметр зупинки алгоритму, тому на цьому етапі процес кластеризації було закінчено.

Матриця *center* містить координати центрів кластерів, *U* є кінцевою матрицею нечіткого розбиття об'єктів на два кластери.

7. У вікні *Command Window* введемо команди для здійснення візуалізації нечіткого розбиття на кластери (рис. 6.11).

8. Результатом виконання вказаних команд буде графічне зображення кластерів (рис. 6.12). Чорним кольором виділені центри 2-х кластерів, синіми кружками – об’єкти першого кластера, червоними трикутниками – об’єкти другого кластера.

```
Command Window
>> % знаходження максимальних значень матриці
>> maxU=max(U);
>> ptsymb = {'bs','r^','md','go','c+'};
>> for i = 1:2
% формування масиву номерів об'єктів кластеру
clust = find(U(i,:)==maxU);
% виведення графіку обєктів кластера
plot(X(clust,1),X(clust,2), ptsymb{i});
% виведення центру кластера
hold on
scatter(center(i,1),center(i,2), 100,'ko','filled');
end
```

Рис. 6.11. Команди для здійснення візуалізації нечіткого розбиття на кластери

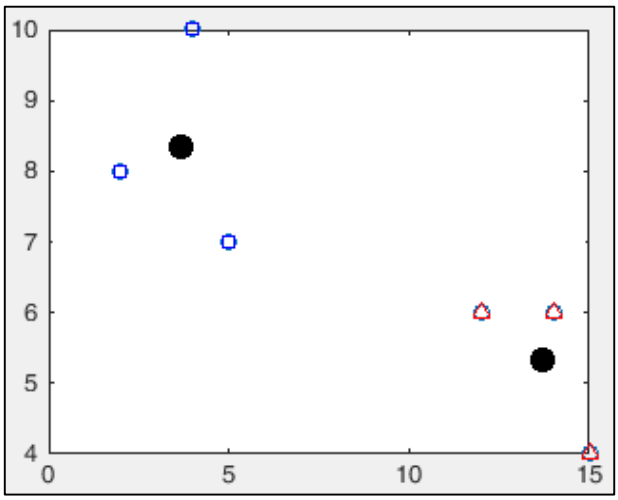


Рис. 6.12. Графічне зображення нечіткого розбиття набору даних на 2 кластери

6.4. ЗАВДАННЯ ДЛЯ САМОСТІЙНОЇ РОБОТИ

Завдання 4. Написати програму мовою MatLab, яка здійснює:

1) нормалізацію (стандартизацію) значень змінних вхідного набору даних;

2) побудову матриці близькості (відстані) об’єктів набору даних за мірою близькості, обраною самостійно;

3) ієрархічну кластеризацію за вказаним відповідно до варіанта методом зв’язку кластерів під час їх об’єднання;

4) чітку та нечітку кластеризацію за алгоритмами k-means і c-means (k – кількість кластерів обрати рівною оптимальній кількісті кластерів, визначеній за ієрархічним алгоритмом кластеризації на попередньому етапі);

5) візуалізацію результатів проведеного за різними алгоритмами кластерного аналізу.

Для формування матриці вхідних даних за індивідуальними варіантами необхідно скористатися таблицею 6.9 та таблицею 6.10.

Деякі рекомендації до виконання завдань самостійної роботи.

1. При візуалізації результатів аналізу можна скористатися функцією plot(X,Y,S), яка аналогічна функції plot(X,Y), проте тип лінії графіка можна задавати за допомогою рядкової констранти S. Значення константи S, які відповідають різним типам лінії графіка, наведені у додатку *3*.

2. Для виведення 3-х мірних графіків (як у завданні самостійної роботи) необхідно використовувати функції *plot3()* та *scatter3()*. Їх синтаксис аналогічний синтаксису функцій *plot()* та *scatter()*, однак додається ще одна змінна.

3. Для виведення декількох графіків у різних вікнах необхідно перед кожним *plot* (*scatter*) писати команду *figure* – вона створює нове графічне вікно.

4. Для визначення кількості рядків та стовпців матриці X необхідно ввести команди:

```
>>g = size(X);
>>n = g(:,1);  % визначення кількості рядків
>>m=g(:,2);  % визначення кількості стовпців
```

КОНТРОЛЬНІ ПИТАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ № 6

1. Типи алгоритмів кластерного аналізу.
2. Алгоритми квадратичної похибки.
3. Чіткий кластерний аналіз, алгоритм k-means.
4. Якими є основні етапи алгоритму кластеризації k-means?
5. У чому полягають переваги та недоліки алгоритму k-means.
6. Нечіткий кластерний аналіз, алгоритм c-means.
7. Якими є основні етапи алгоритму c-means?
8. Функції MatLab для здійснення кластерного аналізу.
9. Етапи здійснення ієрархічного кластерного аналізу в MatLab.
10. Етапи здійснення кластерного аналізу в MatLab за алгоритмом чіткої кластеризації k-means.
11. Етапи здійснення нечіткого кластерного аналізу за алгоритмом c-means засобами MatLab.

Таблиця 6.9

Індивідуальні варіанти даних для виконання завдання 4

№ вар.	Змінні для побудови матриці вхідних даних X	Метод об’єднання кластерів
1	Із 1-го варіанта завдань таблиці 6.10: X_1, X_2, X_3	Найближчого сусіда
2	Із 2-го варіанта завдань таблиці 6.10: X_1, X_2, X_3	Найдальшого сусіда
3	Із 3-го варіанта завдань таблиці 6.10: X_1, X_2, X_3	Середнього зв’язку
4	Із 4-го варіанта завдань таблиці 6.10: X_1, X_2, X_3	Метод Уорда
5	Із 5-го варіанта завдань таблиці 6.10: X_1, X_2, X_3	Найближчого сусіда
6	Із 6-го варіанта завдань таблиці 6.10: X_1, X_2, X_3	Найдальшого сусіда
7	Із 7-го варіанта завдань таблиці 6.10: X_1, X_2, X_3	Середнього зв’язку
8	Із 8-го варіанта завдань таблиці 6.10: X_1, X_2, X_3	Метод Уорда
9	Із 9-го варіанта завдань таблиці 6.10: X_1, X_2, X_3	Найближчого сусіда
10	Із 10-го варіанта завдань таблиці 6.10: X_1, X_2, X_3	Найдальшого сусіда
11	Із 11-го варіанта завдань таблиці 6.10: X_1, X_2, X_3	Середнього зв’язку
12	Із 1-го варіанта завдань таблиці 6.10: X_4, X_5, X_6	Метод Уорда
13	Із 2-го варіанта завдань таблиці 6.10: X_4, X_5, X_6	Найближчого сусіда
14	Із 3-го варіанта завдань таблиці 6.10: X_4, X_5, X_6	Найдальшого сусіда
15	Із 4-го варіанта завдань таблиці 6.10: X_4, X_5, X_6	Середнього зв’язку
16	Із 5-го варіанта завдань таблиці 6.10: X_4, X_5, X_6	Метод Уорда
17	Із 6-го варіанта завдань таблиці 6.10: X_4, X_5, X_6	Найближчого сусіда
18	Із 7-го варіанта завдань таблиці 6.10: X_4, X_5, X_6	Найдальшого сусіда
19	Із 8-го варіанта завдань таблиці 6.10: X_4, X_5, X_6	Середнього зв’язку
20	Із 9-го варіанта завдань таблиці 6.10: X_4, X_5, X_6	Метод Уорда
21	Із 10-го варіанта завдань таблиці 6.10: X_4, X_5, X_6	Найближчого сусіда
22	Із 11-го варіанта завдань таблиці 6.10: X_4, X_5, X_6	Найдальшого сусіда

№ вар.	Змінні для побудови матриці вхідних даних X	Метод об'єднання кластерів
23	Із 1-го варіанта завдань таблиці 6.10: X_7, X_8, X_9	Середнього зв'язку
24	Із 2-го варіанта завдань таблиці 6.10: X_7, X_8, X_9	Метод Уорда
25	Із 3-го варіанта завдань таблиці 6.10: X_7, X_8, X_9	Найближчого сусіда

Таблиця 6.10

Варіанти для формування матриці вхідних даних

№ з/п	X1	X2	X3	X4	X5	X6	X7	X8	X9
1	64	0,29	340	2710	304,92	8368,7	4,98953	4,43939	19602
	31	0,73	370	2770	279,51	8483,3	7,67620	3,10606	7194
	39	0,69	470	2780	415,03	4098,3	5,47801	1,90909	9504
	44	0,54	290	1280	145,68	4728,9	4,74529	1,86363	11814
	87	0,64	340	1460	272,73	5674,6	5,61758	1,84848	15378
	74	0,76	360	1220	448,91	2866	8,72295	1,27272	5148
	54	0,54	710	2480	149,07	8110,7	5,1291	4,10606	18084
	74	0,64	770	1790	220,22	7652,2	6,28053	2,80303	13992
	78	0,85	300	1690	150,76	2780,0	9,35101	4,75757	17622
	87	0,39	560	2770	504,42	8311,4	6,07117	3,96969	19074
2	25	0,82	430	2590	164,31	5416,7	4,50104	4,46969	20262
	67	0,29	270	2870	492,95	3225,1	10,8513	2,98484	7326
	62	0,52	860	1920	528,52	4069,7	9,24633	3,19697	13530
	53	0,54	790	2770	133,82	3926,4	4,67550	4,65151	13068
	42	0,42	610	1100	232,07	5474,0	4,29169	1,72727	6270
	64	0,29	700	2860	264,65	7107,6	9,69993	2,45454	14916
	30	0,57	550	2150	531,91	6305,2	8,72295	2,18181	8646
	41	0,88	800	1840	499,73	6276,5	6,28053	2,10606	12342
	68	0,37	740	1220	381,15	6047,2	5,82693	4,50000	20592
	49	0,58	490	2430	474,32	4384,9	4,57083	4,22727	8976
3	76	0,49	750	3100	247,32	5560,0	8,65317	3,62121	11616
	35	0,49	380	1850	152,46	5760,6	4,04745	1,62121	5214
	43	0,38	290	840	138,90	4958,1	10,5722	1,34848	14916
	52	0,57	860	2030	218,52	8769,9	10,9909	3,28787	10164
	72	0,26	620	1450	203,28	8827,2	9,76971	2,43939	6270
	73	0,28	460	1360	470,93	7824,1	6,62944	2,62121	8844
	88	0,37	75	2230	433,66	8712,6	10,8513	3,96969	94338
	42	0,59	770	1520	252,40	7795,5	10,6420	3,92424	5412
	69	0,72	740	990	320,16	3983,7	3,66364	4,45545	6204
	75	0,31	380	2830	135,52	4470,9	8,51360	4,33333	18150
4	33	0,26	390	2420	526,83	2636,7	3,24494	4,09090	19338
	41	0,29	760	1170	164,31	5846,6	8,09490	4,40909	20328
	69	0,75	640	1890	531,91	5216,1	10,1535	1,68181	11352
	6	0,54	860	2880	177,87	5846,1	3,21004	4,07575	13398
	38	0,52	360	2730	238,85	5531,3	3,66364	4,65151	19536
	57	0,50	680	1230	476,01	7709,5	8,44382	3,06060	8316
	58	0,59	410	2620	367,59	2264,1	2,82623	2,16666	7326
	33	0,69	310	1280	513,28	8110,7	5,23377	3,72727	18216
	60	0,89	800	1630	215,13	6878,4	8,79274	2,31818	14058
	32	0,76	500	2370	216,83	2350,1	6,14096	1,40909	17028

№ з/п	X1	X2	X3	X4	X5	X6	X7	X8	X9
5	59	0,46	460	2300	442,13	4528,2	6,38520	3,83333	13464
	68	0,5	470	1560	282,89	3238,5	3,83810	3,09090	8316
	34	0,34	330	2070	430,27	7050,3	3,03559	3,03559	17754
	38	0,76	400	850	509,89	6706,4	8,19958	8,19958	19866
	56	0,43	890	1140	164,31	5674,6	5,58269	5,58269	5610
	84	0,65	870	1860	203,28	6591,8	4,08234	4,08234	13398
	86	0,6	540	820	166,01	2665,3	5,16399	5,16399	16500
	87	0,76	310	1740	481,09	7594,9	9,66503	9,66503	15840
	60	0,89	250	1230	370,98	7852,8	2,68667	2,68667	19602
	28	0,58	870	1670	210,05	4442,3	6,87369	6,87369	7062
6	86	0,41	510	2140	218,52	8941,9	10,2233	3,31818	17094
	81	0,33	400	1270	226,99	5359,4	5,26866	4,45454	18084
	58	0,74	630	2670	440,44	8769,9	4,67550	2,15151	6336
	40	0,57	490	810	159,23	5703,3	3,38450	2,92424	11352
	55	0,67	460	3060	181,25	4155,7	4,56593	1,98484	9306
	53	0,42	390	920	404,86	8110,7	5,09420	4,07575	9636
	44	0,78	400	2730	272,73	4155,7	4,81507	2,30303	11088
	59	0,79	310	980	143,99	4384,9	8,40893	1,59090	15246
	29	0,45	890	2230	164,31	4069,7	10,9211	1,60606	16962
	88	0,70	330	1890	274,42	2722,7	5,02442	4,18181	7062
7	25	0,42	580	1810	399,78	5101,4	8,12979	4,34848	20790
	69	0,7	800	3040	362,51	5646,0	2,79134	1,54545	10164
	74	0,46	570	2950	459,07	8082,1	3,52407	1,62121	17490
	72	0,52	360	1810	499,73	7766,8	3,83810	3,92424	7062
	86	0,68	790	1610	481,09	2206,8	10,0837	1,40909	20460
	63	0,51	790	2460	442,13	2436,1	10,7815	1,25757	9108
	55	0,62	770	2740	171,09	2579,4	6,21074	2,68181	11946
	84	0,83	820	990	433,66	6047,2	10,3628	2,60606	12540
	51	0,54	740	1610	447,21	4127,0	5,05931	4,77272	15708
	87	0,31	490	870	509,89	2780,0	7,25750	4,34848	15642
8	53	0,5	440	2510	333,71	5273,4	7,53660	4,77272	5214
	36	0,65	350	860	282,89	3840,4	9,63014	1,90909	19866
	32	0,88	550	2350	506,50	8282,7	8,33914	3,25757	9504
	71	0,46	430	1830	171,09	5187,4	7,11793	1,90909	7194
	67	0,89	750	3090	501,42	8082,1	6,35031	3,30303	5940
	77	0,43	330	1290	225,30	5388,1	5,23377	1,46969	5280
	40	0,64	290	2840	465,85	5072,8	5,02442	3,71212	13926
	54	0,86	850	2720	354,04	4872,2	4,44615	1,68181	18612
	49	0,68	820	3130	182,95	3152,6	10,6769	2,62121	17754
	49	0,71	250	2190	409,94	2436,1	7,85066	2,54545	19536
9	48	0,54	800	1670	242,24	7222,3	10,5024	1,78787	18480
	29	0,54	250	1870	154,15	7136,3	8,82763	1,68181	16962
	82	0,62	540	790	169,4	4213,0	6,55966	4,66666	19932
	53	0,76	440	1750	169,4	7938,8	8,16468	3,06060	13200
	68	0,88	820	1000	262,57	1514,8	2,89602	3,75757	6534
	36	0,69	530	2480	188,03	3295,9	5,37334	1,74242	14520
	82	0,86	550	2700	304,92	8827,2	2,96580	1,57575	12408
	82	0,86	820	2980	389,62	6161,9	7,99023	2,37878	10296
	27	0,73	410	1200	514,97	6563,1	7,67620	3,09090	8712
	80	0,34	690	1380	409,94	8368,7	9,59525	4,65151	14916

№ з/п	X1	X2	X3	X4	X5	X6	X7	X8	X9
10	80	0,7	520	2690	154,15	5273,4	10,3977	3,87878	11352
	75	0,39	600	1460	525,14	4213,0	4,53593	3,42424	15444
	83	0,38	490	3130	469,23	7021,7	9,24633	4,27272	6204
	37	0,64	870	890	499,73	6419,8	6,35031	4,39393	7524
	72	0,77	440	1900	398,09	4127,0	10,7117	3,33333	9240
	68	0,31	690	1370	437,05	3525,1	10,8164	3,45454	9570
	40	0,68	460	1940	269,34	3754,4	3,52407	2,74242	8052
	71	0,67	730	1400	367,59	6849,7	8,40893	3,68181	17160
	83	0,75	330	2550	479,40	7365,6	5,16399	4,74242	20394
	89	0,51	440	3110	260,87	7337,5	5,1291	3,53030	8976
11	89	0,51	610	2180	160,93	4786,2	7,78087	1,46969	7788
	36	0,81	270	1580	453,99	2522,0	4,71039	3,07575	7920
	83	0,79	730	2110	406,56	3353,2	8,09490	3,62121	20592
	82	0,54	280	2890	311,69	4814,8	7,32728	3,5000	19404
	25	0,53	760	1190	176,17	7824,1	4,32658	3,98484	5940
	67	0,69	730	2680	462,46	7365,6	4,08234	1,27272	18942
	85	0,51	400	1160	518,36	6849,7	10,0139	1,56060	13596
	62	0,54	330	2510	287,98	7623,5	7,92044	4,77272	10824
	35	0,36	390	1820	428,58	5588,7	4,95464	2,10606	16038
	53	0,28	660	2560	428,58	7623,5	6,00139	1,78787	18282

7. ЗАДАЧА КЛАСИФІКАЦІЇ. ДИСКРИМІНАНТНИЙ АНАЛІЗ ДАНИХ

Лабораторна робота № 7

Мета: закріплення знань про сутність, базові поняття класифікації, етапи дискримінантного аналізу даних. Набуття навичок проведення дискримінантного аналізу засобами MS Excel та MatLab. Закріплення навиків сумісної роботи з даними у середовищах MatLab та MS Excel.

Теоретичні знання: базові поняття задачі класифікації. Проблеми недонавчання та перенавчання. Оцінка точності, ефективності класифікатора. Матриця помилок. Дискримінантний аналіз даних. Побудова дискримінантної функції. Матриця центрованих значень. Константа детермінації. Умова оптимальності поділу об'єктів на класи. Етапи алгоритму обчислень за методом дискримінантного аналізу.

7.1. ЗАДАЧА КЛАСИФІКАЦІЇ В DATA MINING

7.1.1. Базові поняття задачі класифікації

У процесі здійснення аналізу даних часто необхідно визначити, до якого з відомих класів відносяться досліджувані об'єкти, тобто ***класифікувати*** їх.

Метод, за допомогою якого проводять класифікацію, називають ***класифікатором***.

У Data Mining існує багато методів для розв'язання задачі класифікації. Ці методи дозволяють виявити відмінності між групами об'єктів – ***класами***, і надають можливість класифікувати нові об'єкти за принципом максимальної подібності.

Ознаки, які використовуються для того, щоб відрізняти об'єкти одного класу від об'єктів іншого, називають ***незалежними змінними***, а ознаку, за якою здійснюється класифікація об'єктів, – ***залежною змінною***.

Якщо значеннями незалежних і залежної змінної є дійсні числа, то задача називається ***задачею регресії.*** У ***задачі класифікації*** множина значень залежної змінної є скінченною.

Приклад 1. У випадку фільтрації електронної пошти необхідно класифікувати вхідне повідомлення як *спам* (небажана електронна пошта) або як *лист*. Таке рішення приймається на підставі частоти появи у повідомленні певних слів. У цьому прикладі незалежні змінні – частоти слів, залежна змінна – тип повідомлення (можливі значення змінної «spam» і «mail»).

Приклад 2. При оформленні кредиту банківському прицівнику потрібно класифікувати клієнта: визначити, є він кредитоспроможним чи ні. Це рішення приймається на підставі даних про досліджуваний об'єкт – людину, яка звернулася в банк за наданням кредиту. Тут незалежними змінними є: розмір заробітної плати, вік, кількість дітей, наявність об'єктів нерухомості. А залежна змінна – кредитоспроможність клієнта (можливі значення змінної «висока», «середня» і «низька»).

Формально задачу класифікації можна описати наступним чином. Нехай:

$I = \{i_1, i_2, \dots, i_n\}$ – множина об'єктів набору даних, n – кількість об'єктів.

Кожен об'єкт i_j множини I представлений набором атрибутів – значень змінних, серед яких виділяють незалежні змінні та залежну змінну.

$C = \{c_1, c_2, \dots, c_k\}$ – множина значень залежної змінної, яка містить наявні класи, k – кількість класів.

Необхідно побудувати класифікатор $I \to C$: метод, здатний класифікувати довільний об'єкт із множини I, ставлячи йому у відповідність певне значення з множини C.

Розв'язання задачі класифікації здійснюється на наборі даних із об'єктами, для яких відомі значення як незалежних змінних, так і залежної змінної. Тому побудову моделі класифікатора розглядають як *навчання з учителем*.

Набір даних розбивають на дві множини:

a) ***навчаюча множина*** (англ. *Training Set*): містить об'єкти з відомими значеннями незалежних і залежної змінних, які використовуються для побудови класифікатора – моделі визначення залежної змінної;

b) ***тестова множина*** (англ. *Test Set*): також містить об'єкти з відомими значеннями незалежних і залежної змінних, які використовуються для перевірки працездатності моделі.

Побудована на основі навчаючої множини ***модель класифікатора*** може бути представлена:

a) математичною формулою;

b) класифікаційними правилами;

c) деревом рішень;

d) комп'ютерним об'єктом (наприклад, нейронною мережею).

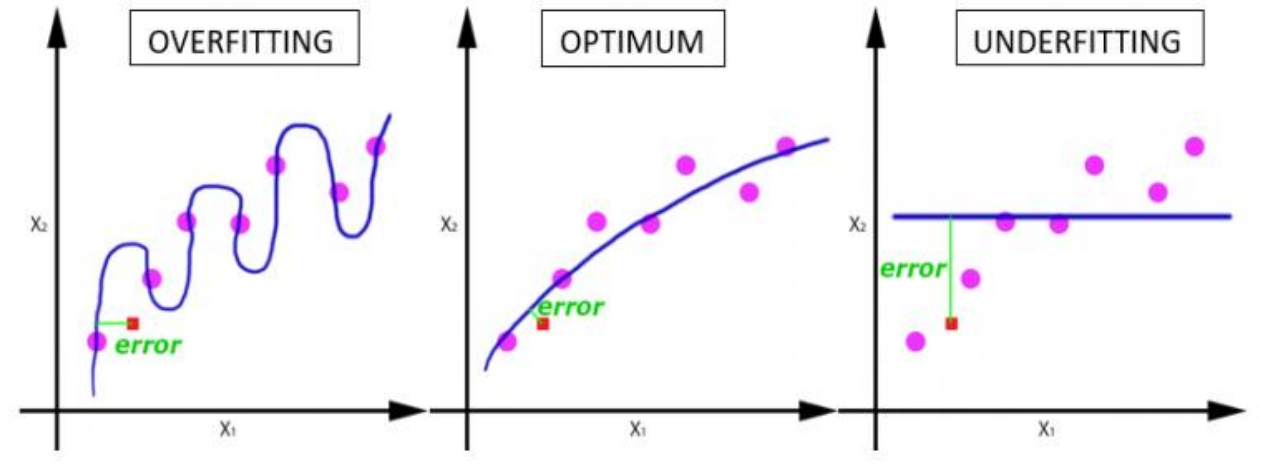
Оцінка точності побудованої моделі відбувається шляхом класифікації об'єктів тестової множини та порівняння отриманих результатів із відомими значеннями тестового набору. За *рівень точності* приймається відсоток правильно класифікованих об'єктів у тестовій множині. Якщо точність моделі є прийнятною, її використовують для класифікації нових об'єктів.

До визначення моделі, яка будується на основі навчаючої множини, необхідно підходити збалансовано. Побудована модель не повинна бути ані перенавченою, ані недонавченою (рис. 7.1).

Проблема перенавчання (англ. *Overfitting*) виникає у випадку, коли класифікатор ідеально описує дані навчаючої множини, інтерпретуючи аномальні значення та помилки як частину внутрішньої структури даних (рис. 7.1, а). Така модель буде некоректно працювати надалі з іншими даними з тестової множини.

Проблема недонавчання (англ. *Underfitting*) – побудована модель на навчаючій множині дає велику кількість помилок, класифікатор не може правильно розділити множину на різні групи споріднених об'єктів – класи (рис. 7.1 в).

Більш зручним для подальшої класифікації є збалансований підхід (рис. 7.1, б). Для різних алгоритмів класифікації розроблені методи уникнення вказаних проблем.



а) перенавчання б) збалансований підхід в) недонавчання

– *опис даних навчаючої множини класифікатором*

– *об'єкти навчаючої множини*

Рис. 7.1. Проблеми побудови моделі класифікатора у двовимірному просторі ознак

7.1.2. Оцінка ефективності класифікатора. Матриця помилок

Для визначення ефективності та продуктивності моделі класифікатора, побудованої за обраним методом, використовують таблицю спряженості, яку прийнято називати матрицею помилок.

Матриця помилок (плутанини) (англ. *Confusion/Error matrix*) у випадку бінарної класифікації є таблицею із 4-ма різними комбінаціями фактичних значень залежної змінної – класу, та визначених за допомогою побудованого класифікатора значень цієї змінної (табл. 7.1).

Враховуючи, що фактичні значення залежної змінної можуть бути істинними та хибними, а передбачені класифікатором – позитивними та негативними, отримаємо наступне тлумачення вмісту комірок матриці помилок.

TP (англ. *True Positive*) – ***істинно позитивні***: кількість об'єктів, які належать класу c_r, правильно класифікованих – віднесених класифікатором до цього класу.

TN (англ. *True Negative*) – ***істинно негативні***: кількість об'єктів, які не належать класу c_r, правильно класифікованих – не віднесених класифікатором до цього класу.

FP (англ. *False Positive*) – ***хибно позитивні***: кількість об'єктів, які не належать до класу c_r, неправильно класифікованих – віднесених класифікатором до цього класу (помилка 1-го роду).

FN (англ. *False Negative*) – ***хибно негативні***: кількість об'єктів, які належать класу c_r, не правильно класифікованих – не віднесених класифікатором до цього класу (помилка 2-го роду).

Таблиця 7.1

Матриця помилок для бінарної класифікації

Належність до класу c_r		Визначені значення (predicted values)	
		«так»	«ні»
Фактичні значення (actual values)	«так»	TP	FN
	«ні»	FP	TN

Матрицю помилок зручно застосовувати для задач, де здійснюється класифікація за двома класами. Однак її можна використовувати і для задач із більшою кількістю класів, додаючи більше рядків та стовпців.

Матриця помилок дає можливість розрахувати багато показників, які дозволяють оцінити ефективність моделі класифікатора. Розглянемо це на практичному прикладі.

Приклад 3. Наведемо приклад побудови матриці помилок та оцінки ефективності класифікатора за результатами бінарної класифікації електронної пошти з метою виявлення спаму. Класифікатор перевіряє кожне вхідне повідомлення й відносить його до одного із двох класів: «spam» (небажана електронна пошта) та «mail» (лист). Є тестовий набір даних із 10 повідомлень, для якого відомі фактичні та визначені класифікатором значення класу (табл. 7.2).

Таблиця 7.2

Результат роботи класифікатора фільтрації електронної пошти

№ з/п	Фактичні значення (actual values)	Визначені значення (predicted values)
1	mail	spam
2	mail	mail
3	spam	spam
4	mail	mail
5	spam	mail
6	spam	spam
7	spam	spam
8	mail	mail
9	mail	spam
10	spam	spam

1. Будуємо матрицю помилок, яка буде мати вигляд, представлений у таблиці 7.3. Загальна кількість фактичних листів – значень «mail» у тестовому наборі даних є сумою значень у рядку «так» (2+3). Загальна кількість спаму – значень «spam» у наборі даних є сумою значень у рядку «ні» (4+1).

Таблиця 7.3

Матриця помилок класифікатора фільтрації електронної пошти

Належність до класу «mail» (n = 10)		Визначені значення (predicted values)	
		«так»	«ні»
Фактичні значення (actual values)	«так»	TP = 3	FN = 2
	«ні»	FP = 1	TN = 4

2. Кількість правильно класифікованих листів є сумою значень, які знаходиться на діагональній лінії від лівого верхнього до правого нижнього краю таблиці (4+3). Кількість неправильно класифікованих листів є сумою значень, які знаходиться на діагональній лінії від лівого нижнього до правого верхнього краю таблиці (2+1).

3. Матриця помилок дає можливість побачити, що прогнозуючи «mail» як «spam», класифікатор дав більше помилок, ніж при прогнозуванні «spam» як «mail».

4. Розраховуємо показники, які характеризують ефективність класифікатора.

4.1. ***Точність класифікації*** (англ. *Accuracy*): $$ACC = \frac{TP + TN}{TP + TN + FP + FN} = \frac{3 + 4}{10} = 0{,}7.$$

Точність становить 0,7 (70%), оскільки класифікатор правильно класифікував 7 із 10 повідомлень.

4.2. ***Частота помилок***: $Error\ Rate = \frac{FP + FN}{TP + TN + FP + FN} = \frac{1+2}{10} = 0{,}3.$

Рівень помилкової класифікації становить 0,3 (30%), оскільки класифікатор неправильно класифікував 3 із 10 повідомлень.

4.3. ***Повнота, частота відклику*** (англ. *Sensitivity, Recall*):

$$SN = True\,Positive\,Rate = \frac{TP}{TP + FN} = \frac{3}{3+2} = 0{,}6.$$

Частота позитивної оцінки становить 0,6 (60%), оскільки класифікатор правильно класифікував 3 із 5 позитивних повідомлень.

4.4. ***Специфічність*** (англ. *Specificity*):

$$SP = True\,Negative\,Rate = \frac{TN}{FP + TN} = \frac{4}{1+4} = 0{,}8.$$

Частота істинних негативних результатів становить 0,8 (80%), оскільки класифікатор правильно класифікував 4 із 5 негативних повідомлень.

4.5. ***Точність позитивних результатів*** (англ. *Precision*):

$$PREC = Positive\ predictive\ value = \frac{TP}{FP + TP} = \frac{3}{1+3} = 0{,}75.$$

Точність позитивних результатів становить 0,75 (75%), оскільки із 4 позитивних повідомлень класифікатор правильно класифікував 3.

4.6. ***Коефіцієнт хибних позитивних результатів***:

$$FPR = False\,Positive\,Rate = \frac{FP}{FP + TN} = \frac{1}{1+4} = 0{,}2.$$

Частота хибних позитивних результатів становить 0,2 (20%), оскільки класифікатор неправильно класифікував 1 із 4 негативних повідомлень.

4.7. ***F-оцінка*** (англ. *F-scores*):

$$F = \frac{2 \cdot PREC \cdot SN}{PREC + SN} = \frac{2 \cdot 0{,}75 \cdot 0{,}6}{0{,}75 + 0{,}6} = \frac{0{,}9}{1.35} = 0{,}67.$$

F-оцінка є мірою точності моделі класифікатора, яка об'єднує частоту позитивного відклику (SN – Recall) та точність позитивних результатів (PREC – Precision). SN та PREC не завжди можуть бути одночасно високими, тоді F-оцінка дозволяє при побудові класифікатора обрати їх оптимальне співвідношення.

7.1.3. Сутність дискримінантного аналізу даних

Дискримінантний аналіз є багатовимірним видом аналізу даних й об'єднує методи, змістом яких є розрізнення (дискримінація) об'єктів набору даних одночасно за певною групою ознак – їх *класифікація*.

Ознаки, які є *незалежними змінними*, називають ***дискримінантними змінними***. Кожна з них повинна вимірюватись у інтервальній шкалі або у шкалі відношень. *Залежна змінна*, за якою здійснюється класифікація, є категоріальною й вимірюється у номінальній або порядковій шкалі.

Під час здійснення дискримінантного аналізу класифікатором є дискримінантна функція, побудова якої дозволяє класифікувати нові об'єкти.

Дискримінантна функція – функція, на основі якої здійснюється класифікація та яка будується за значеннями ознак об'єктів існуючих класів.

Дискримінантна функція має вигляд:

$$f(x) = a_1 x_1 + a_2 x_2 + ... + a_k x_k\,, \qquad (7.1)$$

де x_i – i -та дискримінантна змінна, a_i – i -й коефіцієнт дискримінантної функції,

k – кількість дискримінантних змінних.

В основі знаходження коефіцієнтів дискримінантної функції лежить знаходження умови оптимальності розподілу об'єктів на класи, яке дає значну відмінність між ними: внутрішньогрупова варіація повинна бути мінімальною, а міжгрупова – максимальною.

Варіація – це кількісні зміни ознаки, обумовлені впливом різних факторів. До абсолютних показників варіації відносять різницю між найбільшим та найменшим значенням ознаки, дисперсію, середнє квадратичне відхилення.

Побудова класифікатора при здійсненні дискримінантного аналізу передбачає знаходження коваріаційної матриці.

Коваріаційна матриця – квадратна матриця, складена з попарних коваріацій і дисперсій двох або більше змінних:

$$S = \begin{vmatrix} s_{11} & \dots & s_{1k} \\ \dots & \dots & \dots \\ s_{k1} & \dots & s_{kk} \end{vmatrix}. \tag{7.2}$$

Недіагональні елементи матриці – коваріації тих змінних, на перетині яких вони знаходяться. Діагональні елементи матриці – дисперсії. Коваріаційна матриця є симетричною відносно головної діагоналі й має розмірність $k \times k$, де k – кількість змінних.

Коваріація – міра спільної мінливості (залежності) двох змінних: математичне сподівання добутку їх відхилень від їх особистих математичних сподівань. Для двох змінних, заданих векторами X та Y вона обчислюється за формулою:

$$\operatorname{cov}(X,Y) = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}), \tag{7.3}$$

де n – кількість об’єктів.

Найкращою дискримінантною функцією буде та, що забезпечує максимум відстані за метрикою Махаланобіса між центрами класів.

Центр класу є точкою у багатовимірному просторі ознак, яка містить середнє арифметичне значень ознак об’єктів, що входять до класу.

7.1.4. Етапи алгоритму дискримінантного аналізу даних

Для зручності розглянемо випадок двох дискримінантних змінних та двох класифікаційних груп – класів. Дискримінантне рівняння у цьому випадку матиме вигляд:

$$f(x) = a_1 x_1 + a_2 x_2, \tag{7.4}$$

де x_1 і x_2 – дискримінантні змінні, a_1 і a_2 – коефіцієнти дискримінантної функції.

Основні етапи дискримінантного аналізу будуть наступними.

1. Для кожного класу навчаючої множини даних будуємо матрицю вхідних даних, рядки якої відповідають об’єктам класу, а стовпці – змінним, які відповідають характеристикам об’єктів.

2. Знаходимо вектори середніх значень змінних об’єктів для кожного класу навчаючої множини даних: X_1 та X_2.

3. Знаходимо матриці центрованих значень змінних кожного класу: $\tilde{X}_1$ і $\tilde{X}_2$. ***Центрування матриці*** здійснюється шляхом віднімання від кожного елемента матриці вхідних даних середнього значення відповідної змінної.

4. Розраховуємо об’єднану коваріаційну матрицю досліджуваних ознак за формулою:

$$S = \frac{1}{m_1 + m_2 - 2}(\tilde{X}_1^T \tilde{X}_1 + \tilde{X}_2^T \tilde{X}_2), \tag{7.5}$$

де m_1 та m_2 – кількість об’єктів першого і другого класів навчальної множини даних.

5. Обчислюємо обернену коваріаційну матрицю S^{-1}.

6. Обчислюємо вектор коефіцієнтів дискримінантної функції за формулою:

$$A = S^{-1}(\overline{X}_1 - \overline{X}_2). \tag{7.6}$$

7. Розраховуємо середні значення дискримінантної функції для кожного класу:

$$\overline{f}_1 = \overline{X}_1^T A, \quad \overline{f}_2 = \overline{X}_2^T A.$$

8. Обчислюємо ***константу дискримінації*** C – межу, що розділяє класи, за формулою:

$$C = \frac{\overline{f}_1 + \overline{f}_2}{2}. \tag{7.7}$$

9. Розраховуємо значення дискримінантної функції для нових об'єктів за формулою:

$$f_k = X_k^T A,$$

де X_k – вектор значень змінних нового об'єкту, що підлягає класифікації.

10. Порівнюємо значення f_k із константою дискримінації і відносимо об'єкт залежно від результату до одного з класів. Якщо це значення перевищує константу дискримінації С, то об'єкт відносять до того класу, для якого середнє значення дискримінантної функції більше від С, інакше його відносять до іншого класу, для якого середнє значення дискримінантної функції буде меншим від С.

7.2. ПРОВЕДЕННЯ ДИСКРИМІНАНТНОГО АНАЛІЗУ ДАНИХ ЗАСОБАМИ MS EXCEL ТА MATLAB

Завдання 1. Здійснити класифікацію об'єктів методом дискримінантного аналізу.

Методом дискримінантного аналізу необхідно класифікувати чотири підприємства, які характеризуються двома ознаками (табл. 7.4): x_1 – фондовіддача основних виробничих фондів, x_2 – матеріаломісткість: витрати сировини і матеріалів на одиницю продукції.

Таблиця 7.4

Показники діяльності промислових підприємств

Підприємства	x_1	x_2
1-е	0.78	9.12
2-е	1.44	8.31
3-е	0.62	4.23
4-е	1.17	10.14

Класифікацію необхідно здійснити, спираючись на розбиття множини із 11 підприємств за допомогою кластерного аналізу на 2 класи (табл. 7.5). Перший клас характеризується високим значенням показника фондовіддачі і низькими значеннями матеріаломісткості. До другого класу відносяться інші підприємства.

Таблиця 7.5

Класи промислових підприємств за показниками діяльності

Класи	Номер підприємства	x_1	x_2
I клас	1	1.47	3.94
	2	1.52	5.18
	3	1.27	8.18
	4	1.29	2.17
	5	1.41	4.86
II клас	6	0.53	13.82
	7	0.59	9.08
	8	0.56	9.55
	9	0.79	13.33
	10	0.92	9.84
	11	0.63	12.33

Розв'язок поставленої задачі розіб'ємо на декілька етапів.

7.2.1. Розрахунок середніх значень змінних у класах навчаючої множини даних

1. Здійснюємо розрахунок середніх навчаючої множини даних. У MS Excel створюємо таблиці вхідних даних для розрахунку середніх значень кожної змінної в окремих класах (рис. 7.2). Отримуємо розраховані середні значення змінних:

I клас: $\bar{x}_{11} = 1{,}392,\ \bar{x}_{12} = 4{,}866,$ II клас: $\bar{x}_{21} = 0{,}67,\ \bar{x}_{22} = 11{,}33.$

	A	B	C	D	E
1					
2		Класи підприємств			
3		Класи	Номер підприємства	x_1	x_2
4		I клас	1	1,47	3,94
5			2	1,52	5,18
6			3	1,27	8,18
7			4	1,29	2,17
8			5	1,41	4,86
9			Середні	1,392	4,866
10					
11		Класи	Номер підприємства	x_1	x_2
12		II клас	6	0,53	13,82
13			7	0,59	9,08
14			8	0,56	9,55
15			9	0,79	13,33
16			10	0,92	9,84
17			11	0,63	12,33
18			Середні	0,67	11,33

Рис. 7.2. Розрахунок середніх значень змінних I та II класів підприємств у програмі MS Excel

7.2.2. Побудова матриці центрованих значень

1. Будуємо матриці центрованих значень класу I та класу II в MS Excel. Для побудови матриці центрованих значень необхідно знайдене середнє для кожного стовпця матриці вхідних даних у кожному класі вирахувати зі значення кожного елемента кожного стовпця.

Отримаємо:

$$\tilde{\tilde{X}}_1 = \begin{pmatrix} 0{,}08 & -0{,}93 \\ 0{,}13 & 0{,}31 \\ -0{,}12 & 3{,}31 \\ -0{,}10 & -2{,}70 \\ 0{,}02 & -0{,}01 \end{pmatrix} \qquad \tilde{\tilde{X}}_2 = \begin{pmatrix} -0{,}14 & 2{,}50 \\ -0{,}08 & -2{,}25 \\ -0{,}11 & -1{,}78 \\ 0{,}12 & 2{,}01 \\ 0{,}25 & -1{,}49 \\ -0{,}04 & 1{,}01 \end{pmatrix},$$

де $\tilde{X}_1$ та $\tilde{X}_2$ – матриці центрованих значень класу I та класу II (рис. 7.3).

7.2.3. Налаштування інтеграції MS Excel та MatLab

Наступні розрахунки будемо робити з використанням пакета MATLAB, попередньо налаштувавши інтеграцію Excel і MatLab. Це дасть змогу переносити дані з аркуша MS Excel у MatLab у вигляді матриці й навпаки (переноситися будуть не формули, а тільки значення).

Для налаштування інтеграції MS Excel і MatLab необхідно зробити наступне:

1. У відкритому вікні програми MS Excel відкриємо вкладку *Файл – Параметри – Налаштування*.

2. У нижній правій частині вікна, що відкрилося, в області *Управління* зі списку, що відкривається, обираємо *Налаштування Excel*, натискаємо кнопку *Перейти*.

3. У вікні *Налаштування* натискаємо кнопку *Огляд* – вказуємо шлях до файлу *C:\Program Files\MATLAB\R2015b\toolbox\exlink* і натискаємо кнопку *OK*.

4. У вікні *Налаштування* з'явиться інструмент *LINK EX*, а у вікні MS Excel у вкладці *Головна/Home* з'явиться кнопка *MATLAB* зі списком, що розкривається, який містить пункти, що реалізують основні дії, необхідні для взаємозв'язку MS Excel та MatLab: обмін матричними даними та виконання команд MatLab із середовища MS Excel (рис. 7.4).

5. Інтегруємо вихідні дані з MS Excel в MatLab. Для інтеграції центрованої матриці даних класу I необхідно виділити діапазон комірок з цими даними та у списку, що розкривається MatLab (рис. 7.4) обрати *Send data to MATLAB*. З'явиться вікно з попередженням, що MatLab не запущений. Натисніть *OK*, дочекайтеся відкриття MatLab.

6. З’явиться діалогове вікно Excel з полем введення, призначеним для визначення імені змінної робочого середовища MatLab, у яку варто експортувати дані з виділеного діапазону Excel. Введіть X1 та закрийте вікно за допомогою кнопки *OK*.

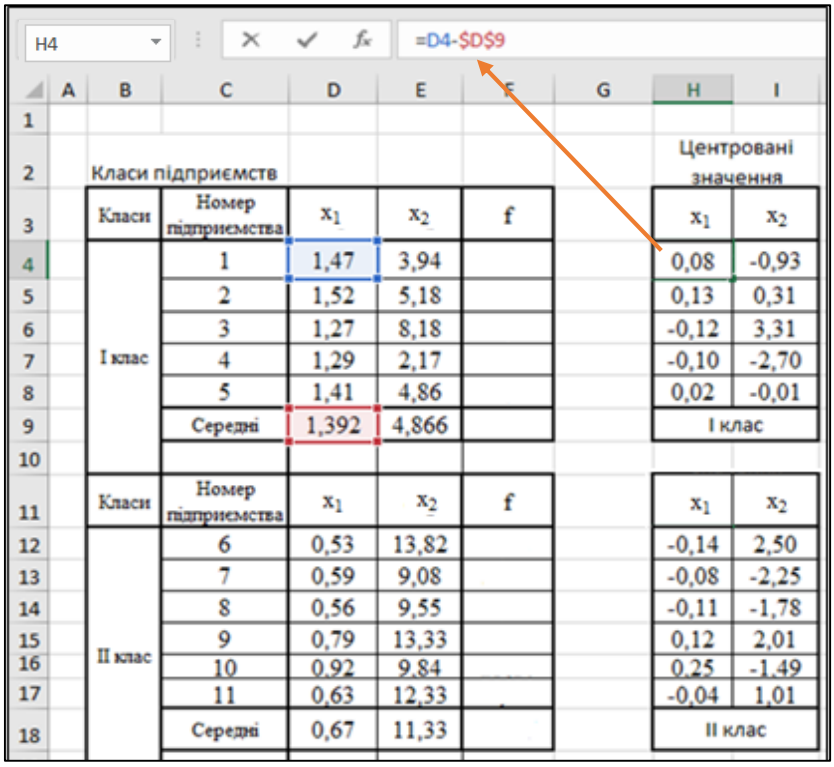


Рис. 7.3. Побудова матриці центрованих значень у програмі MS Excel

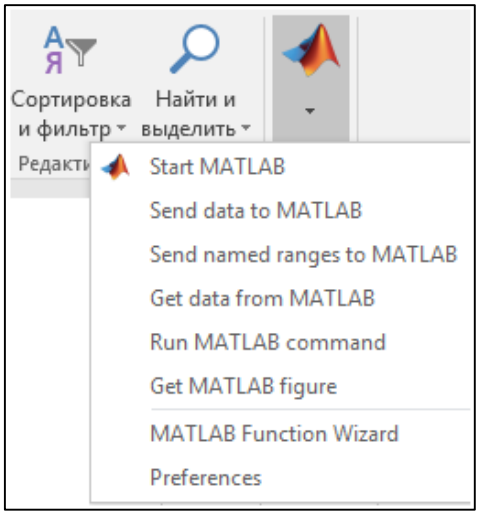


Рис. 7.4. Кнопка MATLAB на вкладці MS Excel Головна/Home з інструментами для взаємозв’язку з MatLab

7. Перейдіть до командного вікна MatLab *Command Window* і переконайтеся, що в робочому середовищі *Workspace* створилася змінна X1, яка містить масив розміром 2×5.

7. Аналогічно імпортуємо з MS Excel в MatLab:

a) центровану матрицю даних класу II, створивши для цього змінну X2;
b) матрицю даних класу I (не центровану), створивши для цього змінну Xn1;
c) матрицю даних класу II (не центровану), створивши для цього змінну Xn2;
d) вектор середніх значень змінних класу I, створивши для цього змінну Sr1;
e) вектор середніх значень змінних класу II, створивши для цього змінну Sr2.

7.2.4. Побудова діаграми розсіювання об’єктів у просторі ознак

1. Для візуального відображення точок об’єктів обох класів у просторі ознак вводимо команди у командному вікні MatLab *Command Window* (рис. 7.5).

2. Отримана діаграма розсіювання точок об’єктів у просторі ознак показує, що класифікація вхідних даних проведена правильно (рис. 7.6).

```
Command Window
>> X = [Xn1; Xn2];
>> % визначення кількості рядків матриць Xn1 та Xn2
>> g1=size(Xn1); n1=g1(:,1);
>> g2=size(Xn2); n2=g2(:,1);
>> % формування вектору належності до класу T
>> for i=1:n1 T1(i) = 1; end
>> for i=1:n2 T2(i) = 2; end
>> T1 = T1'; T2 = T2';  T = [T1; T2];
>> % візуалізація об'єктів у просторі ознак
>> scatter(X(:,1),X(:,2),100, T, 'filled')
```

Рис. 7.5. Команди для візуального відображення імпортованих об’єктів

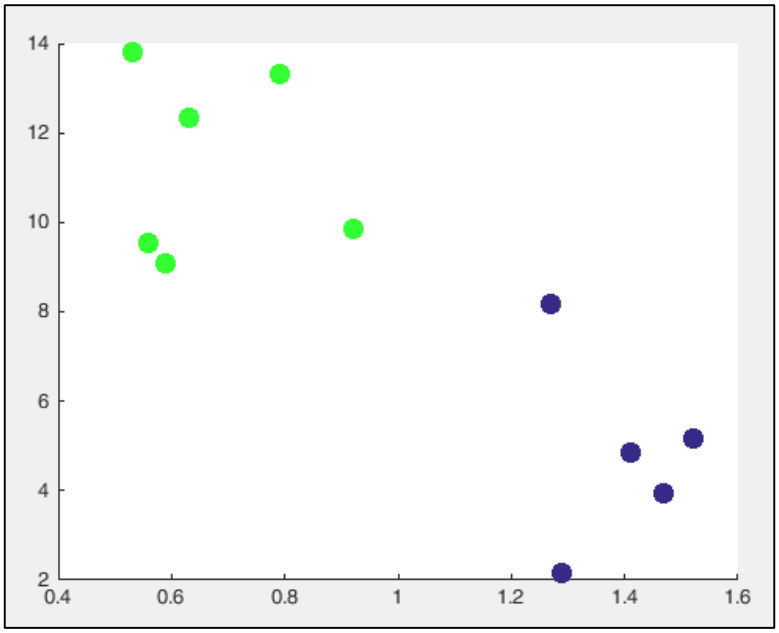


Рис. 7.6. Діаграма розсіювання точок об’єктів у просторі ознак

7.2.5. Знаходження об’єднаної коваріаційної матриці

Об’єднану коваріаційну матрицю розраховуємо за формулою 7.5: $S = \frac{1}{m_1 + m_2 - 2}(\tilde{X}_1^T \tilde{X}_1 + \tilde{X}_2^T \tilde{X}_2)$,

де $\tilde{X}_1^T \tilde{X}_1$ та $\tilde{X}_2^T \tilde{X}_2$ – добутки прямої та транспонованої матриць центрованих значень класу I і класу II,

m_1 – кількість об’єктів класу I; m_2 – кількість об’єктів класу II.

1. У командному вікні MatLab *Command Window* створюємо змінні *m1* та *m2* і привласнюємо їм значення кількості об’єктів класу I (*n1*= 5) і кількості об’єктів класу II (*n2*= 6) (*n1* та *n2* були визначені у попередньому завданні).

2. Розраховуємо в MatLab за вказаною вище формулою об’єднану коваріаційну матрицю *S* (рис. 7.7).

3. Після налаштування інтеграції в MS Excel тепер можна здійснити виклик команди MatLab. Скористаємося цим для розрахунку оберненої коваріаційної матриці S^{-1}. Необхідно перейти до вікна MS Excel та імпортувати з MatLab у Excel розраховані значення об’єднаної коваріаційної матриці *S*.

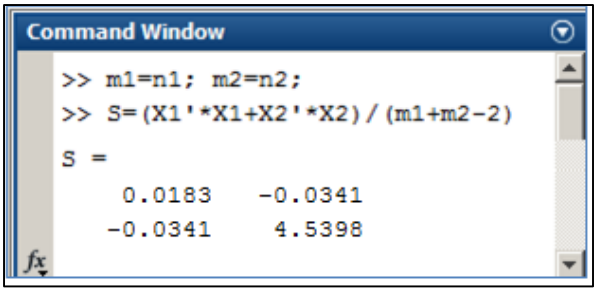


Рис. 7.7. Обчислення об'єднаної коваріаційної матриці в MatLab

Для цього в MS Excel необхідно зробити активною комірку *L19*, натиснути кнопку *MATLAB*, обрати *Get data from MATLAB* і ввести у вікні вводу ім'я масиву, що імпортується з MatLab: *S*. У діапазоні комірок *L19:M20* з'явиться імпортована коваріаційна матриця (рис. 7.8).

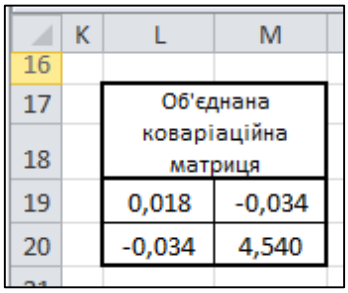


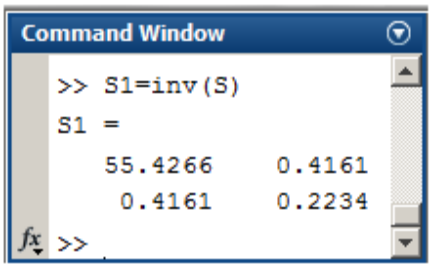
Рис. 7.8. Імпортована з MatLab у MS Excel коваріаційна матриця

4. Для знаходження матриці, оберненої до *S* у MS Excel використовують формулу *МОБР()/MINVERSE()*. Для правильної роботи формулу потрібно ввести як формулу масиву. З цією метою у комірку *P19* необхідно ввести формулу для розрахунку оберненої матриці: *=МОБР(L19:M20) / = MINVERSE(L19:M20).*

5. Потім потрібно виділити діапазон комірок *P19:Q20* та натиснути спочатку клавішу *F2*, а потім комбінацію клавіш *Ctrl+Shift+Enter*. У діапазоні комірок *P19:Q20* з'являться елементи оберненої коваріаційної матриці S^{-1} (рис. 7.9, *а*).

6. У MatLab обернену матрицю можна розрахувати за допомогою функції *inv()* (рис. 7.9, *б*).

а) розрахунок у MS Excel



б) розрахунок у MatLab

Рис. 7.9. Знаходження оберненої коваріаційної матриці

7.2.6. Розрахунок коефіцієнтів дискримінантної функції

Дискримінантна функція має вигляд: $f(x) = a_1x_1 + a_2x_2$.

Її коефіцієнти a_1, a_2 визначаються за формулою 7.6: $A = S^{-1}(\overline{X}_1 - \overline{X}_2)$,

де $\overline{X}_1$, $\overline{X}_2$ – вектори середніх у класах I та II, A – вектор коефіцієнтів, S – об'єднана коваріаційна матриця.

1. У MatLab знаходимо вектор коефіцієнтів дискримінантної функції за наведеною вище формулою (рис. 7.10).

2. Таким чином: $a_1 = 37{,}33$, $a_2 = -1{,}14$. Тоді дискримінантна функція буде мати вигляд:

$$f(x) = 37{,}33x_1 - 1{,}14x_2$$

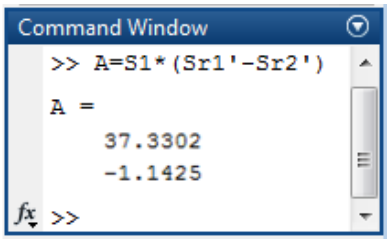


Рис. 7.10. Знаходження вектору коефіцієнтів дискримінантної функції в MatLab

Із аналізу коефіцієнтів цієї функції можна зробити висновок, що перший чинник має більшу вагу, ніж другий.

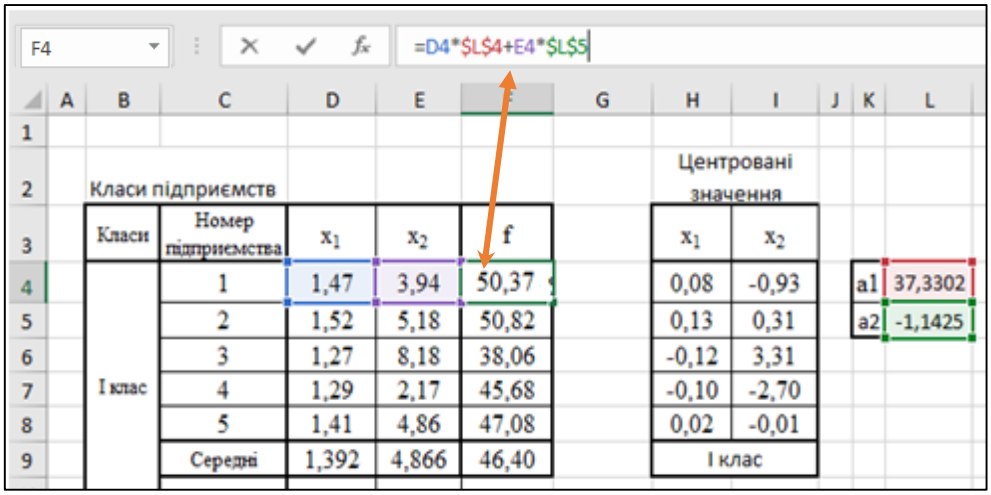
3. У MS Excel здійснимо імпорт коефіцієнтів дискримінантної функції з MatLab та розрахуємо значення дискримінантної функції для кожного підприємства I класу за формулами:

$$f_{11} = 37{,}33 \cdot 1{,}47 - 1{,}14 \cdot 3{,}94 = 50{,}37 \qquad f_{12} = 37{,}33 \cdot 1{,}52 - 1{,}14 \cdot 5{,}18 = 50{,}82$$

$$f_{13} = 37{,}33 \cdot 1{,}27 - 1{,}14 \cdot 8{,}18 = 38{,}06 \qquad f_{14} = 37{,}33 \cdot 1{,}29 - 1{,}14 \cdot 2{,}17 = 45{,}68$$

$$f_{15} = 37{,}33 \cdot 1{,}41 - 1{,}14 \cdot 4{,}86 = 47{,}08$$

4. Отримаємо таблицю в MS Excel із розрахованими значеннями дискримінантної функції f для I класу (рис. 7.11).



Класи	Номер підприємства	x_1	x_2	f	x_1	x_2
I клас	1	1,47	3,94	50,37	0,08	-0,93
	2	1,52	5,18	50,82	0,13	0,31
	3	1,27	8,18	38,06	-0,12	3,31
	4	1,29	2,17	45,68	-0,10	-2,70
	5	1,41	4,86	47,08	0,02	-0,01
	Середні	1,392	4,866	46,40	I клас	

Рис. 7.11. Розрахунок значень дискримінантної функції в MS Excel для підприємств I класу

5. Аналогічно розраховуємо значення дискримінантної функції для кожного підприємства II класу за формулами:

$$f_{21} = 37{,}33 \cdot 0{,}53 - 1{,}14 \cdot 13{,}82 = 4{,}00 \qquad f_{22} = 37{,}33 \cdot 0{,}59 - 1{,}14 \cdot 9{,}08 = 11{,}65$$

$$f_{23} = 37{,}33 \cdot 0{,}56 - 1{,}14 \cdot 9{,}55 = 9{,}99 \qquad f_{24} = 37{,}33 \cdot 0{,}79 - 1{,}14 \cdot 13{,}33 = 14{,}26$$

$$f_{25} = 37{,}33 \cdot 0{,}92 - 1{,}14 \cdot 9{,}84 = 23{,}10 \qquad f_{26} = 37{,}33 \cdot 0{,}63 - 1{,}14 \cdot 12{,}33 = 9{,}43$$

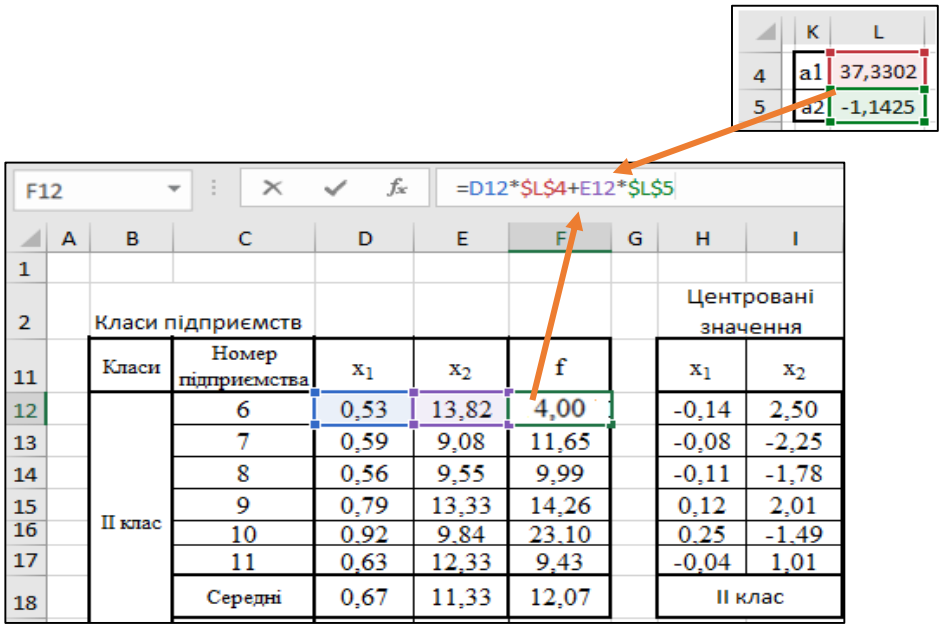
6. Отримаємо таблицю в MS Excel із розрахованими значеннями дискримінантної функції f для I класу (рис. 7.12).

7.2.7. Визначення константи детермінації

1. Розрахуємо в MS Excel середні значення дискримінантної функції для кожного класу за формулами:

$$\overline{f}_1 = 37{,}33 \cdot 1{,}39 - 1{,}14 \cdot 4{,}87 = 46{,}40, \qquad \overline{f}_2 = 37{,}33 \cdot 0{,}67 - 1{,}14 \cdot 11{,}33 = 12{,}07$$

2. За формулою 7.7 розрахуємо константу дискримінації. На робочому аркуші значення $\overline{f}_1$ міститься у комірці *F9*, значення $\overline{f}_2$ – у комірці *F18*. Маємо: $C = \dfrac{\overline{f}_1 + \overline{f}_2}{2} = 29{,}2384$.



	K	L
4	a1	37,3302
5	a2	-1,1425

F12 | =D12*$L$4+E12*$L$5

	A	B	C	D	E	F	G	H	I
1									
2		Класи підприємств						Центровані значення	
11		Класи	Номер підприємства	x_1	x_2	f		x_1	x_2
12			6	0,53	13,82	4,00		-0,14	2,50
13			7	0,59	9,08	11,65		-0,08	-2,25
14			8	0,56	9,55	9,99		-0,11	-1,78
15			9	0,79	13,33	14,26		0,12	2,01
16		II клас	10	0,92	9,84	23,10		0,25	-1,49
17			11	0,63	12,33	9,43		-0,04	1,01
18			Середні	0,67	11,33	12,07		II клас	

Рис. 7.12. Розрахунок значень дискримінантної функції в MS Excel для підприємств II класу

7.2.8. Здійснення класифікації нових об'єктів

1. За допомогою побудованого класифікатора проведемо класифікацію чотирьох підприємств. Для цього розрахуємо для них значення дискримінантних функцій:

$$f_1 = 37{,}33 \cdot 0{,}78 - 1{,}14 \cdot 9{,}12 = 18{,}70 \qquad f_2 = 37{,}33 \cdot 1{,}44 - 1{,}14 \cdot 8{,}31 = 44{,}26$$

$$f_3 = 37{,}33 \cdot 0{,}62 - 1{,}14 \cdot 4{,}23 = 18{,}31 \qquad f_4 = 37{,}33 \cdot 1{,}17 - 1{,}14 \cdot 10{,}143 = 32{,}09$$

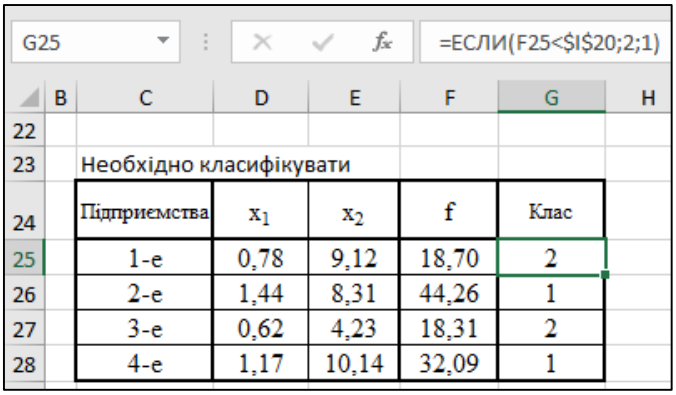
2. Визначаємо належність підприємств до класів (рис. 7.13). Оскільки значення дискримінантної функції 1-го та 3-го підприємств менше за C, то їх необхідно віднести до класу II зі значеннями дискримінантної функції, меншими за C.

3. Для 2-го та 4-го підприємств значення дискримінантної функції більші за C, тому їх необхідно віднести до класу I зі значеннями дискримінантної функції, більшими за C.

7.2.9. Аналіз отриманих результатів

1. Перше з нових підприємств було віднесене до другого класу, а друге – до першого. Цікавість являла класифікація третього та четвертого підприємств. Третє мало низькі значення обох чинників, а четверте – досить високі значення цих чинників.

2. Оскільки вага першого показника у результаті побудови дискримінантної функції виявилася вищою, то третє підприємство було віднесене до другої групи, а четверте – до першої.



G25 | =ЕСЛИ(F25<$I$20;2;1)

	B	C	D	E	F	G	H
22							
23		Необхідно класифікувати					
24		Підприємства	x_1	x_2	f	Клас	
25		1-е	0,78	9,12	18,70	2	
26		2-е	1,44	8,31	44,26	1	
27		3-е	0,62	4,23	18,31	2	
28		4-е	1,17	10,14	32,09	1	

Рис. 7.13. Здійснення класифікації підприємств

7.3. ЗАВДАННЯ ДЛЯ САМОСТІЙНОЇ РОБОТИ

Завдання 2. Побудувати класифікатор для здійснення класифікації країн методом дискримінінтного аналізу на основі набору даних, отриманого відповідно до варіанта.

1. Об'єкти набору даних – країни та їх характеристики (показники) за номерами для кожного варіанта представлені у таблиці 7.6. Країни для класифікації та значення їх показників за номерами необхідно брати з таблиці 7.7. Сутність соціально-економічних показників країн за номерами розкрито у таблиці 7.8.

2. Для відібраних відповідно до варіанта країн із урахуванням їх соціально-економічних показників необхідно попередньо провести кластеризацію у середовищі MatLab методом k-means, розділивши їх на два кластери.

3. Для побудови класифікатора методом дискримінантного аналізу відібрати 7 країн із 10, віднісши до першого класу країни, які потрапили до першого кластера, а до другого класу – країни, які було віднесено до другого кластера.

4. Використовуючи отримане дискримінантне рівняння, здійснити класифікацію трьох країн із набору даних, які не були задіяні у побудові моделі класифікатора. Визначити точність класифікації із урахуванням попередньо проведеної кластеризації. Надати змістову інтерпретацію результатам.

КОНТРОЛЬНІ ПИТАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ № 7

1. Базові поняття задачі класифікації.
2. У чому полягають проблеми перенавчання та недонавчання?
3. Матриця помилок. Яким чином здійснюється оцінка ефективності та точності класифікатора?
4. Сутність дискримінантного аналізу даних.
5. Як визначають матрицю центрованих значень та константу детермінації.
6. Якою є умова оптимальності поділу об'єктів на класи.
7. Основні етапи алгоритму дискримінантного аналізу.

Таблиця 7.6

Об'єкти для класифікації – країни та їх характеристики (ознаки) за номерами

Варіант	Об'єкти (країни для класифікації)	Ознаки				Варіант	Об'єкти (країни для класифікації)	Ознаки			
1	1–10	1	2	3	4	26	1–10	2	3	4	9
2	11–20	1	2	3	10	27	11–20	3	5	7	9
3	21–30	1	5	11	12	28	21–30	1	2	10	11
4	31–40	1	6	8	9	29	31–40	1	2	11	12
5	41–50	1	6	8	10	30	41–50	6	7	11	12
6	43–52	2	3	4	7	31	43–52	5	8	10	11
7	33–42	6	8	9	11	32	33–42	2	4	6	8
8	23–32	6	8	9	12	33	23–32	2	6	8	10
9	13–22	6	7	8	9	34	13–22	3	4	6	7
10	3–12	5	8	9	10	35	3–12	5	6	8	9
11	1–20 (парні)	2	3	4	9	36	1–20 (парні)	1	2	3	4
12	1–20 (не парні)	3	5	7	9	37	1–20 (не парні)	1	2	3	10
13	21–40 (парні)	1	2	10	11	38	21–40 (парні)	1	5	11	12
14	21–40 (не парні)	1	2	11	12	39	21–40 (не парні)	1	6	8	9
15	33–52 (парні)	6	7	11	12	40	33–52 (парні)	1	7	8	10
16	33–52 (не парні)	5	8	10	11	41	33–52 (не парні)	2	3	4	7
17	5–14	2	4	6	8	42	5–14	6	8	9	11
18	15–24	2	6	8	10	43	15–24	6	8	9	12
19	25–34	3	4	6	7	44	25–34	6	7	8	9

Варіант	Об’єкти (країни для класифікації)	Ознаки				Варіант	Об’єкти (країни для класифікації)	Ознаки			
20	35–44	5	6	8	9	45	35–44	5	8	9	10
21	1–5, 48–52	1	2	3	4	46	1–5, 48–52	5	8	10	11
22	6–10, 43–47	1	2	3	10	47	6–10, 43–47	2	4	6	8
23	11–15, 38–42	1	5	11	12	48	11–15, 38–42	2	6	8	10
24	16–20, 33–37	1	6	8	9	49	16–20, 33–37	3	4	6	7
25	21–25, 28–32	1	7	8	10	50	21–25, 28–32	5	6	8	9

Таблиця 7.7

Значення соціально-економічних показників країн світу

№ з/п	Країна	Показники (ознаки)											
		1	2	3	4	5	6	7	8	9	10	11	12
1	Австралія	17800	15	8	7,3	1,9	16848	2,3	85	100	1,38	74	80
2	Австрія	8000	12	11	6,7	1,5	18396	94	58	99	0,2	73	79
3	Аргентина	33900	20	9	25,6	2,8	3408	12	86	95	1,3	68	75
4	Бангладеш	125000	35	11	106	4,7	202	800	16	35	2,4	53	53
5	Бельгія	10100	12	11	7,2	1,7	17912	329	96	99	0,2	73	79
6	Білорусь	10300	13	11	19	1,88	6500	50	65	99	0,32	66	76
7	Бразилія	156600	21	9	66	2,7	2354	18	75	81	1,28	57	67
8	Буркіна-Фасо	10000	47	18	118	6,94	357	36	15	18	2,81	47	50
9	Великобританія	58400	13	11	7,2	1,83	15974	237	89	99	0,2	74	80
10	В’єтнам	73100	27	8	46	3,33	230	218	20	88	1,78	63	68
11	Гаїті	6500	40	19	109	5,94	383	231	29	53	1,63	43	47
12	Гондурас	5600	35	6	45	4,9	1030	46	44	73	2,73	65	70
13	Гонконг	5800	13	6	5,8	1,4	14641	5494	94	77	0,09	75	80
14	Ефіопія	55200	45	14	110	6,81	122	47	12	24	3,1	51	54
15	Єгипет	60000	29	9	76,4	3,77	748	57	44	48	1,95	60	63
16	Замбія	9100	46	18	85	6,68	573	11	42	73	2,8	44	45
17	Індія	911600	29	10	79	4,48	275	283	26	52	1,9	58	59
18	Ірландія	3600	14	9	7,4	1,99	12170	51	57	98	0,3	73	78
19	Іспанія	39200	11	9	6,9	1,4	13047	77	78	95	0,25	74	81
20	Італія	58100	11	10	7,6	1,3	17500	188	69	97	0,21	74	81
21	Канада	29100	14	8	6,8	1,8	19904	2,8	77	97	0,7	74	81
22	Китай	1205200	21	7	52	1,84	337	124	26	78	1,1	67	69
23	Колумбія	35600	24	6	28	2,47	1538	31	70	87	2	69	75
24	Коста-Рика	3300	26	4	11	3,1	2031	64	47	93	2,3	76	79
25	Куба	11100	17	7	10,2	1,9	1382	99	74	94	0,95	74	78
26	Малайзія	19500	29	5	25,6	3,51	2995	58	43	78	2,3	66	72
27	Марокко	26600	29	6	50	3,83	1062	63	46	50	2,12	66	70
28	Мексика	91800	28	5	35	3,2	3604	46	73	87	1,9	69	77
29	Нідерланди	15400	13	9	6,3	1,58	17245	366	89	99	0,58	75	81
30	Німеччина	81200	11	11	6,5	1,47	17539	227	85	99	0,36	73	79
31	Нова Зеландія	3524	16	8	8,9	2,03	14381	13	84	99	0,57	73	80
32	Норвегія	4300	13	10	6,3	2	17755	11	75	99	0,4	74	81
33	ОАЕ	2800	28	3	22	4,5	14193	32	81	68	4,8	70	74
34	ПАР	43900	34	8	47,1	4,37	3128	35	49	76	2,6	62	68
35	Південна Корея	45000	16	6	21,7	1,65	6627	447	72	96	1	68	74
36	Північна Корея	23100	24	6	27,7	2,4	1000	189	60	99	1,83	67	73
37	Польща	36600	14	10	13,8	1,94	4429	123	62	99	0,3	69	77
38	Португалія	10500	12	10	9,2	1,5	9000	108	34	85	0,36	71	78

№ з/п	Країна	Показники (ознаки)											
		1	2	3	4	5	6	7	8	9	10	11	12
39	Росія	149200	13	11	27	1,83	6680	8,8	74	99	0,2	64	74
40	Саудівська Аравія	18000	38	6	52	6,67	6651	7,7	77	62	3,2	66	70
41	Сінгапур	2900	16	6	5,7	1,88	14990	4456	100	88	1,2	73	79
42	США	260800	15	9	8,11	2,06	23474	26	75	97	0,99	73	79
43	Таїланд	59400	19	6	37	2,1	1800	115	22	93	1,4	65	72
44	Туреччина	62200	26	6	49	3,21	3721	79	61	81	2,02	69	73
45	Україна	51800	12	13	20,7	1,82	2340	87	67	98	0,05	65	75
46	Філіппіни	69800	27	7	51	3,35	867	221	43	90	1,92	63	68
47	Фінляндія	5100	13	10	5,3	1,8	15877	39	60	100	0,3	72	80
48	Франція	58000	13	9	6,7	1,8	18944	105	73	99	0,47	74	82
49	Чилі	14000	23	6	14,6	2,5	2591	18	85	93	1,7	71	78
50	Швейцарія	7000	12	9	6,2	1,6	22384	170	62	99	0,7	75	82
51	Швеція	8800	14	11	5,7	2,1	16900	19	84	99	0,52	75	81
52	Японія	125500	11	7	4,4	1,55	19860	330	77	99	0,3	76	82

Таблиця 7.8

Соціально-економічні показники 52 країн світу

№	Розкриття змісту показника
1	Чисельність населення (тис. ос.)
2	Народжуваність (на 1000 ос.)
3	Смертність (на 1000 ос.)
4	Смертність серед малюків (на 1000 ос.)
5	Середнє число дітей у родині
6	ВВП на душу населення (у дол. США за купівельною спроможністю валют)
7	Густина населення (кількість ос. на кв. км)
8	Відсоток міського населення
9	Відсоток грамотних
10	Приріст населення (% на рік)
11	Тривалість життя чоловіків (у роках)
12	Тривалість життя жінок (у роках)

8. ОСНОВНІ МЕТОДИ РОЗВ'ЯЗАННЯ ЗАДАЧІ КЛАСИФІКАЦІЇ

Лабораторна робота № 8

Мета: закріплення знань про сутність та основні методи розв'язання задачі класифікації: One Rule, Naive Bayes, k-найближчих сусідів, опорних векторів, дерева рішень. Набуття навичок проведення класифікації з допомогою вказаних методів у програмі MatLab.

Теоретичні знання: постановка задачі класифікації. Основні групи методів для розв'язання задачі класифікації. Правила класифікації. Метод One Rule. Метод Naive Bayes. Метод k-найближчих сусідів. Метод опорних векторів. Дерева рішень.

8.1. СУЧАСНІ ПІДХОДИ ДО РОЗВ'ЯЗАННЯ ЗАДАЧІ КЛАСИФІКАЦІЇ

8.1.1. Постановка задачі класифікації

Здійснимо постановку задачі класифікації, описавши її наступним чином.

Нехай є n об'єктів навчаючої множини даних: $I = \{i_1, i_2, \dots, i_n\}$.

Кожен об'єкт i_j множини I характеризується набором m незалежних змінних $\{x_{j1}, x_{j2}, \dots x_{jm}\}$ та однією залежною змінною y_j, яка визначає належність об'єкта до якогось класу.

Множина значень залежної змінної є скінченною й містить сукупність класів, до яких належать об'єкти навчаючої множини даних: $C = \{c_1, c_2, \dots, c_l\}$, де l – кількість класів.

Необхідно визначити клас нового об'єкту $i_h = \{x_{h1}, x_{h2}, \dots x_{hm}\}$, для якого значення залежної змінної – класу y_h, є невідомим.

Різні методи розв'язання задачі класифікації використовують різні способи визначення значень залежної змінної нового об'єкта – класу, до якого він належить.

Умовно можна виділити такі ***види методів*** класифікації.

1. ***Статистичні методи:*** дискримінантний аналіз, One Rule (1R), Naive Bayes, логістична регресія.
2. ***Методи машинного навчання:*** k-найближчих сусідів, опорних векторів, дерева рішень, штучні нейронні мережі.

Визначившись із методом, який буде застосовано для розв'язання задачі класифікації, аналітик здійснює побудову класифікатора.

8.1.2. Правила класифікації

Класифікація передбачає пошук правил, які розбивають набір даних на непересічні групи споріднених об'єктів – класи.

Правила класифікації (*класифікаційні правила*, анг. *classification rules*) є правилами, які складаються з двох частин – *умови* та *висновку*:

якщо (*умова*) ***тоді*** (*висновок*).

Умовою є перевірка значень однієї або декількох незалежних змінних. Перевірка значень декількох незалежних змінних реалізується шляхом їх об'єднання за допомогою логічних операцій «***і***», «***або***» і «***не***».

Висновком є значення залежної змінної – клас, який спостерігається при визначених в умові значеннях незалежних змінних.

Наприклад, при ***визначенні*** кредитоспроможності клієнта банку класифікаційні правила можуть мати такий вигляд:

якщо («*вік>35 років*» ***і*** «*дохід>10 000 гр*») ***тоді*** («*надати кредит*»),

якщо («*дохід<6 000 грн*» ***і*** «*вік<22 років*» ***або*** «*вісутнє власне житло*») ***тоді*** («*відмовити у кредиті*»).

Основною перевагою правил класифікації є легкість їх сприйняття, відносна незалежність, запис природною мовою та легкість додавання нових правил.

8.1.3. Метод 1R (One Rule)

Метод 1R (англ. *1-rule, One Rule, OneR*) є одним із найпростіших методів формування єлементарних правил для класифікації об'єктів на основі значень однієї незалежної змінної.

Метод є ефективним, якщо об'єкти класифікують по одному атрибуту, який приймає дискретні (категоріальні) значення.

Основні етапи методу 1R є наступними.

1. Якщо значення незалежних змінних навчаючої множини є неперервними, їх дискретизують.
2. Для будь-якого можливого значення кожної незалежної змінної формують правило, що класифікує об'єкти. При цьому у висновку правила вказується те значення залежної змінної – клас, яке найбільш часто зустрічається у об'єктів навчаючої множини з обраним значенням незалежної змінної.
3. Для кожного сформованого правила розраховують ***помилку правила*** – кількість об'єктів, які мають дане значення незалежної змінної, але не відносяться до обраного класу.
4. Для кожної незалежної змінної отримують набір правил для кожного її значення. Оцінивши ступінь помилок кожного набору, обирають ту незалежну змінну, правила для якої побудовано з найменшою помилкою. Саме цю змінну використовують при класифікації нових об'єктів.

8.1.4. Метод Naive Bayes

Наївний байєсівський класифікатор є одним із популярних методів розв'язання задачі класифікації, який дає можливість визначити ймовірність належності до певного класу на основі попереднього знання умов, при яких об'єкти навчаючої множини належать до певних класів.

Термін «*naive*» (наївний) у назві методу походить від припущення, що усі змінні є незалежними. Однак навіть коли припущення про незалежність змінних не виконується, класифікатор працює досить добре на великих наборах даних. Передбачається також, що кожна незалежна змінна й залежна змінна мають нормальний розподіл. Робота методу Naive Bayes базується на теоремі Байєса для визначення умовної ймовірності події.

Апріорна ймовірність (англ. *Prior probability*) – це безумовна ймовірність події A за відсутності будь-якої іншої інформації, пов'язаної з цією подією.

Апостеріорна ймовірність, умовна ймовірність (англ. *Posterior probability*) – це ймовірність події A за умови, що подія B уже відбулася.

У методі Naive Bayes:

1) апріорна ймовірність певного значення *k*-ї незалежної змінної об'єкта i_j розраховується за формулою:

$$P(x_{jk}) = \frac{n_{jk}}{n}; \tag{8.1}$$

2) апріорна ймовірність належності об'єкта до певного класу c_r розраховується без урахування значень ознак об'єкта за формулою:

$$P(c_r) = \frac{n_r}{n}; \tag{8.2}$$

3) апостеріорна ймовірність певного значення *k*-ї незалежної змінної об'єкта i_j за умови належності цього об'єкта до певного класу c_r розраховується за формулою:

$$P(x_{jk} \mid y_j = c_r) = \frac{n_{jk}^r}{n_{jk}}. \tag{8.3}$$

У наведених вище формулах:

n_{jk} – кількість об'єктів навчаючої множини, які мають значення *k*-ї незалежної змінної, рівне x_{jk},

n_r – кількість об'єктів навчаючої множини, які відносяться до класу c_r,

n_{jk}^r – кількість об'єктів навчаючої множини, які мають значення *k*-ї незалежної змінної, рівне x_{jk} і відносяться до класу c_r,

n – загальна кількість об'єктів навчаючої множини.

Знання перерахованих вище ймовірностей є необхідними для обчислення апостеріорних імовірностей у класифікаторі Байєса, робота якого базується на принципі максимуму апостеріорної ймовірності.

Основна ідея методу Naive Bayes полягає у розрахунку умовної ймовірності належності об'єкта до одного з класів із множини значень залежної змінної *C* при рівності його незалежних змінних певним значенням.

Зробимо необхідні для побудови класифікатора позначення:

1) E_j – подія, що відповідає рівності кожної з незалежних змінних, які характеризують об'єкт i_j, певним значенням $\{x_{j1}, x_{j2}, \ldots x_{jm}\}$;

2) $y_j = c_r$ – подія, що відповідає рівності залежної змінної y_j об'єкта i_j значенню c_r із множини значень залежної змінної C – класу c_r;

3) $P(y_j = c_r)$ – апріорна ймовірність того, що об'єкт i_j навчаючої множини I відноситься до класу c_r;

4) $P(E_j|y_j = c_r)$ – апостеріорна, умовна ймовірність: ймовірність події E_j за умови, що має місце подія $y_j = c_r$ й оскільки змінні є незалежними одна від одної, то ця ймовірність буде рівна добутку апостеріорних ймовірностей значень кожної незалежної змінної, які відповідають події E_j за умови, що має місце подія $y_j = c_r$:

$$P(E_j \mid y_j = c_r) = P(x_{1j} \mid y_j = c_r) \cdot P(x_{2j} \mid y_j = c_r) \cdot \ldots \cdot P(x_{mj} \mid y_j = c_r);$$

5) $P(E_j)$ – ймовірність події E_j, рівна:

$$P(E_j) = \sum_{r=1}^{l} P(E_j \mid y = c_r) \cdot P(y = c_r), \tag{8.4}$$

де l – кількість можливих значень залежної змінної – класів, а $P(E_j) \neq 0$.

Тоді апостеріорну ймовірність події $y_j = c_r$ за умови, що має місце подія E_j, можна розрахувати за ***формулою Байєса***:

$$P(y_j = c_r \mid E_j) = \frac{p(E_j \mid y_j = c_r) \cdot P(y_j = c_r)}{P(E_j)}. \tag{8.5}$$

Відповідно до Байєсівського класифікатора новий об'єкт i_h буде належати до того класу c_r, для якого його апостеріорна ймовірність $P(y_h = c_r \mid E_h)$, розрахована за формулою Байєса, є максимальною.

Основні етапи методу Naive Bayes є наступними.

1. Формують таблицю, в якій рядкам відповідають об'єкти навчаючої множини, а стовпцям – змінні, що характеризують ці об'єкти.
2. На основі значень незалежних та залежної змінних навчаючої множини розраховують:
   a) апріорні ймовірності кожного значення кожної незалежної змінної в навчаючій множині даних за формулою 8.1;
   b) апріорні ймовірності кожного класу в навчаючій множині даних за формулою 8.2;
   c) апостеріорні ймовірності всіх значень кожної незалежної змінної при кожному фіксованому значенні залежної змінної – класу, за формулою 8.3.
3. Для класифікації нового об'єкта розраховують за формулою 8.5 апостеріорну ймовірність появи кожного класу при значеннях незалежних змінних, які характеризують новий об'єкт. Об'єкт відносять до того класу, ймовірність якого буде найбільшою.

Проблемним місцем у роботі байєсівського класифікатора є випадок, коли значення однієї із незалежних змінних не зустрічається у наборі даних навчаючої множини зі значенням певного класу. Виникає проблема нульової частоти: умовна ймовірність буде рівна нулю і при множенні ймовірностей отримують нуль. При цьому втрачається інформація, класифікатор не зможе правильно визначити клас. Для вирішення цієї проблеми застосовують методику ***згладжування Лапласа*** – до кожної частоти додають одиницю. Це дозволяє уникнути нульової частоти, хоч і не досить суттєво зміщує оцінку наявних ймовірностей у сторону їх зменшення.

Метод Naive Bayes має високу продуктивність, не потребує великої за обсягом навчаючої множини, ідеально підходить для класифікації у режимі реального часу. Він добре працює з категоріальними ознаками, є ефективним для застосування у випадку класифікації текстів, у Web-аналітиці та у рекомендаційних системах.

8.1.5. Дерева рішень

Дерева рішень (англ. *Decision Trees, DTs*) є одним із досить популярних методів класифікації, який дозволяє будувати модель класифікатора у вигляді деревоподібної, ієрархічної структури. Дерево рішень графічно відображає роботу класифікатора у вигляді блок-схеми, яка складається із правил класифікації.

На рисунку 8.1 наведено приклад дерева рішень, завдання якого – визначити, чи можна проводити гру на відкритому стадіоні при заданих погодних умовах. У цьому випадку дерево рішень дає можливість отримати два рішення: «Грати» – якщо погодні умови є сприйнятними та «Не грати» – у протилежному випадку.

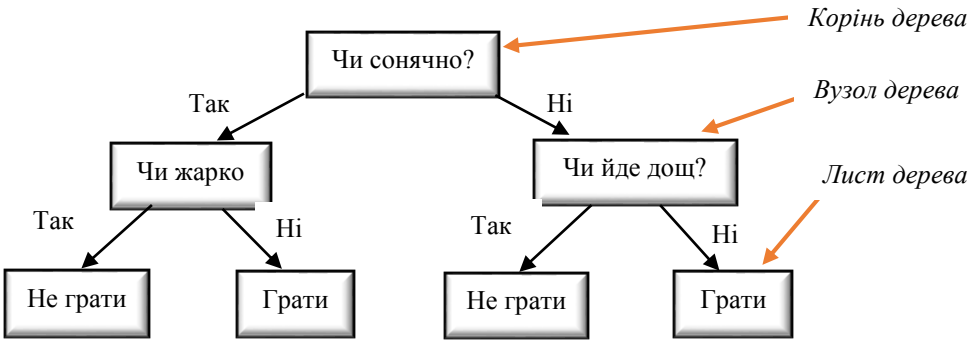


Рис. 8.1. Приклад побудови дерева рішень для визначення можливості проведення гри при заданих погодних умовах

Дерево рішень має корінь, гілки, вузли та листи.

Корінь дерева є вузлом перевірки – умовою перевірки значень такої незалежної змінної, яка є найбільш значимою для класифікації об'єктів навчаючої множини.

Кожен ***вузол дерева*** включає перевірку значень певної незалежної змінної. Іноді у вузлі дерева дві незалежні змінні порівнюються одна з одною або визначається деяка функція від однієї або декількох змінних.

Листи дерев відповідають значенням залежної змінної, тобто ***класам***.

Об'єкт належить певному класу, якщо значення його незалежних змінних задовольняють умови, записані у вузлах дерева на шляху від кореня до листа, що відповідає цьому класу.

Етапи конструювання дерева рішень є наступними.

1. ***Побудова дерева*** (англ. *tree building*): створення дерева зверху вниз від кореня до листів на наборі даних навчаючої множини шляхом послідовного вибору критеріїв розгалуження у кожному вузлі та визначення правила зупинки.

Критерій розгалуження повинен бути такою умовою, яка максимально наближено розбиває навчаючу множину на представників кожного класу. Ключовим елементом побудови дерева рішень є спосіб вибору кожної наступної значимої незалежної змінної при побудові чергового вузла дерева.

Зупинка є моментом у побудові дерева, коли подальше розгалуження слід припинити. У випадку сильно розгалужених дерев формулюють ***правила зупинки*** шляхом обмеження глибини дерева, ранньої зупинки, задання мінімальної кількості об'єктів у кінцевих вузлах.

2. ***Скорочення дерева*** (англ. *tree pruning*): вирішує питання відсікання деяких його гілок у випадку, коли дерево є досить розгалуженим.

Зупинка алгоритму відповідно до прийнятого правила зупинки та використаного способу відсікання гілок дерева з метою його скорочення не повинна зменшувати суттєво точність класифікації.

До основних алгоритмів побудови дерев рішень відносять: алгоритм CART (англ. *classification and regression trees*), алгоритм ID3 (англ. *Iterative Dichotomizer 3*), алгоритми C4.5, C5, алгоритм покриття. Є ***алгоритм випадкового лісу*** (англ. *Random forest*), який формує набір дерев рішень і видає результат на основі аналізу результатів усіх дерев рішень.

Різні алгоритми генерації дерев рішень відрізняються один від одного:

1) способами вибору найбільш доцільних для класифікації незалежних змінних, значення яких будуть перевірятися у вузлах;
2) кількістю розгалужень у вузлах;
3) методами відсікання гілок при скороченні дерев;
4) різними підходами до формування правил зупинки.

Частина алгоритмів може працювати як з дискретними, так і з числовими значеннями змінних. Тому в цілому побудова дерев рішень дозволяє вирішувати задачі *класифікації*, *регресії* та *прогнозування*.

До недоліків алгоритмів побудови дерев рішень відносять схильність до перенавчання, чутливість до шумів, складність пошуку оптимального дерева рішень. До переваг методу дерев рішень відносять інтуїтивно зрозумілу модель класифікатора та швидкий процес її побудови на великих наборах даних навчаючої множини, високу точність класифікації, витяг правил класифікації природною мовою.

8.1.6. Метод k-найближчих сусідів

Метод ***k-найближчих сусідів*** (англ. *k-Nearest Neighbor, KNN*) є методом класифікації, здатним виділити у багатовимірному просторі ознак серед усіх об'єктів навчаючої множини k -об'єктів, найбільш схожих на новий об'єкт. Висновок про належність нового об'єкта до певного класу виноситься на основі класів, до яких належать k -найближчих до цього об'єкта сусідів.

Основні етапи методу *k-найближчих сусідів* є наступними.

1. Задається число k – кількість найближчих сусідів.

2. Перевіряють, чи належать значення незалежних змінних до одного числового діапазону й у разі необхідності – здійснюють нормалізацію (чи стандартизацію) їх значень для отримання однакового внеску у розрахунок близькості між об'єктами.

3. Обирають метод розрахунку близькості та розраховують за цим методом міри близькості – подібності чи несхожості (відстані – для метричних даних) від нового об'єкта до кожного об'єкта навчаючої множини.

4. Здійснюють пошук k найближчих сусідів: знаходять k -об'єктів навчаючої множини, які є найближчими до нового об'єкта на основі визначених мір близькості.

5. Визначають клас нового об'єкта, використовуючи ***функцію сполучення*** (англ. *combination function*), яка може бути такою:

   a) ***просте незважене голосування*** (англ. *simple unweighted voting*): класом нового об'єкта буде клас, який найбільш часто зустрічається серед виявлених k -найближчих сусідів (величина мір близькості до сусідів не враховується);

   b) ***зважене голосування*** (англ. *weighted voting*): класом нового об'єкта буде клас, який набрав найбільшу кількість голосів, визначених із урахуванням мір близькості до нового об'єкта серед знайдених k-найближчих сусідів.

У випадку зваженого голосування залежно від методу, який було обрано для розрахунку мір близькості, ***вагу голосу*** $V(c_j)$ класу c_j, який є серед k-найближчих сусідів, при зваженому голосуванні розраховують за однією з формул:

при розрахунку мір несхожості: $$V(c_j)=\sum_{k=1}^{n_j}\frac{1}{d^2(i_h,i_{kj})}, \tag{8.6}$$

при розрахунку мір подібності: $$V(c_j)=\sum_{k=1}^{n_j}s^2(i_h,i_{kj}), \tag{8.7}$$

де $d^2(i_h,i_{kj})$ – квадрат міри несхожості (відстані для метричних даних) між новим об'єктом i_h та k-м об'єктом j-го класу,

$s^2(i_h,i_{kj})$ – квадрат міри подібності між новим об'єктом i_h та k-м об'єктом j-го класу,

n_j – кількість об'єктів j-го класу серед k найближчих сусідів.

Оскільки незбалансованість навчаючої множини за класами може привести до невірних результатів, використання зваженого голосування при визначенні класу нового об'єкта може покращити результат.

Особливістю методу k-найближчих сусідів є те, що структура моделі побудованого класифікатора не задається жорстко наперед, а визначається даними. Це певною мірою є недоліком, оскільки модель не можна відокремити від даних.

Вузьким місцем методу є правильний вибір параметра k – кількості найближчих сусідів. Якщо прийняти $k=1$, то алгоритм утратить узагальнюючу здатність, а якщо встановити занадто велике значення k, то багато локальних особливостей не будуть виявлені.

На практиці для визначення *оптимального значення* k застосовують ***метод ковзного контролю***. Відповідно до цього методу, оптимальним вважають таке значення k, яке буде давати найменшу помилку класифікації кожного об'єкта навчаючої множини. Це здійснюють у декілька етапів, на яких при різних фіксованих значеннях k здійснюють класифікацію кожного об'єкта, попередньо вилучивши його з навчаючої множини.

До переваг методу k-найближчих сусідів відносять: стійкість до аномальних викидів, достатньо просту програмну реалізацію та інтерпретацію отриманих результатів.

До недоліків методу відносять підвищені вимоги до репрезентативності даних навчаючої множини та високу обчислювальну трудомісткість, яка стає суттєвою у випадку великої кількості об'єктів навчаючої множини.

8.1.7. Приклад класифікації за методом k-найближчих сусідів

Приклад 1. За результатами попередньо проведеного аналізу було з'ясовано, що вхідні дані можуть бути розбиті на 2 класи, у першому *класі 1* будуть об'єкти 1, 2 і 3, у другому *класі 2* – об'єкти 4, 5 і 6 (табл. 8.1). Здійснити класифікацію нового об'єкта, ознаки якого рівні: x = 11, y = 9, обравши для класифікації k = 4 найближчих сусіди.

Таблиця 8.1

Значення ознак об'єктів

Ознаки	Об'єкти					
	1	2	3	4	5	6
x	2	4	5	12	14	15
y	8	10	7	6	6	4
Клас	1			2		

1. Значення незалежних змінних знаходяться у одному числовому діапазоні, тому нормалізацію значень здійснювати немає потреби.

2. Для оцінки близькості об'єктів оберемо відстань Евкліда та розрахуємо в MS Excel відповідні відстані від нового об'єкта до кожного з уже класифікованих шести об'єктів (рис. 8.2).

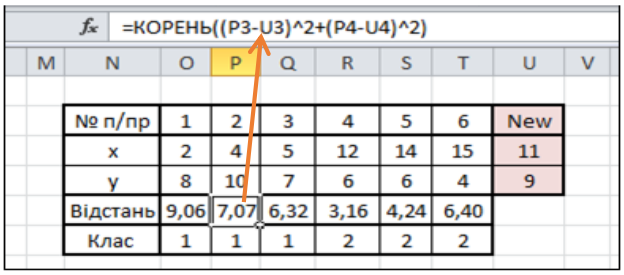


Рис. 8.2. Розрахунок відстаней від нового об'єкта до уже класифікованих об'єктів

3. Серед знайдених відстаней знайдемо 4 найменші та визначимо, що найближчими сусідами до нового об'єкта будуть об'єкти 3, 4, 5 і 6 (рис. 8.3).

№ п/пр	1	2	3	4	5	6	New
x	2	4	5	12	14	15	11
y	8	10	7	6	6	4	9
Відстань	9,06	7,07	6,32	3,16	4,24	6,40	
Клас	1	1	1	2	2	2	

Рис. 8.3. Визначення найближчих до нового об'єкта сусідів

4. Серед 4-х найближчих об'єктів до класу 1 відноситься один об'єкт, а до класу 2 – три. Використовуючи просте незважене голосування, робимо висновок: новий об'єкт відноситься до класу 2.

5. Визначимо належність нового об'єкта до одного із двох класів, використовуючи зважене голосування. Розрахуємо ваги голосів класу 1 та класу 2 за формулою 8.6:

$$V(y_1)=\frac{1}{6{,}32^2}=0{,}025,\qquad V(y_2)=\frac{1}{3{,}16^2}+\frac{1}{4{,}24^2}+\frac{1}{6{,}40^2}=0{,}180\cdot$$

Використання зваженого голосування також дозволяє зробити висновок про те, що новий об'єкт відноситься до класу 2, оскільки $V(y_2) > V(y_1)$.

8.1.8. Метод опорних векторів

Метод опорних векторів (англ. *Support Vector Machines, SVM*) – належить до сімейства популярних класифікаторів, здатних виконувати лінійну класифікацію.

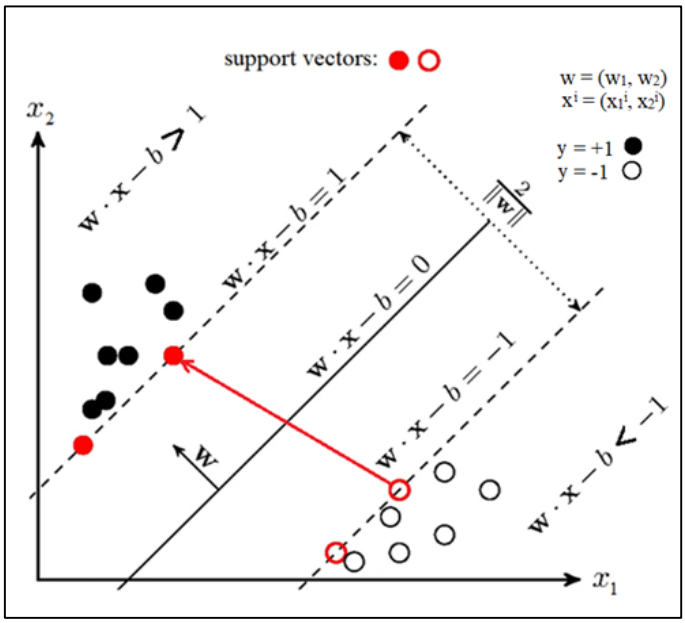
Об'єкти навчаючої множини розглядають як вектори з m незалежними ознаками та однією залежною ознакою: $\{x_i, y_i; i = \overline{1,m}\}$, де $x_i = (x_{i1}, x_{i2}, \ldots x_{im})$ – вектор незалежних змінних об'єкта, а y_i позначає належність об'єкта до одного із двох класів і може приймати два значення: +1 або –1.

Побудова лінійного класифікатора за методом опорних векторів – ***SVM-класифікатора***, полягає у знаходженні такої функції $y = F(x)$, яка, приймаючи як аргументи значення незалежних змінних нового об'єкта, буде видавати значення y, рівні +1 або –1, визначаючи належність нового об'єкта до певного класу. Функція $F(x)$ буде задавати гіперплощину у просторі розмірністю m:

$$w_1 x_1 + w_2 x_2 + \ldots + w_m x_m + w_0 = 0. \tag{8.8}$$

Ця гіперплощина буде розділяти набір даних навчаючої множини на два класи таким чином, що об'єкти одного класу будуть знаходитися з однієї її сторони, а об'єкти іншого класу – з іншої.

У процесі побудови класифікатора необхідно знайти коефіцієнти рівняння гіперплощини w_i, де $i = \overline{1,m}$. Їх знаходять таким чином, щоб об'єкти навчаючої множини лежали якомога далі від розподіляючої гіперплощини, з максимальним зазором (відстанню) між нею та об'єктами різних класів, які розташовані ближче всього до неї. Такі об'єкти називають ***опорними векторами*** (англ. *Support Vectors*) (рис. 8.4).



($mym\ b = -w_0$)

Рис. 8.4. Ілюстрація методу опорних векторів у двовимірному просторі ознак

Через опорні вектори проводять площини, паралельні розподіляючій гіперплощині. Вони утворюють зазор – коридор між класами, ширина якого рівна: $\frac{2}{\|w\|}$, де $\|w\| = \sqrt{w \cdot w^T}$ – евклідова норма вектора $w = (w_1, w_2, \ldots w_m)$. Усі об'єкти навчаючої множини будуть знаходитися за межами цього зазору або на його межах.

Задача побудови оптимальної гіперплощини полягає у розв'язанні оптимізаційної задачі – знаходженні із застосуванням методу множників Лагранжа таких $w = (w_1, w_2, \ldots w_m)$ і w_0, які мінімізують функцію $\Phi(w) = \frac{1}{2}\|w\|^2$ у разі виконання умов $y_i(wx_i^T + w_0,) \geq 1;\ \ i = \overline{1,m}$.

Метод розроблено для бінарної класифікації. Для задач із більшою кількістю класів можна будувати розподіляючі гіперплощини для кожної пари.

На практиці класи, які є лінійно відокремлюваними, зустрічаються рідко. У цьому випадку розподіляючу гіперплощину будують, допускаючи помилки класифікації: частина об'єктів може лежати у зазорі, а деякі – у області іншого класу.

Нелінійне узагальнення лінійного SVM-класифікатора базується на розширенні розмірності простору змінних за допомогою спеціальних ***ядерних функцій***. Використання поліноміального, радіального, сигмоідального та інших ядер дозволяє будувати моделі класифікатора з використанням нелінійних розподіляючих поверхонь різних форм.

До недоліків методу опорних векторів слід віднести чутливість до шуму, а до переваг – те, що модель класифікатора будують шляхом розв'язання математичної задачі, яка має єдиний розв'язок.

8.2. КЛАСИФІКАЦІЯ ЗА ДОПОМОГОЮ МЕТОДІВ 1R, NAIVE BAYES

8.2.1. Навчаюча множина даних для побудови класифікатора

Необхідно побудувати класифікатор, який буде визначати можливість проведення гри у футбол за певних погодних умов, на підставі інформації про проведення ігор за різних погодних умов у минулому (табл. 8.2).

Таблиця 8.2

Дані про погодні умови при проведенні ігор у минулому

№	Спостереження	Температура	Вологість	Вітер	Гра
1	Сонце	Спекотно	Висока	Немає	Немає
2	Сонце	Спекотно	Висока	Є	Немає
3	Хмарність	Спекотно	Висока	Немає	Так
4	Дощ	Норма	Висока	Немає	Так
5	Дощ	Холодно	Норма	Немає	Так
6	Дощ	Холодно	Норма	Є	Немає
7	Хмарність	Холодно	Норма	Є	Так
8	Сонце	Норма	Висока	Немає	Немає
9	Сонце	Холодно	Норма	Немає	Так
10	Дощ	Норма	Норма	Немає	Так
11	Сонце	Норма	Норма	Є	Так
12	Хмарність	Норма	Висока	Є	Так
13	Хмарність	Спекотно	Норма	Немає	Так
14	Дощ	Норма	Висока	Є	Немає

Для побудови класифікатора зробимо аналіз таблиці 8.2, яка містить навчальну множину даних. Маємо:

1) чотири незалежні змінні:

X_1 – *Спостереження*: зі значеннями «Сонце», «Дощ» та «Хмарність»;

X_2 – *Температура*: зі значеннями «Спекотно», «Норма» та «Холодно»;

X_3 – *Вологість*: зі значеннями «Висока» та «Норма»;

X_4 – *Вітер*: зі значеннями «Є» та «Немає»;

2) одну залежну змінну:

Y – *Гра:* зі значеннями «Так» і «Немає».

У нашому випадку класів буде два, вони відповідають двом можливим значенням незалежної змінної *Y* (*Гра*) «Так» та «Немає». Новий об'єкт, для якого необхідно визначити значення незалежної змінної, може бути віднесений до одного з цих класів.

Поставлену задачу розв'яжемо з використанням двох методів: 1R та Naive Bayes.

8.2.2. Приклад класифікації з використанням методу 1R

Приклад 2. Визначити, чи відбудеться гра у футбол за погодних умов, заданих у таблиці 8.3, на підставі інформації про проведення ігор за різних погодних умов у минулому (табл. 8.2) із використанням методу 1R.

1. На наборі даних навчаючої множини, представленому у таблиці 8.2, для усіх можливих значень кожної незалежної змінної визначаємо частоти кожного значення залежної змінної – класу, який зустрічається з обраним значенням незалежної змінної (табл. 8.4).

2. На основі виявлених частот для кожного значення кожної незалежної змінної визначаємо клас, який найбільш часто зустрічається з даним значенням незалежної змінної, та формуємо відповідні правила класифікації і

визначаємо помилку кожного правила – кількість об'єктів, які мають дане значення незалежної змінної, але не відносяться до обраного класу (табл. 8.5).

Таблиця 8.3

Дані про погодні умови у дні, коли планується гра

№	Спостереження	Температура	Вологість	Вітер	Гра
1	Сонце	Спекотно	Норма	Немає	?

Таблиця 8.4

Визначення частот кожного значення залежної змінної для кожного значення кожної незалежної змінної

Незалежні змінні X_i		Всього	Залежна змінна Y	Кількість	Частота
Назва	Значення				
X_1 Спостереження	Сонце	5	Так	2	2/5
			Немає	3	3/5
	Дощ	5	Так	3	3/5
			Немає	2	2/5
	Хмарність	4	Так	4	4/4
			Немає	0	0/4
X_2 Температура	Спекотно	4	Так	2	2/4
			Немає	2	2/4
	Холодно	4	Так	3	3/4
			Немає	1	1/4
	Норма	6	Так	4	4/6
			Немає	2	2/6
X_3 Вологість	Висока	7	Так	3	3/7
			Немає	4	4/7
	Норма	7	Так	6	6/7
			Немає	1	1/7
X_4 Вітер	Є	6	Так	3	3/6
			Немає	3	3/6
	Немає	8	Так	6	6/8
			Немає	2	2/8

Таблиця 8.5

Визначення помилок сформованих правил класифікації

№ з/п	Правило	Помилка правила
1	*Якщо* (X_1=Сонце) *то* (Y = Немає)	2/5=0,4
2	*Якщо* (X_1=Дощ) *то* (Y = Немає)	2/5=0,4
3	*Якщо* (X_1=Хмарність) *то* (Y = Так)	0/4=0
4	*Якщо* (X_2=Спекотно) *то* (Y = Немає)*	2/4=0,5
5	*Якщо* (X_2=Холодно) *то* (Y = Так)	1/4=0,25
6	*Якщо* (X_2=Норма) *то* (Y = Так)	2/6=0,33
7	*Якщо* (X_3=Висока) *то* (Y = Немає)	3/7=0,43
8	*Якщо* (X_3=Норма) *то* (Y = Так)	1/7=0,15
9	*Якщо* (X_4=Є) *то* (Y = Немає)*	3/6=0,5
10	*Якщо* (X_4=Немає) *то* (Y = Так)	2/8=0,25

3. Оцінюємо ступінь помилок сформованого набору правил та обираємо ту незалежну змінну, правило для якої побудоване з найменшою помилкою. Такою змінною у нашому прикладі є змінна X_3 – *Вологість*. Саме цю змінну будемо використовувати при класифікації нових об'єктів.

4. Для погодних даних, представлених у таблиці 8.3, значення змінної X_3= Норма, тому за побудованим із використанням методу 1R класифікатором значення залежної змінної *Y* (*Гра*) буде «Так», гра відбудеться.

8.2.3. Приклад класифікації з використанням методу Naive Bayes

Приклад 3. Визначити, чи відбудеться гра у футбол за погодних умов, заданих у таблиці 8.3, на підставі інформації про проведення ігор за різних погодних умов у минулому (табл. 8.2) із використанням методу Naive Bayes.

1. Розраховуємо апріорну ймовірність появи кожного класу в наборі даних навчаючої множини (табл. 8.6).

2. Користуючись інформацією, представленою у таблиці 8.2, розраховуємо апріорні ймовірності кожного значення кожної незалежної змінної навчаючої множини (табл. 8.7).

Таблиця 8.6

Розрахунок апріорної ймовірності кожного класу навчаючої множини

Залежна змінна, Y	Значення Y	Кількість	Апріорна ймовірність	
			P(Y)	Розрахункова формула
Клас 1	Немає	5	0,36	P(Y=Немає) = 5/14
Клас 2	Так	9	0,64	P(Y=Так) = 9/14
Всього		14	1	

Таблиця 8.7

Розрахунок апріорних ймовірностей значень незалежних змінних

Незалежні змінні, X_i		Кількість	Апріорна ймовірність	
Назва	Значення		P(X_i)	Розрахункова формула
X_1 Спостереження	Сонце	5	0,36	P(X_1=Сонце) = 5/14
	Дощ	5	0,36	P(X_1=Дощ) = 5/14
	Хмарність	4	0,29	P(X_1=Хмарність) = 4/14
X_2 Температура	Спекотно	4	0,29	P(X_2=Спекотно) = 4/14
	Холодно	4	0,29	P(X_2=Холодно) = 4/14
	Норма	6	0,43	P(X_2=Норма) = 6/14
X_3 Вологість	Висока	7	0,50	P(X_3=Висока) = 7/14
	Норма	7	0,50	P(X_3=Норма) = 7/14
X_4 Вітер	Є	6	0,43	P(X_4=Є) = 4/14
	Немає	8	0,57	P(X_4=Немає) = 8/14

3. Користуючись інформацією, представленою у таблиці 8.2, розраховуємо апостеріорні ймовірності значень кожної змінної за умови належності до першого та другого класу (табл. 8.8).

4. Розрахунок апріорних та апостеріорних ймовірностей, представлених у таблицях 8.5, 8.6, 8.7, дає можливість здійснити класифікацію нового об'єкта: визначити, чи буде проводитися гра за вказаних у таблиці 8.3 погодних умов. Для об'єкта, представленого у цій таблиці, маємо наступні значення незалежних змінних:

Спостереження: X_1 = Сонце,
Температура: X_2 = Спекотно
Вологість: X_3 = Норма
Вітер: X_4 = Немає

Позначимо через E подію, що відповідає рівності кожної з незалежних змінних, які характеризують новий об'єкт – погодні умови, цим значенням.

5. Відберемо з таблиць 8.6 та 8.7 апріорні й апостеріоні ймовірності значень незалежних змінних, які відповідають події E (табл. 8.9) та розрахуємо апостеріорні ймовірності події E за умови, що значення незалежної змінної Y буде приймати значення «Так» та «Немає»:

$$P(E|Y = Так) = 0{,}22 \cdot 0{,}22 \cdot 0{,}67 \cdot 0{,}67 = 0{,}02195,$$
$$P(E|Y = Немає) = 0{,}6 \cdot 0{,}4 \cdot 0{,}2 \cdot 0{,}4 = 0{,}0192.$$

Таблиця 8.8

Розрахунок апостеріорних ймовірностей значень незалежних змінних

Незалежні змінні X_i		Залежна змінна Y	Кількість	Апостеріорна ймовірність	
Назва	Значення			$P(X_i \mid Y)$	Розрахункова формула
X_1 Спостереження	Сонце	Так	2	0,22	$P(X_1$=Сонце | Y=Так) = 2/9
		Немає	3	0,6	$P(X_1$=Сонце | Y=Немає) = 3/5
	Дощ	Так	3	0,33	$P(X_1$=Дощ | Y=Так) = 3/9
		Немає	2	0,4	$P(X_1$=Дощ | Y=Немає) = 2/5
	Хмарність	Так	4	0,44	$P(X_1$=Хмарність | Y=Так) = 4/9
		Немає	0	0	$P(X_1$=Хмарність | Y=Немає) = 0/5
X_2 Температура	Спекотно	Так	2	0,22	$P(X_2$=Спекотно | Y=Так) = 2/9
		Немає	2	0,4	$P(X_2$= Спекотно | Y=Немає) = 2/5
	Холодно	Так	3	0,33	$P(X_2$=Холодно | Y=Так) = 3/9
		Немає	1	0,2	$P(X_2$=Холодно | Y=Немає) = 1/5
	Норма	Так	4	0,44	$P(X_2$=Норма | Y=Так) = 4/9
		Немає	2	0,4	$P(X_2$=Норма | Y=Немає) = 2/5
X_3 Вологість	Висока	Так	3	0,33	$P(X_3$=Висока | Y=Так) = 3/9
		Немає	4	0,8	$P(X_3$=Висока | Y=Немає) = 4/5
	Норма	Так	6	0,67	$P(X_3$=Норма | Y=Так) = 6/9
		Немає	1	0,2	$P(X_3$=Норма | Y=Немає) = 1/5
X_4 Вітер	Є	Так	3	0,33	$P(X_4$=Є | Y=Так) = 3/9
		Немає	3	0,6	$P(X_4$=Є | Y=Немає) = 3/5
	Немає	Так	6	0,67	$P(X_4$=Немає | Y=Так) = 6/9
		Немає	2	0,4	$P(X_4$=Немає | Y=Немає) = 2/5

6. Із таблиці 8.6 маємо наступні апріорні значення залежної змінної Y:

$$P(Y = Так) = \frac{9}{14} = 0{,}64 \qquad P(Y = Немає) = \frac{5}{14} = 0{,}36 \cdot$$

Тоді ймовірність події E, розрахована за формулою 8.4, буде рівна:

$$P(E) = P(E \mid Y = Так) \cdot P(Y = Так) + P(E \mid Y = Немає) \cdot P(Y = Немає) = 0{,}02195 \cdot 0{,}64 + 0{,}0192 \cdot 0{,}36 = 0{,}021.$$

Таблиця 8.9

Розрахунок апріорних ймовірностей значень незалежних змінних

Незалежні змінні, X_i	Апріорні ймовірності $P(X_i)$	Залежна змінна Y	Апостеріорні ймовірності
X_1= Сонце	$P(X_1$=Сонце) = 5/14=0,36	Так	$P(X_1$=Сонце | Y=Так) = 2/9=0,22
		Немає	$P(X_1$=Сонце | Y=Немає) = 3/5=0,6
X_2= Спекотно	$P(X_2$=Спекотно) = 4/14=0,29	Так	$P(X_2$=Спекотно | Y=Так) = 2/9=0,22
		Немає	$P(X_2$= Спекотно | Y=Немає) = 2/5=0,4
X_3= Норма	$P(X_3$=Норма) = 7/14=0,5	Так	$P(X_3$=Норма | Y=Так) = 6/9=0,67
		Немає	$P(X_3$=Норма | Y=Немає) = 1/5=0,2
X_4= Немає	$P(X_4$=Немає) = 8/14=0,57	Так	$P(X_4$=Немає | Y=Так) = 6/9=0,67
		Немає	$P(X_4$=Немає | Y=Немає) = 2/5=0,4

7. Апостеріорну ймовірність того, що при погодних умовах, що відповідають події E, гра відбудеться – значення незалежної змінної Y (*Гра*) буде «Так», відповідно до теореми Байєса розраховуємо за формулою:

$$P(Y = Так \mid E) = \frac{p(E \mid Y = Так) \cdot P(Y = Так)}{P(E)} = \frac{0{,}02195 \cdot 0{,}64}{0{,}021} = 0{,}673.$$

8. Апостеріорну ймовірність того, що за даних погодних умов, що відповідають події E, гра не відбудеться – значення незалежної змінної Y (*Гра*) буде «Немає», відповідно до теореми Байєса розраховуємо за формулою:

$$P(Y = Немає \mid E) = \frac{p(E \mid Y = Немає) \cdot P(Y = Немає)}{P(E)} = \frac{0{,}0192 \cdot 0{,}36}{0{,}021} = 0{,}327.$$

9. При порівнянні розрахованих апостеріорних ймовірностей ми отримали:

$$P(y = \text{Так} \mid E) > P(y = \text{Немає} \mid E), \text{ оскільки } 0{,}673 > 0{,}327.$$

У даному випадку можна стверджувати, що за погодних умов, які відповідають події E:

1) гра відбудеться (клас 2) із ймовірністю 0,673;
2) гра не відбудеться (клас 1) із ймовірністю 0,327.

Отриманий результат дає підстави віднести погодні умови, представлені у таблиці 8.3, до класу 2 – гра відбудеться.

Завдання 1. Використовуючи побудований у прикладі 3 класифікатор, здійснити класифікацію інших об'єктів. Визначити, чи відбудеться гра у футбол за погодних умов, заданих у таблиці 8.10 для об'єктів 1 та 2 (погодні умови, представлені у першому та другому рядках цієї таблиці).

Таблиця 8.10

Дані про погодні умови у дні, коли планується гра

№	Спостереження	Температура	Вологість	Вітер	Гра
1	Дощ	Холодно	Висока	Немає	?
2	Хмарність	Норма	Висока	Є	?

8.3. РОЗВ'ЯЗОК ЗАДАЧІ КЛАСИФІКАЦІЇ ІРИСІВ У СЕРЕДОВИЩІ MATLAB

8.3.1. Класична задача класифікації ірисів

У задачі класифікації ірисів розглядається класифікація квіток ірисів за такими чотирма ознаками:

1) x_1 – довжина чашолистка;
2) x_2 – ширина чашолистка;
3) x_3 – довжина пелюстки;
4) x_4 – ширина пелюстки.

Завдання полягає у розробці моделі, яка на основі ознак (x_1, x_2, x_3, x_4) правильно відносить ірис до одного з 3-х класів (рис. 8.5):

1) *C1* – ірис сетоса (англ. *Iris-setosa*);
2) *C2* – ірис веселковий (англ. *Iris-versicolor*);
3) *C3* – ірис віржиніка (англ. *Iris-virginica*).

Рис. 8.5. Фото ірисів різних класів

Необхідно побудувати класифікатор для набору даних класичної задачі класифікації ірисів Фішера та виконати класифікацію нового екземпляра.

Виконання поставленої задачі базується на наборі даних, зібраному Р. Фішером, який містить описані вище характеристики для 150 квіток ірисів трьох класів, по 50 представників кожного класу. Набір даних міститься у файлі *fisheriris.mat*, доступному за посиланням:

https://drive.google.com/file/d/10jUJyJfwnn3g7S5zG-6l74dNhIKKUeuo/view?usp=sharing.

Розглянемо різні підходи до розв'язання поставленої задачі.

8.3.2. Класифікація методом k-найближчих сусідів

Завдання 2. Розв'язати задачу класифікації ірисів із використанням методу k-найближчих сусідів.

1. Файл із необхідним набором даних *fisheriris.mat* потрібно завантажити у папку та відкрити у середовищі MatLab із допомогою команди *load* або команд меню.

2. Після відкриття файлу у робочому середовищі *Workspace* MatLab з'являються дві матриці з вхідними даними:

a) *means* – матриця, що має чотири стовпці, які відповідають характеристикам квіток ірису – незалежним змінним x_1, x_2, x_3, x_4 та 150 рядків, кожен з яких містить значення незалежних змінних для окремої квітки одного із трьох класів;

b) *species* – матриця, що містить значення залежної змінної для кожного із об'єктів, представлених у матриці means – клас квітки.

Побудову класифікатора здійснимо, відібравши із 4-х наявних дві характеристики ірисів – 1-шу та 3-тю.

3. Візуалізуємо набір даних та позначимо на графіку новий екземпляр, який необхідно класифікувати, ввівши у вікні *Command Window* команди:

```
>> x = [meas(:,1) meas(:,3)]; % відбір 1 та 3 ознак
>> gscatter(x(:,1), x(:,2), species, 'rgb')
>> set(legend, 'location', 'best') % задання властивостей графічних об'єктів
>> hold on % збереження поточного графіку
>> newpoint = [6.2 4.9]; % задання  нової точки
>> line(newpoint(1), newpoint(2), 'marker', 'x', 'color', 'black',...
'markersize',8,'linewidth',2) % візуалізація нової точки
```

4. У окремому вікні буде виведено графічне зображення набору даних та точки – нової квітки, яку необхідно класифікувати (рис. 8.6).

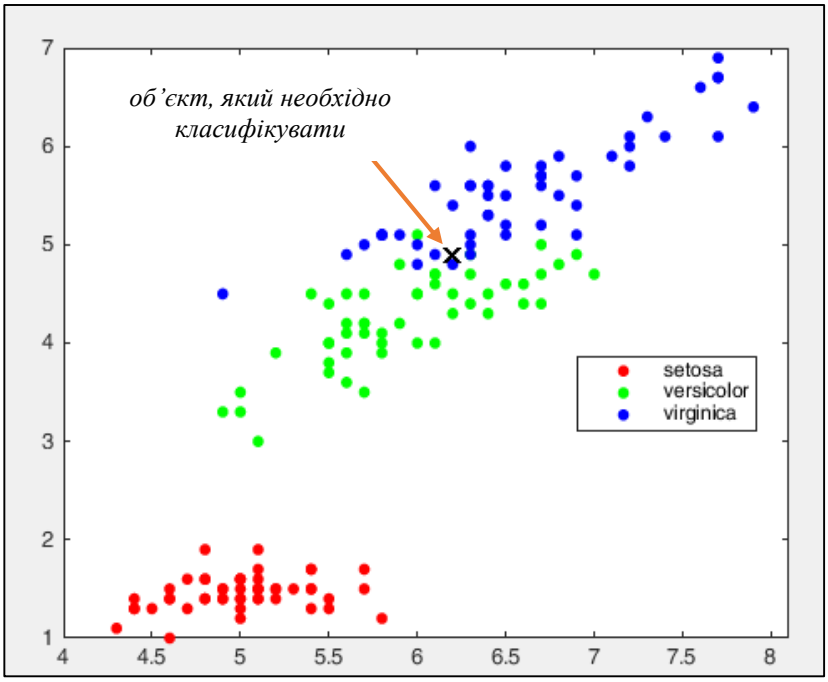


Рис. 8.6. Візуалізація набору даних навчаючої множини та нового об'єкту, який необхідно класифікувати

5. Для визначення найближчих сусідів необхідно вказати їх кількість k=10 та ввести команди для отримання їх характеристик:

```
>> [n,d] = knnsearch(x, newpoint, 'k', 10);
>>  class = species(n,1);
>> iris = [x(n,1), x(n,2)]
```

Тут:

knnsearch() – функція MatLab для знаходження k-найближчих сусідів;

n – порядковий номер сусіда у навчаючій множині;

d – відстань від сусіда до нового об’єкта;

class – містить назви класів найближчих сусідів;

iris – містить значення ознак найближчих сусідів.

6. У вікні *Command Window* введемо команди для виведення значень змінних найближчих до нового об’єкта сусідів (рис. 8.7). Серед них є об’єкти, які досить близько розташовані один до одного, та 2 об’єкти, які зовсім співпадають.

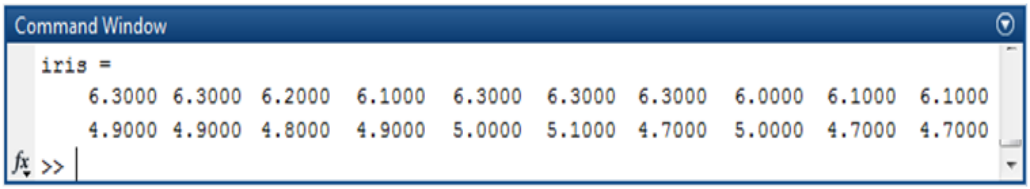


Рис. 8.7. Координати найближчих сусідів нового об’єкта

7. Для візуального представлення найближчих сусідів введемо команду:

```
>> line(x(n,1), x(n,2), 'color', [.5 .5 .5], 'marker', 'o','linestyle', 'none', 'markersize',10)
```

На графіку з’явиться зображення найближчих сусідів, виділених контурними лініями (рис. 8.8, рис. 8.9). На графіку зображено лише 8 точок, оскільки дві з них мають однакові координати.

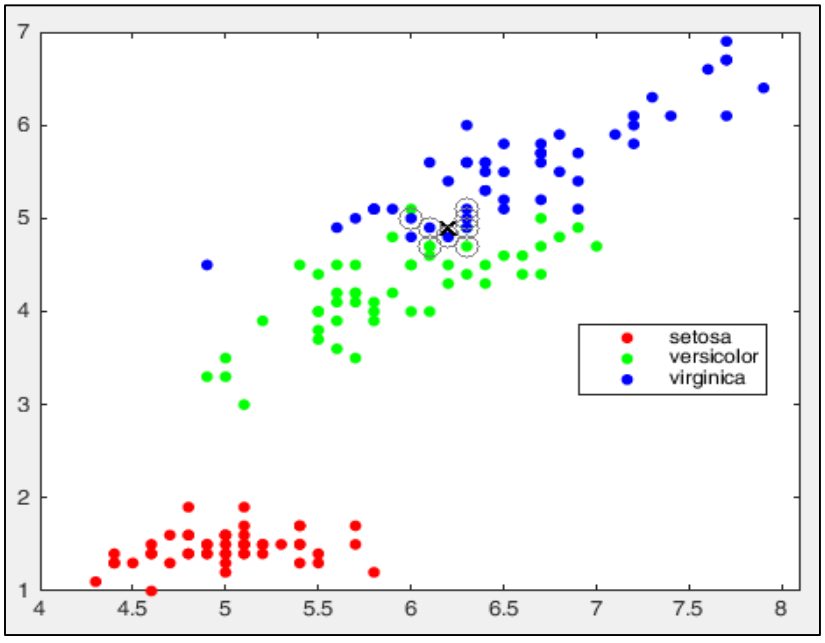


Рис. 8.8. Графічна інтерпретація найближчих сусідів

9. Для прийняття рішення, до якого класу слід віднести новий об’єкт, за допомогою команди *tabulate* визначимо відсоткове співвідношення класів знайдених сусідів (рис. 8.10). Ми бачимо, що серед найближчих сусідів є представники двох класів: *Virginica* та *Versicolor* (рис. 8.11).

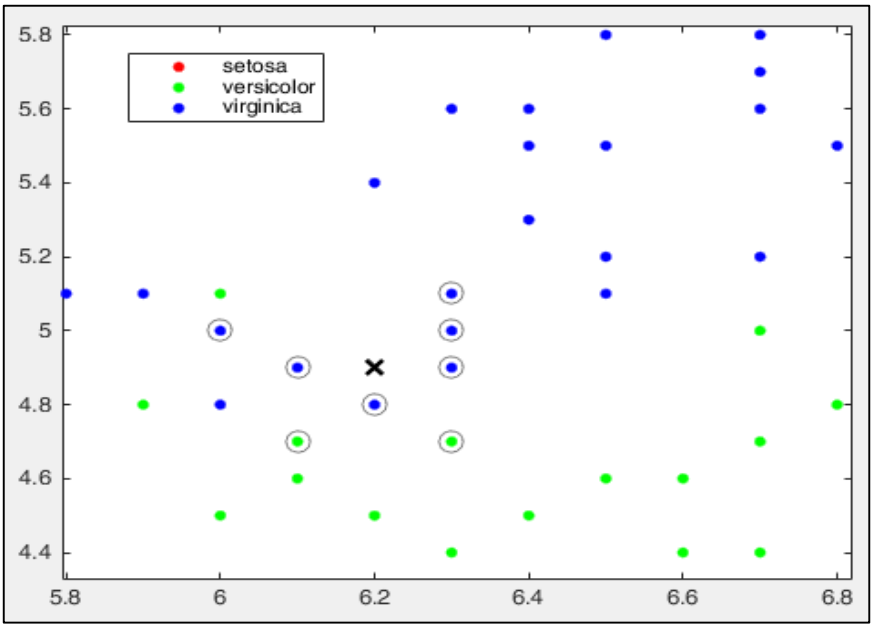


Рис. 8.9. Графічне зображення найближчих сусідів у збільшеному масштабі

```
>> tabulate(species(n))
        Value     Count    Percent
   versicolor         4     40.00%
    virginica         6     60.00%
fx >>
```

Рис. 8.10. Процентне співвідношення класів найближчих сусідів

```
Command Window
class =
    'versicolor'
    'virginica'
    'virginica'
    'virginica'
    'virginica'
    'virginica'
    'versicolor'
    'virginica'
    'versicolor'
    'versicolor'
```

а) класи найближчих сусідів

```
Command Window
>> d=d'
d =
    0.1000
    0.1000
    0.1000
    0.1000
    0.1414
    0.2236
    0.2236
    0.2236
    0.2236
    0.2236
```

б) відстані до нового об’єкта

Рис. 8.11. Інформація про найближчих сусідів нового об’єкта

10. Припустимо, що використовується *просте незважене голосування*. Тоді можна зробити висновок про належність нового об’єкта до класу *Virginica*, бо до цього класу відноситься 60% найближчих сусідів.

11. Визначимо належність нового об’єкта до одного із двох класів, використовуючи зважене голосування. Розрахуємо ваги голосів класу *Virginica* та класу *Versicolor* за формулою 8.6:

$$V(Virginica) = \frac{1}{0{,}1^2} + \frac{1}{0{,}1^2} + \frac{1}{0{,}1^2} + \frac{1}{0{,}1414^2} + \frac{1}{0{,}2236^2} + \frac{1}{0{,}2236^2} = 390{,}02,$$

$$V(Versicolor) = \frac{1}{0{,}1^2} + \frac{1}{0{,}2236^2} + \frac{1}{0{,}2236^2} + \frac{1}{0{,}2236^2} = 160{,}00.$$

Таким чином, використання зваженого голосування також дозволяє зробити висновок про те, що новий об'єкт відноситься до класу *Virginica*, оскільки $V(Virginica) > V(Versicolor)$.

8.3.3. Класифікація методом опорних векторів

Завдання 3. Розв'язати задачу класифікації ірисів із використанням методу опорних векторів.

1. Працюємо з набором даних класичної задачі класифікації квіток ірисів трьох класів: Iris Setosa, Iris Versicolour, Iris Virginica, який уже представлений у робочому середовищі *Workspace* MatLab двома матрицями:

a) *means* – матриця, що має чотири стовпці, які відповідають характеристикам квіток ірису – незалежним змінним x_1, x_2, x_3, x_4 та 150 рядків, кожен з яких містить значення незалежних змінних для окремої квітки одного із трьох класів;

b) *species* – матриця, що містить значення залежної змінної для кожного із об'єктів, представлених у матриці means – клас квітки.

Побудову класифікатора здійснимо, відібравши із 4-х наявних дві характеристики ірисів: 1-шу – довжина чашолистки та 3-тю – довжина пелюстки, а також вихідний – клас квітки.

Після побудови класифікатора методом опорних векторів необхідно визначити клас екземплярів із наступними значеннями довжини чашолистка і пелюстки в сантиметрах: (3,2; 1,5) та (5,4; 2).

2. Побудуємо *SVM-класифікатор* для задачі класифікації ірисів, створивши скрипт у MatLab із наступним кодом:

```
clc; clear;
load fisheriris % завантаження набору даних

% формування набору даних для побудови класифікатора
inds = ~strcmp(species, 'setosa'); % видалення елементів класу 'setosa'
X1 = meas(inds,3); X2 = meas(inds,4); % відбір 3-ї та 4-ї ознак
X = [X1 X2]; % матриця 2-х ознак 3-ї та 4-ї
y = species(inds); % вектор-стовпець 2-х класів 'versicolor' та 'virginica'

% навчання SVM класифікатора на сформованому наборі даних
SVMModel = fitcsvm(X,y)
classOrder = SVMModel.ClassNames % визначення порядку класів

% побудова діаграми даних та обведення опорних векторів
sv = SVMModel.SupportVectors;
figure
gscatter(X(:,1),X(:,2),y)
hold on
plot(sv(:,1),sv(:,2),'ko','MarkerSize',10)
legend('versicolor','virginica','Support Vector')

% побудова лінії розмежування класів
d = 0.02;
[x1Grid,x2Grid] = meshgrid(min(X(:,1)):d:max(X(:,1)),...
    min(X(:,2)):d:max(X(:,2)));
xGrid = [x1Grid(:),x2Grid(:)];
[~,scores] = predict(SVMModel,xGrid);
hold on
contour(x1Grid,x2Grid,reshape(scores(:,2),size(x1Grid)),[0 0],'k');

% навчання та перехресна перевірка моделі класифікатора
CVSVMModel = crossval(SVMModel);
classLoss = kfoldLoss(CVSVMModel) % оцінка якості класифікації

% задання нових об'єктів для класифікації
newpoint1 = [3.2 1.5];  newpoint2 = [5.4 2];
% візуалізація нових об'єктів
```

```
hold on;  plot(3.2, 1.5, 'bo', 'MarkerSize', 10, 'linewidth', 2);
hold on;  plot(5.4, 2, 'bo', 'MarkerSize', 10, 'linewidth', 2);

% класифікація нових об'єктів
[label1,score1] = predict(SVMModel,newpoint1);
[label2,score2] = predict(SVMModel,newpoint2);
label1 % виведення на екран класу 1-го об'єкта
label2 % виведення на екран класу 2-го об'єкта
```

Пояснення до коду:

SVMModel – це навчена ***модель SVM-класифікатора***. Класифікація по замовчуванню здійснюється з використанням лінійної функції ядра.

Опорні вектори – це точки (дані), які знаходяться на границі класів або недалеко від неї.

CVSVMModel – здійснює ***перехресну перевірку*** моделі класифікатора (по замовчуванню прехресна перевірка є 10-кратною).

Оцінка якості класифікації міститься у змінній *classLoss*.

Класифікація нового об'єкта здійснюється за допомогою функції *predict()*, яка повертає ймовірний клас нового об'єкта.

Змінні *label1* та *label2* містять клас першого та другого нових об'єктів, які було класифіковано за допомогою побудованої моделі SVM-класифікатора.

3. У вікні *Command Window* введемо команди для виведення значень змінних *label1* та *label2* – класів нових об'єктів (3,2; 1,5) і (5,4; 2) та змінної *classLoss* з оцінкою якості класифікації (рис. 8.12).

Виведені на екран значення свідчать про те, що у результаті роботи SVM-класифікатора було визначено, що квітка із характеристиками (3,2; 1,5) відноситься до класу *Versicolor*, а квітка із характеристиками (5,4; 2) відноситься до класу *Virginica*. Значення змінної *classLoss*, рівне 0,05, свідчить про те, що помилка класифікації складає близько 5%.

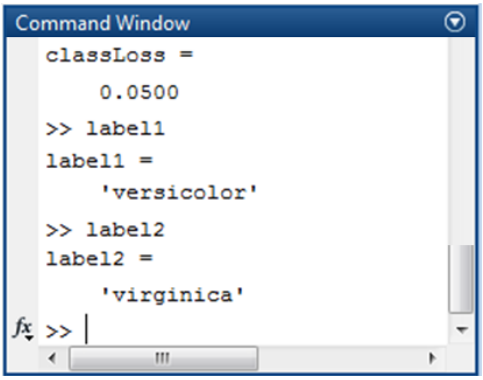


Рис. 8.12. Класи нових об'єктів, визначені за допомогою SVM-класифікатора

4. На рисунку 8.13 зображено результат роботи створеного скрипта для побудови класифікатора з використанням методу опорних векторів. Чорними кружечками обведено опорні вектори, синіми – нові об'єкти, які відносяться до різних класів і тому знаходяться по різні сторони від прямої, яка розмежовує об'єкти двох класів.

8.3.4. Класифікація шляхом побудови дерева рішень та випадкового лісу

Завдання 4. Розв'язати задачу класифікації ірисів шляхом побудови дерева рішень.

1. Побудуємо дерево рішень для задачі класифікації ірисів із допомогою функції ***fitctree()***, ввівши команду:

```
>> T = fitctree(meas,species)
```

У вікні *Command Window* буде виведена інформація про параметри дерева класифікації, яка міститься у змінній T (рис. 8.14).

2. Для більш детального виведення інформації про дерево рішень у вікні *Command Window* необхідно ввести команду:

```
>> view(T)
```

У вікні *Command Window* буде виведена більш детальна інформація про параметри побудованого дерева класифікації: вказано критерії розгалуження у вузлах дерева від його кореня до листів та класи, які відповідають листам (рис. 8.15).

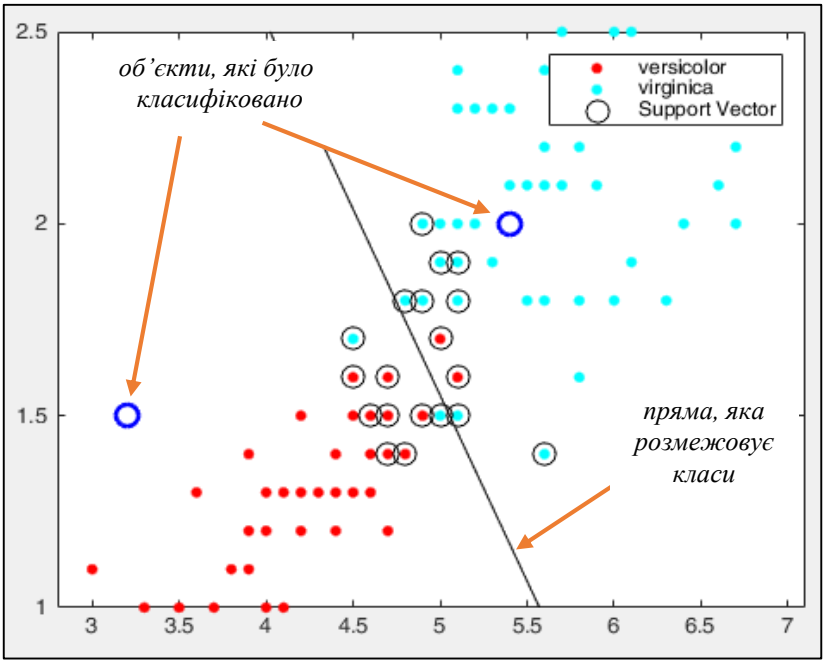


Рис. 8.13. Візуалізація побудови SVM-класифікатора та класифікації нових об'єктів методом опорних векторів

```
Command Window
>> T = fitctree(meas,species)
T =
  ClassificationTree
           PredictorNames: {'x1'  'x2'  'x3'  'x4'}
             ResponseName: 'Y'
    CategoricalPredictors: []
               ClassNames: {'setosa'  'versicolor'  'virginica'}
           ScoreTransform: 'none'
          NumObservations: 150
  Properties, Methods
fx >>
```

Рис. 8.14. Інформація про параметри дерева класифікації ірисів

```
Command Window
>> view(T)

Decision tree for classification
1  if x3<2.45 then node 2 elseif x3>=2.45 then node 3 else setosa
2  class = setosa
3  if x4<1.75 then node 4 elseif x4>=1.75 then node 5 else versicolor
4  if x3<4.95 then node 6 elseif x3>=4.95 then node 7 else versicolor
5  class = virginica
6  if x4<1.65 then node 8 elseif x4>=1.65 then node 9 else versicolor
7  class = virginica
8  class = versicolor
9  class = virginica

fx >>
```

Рис. 8.15. Детальна інформація про параметри побудованого дерева рішень

3. Для отримання візуального відображення побудованого дерева необхідно ввести команду:

```
>> view(T,'Mode','graph');
```

У окремому вікні буде виведено графічне зображення побудованого дерева (рис. 8.16):

a) у вершині дерева здійснюється перевірка змінної x_3;
b) у вузлі 2 знаходиться лист – клас setosa;
c) у вузлі 3 здійснюється перевірка змінної x_4;
d) у вузлі 4 здійснюється перевірка змінної x_3;
e) у вузлі 5 знаходиться лист – клас virginica;
f) у вузлі 6 здійснюється перевірка змінної x_4;
g) у вузлі 7 знаходиться лист – клас virginica;
h) у вузлі 8 знаходиться лист – клас versicolor;
i) у вузлі 9 знаходиться лист – клас virginica.

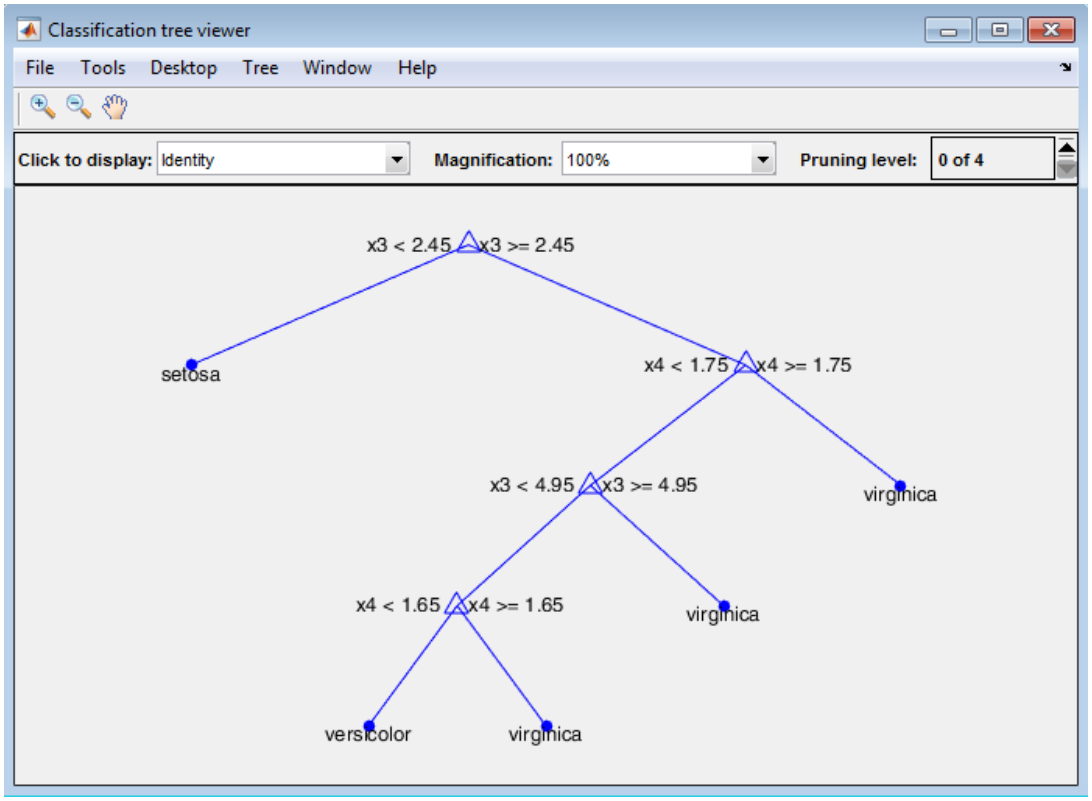
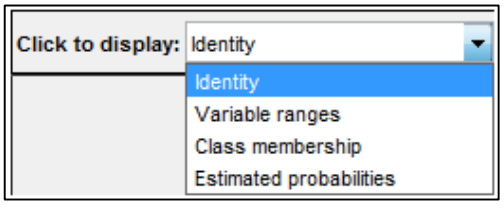


Рис. 8.16. Графічне зображення побудованого дерева рішень

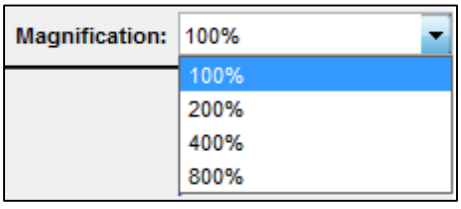
4. Вікно *Classification tree viewer* під рядком меню містить списки, що розкриваються, які дозволяють відображати різні характеристики дерева:

1) список *Click to display* (рис. 8.17, *а*) дозволяє змінювати інформацію, яка виводиться при клацанні лівою кнопкою миші на елементи дерева (табл. 8.11);

2) список *Manification* дозволяє змінювати відсоток графічного відображення дерева у вікні (рис. 8.17, *б*).



а) список *Click to display*

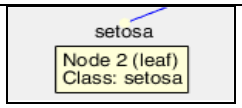
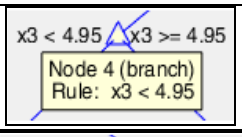
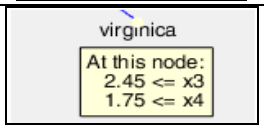
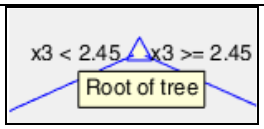
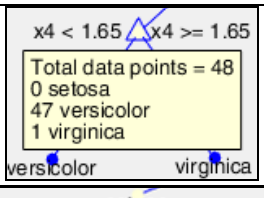
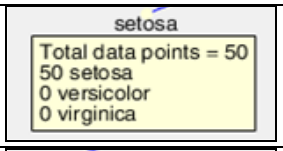
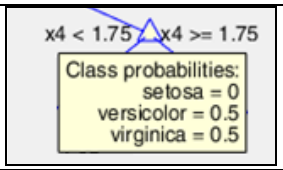
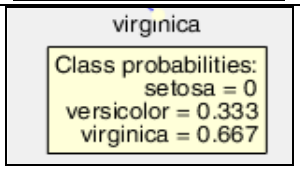


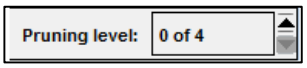
б) список *Manification*

Рис. 8.17. Список *Click to display*

Таблиця 8.11

Призначення команд списку *Click to display* вікна *Classification tree viewer*

Команда	Графічне зображення	Інформація, яка виводиться
identify		Номер гілки, назва класу
		Номер гілки, правило у вузлі
variable ranges		Правило
		Корінь дерева
class membership		Кількість об’єктів кожного класу у вузлі
		Кількість об’єктів кожного класу у листі
Estimated probabilities		Частка елементів кожного класу у вузлі
		Частка елементів кожного класу у листі

5. Перемикач  дозволяє відсікати гілки дерева у випадку його сильного розгалуження. Відсікати гілки можна також із допомогою команди *prune()*. Відсічемо 2 рівня гілок дерева та виведемо обрізане дерево спочатку з використанням перемикача у вікні *Classification tree viewer* (рис. 8.18), а потім ввівши команди:

```
>> t1 = prune(T,'level',2)
>> view(t1,'Mode', 'graph');
```

6. Після побудови дерева рішень визначимо клас квітки із наступними значеннями незалежних змінних: (4,2; 2,9; 2,1; 1,5).

Класифікація за деревом рішень проводиться за допомогою функції ***predict()***. Для здійснення класифікації введемо у вікні *Command Window* команди:

```
>> newpoint = [4.2 2.9 2.1 1.5]; % задання  нової точки
>> y = predict(T, newpoint)  % визначення класу нового об’єкту
```

Результат класифікації за побудованим деревом рішень буде виведено на екран – це клас *setosa* (рис. 8.19).

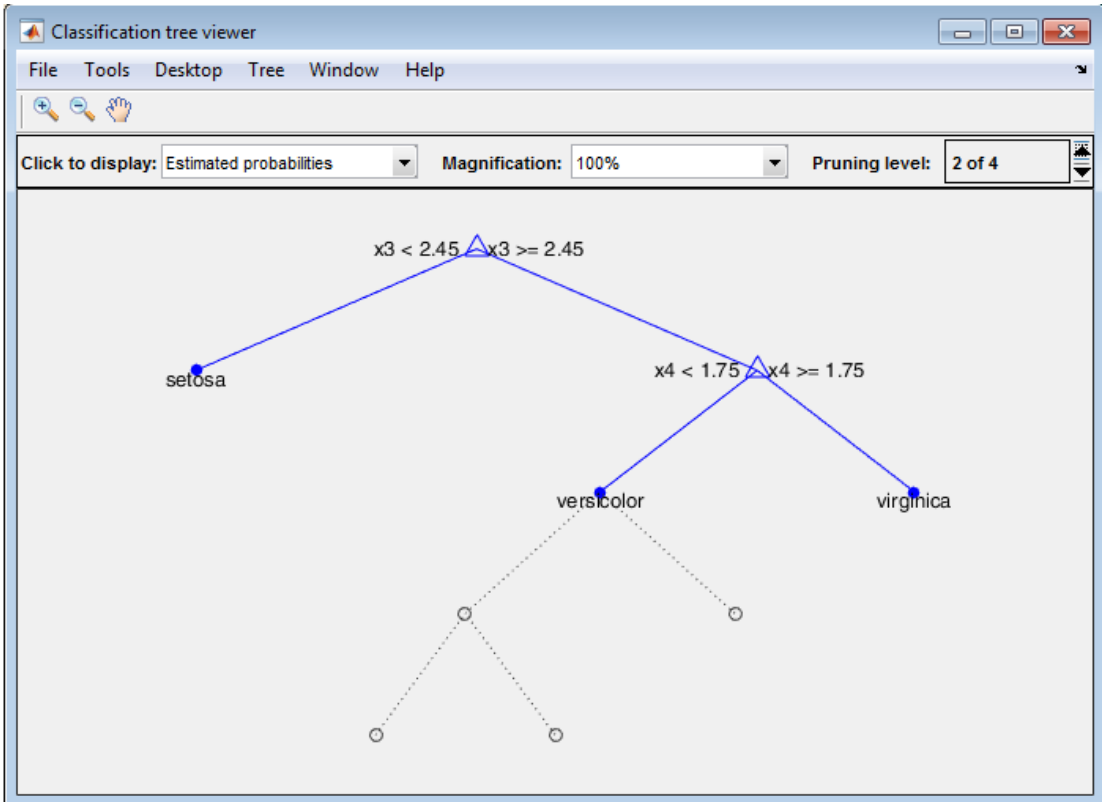


Рис. 8.18. Графічне зображення дерева з відсіченими гілками

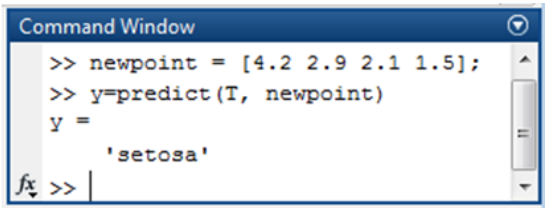


Рис. 8.19. Класифікація нового об'єкту за побудованим деревом рішень

Завдання 5. Розв'язати задачу класифікації ірисів шляхом побудови випадкового лісу.

Випадковий ліс (англ. *Random forest*) – алгоритм класифікації, який полягає в отриманні ансамблю дерев рішень, кожне з яких навчається на різних підмножинах простору ознак набору даних, що обираються випадковим чином.

Кожне з дерев дає невисоку якість класифікації, однак за рахунок їх великої кількості результат є хорошим. Випадковість вибору підмножин дає можливість знизити корельованість між деревами та уникнути перенавчання.

1. Працюємо з набором даних класичної задачі класифікації квіток ірисів трьох класів: Iris Setosa, Iris Versicolour, Iris Virginica, який уже представлений у робочому середовищі *Workspace* MatLab двома матрицями: *means* та *species*.

2. На основі навчання вхідного набору даних у процесі побудови класифікатора ми можемо сформувати ліс дерев (випадковий ліс) із використанням функції ***TreeBagger()***. Вводимо команду для формування випадкового тренувального лісу 50 дерев рішень:

```
>> TR = TreeBagger(50,meas,species)
```

3. У вікні *Command Window* буде виведена інформація про тренувальний набір дерев, яка міститься у змінній TR (рис. 8.20).

3. Візуалізуємо 20-те дерево з набору, ввівши команди:

```
>> Tree20 = TR.Trees{20};
>> view(Tree20, 'Mode', 'graph');
```

4. У вікні *Classification tree viewer* буде виведено графічне зображення обраного 20-го дерева (рис. 8.21).

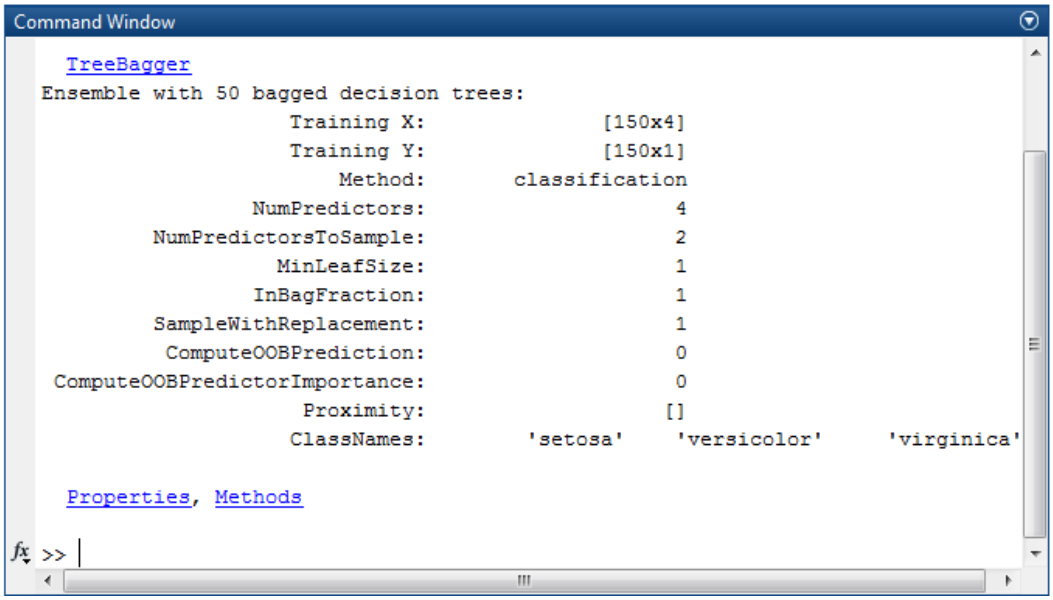


Рис. 8.20. Інформація про ліс класифікаційних дерев

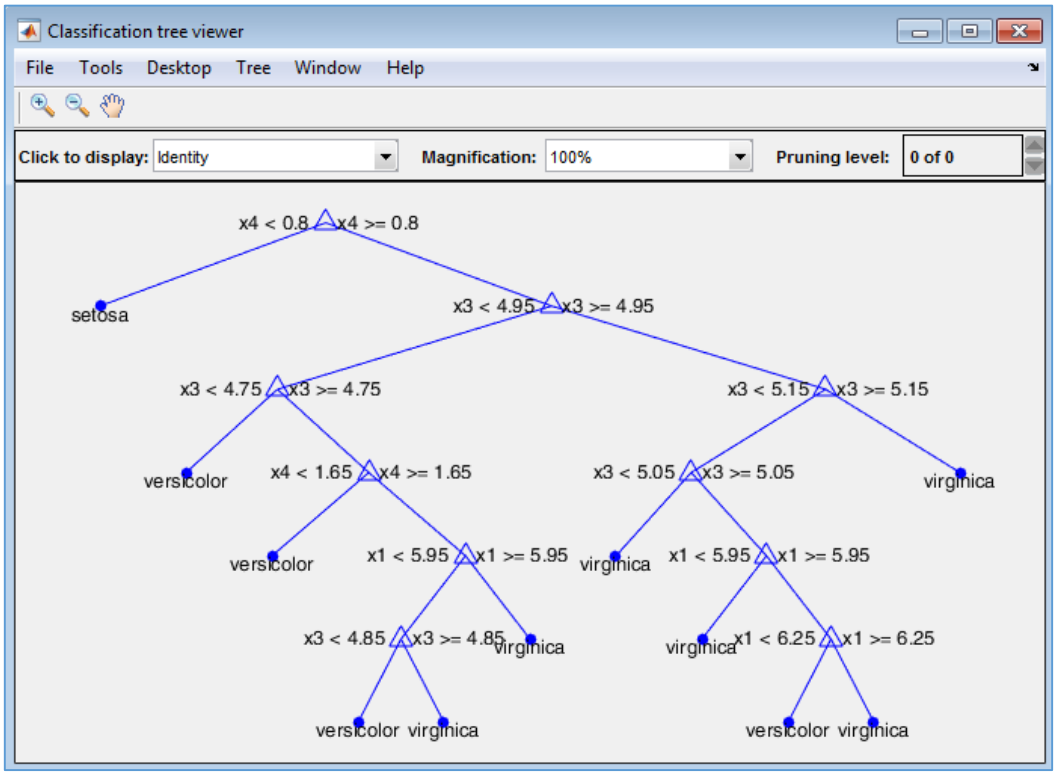


Рис. 8.21. Графічне зображення 20-го дерева лісу класифікаційних дерев

5. Після побудови випадкового лісу дерев визначимо клас квітки із такими значеннями незалежних змінних: (4,3; 2,9; 2,2; 1,4).

```
>> newpoint = [4.3 2.9 2.2 1.4]; % задання  нової точки
>> y = predict(TR, newpoint) % визначення класу нового об’єкту
```

Результат класифікації на побудованим лісом дерев буде виведено на екран – це клас *versicolor* (рис. 8.22).

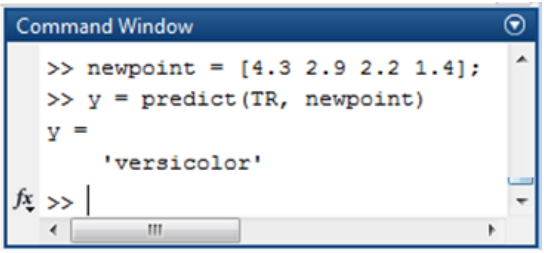


Рис. 8.22. Класифікація нового об’єкта за побудованим лісом дерев

8.4. ЗАВДАННЯ ДЛЯ САМОСТІЙНОЇ РОБОТИ

Завдання 6. Написати програму MatLab, яка вирішує наступні завдання:

1. Здійснює побудову класифікатора для набору даних класичної задачі класифікації ірисів Фішера з використанням:

a) методу k-найближчих сусідів (за 2-ма ознаками);

b) методу опорних векторів (за 2-ма ознаками для 2-х класів);

c) дерева рішень (за 4-ма ознаками);

d) випадкового лісу (за 4-ма ознаками).

2. Проводить класифікацію нових екземплярів кожним із побудованих класифікаторів, використовуючи дані, представлені за варіантами у таблиці 8.12, де:

a) P1 і P2 – координати нових точок, які необхідно класифікувати;

b) n і m – номери ознак, відібраних для класифікації (для методів k-a) найближчих сусідів та опорних векторів);

c) k – кількість найближчих сусідів (для методу k-найближчих сусідів).

3. Здійснює візуалізацію результатів класифікації за кожним із методів.

Вхідні дані представлені у файлі *fisheriris.mat*.

Завдання 7. Визначити за допомогою методу Байєса, чи відбудеться гра у футбол за погодних умов, заданих у таблиці 8.10, на підставі інформації про проведення ігор за різних погодних умов у минулому, представленими у таблиці, утвореній після видалення з таблиці 8.2 рядків *n* та *m*.

Значення *n* і *m* за варіантами представлені у таблиці 8.13.

КОНТРОЛЬНІ ПИТАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ № 8

1. Постановка задачі класифікації.
2. Основні групи методів розв’язання задачі класифікації.
3. Яку структуру мають правила класифікації?
4. У чому полягає метод 1R? Якими є його етапи?
5. Формула Байєса для визначення апостеріорної ймовірності та її застосування при розв’язанні задачі класифікації.
6. Опишіть основні етапи методу Naive Bayes.
7. Структура дерева рішень, основні етапи побудови.
8. Побудова дерева рішень та класифікація нових об’єктів за деревом рішень і шляхом побудови випадкового лісу засобами MatLab.
9. Яким чином розв’язують задачу класифікації методом k найближчих сусідів?
10. Здійснення класифікації методом k-найближчих сусідів у середовищі MatLab.
11. У чому полягає метод опорних векторів?
12. Здійснення класифікації методом опорних векторів засобами MatLab.

Таблиця 8.12

Дані за варіантами для виконання завдання 6

(*клас вилучати тільки для методу опорних векторів*)

Варіант	n	m	k	P1				P2			
				x_1	x_2	x_3	x_4	x_1	x_2	x_3	x_4
1	1	4	8	4,69	2,58	1,73	0,14	5,38	2,90	4,24	1,60
				Вилучення елементів класу				virginica			
2	2	3	7	4,37	3,24	1,47	0,54	5,37	3,57	6,79	1,75
				Вилучення елементів класу				virginica			
3	1	2	9	6,54	3,33	4,37	1,53	6,82	3,33	6,85	2,12
				Вилучення елементів класу				virginica			
4	1	3	12	5,69	3,00	1,29	0,45	5,44	2,34	3,72	1,19
				Вилучення елементів класу				virginica			
5	2	4	10	5,60	2,44	1,06	0,40	5,39	3,74	6,23	1,93
				Вилучення елементів класу				versicolor			
6	1	3	11	6,84	2,05	4,08	1,60	5,28	3,02	6,87	2,34
				Вилучення елементів класу				setosa			
7	2	4	12	5,07	3,33	1,45	0,47	5,73	3,03	3,19	1,39
				Вилучення елементів класу				virginica			
8	1	3	8	5,20	3,55	1,49	0,44	5,73	2,98	5,13	2,46
				Вилучення елементів класу				versicolor			
9	2	4	7	6,70	2,74	4,07	1,48	5,57	2,99	4,75	1,52
				Вилучення елементів класу				setosa			
10	1	3	9	4,65	3,41	1,86	0,38	6,47	2,84	3,95	1,22
				Вилучення елементів класу				virginica			
11	2	4	11	4,74	3,38	1,17	0,38	7,13	2,28	4,98	1,69
				Вилучення елементів класу				versicolor			
12	1	3	9	6,45	2,72	4,40	1,44	7,83	3,06	6,15	1,83
				Вилучення елементів класу				setosa			
13	1	4	12	5,26	4,36	1,44	0,14	5,58	3,34	4,38	1,29
				Вилучення елементів класу				virginica			
14	1	2	13	4,30	3,42	1,20	0,31	6,23	2,47	4,87	1,43
				Вилучення елементів класу				versicolor			
15	2	3	7	6,59	3,00	3,46	1,58	5,26	3,52	4,92	1,74
				Вилучення елементів класу				setosa			
16	3	4	8	4,74	3,93	1,78	0,17	5,90	3,15	4,96	1,36
				Вилучення елементів класу				virginica			
17	2	4	9	4,42	3,65	1,43	0,20	5,90	3,34	6,24	1,73
				Вилучення елементів класу				versicolor			
18	1	3	13	6,68	2,19	3,07	1,52	5,46	3,47	6,23	1,60
				Вилучення елементів класу				setosa			
19	1	2	11	4,84	4,25	1,71	0,25	6,41	2,37	3,09	1,69
				Вилучення елементів класу				virginica			
20	3	4	10	5,37	4,30	1,34	0,58	5,96	2,33	4,70	1,91
				Вилучення елементів класу				versicolor			
21	3	4	12	4,90	2,33	4,47	1,45	7,24	2,61	4,54	2,49
				Вилучення елементів класу				setosa			
22	1	2	8	5,02	3,23	1,08	0,25	5,53	3,32	4,59	1,64
				Вилучення елементів класу				virginica			
23	2	4	9	4,92	4,00	1,67	0,27	5,98	2,26	5,14	1,76
				Вилучення елементів класу				versicolor			
24	1	3	10	5,76	2,18	4,07	1,74	6,02	3,58	6,75	1,59
				Вилучення елементів класу				setosa			
25	2	4	11	4,70	3,99	1,65	0,55	6,08	2,68	3,84	1,71

Таблиця 8.13

Значення *n* та *m* за варіантами для виконання завдання 7

Варіант	n	m	Варіант	n	m
1	1	14	14	13	14
2	1	2	15	1	6
3	2	3	16	2	7
4	4	10	17	3	8
5	3	5	18	4	9
6	6	11	19	5	10
7	4	7	20	6	11
8	8	12	21	7	12
9	6	9	22	8	13
10	7	10	23	9	14
11	8	11	24	2	10
12	9	12	25	3	11
13	5	13	26	4	12

9. ПОШУК АСОЦІАТИВНИХ ПРАВИЛ. АЛГОРИТМ APRIORI

Лабораторна робота № 9

Мета: формування та закріплення знань про асоціативні правила й їх оцінки, алгоритми пошуку асоціативних правил. Набуття навичок пошуку асоціативних правил.

Теоретичні знання: поняття асоціативних правил. Постановка задачі пошуку асоціативних правил. Оцінки асоціативних правил: підтримка, достовірність, ліфт, леверідж, поліпшення. Алгоритми пошуку асоціативних правил. Етапи алгоритму Apriori.

9.1. АСОЦІАТИВНІ ПРАВИЛА: ОСНОВНІ ПОНЯТТЯ

9.1.1. Сутність асоціативних правил. Постановка задачі пошуку асоціативних правил

Алгоритми пошуку асоціативних правил дозволяють знаходити між подіями, які відбуваються спільно, приховані закономірності – правила кількісного опису їх взаємозв'язку (асоціації).

Уперше задачу пошуку асоціативних правил було застосовано для визначення типової поведінки покупців під час покупок у супермаркетах. Це обумовило специфіку впроваджуваної термінології. Було введено поняття ***ринкового кошику*** – набору товарів, які один покупець відбирає у супермаркеті, здійснюючи одну покупку.

Аналіз ринкового кошика (англ. *market basket analysis*) – це аналіз усіх наборів товарів, які було сформовано на основі покупок усіх відвідувачів супермаркета. Серед цих наборів є такі комбінації товарів, які зустрічаються набагато частіше інших і можуть бути віднесені до закономірних – типових шаблонів покупок. Наприклад, може бути виявлено, що 75% покупок, які містять молоко, також одночасно містять і хліб. Асоціативні правила дозволяють виявляти й кількісно описувати такі збіги.

Сьогодні пошук асоціативних правил широко застосовується в різних сферах життєдіяльності: для визначення профілю відвідувачів вебресурсів у рекомендаційних системах, для дослідження ефективності впроваджуваних ліків, для розв'язання задач медичної та технічної діагностики і прогнозування, інформаційної безпеки тощо.

До *базових понять* теорії асоціативних правил відносять поняття ***транзакції*** (англ. *transaction*) – деякої множини подій, що відбуваються спільно.

Предметний набір – не порожня множина об'єктів (подій), що з'явилися в одній транзакції.

Асоціативні правила є механізмом знаходження логічних закономірностей між пов'язаними елементами (подіями або об'єктами) на основі аналізу інформації, яка присутня у транзакціях набору даних.

Формально задачу пошуку асоціативних можна описати наступним чином. Нехай:

$I = \{i_1, i_2, \dots, i_n\}$ – множина об'єктів набору даних, які називаються ***елементами*** (предметами, товарами), де n – кількість об'єктів;

$D = \{T_1, T_2, \dots, T_m\}$ – множина транзакцій, яку називають ***базою даних,*** де m – кількість транзакцій.

Кожна ***транзакція*** $T_j = \{i_k | i_k \in I\}$ містить деякий довільний набір елементів множини I.

Транзакцію доцільно представити у вигляді бінарного вектора:

$$T_j = (t_{j1}, t_{j2}, \dots, t_{jk}, \dots, t_{jn}),$$

де $t_{jk} = 1$, якщо елемент i_k присутній у транзакції T_j, $t_{jk} = 0$, якщо елемент i_k відсутній у транзакції T_j.

Асоціативним правилом (англ. *Association Rules*) називається імплікація $X \to Y$, де X та Y є наборами елементів множини I: $X \subset I, Y \subset I$, серед яких немає однакових елементів $X \cap Y = \emptyset$.

Формулюють асоціативне правило $X \to Y$ так:

«*якщо* X, *то* Y» або «*якщо* ***умова***, *то* ***наслідок***»,

де ***умова*** або ***антецедент*** (англ. *Antecedent*) та ***наслідок*** або ***консеквент*** (англ. *Consequent*) є наборами елементів X та Y однієї й тієї ж множини I.

Умову (антецедент) правила називають також ***тілом правила*** (англ. *Rule body*).

Наслідок (консеквент) правила називають ***заголовком правила*** (англ. *Rule head*).

Загальну кількість елементів множини I, які входять до умови та наслідку асоціативного правила, називають ***довжиною правила***.

Пошук асоціативних правил полягає у здійсненні аналізу множини транзакцій T, які мітяться у базі даних D, із метою виявлення наступних залежностей: якщо у транзакції зустрічається набір елементів X, то на основі цього можна зробити висновок, що набір елементів Y з високою ймовірністю також повинен бути у цій транзакції.

9.1.2. Оцінки асоціативних правил

У процесі пошуку асоціативних правил знаходять такі правила, які часто зустрічаються – мають високу ймовірність. Головною проблемою при здійсненні пошуку є велика кількість правил під час дослідження великих наборів даних. Використання оцінок асоціативних правил дозволяє фільтрувати знайдені правила й у подальшому аналізувати тільки ті, оцінки яких задовольняють задані граничні значення.

Для кожного елемента i_k множини I можна розрахувати ***підтримку*** $S(i_k)$ як відношення кількості транзакцій у базі даних D, які містять елемент i_k до загальної кількості транзакцій у ній. Тобто, підтримка $S(i_k)$ – це ймовірність появи елемента i_k у транзакціях бази даних D: $S(i_k) = P(i_k)$.

Підтримка двоелементних наборів множини I– $S(i_k \cup i_m)$ буде рівна відношенню кількості транзакцій у базі даних D, які містять пару елементів i_k та i_m до загальної кількості транзакцій у ній. Тобто, підтримка $S(i_k \cup i_m)$ – це ймовірності появи елементів i_k та i_m у транзакціях бази даних D одночасно: $S(i_k \cup i_m) = P(i_k \cup i_m)$.

Аналогічно може бути розрахована підтримка наборів із більшою кількістю елементів множини I.

До основних ***об'єктивних оцінок*** асоціативних правил відносять підтримку S та достовірність C.

Для асоціативного правила $X \to Y$ позначимо:

$P(X)$ – ймовірність появи у транзакціях бази даних D умови X;

$P(Y)$ – ймовірність появи у транзакціях бази даних D наслідку Y;

$P(X \cup Y)$ – ймовірність появи у транзакціях бази даних D умови X і наслідку Yодночасно.

Підтримка $S(X \to Y)$ (англ. *Support*) асоціативного правила $X \to Y$ – це відношення кількості транзакцій k, які містять умову і наслідок, до загальної кількості транзакцій m:

$$S(X \to Y) = P(X \cup Y). \tag{9.1}$$

Правило $X \to Y$ має підтримку S, якщо S% транзакцій з D містять $X \cup Y$.

Достовірність $C(X \to Y)$ (англ. *Confidence*) асоціативного правила $X \to Y$ є його мірою точності й визначається як відношення кількості транзакцій k, що містять умову і наслідок, до кількості транзакцій l, що містять тільки умову. Отже, достовірність асоціативного правила може бути розрахована за формулою:

$$C(X \to Y) = \frac{S(X \cup Y)}{S(X)}. \tag{9.2}$$

Правило $X \to Y$ справедливе з достовірністю C, якщо C% транзакцій з D, які містять X, містять також Y.

До ***сильних правил*** відносять ті, для яких значення підтримки та достовірності перевищують певні задані користувачем порогові значення.

Крім підтримки та достовірності є ***суб'єктивні оцінки*** асоціативних правил, використання яких покращує розуміння виявлених закономірностей. До таких оцінок відносять ліфт L, леверідж Lv та поліпшення Im.

Серед виявлених асоціативних правил, які мають достовірність, вищу за порогове значення, можуть бути такі, що містять наслідок, який не є частим набором елементів. Це може приводити до створення хибних наборів правил. Для вирішення цієї проблеми розраховують ліфт асоціативного правила.

Ліфт $L(X \to Y)$ (англ. *Lift*) асоціативного правила $X \to Y$ є відношенням достовірності правила до його очікуваної достовірності, яка визначається як частота появи наслідку в цілому (підтримка наслідку цього правила):

$$L(X \to Y) = \frac{C(X \to Y)}{S(Y)}. \tag{9.3}$$

Ліфт може приймати значення від 0 до ∞:

1) значення $L(X \to Y)$, більші за 1, є значимими й свідчать про наявність позитивного зв'язку – умова X частіше зустрічається у транзакціях, які містять наслідок Y, ніж у інших транзакціях;
2) значення $L(X \to Y)$, близькі до 1, свідчать про відсутність зв'язку – умова X і наслідок Y зустрічаються у транзакціях із однаковою частотою як окремо, так і разом;
3) значення $L(X \to Y)$ близькі до нуля, свідчать про наявність негативного зв'язку – умова X частіше зустрічаються у транзакціях, які не містять наслідок Y, ніж у транзакціях, які його містять.

Леверідж $Lv(X \to Y)$ (англ. *Leverage*) асоціативного правила $X \to Y$ – це різниця між спостережуваною частотою, з якою умова й наслідок з'являються спільно та добутком частот появи умови й наслідку окремо. Розрахувати леверідж можна як різницю між підтримкою умови й наслідку разом та добутком підтримок умови й наслідку окремо:

$$Lv(X \to Y) = S(X \cup Y) - S(X) \cdot S(Y). \tag{9.4}$$

Леверідж дозволяє аналізувати ситуації, коли достовірність і ліфт правил ідентичні, але їх значимості відрізняються: більше значення леверідж свідчить про те, що правило зустрічається частіше й є більш значимим.

Поліпшення $Im(X \to Y)$ (англ. *Improvement*) асоціативного правила $X \to Y$ – це відношення частоти, з якою умова й наслідок з'являються спільно, до добутку частот появи умови й наслідку окремо. Розрахувують поліпшення асоціативного правила за формулою:

$$\text{Im}(X \to Y) = \frac{P(X \cup Y)}{P(X) \cdot P(Y)}. \tag{9.5}$$

Поліпшення показує, наскільки правило забезпечує правильний прогноз краще, ніж випадкове вгадування.

Усі правила з $Im(X \to Y) \leq 1$ не є значимими. Якщо $Im(X \to Y) > 1$, то ймовірність передбачення наслідку Y за правилом буде більшою за випадкове вгадування.

9.1.3. Алгоритми пошуку асоціативних правил. Алгоритм Apriori

До алгоритмів пошуку асоціативних правил відносять алгоритми AIS, SETM, PARTITION, DIC та алгоритм Apriori і його модифікації AprioriTid, AprioriHybrid, DHP.

Найпростіший алгоритм пошуку асоціативних правил розглядає всі можливі комбінації умов і наслідків, оцінює для них підтримку й достовірність, а потім виключає всі асоціації, які не задовольняють задані обмеження.

Однак число можливих асоціацій зі збільшенням елементів набору даних зростає експоненційно. Виявлення всіх асоціацій, підтримка й достовірність яких перевищують заданий мінімум, є задачею з високою обчислювальною трудомісткістю. Тому для ***скорочення простору*** можливих рішень застосовують спеціальні методи, одним із яких є обмеження кількості елементів, які може містити асоціативне правило.

Найбільш поширеним методом скорочення простору пошуку є ***виявлення частих наборів***.

Генерація асоціативного правила розбивається на два кроки:

1) задається ***мінімальний поріг підтримки*** S_{min}, який використовується для пошуку всіх частих наборів елементів у базі даних, яка містить транзакції;
2) задається ***обмеження мінімальної достовірності*** C_{min}, яке у процесі формування правила застосовується саме для цих частих наборів.

Процедура ***алгоритму Apriori*** з визначення наборів, які часто зустрічаються, є ітераційною. ***Основні етапи*** алгоритму Apriori є наступними:

1. Привласнити $k = 1$ і виконати відбір усіх 1-елементних наборів, у яких підтримка більша мінімально заданої користувачем S_{min} – сформувати множину L_1.
2. Привласнити $k = k + 1$ та здійснити ***формування кандидатів*** (англ. *candidate generation*): створити множину k-елементних наборів кандидатів у часті набори із відібраних на попередньому етапі (k-1)-елементних частих наборів. Якщо не вдається створити k-елементні набори, то виявлення частих наборів завершується – необхідно перейти до кроку 5, інакше виконується наступний крок.
3. Здійснити підрахунок ***підтримки кандидатів*** (англ. *candidate counting*): формування множини k-елементних частих наборів L_k шляхом відбору всіх k-елементних наборів, у яких підтримка більша мінімально заданої користувачем S_{min} .
4. Повернення до кроку 2.
5. Об'єднати усі множини L_k в одну множину: $\{L_1, L_2, \ldots, L_k\}$, яка буде містити усі часті набори, що задовольняють задані граничні умови для підтримки – це є результатом роботи алгоритму.

Після знаходження за допомогою алгоритму Apriori частих наборів необхідно на отриманій множині наборів згенерувати правила та визначити для кожного з них достовірність. Із множини згенерованих правил необхідно відібрати такі, достовірність яких буде більшою за мінімально задану користувачем C_{min}.

На цьому процес генерації асоціативних правил буде завершено.

9.2. РОЗВ'ЯЗАННЯ ПРАКТИЧНИХ ЗАДАЧ З ПОШУКУ АСОЦІАТИВНИХ ПРАВИЛ

9.2.1. Визначення оцінок асоціативних правил

Приклад 1. Для набору транзакцій, заданих у таблиці 9.1, здійснити оцінку правил, визначивши підтримку, достовірність, ліфт, леверідж та поліпшення.

1. Визначимо підтримку та достовірність правил *цукерки → помідори* та *салат → помідори*. Розглянувши правило *салат → помідори* для даних, представлених у таблиці 9.1, за формулами 9.1 та 9.2 маємо:

S(*салат → помідори*) = 4/10 = 0,4 C(*салат → помідори*) = 4/4 = 1

Дане правило зустрічається у 40% транзакцій, тому його підтримка S = 0,4.

У всіх випадках, коли покупець купує салат, він купує й помідори, тому достовірність правила C = 1.

2. Розглянувши асоціацію *цукерки → помідори* для даних, представлених у таблиці 9.1, отримуємо S(*цукерки → помідори*) = 4/10 = 0,4. Однак достовірність буде меншою за достовірність попереднього правила: C(*цукерки → помідори*) = 4/6 = 0,67.

Таблиця 9.1

Набір транзакцій

№	Транзакція
1	Сливи, салат, помідори
2	Селера, цукерки
3	Цукерки
4	Яблука, морква, помідори, картопля, цукерки
5	Яблука, апельсини, салат, цукерки, помідори
6	Персики, апельсини, селера, помідори
7	Квасоля, салат, помідори
8	Апельсини, салат, морква, помідори, цукерки
9	Яблука, банани, сливи, морква, помідори, цибуля, цукерки
10	Яблука, картопля

3. Визначимо ліфт для правил *помідори → салат* та *помідори → цукерки*. З таблиці 9.1 для цих правил маємо однакову підтримку та достовірність:

S(*помідори → салат*) = 4/10 = 0,4 C(*помідори → салат*) = 4/7 = 0,57

S(*помідори → цукерки*) = 4/10 = 0,4 C(*помідори → цукерки*) = 4/7 = 0,57

Проте після обчислення ліфта за формулою 9.3 для кожного з правил виявляємо значно сильніший зв'язок для умови та наслідку у правила (*помідори → салат*):

P(*салат*) = 4/10 = 0,4 L(*помідори → салат*) = 0,57/0,4 = 1,425

S(*цукерки*) = 6/10 = 0,6 L(*помідори → цукерки*) = 0,57/0,6 = 0,95

У правила *помідори → цукерки* обчислене значення ліфта близьке до одиниці, тому можемо стверджувати, зв'язок між умовою та наслідком для цього правила майже відсутній.

4. Визначимо за формулою 9.4 леверідж правил *салат → помідори* та *морква → помідори*. З таблиці 9.1 маємо:

C(*салат → помідори*) = 1 L(*салат → помідори*) = 1/0,7 = 1,43

C(*морква → помідори*) = 1 L(*морква → помідори*) = 1/0,7 = 1,43

Як бачимо, ці правила мають однакові достовірність та ліфт. Проте леверідж правил буде різним:

Lv(*салат → помідори*) = 0,4 – 0,4*0,7 = 0,12

Lv(*морква → помідори*) = 0,3 - 0,3*0,7 = 0,09

Це свідчить про те, що правило *салат → помідори*, яке має більший леверідж, становить більший інтерес та є більш цінним. Це правило буде більш значимим, тому що воно зустрічається частіше – застосовується для більшого числа покупців.

5. Визначимо за формулою 9.5 поліпшення правил *салат → помідори* та *морква → помідори*:

Im(*салат → помідори*) = 0,4 /(0,4*0,7) = 1,43

Im(*морква → помідори*) = 0,3 /(0,3*0,7) = 1,43

Поліпшення для обох правил виявилося однаковим та більшим за 1, що свідчить про те, що ймовірність передбачення наслідку за кожним із цих правил буде більшою за випадкове вгадування.

9.2.2. Пошук асоціативних правил за допомогою алгоритму Apriori

Приклад 2. Для множини об'єктів I = {*шоколад*, *чіпси*, *кокоси*, *вода*, *пиво*, *горіхи*}, які є товарами (табл. 9.2), та множини D транзакцій (табл. 9.3) за допомогою алгоритму Apriori здійснити пошук асоціативних правил за порогої мінімальної підтримки S_{min} = 0,5 та мінімальній достовірності C_{min} = 0,75.

Таблиця 9.2

Прайс-лист товарів

Ідентифікатор	Назва товару	Ціна
0	Шоколад	80,00
1	Чіпси	32,00
2	Кокоси	40,00
3	Вода	24,00
4	Пиво	34,00
5	Горіхи	35,00

Таблиця 9.3

Набори товарів у транзакціях множини D

Номер транзакції	Номер товару	Найменування товару	Ціна, *грн*
0	1	Чіпси	32,00
0	3	Вода	24,00
0	4	Пиво	34,00
1	2	Кокоси	40,00
1	3	Вода	24,00
1	5	Горіхи	35,00
2	5	Горіхи	35,00
2	2	Кокоси	40,00
2	1	Чіпси	32,00
2	2	Кокоси	40,00
2	3	Вода	24,00
3	2	Кокоси	40,00
3	5	Горіхи	35,00
3	2	Кокоси	40,0

Задача знаходження асоціативних правил розбивається на дві підзадачі:

1. Знаходження усіх наборів елементів, які задовольняють мінімальному порогу підтримки S_{min} = 0,5.
2. Генерація правил із знайдених наборів елементів із достовірністю, яка задовольняє заданому порогу C_{min} = 0,75.

Число транзакцій у заданому наборі даних рівне 4-м.

1. Привласнюємо $k = 1$ та формуємо множину кандидатів M_1 – усіх можливих одноелементних наборів товарів, для кожного з яких розраховуємо підтримку S (табл. 9.4).

Таблиця 9.4

Множина M_1 одноелементних наборів товарів

№	Набір	Підтримка S
1	{0}	0/4 = 0
2	{1}	2/4 = 0,5
3	{2}	3/4 = 0,75
4	{3}	3/4 = 0,75
5	{4}	1/4 = 0,25
6	{5}	3/4 = 0,75

2. У множині M_1 заданій мінімальній підтримці S_{min} = 0,5 відповідають кандидати 2, 3, 4 та 6 із товарами 1, 2, 3 та 5 відповідно. Формуємо множину виявлених одноелементних частих наборів:

$$L_1 = \{\{1\},\{2\},\{3\},\{5\}\}.$$

Відсікаються кандидати 1 та 6 із товарами 0 і 4: $\{\{0\},\{4\}\}$.

3. Привласнюємо $k = 2$ та формуємо множину кандидатів M_2– усіх можливих двоелементних наборів товарів, для кожного з яких розраховуємо підтримку S (табл. 9.5).

Таблиця 9.5

Множина M_2 двоелементних наборів товарів

№	Набір	Підтримка S
1	{1,2}	0,25
2	{1,3}	0,5
3	{1,5}	0,25
4	{2,3}	0,5
5	{2,5}	0,75
6	{3,5}	0,5

4. У множині M_2 заданій мінімальній підтримці S_{min} = 0,5 відповідають кандидати 2, 4, 5 та 6 із товарами {1,3}, {2,3}, {2,5} та {3,5} відповідно. Формуємо множину виявлених двоелементних частих наборів:

$$L_2 = \{\{1,3\},\{2,3\},\{2,5\},\{3,5\}\}.$$

Відсікаються кандидати 1 та 3 : $\{\{1,2\},\{1,5\}\}$.

5. Привласнюємо $k = 3$ та формуємо множину кандидатів M_3 можливих трьохелементних наборів товарів, яка буде складатися з одного кандидата, для якого розраховуємо підтримку S (табл. 9.6).

Таблиця 9.6

Множина M_3 трьохелементних наборів товарів

№	Набір	Підтримка S
1	{2,3,5}	0,5

6. Підтримка, розрахована для кандидата множини M_3, задовольняє заданій мінімальній підтримці S_{min} = 0,5. На цьому кроці отримуємо, що множина виявлених трьохелементних частих наборів буде складатися із одного кандидата:

$$L_3 = \{\{2,3,5\}\}.$$

7. Оскільки 4-елементні набори створити уже не вдається, то алгоритм завершується. Результатом роботи алгоритму Apriori є множина частих наборів, серед яких буде здійснюватися пошук асоціативних правил:

$$L = L_1 \cup L_2 \cup L_3 = \{\{1\},\{2\},\{3\},\{5\},\{1,3\},\{2,3\},\{2,5\},\{3,5\},\{2,3,5\}\}.$$

8. На отриманій множині наборів можна згенерувати правила, представлені у таблиці 9.7. Для кожного з цих правил знаходимо достовірності.

Таблиця 9.7

Згенеровані асоціативні правила

№	Правило	Достовірність C
1	«*якщо* 1 *то* 3»	1
2	«*якщо* 2 *то* 3»	2/3=0,67
3	«*якщо* 2 *то* 5	1
4	«*якщо* 2 *то* 3 і 5»	2/3=0,67
5	«*якщо* 3 *то* 1»	2/3=0,67
6	«*якщо* 3 *то* 2»	2/3=0,67
7	«*якщо* 3 *то* 5»	2/3=0,67
8	«*якщо* 3 *то* 2 і 5»	2/3=0,67
9	«*якщо* 5 *то* 2»	1
10	«*якщо* 5 *то* 3»	2/3=0,67
11	«*якщо* 5 *то* 2 і 3»	2/3=0,67

9. Із згенерованих асоціативних правил залишаємо ті, достовірність яких не менша за $C_{\min}$ = 0,75. Цю умову задовольняють три правила:

правило 1: «*якщо* 1 *то* 3» або «*якщо* чіпси *то* вода»;
правило 2: «*якщо* 2 *то* 5» або «*якщо* кокоси *то* горіхи»;
правило 3: «*якщо* 5 *то* 2» або «*якщо* горіхи *то* кокоси».

9.3. ЗАВДАННЯ ДЛЯ САМОСТІЙНОЇ РОБОТИ

Завдання 1. Для множини об'єктів $I = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$, які є товарами, та множини транзакцій D за допомогою алгоритму Apriori здійснити пошук асоціативних правил при порогових мінімальній підтримці S_{min} =0,5 та мінімальній достовірності C_{min}= 0, 75. Вхідні дані – множина транзакцій D за варіантами представлені у таблиці 9.8.

Завдання 2. Для знайдених у завданні 1 асоціативних правил визначити ліфт, леверідж і поліпшення та здійснити інтерпретацію отриманих результатів.

Завдання 1 та 2 можуть бути виконані у середовищі MS Excel або за допомогою самостійно розробленої програми (мова програмування та середовище розробки – за власним вибором студента).

КОНТРОЛЬНІ ПИТАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ № 9

1. Що таке асоціативне правило? Яким є їх призначення?
2. Назвіть основні складові асоціативного правила.
3. Що таке довжина асоціативного правила?
4. Як визначаються підтримка та достовірність асоціативного правила?
5. Як визначаються ліфт, леверідж та поліпшення асоціативного правила?
6. Яке призначення алгоритмів пошуку асоціативних правил?
7. На які підзадачі розбивається задача знаходження асоціативних правил?
8. У чому полягає сутність алгоритму Apriori?
9. Опишіть послідовність етапів виконання алгоритму Apriori.
10. Як відбувається генерація асоціативних правил на основі частих наборів, знайдених за алгоритмом Apriori?

Таблиця 9.8

Вхідні дані за варіантами для виконання завдання 1

Варіант		Завдання								Варіант		Завдання							
		№	Елементи									№	Елементи						
1	Множина транзакцій D	1	0	1	3	4	7			2	Множина транзакцій D	1	0	1	2	5	7	9	
		2	0	1	2							2	1	2	7	8			
		3	0	3	4	5	6					3	0	5	9	8	9		
		4	0	1	2	3	4	5	7			4	1	2	4	5	7		
		5	1	2	3	4	5					5	0	1	2	5	9		
		6	1	3	4							6	0	1	2	9			
		7	3	7	9							7	0	3	4	5	8	9	2
		8	0	1	2	3	7					8	3	4	1	2	7	8	
		9	1	2	3	6	7					9	5	6	7				
		10	0	7	8							10	0	5	6	8	9		
3	Множина транзакцій D	№	Елементи							4	Множина транзакцій D	№	Елементи						
		1	0	1	3	5	7					1	0	1	2	5	7	9	
		2	0	1	2							2	1	2	7	8			
		3	0	3	4	5	6	7				3	0	5	9	8	9		
		4	0	1	2	3	4	5	8			4	1	2	4	5	7		
		5	1	2	4	5	7					5	0	1	5	9			
		6	1	3	9							6	0	1	2	9			
		7	3	7	9							7	0	3	4	8	9		
		8	0	1	2	3	7					8	3	4	1	2	7	8	
		9	1	2	3	6	7					9	5	6	7				
		10	1	7	8							10	0	5	6	8			

Варіант	Завдання								Варіант	Завдання								
		№	Елементи									№	Елементи							
5	Множина транзакцій D	1	0	1	4	9				6	Множина транзакцій D	1	0	9						
		2	0	1	2	3	8	9				2	1	2	7	8	5	4	3
		3	0	3								3	0	5	9	8				
		4	0	1								4	1	3	4	6	7			
		5	1	2	7	8	9					5	0	1	2	6	8			
		6	1	5	6							6	0	1	2	9				
		7	3	0	1	2	5	8	9			7	0	3	4	5	8	9		
		8	0	1	2	3	7	9				8	3	4	1	2	7	8		
		9	1	2	5	6	7					9	5	6	7	3	2			
		10	0	7	8	3						10	0	5	6	8	9			
7	Множина транзакцій D	№	Елементи							8	Множина транзакцій D	№	Елементи							
		1	0	2	3	6	8					1	0	1	3	4	7	9		
		2	0	5	7							2	1	3	5	8				
		3	0	2	3	5	6	7				3	0	1	7	8	9			
		4	0	1	2	7	9	5	8			4	1	3	4	6	7			
		5	1	3	4	6	7					5	0	4	5	8				
		6	1	2	8							6	0	1	3	7				
		7	2	7	8							7	0	2	4	7	9			
		8	0	1	2	5	9					8	6	3	1	2	7	8		
		9	1	2	4	5	7					9	1	6	7					
		10	1	6	8							10	2	5	7	8				
9	Множина транзакцій D	№	Елементи							10	Множина транзакцій D	№	Елементи							
		1	0	2	5	8	7					1	0	1	2	4	8	9		
		2	0	4	7							2	1	3	6	8				
		3	0	2	3	7	9					3	1	4	7	8	9			
		4	0	1	2	3	5	8	9			4	1	3	6	5	7			
		5	1	2	5	7	8					5	0	1	2	7	8			
		6	1	5	6							6	0	1	3	4				
		7	3	4	7							7	0	2	5	1	8	7		
		8	0	1	2	5	7					8	2	4	6	0	7	8		
		9	1	2	4	8	7					9	3	5	7					
11	Множина транзакцій D	№	Елементи							12	Множина транзакцій D	№	Елементи							
		1	0	2	3	8	7					1	0	1	2	1	8	9		
		2	0	4	2							2	1	3	7	8				
		3	1	2	4	5	8	7				3	0	4	7	8	9			
		4	0	1	2	9	4	5	8			4	1	3	4	5	8			
		5	1	2	7							5	0	1	4	9				
		6	1	0	3	5	7					6	0	1	2	8	9			
		7	0	1	4	8	7					7	1	3	9	2	7	8		
		8	1	4	3	8	7					8	2	6	8					
		9	1	4	6							9	1	3	6	8				
13	Множина транзакцій D	№	Елементи							14	Множина транзакцій D	№	Елементи							
		1	1	2	3	5	8					1	0	1	2	5	7	9		
		2	0	3	4							2	1	2	7	8				
		3	0	2	4	7	6					3	0	5	9	8	9			
		4	0	9	2	3	4	5	7			4	1	2	4	5	7			
		5	1	7	3	4	5					5	0	1	2	5	9			
		6	1	3	4							6	0	1	2	9				
		7	3	5	9	4						7	0	3	4	5	8	9		
		8	0	1	2	3	7	9				8	3	4	1	2	7	8		
		9	1	2	3	6	7					9	5	6	7					

Варіант		Завдання								Варіант		Завдання							
15	Множина транзакцій D	№	Елементи							16	Множина транзакцій D	№	Елементи						
		1	1	2	3	6	9					1	0	1	3	4	7	8	
		2	0	5	7	8						2	1	3	5	8			
		3	0	2	3	5	6					3	0	1	7	8	9		
		4	0	1	2	7	9					4	1	3	5	6	7		
		5	1	3	4	6	7	2	9			5	0	4	5	8			
		6	1	2	8	4	5					6	0	1	2	7			
		7	2	7	8	0	3					7	0	2	4	8	9		
		8	0	1	2	5	9					8	6	3	4	2	7	8	1
		9	1	2	4	5	7					9	2	5	7	9			
		10	0	2	4							10	0	2	4	9			
		11	0	7	8	5						11	1	5	6	8			
17	Множина транзакцій D	№	Елементи							18	Множина транзакцій D	№	Елементи						
		1	1	2	4	8	7					1	0	1	2	4			
		2	0	4	7	9						2	1	3	6	8			
		3	0	2	3	7	9	8				3	1	4	7	5	9		
		4	0	1	2	3	5					4	1	3	6	5	7		
		5	1	2	5	7	8	3				5	0	1	6	7	8		
		6	1	5	6	8						6	0	1	3	4	5		
		7	3	4	7							7	0	2	5	1	8		
		8	0	1	2							8	2	4	6	0	7		
		9	1	2	4	8	7	3				9	3	5	7				
		10	0	2	5							10	0	1	3	7			
		11	1	5	6							11	0	9	8				
19	Множина транзакцій D	№	Елементи							20	Множина транзакцій D	№	Елементи						
		1	0	1	3	5	7					1	0	1	2	5	7	9	
		2	0	1	2							2	1	2	7	8			
		3	0	3	4	5	6	7				3	0	5	9	8	9		
		4	0	1	2	3	4	5	8			4	1	2	4	5	7		
		5	1	2	4	5	7					5	0	1	5	9			
		6	1	3	9							6	0	1	2	9			
		7	3	7	9							7	0	3	4	8	9		
		8	0	1	2	3	7					8	3	4	1	2	7	8	
		9	1	2	3	6	7					9	5	6	7				
		10	1	7	8							10	0	5	6	8			
		11	0	7	8	4						11	0	4	7				
21	Множина транзакцій D	№	Елементи							22	Множина транзакцій D	№	Елементи						
		1	0	1	2	7						1	0	9	2	5			
		2	0	1	2	5	8	9				2	1	2	7	8	5		
		3	0	4								3	0	5	9	4			
		4	0	2								4	1	3	4	6	7		
		5	1	2	3	8	9					5	0	1	9	6	8		
		6	1	5	8							6	0	1	4	9			
		7	3	0	1	2	5	7	9			7	0	2	4	5	8	9	
		8	0	1	2	3	7	9				8	3	4	1	2	7	8	
		9	1	2	3	6	7					9	5	6	7	3	2		
		10	0	7	4	3						10	0	5	6	8	9		
		11	0	2	3	8						11	1	4	9				
		12	0	5								12	1	6	9				

Варіант		Завдання									Варіант		Завдання							
23	Множина транзакцій D	№	Елементи								24	Множина транзакцій D	№	Елементи						
		1	0	1	3	6	9						1	0	1	2	5	7	8	
		2	0	1	2	7							2	1	2	7	8			
		3	0	3	4	5	6	7	8	9			3	0	5	9	8	9		
		4	0	1	2	3	4						4	1	2	4	8	7		
		5	1	2	4	5	7						5	0	2	5	9			
		6	1	3	5								6	0	1	2	9			
		7	3	7	9	2							7	0	3	4	8	9		
		8	0	1	2	3	7	9					8	3	4	1	2	7	8	
		9	1	2	3	6	7	8					9	5	6	7				
		10	1	4	8								10	0	3	6	8			
25	Множина транзакцій D	№	Елементи								26	Множина транзакцій D	№	Елементи						
		1	0	1	4	6	7						1	0	1	2	5	7	9	
		2	0	2									2	1	2	7	8			
		3	0	3	4	5	6						3	0	5	9	8	9		
		4	0	1	2	3	4	5	7				4	1	2	4	5	7		
		5	1	2	3	4	5						5	0	1	2	5	9		
		6	1	3	4								6	0	1	2	9			
		7	3	7	9								7	0	3	4	5	8	9	
		8	0	1	2	3	7						8	3	4	1	2	7	8	
		9	1	2	3	6	7						9	5	6	7				
		10	0	7	8								10	0	5	6	8	9		

10. ЗАДАЧА ПРОГНОЗУВАННЯ. АНАЛІЗ ЧАСОВИХ РЯДІВ

Лабораторна робота № 10

Мета: формування та закріплення знань про сутність задачі прогнозування в Data Mining, основні поняття та характеристики часового ряду, аналіз часових рядів, основні етапи побудови моделі часового ряду. Набуття навичок проведення аналізу та прогнозування часового ряду з використанням засобів MS Excel та MatLab.

Теоретичні знання: задача прогнозування в Data Mining. Основні поняття та характеристики, аналіз часових рядів. Структура часового ряду. Основні етапи побудови моделі часового ряду, оцінка її точності та адекватності. Виявлення аномальних відхилень, наявності тенду, автокореляційних аналіз часового ряду. Згладжування часового ряду. Авторегресійний аналіз. Етапи побудови адитивної (мультиплікативної) моделі часового ряду.

10.1. ЧАСОВІ РЯДИ: АНАЛІЗ ТА ПРОГНОЗУВАННЯ

10.1.1. Задача прогнозування в Data Mining

Задача прогнозування є однією з найбільш поширених і водночас найбільш складних задач інтелектуального аналізу даних.

Прогнозування дозволяє зменшувати ризик прийняття неправильних, суб'єктивних рішень і є важливим елементом організації управління у різноманітних сферах життєдіяльності: науці, економіці, виробництві, бізнесі, web-аналітиці, освіті, медицині. Наприклад, прогнозування динаміки курсу валюти, фінансової стійкості підприємства, врожайності агрокультури, попиту на певний товар, темпів розповсюдження хвороби, трендів у цифровому бізнесі тощо.

Складність процесу прогнозування пов'язана з необхідністю аналізу й оцінювання великих обсягів даних, ускладненням методів, появою концептуально нових підходів до прогнозування процесів різної природи. Тому на сьогодні стан розвитку методів прогнозування тісно пов'язаний із розвитком сучасних напрямів галузі інформаційних технологій у області аналізу даних.

Прогнозом називають науково обґрунтований висновок про майбутній стан об'єкта на основі ретроспективних даних про нього. Як ***об'єкт прогнозування*** можуть виступати процеси, явища, події предметної області, у якій здійснюється прогностичний аналіз.

Прогнозування (англ. *Forecasting*) є процесом оцінки майбутніх значень змінних, які характеризують об'єкт прогнозування, на основі аналізу особливостей набору даних про цей об'єкт.

Розробка ***прогностичної моделі*** передбачає ретельний аналіз об'єкта прогнозування, вибір методу прогнозування та оцінку адекватності і точності прогнозу. Якщо розроблена модель є адекватною й забезпечує високу точність прогнозу, її можна використовувати для прогнозування значень змінних, які характеризують об'єкт прогнозування.

Прогнозування має спільні риси з задачами класифікації та регресії, тому що при їх розв'язанні також здійснюється прогноз – оцінка значень залежної змінної, яка характеризує об'єкт. Тому багато методів Data Mining, які застосовуються для розв'язання задач класифікації та регресії, використовують і для прогнозування. Пошук асоціативних правил також пов'язаний із прогнозом, оскільки знайдені асоціації є закономірностями, які проявляють себе у майбутньому.

Можна виділити такі ***види методів*** прогнозування.

1. *Статистичні методи:* регресійний аналіз, методи прогнозування часових рядів – авторегресії, експоненціального згладжування, ковзного середнього, екстраполяції трендів.
2. *Кібернетичні методи*: штучні нейронні мережі, генетичні алгоритми, дерева рішень, метод випадкового лісу, лісу з квантильною регресією, метод опорних векторів.

Різні методи розв'язання задачі прогнозування використовують різні способи визначення майбутніх значень змінних, які характеризують об'єкт прогнозування.

Стан об'єкта прогнозування характеризується набором змінних, значення яких може змінюватися з плином часу. Тому основою для прогнозування є інформація, яка зберігається у вигляді часових рядів.

10.1.2. Часовий ряд: основні поняття та характеристики

Часові ряди виникають у результаті впорядкованого за часом виміру деякої ознаки набору даних: характеристики технічних, природних, соціальних, економічних, екологічних, інформаційних та інших систем.

Часовий ряд (англ. *Time Series*) – це послідовність значень досліджуваної ознаки, впорядкованих у хронологічному порядку: $y_1, y_2, \ldots, y_t, \ldots, y_n$.

Кожне окреме значення ознаки y_t є показником, який називається ***рівнем часового ряду***, n – кількість рівнів ряду, яку називають ***довжиною часового ряду***.

Ми будемо дотримуватися наведеного вище означення довжини часового ряду. Проте зазначимо, що довжиною часового ряду можуть також називати час, що минув від першого до останнього моментів спостережень, відображених у рівнях часового ряду.

Таким чином, часовий ряд складається із двох елементів:

1) ***періоду часу*** *t,* за який або за станом на який приводяться числові значення;
2) ***рівнів ряду*** – числових значень ознаки y_t для кожного періоду часу.

Залежно від характеру рівнів ряду розрізняють такі ***види часових рядів***: моментні, інтервальні та похідні часові ряди.

Моментний часовий ряд – це часовий ряд, значення рівнів якого характеризують стан явища, що вивчається, на певний момент часу (табл. 10.1).

Таблиця 10.1

Динаміка зміни курсу долара USD до української гривні UAN, купівля
(*моментний часовий ряд*)

Дата	23.06.22	24.06.22	25.06.22	26.06.22	27.06.22	28.06.22
Курс НБУ, *грн*	34,1813	34,2557	34,1215	34,1215	34,2443	34,1791

Інтервальний часовий ряд – це часовий ряд, значення рівнів якого характеризують стан явища, що вивчається, за певні періоди часу (день, місяць, квартал і т.д.). Рівні інтервального ряду утворюють шляхом агрегування за певний проміжок часу (табл. 10.2).

Таблиця 10.2

Динаміка зміни пошукових запитів «*купити нетбук*» по Україні за даними Google Trends
(*інтервальний часовий ряд*)

Інтервал часу	Листопад 2021 р.	Грудень 2021 р.	Січень 2022 р.	Лютий 2022 р.	Березень 2022 р.	Квітень 2022 р.
Кількість запитів	227	233	177	185	43	80

Похідний часовий ряд – це часовий ряд, значення рівнів якого утворені середніми або відносними значеннями ознак за певні періоди часу (табл. 10.3).

Таблиця 10.3

Середня заробітна плата в Україні
(*похідний часовий ряд*)

Місяць року	Серпень 2021 р.	Вересень 2021 р.	Жовтень 2021 р.	Листопад 2021 р.	Грудень 2021 р.	Січень 2022 р.
Розмір зарплати, *грн*	13997	14239	14045	14282	17543	14577

Наведемо важливі для подальшого аналізу числові характеристики часового ряду y_t, що містить n рівнів: $y_1, y_2, \ldots, y_n$.

Статистичні оцінки математичного сподівання та дисперсії можна визначити за наведеними нижче формулами.

Математичне сподівання розраховують як середнє арифметичне рівнів ряду:

$$\bar{y} = \frac{1}{n}\sum_{t=1}^{n} y_t \,. \qquad (10.1)$$

Дисперсія:

а) зміщена оцінка: $$\sigma^2 = \frac{1}{n}\sum_{t=1}^{n} (y_t - \bar{y})^2 \,, \qquad (10.2)$$

б) незміщена оцінка: $$\sigma^2 = \frac{1}{n-1}\sum_{t=1}^{n}(y_t - \bar{y})^2 . \quad (10.3)$$

Рівні часового ряду не є статистично незалежними і не обов'язково є однаково розподіленими. Тому для аналізу часових рядів застосовують спеціальні методи дослідження, відмінні від методів аналізу статистичних вибірок.

Залежно від структури часового ряду значення кожного наступного рівня може залежати від попередніх значень. Кількісно таку залежність вимірюють, розраховуючи автоковаріацію та коефіцієнти автокореляції між рівнями аналізованого часового ряду та його рівнями, зсунутими на декілька періодів у часі.

Часовий лаг – це кількість періодів часу між значеннями тих рівнів часового ряду, залежність між якими досліджується.

Автоковаріація порядку p – це коваріація між значеннями рівнів часового ряду, зсунутими один відносно одного на p періодів часу:

$$c_p = \frac{\sum_{t=1}^{n-p}(y_t - \bar{y}_1)(y_{t+p} - \bar{y}_2)}{n-p}, \quad (10.4)$$

де $\bar{y}_1 = \frac{\sum_{t=1}^{n-p} y_t}{n-p}$, $\bar{y}_2 = \frac{\sum_{t=p+1}^{n} y_t}{n-p}$, p – часовий лаг.

Для достатньо великих n різницею між $\bar{y}_1$ та $\bar{y}_2$ можна знехтувати, припустивши, що $\bar{y}_1 = \bar{y}_2 = \bar{y}$. Тоді автоковаріацію порядку p визначають за формулою:

а) зміщена оцінка:

$$c_p = \frac{1}{n}\sum_{t=1}^{n-p}(y_t - \bar{y})(y_{t+p} - \bar{y}) = \frac{1}{n}\sum_{t=p+1}^{n}(y_t - \bar{y})(y_{t-p} - \bar{y}), \quad (10.5)$$

б) незміщена оцінка:

$$c_p = \frac{1}{n-p}\sum_{t=1}^{n-p}(y_t - \bar{y})(y_{t+p} - \bar{y}) = \frac{1}{n-p}\sum_{t=p+1}^{n}(y_t - \bar{y})(y_{t-p} - \bar{y}), \quad (10.6)$$

Автоковаріація порядку $p = 0$ дорівнює дисперсії часового ряду:

$$c_0 = \frac{1}{n}\sum_{t=1}^{n}(y_t - \bar{y})^2 . \quad (10.7)$$

Зі збільшенням лагу кількість пар значень, за якими розраховують автоковаріцію, зменшується. Доцільно для забезпечення достовірності дотримуватися правила – максимальний лаг не повинен бути більшим за $n/4$.

Значення автоковаріації залежить від одиниць вимірювання y_t. Тому для аналізу часових рядів зручніше користуватися безрозмірною величиною – коефіцієнтом автокореляції, який може приймати значення в інтервалі [-1, 1].

Автокореляція рівнів ряду – кореляційна залежність між послідовними рівнями часового ряду.

Коефіцієнт автокореляції r_p порядку p – характеризує тісноту зв'язку рівнів часового ряду, зсунутих один відносно одного на p періодів часу:

$$r_p = \frac{\sum_{t=1}^{n-p}(y_t - \bar{y}_1)(y_{t+p} - \bar{y}_2)}{\sqrt{\sum_{t=1}^{n-p}(y_t - \bar{y}_1)^2 \cdot \sum_{t=p+1}^{n}(y_{t+p} - \bar{y}_2)^2}}, \quad (10.8)$$

де $\bar{y}_1 = \frac{\sum_{t=1}^{n-p} y_t}{n-p}$, $\bar{y}_2 = \frac{\sum_{t=p+1}^{n} y_t}{n-p}$.

Для великих n припускаємо, що $\bar{y}_1 = \bar{y}_2 = \bar{y}$. Тоді коефіцієнт автокореляції визначають як відношення автоковаріацій порядку p та $p = 0$ за формулою:

$$r_p = \frac{c_p}{c_0} = \frac{\sum_{t=1}^{n-p}(y_t - \bar{y})(y_{t+p} - \bar{y})}{\sum_{t=1}^{n}(y_t - \bar{y})^2}. \qquad (10.9)$$

Чим ближчим до 1 або до -1 буде значення коефіцієнта автокореляції, тим тіснішим буде зв'язок між рівнями часового ряду. Значення коефіцієнта, близькі до нуля, говорять про відсутність зв'язку між рівнями ряду.

У часових рядів описані вище характеристики можуть змінюватися з плином часу або залишатися постійними. Тому часові ряди можна розділити на ***два основних види***:

1) *стаціонарні часові ряди* – ряди, значення значення рівнів яких коливаються навколо постійного середнього значення ознаки;

2) *нестаціонарні часові ряди* – ряди, значення рівнів яких коливаються навколо середнього значення ознаки, яке з плином часу змінюється під впливом різних факторів.

Стаціонарний часовий ряд – це часовий ряд, який має сталі й скінчені математичне сподівання і дисперсію для кожного періоду часу та сталу й скінчену автоковаріацію будь-якого порядку в усі моменти часу.

Приклад 1. Для часового ряду, представленого у таблиці 10.2, розрахувати автоковаріацію нульового і першого порядків та коефіцієнт автокореляції першого порядку.

Автоковаріацію часового ряду порядку $p = 0$ розраховуємо за формулою 10.7:

$$c_0 = \frac{1}{6}((227-157{,}5)^2 + (233-157{,}5)^2 + (177-157{,}5)^2 + (185-157{,}5)^2 + (43-157{,}5)^2 + (80-157{,}5)^2) = 5130{,}6.$$

Автоковаріацію часового ряду порядку $p = 1$ будемо розраховувати за формулою 10.4, оскільки часовий ряд має невелику довжину $n = 6$. Маємо:

$$\bar{y}_1 = \frac{1}{5}(227 + 233 + 177 + 185 + 43) = 173,\ \bar{y}_2 = \frac{1}{5}(233 + 177 + 185 + 43 + 80) = 143{,}6,$$

$$c_1 = \frac{1}{6-1}\sum_{t=1}^{6-1}(y_t - \bar{y}_1)(y_{t+1} - \bar{y}_2) = \frac{1}{5}((y_1 - \bar{y}_1)(y_2 - \bar{y}_2) + (y_2 - \bar{y}_1)(y_3 - \bar{y}_2) + (y_4 - \bar{y}_1)(y_5 - \bar{y}_2) +$$

$$+ (y_5 - \bar{y}_1)(y_6 - \bar{y}_2) = 2759{,}6.$$

Коефіцієнт автокореляції порядку $p = 1$ розраховуємо за формулою 10.8. Маємо:

$$\sum_{t=1}^{n-1}(y_t - \bar{y}_1)^2 = \frac{1}{5}((227-173)^2 + (233-173)^2 + (177-173)^2 + +(185-173)^2 + (43-173)^2) = 23576,$$

$$\sum_{t=1}^{n-1}(y_{t-1} - \bar{y}_2)^2 = \frac{1}{5}((233-143{,}6)^2 + (177-143{,}6)^2 + (185-143{,}6)^2 + +(43-143.6)^2 +$$

$$+ (80-143.6)^2) = 24736,$$

$$\sum_{t=1}^{n-1}(y_t - \bar{y}_1)(y_{t-1} - \bar{y}_2) = ((227-173)(233-143{,}6) + (233-173)(177-143{,}6) +$$

$$+ (177-173)(185-143{,}6) + (185-173)(43-143{,}6) + (43-173)(80-143{,}6) = 13798,$$

$$r_1 = \frac{\sum_{t=1}^{n-1}(y_t - \bar{y}_1)(y_{t+1} - \bar{y}_2)}{\sqrt{\sum_{t=1}^{n-1}(y_t - \bar{y}_1)^2 \cdot \sum_{t=p+1}^{n}(y_{t+1} - \bar{y}_2)^2}} = \frac{13798}{\sqrt{23576 \cdot 24736}} = 0{,}571.$$

10.1.3. Аналіз часових рядів

Аналіз часових рядів (англ. *Time Series Analysis*) – сукупність методів аналізу, призначених для виявлення структури часових рядів та їх прогнозування.

Виявлення структури часового ряду необхідне для побудови моделі явища, яке є джерелом аналізованого часового ряду. Побудована модель використовується для визначення майбутніх або невідомих пропущених значень часового ряду.

Відповідно до цього при здійсненні аналізу часових рядів розрізняють такі інструменти, як інтерполяція та екстраполяція.

Інтерполяція – знаходження значень у середині часового ряду на основі закономірностей розвитку явища, що досліджується.

Екстраполяція – поширення закономірностей, зв'язків і відношень, виявлених у певному періоді, за його межі. Якщо це роблять на перспективу, то тоді здійснюється прогнозування.

За тривалістю періоду упередження – горизонтом прогнозу – розрізняють прогнози короткострокові, середньострокові та довгострокові.

Горизонт прогнозу (англ. *Forecast Horizon*) є крайнім терміном, для якого прогноз дійсний із заданою точністю. Зазвичай при визначенні горизонту прийнято дотримуватися таких співвідношень у побудові прогнозу вперед відносно обсягу спостережень, відображених у рівнях часового ряду:

1) ***короткостроковий прогноз***: не більше ніж на 3% (на 1-2 кроки уперед);
2) ***середньостроковий прогноз***: не більше ніж на 3-5% (не більше ніж 7-12 кроків уперед);
3) ***довгостроковий прогноз***: більше ніж на 5%.

Однак таке розмежування не є чітко установленим, оскільки у цілому горизонт прогнозу залежить також від специфіки об'єкта, який досліджується, методів дослідження та виявлених закономірностей і тенденцій.

У структурі часового ряду y_t, який містить n рівнів: $y_1, y_2, \ldots, y_n$, можна виділити такі ***складові компоненти***: тренд T_t, циклічну компоненту U_t, сезонну компоненту S_t, випадкову компоненту E_t.

Графічне зображення часового ряду, який має у своєму складі тренд, сезонну та випадкову компоненти, наведено на рисунку 10.1 (дані для побудови часового ряду взято за адресою: https://fred.stlouisfed.org/series/MRTSSM44112USN).

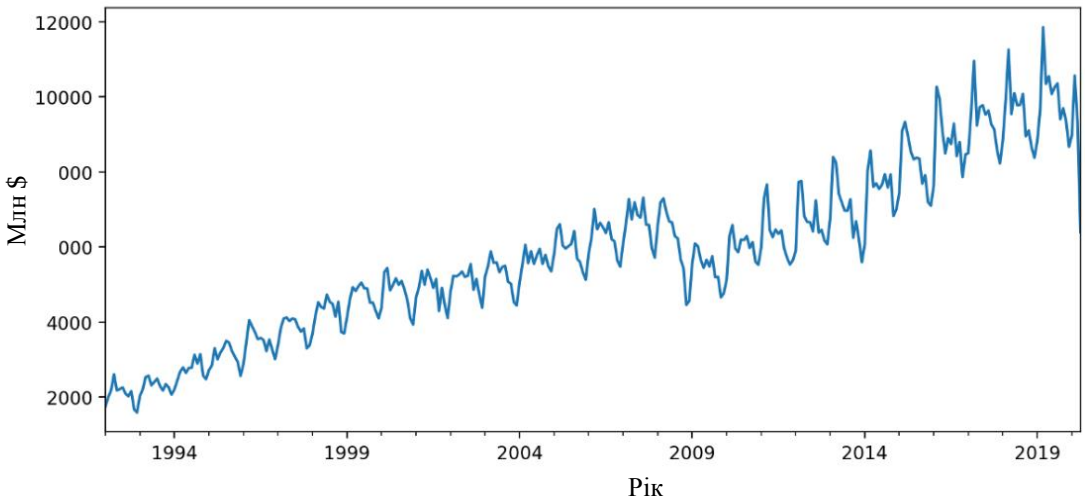


Рис. 10.1. Роздрібний продаж вживаних автомобілів у США

Основним структурним компонентом часового ряду є тренд. Він характеризує наявність загального напряму зміни ознаки, що досліджується, протягом тривалого часу. Тому при здійсненні аналізу часових рядів перш за все намагаються визначити тренд як основний, характерний напрям зміни y_t за деякий проміжок часу.

Трендом (англ. *Trend*) називають невипадкову функцію $f(t)$, що формується під дією загальних або довгострокових тенденцій, які впливають на часовий ряд.

Коливання – складова часового ряду, яка містить значення, близькі до тих, що повторюються, відносно основної тенденції – тренду. Коливання й відповідні їм компоненти часового ряду бувають сезонними та циклічними.

Сезонна компонента (англ. *Seasonal Component*) часового ряду містить коливання з невеликим періодом (день, місяць, рік), обумовлені впливом природно-кліматичних умов на досліджувану ознаку. Такі коливання проявляються у добувних галузях, сільському господарстві, споживанні енергоносіїв, води, певних груп товарів тощо.

Циклічна компонента (англ. *Cyclical Component*) часового ряду містить коливання з достатньо великим періодом (декілька років). Наприклад: часовий ряд інтенсивності сонячної активності, що має повторювані цикли з періодом близько 11 років.

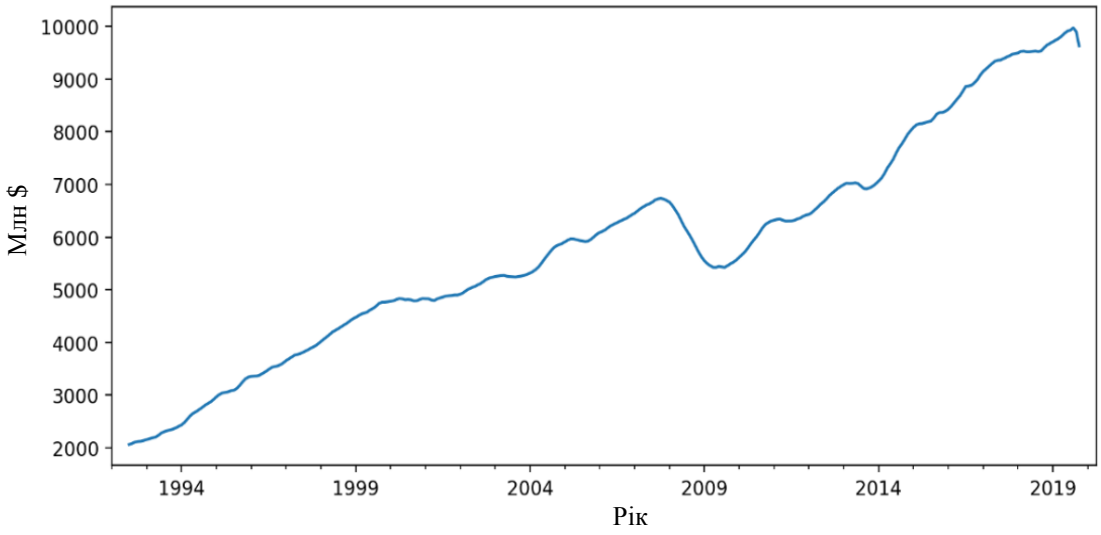
Випадкова (залишкова) компонента (англ. *Random/Remainder Component*) – складова часового ряду, обумовлена слабкими впливами великого числа випадкових або неврахованих невипадкових факторів.

Успішність аналізу часових рядів залежить від правильного вибору інтервалу між сусідніми рівнями. Вибір надмірно великого чи замалого інтервалу може призвести до втрати інформації про особливості ряду. Наприклад, можна не виявити наявність сезонної чи циклічної компонент часового ряду.

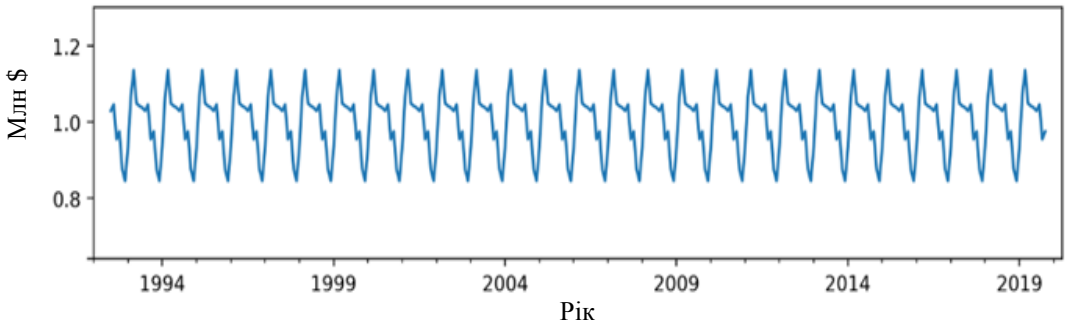
У структурі часового ряду можуть бути представлені усі перераховані вище компоненти або тільки деякі з них. Проводячи аналіз часового ряду з метою дослідження його структури та природи застосовують спеціальні методи для виявлення наявних компонент та здійснення декомпозиції часового ряду.

Декомпозиція часового ряду (англ. *Time Series Decomposition*) – виділення його компонент та дослідження кожної компоненти окремо.

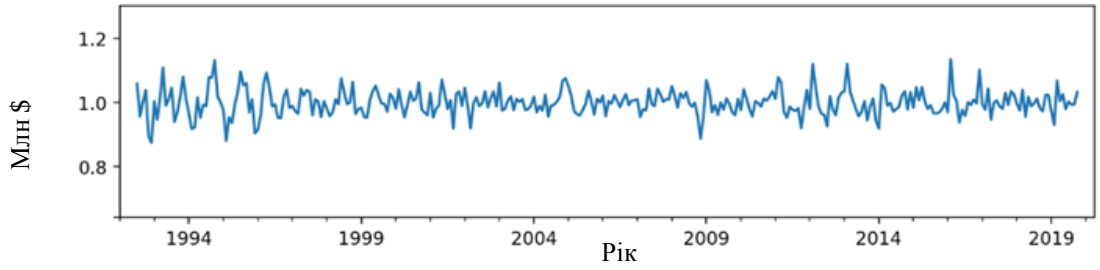
Після декомпозиції часового ряду, зображеного на рисунку 10.1, було виділено окремі компоненти, показані на рисунку 10.2 (дані з виділеними компонентами часового ряду взято за адресою: https://timeseriesreasoning.com/contents/time-series-decomposition/).



а) тренд



б) сезонна компонента



в) випадкова компонента

Рис. 10.2. Виділені у результаті декомпозиції компоненти часового ряду

Тренд і циклічні та сезонні коливання є ***регулярними***, або ***систематичними компонентами***. Вони характеризують основні зміни досліджуваної ознаки з плином часу.

Випадкова компонента – ***випадковий шум*** (англ. *Random Noise*), є ***нерегулярною компонентою*** часового ряду, яка ускладнює виявлення основних регулярних компонент.

Особливе місце в аналізі даних посідають стаціонарні часові ряди, у яких характеристики не змінюються з плином часу. Вони не мають у своєму складі тренду та коливань. Значення кожного рівня стаціонарного ряду рівне сумі середнього значення рівнів ряду та випадкової компоненти.

10.1.4. Побудова моделі часового ряду, оцінка її точності та адекватності

Визначення структури й закономірностей часового ряду передбачає виявлення шумів, викидів, тренду, сезонної та циклічної компонент і є основою для побудови ***моделі часового ряду***.

Можна виділити два ***основних напрями моделювання*** часових рядів:

1) моделювання регулярних компонент у сукупності;
2) розкладання часового ряду на компоненти і моделювання кожної компоненти окремо.

Виявлення структури часового ряду та типу трендової моделі здійснюють шляхом проведення:

1) ***аналізу графічного зображення часового ряду***: на етапі попереднього аналізу даних на координатну площину наносять точки з координатами (t, y_t) і по характеру їх розташування роблять висновок про наявність тренду і коливань, вид функції тренду, тип та період коливань часового ряду;
2) ***автокореляційного аналізу:*** висновок про структуру часового ряду роблять на основі аналізу динаміки змін коефіцієнта автокореляції зі зростанням величини лагу.

Виокремлення виявлених у структурі часового ряду складових називають ***фільтрацією компонент***. Багато методів дослідження часових рядів включають різні способи фільтрації шуму для більш чіткого виділення регулярних складових. При здійсненні фільтрації важливою є адекватна декомпозиція часового ряду.

Залежно від виду зв'язку між складовими часового ряду (T_t – трендом, U_t – циклічною, S_t – сезонною та E_t – випадковою компонентами) може бути побудована адитивна, мультиплікативна та змішана моделі часового ряду.

Адитивна модель (англ. *Additive model*) часового ряду передбачає, що кожен рівень ряду може бути представлений як сума його компонент:

$$y_t = T_t + U_t + S_t + E_t.$$

Мультиплікативна модель (англ. *Multiplicative model*) часового ряду передбачає, що кожен рівень ряду може бути представлений як добуток його компонент:

$$y_t = T_t \cdot U_t \cdot S_t \cdot E_t.$$

Змішана модель часового ряду може бути представлена як сума та добуток його компонент. Наприклад:

$$y_t = T_t + U_t \cdot S_t + E_t.$$

Вибір однієї з моделей проводиться на основі дослідження структури сезонних коливань по графічному зображенню часового ряду (рис. 10.3):

1) якщо амплітуда коливань приблизно постійна, будують адитивну модель часового ряду, в якій значення сезонної компоненти передбачаються постійними для різних циклів;
2) якщо амплітуда сезонних коливань зростає або зменшується, будуть мультиплікативну модель часового ряду, яка ставить рівні ряду в залежність від значень сезонної компоненти.

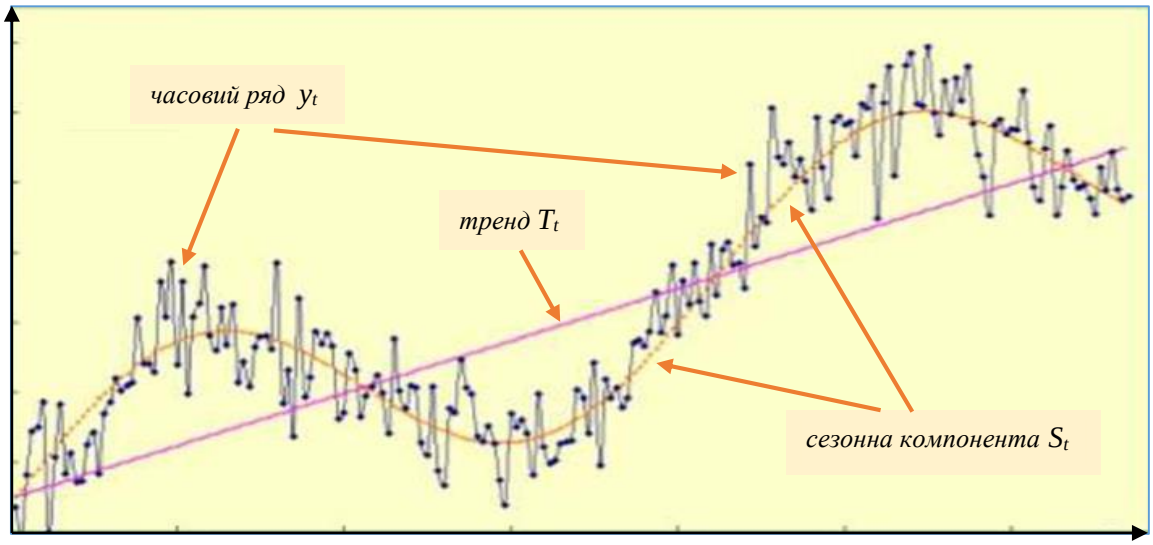
Для розв'язання задачі прогнозування важливим є моделювання тренду як основної тенденції часового ряду. Трендова модель, побудована за значеннями ознаки в минулому, використовується для прогнозування значень цієї ознаки у майбутньому, а врахування випадкової компоненти та коливань дозволяє визначити ступінь відхилення майбутніх значень від виявленого закономірного розвитку.

Побудова трендової моделі передбачає правильне виділення усіх компонент часового ряду. Якщо часовий ряд містить сезонні чи циклічні коливання, їх виокремлюють зі структури часового ряду і далі будують трендову модель таким чином, щоб виділена випадкова компонента була близькою до нуля однаково розподіленою незалежною випадковою величиною – процесом типу ***білого шуму***.

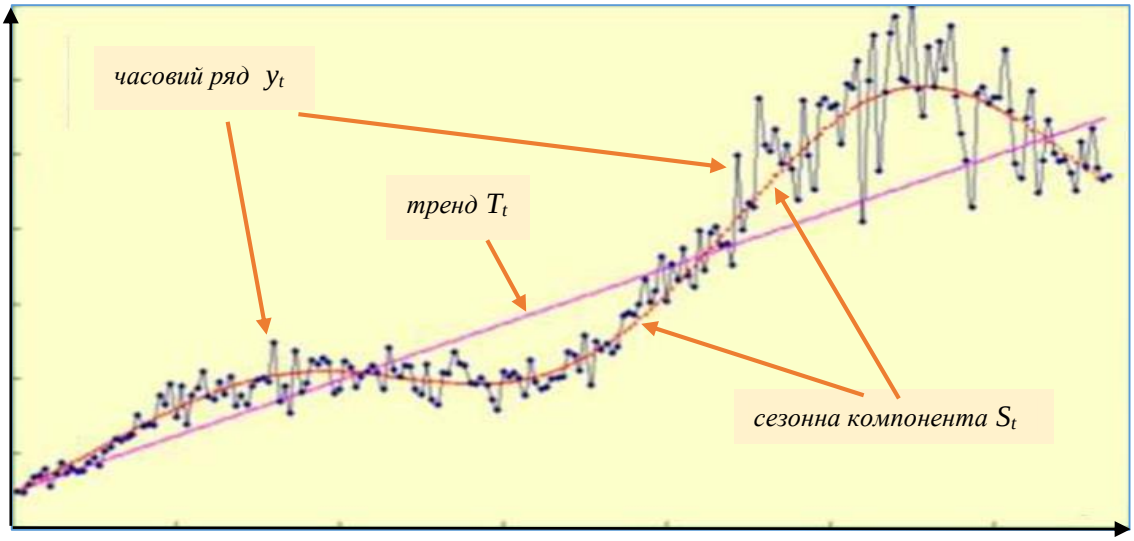
Білий шум – це стаціонарний часовий ряд, який має сталу дисперсію, середню, що дорівнює нулю, та нульову автокореляцію.

Білим шумом може бути часовий ряд, тоді значення досліджуваної ознаки є випадковими і не підлягають моделюванню, а прогнозування не є можливим. Якщо білим шумом є виділена випадкова компонента часового

ряду, це свідчить про те, що у побудованій моделі була правильно використана уся інформація про значення досліджуваної ознаки об'єкта прогнозування.



а) адитивна модель



б) мультиплікативна модель

Рис. 10.3. Зображення часових рядів різного компонентного складу

Таким чином, моделювання тренду – основної тенденції зміни досліджуваної ознаки – здійснюють відфільтровуючи випадкову компоненту часового ряду та коливання. З цією метою використовують ***методи згладжування***, які можна розділити на такі основні групи.

1. ***Механічне вирівнювання*** часового ряду: згладжування рівнів ряду проводиться з використанням значень сусідніх рівнів шляхом переходу до усереднених на деякому інтервалі значень.
2. ***Аналітичне вирівнювання*** часового ряду: реалізується за допомогою регресійних та адаптивних методів і базується на відомому загальному вигляді регулярних компонент часового ряду.

Методи механічного згладжування добре справляються з виокремленням сезонної та випадкової компонент, залишаючи значення, які містять тренд та можливо циклічну компоненту. Виявлення циклічної компоненти часового ряду є складнішою задачею, оскільки потребує наявності даних за тривалий період часу. У переважній більшості випадків при аналізі часових рядів наявністю циклічної компоненти нехтують. А якщо потрібно враховувати циклічну компоненту, то для її виділення застосовують спеціальні методи, засновані на спектральному аналізі.

Використання для ***моделювання тренду*** регресійних методів передбачає побудову аналітичної функції, яка характеризує залежність рівнів ряду від часу – тенденції часового ряду й є одним із поширених методів моделювання тренду. На практиці побудова трендової моделі полягає у формуванні набору апроксимуючих функцій та оцінюванні їх параметрів і адекватності. Як тренд можуть бути обрані лінійна, показникова, параболічна, логарифмічна та інші функції. Для оцінки параметрів рівняння тренду, використовуючи t як незалежну змінну, а y_t – як залежну змінну, застосовують метод найменших квадратів та інші методи.

Для нелінійних трендів оцінка їх параметрів може бути складною у розрахунковому відношенні. У таких випадках проводять лінеаризацію апроксимуючої функції шляхом введення нових змінних або застосовують методи нелінійної оптимізації.

Після оцінки параметрів трендових моделей критерієм відбору кращої із них є найбільше значення ***коефіцієнту детермінації***:

$$R^2 = 1 - \frac{\sum_{t=1}^{n}(y_t - \hat{y}_t)^2}{\sum_{t=1}^{n}(y_t - \overline{y}_t)^2}, \tag{10.10}$$

де $\hat{y}_t$ – значення рівнів часового ряду, розрахованих за трендовою моделлю;

$\overline{y}_t$ – середнє фактичних значень рівнів часового ряду (з вилученими значеннями сезонної чи циклічної компонент у разі їх наявності у структурі ряду).

Коефіцієнт детермінації дозволяє оцінити ***ступінь апроксимації*** трендовою моделлю основної тенденції зміни досліджуваної ознаки. Із побудованих моделей обирають ту, яка має найбільше значення коефіцієнта детермінації, враховуючи такі емпіричні правила: $R^2 > 0{,}95$ – висока точність апроксимації, $0{,}8 < R^2 < 0{,}95$ – задовільна точність апроксимації, $R^2 < 0{,}6$ – незадовільна точність апроксимації.

Основою ***перевірки моделі на адекватність*** є дослідження випадкової компоненти часового ряду. Якщо випадкова компонента є білим шумом, то фільтрація компонент часового ряду була проведена правильно, а побудована модель є адекватною.

Таким чином, ***основні етапи аналізу*** часових рядів із метою побудови моделі та здійснення прогнозування є наступними:

1. Попередній аналіз даних: виявлення аномальних відхилень, перевірка на наявність тренду.
2. Графічне зображення та попередній аналіз динаміки, перевірка на стаціонарність, визначення структури часового ряду.
3. Згладжування, фільтрація, виокремлення компонент часового ряду.
4. Побудова трендових моделей: формування набору апроксимуючих функцій і чисельне оцінювання параметрів цих моделей.
5. Перевірка адекватності трендових моделей, оцінка точності апроксимації та вибір кращої моделі.
6. Дослідження випадкової компоненти, перевірка адекватності побудованої моделі часового ряду.
7. Оцінка точності прогнозу, прогнозування з використанням побудованої моделі часового ряду.

Точність побудованої моделі часового ряду перевіряють на даних, для яких відомі фактичні значення ознаки. Застосування таких підходів оцінки точності моделі часового ряду є доцільним для короткострокових прогнозів та випадків прогнозування для періодів минулого, для яких є фактичні значення. Ця точність прогнозу береться за основу для значень, які будуть отримані у майбутньому. Однак у випадку довготривалих прогнозів можуть з'явитися фактори, не передбачені моделлю. Тому вона може потребувати перегляду.

Оцінка точності прогнозу здійснюється з використанням наступних показників.

1. ***MFE*** (англ. *Mean Forecast Error*) – середня похибка прогнозу є мірою середнього відхилення прогнозних значень від фактичних, показує напрям зміщення прогнозу – заниженими чи завищеними є прогнозні значення:

$$MFE = \frac{1}{n}\sum_{t}^{n}(y_t - \hat{y}_t), \tag{10.11}$$

де $\hat{y}_t$ та y_t – прогнозоване та фактичне значення рівнів ряду.

2. ***MAE*** або ***MAD*** (англ. *Mean Absolute Error/Mean Absolute Derivation*) – середнє абсолютне відхилення прогнозованих значень від фактичних:

$$MAE = MAD = \frac{1}{n}\sum_{t}^{n}\left|y_t - \hat{y}_t\right|, \tag{10.12}$$

3. ***MAPE*** (англ. *Mean Absolute Percentage Error*) – середня абсолютна похибка у відсотках:

$$MAPE = \frac{100}{n}\sum_{t=1}^{n}\left|\frac{y_t - \hat{y}_t}{y_t}\right|. \quad (10.13)$$

Точність прогнозу є: дуже високою, якщо MAPE менше 10%; високою, якщо MAPE становить 10-20%; задовільною, якщо MAPE становить 20-50%; незадовільною, якщо MAPE більше 50%.

4. ***MPE*** (англ. Mean Percentage Error) – середня похибка у відсотках:

$$MPE = \frac{100}{n}\sum_{t}^{n}\left(\frac{y_t - \hat{y}_t}{y_t}\right). \quad (10.14)$$

5. ***MSE*** (англ. *Mean Squared Error*) – середньоквадратична похибка дає загальне уявлення, чи є помилки при прогнозуванні:

$$MSE = \frac{1}{n}\sum_{t}^{n}(y_t - \hat{y}_t)^2. \quad (10.15)$$

При підборі параметрів моделі та порівнянні декількох моделей обирають ті з них, для яких MSE буде мати менше значення.

Чим меншими є значення наведених показників, тим вищою є точність прогнозу.

10.1.5. Виявлення аномальних відхилень

На етапі підготовки даних до аналізу при здійсненні очищення набору даних для ***виявлення викидів*** – аномальних рівнів часового ряду – застосовують різні методи. Одним із них є метод Ірвіна, суть якого полягає у розрахунку критерію Ірвіна λ_t для всіх рівнів часового ряду та порівняння їх із критичним (табличним) значенням критерію Ірвіна $\lambda_{кр}$, визначеним на рівні значущості α (зазвичай $\alpha = 0{,}05$ або $\alpha = 0{,}01$).

Критерій Ірвіна для часового ряду y_t, що містить n рівнів: $y_1, y_2, \ldots, y_n$, розраховують за формулою:

$$\lambda_t = \frac{|y_t - y_{t-1}|}{\sigma}, \quad (10.16)$$

де $\sigma = \sqrt{\sigma^2}$ – оцінка середньоквадратичного відхилення часового ряду.

Аномальним рівнем – значенням часового ряду y_t довжиною n, яке сильно вибивається із загальної тенденції, з ймовірністю $p = 1 - \alpha$ буде вважатися таке, для якого розраховане значення критерію Ірвіна буде більшим за критичне значення, визначене на рівні значущості α для даного n: $\lambda_t > \lambda_{кр}$.

Критичні значення критерію Ірвіна наведено у додатку *П* та у таблиці 10.4. Зазначимо, що з ростом довжини часового ряду критичне значення критерію зменшується. Цю властивість використовують у розрахунках з виявлення аномальних відхилень.

Таблиця 10.4

Критичні значення критерію Ірвіна $\lambda_{кр}$

Довжина ряду n		2	6	10	20	30	40	50	100	200
$\lambda_{кр}$	$\alpha = 0{,}05$	2,77	1,67	1,46	1,27	1,20	1,15	1,11	1,02	0,95

Після виявлення аномальних значень з' їх природу. Якщо це технічні помилки, що виникають при зборі та передачі даних, агрегуванні значень (помилки першого роду), роблять заміну аномального значення значенням, яке рівне середній арифметичній сусідніх з аномальним рівнів ряду. Якщо аномальні значення відображають вплив об'єктивних факторів, які проявляють себе рідко та епізодично (помилки другого роду), аномальні значення заміні та вилученню не підлягають.

Приклад 2. Для часового ряду, представленого у таблиці 10.2, на рівні значущості $\alpha = 0{,}05$ виявити наявність викидів із використанням критерію Ірвіна.

Незміщена оцінка дисперсії часового ряду, розрахована за формулою 10.3, буде рівна:

$$\sigma^2 = \frac{1}{5}((227 - 157{,}5)^2 + (233 - 157{,}5)^2 + (177 - 157{,}5)^2 +$$
$$+(185 - 157{,}5)^2 + (43 - 157{,}5)^2 + (80 - 157{,}5)^2) = 6156{,}7.$$

Тоді незміщена оцінка середньоквадратичного відхилення для даного часового ряду рівна:

$$\sigma = \sqrt{6156{,}7} = 78{,}46.$$

За формулою 10.17 розрахуємо критерій Ірвіна для кожного з рівнів:

$$\lambda_2 = \frac{|y_2 - y_1|}{\sigma} = \frac{|233-227|}{78{,}46} = \frac{6}{78{,}46} = 0{,}076, \qquad \lambda_3 = \frac{|y_3 - y_2|}{\sigma} = \frac{|177-233|}{78{,}46} = \frac{56}{78{,}46} = 0{,}714,$$

$$\lambda_4 = \frac{|y_4 - y_3|}{\sigma} = \frac{|185-177|}{78{,}46} = \frac{8}{78{,}46} = 0{,}102, \qquad \lambda_5 = \frac{|y_5 - y_4|}{\sigma} = \frac{|43-185|}{78{,}46} = \frac{142}{78{,}46} = 1{,}810,$$

$$\lambda_6 = \frac{|y_6 - y_5|}{\sigma} = \frac{|80-43|}{78{,}46} = \frac{37}{78{,}46} = 0{,}472.$$

У результаті на рівні значущості $\alpha = 0{,}05$ одне із розрахованих значень критерію Ірвіна є більшим за критичне значення $\lambda_{\text{кр}} = 1{,}67$ для $n = 6$: $\lambda_5 = 1{,}810$.

Отже, $\lambda_5 > \lambda_{кр}$, тому з ймовірністю 95% ми можемо стверджувати, що на 5-му рівні часового ряду є викид.

Різке зменшення кількості пошукових запитів «*купити нетбук*» в Україні у березні 2022 року обумовлене об'єктивними причинами зменшення попиту на них у зв'язку з запровадженням військового стану та веденням бойових дій на території країни.

10.1.6. Перевірка наявності тренду

Наявність тренду не завжди можна виявити, аналізуючи графічне зображення ряду. Для більш достовірного підтвердження наявності чи відсутності тренду у структурі часового ряду використовують спеціальні критерії перевірки гіпотези про наявність тренду. З цією метою застосовують різні методи: метод перевірки різниць середніх, метод Форстера–Стьюарта, критерій серій та інші.

Метод ***перевірки різниць середніх рівнів*** зводиться до розбиття часового ряду на дві майже однакові по числу рівнів частини, кожна з яких розглядається як нормально розподілена сукупність значень. Якщо часовий ряд має тренд, то середні, розраховані для кожної частини ряду, повинні істотно (значимо) розрізнятися між собою. Якщо ж розбіжність несуттєва, то часовий ряд не має тренду. Тому перевірка наявності тренду зводиться до перевірки гіпотези про рівність середніх двох нормально розподілених сукупностей, що перевіряється за допомогою t-критерію Стьюдента. Однак цей метод можна застосовувати тільки у тих випадках, коли обидві частини ряду будуть мати однакову дисперсію. Для перевірки гіпотези про рівність дисперсій використовують F-критерій Фішера.

Послідовність етапів методу перевірки різниць середніх рівнів є наступною.

1. Часовий ряд довжиною n розбивається на дві майже однакові частини n_1 та n_2 такі, що $n = n_1 + n_2$.
2. Розраховують середні обох частин:

$$\bar{y}_1 = \frac{1}{n_1}\sum_{t=1}^{n-n_1} y_t \text{ та } \bar{y}_2 = \frac{1}{n_2}\sum_{t=n_1+1}^{n_1} y_t .$$

3. Розраховують дисперсії обох частин ряду: $\sigma_1^2 = \frac{1}{n_1 - 1}\sum_{t=1}^{n_1}(y_t - \bar{y}_1)^2$ та $\sigma_2^2 = \frac{1}{n_2 - 1}\sum_{t=n_1+1}^{n}(y_t - \bar{y}_2)^2$.
4. Розраховують емпіричне значення F-критерію Фішера за формулою:

$$F_{емп} = \begin{cases} \sigma_1^2 / \sigma_2^2, & якщо \quad \sigma_1^2 > \sigma_2^2 \\ \sigma_2^2 / \sigma_1^2, & якщо \quad \sigma_1^2 < \sigma_2^2 \end{cases} .$$

5. Якщо емпіричне значення критерію Фішера $F_{\text{емп}} < F_{\text{кр}}$ менше за критичне на заданому рівні значущості α, взяте зі ступенями свободи $n_i - 1$ та $n_k - 1$ (i – індекс тієї частини ряду, яка має більшу дисперсію), то гіпотеза про однорідність дисперсій приймається і необхідно переходити до наступного пункту.
6. Якщо $F_{емп} \geq F_{кр}$, то гіпотеза про однорідність дисперсій відхиляється, метод не дає відповіді на питання про наявність тренду.
7. Розраховують емпіричне значення t-критерію Стьюдента за формулою:

$$t_{емп} = \frac{|\bar{y}_1 - \bar{y}_2|}{\sigma\sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}, \tag{10.17}$$

де σ – оцінка середньоквадратичного відхилення різниці середніх:

$$\sigma = \sqrt{\frac{(n_1-1)\sigma_1^2 + (n_2-1)\sigma_2^2}{n_1+n_2-2}}\,. \tag{10.18}$$

8. Якщо емпіричне значення критерію Стьюдента менше за табличне $t_{\text{емп}} < t_{\text{кр}}$ на заданому рівні значущості α із числом ступенів свободи $n_1 + n_2 - 2$, то гіпотеза про рівність середніх приймається – тренду немає.
9. Якщо $t_{емп} \geq t_{кр}$, то гіпотеза про рівність середніх відхиляється, тренд є.

При здійсненні описаних вище розрахунків для достовірного підтвердження наявності чи відсутності тренду можна скористатися наступними функціями MS Excel та MatLab.

1. Середнє значення: *MS Excel*: $СРЗНАЧ()/AVERAGE()$ *MatLab*: $mean()$
2. Незміщена оцінка дисперсії: *MS Excel*: $ДИСП.В()/VAR.S()$ *MatLab*: $var()$
3. Критичне (табличне) значення F-критерію Фішера $F_{кр}$:
   MS Excel: $FРАСПОБР(\alpha; n_1; n_2)/FINV(\alpha; n_1; n_2)$, $F.ОБР.ПХ(\alpha; n_1; n_2)/F.INV.RT(\alpha; n_1; n_2)$
   MatLab: $finv(1-\alpha, n_1, n_2)$
4. Критичне (табличне) значення t-критерію Стьюдента $t_{кр}$:
   MS Excel: $СТЬЮДРАСПОБР(\alpha; n)/TINV(\alpha; n)$, $СТЬЮДЕНТ.ОБР.2Х(\alpha; n)/T.INV.2T(\alpha; n)$
   MatLab: $tinv(1-\alpha, n)$
5. Функція MS Excel, яка повертає результат F-тесту: двосторонню ймовірність того, що різниця між дисперсіями рядів у діапазонах *масив*1 та *масив*2 є несуттєвою:
   $$F.ТЕСТ(масив1; масив2)/F.TEST(масив1; масив2)$$
6. Функція MatLab, яка повертає результат t-тесту: $ttest2(y_1, y_2, \alpha)$. Якщо функція повертає значення, рівне нулю, то гіпотеза про рівність середніх рядів приймається. Якщо функція повертає значення, рівне одиниці, то гіпотеза про рівність середніх рядів відхиляється на рівні значущості α.

Функції для розрахунку критичних значень критеріїв Фішера та Стьюдента в MS Excel як аргумент містять рівень значущості α, а в MatLab – рівень достовірності $1-\alpha$.

Недоліком методу перевірки різниць середніх рівнів є неможливість установити наявність тренду у разі зміни тенденції в середині ряду. Тоді для підтвердження наявності тренду застосовують інші методи.

Ще одним методом, за допомогою якого можна установити наявність тренду у структурі часового ряду, є використання ***критерію серій***. Послідовність його етапів є наступною.

1. З часового ряду y_t довжиною n: $y_1, y_2, \ldots, y_n$, утворюють ранжований ряд та визначають його медіану m.
2. Для часового ряду y_t формують послідовність „+” та «-» за правилом:
$$\delta_i = \begin{cases} +, якщо & y_t > m \\ -, якщо & y_t < m \end{cases}, \tag{10.19}$$
де $t = 1,2,\ldots,n$. Якщо значення рівне медіані, це значення пропускають.
3. Підраховують $v(n)$ – число серій у сукупності δ_i, де серія – це послідовність плюсів або мінусів, які ідуть підряд (один «+» та «-» також є серією).
4. Підраховують t_{max} – довжину найбільшої серії, яка має найбільшу кількість «+» або «-».
5. Розраховують критичне значення кількості серій $v_{\text{кр}}(n)$ для ряду довжиною *n* на рівні значущості 0,05:
$$v_{кр}(n) = \left[\frac{1}{2}\cdot(n+1-1.96\sqrt{n-1}\right], \tag{10.20}$$
(квадратні дужки у правій частині нерівності означають цілу частину числа).
6. Розраховують критичне значення довжини найдовшої серії $t_{\text{кр}}(n)$ для ряду довжиною *n* на рівні значущості 0,05:
$$t_{кр}(n) = [3.3\cdot(\lg n+1)]\,. \tag{10.21}$$
7. Перевіряють, чи виконуються нерівності:
$$\begin{cases} t_{\max}(n) < t_{кр}(n) \\ v(n) > v_{кр}(n) \end{cases}. \tag{10.22}$$

Для підтвердження гіпотези про відсутність у структурі ряду тренду довжина найбільшої серії не повинна бути дуже великою, а кількість серій – дуже малою.

8. Якщо обидві нерівності виконуються – тренд відсутній. Якщо хоча б одна з нерівностей не виконується – гіпотеза про відсутність тренду відхиляється: тренд є.

Приклад 3. За даними щоквартальних значень змінної y за 5 років установити наявність тренду з використанням методу перевірки середніх різниць (табл. 10.5).

Таблиця 10.5

Дані щоквартальних значень змінної y за 5 років

№ кварталу, t	1	2	3	4	5	6	7	8	9	10
y_t	7,2	8,0	-3,3	13,2	10,7	10,9	0,4	16,2	14,6	16,2
№ кварталу, t	11	12	13	14	15	16	17	18	19	20
y_t	4,6	21,0	19,5	18,8	7,8	25,0	20,9	23,9	12,3	28,4

1. Розрахунки будемо проводити в MatLab. Для розв’язання поставленої задачі створюємо скрипт:

```
% часовий ряд задаємо як вектор y
y=[7.2 8.0 -3.3 13.2 10.7 10.9 0.4 16.2 14.6 16.2 4.6 21.0 19.5 18.8 7.8 25.0 20.9 23.9 12.3 28.4];
n=length(y); % визначаємо довжину часового ряду
% розбиття часового ряду на дві майже рівні частини
n1=floor(n/2); n2=n-n1;
y1=y(1:n1); y2=y((n1+1):n);
% розрахунок середніх значень, дисперсій кожної частини
mean1=mean(y1);
mean2=mean(y2);
var1=var(y1);
var2=var(y2);
% обчислення емпіричного значення критерію Фішера
if var1>var2
   Fr=var1/var2
else
   Fr=var2/var1
end
% визначення критичного значення критерію Фішера
Fcr=finv(0.95,n1,n2) % на рівні значущості 0,05
% перевірка гіпотези про рівність дисперсій
if Fr>=Fcr
   disp('метод не може визначити наявність тренду');
else
   % обчислення розрахункового значення критерію Стьюдента
   sr=sqrt(((n1-1)*var1+(n2-1)*var2)/(n1+n2-2));
   t_r=abs(mean1-mean2)/(sr*sqrt(1/n1+1/n2))
   % визначення критичного значення критерію Стьюдента
   t_cr=tinv(0.95,n1+n2-2) % на рівні значущості 0,05
   % перевірка гіпотези про рівність середніх
   if t_r<t_cr
      disp('тренд відсутній')
   else
      disp('тренд є')
   end
end
% ймовірність того, різниця середніх двох частин є несуттєвою
vt_cr=ttest2(y1,y2, 0.05)
```

Результатом виконання цього скрипта буде підтвердження з ймовірністю 95% наявності або відсутності тренду в структурі часового ряду або повідомлення про те, що такий метод не дає можливості установити наявність тренду.

2. Здійснимо аналіз отриманих результатів (рис. 10.4). Емпіричне значення критерію Фішера $F_{\text{емп}} = 1{,}3641$ на рівні значущості $\alpha = 0{,}05$ виявилося меншим за критичне $F_{кр} = 2{,}9782$, тому гіпотеза про однорідність дисперсій приймається, можна переходити до розрахунку критерію Стьюдента.

3. Емпіричне значення критерію Стьюдента $t_{\text{емп}} = 2{,}7632$ на рівні значущості $\alpha = 0{,}05$ виявилося більшим за критичне $t_{кр} = 1{,}7341$ (рис. 10.4), тому гіпотеза про рівність середніх відхиляється. З ймовірністю 95% можемо стверджувати, що у структурі аналізованого часового ряду тренд є.

4. Про наявність тренду свідчить також результат t-тесту (рис. 10.4): на рівні значущості $\alpha = 0{,}05$ гіпотезу про рівність середніх двох частин часового ряду було відхилено. Отже, з ймовірністю 95% було достовірно установлено, що аналізований часовий ряд має тренд.

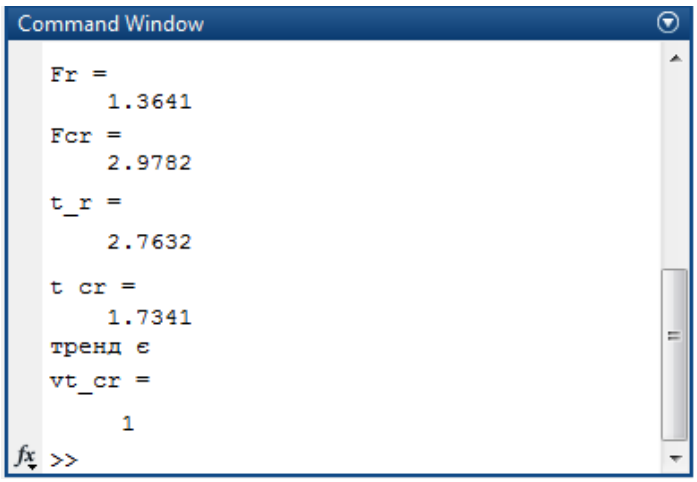


Рис. 10.4. Результат визначення наявності тренду методом перевірки різниць середніх рівнів

Приклад 4. За даними про споживання електроенергії мешканцями регіону за 4 роки виявити структуру часового ряду (табл. 10.6). З цією метою:

1) здійснити аналіз характерної зміни ознаки за графічним зображення ряду;

2) виявити наявність тренду з використанням методу перевірки різниць середніх;

3) виявити наявність чи відсутність тренду за допомогою критерію серій.

Таблиця 10.6

Дані про споживання електроенергії за 4 роки (млн кВт·г)

№ кварталу, t	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Спожив. ел. енергії, y_t	6	4,4	5	9	7,2	4,8	6	10	8	5,6	6,4	11	9	6,6	7	10,8

1. ***Аналіз характерної зміни ознаки за графічним зображення ряду.***

1.1. У MS Excel формуємо електронну таблицю з вхідними даними часового ряду та будуємо точкову діаграму (t, y_t) (рис. 10.5).

1.2. Аналіз графічного зображення часового ряду показує, що загальна тенденція зміни значень рівнів ряду є зростаючою. Також ряд містить сезонні коливання з періодичністю 4 квартали та приблизно рівною амплітудою коливань. Це дає підстави припустити, що для моделювання даного часового ряду може бути обрана адитивна модель, яка містить тренд, сезонну та випадкову компоненти: $y_t = T_t + S_t + E_t$.

2. ***Виявлення наявності тренду з використанням методу перевірки середніх різниць.***

2.1. Установимо наявність тренду з використанням методу перевірки середніх різниць. Для цього часовий ряд у MS Excel розділимо на дві рівні частини $n_1 = 8$ та $n_2 = 8$, роміщені в діапазонах комірок C7:C14 і C15:C22 та перевіримо ймовірність того, що різниця між дисперсіями двох частин є несуттєвою, ввівши формулу:

$= F.TECT(C7:C14; C15:C22)$. У комірці з формулою отримаємо значення 0,997, що свідчить про те, що з ймовірністю 0,997% можемо стверджувати, що дисперсії двох частин часового ряду є однорідними.

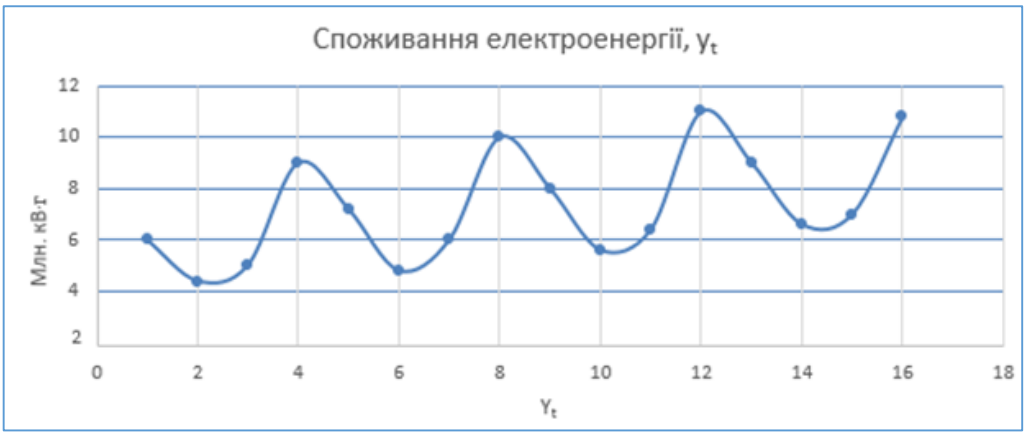


Рис. 10.5. Споживання електроенергії мешканцями регіону (*млн кВт·г*)

2.2. Для перевірки гіпотези про рівність середніх розрахуємо середні та дисперсії двох частин часового ряду за формулами:

1) для першої частини: $=$ СРЗНАЧ(C7:C14) та $= ДИСП.В(C7:C14)$;

2) для другої частини: $=$ СРЗНАЧ(C15:C22) та $= ДИСП.В(C15:C22)$.

Для першої частини ряду середнє рівне 6,55, дисперсія – 4,146. Для другої частини ряду середнє рівне 8,05, дисперсія – 4,157.

2.3. Розрахуємо σ за формулою 9.18, отримаємо $\sigma = 2{,}038$.

2.4. Далі за формулою 10.17 обчислимо емпіричне значення t-критерію Стьюдента, ввівши формулу: $= ABS(D16 - H16)/(F17 *$ КОРЕНЬ$(1/7 + 1/7))$ (у комірках D16 і H16 знаходяться дисперсії двох частин ряду, а у комірці F17 – обчислене значення σ). Отримаємо емпіричне значення критерію Стьюдента, рівне $t_{емп} = 1{,}377$.

2.6. Для визначення критичного значення критерію Стьюдента на рівні значущості 0,05 введемо формулу: $=$ СТЬЮДЕНТ.ОБР.2Х$(0{,}05; 14)$. У комірці з формулою отримаємо значення $t_{кр} = 2{,}145$.

2.7. Оскільки розрахункове значення критерію Стьюдента $t_р = 1{,}377$ є меншим за критичне $t_{кр} = 2{,}145$, визначене на рівні значущості 0,05, гіпотеза про рівність середніх приймається. Тому з ймовірністю 95% можемо стверджувати що у структурі часового ряду тренду немає.

3. ***Виявлення наявності тренду з використанням критерію серій.***

3.1. Установимо наявність тренду у структурі часового ряду, представленого у таблиці 10.6, із використанням критерію серій. Розрахунки будемо проводити в MS Excel. У комірки B5:B20 введемо значення рівнів часового ряду та визначимо медіану, ввівши у комірці G4 формулу: $= МЕДИАНА(B5:B20)$. Розраховане значення медіани буде рівне 6,8 (рис. 10.6).

3.2. У стовпці *Серії* у діапазоні комірок C5:C20 проставляємо: «+» – якщо значення рівня ряду ліворуч більше за медіану, «-» – якщо менше за медіану та пропускаємо, якщо рівне медіані (рис. 10.6).

3.3. За формулами 10.20 та 10.21 розраховуємо критичні значення кількості серій $v_{кр}(n)$ та довжини найдовшої серії $t_{кр}(n)$ для ряду довжиною $n = 16$ на рівні значущості 0,05:

a) для розрахунку $t_{кр}(16)$ у комірку I11 вводимо формулу: $=$ ТОБР$(3{,}3 * (LOG10(16) + 1))$, отримуємо $t_{кр}(16) = 7$;

b) для розрахунку $v_{кр}(16)$ у комірку I7 вводимо формулу:
$=$ ТОБР$(0{,}5 * (16 + 1 - 1{,}96 *$ КОРЕНЬ$(16 - 1)))$, отримуємо $v_{кр}(16) = 4$.

ТОБР() – функція MS Excel, яка бере цілу частину від числа, відкидаючи дробову.

3.4. Рахуємо у стовпці *Серії* (рис. 10.6) кількість серій $v(n) = 8$ – послідовність плюсів або мінусів, які ідуть підряд і довжину найдовшої серії t_{max} – яка має найбільшу кількість «+» або «-», та порівнюємо їх із критичними. Маємо:

$$\begin{cases} t_{\max}(n) = 3 < t_{кр}(n) = 4 \\ v(n) = 8 > v_{кр}(n) = 4 \end{cases}.$$

Оскільки система нерівностей 10.22 виконується, з ймовірністю 95% можна стверджувати, що тренд у структурі часового ряду відсутній.

Таким чином, аналіз графічного зображення часового ряду, представленого у таблиці 10.6, дав підстави для припущення про наявність у структурі ряду зростаючої тенденції. Однак застосування методу перевірки середніх різниць та критерію серій дозволило установити, що з ймовірністю 95% можна стверджувати про відсутність у структурі ряду тренду.

	A	B	C	D	E	F	G	H	I	J	K
1											
2	Перевірка наявності тренду за допомогою критерію серій										
3											
4	t	y_t	Серії		Медіана		6,8				
5	1	6	-								
6	2	4,4	-		Число серій						
7	3	5	-			v(n)	8	>	4	$v_{кр}(n)$	
8	4	9	+								
9	5	7,2	+								
10	6	4,8	-		Довжина самої довшої серії						
11	7	6	-			$t_{max}(n)$	3	<	7	$t_{кр}(n)$	
12	8	10	+								
13	9	8	+								
14	10	5,6	-		**Тренд відсутній**						
15	11	6,4	-								
16	12	11	+		"+" - більше за медіану						
17	13	9	+		"-" - менше за медіану						
18	14	6,6	-								
19	15	7	+								
20	16	10,8	+								

Рис. 10.6. Перевірка наявності тренду за допомогою критерію серій

10.1.7. Автокореляційний аналіз часового ряду

За наявності у структурі часового ряду тренду і циклічних та сезонних коливань обов'язково буде мати місце автокореляція, оскільки значення кожного наступного рівня ряду буде залежати від значень попередніх рівнів.

Автокореляційний аналіз дозволяє виявити структуру часового ряду та визначити, які невипадкові чинники беруть участь у формування його значень. Для його проведення розраховують коефіцієнти автокореляції з лагом p, значення якого послідовно зростає, починаючи від $p = 1$. Розраховані значення дозволяють побудувати графік автокореляційної функції.

Автокореляційна функція часового ряду відображає залежність значень коефіцієнта автокореляції від величини лагу та є послідовністю коефіцієнтів автокореляції першого, другого та ін. порядків.

Корелограма – графік автокореляційної функції. Побудова корелограми полегшує проведення аналізу, оскільки вона дозволяє візуально відобразити динаміку зміни коефіцієнта автокореляції зі зростанням величини лагу.

При здійсненні ***аналізу значень коефіцієнта автокореляції*** необхідно враховувати наступне.

1. Коефіцієнт автокореляції дозволяє характеризувати тісноту лінійного зв'язку між рівнями часового ряду.
2. Близькість значень коефіцієнта автокореляції до 1 або до -1 говорить про тісний лінійний зв'язок між рівнями часового ряду.
3. Для деяких часових рядів із сильною нелінійною тенденцією коефіцієнт автокореляції може бути близьким до нуля.
4. Знак коефіцієнта автокореляції не дозволяє робити висновок про напрям зміни тенденції. При додатному чи від'ємному значенні коефіцієнта автокореляції часовий ряд може мати як спадну, так і зростаючу тенденцію.

Аналіз корелограми дозволяє зробити висновок про структуру часового ряду, виходячи із наступних положень.

1. Для стаціонарного часового ряду характерним є чергування затухаючих додатніх та від'ємних статистично незначущих значень коефіцієнтів автокореляції.
2. Якщо найбільшим по модулю виявилося значуще значення коефіцієнта автокореляції 1-го порядку, досліджуваний ряд містить тільки тренд.
3. Якщо корелограма містить багато максимальних і мінімальних значень коефіцієнтів автокореляцій, а найбільшим по модулю виявилося значуще значення коефіцієнта автокореляції k-го порядку, досліджуваний ряд містить коливання з періодом k.
4. Якщо жоден із коефіцієнтів автокореляції не є значущим, то можна зробити припущення про те, що ряд не містить тренду та коливань або ряд має сильний нелінійний тренд і потребує додаткового аналізу.

За наявності тренду коефіцієнти автокореляції є значущими. Для перевірки коефіцієнта автокореляції на значущість у разі відсутності тренду можна скористатися критерієм стандартної похибки. Якщо значення коефіцієнта автокореляції r_p порядку p не виходить на межі інтервалу:

$$-1{,}96 \cdot \frac{1}{\sqrt{n}} \le r_p \le 1{,}96 \cdot \frac{1}{\sqrt{n}}, \qquad (10.23)$$

Де n – довжина часового ряду, то коефіцієнт автокореляції r_p не буде значимим і автокореляція порядку p відсутня. А якщо значення r_p виходить за межі цього інтервалу, то з ймовірністю 95% можемо стверджувати, що коефіцієнт автокореляції r_p є значимим.

Якщо усі коефіцієнти автокореляції часового ряду не є значимими – не виходять за межі довірчого інтервалу, заданого нерівністю 10.23, часовий ряд є стаціонарним.

Після знаходження коефіцієнтів автокореляції здійснюють побудову корелограми та проводять її аналіз для виявлення структури часового ряду.

Автокореляційний аналіз випадкової компоненти дозволяє оцінити правильність побудованої моделі часового ряду.

1. Якщо при побудові моделі часового ряду тренд і сезонну компоненту було виокремлено та описано правильно, корелограма випадкової компоненти не виходить за межі довірчого інтервалу й згасає при $p \to \infty$, де p – порядок автокореляції (лаг).
2. Якщо при при побудові моделі часового ряду тренд було визначено та виокремлено не правильно, автокореляційна функція випадкової компоненти не згасає при $p \to \infty$.
3. Якщо при при побудові моделі часового ряду неправильно визначено та виокремлено сезонну компоненту, автокореляційна функція випадкової компоненти зростає скачками кратно періоду коливань.

Для здійснення ***розрахунку коефіцієнтів автокореляції*** можна скористатися такими формулами MS Excel та MatLab.

1. У MS Excel: *КОРРЕЛ(масив1; масив2) / CORREL(масив1; масив2)*, де *масив1* – діапазон комірок з рівнями аналізованого часового ряду, *масив2* – діапазон комірок з рівнями часового ряду, зсунутими відносно заданого на p періодів.
2. У MatLab: *autocorr(y_t)*, де y_t – вектор зі значеннями рівнів часового ряду.

Приклад 4. За даними щоквартальних значень змінної y за 5 років (табл. 9.5) дослідити динаміку зміни коефіцієнта автокореляції та автокореляційну функцію – корелограму і виявити значущість коефіцієнтів автокореляції.

1. У MatLab сворюємо скрипт, який містить команди для побудови графіку часового ряду, розрахунку коефіцієнтів автокореляції для кожного лагу від 0 до 19 та побудови корелограми – функції, яка містить послідовність коефіцієнтів автокореляції:

```
% y - вектор з рівнями часового ряду
y=[ 7.2 8.0 -3.3 13.2 10.7 10.9 0.4 16.2 14.6 16.2 4.6 21.0 19.5 18.8 7.8 25.0 20.9 23.9 12.3 28.4];
[acf, lag]=autocorr(y); % розрахунок коефіцієнтів автокореляції acf
acf=acf'; lag=lag';
[acf lag] % виведення коефіцієнтів автокореляції, lag – лаг
subplot(2,1,1)
plot(y) % графік часового ряду
```

```
subplot(2,1,2)
autocorr(y)% побудова корелограми
```

У результаті виконання цього скрипта у вікні Command Window буде виведено значення коефіцієнтів автокореляції й у окремому вікні – графіки часового ряду та корелограми. Графік часового ряду зображено у верхній частині рисунку, а графік корелограми – у нижній (рис. 10.7).

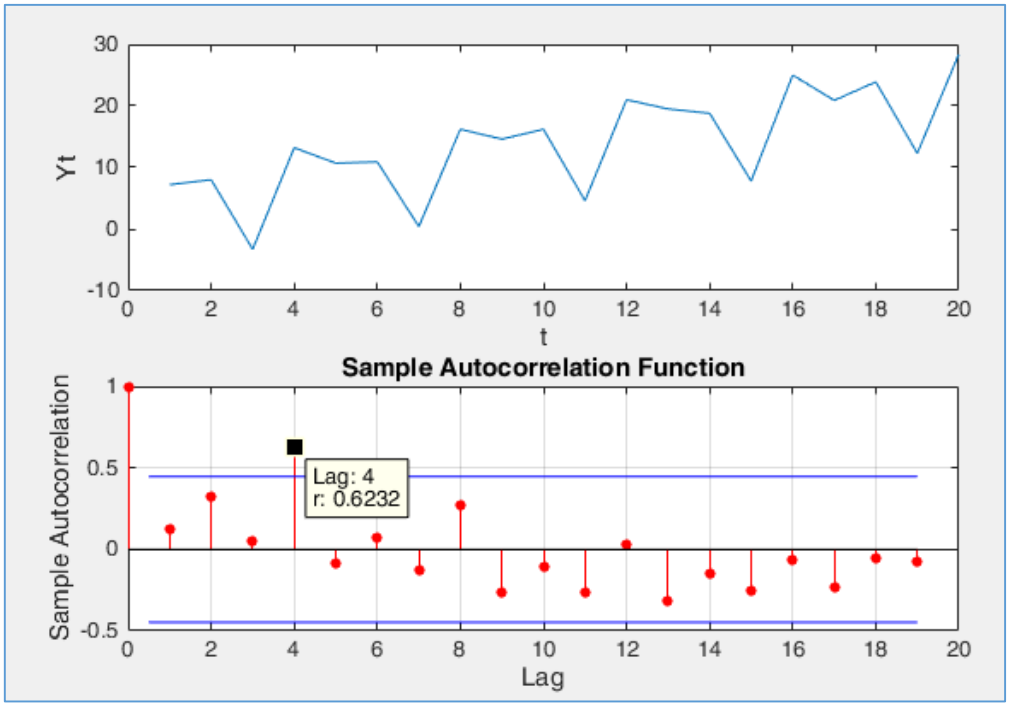


Рис. 10.7. Графік та корелограма часового ряду

2. Аналіз графічного зображення часового ряду показує загальну зростаючу тенденцію зміни значень рівнів ряду та наявність коливань із періодичністю 4 квартали та приблизно рівною амплітудою коливань. Це дає підстави припустити, що часовий ряд може бути описаний за допомогою адитивної моделі, яка містить тренд, сезонну та випадкову компоненти.

У прикладі 3 за допомогою методу перевірки середніх різниць із ймовірністю 95% було достовірно установлено, що даний часовий ряд має тренд. Для подальшого з'ясування структури часового ряду необхідно провести автокореляційний аналіз та визначити значущість коефіцієнтів автокореляції.

3. Аналіз графіку автокореляційної функції показує, що корелограма містить максимальні та мінімальні значення коефіцієнтів автокореляції, які відповідають максимумам та мінімумам часового ряду (рис. 10.7). Найбільшим по модулю виявився коефіцієнт автокореляції 4-го порядку.

4. Аналіз значень коефіцієнтів автокореляції та їх значущості показав, що усі коефіцієнти автокореляції, крім коефіцієнта 4-го порядку, не є значимими. Усі коефіцієнти, крім коефіцієнта 4-го порядку, не виходять за межі довірчого інтервалу [–0,44; 0,44], розрахованого за формулою 10.24 для ряду довжиною $n = 20$, який на графіку корелограми находиться між двома синіми лініями. Це дає підстави стверджувати, що досліджуваний часовий ряд містить сезонні коливання з періодом 4 квартали.

5. Таким чином, проведений аналіз дозволив установити, що у структурі часового ряду є тренд, сезонна компонента з періодом 4 квартали і приблизно рівною амплітудою коливань та випадкова компонента. Тому для моделювання часового ряду доцільно обрати адитивну модель: $y_t = T_t + S_t + E_t$.

10.1.8. Моделювання часового ряду на основі механічного згладжування

Механічні методи згладжування часового ряду використовують значення сусідніх рівнів ряду й автоматичне налагодження на зміну досліджуваної ознаки, не визначаючи аналітичний вид згладженої функції. Побудована таким чином модель наприкінці заданого інтервалу часу буде відображати тенденцію, що склалася на поточний момент часу.

Моделі часового ряду, в основі яких лежать механічні методи згладжування, застосовують для моделювання часових рядів, у яких значення кожного наступного рівня залежить від попередніх значень із запізненням. До основних ***методів механічного згладжування*** відносять: згладжування по двох точках, згладжування методом простої ковзкої середньої, згладжування методом зваженої ковзкої середньої, експоненційне згладжування.

Модель ковзної середньої (англ. *Moving Average, MA*) дозволяє від початкових значень рівнів ряду перейти до середніх на деякому інтервалі значень. Методи згладжування можуть бути такими: простої ковзкої середньої та зваженої ковзкої середньої.

Метод ***простої ковзкої середньої*** (англ. *Simple Moving Average, SMA*):

$$\tilde{y}_t = \frac{\sum_{i=t-p}^{t+p} y_i}{2 \cdot p + 1}, \tag{10.24}$$

де $\tilde{y}_t$ – згладжене значення рівня часового ряду y_t, p – довжина інтервалу згладжування.

При розрахунку згладжених рівнів ряду інтервал „ковзає" вздовж ряду. Згладжування відбувається за рахунок того, що дисперсія середніх значень є меншою в p раз за дисперсію аналізованого часового ряду.

Метод ковзної середньої застосовують для моделювання стаціонарних часових рядів та у разі наявності сезонних коливань. Процедура згладжування приводить до повного усунення коливань у часовому ряді, якщо довжина інтервалу згладжування береться рівною або кратною періоду коливань.

Метод ***зваженої ковзкої середньої*** (англ. *Weighted Moving Average, WMA*):

$$\tilde{y}_t = \frac{\sum_{i=t-p}^{t+p} \alpha_i \cdot y_i}{\sum_{i=t-p}^{t+p} \alpha_i}, \tag{10.25}$$

де α_i – ваги згладжування, $p < t < n - p$, n – довжина часового ряду, інші компоненти формули 10.25 аналогічні компонентам формули 10.24.

Цей метод застосовують, коли рівні ряду змінюються нелінійно. Згладжування всередині інтервалу відбувається не по прямій, а по кривій більш високого порядку – підсумовування членів ряду, що входять в інтервал згладжування, відбувається з вагами, розрахованими за методом найменших квадратів.

При згладжуванні за методом ковзної середньої моментних рядів по $2k+1$ точкам, k точок на початку та в кінці ряду залишають незгладженими. Ці точки або виключають із розгляду (часто небажано), або використовують спеціальні формули згладжування для крайніх точок.

Модель експоненційного згладжування (англ. *Exponential Moving Average, EMA*) базується на ідеї постійного перегляду прогнозних значень в міру надходження фактичних. Модель привласнює експоненційно спадаючі ваги рівням ряду при їх віддалені від поточного рівня ряду. Таким чином значення останніх рівнів часового ряду мають більший вплив на прогнозне значення, чим більше віддалені рівні.

Для часового ряду y_t вихід експоненційного згладжування записують як $\tilde{y}_t$. Вважаючи, що $\tilde{y}_1 = y_1$, найпростіша формула експоненційного згладжування буде мати вигляд:

$$\tilde{y}_t = \alpha \cdot y_t + (1-\alpha) \cdot \tilde{y}_{t-1}, \tag{10.26}$$

де α – коефіцієнт згладжування, $0 < \alpha < 1$.

10.1.9. Авторегресійний аналіз часового ряду

Авторегресійний аналіз застосовують для опису стаціонарних часових рядів.

Авторегресійна модель (англ. *AutoRegressive Model, AR*) є моделлю стаціонарного часового ряду, у якій значення ряду в певний момент часу лінійно залежить від попередніх значень цього ж ряду.

Для часового ряду $y_1, y_2, ..., y_t, ..., y_n$ авторегресійний процес порядку p (AR(p)-процес) задається рівнянням:

$$y_t = a_0 + a_1 y_{t-1} + a_2 y_{t-2} + ... + a_p y_{t-p} + e_t, \tag{10.27}$$

де a_i – коефіцієнти авторегресії, зазвичай $a_0 = 0$, e_t – випадкова величина.

У авторегресійній моделі поточне значення часового ряду y_t є лінійною комбінацією попередніх значень та випадкової похибки.

Основні ***етапи побудови авторегресійної моделі*** є наступними:

1. Перевірка часового ряду на стаціонарність. Якщо ряд виявився нестаціонарним, здійснюють перетворення, які приводять його до стаціонарного виду.
2. Ідентифікація порядку авторегресійної моделі p. Її можна здійснити шляхом аналізу послідовності коефіцієнтів автокреляції часового ряду – корелограми. Авторегресійний часовий ряд має плавно гасаючу автокореляційну функцію. На практиці порядок авторегресійної моделі p частіше всього не є більшим за 2.
3. Оцінка параметрів авторегресійної моделі – коефіцієнтів авторегресії, здійснюється за допомогою методу найменших квадратів або інших методів.
4. Перевірка адекватності побудованої моделі. Якщо рівняння авторегресії для моделювання часового ряду підібране правильно, то випадкова компонента є білим шумом.
5. Застосування моделі для пронозування.

Авторегресійні моделі не призначені для опису часових рядів із тенденцією, однак їх використовують для опису коливань. При аналізі нестаціонарних часових рядів авторегресійна модель об'єднується з іншими методами аналізу, що суттєво розширює сферу їх застосування.

Модель ARMA(p,q) (англ. *AutoRegressive Moving Average Model*) є об'єднанням авторегресійної моделі AR(p) з моделлю ковзного середнього (MA).

Модель ARIMA(p,d,q) (англ. *AutoRegressive Integrated Moving Average Model*) є розширенням моделі ARMA(p,q) для нестаціонарних часових рядів, які перетворюють у стаціонарні з порядком інтеграції *d*.

Порядок інтеграції d – це число, яке показує, скільки разів необхідно застосувати оператор перших різниць, щоб часовий ряд став стаціонарним.

Оператор перших різниць – взяття різниць першого порядку від часового ряду y_t з утворенням ряду перших різниць: $\nabla^1 y_t = y_t - y_{t-1}$.

Якщо утворений часовий ряд перших різниць $\nabla^1 y_t$ є стаціонарним то порядок інтеграції $d = 1$. Якщо ні, від цього ряду беруть перші різниці: $\nabla^2 y_t = \nabla^1 y_t - \nabla^1 y_{t-1}$ та перевіряють утворений ряд на стаціонарність. Якщо ряд буде стаціонарним, стаціонарним то порядок інтеграції $d = 2$. Якщо ні – процедуру знову повторюють, доки утворений ряд не буде стаціонарним.

10.1.10. Основні етапи побудови адитивної моделі часового ряду

Відомо декілька підходів до моделювання часових рядів, що містять тренд та сезонні коливання. Найпростіший підхід – розрахунок значень сезонної компоненти методом ковзної середньої та побудова адитивної $y_t = T_t + S_t + E_t$ або мультиплікативної $y_t = T_t \cdot S_t \cdot E_t$ моделі часового ряду, де T_t – тренд, S_t – сезонна компонента, E_t – випадкова компонента. Етапи побудови адитивної моделі можуть бути застосовані також і для побудови мультиплікативної моделі, оскільки вона зводиться до адитивної моделі логарифмуванням.

Процес побудови моделі включає наступні ***етапи***:

1. ***Попередній аналіз даних***, визначення структури часового ряду.
2. ***Вирівнювання вихідного ряду методом ковзного середнього***. Для цього необхідно:
   а) просумувати рівні ряду послідовно за кожні *і* періодів із зсувом на один момент часу та визначити умовні значення – укрупнити дані;
   б) розділити отримані суми на число, яке дорівнює кількості періодів, що укрупнювалися;
   в) привести ці значення відповідно з фактичними моментами часу: знайти середні значення з двох послідовних ковзних середніх – центровані ковзні середні.
3. ***Розрахунок значень сезонної компоненти*** S_t. Оцінку сезонної компоненти знаходимо як різницю між фактичними рівнями ряду і центрованими ковзними середніми. Після розрахунку оцінки сезонної компоненти для рівнів ряду, необхідно знайти середнє значення цієї компоненти для однойменних періодів-сезонів (напри-

клад, кварталів, місяців). Скореговане значення сезонної компоненти знаходять як різницю між її середньою оцінкою і коригуючим коефіцієнтом. Коригуючий коефіцієнт розраховують як середнє значення середніх сезонної компоненти для однойменних періодів-сезонів.

4. ***Усунення сезонної компоненти*** з вихідних рівнів ряду і отримання вирівняних даних: ці значення розраховують для кожного моменту часу і містять тільки тенденцію (тренд) і випадкову компоненту: $T_t + E_t$. Для усунення сезонної компоненти з рівнів ряду необхідно вирахувати її значення з кожного рівня вхідного ряду за формулою $T_t + E_t = y_t - S_i$.

5. ***Побудова трендових моделей***: формування набору апроксимуючих функцій, чисельне оцінювання їх параметрів, оцінка точності апроксимації, вибір кращої моделі.

6. ***Аналітичне вирівнювання рівнів***, розрахунок значень T_t з використанням обраної трендової моделі.

7. ***Оцінка значень випадкової компоненти*** шляхом розрахунку значень для кожного рівня ряду за формулою: $E_t = y_t - (T_t + S_t)$.

8. ***Перевірка адекватності побудованої моделі, оцінка точності*** прогнозу.

9. ***Прогнозування*** з використанням побудованої моделі часового ряду. Прогнозне значення рівнів часового ряду є сумою трендової та сезонної компонент.

10.1.11. Засоби прогнозування в MS Excel

Створення прогнозу в MS Excel може бути здійснене з використанням наступних інструментальних засобів:

1. Екстраполяція, виконана шляхом ***побудови лінії тренду***. Для цього необхідно:
   a) побудувати точкову діаграму рівнів часового ряду;
   b) викликати контекстне меню побудованої діаграми та обрати пункт *Додати лінію тренду*;
   c) на панелі *Формат лінії тренду*, яка з'явиться праворуч, налаштувати параметри: тип ліній тренду та формат її виведення.
2. Функція $ЛИНЕЙН(y_t, t) / LINEST(y_t, t)$ дозволяє розрахувати коефіцієнти лінійного наближення $y_t = at + b$. За отриманим рівнянням можна здійснювати точковий прогноз на певний період.
3. Функція $ПРЕДСКАЗ\ (t_{new}, y_t, t) / FORECAST\ (t_{new}, y_t, t)$: дозволяє здійснювати точковий прогноз даних на певний період на основі лінійної залежності.
4. Функція $ТЕНДЕНЦИЯ\ (y_t, t, t_{new}) / TREND\ (y_t, t, t_{new})$: дозволяє отримувати прогноз для декількох наступних періодів на основі лінійної залежності.
5. Функція $РОСТ\ (y_t, t, t_{new}) / GROWTN\ (y_t, t, t_{new})$: дозволяє отримувати прогноз для декількох наступних періодів на основі експоненційної залежності.

Аргументи функцій: y_t – рівні часового ряду, t – періоди часу, t_{new} – наступний період (періоди) часу, для якого (яких) здійснюється прогноз.

Спосіб введення функцій масиву: описані вище функції є функціями масиву, тому для їх правильного введення необхідно:

a) ввести у комірку потрібну функцію з аргументами та натиснути клавішу *Enter*;
b) виділити комірку з формулою та необхідну кількість суміжних комірок, натиснути клавішу *F2* а потім одночасно клавіши *Shift + Ctrl + Enter*.

6. ***Лист прогнозу*** MS Excel: дозволяє миттєво створювати прогноз за даними часового ряду на основі заданих параметрів: довірчого інтервалу, початку та кінця прогнозного періоду, врахування сезонності (вручну та автоматично), способів заповнення пропущених значень та обробки дублікатів.

10.2. ПОБУДОВА АДИТИВНОЇ ТРЕНД-СЕЗОННОЇ МОДЕЛІ ЧАСОВОГО РЯДУ

Приклад 5. Здійснити побудову моделі часового ряду за даними щоквартальних значень змінної *y* за 5 років, представленими у таблиці 10.5, та здійснити прогнозування значень із горизонтом прогнозу – 1 рік.

10.2.1. Попередній аналіз даних, виявлення структури часового ряду

1. У MS Excel формуємо електронну таблицю з даними часового ряду та будуємо його графік – точкову діаграму (t, y_t) (рис. 10.8).

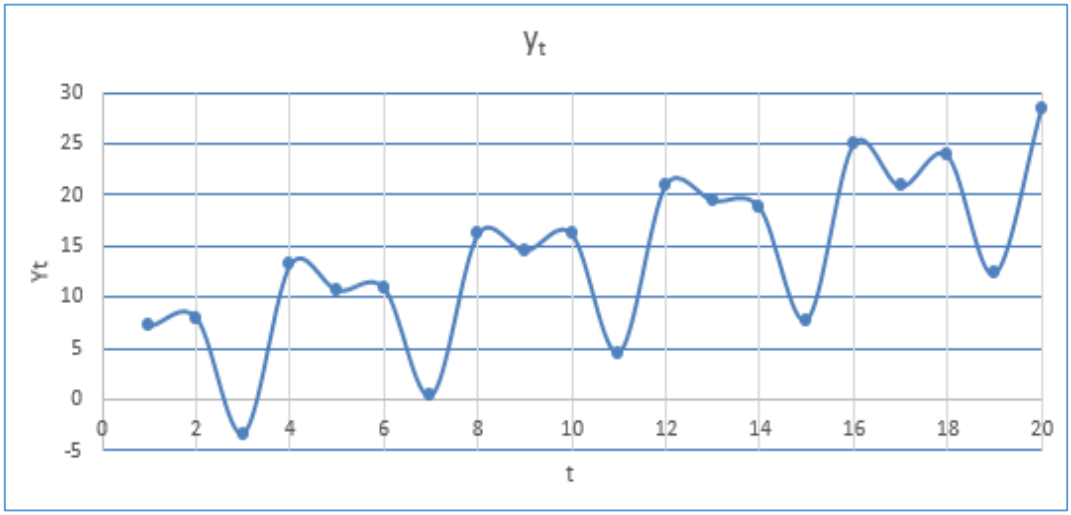


Рис. 10.8. Графік часового ряду

2. Аналіз графічного зображення часового ряду дає підстави стверджувати, що ряд містить сезонні коливання з періодичністю 4 квартали і приблизно рівною амплітудою коливань та зростаючу загальну тенденцію зміни значень рівнів ряду. Такий же висновок було зроблено при здійсненні аналізу корелограми та значущості коефіцієнтів автокореляції цього часового ряду у прикладі 4.

3. У прикладі 3 із використанням методу перевірки середніх різниць було достовірно установлено, що цей часовий ряд має тренд.

4. Отже, для моделювання часового ряду може бути обрана адитивна модель, яка містить тренд, сезонну та випадкову компоненти: $y_t = T_t + S_t + E_t$.

10.2.2. Вирівнювання вихідних рівнів ряду методом ковзної середньої

Розглянемо послідовність розрахунків для вирівнювання вихідних рівнів ряду методом ковзної середньої (рис. 10.9).

1. Розрахуємо суми послідовних членів ряду:
   a) у комірку D8 введемо формулу для обчислення суми перших 4-х членів ряду: =C7+C8+C9+C10;
   b) у комірку D9 – формулу для обчислення суми 2-го, 3-го, 4-го та 5-го членів ряду: =C8+C9+C10+C11;
   c) у наступних комірках стовпця D електронної таблиці вводимо формули для обчислення суми 3-го, 4-го, 5-го та 6-го членів ряду та ін. до комірки D24, в яку необхідно ввести формулу для обчислення суми 17-го, 18-го, 19-го та 20-го членів ряду: =C23+C24+C25+C26.
2. У комірках стовпця E електронної таблиці розраховуємо ковзну середню за 4 квартали:
   a) у комірку E8 введемо формулу: =D8/4;
   b) у комірку E9 – формулу: =D9/4;
   c) далі – аналогічно до комірки E24, в яку необхідно ввести формулу: =D24/4.
3. У комірках стовпця F електронної таблиці розраховуємо центровану ковзну середню:
   a) у комірку F9 введемо формулу: =(E8+E9)/2;
   b) у комірку F10 введемо формулу: =(E9+E10)/2;
   c) далі – аналогічно до комірки F24, в яку необхідно ввести формулу: =(E23+E24)/2.
4. У комірках стовпця G електронної таблиці розраховуємо оцінку сезонної компоненти як різниці між значенням рівня ряду та значення відповідної рівню центрованої ковзної середньої:
   a) у комірку G9 введемо формулу: =C9-F9;
   b) далі – аналогічно до комірки G24, в яку необхідно ввести формулу: =C24-F24.

10.2.3. Розрахунок середніх оцінок сезонної компоненти по кварталам

1. Сформуємо електронну таблицю, яка буде містити значення оцінок сезонної компоненти за кварталами та роками (рис. 10.10).

№ кварталу, t	y_t	Разом за 4 квартали	Ковзна середня за 4 квартали	Центрована ковзна середня	Оцінка сезонної компоненти
1	7,2				
2	8	25,1	6,28		
3	-3,3	28,6	7,15	6,7125	-10,013
4	13,2	31,5	7,88	7,5125	5,688
5	10,7	35,2	8,80	8,3375	2,363
6	10,9	38,2	9,55	9,175	1,725
7	0,4	42,1	10,53	10,0375	-9,638
8	16,2	47,4	11,85	11,1875	5,013
9	14,6	51,6	12,90	12,375	2,225
10	16,2	56,4	14,10	13,5	2,700
11	4,6	61,3	15,33	14,7125	-10,113
12	21	63,9	15,98	15,65	5,350
13	19,5	67,1	16,78	16,375	3,125
14	18,8	71,1	17,78	17,275	1,525
15	7,8	72,5	18,13	17,95	-10,150
16	25	77,6	19,40	18,7625	6,238
17	20,9	82,1	20,53	19,9625	0,938
18	23,9	85,5	21,38	20,95	2,950
19	12,3				
20	28,4				

Рис. 10.9. Вирівнювання вхідних рівнів ряду методом ковзної середньої

Показник	Рік	Номер кварталу, і				
		I	II	III	IV	
	1			-10,013	5,688	
	2	2,363	1,725	-9,638	5,013	
	3	2,225	2,700	-10,113	5,350	
	4	3,125	1,525	-10,150	6,238	
	5	0,938	2,950			
Σ за і-й квартал (за всі роки)	x	8,650	8,900	-39,913	22,288	Σ
Середня оцінка сезонної компоненти для і-го кварталу, $S_{iср}$	x	2,163	2,225	-9,978	5,572	-0,019
Скорегована сезонна компонента, S_i	x	2,167	2,230	-9,973	5,577	0,000
Визначення корегуючого коефіцієнта	K	-0,00469				

Рис. 10.10. Розрахунок середніх оцінок сезонної компоненти

2. Розраховуємо суму оцінок за і-й квартал за всі роки:

a) за 1-й квартал – у комірку K35 введемо: =СУММ(K30:K34);
b) за 2-й квартал – у комірку L35 введемо: =СУММ(L30:L34);
c) за 3-й квартал – у комірку M35 введемо: =СУММ(M30:M34);
d) за 4-й квартал – у комірку N35 введемо: =СУММ(N30:N34).

3. Розраховуємо середню оцінку сезонної компоненти за квартали:

a) за 1-й квартал – у комірку K36 введемо: =K35/4;
b) за 2-й квартал – у комірку L36 введемо: =L35/4;
c) за 3-й квартал – у комірку M36 введемо: =M35/4;
d) за 4-й квартал – у комірку N36 введемо: =N35/4.

4. Розрахуємо коригуючий коефіцієнт *k*. Для даної моделі маємо:

2,163 + 2,225 – 9,978 + 5,572 = -0,019, тоді *k* = -0,019/4 = -0,00469.

5. Розрахуємо скориговані значення сезонної компоненти як різницю між її середньою оцінкою і коригуючим коефіцієнтом $S_i = \overline{S}_i - k$.

6. Перевіримо рівність нулеві суми значень сезонної компоненти: 2,167 + 2,230 – 9,973 + 5,577 = 0 та занесемо їх у таблицю для відповідних кварталів.

10.2.4. Усунення сезонної компоненти з вихідних рівнів часового ряду

1. Сформуємо електронну таблицю, яка буде містити розраховані значення компонент моделі (рис. 10.11).

2. Для усунення сезонної компоненти необхідно вирахувати її значення з кожного рівня вихідного ряду $T_t + E_t = y_t - S_t$ (комірки M43:M62). Ці значення розраховуються для кожного періоду часу і містять тільки тенденцію та випадкову компоненту.

	J	K	L	M
41				
42	*t*	y_t	S_t	$T_t+E_t = y_t - S_t$
43	1	7,2	2,167	5,033
44	2	8	2,230	5,770
45	3	-3,3	-9,973	6,673
46	4	13,2	5,577	7,623
47	5	10,7	2,167	8,533
48	6	10,9	2,230	8,670
49	7	0,4	-9,973	10,373
50	8	16,2	5,577	10,623
51	9	14,6	2,167	12,433
52	10	16,2	2,230	13,970
53	11	4,6	-9,973	14,573
54	12	21	5,577	15,423
55	13	19,5	2,167	17,333
56	14	18,8	2,230	16,570
57	15	7,8	-9,973	17,773
58	16	25	5,577	19,423
59	17	20,9	2,167	18,733
60	18	23,9	2,230	21,670
61	19	12,3	-9,973	22,273
62	20	28,4	5,577	22,823

Рис. 10.11. Усунення сезонної компоненти з вихідних рівнів часового ряду

10.2.5. Побудова трендових моделей

1. Для визначення типу моделі тренду побудуємо точкову діаграму для вирівняних даних $T_t + E_t = y_t - S_t$ (рис. 10.12). Визиваємо контекстне меню діаграми та обираємо пункт *Додати лінію тренду*.

2. У вікні *Формат лінії тренду*, яке з'явиться праворуч (рис. 10.13), налаштовуємо параметри та формат трендової моделі: тип ліній тренду – обираємо *Лінійна*, прогноз вказуємо на 4 періоди вперед, ставимо прапорці навпроти опцій – показувати рівняння на діаграмі та розмістити коефіцієнт детермінації R^2.

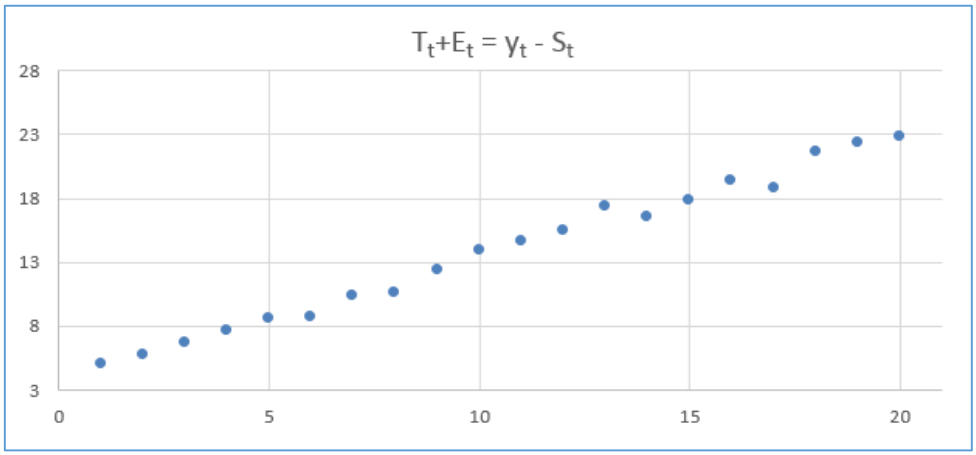


Рис. 10.12. Точкова діаграма вирівняних даних $T_t + E_t = y_t - S_t$

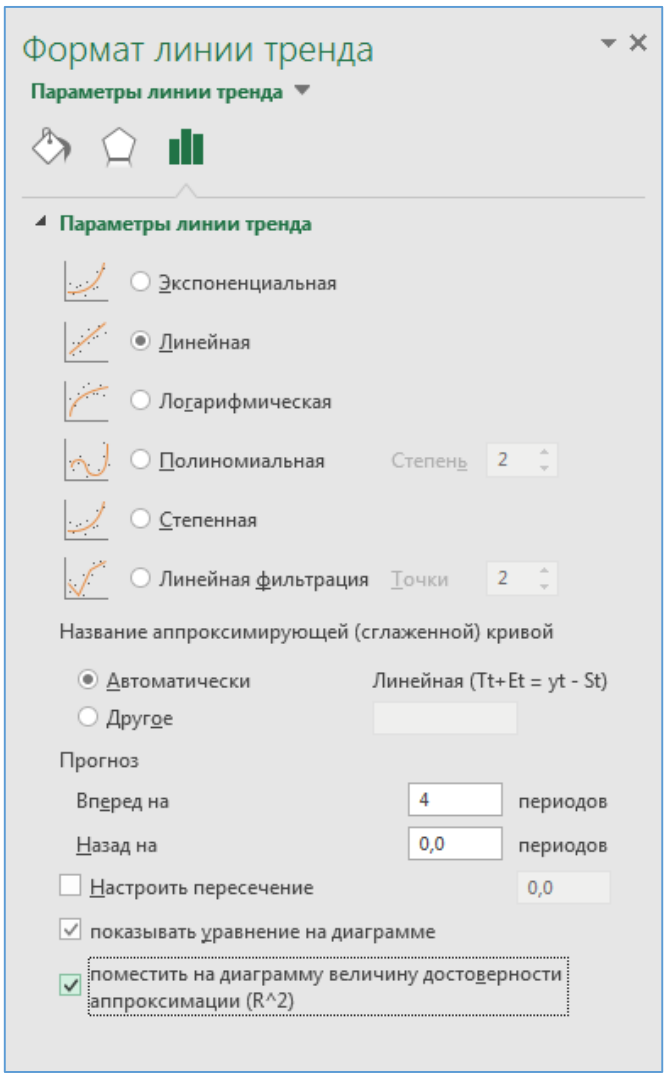


Рис. 10.13. Вікно Формат лінії тренду

3. На діаграмі з’явиться лінійна апроксимуюча функція з прогнозом на 4 періоди вперед, її рівняння та коефіцієнт детермінації, який показує точність апроксимації функцією вирівняних даних (рис. 10.14).

4. Для формування набору апроксимуючих функцій, чисельного оцінювання їх параметрів та оцінки точності апроксимації діаграму вирівняних даних (рис. 10.12) копіюємо та виконуємо декілька раз пункти 2, 3 і 4, обираючи для апроксимації по черзі експоненціальну, логарифмічну, степеневу та поліноміальну функції. На кожній із діаграм з'явиться апроксимуюча функція з прогнозом на 4 періоди вперед, рівнянням та коефіцієнтом детермінації (рис. 10.15 – рис. 10.18).

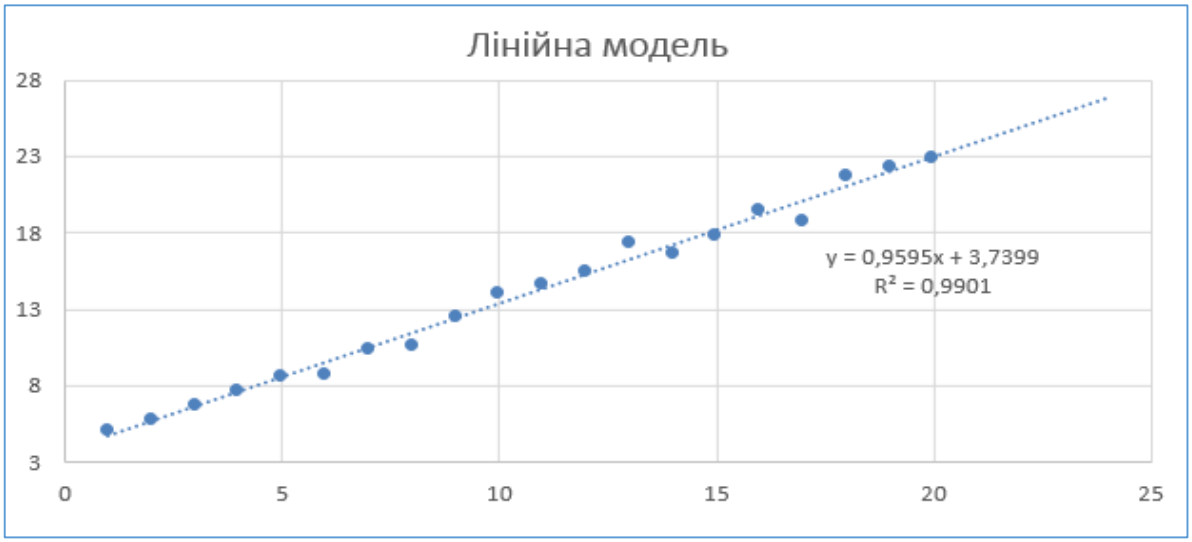


Рис. 10.14. Лінійна трендова модель

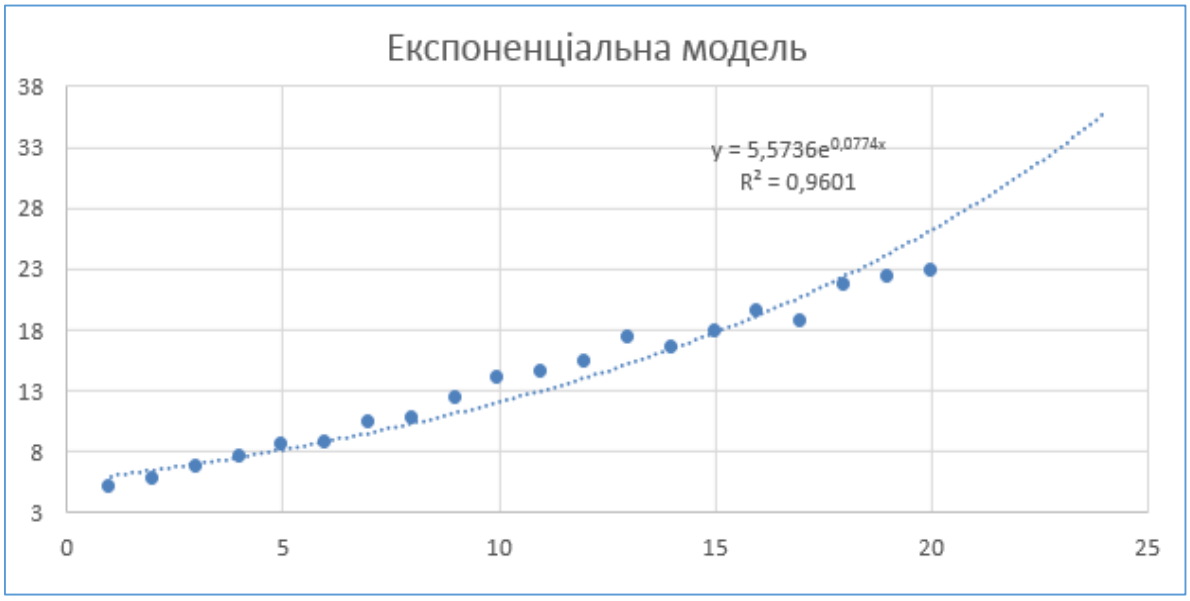


Рис. 10.15. Експоненціальна трендова модель

5. Здійснюємо порівняння коефіцієнтів детермінації та робимо висновок, що кращими моделями є лінійна та поліноміальна, оскільки саме для них рівень апроксимації вирівняного ряду є найвищим. Для лінійної моделі коефіцієнт детермінації $R^2 = 0{,}9901$, для поліноміальної моделі $R^2 = 0{,}9912.$ Різниця між коефіцієнтами є незначною, тому для подальшої побудови моделі часового ряду обираємо лінійну трендову модель, задану рівнянням $\hat{T}_t = 0{,}9595 \cdot t + 3{,}7399.$

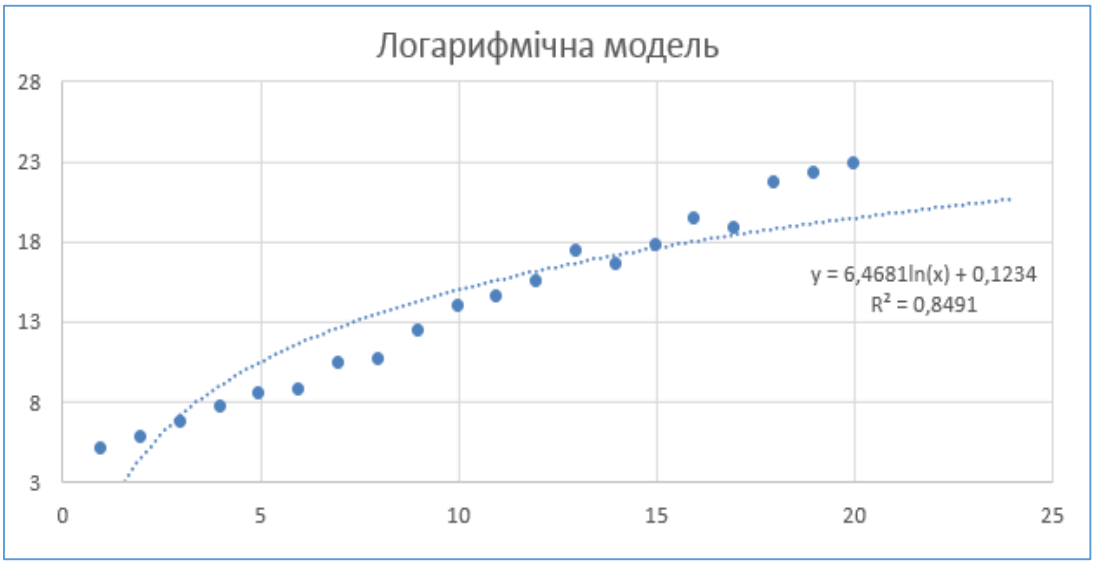


Рис. 10.16. Логарифчічна трендова модель

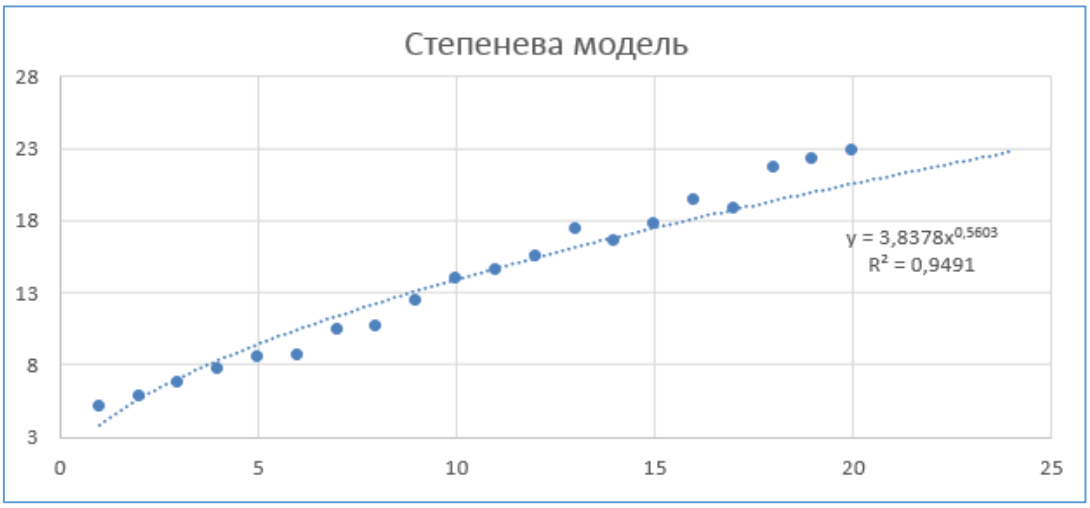


Рис. 10.17. Степенева трендова модель

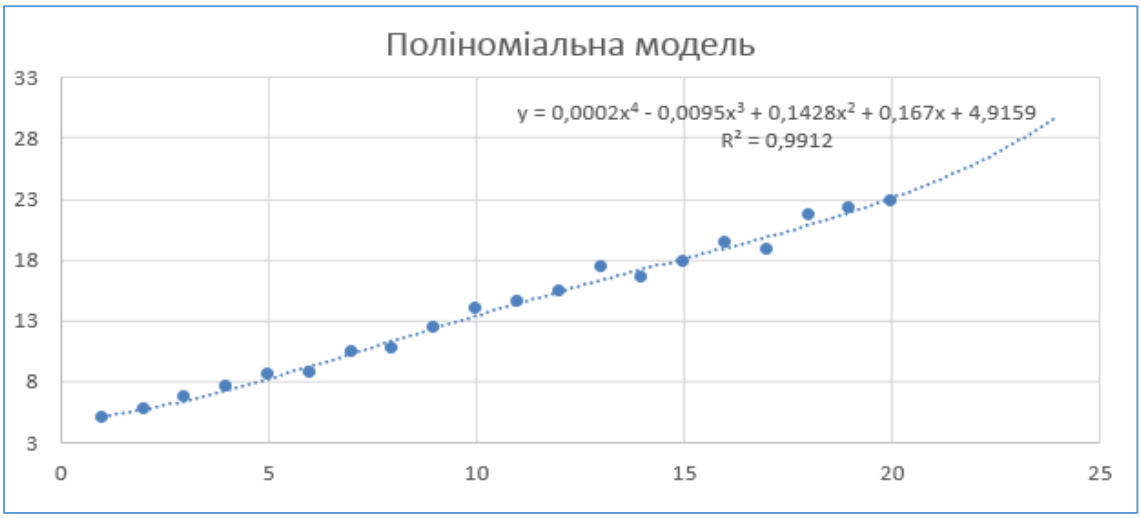


Рис. 10.18. Поліноміальна трендова модель

10.2.6. Аналітичне вирівнювання рівнів ряду, оцінка значень випадкової компоненти

1. Визначимо компоненту $\hat{T}_t$ даної моделі. Для цього проведемо аналітичне вирівнювання ряду $\hat{T}_t + S_t$ за допомогою лінійного тренду $\hat{T}_t = 0{,}9595 \cdot t + 3{,}7399.$

2. У комірки V56 і W56 введемо коефіцієнти лінійної функції $a = 0{,}9595$ і $b = 3{,}7399$ (рис. 10.19). Продовжимо формування таблиці для розрахунку компонент часового ряду (рис. 10.20), доповнивши таблицю на рисунку 10.11 стовпцем розрахунку трендової компоненти $\hat{T}_t = 0{,}9595 \cdot t + 3{,}7399$ для кожного рівня ряду (комірки V64:W64).

	U	V	W	X	Y	Z
62	Розрахунок коефіцієнтів тренду Y = at+b					
63		*a*	*b*			
64		0,9595	3,7399			
65						

Рис. 10.19. Коефіцієнти лінійного рівняння тренду

4. Знайдемо значення рівнів ряду, отримані за адитивною моделлю. Для цього додаємо до рівнів $\hat{T}_t$ значення сезонної компоненти для відповідних кварталів $\hat{T}_t + S_t$ (комірки O43:O62) та оцінюємо значення випадкової компоненти – розраховуємо абсолютну похибку за формулою $E_t = y_t - (\hat{T}_t + S_t)$ (комірки P43:P62).

5. Проведена декомпозиція часового ряду (виділення компонент) дозволяє побудувати діаграми часового ряду та його компонент (рис. 10.21): лінії тренду, з якої вилучено сезонну компоненту, сезонної та випадкової компонент, (користуючись даними, розрахованими у таблиці на рисунку 10.20).

	J	K	L	M	N	O	P
41							
42	*t*	y_t	S_t	$T_t+E_t = y_t - S_t$	$\check{T}_t$	$\check{T}t+S_t$	$E = y_t - (\check{T}t+St)$
43	1	7,2	2,167	5,033	4,699	6,867	0,333
44	2	8	2,230	5,770	5,659	7,889	0,111
45	3	-3,3	-9,973	6,673	6,618	-3,355	0,055
46	4	13,2	5,577	7,623	7,578	13,154	0,046
47	5	10,7	2,167	8,533	8,537	10,705	-0,005
48	6	10,9	2,230	8,670	9,497	11,727	-0,827
49	7	0,4	-9,973	10,373	10,456	0,483	-0,083
50	8	16,2	5,577	10,623	11,416	16,992	-0,792
51	9	14,6	2,167	12,433	12,375	14,543	0,057
52	10	16,2	2,230	13,970	13,335	15,565	0,635
53	11	4,6	-9,973	14,573	14,294	4,321	0,279
54	12	21	5,577	15,423	15,254	20,830	0,170
55	13	19,5	2,167	17,333	16,213	18,381	1,119
56	14	18,8	2,230	16,570	17,173	19,403	-0,603
57	15	7,8	-9,973	17,773	18,132	8,159	-0,359
58	16	25	5,577	19,423	19,092	24,668	0,332
59	17	20,9	2,167	18,733	20,051	22,219	-1,319
60	18	23,9	2,230	21,670	21,011	23,241	0,659
61	19	12,3	-9,973	22,273	21,970	11,997	0,303
62	20	28,4	5,577	22,823	22,930	28,506	-0,106

Рис. 10.20. Розрахунок компонент адитивної моделі часового ряду

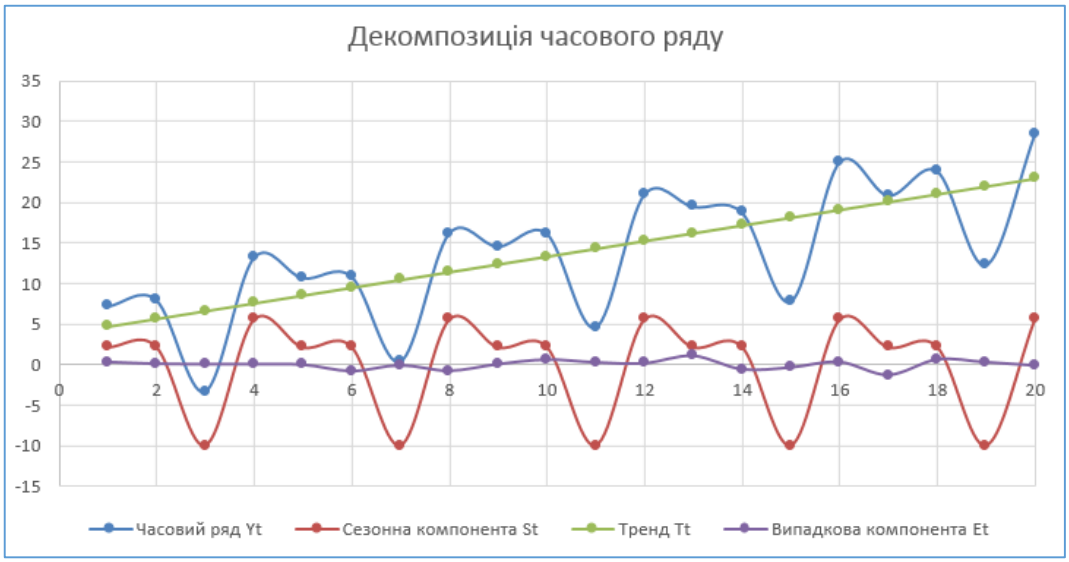


Рис. 10.21. Графіки часового ряду та його компонент

10.2.7. Перевірка адекватності побудованої моделі, оцінка точності прогнозу

1. Для оцінки адекватності побудованої моделі часового ряду імпортуємо значення випадкової компоненти – стопця E з таблиці MS Excel (рис. 10.20, комірки P43:P62) до MatLab та з допомогою функції $autocorr(E_t)$ побудуємо її корелограму.

2. Аналіз графіка автокореляційної функції показує, що корелограма містить чергування затухаючих додатних та від'ємних значень коефіцієнтів автокореляції (рис. 10.22). Усі коефіцієнти автокореляції не є значимими – не виходять за межі довірчого інтервалу, який на графіку корелограми знаходиться між двома синіми лініями. Отже, випадкова компонента є стаціонарним рядом, а побудована модель часового ряду є адекватною.

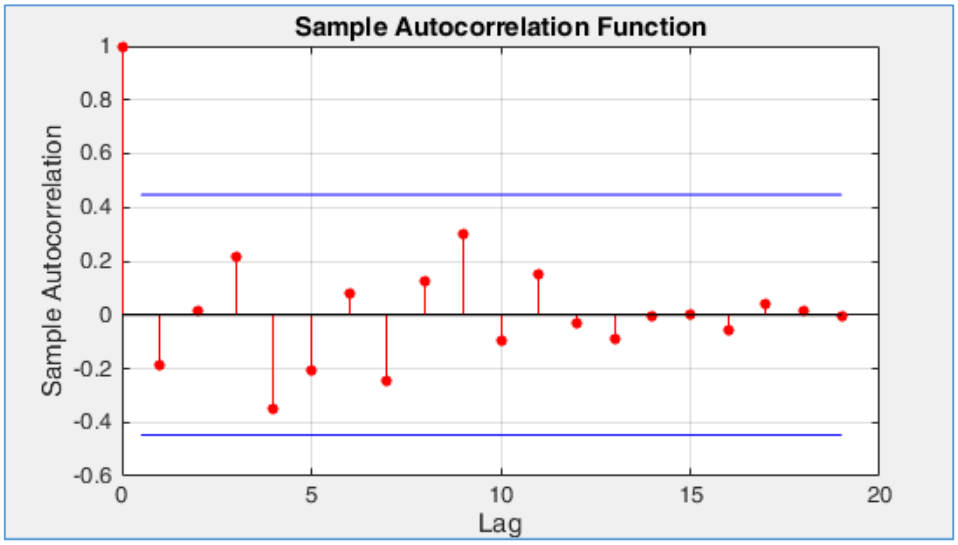


Рис. 10.22. Оцінка адекватності моделі часового ряду

3. Для оцінки точності прогнозу за побудованою моделлю часового ряду розраховуємо E^2 (комірки Q43:Q62) і знаходимо суму квадратів похибок та розрахуємо значення $|E|$ та $|E/y_t|$ (рис. 10.23). Аналізуємо отримані результати оцінки прогнозної якості побудованої моделі:

а) ***MAD*** – середнє абсолютне відхилення розраховане за формулою 10.12: в середньому прогнозовані значення будуть відхилятися від фактичних у більшу або меншу сторону на 0,4 (формула у комірці R64: *=СРЗНАЧ(R43:R62)*.

b) ***MSE*** – середня квадратична помилка становить 0,3046;

c) ***MAPE*** – середня похибка апроксимації, розрахована за формулою 10.13, становить 3,96%, що свідчить про високу точність прогнозу побудованої моделі (формула у комірці S66: *=S64*100*).

d) ***MFE*** – середня похибка, розрахована за формулою 10.11, також близька до 0 та має додатній знак, що означає незначне завищення показника, в цілому прогноз близький до незміщеного (формула у комірці P64: *=СРЗНАЧ(P43:P62)*.

	J	K	L	M	N	O	P	Q	R	S
42	t	y_t	S_t	$T_t+E_t = y_t - S_t$	$\check{T}_t$	$\check{T}t+S_t$	$E = y_t - (\check{T}t+St)$	E^2	$\|E\|$	$\|E/y_t\|$
43	1	7,2	2,167	5,033	4,699	6,867	0,333	0,111	0,333	0,046
44	2	8	2,230	5,770	5,659	7,889	0,111	0,012	0,111	0,014
45	3	-3,3	-9,973	6,673	6,618	-3,355	0,055	0,003	0,055	0,017
46	4	13,2	5,577	7,623	7,578	13,154	0,046	0,002	0,046	0,003
47	5	10,7	2,167	8,533	8,537	10,705	-0,005	0,000	0,005	0,000
48	6	10,9	2,230	8,670	9,497	11,727	-0,827	0,683	0,827	0,076
49	7	0,4	-9,973	10,373	10,456	0,483	-0,083	0,007	0,083	0,207
50	8	16,2	5,577	10,623	11,416	16,992	-0,792	0,628	0,792	0,049
51	9	14,6	2,167	12,433	12,375	14,543	0,057	0,003	0,057	0,004
52	10	16,2	2,230	13,970	13,335	15,565	0,635	0,404	0,635	0,039
53	11	4,6	-9,973	14,573	14,294	4,321	0,279	0,078	0,279	0,061
54	12	21	5,577	15,423	15,254	20,830	0,170	0,029	0,170	0,008
55	13	19,5	2,167	17,333	16,213	18,381	1,119	1,253	1,119	0,057
56	14	18,8	2,230	16,570	17,173	19,403	-0,603	0,363	0,603	0,032
57	15	7,8	-9,973	17,773	18,132	8,159	-0,359	0,129	0,359	0,046
58	16	25	5,577	19,423	19,092	24,668	0,332	0,110	0,332	0,013
59	17	20,9	2,167	18,733	20,051	22,219	-1,319	1,739	1,319	0,063
60	18	23,9	2,230	21,670	21,011	23,241	0,659	0,435	0,659	0,028
61	19	12,3	-9,973	22,273	21,970	11,997	0,303	0,092	0,303	0,025
62	20	28,4	5,577	22,823	22,930	28,506	-0,106	0,011	0,106	0,004
63	Σ	276,3	0	276,3	276,293	276,29	0,007	6,092	8,193	0,793
64	Середнє	13,815	0	13,815	13,81465	13,815	0,0003	0,3046	0,410	0,0396
65							**MFE**	**MSE**	**MAD**	
66									**MAPE=**	3,963

Рис. 10.23. Розрахунок оцінок точності прогнозу

Отримані оцінки свідчать про те, що побудована модель має високу точність, а отриманий на її основі прогноз близький до незміщеного.

10.2.8. Прогнозування з використанням побудованої моделі часового ряду

1. За допомогою побудованої адитивної тренд-сезонної моделі часового ряду здійснимо прогнозування значень з горизонтом прогнозу в 1 рік – на 4 квартали вперед.

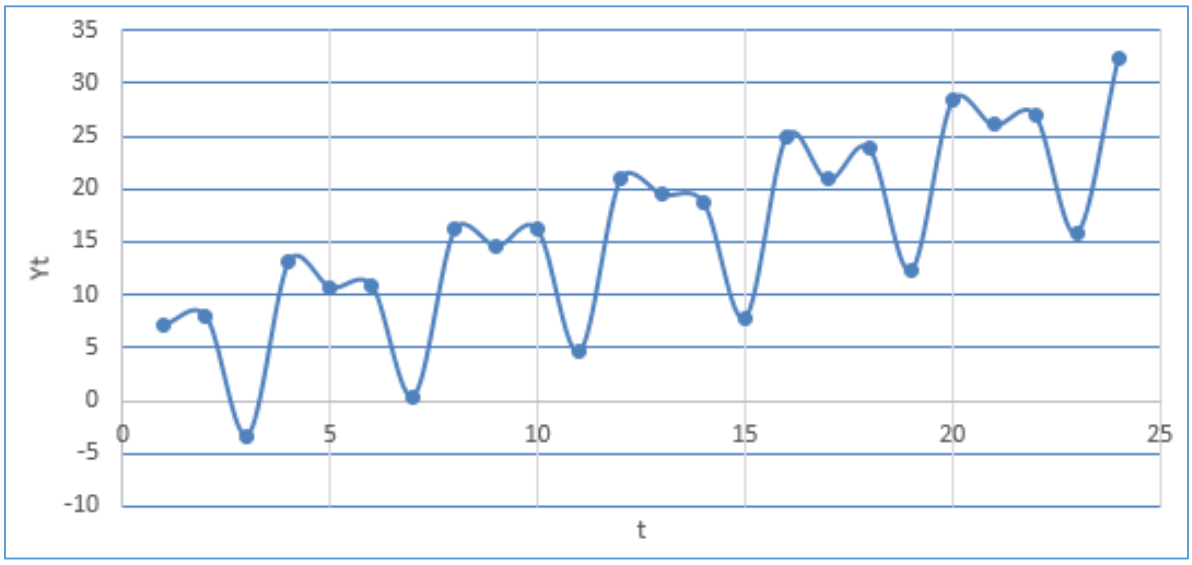
2. Сформуємо таблицю MS Excel для розрахунку значень часового ряду для наступних після $t = 20$ чотирьох періодів, ввівши у стовпці t значення: 21, 22, 23 та 24. Значення у стовці $\hat{T}_t$ розраховуємо за формулою $\widetilde{T}_t = 0{,}9595 \cdot t + 3{,}7399.$ У стовпці S_t вводимо відповідні їм значення сезонної компоненти (рис. 10.24, а).

3. Прогнозовані значення рівнів часового ряду $\hat{y}_t$ є сумою компоненти $\hat{T}_t$, розрахованої за побудованою лінійною трендовою моделлю та відповідного значення сезонної компоненти: $\hat{y}_t = \hat{T}_t + S_t$. Ці значення можуть відхилятися від фактичних у більшу або меншу сторону не більше ніж на 0,4, оскільки розраховане середнє абсолютне відхилення MAD =0,4.

4. Побудуємо графік часового ряду (t, y_t), де перші 20 періодів будуть містити фактичні значення рівнів часового ряду, а наступні 4 після них – значення, спрогнозовані за побудованою моделлю (рис. 10.24, б).

	J	K	L	M
68	t	$\check{T}_t$	S_t	$\hat{y}t=\check{T}t+S_t$
69	21	23,889	2,167	26,057
70	22	24,849	2,230	27,079
71	23	25,808	-9,973	15,835
72	24	26,768	5,577	32,344

а) розрахунок прогнозованих значень часового ряду



б) графічне зображення часового ряду та прогнозованих значень

Рис. 10.24. Здійснення прогнозу за побудованою моделлю

10.3. ПРОГНОЗУВАННЯ З ВИКОРИСТАННЯМ ЛИСТА ПРОГНОЗУ MS EXCEL

У MS Excel є можливість для швидкого створення прогнозу з використанням інструментального засобу ***Лист прогнозу***.

Приклад 6. За даними часового ряду, що містить щоквартальні значення змінної *y* за 5 років, представлені у таблиці 10.5, здійснити побудову прогнозу на 3 наступні роки.

1. Для побудови прогнозу необхідно виділити дані з періодами та значеннями часового ряду: B7:C26 та обрати вкладку *Дані* – групу *Прогноз – Лист прогнозу*. Відкриється вікно *Создание листа прогноза* (рис. 10.25).

2. У вікні *Создание листа прогноза* необхідно натиснути кнопку *Параметри* та задати потрібні параметри прогнозу:

a) початок прогнозу: *20*;
b) довірчий інтервал: *95%*;
c) сезонність: *установити вручну 4*;
d) заповнити відсутні точки: *інтерполяція*;
e) об’єднати дублікати даних: *середнє*;
f) завершення прогнозу: *32*.

3. Натиснувши кнопку  у правому верхньому куті вікна *Создание листа прогноза*, можна переглянути діаграму з прогнозованими значеннями (рис. 10.26).

4. Натиснути кнопку *Создать*, відкриється новий робочий аркуш з відображенням результату прогнозу часового ряду у графічному вигляді та у вигляді таблиці (рис. 10.27).

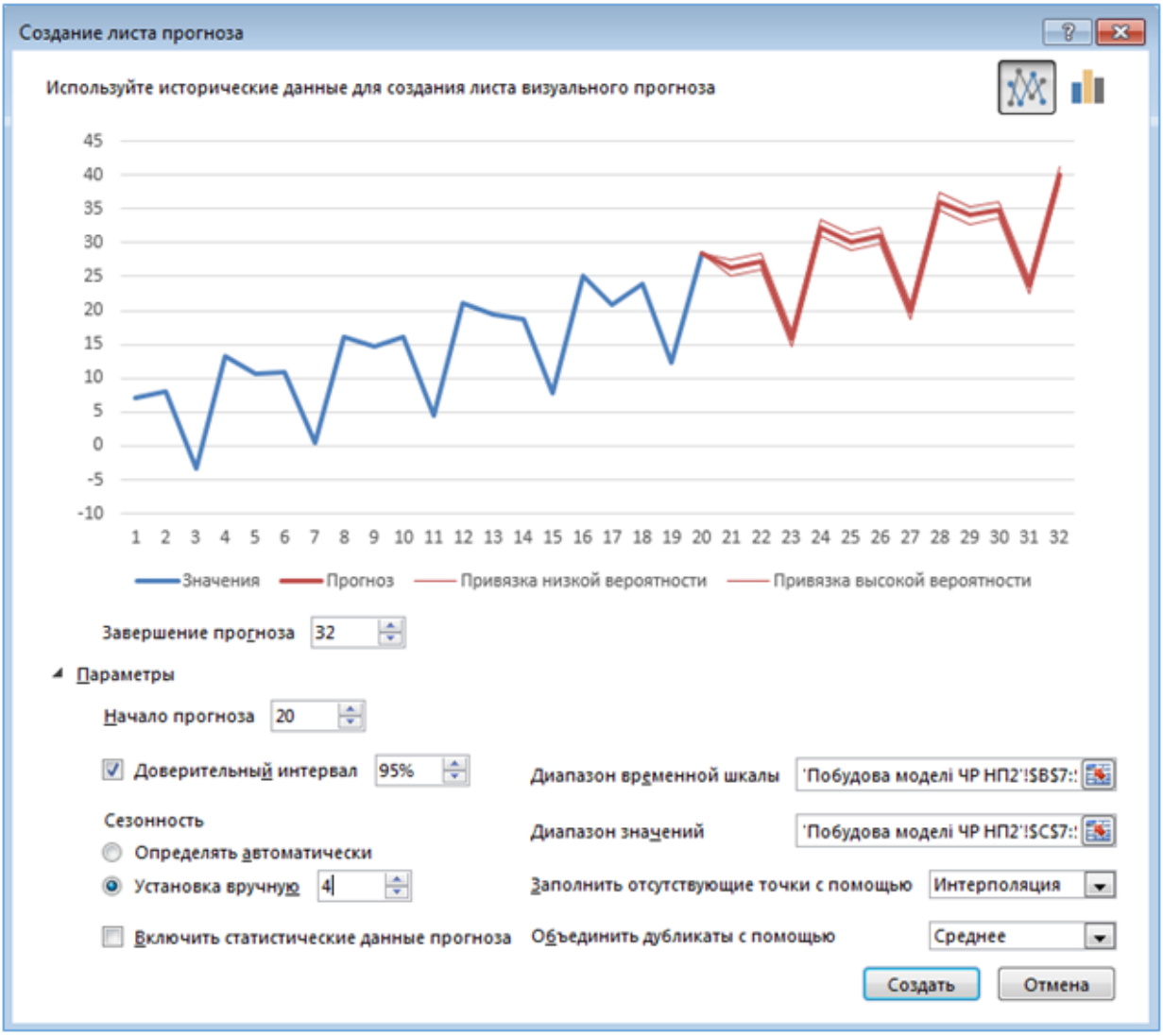


Рис. 10.25. Вікно Створення листа прогнозу

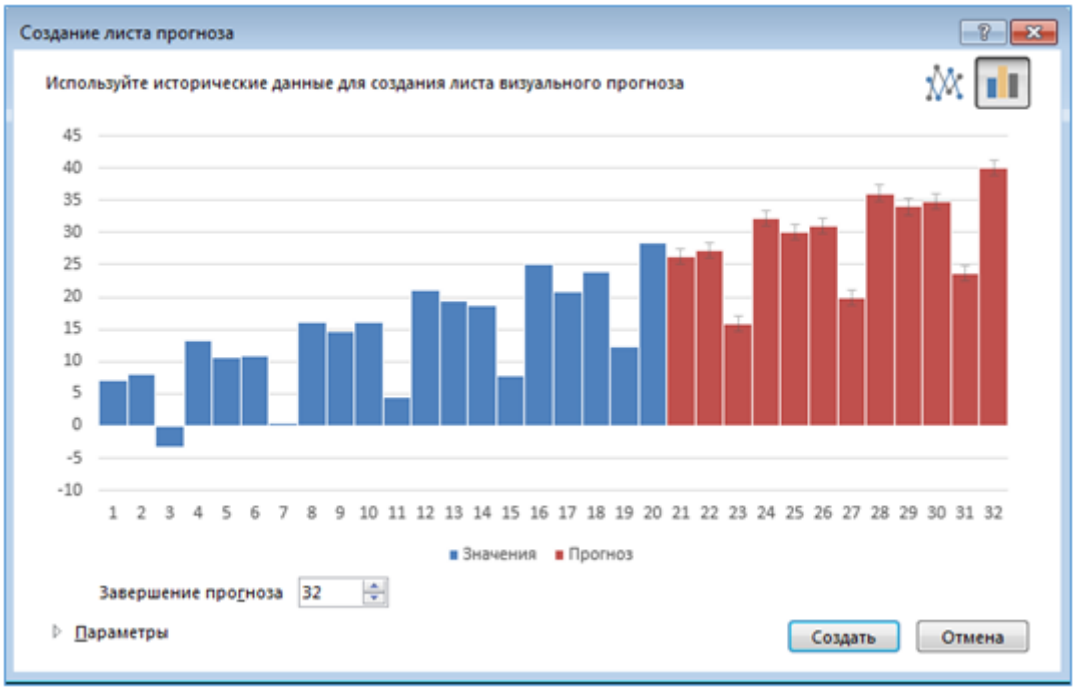
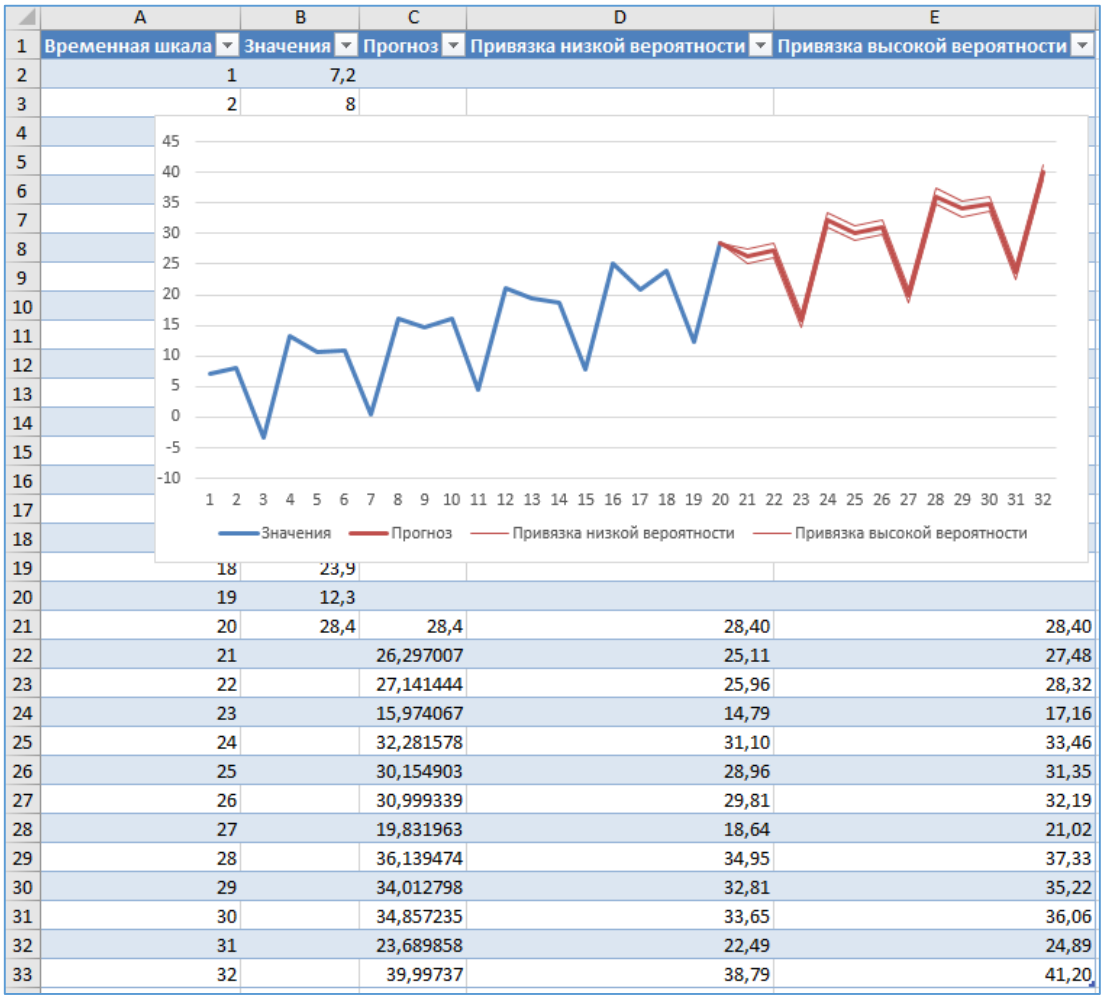


Рис. 10.26. Діаграма часового ряду з прогнозованими значеннями



	A	B	C	D	E
1	Временная шкала	Значения	Прогноз	Привязка низкой вероятности	Привязка высокой вероятности
2	1	7,2			
3	2	8			
19	18	23,9			
20	19	12,3			
21	20	28,4	28,4	28,40	28,40
22	21		26,297007	25,11	27,48
23	22		27,141444	25,96	28,32
24	23		15,974067	14,79	17,16
25	24		32,281578	31,10	33,46
26	25		30,154903	28,96	31,35
27	26		30,999339	29,81	32,19
28	27		19,831963	18,64	21,02
29	28		36,139474	34,95	37,33
30	29		34,012798	32,81	35,22
31	30		34,857235	33,65	36,06
32	31		23,689858	22,49	24,89
33	32		39,99737	38,79	41,20

Рис. 10.27. Робочий аркуш із результатами прогнозу

10.4. ЗАВДАННЯ ДЛЯ САМОСТІЙНОЇ РОБОТИ

Завдання 1. За даними часового ряду, що містить щоквартальні значення змінної y за 4 роки, представлені у таблиці 10.7 (за варіантами), необхідно здійснити аналіз даних із метою побудови адитивної (чи мультиплікативної) моделі часового ряду:

1) побудувати графік часового ряду та здійснити попередній аналіз даних: виявлення аномальних відхилень, аналіз динаміки зміни значень рівнів ряду, визначення структури часового ряду;

2) установити наявність тренду з використанням методу перевірки середніх різниць або за допомогою критерію серій;

3) побудувати корелограму та провести автокореляційний аналіз часового ряду для уточнення його структури;

4) здійснити згладжування рівнів ряду: вирівнювання вихідного ряду методом ковзного середнього;

5) розрахувати значення сезонної компоненти та усунути її з вихідних рівнів ряду, провести декомпозицію часового ряду;

6) здійснити побудову трендових моделей: формування набору апроксимуючих функцій, чисельне оцінювання їх параметрів, оцінку точності апроксимації, вибір кращої моделі;

7) провести аналітичне вирівнювання, розрахунок значень T_t з використанням обраної трендової моделі;

8) дослідити випадкову компоненту на стаціонарність, перевірити адекватність побудованої моделі часового ряду;

9) оцінити точності прогнозу з використанням побудованої моделі часового ряду.

10) здійснити побудову прогнозу на 4 наступні квартали;

11) скористатися листом прогнозу MS Excel для побудови прогнозу на наступний рік.

КОНТРОЛЬНІ ПИТАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ № 10

1. Задача прогнозування в Data Mining.
2. Часові ряди: основні поняття та характеристики.

3. Види часових рядів залежно від характеру рівнів ряду та динаміки змін основних характеристик.
4. Автоковаріація та автокореляція рівнів часового ряду.
5. Структура часового ряду та основні методи її виявлення.
6. У чому полягає аналіз часових рядів? Декомпозиція часового ряду.
7. Види моделей часового ряду, основні підходи до побудови моделей.
8. Оцінка точності та адекватності побудованої моделі часового ряду.
9. Основні етапи прогнозування процесів, представлених одномірними часовими рядами.
10. Виявлення аномальних відхилень, критерій Ірвіна.
11. Перевірка наявності тренду методом перевірки різниць середніх рівнів та з використання критерію серій.
12. Автокореляційний аналіз часового ряду.
13. Моделювання часового ряду на основі механічного згладжування.
14. Авторегресійний аналіз часового ряду.
15. Яким чином здійснюється розрахунок значень сезонної компоненти методом ковзного середнього?
16. Основні етапи побудови адитивної/мультиплікативної моделей часового ряду.
17. Прогнозування в MS Excel з використанням побудови лінії тренду, листа прогнозу.

Таблиця 10.7

Щоквартальні значення змінної *y* за 4 роки

Варіант	1	2	3	4	5	6	7	8	9	10	11	12	13
Період, t	y_t	y_t	y_t	y_t	y_t	y_t	y_t	y_t	y_t	y_t	y_t	y_t	y_t
1	6,1	5,2	5,2	4,9	5,4	5,2	4,2	4,9	4,2	3,1	5,9	4,4	4,8
2	2,8	3,6	7,9	6,3	2,6	5,8	1,6	1,9	2,0	1,4	5,4	2,1	3,0
3	8,4	9,2	8,7	8,6	8,8	8,7	8,2	8,0	8,4	8,6	8,4	8,4	9,5
4	1,9	1,5	0,1	0,1	3,1	0,1	3,1	0,9	0,1	0,4	0,1	0,9	2,4
5	8,1	9,9	8,5	7,9	7,9	8,2	8,5	7,9	8,4	8,9	7,4	8,7	10,9
6	5,6	8,8	10,1	9,1	5,2	9,4	6,1	5,3	6,3	7,4	8,9	7,3	8,6
7	11	14,9	10,8	11,3	10,4	11,8	12,7	10,6	12,4	13,3	11,0	14,0	15,4
8	4,7	7,8	2,9	2,3	5,5	3,0	9,2	2,7	4,0	5,5	2,2	6,2	7,4
9	10,8	15,8	10,0	10,1	9,6	10,8	12,3	9,9	12,0	13,5	9,6	14,6	15,8
10	8,1	14,1	12,0	11,7	7,7	11,7	10,5	8,2	10,1	12,4	11,3	12,1	14,4
11	14,2	20,4	13,8	12,8	13,4	14	16,5	13,7	16,1	18,4	12,2	18,7	20,5
12	9,9	13,1	5,0	5,3	7,6	5,5	11,8	8,4	8,3	11,3	4,4	11,4	13,3
13	12,7	22,1	13,1	12,6	12,4	13	17,3	12,7	16,5	18,9	11,6	19,6	22,1
14	10,4	20,0	14,2	14,3	10,1	13,6	15	10,9	13,9	17,8	13,8	18,3	19,9
15	16,6	26,3	16,3	15,9	15,7	15,8	21,3	16	20,1	23,5	14,4	24,4	26,1
16	12,1	18,4	8,2	7,2	9,6	7,8	16,5	10,7	12,2	16,1	6,5	16,1	19,2
Варіант	14	15	16	17	18	19	20	21	22	23	24	25	26
Період, t	y_t	y_t	y_t	y_t	y_t	y_t	y_t	y_t	y_t	y_t	y_t	y_t	y_t
1	6,2	4,0	4,4	4,5	4,6	5,4	4,6	6,1	4,3	4,1	4,8	4,3	5,4
2	4,6	1,4	2,9	8,6	2,5	3,0	2,5	4,6	2,1	2,4	2,6	3,4	4,8
3	10,7	8,6	9,1	8,1	8,8	8,2	8,9	11,7	7,8	7,8	8,8	10,1	8,6
4	3,1	0,4	2,0	0,1	0,1	0,1	1,6	3,8	0,9	0,1	0,1	2,3	0,1
5	11,4	9,0	0,6	7,7	8,3	7,4	9,0	12,3	8,5	7,6	7,8	10,2	7,2
6	10,3	7,1	9,1	9,9	6,1	4,5	7,5	10,7	6,6	6,0	6,2	8,9	7,1
7	16,4	12,9	15,4	10,6	11,4	10,2	13,6	17,1	13,4	12,1	11,4	15,5	10,4
8	8,6	5,7	7,6	2,1	3,1	1,2	5,8	10,0	4,8	3,9	3,4	7,7	2,4
9	17,2	13,7	15,6	10,4	11,2	9,0	14,1	18,2	13,6	11,2	10,9	16,1	9,8
10	15,6	12,4	14,4	12,1	8,8	6,3	11,4	16,9	11,3	10,0	9,1	14,1	9,6
11	22,1	19,0	20,5	13,2	15,0	11,7	17,7	23,3	18,2	15,1	14,7	20,7	13,2
12	14,2	10,4	13,5	5	6,5	3,3	9,9	15,4	9,7	7,2	6,3	12,8	5,1
13	22,7	19,1	22,0	12,7	14,5	10,0	18,2	24,0	18,5	15,5	14,7	21,8	12,3
14	20,8	17,4	19,7	13,4	11,8	7,4	16,6	22,8	15,9	13,1	11,8	19,7	11,9
15	27,8	23,4	26,4	15,9	17,7	13,3	22,0	29,6	22,6	18,6	17,7	26,1	15,9
16	20,0	16,4	18,3	8,2	9,5	4,9	14,0	22,1	15,2	10,7	10,3	18,8	7,0

11. РОБОТА З НЕЙРОННИМИ МЕРЕЖАМИ У СЕРЕДОВИЩІ MATLAB: КЛАСИФІКАЦІЯ ТА ПРОГНОЗУВАННЯ

Лабораторна робота № 11

Мета: закріплення знань про сутність та основні принципи роботи штучних нейронних мереж. Набуття навичок проведення класифікації та прогнозування шляхом створення та навчання нейромережі у середовищі MatLab.

Теоретичні знання: штучні нейронні мережі, штучний нейрон. Типові функції активації. Етапи побудови нейронної мережі для розв'язання задач Data Mining. Архітектура нейронних мереж. Навчання нейронної мережі. Робота з нейронною мережею у середовищі MatLab у командному режимі та графічному режимі GUI. Створення нейронної мережі для прогнозування часового ряду за допомогою майстра MatLab.

11.1. ШТУЧНІ НЕЙРОННІ МЕРЕЖІ

11.1.1. Штучні нейронні мережі: базові поняття

Серед задач Data Mining, які розв'язують за допомогою штучних нейронних мереж, можна виділити класифікацію, кластеризацію та прогнозування. Нейронні мережі широко використовуються для автоматизації процесів розпізнавання образів, медичної, фінансово-економічної та технічної діагностики, прогнозування соціально-економічних показників тощо.

Штучна нейронна мережа*, *ШНМ (англ. *Artificial Neural Networks, ANN*) – здатна до навчання система, яка складається із сукупності з'єднаних між собою штучних нейронів, що імітують діяльність людського мозку, моделюючи способи передачі сигналів, аналогічні способам їх передачі в біологічних нейронах.

Базовим елементом будь-якої нейронної мережі є штучний (формальний) нейрон.

Штучний нейрон (англ. *Artificial Neuron*) – обчислювальний пристрій, який має кілька входів та один вихід і моделює функції біологічного нейрона, пов'язані з формуванням вихідного сигналу на основі обробки вхідних сигналів, що поступають на його входи.

Основні компоненти штучного нейрона (рис. 11.1):

1) ***однонаправлені входи***, які приймають вхідні сигнали (аналог синапсів біологічного нейрона);
2) ***ваги, вагові коефіцієнти*** – значення, пов'язані з кожним входом, які можуть збуджувати або гальмувати сигнал, вказуючи на його важливість у отриманні вихідного сигналу;
3) ***внутрішній стан нейрона*** – є сигналом активації, який визначається як агрегація зважених вхідних сигналів;
4) ***функція активації*** – функція, яка перетворює внутрішній стан нейрона у вихідний сигнал;
5) ***вихід*** – видає сформований вихідний сигнал (аналог аксона біологічного нейрона).

Основною особливістю нейронних мереж є паралельна обробка інформації за рахунок об'єднання великої кількості з'єднаних між собою нейронів. ***Вхідні нейрони*** (англ. *Input Neuron*) мережі як вхідні сигнали отримують на входах значення ознак об'єктів набору даних, який формують для розв'язання задачі Data Mining. Якщо нейронна мережа має ***внутрішні, приховані нейрони*** (англ. *Hidden Neuron*), то на їх входи як вхідні сигнали подаються вихідні сигнали інших нейронів. ***Вихідні нейрони*** (англ. *Output Neuron*) нейронної мережі видають вихідні сигнали, які утворюють результат розв'язання поставленої задачі – вихідні змінні.

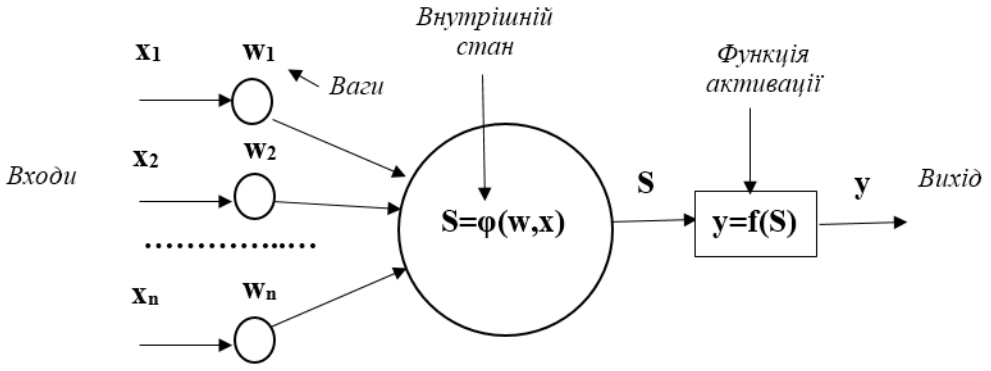


Рис. 11.1. Схема штучного (формального) нейрона

Кожен вхід нейрона x_i має ***вагу*** w_i, яка визначає, наскільки цей вхід впливає на його внутрішній стан.

Внутрішній стан нейрона – сигнал активації, є результатом застосування постсинаптичної функції, яка поєднує зважені вхідні сигнали:

$$S = \varphi(w, x).$$

Постсинаптична (вагова) функція нейрона може бути наступною:

1) *зважена сума* – більш поширена:

$$\varphi(w, x) = \sum_{i=1}^{n} w_i x_i , \tag{11.1}$$

де n – число входів нейрона, x_i – значення, яке подається на і-й вхід нейрона, w_i – вага і-го входу;

2) *зважений добуток*:

$$\varphi(w, x) = \prod_{i=1}^{n} w_i x_i ; \tag{11.2}$$

3) *евклідова відстань*:

$$\varphi(w, x) = \sum_{i=1}^{n} (w_i - x_i)^2 . \tag{11.3}$$

Вихід нейрона визначається як функція його стану, яка перетворює поточний внутрішній стан нейрона у вихідний сигнал:

$$Y = f(S) , \tag{11.4}$$

де f – ***функція активації*** (англ. Activation Function).

Нелінійний перетворювач – елемент формального нейрону, який перетворює поточний стан нейрону у вихідний сигнал по деякому закону, заданому функцією активації.

Вихідний сигнал нейрону подається на вхід точки розгалуження.

Точка розгалуження – це елемент формального нейрону, який відсилає його вихідний сигнал по декільком адресами, має один вхід і декілька виходів.

Від виду активаційної функції залежить набір функціональних можливостей нейронної мережі та методи її навчання. ***Типові функції активації***:

1) *порогова уніполярна* (одиничного стрибка):

$$f(S) = \begin{cases} 1, & S \ge w_0 \\ 0, & S < w_0 \end{cases} ; \tag{11.5}$$

2) *порогова біполярна*:

$$f(S) = \begin{cases} 1, & S \ge w_0 \\ -1, & S < w_0 \end{cases} ; \tag{11.6}$$

3) *лінійна* та її різновиди (уніполярна, біполярна):

$$f(S) = kS; \tag{11.7}$$

4) *сигмоїдальна*: логістична уніполярна:

$$f(S) = \frac{1}{1 + e^{-\alpha S}} . \tag{11.8}$$

Найчастіше в якості функції активації використовується сигмоїдальна функція. Основна перевага цієї функції в тому, що вона диференційована на всій області визначення й має похідну, яка виражається через саму функцію:

$$f'(S) = \alpha F(S)(1 - F(S)) . \tag{11.9}$$

Значення вихідного сигналу у разі застосування сигмоїдальної функції активації лежать у діапазоні [0, 1]. При зменшенні параметра α сигмоїда стає більш пологою, вироджуючись в горизонтальну лінію на рівні 0,5 при $\alpha = 0$. При збільшенні α сигмоїда все більше наближається до функції одиничного стрибка.

Вибір функції активації визначається специфікою поставленої задачі та обмеженнями, які накладаються деякими алгоритмами навчання.

Різні моделі нейронних мереж мають різну архітектуру й можуть відрізняються способами зв'язку нейронів

між собою – топологією, функціями активації, механізмами і напрямками поширення сигналів по мережі, методами навчання.

Послідовність ***етапів побудови нейронної мережі*** для розв'язання поставленої задачі Data Mining є наступною:

1. ***Попередня підготовка вхідного набору даних*** – формування навчаючої множини, визначення кількості інформативних ознак. Усі вхідні і вихідні значення ознак повинні бути представлені у вигляді дійсних чисел із плаваючою точкою або у бінарному вигляді. Тому формування набору даних, який буде використано для навчання нейронної мережі, супроводжується відповідним перекодуванням даних, яке включає нормалізацію числових значень ознак, перетворення категоріальних ознак у числові, Dummy-кодування. Після перекодування набір даних розбивають на дві множини: навчаючу та тестову (або на три: навчаючу, тестову і перевірочну).

При побудові класифікатора існує компроміс між кількістю ознак, які обрано для здійснення аналізу, та кількістю об'єктів навчаючої множини. Якщо ознак мало – мережу навчити неможливо, один і той же набір може відповідати об'єктам із різних класів. А зі збільшенням розмірності простору ознак об'єктів може бути недостатньо для навчання. Важливим є також визначення оптимальної кількості елементів навчаючої множини, оскільки невелика кількість об'єктів може викликати перенавчання мережі.

2. ***Вибір архітектури та топології нейронної мережі*** – кількості нейронів та способу їх зв'язку між собою, функції активації, методу навчання. Нейронні мережі з невеликою кількістю нейронів та лінійною функцією активації не завжди можуть розв'язати поставлену задачу. Однак надлишкова кількість нейронів мережі може призвести до проблеми перенавчання.

3. ***Навчання нейронної мережі*** включає реалізацію обраного методу навчання, підбір вагових коефіцієнтів та параметрів функції активації, при яких мережа оптимально вирішує поставлену задачу. Передбачає розв'язання задачі багатовимірної нелінійної оптимізації з використанням градієнтних або стохастичних методів.

4. ***Оцінка точності та оптимізація нейронної мережі*** з метою зменшення розрахункової складності та зростання швидкості обчислень. За необхідності роблять переналаштування мережі, повернувшись до 1-го та 2-го етапів. Оптимізацію проводять, зменшуючи кількість нейронів та зв'язків між ними так, щоб точність моделі нейронної мережі не була суттєво зменшена.

5. ***Використання навченої нейронної мережі*** для розв'язання поставленої задачі.

11.1.2. Архітектура нейронних мереж

Архітектура штучної нейронної мережі визначається її топологією, способом з'єднання нейронів, наявністю прямих та зворотних зв'язків між ними, функцією активації. У процесі розвитку напряму штучних нейронних мереж виникли складні архітектури мереж, які застосовують для розв'язання широкого кола задач.

Нейронні мережі можуть бути синхронними й асинхронними.

У ***синхронній нейронній мережі*** у кожен момент часу свій стан міняє лише один нейрон.

В ***асинхронній нейронній мережі*** – стан міняється відразу у цілої групи нейронів.

За ***топологією зв'язків*** між нейронами можна виділити дві базові архітектури – повнозв'язні та шаруваті нейронні мережі.

Повнозв'язна нейронна мережа є сукупністю нейронів, кожен з яких передає свій вихідний сигнал іншим нейронам, включаючи самого себе. Вихідними сигналами мережі можуть бути всі або деякі вихідні сигнали нейронів після декількох тактів функціонування мережі. Всі вхідні сигнали подаються всім нейронам.

Шарувата нейронна мережа є сукупністю нейронів, які організовані шарами так, що обробка інформації здійснюється пошарово.

Шар – один або декілька нейронів, на входи яких подається один і той же сигнал.

У рамках одного шару дані обробляються паралельно, а в масштабах всієї мережі обробка ведеться послідовно – від шару до шару. Число нейронів у кожному шарі може бути довільним і не пов'язане з кількістю нейронів у інших шарах.

За ***кількістю шарів*** розрізняють одношарові та багатошарові нейронні мережі.

Одношарова нейронна мережа – це мережа, яка складається з одного шару нейронів. Одношарові нейронні мережі можна застосовувати для розв'язання багатьох задач, таких як прогнозування погоди, аналіз кардіограми, штучний зір. Однак одношарові мережі теоретично не здатні розв'язати багато простих задач, у тому числі реалізувати функцію «виключного АБО».

Багатошарова нейронна мережа – це мережа, яка має кілька шарів. Вхідний шар мережі організований із вхідних нейронів, які одержують дані й поширюють їх на входи нейронів наступного прихованого шару мережі. Вихідні нейрони, з яких організований вихідний шар мережі, видають результат роботи нейронної мережі.

За ***характером зв'язків між нейронами*** розрізняють нейронні мережі прямого поширення та рекурентні нейронні мережі.

Нейронна мережа прямого поширення (англ. *Feedforward Neural Network*) – це мережа, у якій інформація поширюється в одному напрямку від вхідного шару нейронів через приховані шари до вихідного шару, який видає результат опрацювання сигналу (рис. 11.2, рис. 11.3).

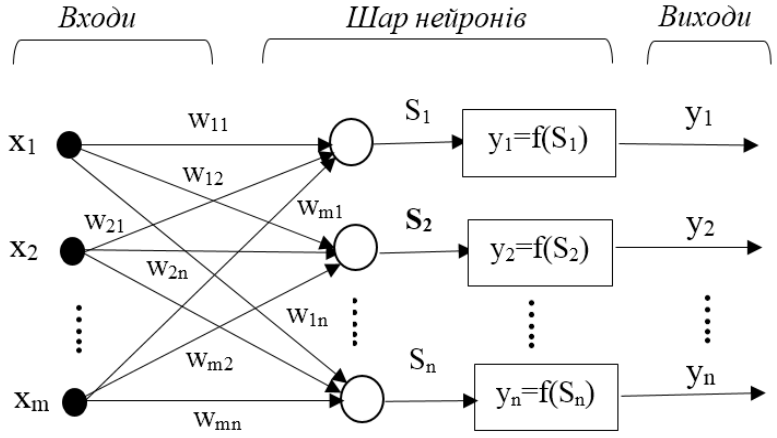


Рис. 11.2. Одношарова нейронна мережа прямого поширення

Нейрони кожного шару у багатошарових мережах такого типу пов'язані з нейронами попереднього та наступного шарів, а всередині кожного шару зв'язки відсутні. Нейрони першого шару отримують вхідні сигнали, перетворюють їх у вихідний сигнал і через точку розгалуження передають його нейронам наступного шару, для яких передані сигнали будуть вхідними. І так до останнього шару, який приймає виходи нейронів передостаннього шару на входи своїх нейронів та видає вихідні сигнали нейронної мережі.

Прикладом нейромереж прямого поширення є одношаровий і багатошаровий перцептрон. ***Перцептрон*** був однією з перших створених нейронних мереж і складався з одного нейрона з пороговою функцією активації, внутрішній стан якого визначався як зважена сума входів. У подальшому з розвитком напряму виникли різні типи нейронних мереж, проте перцептрон є основним структурним елементом більшості моделей нейронних мереж.

Одношаровий перцептрон – це нейронна мережа прямого поширення з одним шаром нейронів, кожен з яких є перцептроном, а багатошаровий перцептрон містить декілька таких шарів.

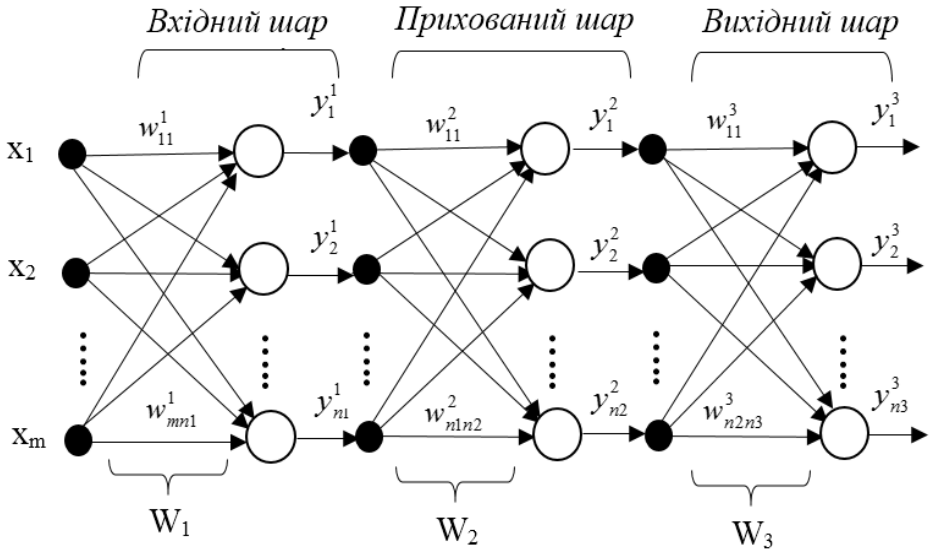


Рис. 11.3. Трьохшарова нейронна мережа прямого поширення

Зв'язки входів із відповідними нейронами у нейронній мережі зручно задавати у вигляді векторів і матриць. В одношаровому перцептроні у матричному вигляді перетворення вхідних даних у вихідні може бути представлене у вигляді рівняння:

$$Y = f(W \cdot X + b), \tag{11.10}$$

де X – вхідний вектор, Y – вихідний вектор,

$f(W \cdot X + b)$ – функція активації, W – матриця вагових коефіцієнтів,

b – вектор зміщень, який дозволяє зміщувати функцію активації шляхом додавання константи до вхідних даних.

Вагові коефіцієнти можуть бути представлені матрицею розмірністю $n \times m$:

$$W = \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1m} \\ w_{21} & w_{22} & \dots & w_{2m} \\ \dots & \dots & \dots & \dots \\ w_{n1} & w_{n2} & \dots & w_{nm} \end{bmatrix}, \tag{11.11}$$

де m – кількість нейронів на вході, n – кількість нейронів на виході.

При цьому якщо $w_{ij} = 0$ – зв'язок відсутній, якщо $w_{ij} < 0$ – зв'язок гальмує сигнал, а якщо $w_{ij} > 0$ – зв'язок посилює (збуджує) сигнал.

У багатошарових нейронних мережах розрізняють матрицю ваг на входах мережі та матриці ваг між шарами (див. рис. 11.3). Опис зв'язків між входами, нейронами та виходами таких мереж задають у вигляді вхідного, вихідного векторів та матриці ваг на входах і послідовної сукупності матриць ваг між кожною парою шарів мережі.

Рекурентною нейронною мережею (англ. *Recurrent Neural Network*) називають нейронну мережу зі зворотними зв'язками, завдяки наявності яких інформація в мережі може поширюватися як у прямому, так і у зворотному напрямках. Рекурентні нейронні мережі можуть бути одношаровими та багатошаровими.

Рекурентні мережі є динамічними і дозволяють зберігати інформацію про попередні стани. Нейрони шару рекурентної нейронної мережі на кожному кроці ітерації отримують інформації не тільки від попереднього шару, а й від самих себе на попередніх кроках. Наявність таких зворотних зв'язків дозволяє обробляти та моделювати послідовності даних – часові ряди, текст, мову, зміну положення об'єкта під час руху.

Згорткова нейронна мережа (англ. *Convolutional Neural Network*) – мережа, яка використовує операцію згортки для вхідних даних при передачі інформації до наступного шару. Згорткові нейронні мережі відносять до ***глибинних нейронних мереж*** (англ. *Deep Neural Network*), які завдяки наявності багатьох шарів обробки даних перетворюють вхідні дані у вихідні, ієрархічно виділяючи та агрегуючи ознаки шляхом підвищення рівня абстракції даних у напрямку від входів до виходів мережі. Завдяки своїй будові такі мережі добре витягують ознаки із зображення і використовуються в задачах класифікації, розпізнавання, сегментації та багатьох інших.

Кожна архітектура нейронної мережі призначена для розв'язання певного класу задач Data Mining: регресії, класифікації, прогнозування, кластеризації й використовує спеціальні методи навчання.

11.1.3. Навчання нейронної мережі

Процес навчання нейронної мережі полягає у налаштуванні її внутрішніх параметрів для розв'язання поставленої задачі.

Виділяють наступні режими навчання штучних нейронних мереж:

1. ***Навчання із вчителем*** (англ. *Supervised Learning*): нейронна мережа отримує на вході навчаючу множину, для якої вхідні та вихідні сигнали є відомими, й налаштовує свої параметри з метою мінімізації помилки відповідно між вхідними та необхідними вихідними сигналами. На вхід нейронної мережі почергово подають елементи навчаючої множини, розраховують похибки між отриманими і фактичними значеннями виходів та корегують параметри мережі у сторону зменшення похибок. Такий підхід застосовують для розв'язання задач класифікації та прогнозування.

2. ***Навчання без учителя*** (англ. *Unsupervised Learning*): нейронна мережа отримує на вході навчаючу множину, для якої відомими є вхідні сигнали. Вихідні сигнали на входи мережі не подаються, а формуються самостійно у процесі її навчання. Такий підхід є доцільним при розв'язанні задачі кластеризації, коли необхідно виявити внутрішню структуру набору даних.

3. ***Навчання з підкріпленням*** (англ. *Reinforcement Leaning*): є змішаним варіантом попередніх підходів, базується на визначенні евристичної похибки.

Навчання мережі є складним та трудомістким процесом, алгоритми навчання мають різні параметри налаштування. Частіше всього навчання нейронної мережі зводиться до корекції вагових коефіцієнтів.

Одночасне налаштування параметрів функції активації та вагових коефіцієнтів призводить до різкого збільшення часу навчання. Тому при розв'язанні поставленої задачі спочатку визначаються з вибором постсинаптичної функції та функції активації, топологією нейронної мережі. Від їх вибору залежить не тільки швидкість, а й метод навчання.

Існуючі методи навчання нейронних мереж можна розділити на дві групи методів: детерміновані та стохастичні. До детермінованих методів навчання відносять методи, які базуються на ітеративній корекції параметрів мережі. Стохастичні методи навчання змінюють параметри мережі випадковим чином, зберігаючи ті з них, які призвели до поліпшення роботи мережі.

Процес навчання штучної нейронної мережі є ітераційним і здійснюється на наборі даних, який розбивають на навчаючу та тестову множини (або навчаючу, тестову та перевірочну).

Навчання нейронної мережі складається із певної кількості епох.

Епоха (цикл) – процес, який включає оновлення внутрішніх параметрів моделі нейронної мережі у результаті застосування алгоритму навчання до всіх об'єктів набору даних та, можливо, оцінку її точності на тестовій множині (чи тестовій і перевірочній множинах).

Кожна епоха складається з ***ітерацій***. Якщо набір даних є великим, під час однієї ітерації на входи мережі подається пакет – ***батч*** (англ. *batch*), частина набору даних. Кількість ітерацій за одну епоху рівна кількості частин – пакетів, на яку поділено набір даних.

Після кожного кроку ітерації на основі аналізу вхідних та вихідних даних ваги мережі переналаштовуються таким чином, щоб мінімізувати різницю між бажаним виходом та отриманим у результаті застосування моделі нейронної мережі. Цю різницю називають ***помилкою навчання*** (похибкою). Помилку навчання для конкретної конфігурації нейронної мережі визначають шляхом прогону через мережу усіх об'єктів навчаючої множини і порівняння отриманих значень виходів із їх фактичними ***цільовими значеннями***. У якості функції, яку необхідно оптимізувати (мінімізувати), частіше всього беруть суму квадратів помилок.

Після багаторазового пред'явлення вхідного набору даних ваги мережі стабілізуються, мережа дає правильні виходи для усіх об'єктів навчаючої множини – ***мережа навчена*** (натренована).

У загальному випадку ***задача навчання*** нейронної мережі зводиться до знаходження функціональної залежності:

$$Y = f(\varphi(W, X)), \tag{11.10}$$

де X – вхідний вектор, Y – вихідний вектор, W – матриця ваг,

$\varphi(W, X)$ – постсинаптична вагова функція, $f(\varphi(W, X))$ – функція активації.

Під час навчання на першому кроці ітерації початкові значення ваг – елементи матриці W^1, обирають випадковим чином і, як правило, установлюють близькими до нуля. Далі на кожному наступному кроці ітерації їх перераховують відповідно до обраного методу навчання: $W^2, W^3, ..., W^k$. Переналаштування ваг продовжується до тих пір, поки помилка навчання – різниця між відповідними значеннями розрахованого $\hat{Y}$ та цільового Y вихідних векторів, не досягне мінімуму.

Навчання нейронних мереж може базуватися на прямому та зворотному поширенні помилки.

Основна ідея ***алгоритму зворотного поширення помилки*** (англ. *Backpropagation Algorithm*) полягає у поширенні помилки після її обчислення на виході мережі у напрямку, зворотному прямому поширенню сигналів. Спочатку розраховують помилки в останньому шарі (на основі вхідних та цільових вихідних значень), далі – у передостанньому і так аж до першого шару. Значення ваг, розраховані на певному кроці ітерації, при зворотному проході налаштовуються з метою мінімізації помилки.

Алгоритм зворотного поширення помилки має різні модифікації. Його застосовують для штучних нейронних мереж із будь-якою кількістю шарів як прямого поширення, так і таких, які мають зворотні зв'язки. Такі мережі є потужним інструментом пошуку закономірностей, прогнозування, якісного аналізу.

Рекурентні нейронні мережі зі зворотними зв'язками використовують алгоритм зворотного поширення помилки з деякими змінами – ***алгоритм зворотного поширення похибки через час*** (англ. *Backpropagation Through Time, BPTT*). Це пов'язано з тим, що параметри мережі на поточному кроці ітерації під час її навчання залежать від розрахованих параметрів на попередніх кроках. Рекурентні нейронні мережі працюють із часовим рядом і

добре справляються з задачами, пов'язаними із класифікацією послідовностей: розпізнавання мови, подавлення шуму, класифікація тексту.

Останнім часом все більшої популярності набуває ***глибинне навчання*** (англ. *Deep Learning*) нейронних мереж, яке здійснює побудову високорівневих ієрархічних абстракцій ознак із сирих необроблених даних. Ознаки вищих рівнів виводяться з ознак нижчих рівнів, формуючи ієрархічне їх представлення. Для реалізації глибинного навчання використовують рекурентні та глибинні нейронні мережі, зокрема, згорткові. Великою перевагою глибинного навчання є заміна ручної роботи аналітика з виділення ознак автоматичними алгоритмами їх ієрархічного виділення у процесі навчання мережі.

11.1.4. Методи навчання нейронних мереж

Методи навчання нейронних мереж застосовують різні підходи для корегування ваг зв'язку між нейронами. Розглянемо найбільш поширені з них.

1. ***Правило Хеба***: вага w_{ij} зв'язку між нейронами i та j змінюється пропорційно добутку вхідного та вихідного сигналів:

$$\Delta w_{ij} = \eta \cdot x_i y_j, \tag{11.11}$$

де x_i – вихід i-го та вхід j-го нейронів, y_j – вихід j-го нейрона, η – коефіцієнт навчання, що впливає на його швидкість.

Відповідно до цього правила вага, яка на кроці ітерації навчання з номером t була рівна $w_{ij}(t)$, на наступному $(t+1)$-му кроці ітерації буде розрахована за формулою:

$$w_{ij}(t+1) = w_{ij}(t) + \eta \cdot x_i(t) y_j(t), \tag{11.12}$$

де $x_i(t)$ – вихід i-го та вхід j-го нейронів на кроці ітерації t, $y_j(t)$ – вихід j-го нейрона на кроці ітерації навчання з номером t.

Правило Хеба було сформульоване для перцептрона, розраховується для кожного зв'язку нейронів і базується на властивості біологічних нейронів, яка полягає у тому, що зв'язок між нейронами посилюється, якщо вони обидва є активними. Правило має багато модифікацій і може застосовуватися для різних типів нейронних мереж із різними функціями активації.

2. ***Правило Хопфілда***: є подібним до правила Хеба, однак за ним визначається величина посилення або послаблення зв'язку: вага збільшується на Δw_{ij}, якщо вхідний та вихідний сигнали нейрону є одночасно активними або неактивними. Інакше вага зменшується на Δw_{ij}.

3. ***Дельта-правило*** є одним із найбільш часто використовуваних: під час навчання ваги збільшуються або зменшуються таким чином, щоб була зменшена різниця між розрахованим та фактичним значенням виходу нейрона.

Відповідно до дельта-правила вагу w_{ij} зв'язку між нейронами i та j змінюють так, щоб зменшити розбіжність між цільовим та розрахованим виходом j-го нейрона на величину:

$$\Delta w_{ij} = \eta \cdot x_i (y_j - \hat{y}_j), \tag{11.13}$$

де x_i – вихід i-го та вхід j-го нейронів, y_j і $\hat{y}_j$ – цільове та розраховане значення виходу j-го нейрона, η – коефіцієнт швидкості навчання.

Тоді формула коригування ваги w_{ij} зв'язку між нейронами i та j є наступною:

$$w_{ij}(t+1) = w_{ij}(t) + \eta \cdot x_i (y_j - \hat{y}_j), \tag{11.14}$$

де $w_{ij}(t)$ та $w_{ij}(t+1)$ – значення ваг на поточному та наступному кроках ітерації навчання.

Дельта-правило є модифікацією правила Хеба. У випадку багатошарової нейронної мережі відповідно до цього методу на кожному кроці ітерації для кожного об'єкта навчаючої множини обчислюють значення виходу, порівнюють його з відповідним цьому об'єкту цільовим значенням виходу, розраховують помилку й корегують

ваги, мінімізуючи її. Обчислення помилки та корегування ваг відбувається пошарово у зворотному напрямі від вихідного шару мережі до вхідного. Вага зв'язку між j - м та кожним i - м нейроном змінюється в сторону зменшення помилки пропорціонально величині сумарної помилки j -го нейрона з використанням алгоритму зворотного поширення помилки.

4. ***Градієнтні методи навчання***: засновані на оптимізації цільової функції й передбачають диференційованість функції активації. Більшість таких методів зводяться до використання методу найменших квадратів. При навчанні ставиться задача мінімізації цільової функції E – помилки нейронної мережі, яку знаходять за методом найменших квадратів:

$$E = \sum_{j=1}^{n} (y_j - \hat{y}_j)^2 , \tag{11.15}$$

де $y_j = \{y_{j1}, y_{j2}, ..., y_{jn}\}$ – цільове значення j -го виходу,

$\hat{y}_j = \{\hat{y}_{j1}, \hat{y}_{j2}, ..., \hat{y}_{jn}\}$ – розраховане значення j -го виходу,

n – кількість нейронів у вихідному шарі.

Навчання нейронної мережі здійснюється методом градієнтного спуску, при якому зміна ваги w_{ij} зв'язку між нейронами i та j розраховується за формулою:

$$\Delta w_{ij} = -\eta \cdot \frac{\partial E}{\partial w_{ij}}, \tag{11.16}$$

де $\frac{\partial E}{\partial w_{ij}}$ – градієнт, η – коефіцієнт швидкості навчання, який частіше усього задають як константу $0 < \eta < 1$.

Градієнтом називають вектор часткових похідних цільової функції похибок по вагам нейронної мережі. Однією з його властивостей є те, що він показує напрям оптимізації цільової функції. Формула 11.16 виражає основну ідею ***методу градієнтного спуску***: для мінімізації помилки вагу необхідно змінювати у напрямку, протилежному градієнту. По мірі наближення до мінімуму цільової функції величина градієнта буде зменшуватися.

Для нейронної мережі градієнт визначається за формулою:

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial y_j} \frac{\partial y_j}{\partial S_j} \frac{\partial S_j}{\partial w_{ij}}, \tag{11.17}$$

де E – цільова функція помилок, визначена за методом найменших квадратів (формула 11.15),

w_{ij} – вага зв'язку між нейронами i та j, y_j – вихід j -го нейрона, S_j – внутрішній стан j -го нейрона.

Розглянемо окремо складові формули 11.17:

1) $\frac{\partial S_j}{\partial w_{ij}} = x_i$ – вихід i -го та вхід j -го нейронів;

2) $\frac{\partial y_j}{\partial S_j}$ – значення похідної активаційної функції по її аргументу S_j для нейрона j ;

3) $\frac{\partial E}{\partial y_j}$ – похибка j -го нейрона.

Значення похибки $\frac{\partial E}{\partial y_j}$ визначені явно тільки для нейронів вихідного шару. Для визначення похибки нейронів прихованих шарів застосовують метод зворотного поширення помилки, який дає рекурсивні формули для розрахунку похибок:

1) для вихідного шару:

$$\delta_j = \frac{\partial E}{\partial y_j}, \tag{11.18}$$

2) для прихованого шару з номером k :

$$\delta_j = \frac{\partial y_j}{\partial S_j} \cdot \sum_{i=1}^{n_{k+1}} \delta_i^{k+1} w_{ij}^{k+1}, \tag{11.19}$$

де δ_{ij}^{k+1}, w_{ij}^{k+1}, n_{k+1} – похибки, ваги та число нейронів шару шару $k+1$, який іде після шару з номером k.

Таким чином, зміна ваги зв'язку між нейронами i та j розраховується за формулою:

$$\Delta w_{ij} = -\eta \cdot \delta_j \cdot x_i, \tag{11.20}$$

де x_i – вихід i-го та вхід j-го нейронів, δ_j – розраховують за формулою 11.18 для вихідного шару й формулою 11.19 для прихованих шарів, η – коефіцієнт швидкості навчання.

Тоді формула для корегування ваги зв'язку між нейронами i та j буде мати вигляд:

$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij}(t), \tag{11.21}$$

де $w_{ij}(t)$ та $w_{ij}(t+1)$ – значення ваг на поточному та наступному кроках ітерації навчання.

11.1.5. Перцептрон: його властивості та обмеження

Перцептрон – нейронна мережа, яка складається зі штучного нейрону з пороговою функцією активації, що видає 1, якщо $S > w_0$ та 0, якщо $S < w_0$, де w_0 – задане значення ***порогу активації***, а S – зважена сума входів (рис. 11.4).

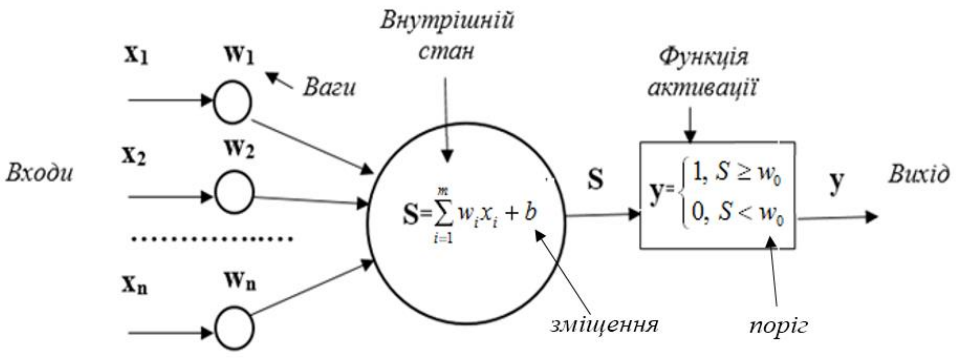


Рис. 11.4. Загальна схема перцептрона

Перцептрон можна розглядати як приклад простої одношарової нейронної мережі. Математична модель перцептрона описується співвідношеннями:

$$y = f(S) = \begin{cases} 1, & S \geq w_0 \\ 0, & S < w_0 \end{cases}, \tag{11.22}$$

$$S = \sum_{i=1}^{m} w_i x_i + b, \tag{11.23}$$

де w_i – вага i-го входу, x_i – i-й вхідний сигнал,

w_0 – поріг активації, S – внутрішній стан перцептрона,

y – вихід, множина значень якого називається ***цільовим вектором,***

b – ***зміщення***, що є константою, яка інколи вводиться для зсуву сигналу.

Перцептрон дозволяє розв'язувати задачі класифікації, пов'язані з множинами лінійно роздільних класів. Однак є певні класи об'єктів, які перцептрон розрізнити не може. Перцептрон з одним нейроном може здійснювати класифікацію множини об'єктів набору даних на два лінійно роздільних класи. Якщо стоїть задача побудови

класифікатора для множин об'єктів, представлених кількістю класів, більшою, ніж два, кількість нейронів у перцептроні повинна бути більшою.

Для одношарового перцептрона ***основні етапи навчання*** є наступними.

1. Усім вагам w_i з набору ваг $w=\{w_1,...,w_k\}$, зміщенню b та порогу w_0 присвоюють випадкові значення (k – кількість ознак у об'єктів навчаючої множини).

2. На входи нейронної мережі по черзі подаються значення ознак об'єктів навчаючої множини $x_j=\{x_{1j},...,x_{kj}\}$, де x_{ij} – i-та ознака j-го об'єкта ($i\in\{1,2,...,k\}$), $j\in\{1,2,...,m\}$, m – кількість об'єктів навчаючої множини) та відповідні їм вихідні цільові значення y_j.

3. На виході перцептрона для кожного чергового *j*-го об'єкта навчаючої множини x_j обчислюють значення виходу $\hat{y}_j = f(\varphi(w, x_j)$ і порівнюють його з відповідним j-му об'єкту цільовим значенням виходу y_j.

4. Якщо різниця між y_j та $\hat{y}_j$ є суттєвою, ваги перцептрона коригують відповідно до обраного методу навчання й повертаються до кроку 2.

5. Якщо різниця м між y_j та $\hat{y}_j$ незначна – процедуру налаштування ваг завершують. Мережа є навченою й готова для подальшого використання з метою розв'язання поставленої задачі.

Приклад 1. Маємо множину об'єктів, кожен з яких характеризується двома ознаками. Для кожного об'єкта відоме значення цільової змінної – класу, до якого він відноситься (табл. 11.1). Необхідно побудувати штучну нейронну мережу з одного перцептрона, яка здійснює класифікацію набору даних на два класи з використанням дельта-правила, та здійснити класифікацію нового об'єкта за допомогою створеної мережі.

1. Маємо навчаючу множину об'єктів x_j із двома ознаками, які належать двом класам. Значення цільової змінної, яка відповідає виходу нейрона, для навчаючої множини може приймати два значення: 0 (*клас 1*) та 1 (*клас 2*).

Тому для розв'язання задачі класифікації доцільно застосувати перцептрон, який має два входи та один вихід.

Таблиця 11.1

Набір об'єктів навчаючої множини даних

Об'єкти, x_j	Ознаки об'єктів		Клас, y_j
	x_{j1}	x_{j2}	
x_1	2	2	0
x_2	1	-2	1
x_3	-2	2	0
x_4	-1	1	1

2. Усім вагам w^{li}, зміщенню b_i^l та порогу w_0 присвоюємо значення, рівні нулю (l – номер епохи, i – номер об'єкта навчаючої множини: $i\in\{1,2,34\}$). Відповідно до початкових налаштувань маємо:

$$w^{11}=[w_1^{11}w_2^{11}]=[00],\ b_1^1=0.$$

Функція активації є пороговою ($w_0=0$), і приймає значення відповідно до формули 11.22:

$$\hat{y}=f(S)=\begin{cases}1, & S\geq 0\\ 0, & S<0\end{cases}.$$

3. Здійснюємо перший прогін об'єктів навчаючої множини через перцептрон – номер епохи рівний одиниці:

$l = 1$. Епоха буде складатися з ітерацій, які включають подання на входи перцептрона одного з об'єктів навчаючої множини. Кожна епоха буде містити чотири ітерації.

4. Розраховуємо внутрішній стан перцептрона за формулою 11.23 та його вихід для першого об'єкта:

$$S_1^1 = w_1^{11} \cdot x_{11} + w_2^{11} \cdot x_{12} + b_1^1 = 0 \cdot 2 + 0 \cdot 2 + 0 = 0,$$

$$S_1^1 = 0 \geq 0, \ \hat{y}_1^1 = f(S_1^1) = f(0) = 1.$$

5. Вихід $\hat{y}_1^1 = 1$ не співпадає з цільовим виходом $y_1 = 0$. Розрахуємо помилку – різницю між цільовим і розрахованим виходом, та зміну ваг і зміщення:

помилка: $e_1^1 = y_1 - \hat{y}_1^1 = 0 - 1 = -1$;

зміна ваг: відповідно до дельта-правила визначаємо її за формулою 11.13, надавши коефіцієнту швидкості навчання η значення, рівне 1:

$$\Delta w_1^{11} = e_1^1 \cdot x_{11} = -1 \cdot 2 = -2, \ \Delta w_2^{11} = e_1^1 \cdot x_{12} = -1 \cdot 2 = -2;$$

зміна зміщення: $\Delta b_1^1 = e_1^1 = -1$.

6. Розраховуємо нові ваги за формулою 11.14:

$$w_1^{12} = w_1^{11} + \Delta w_1^{11} = 0 - 2 = -2,$$

$$w_2^{12} = w_2^{11} + \Delta w_2^{11} = 0 - 2 = -2.$$

Маємо: $w^{12} = [-2 - 2]$.

7. Розраховуємо нове зміщення: $b_2^1 = b_1^1 + \Delta b_1^1 = 0 - 1 = -1$.

8. Розраховуємо внутрішній стан перцептрона за формулою 11.23 та його вихід для другого об'єкта:

$$S_2^1 = w_1^{12} \cdot x_{21} + w_2^{12} \cdot x_{22} + b_2^1 = (-2) \cdot 1 + (-2) \cdot (-2) - 1 = 1,$$

$$S_2^1 = 1 \geq 0, \ \hat{y}_2^1 = f(S_2^1) = f(1) = 1.$$

9. Вихід $\hat{y}_2^1 = 1$ співпадає з цільовим виходом $y_2 = 1$. Тому помилка e_2^1 буде рівна нулю і зміни ваг і зміщення не буде:

$$w_1^{13} = w_1^{12} = -2, \ w_2^{13} = w_2^{12} = -2, \ b_3^1 = b_2^1 = -1.$$

10. Розраховуємо внутрішній стан перцептрона за формулою 11.23 та його вихід для третього об'єкта:

$$S_3^1 = w_1^{13} \cdot x_{31} + w_2^{13} \cdot x_{32} + b_3^1 = (-2) \cdot (-2) + (-2) \cdot 2 - 1 = -1,$$

$$S_3^1 = -1 < 0, \ \hat{y}_3^1 = f(S_3^1) = f(-1) = 0.$$

11. Вихід $\hat{y}_3 = 0$ співпадає з цільовим виходом $y_3 = 0$. Тому помилка e_3^1 буде рівна нулю і зміни ваг і зміщення не буде:

$$w_1^{14} = w_1^{13} = -2, \ w_2^{14} = w_2^{13} = -2, \ b_4^1 = b_3^1 = -1.$$

12. Розраховуємо внутрішній стан перцептрона за формулою 11.23 та його вихід для четвертого об'єкта:

$$S_4^1 = w_1^{14} \cdot x_{41} + w_2^{14} \cdot x_{42} + b_4^1 = (-2) \cdot (-1) + (-2) \cdot 1 - 1 = -1,$$

$$S_4^1 = -1 < 0, \ \hat{y}_4^1 = f(S_4^1) = f(-1) = 0.$$

13. Вихід $\hat{y}_4^1 = 0$ не співпадає з цільовим виходом $y_4 = 1$. Розрахуємо помилку – різницю між розрахованим та цільовим виходом, та зміну ваг і зміщення:

помилка: $e_4^1 = y_4 - \hat{y}_4^1 = 1 - 0 = 1$;

зміна ваг: $\Delta w_1^4 = e_1^4 \cdot x_{41} = 1 \cdot (-1) = -1, \ \Delta w_2^4 = e_1^4 \cdot x_{42} = 1 \cdot 1 = 1$;

зміна зміщення: $\Delta b_1^4 = e_1^4 = 1$.

14. Здійснюємо другий прогін об'єктів навчаючої множини через перцептрон – номер епохи рівний двом: $l = 2$. Розраховуємо нові ваги :

$$w_1^{21} = w_1^4 + \Delta w_1^4 = -2 - 1 = -3,$$
$$w_2^{21} = w_2^4 + \Delta w_2^4 = -2 + 1 = -1.$$

Маємо: $w_1^{21} = [-3 - 1]$.

15. Розраховуємо нове зміщення:

$$b_2^1 = b_1^4 + \Delta b_1^4 = 1 - 1 = 0.$$

16. Розраховуємо внутрішній стан перцептрона за формулою 11.23 та його вихід для першого об'єкта:

$$S_1^2 = w_1^{21} \cdot x_{11} + w_2^{21} \cdot x_{12} + b_2^1 = -3 \cdot 2 + (-1) \cdot 2 + 0 = -8,$$
$$S_1^2 = -8 < 0, \ \hat{y}_1^2 = f(S_1^2) = f(-8) = 0.$$

17. Вихід $\hat{y}_1^2 = 0$ співпадає з цільовим виходом $y_1 = 0$. Тому помилка e_1^2 буде рівна нулю і зміни ваг і зміщення не буде:

$$w_1^{22} = w_1^{21} = -3, \ w_2^{22} = w_2^{21} = -1, \ b_2^2 = b_1^2 = 0.$$

18. Далі розрахунки проводяться аналогічно наведеним вище. У таблиці 11.2 наведено значення розрахованих параметрів перцептрона на наступних кроках навчання. Рішення буде задовільним, якщо розраховані ваги та зміщення перестали змінюватися, а прогін усіх об'єктів навчаючої множини через перцептрон при цих значеннях дає виходи, які співпадають із цільовими.

Таблиця 11.2

Налаштування перцептрона на кроках ітерацій навчання

Епоха, L	Ітерації	Об'єкт, x_j	w_1	w_2	b	S	Клас		e
							$\hat{y}_j$	y_j	
2	5	x_1	-3	-1	0	-8	0	0	0
	6	x_2	-3	-1	1	-1	0	1	1
	7	x_3	-2	-3	1	-1	0	0	0
	8	x_4	-2	-3	1	0	1	1	0
3	9	x_1	-2	-3	1	-9	0	0	0
	10	x_2	-2	-3	1	5	1	1	0
	11	x_3	-2	-3	1	-1	0	0	0
	12	x_4	-2	-3	1	0	1	1	0
4	13	x_1	-2	-3	1	-9	0	0	0
	14	x_2	-2	-3	1	5	1	1	0
	15	x_3	-2	-3	1	-1	0	0	0
	16	x_4	-2	-3	1	0	1	1	0

19. Під час другої епохи, при $l = 2$, на п'ятій ітерації розраховані та цільові значення виходів не співпадають, помилка не є нульовою. А при третьому прогоні об'єктів навчаючої множини через перцептрон – під час третьої

епохи, при $l=3$, усі помилки є нульовими, а ваги та зміщення перестали змінюватися. Такі ж самі значення спостерігаємо і під час четвертої епохи, при $l=4$. Робимо висновок – мережа є навченою.

20. Фінальні значення ваг входів перцептрона будуть рівні $w_1=-2$ та $w_2=-3$: $w=[w_1\ w_2]=[-2\ -3]$, зміщення $b=1$.

21. За допомогою побудованого класифікатора здійснимо класифікацію нового об'єкта, який не входив у навчаючу множину, зі значеннями ознак $x_1=1$ і $x_2=1{,}5$. Розраховуємо внутрішній стан перцептрона та його вихід для нового об'єкта:

$$S^{new}=w_1\cdot x_1^{new}+w_2\cdot x_2^{new}+b=-2\cdot 1+(-3)\cdot 1{,}5+1=-5{,}5,$$

$$S^{new}=-5{,}5<0,\ \ \hat{y}^{new}=f(S^{new})=f(-5{,}5)=0.$$

Розрахований вихід перцептрона для нового об'єкта рівний 0, отже цей об'єкт можна віднести до *класу 1*.

Зауваження. У даному прикладі для наочності демонстрації налаштувань параметрів перцептрона на кроках ітерацій навчання було взято набір даних із невеликою кількістю об'єктів. Для побудови класифікатора доцільно використовувати навчаючу множину, яка має значно більше об'єктів. Це супроводжується збільшенням розрахункової складності. Тому такі задачі розв'язують із використанням бібліотек та програмних засобів із вбудованими алгоритмами побудови та навчання нейронних мереж.

11.2. ЗДІЙСНЕННЯ КЛАСИФІКАЦІЇ ЗА ДОПОМОГОЮ НЕЙРОННИХ МЕРЕЖ У СЕРЕДОВИЩІ MATLAB

11.2.1. Класифікація одношаровим перцептроном на два класи

Приклад 2. Побудувати штучну нейронну мережу з одного перцептрона, яка здійснює класифікацію набору даних у двовимірному просторі ознак на два класи, та здійснити класифікацію нового об'єкта за допомогою створеної мережі.

1. Створюємо скрипт у MATLAB, який задає навчаючу множину – вхідний набір даних для класифікації, об'єкти якого представлені двома ознаками, та цільовий вектор зі значеннями класів кожного об'єкта навчаючої множини:

```
% задаємо матрицю X – вхідний набір даних (навчаючу множину)
X=[-0.5 -0.5 -0.6 0.3 0.1; -0.5 0.5 0.0 0.5 1.0];
% задаємо вектор T – цільові класи для об'єктів навчаючої множини
T=[1 1 1 0 0];
```

Цільовий вектор може приймати значення тільки два значення: $T=\{0,1\}$, де 0 та 1 відповідають належності об'єкта навчаючої множини до одного із двох класів. Для кожного об'єкта навчаючої множини ми знаємо відповідне йому значення цільового вектора.

2. Створимо нову нейронну мережу – перцептрон із одним нейроном, у якій є два входи x_1 та x_2, що відповідають ознакам об'єктів навчаючої множини, та один вихід y – клас об'єкта. Для цього доповнимо скрипт кодом, у якому здійснюється налаштування мережі на розв'язок поставленої задачі:

```
plotpv(X,T); % візуалізація навчаючої множини
net=perceptron; % створення нейронної мережі – перцептрона
net=configure(net,X,T); % налаштування мережі для роботи з даними
XX = repmat(con2seq(X),1,3); % створення серії для вхідних даних
TT = repmat(con2seq(T),1,3); % створення серії для вихідних даних
% налаштування нейронної мережі шляхом пред'явлення серій
net = adapt(net,XX,TT);
plotpc(net.IW{1},net.b{1});% візуалізація розподільної лінії перцептрона
```

Пояснення до коду:

perceptron() – функція, яка створює перцептрон;

configure(net,X,T) – функція, яка приймає вхідні дані X та цільові дані T й налаштовує входи та виходи мережі відповідно до них (початкові ваги входів та зміщення налаштовані в 0);

repmat() – команда, яка формує серії XX та TT у вигляді послідовності – масиву комірок, де кожен стовпець вказує часовий крок і копіюється тричі;

adapt() – команда, яка оновлює мережу для кожного часового кроку й повертає нову мережу, яка є кращим класифікатором від попередньої.

3. Результатом виконання даного скрипта буде візуалізація об'єктів навчаючої множини та лінії, яка розділяє об'єкти на два класи (рис. 11.5).

Ми бачимо, що у результаті трьох тактів навчання один перцептрон у одному шарі навчився надійно класифікувати класи, які є лінійно роздільними у двовимірному просторі ознак, які подаються на входи мережі.

4. У вікні Command Window введемо команди для виведення ваг та зміщення створеної мережі (рис. 11.6). Маємо: ваги входів будуть рівні $w_1 = -1{,}6$ та $w_2 = -1$, зміщення $b = 0$.

Це значить, що для даної мережі зважена сума знаходиться за формулою: $S = -1{,}6x_1 - x_2$, а рівняння прямої, яка розділяє два класи має вигляд: $-1{,}6x_1 - x_2 = 0$.

5. Введемо у вікні Command Window команду для візуалізації схеми створеної нейронної мережі:

```
>> view(net)
```

У окремому вікні буде виведена схема перцептрона, яка містить один нейрон, має два входи й один вихід (рис. 11.7).

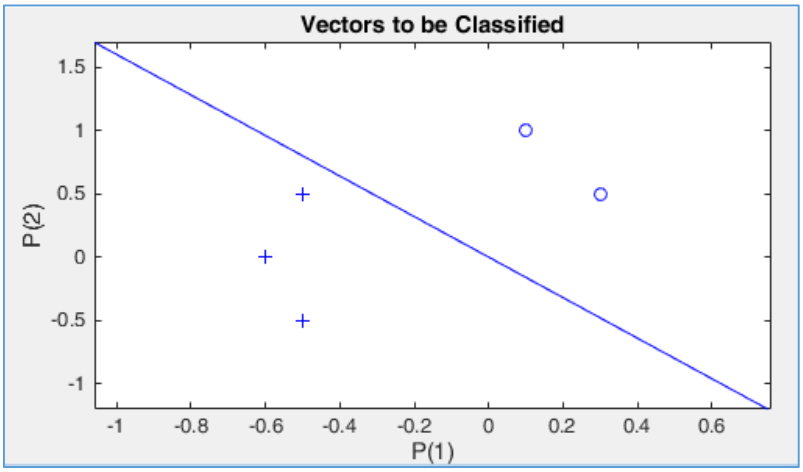


Рис. 11.5. Візуалізація результату роботи класифікатора

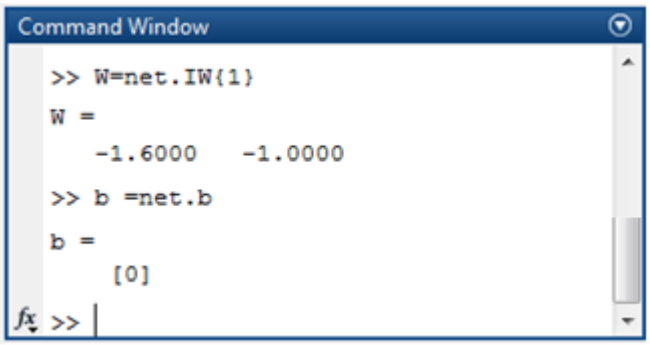


Рис. 11.6. Налаштовані параметри нейронної мережі – перцептрона

6. Збережемо створений класифікатор – навчену мережу net, та навчаючу і цільову множини, ввівши у вікні Command Window команду:

```
>> save DataNet.mat net X T
```

7. За допомогою створеної нейронної мережі здійснимо класифікацію нового об'єкта, який не входив у навчаючу множину, зі значеннями $x_1 = 0{,}5$ і $x_2 = 1{,}3$. Для цього створимо скрипт із наступним кодом:

```
% завантажуємо створену мережу – класифікатор, навчаючу множину й
% цільові класи
load DataNet.mat;
x = [0.5; 1.3]; % задаємо новий об’єкт x
y = net(x); % y - результат класифікації об’єкту x
hold on;
plotpv(x,y); % візуалізація результату роботи мережі
point = findobj(gca,'type','line'); % виділення нової точки
point.Color = 'red'; % зафарбовування нової точки
hold on;  plotpv(X,T);% виведення об’єктів навчаючої множини
% візуалізація розподільної лінії перцептрона
plotpc(net.IW{1},net.b{1});
```

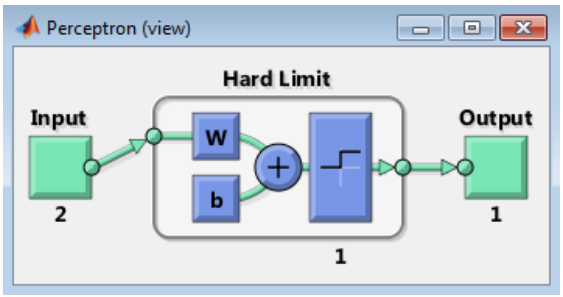


Рис. 11.7. Схема нейронної мережі – перцептрона

8. Результатом виконання даного скрипта буде візуалізація об’єктів навчаючої множини, лінії, яка розподіляє об’єкти на два класи, та класифікація нового об’єкта, якому на графіку відповідає точка, виділена червоним кольором (рис. 11.8). Ми бачимо, що класифікатор правильно її класифікував, віднісши до нульового класу, який позначено кружечком.

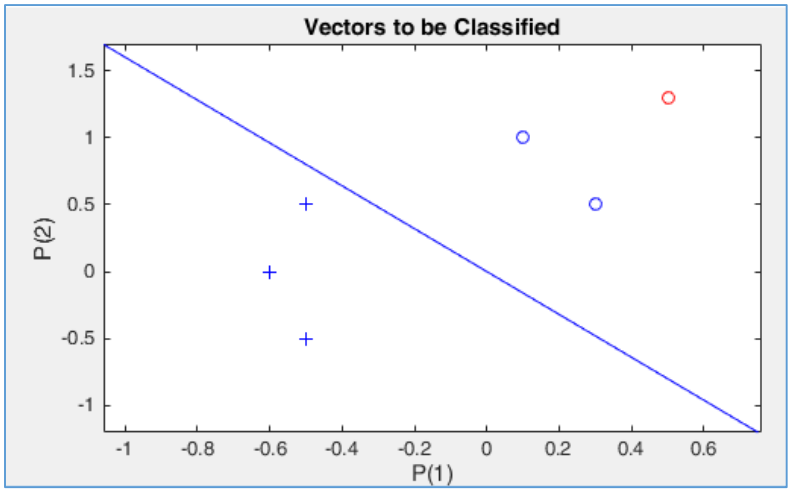


Рис. 11.8. Візуалізація класифікації нового об’єкта створеним класифікатором

Приклад 3. Побудувати нейронну мережу, яка здійснює класифікацію об’єктів у двовимірному просторі ознак на 4 класи. Навчаючу множину лінійно роздільних класів згенерувати засобами MatLab.

11.2.2. Класифікації об’єктів на 4 класи. Підготовка даних до навчання

1. Створюємо множину значень двох змінних об’єктів першого класу з центром у точці (2;0), розкидом від центру *rand*(1,50), кількістю елементів 50: масиву розмірністю 1x50 шляхом введення у вікні Command Window команд:

```
>> x1=2+rand(1,50);  y1=0+rand(1,50);    % 1-й клас
```

2. Створюємо множину значень змінних об'єктів наступних трьох класів із розкидом від центру *rand*(1,50) та кількістю елементів по 50 кожен, із центрами в точках (-2;0), (0;2), (0;-2) шляхом введення у вікні Command Window команд:

```
>> x2=-2+rand(1,50);   y2=0+rand(1,50);   % 2-й клас
>> x3=0+rand(1,50);    y3=2+rand(1,50);   % 3-й клас
>> x4=0+rand(1,50);     y4=-2+rand(1,50);   % 4-й клас
```

3. Формуємо навчаючу множину для нейронної мережі, що буде подаватися на її входи й міститиме об'єкти усіх чотирьох класів. Для цього сполучаємо всі вхідні дані в одну матрицю **z**, яка буде містити координати об'єктів навчаючої множини:

```
>> x(1:50)=x1; x(51:100)=x2; x(101:150)=x3; x(151:200)=x4;
>> y(1:50)=y1; y(51:100)=y2; y(101:150)=y3; y(151:200)=y4;
>> z(1,1:200)=x; z(2,1:200)=y;
```

4. Цільовий вектор може приймати чотири значення, які відповідають чотирьом класам: 1, 2, 3 і 4. Задаємо значення цільового вектора для кожного об'єкта кожного класу навчаючої множини:

```
>> T1(1:50)=1;  T2(1:50)=2;  T3(1:50)=3;  T4(1:50)=4;
```

5. З'єднуємо цільові вектори класів, формуючи масив результатів класифікації:

```
>> T(1:50)=T1;  T(51:100)=T2;  T(101:150)=T3;   T(151:200)=T4;
```

6. Для візуалізації об'єктів усіх 4-х класів навчаючої множини (рис. 11.9) введемо команди:

```
>> figure (1)
>> hold on  % збереження графіку для додавання нових елементів
>> z = z'; scatter(z(:,1),z(:,2),50, T, 'filled')
>> grid on
>>% наносимо мітки класів
>>text(2.2,1.2,'1 Class'); text(-1.8,1.2,'2 Class');
>>text(0.2,1.8,'3 Class'); text(0.2,-0.8,'4 Class');
```

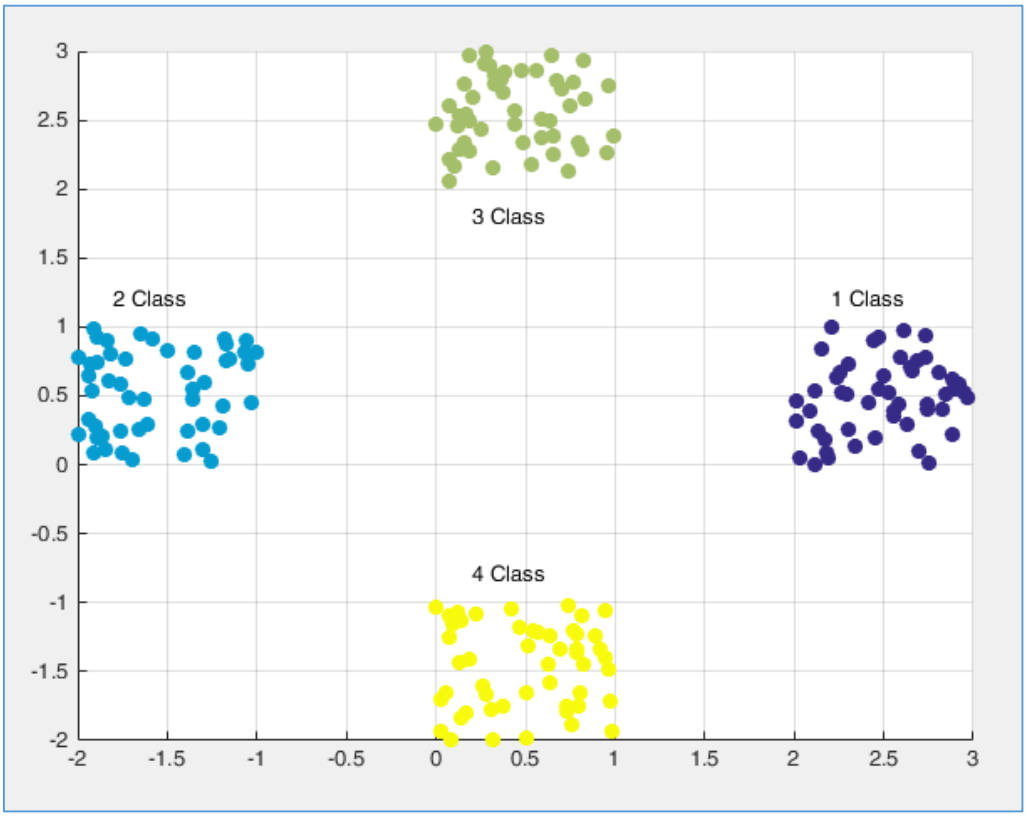


Рис. 11.9. Візуалізація об'єктів 4-х класів навчаючої множини

11.2.3. Створення та навчання тришарової нейронної мережі

1. Створюємо структуру нейронної мережі – тришарову мережу з 10 нейронами у першому шарі, 2 нейронами у другому шарі та 1 нейроном у третьому шарі з виведенням інформації про неї у вікні Command Window:

```
>> % створюємо трьохшарову нейронну мережу
>> net = feedforwardnet([10,2,1], 'trainlm')
```

Пояснення до коду:

feedforwardnet() – функція, яка створює багатошарову нейронну мережу;

[10,2,1] – аргумент функції, що вказує кількість шарів та кількість нейронів у кожному шарі;

trainlm – метод навчання мережі: Левенберга–Марквардта.

Метод Левенберга–Марквардта установлений по замовчуванню для функції feedforwardnet(), тому він буде застосований і у випадку, якщо другий параметр функції не буде вказано. Навчання за алгоритмом Левенберга–Марквардта проводиться методом градієнтного спуску зі зворотним поширенням помилки у комбінації з методом найменших квадратів.

2. Здійснюємо налаштування створеної нейронної мережі:

```
>> % проводимо налаштування мережі
>> z=z';  net = configure(net,z,T); % z – навчаюча множина,   T – цільовий вектор
>> net.layers{1}.transferFcn='logsig' % функція активації 1-го шару
>> net.layers{3}.transferFcn='purelin' % функція активації 3-го шару
```

Пояснення до коду:

configure() – функція, яка здійснює налаштування параметрів створеної мережі відповідно до вхідних та цільових даних;

net.layers{1}.transferFcn – параметр створеної мережі ***net***, що задає функцію активації шару, по замовчуванню значення параметру рівне 'tansig' (такою буде функція активації 2-го шару – вона не задана).

Для шарів створюваної нейронної мережі було установлено такі функції активації:

logsig() – сигмоїдна (логістична) функція;
purelin() – лінійна функція;
tansig() – гіперболічний тангенс.

3. Здійснюємо навчання нейронної мережі:

```
>> % проводимо навчання мережі
>> net = train(net,z,T); % train() – функція, яка здійснює навчання
```

4. Після введення команди з'явиться вікно *Neural Network Training*, у якому буде відображатися процес навчання мережі та його характеристики (рис. 11.10). У верхній частині вікна *Neural Network* розміщена схема нейронної мережі, де зображено кількість шарів, кількість нейронів у кожному шарі, кількість ознак об'єктів навчаючої множини, які подаються на входи нейронів першого шару мережі.

5. У області ***Algorithms*** вікна *Neural Network Training* вказано спосіб розділення об'єктів вхідного набору даних та навчаючу, тестову та перевірочну множини – випадковий, метод навчання мережі – Левенберга–Марквардта, тип функції помилки – MSE (середньоквадратична помилка).

6. У області ***Progress*** вікна *Neural Network Training* візуально відображено характеристики процесу навчання мережі та граничні значення, досягнення яких зупиняє процес навчання:

a) кількість епох – 163 (граничне значення – 1000);
b) час навчання – 1 с; точність – $3{,}61*10^{-11}$ (початкове значення – 0);
c) градієнт – $9{,}66*10^{-8}$ (максимальне значення – 1,84, мінімальне – $1*10^{-7}$);
d) параметр точності помилки Mu, який застосовують для додавання модуляції до ваги для уникнення попадання у локальний мінімум – $1*10^{-11}$, (мінімальне значення – 0);
e) Volidation Checks: кількість епох, протягом яких значення помилки зростало, – 0 (граничне значення 6).

Зупинка процесу навчання відбувається, коли одне із граничних значень буде досягнуто. Наприклад, коли значення функціоналу помилки зростає 6 епох підряд – такі обмеження установлені по замовчуванню для методу навчання Левенберга–Марквардта. Їх можна змінити, задавши перед навчанням мережі для параметра *net.trainParam.max_fail* інше значення. У нас процес навчання зупинився, коли градієнт досяг значення $9{,}66*10^{-8}$ (діаграма градієнта забарвлена зеленим кольором).

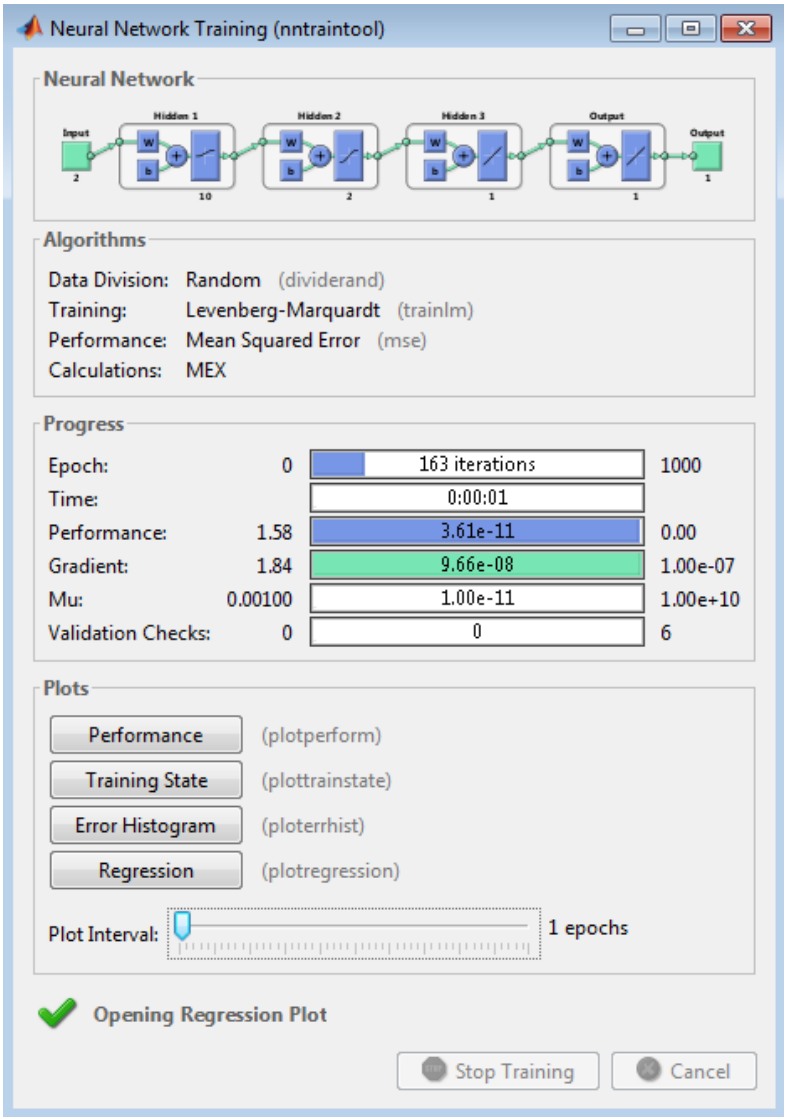


Рис. 11.10. Вікно Neural Network Training

7. При натисненні на кнопку ***Performance*** у лівій частині області *Plots* вікна *Neural Network Training*, у окремому вікні здійснюється виведення графіка функції помилки по епохам навчання для навчаючої, тестової та перевірочної множин (рис. 11.11). Середньоквадратична помилка MSE має порядок 10^{-11}, що свідчить про високу точність роботи класифікатора.

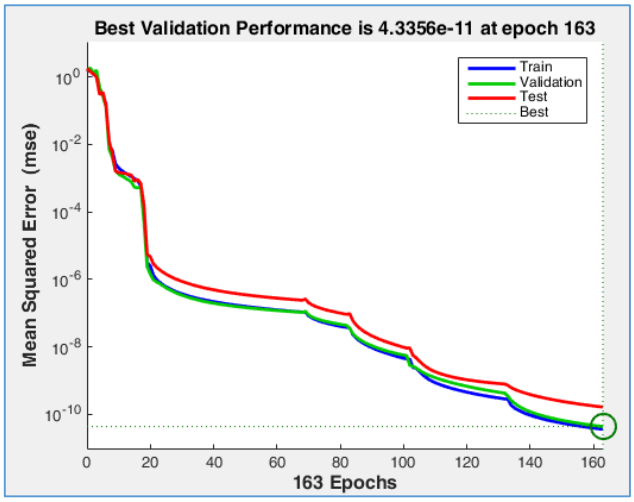


Рис. 11.11. Вікно Neural Network Training Performance

8. При натисненні на кнопку ***Training State*** у лівій частині області *Plots* вікна *Neural Network Training* у окремому вікні здійснюється графічне виведення характеристик стану навчання (рис. 11.12): кількість епох, досягнена величина градієнта, значення Mu, кількість епох, протягом яких значення помилки зростало.

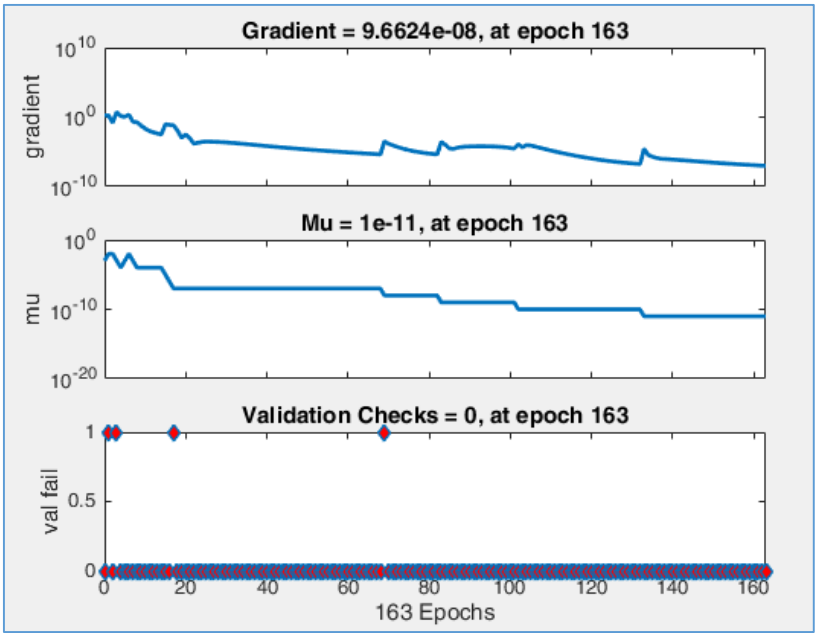


Рис. 11.12. Вікно Neural Network Training State

9. При натисненні на кнопку ***Error Histogram*** у лівій частині області *Plots* вікна *Neural Network Training*, у окремому вікні здійснюється виведення гістограми помилок, у якій різними кольорами виділені частки помилок для навчаючої (синій колір), тестової (червоний колір) та перевірочної (зелений колір) множин (рис. 11.13).

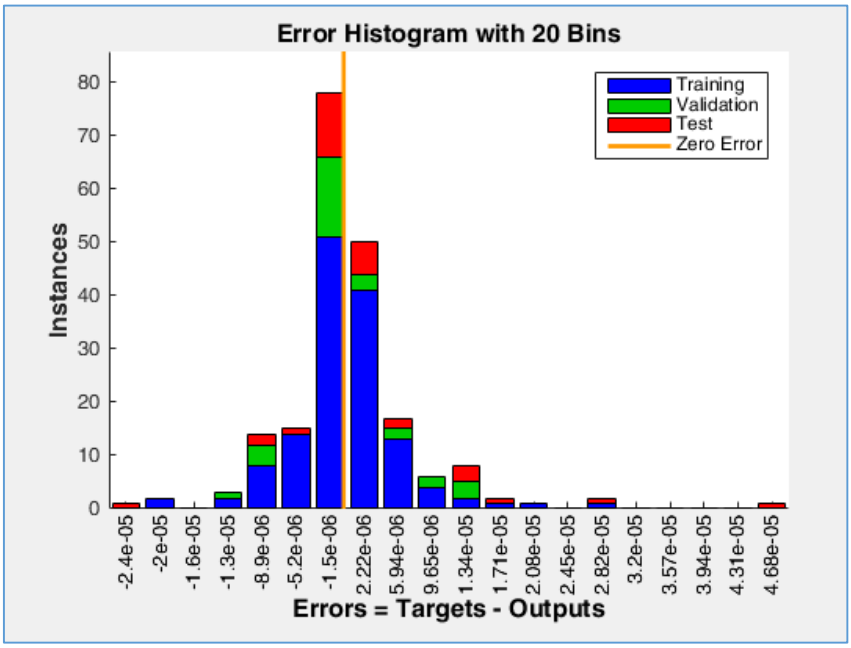


Рис. 11.13. Вікно Neural Network Error Histogram

10. При натисненні на кнопку ***Regression*** у лівій частині області *Plots* вікна *Neural Network Training*, у окремому вікні здійснюється виведення функції регресії розрахованих вихідних значень від фактичних цільових

(рис. 11.14). Побудова такої функції є одним із інструментів оцінки результату навчання. Аналізуючи отриманий результат, можемо стверджувати, що точність класифікації є високою, оскільки коефіцієнт детермінації R = 1.

11.2.4. Перевірка результатів навчання мережі

1. Здійснюємо перевірку мережі на множинах, на яких здійснювалося навчання мережі – функція ***sim()*** моделює нейронну мережу:

```
>> a1=sim(net,z(:,1:50))      % перевірка 1-го класу
>> a2=sim(net,z(:,51:100))    % перевірка 2-го класу
>> a3=sim(net,z(:,101:150))   % перевірка 3-го класу
>> a4=sim(net,z(:,151:200))   % перевірка 4-го класу
```

Після введення кожної з наведених вище команд буде виводитися належність кожного елемента з вказаної множини до певного класу. Для множини a1 усі значення рівні 1, для множини a2 – рівні 2, для множини a3 – рівні 3, для множини a4 – рівні 4. Це відповідає цільовим значенням виходів об’єктів цих множин та свідчить про те, що мережа є навченою добре.

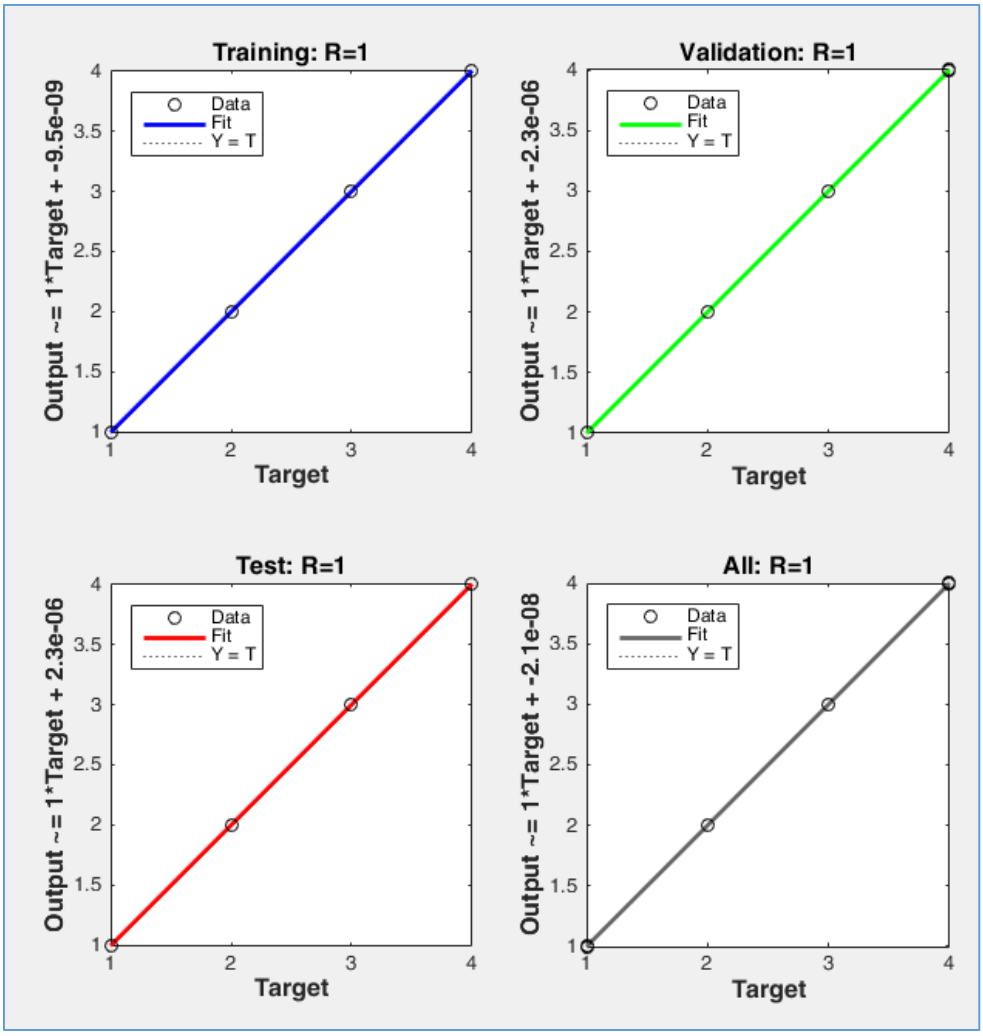


Рис. 11.14. Neural Network Training Regression

11.2.5. Здійснення класифікації нових об’єктів

1. Здійснюємо формування масиву перевірочних нових об’єктів, які потрібно класифікувати у навченій нейронній мережі:

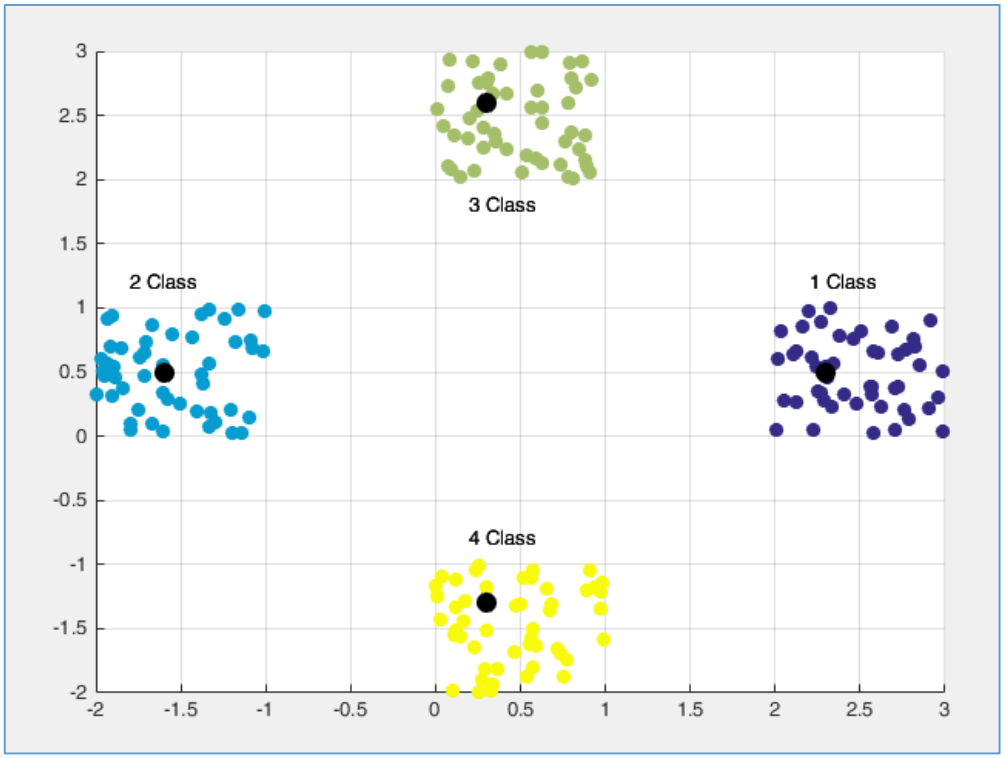
```
>> % формуємо масив F так, щоб він мав об'єкти 4-х класів
>> F = [0.3 0.3 2.3 -1.6; 2.6 -1.3 0.5 0.5];
```

2. Здійснюємо класифікацію нових даних, які було сформовано, з допомогою створеної мережі, яка пройшла навчання:

```
>> a5=sim(net,F);
```

3. Для графічного відображення на одному екрані об’єктів навчаючої множини та нових класифікованих об’єктів введемо команди (рис. 11.15):

```
>> F=F'; % візуалізація результату роботи класифікатора
>> figure(1);  hold on;
>> scatter(F(:,1), F(:,2),100, 'ko', 'filled')
>>text(2.2,1.2,'1 Class'); text(-1.8,1.2,'2 Class');
>>text(0.2,1.8,'3 Class'); text(0.2,-0.8,'4 Class');
```



Чорними точками на малюнку відображено перевірочні об’єкти, кожен із яких знаходиться у множині певного класу

Рис. 11.15. Результат класифікації нових об’єктів

4. Виводимо значення змінної *a5* на екран – масив буде містити результати класифікації перевірочних об’єктів, здійснені навченою нейронною мережею (рис. 11.16):

```
>> a5
a5 =
    3.0000    4.0000    1.0000    2.0000
>>
```

Рис. 11.16. Результат класифікації нових об’єктів

5. Аналіз візуального відображення результату класифікації та значень змінної *a5* показав, що усі чотири об’єкти було класифіковано правильно.

11.3. РОБОТА З НЕЙРОННОЮ МЕРЕЖЕЮ У РЕЖИМІ ГРАФІЧНОГО ІНТЕРФЕЙСУ

Графічний інтерфейс користувача NNTool MatLab дозволяє вибирати структури нейронних мереж із великого переліку і надає велику кількість алгоритмів навчання для кожного типу мережі.

Приклад 4. Побудувати нейронну мережу, яка здійснює класифікацію об'єктів у двовимірному просторі ознак на 2 класи, з використанням GUI-інтерфейсу MatLab.

11.3.1. Створення нейронної мережі

1. Для відкриття основного вікна графічного інтерфейсу користувача необхідно ввести команду:

>> nntool

2. У результаті виконання команди буде відкрите вікно створення нейронної мережі *Neural Network / Data Manager (nntool)* (рис. 11.17).

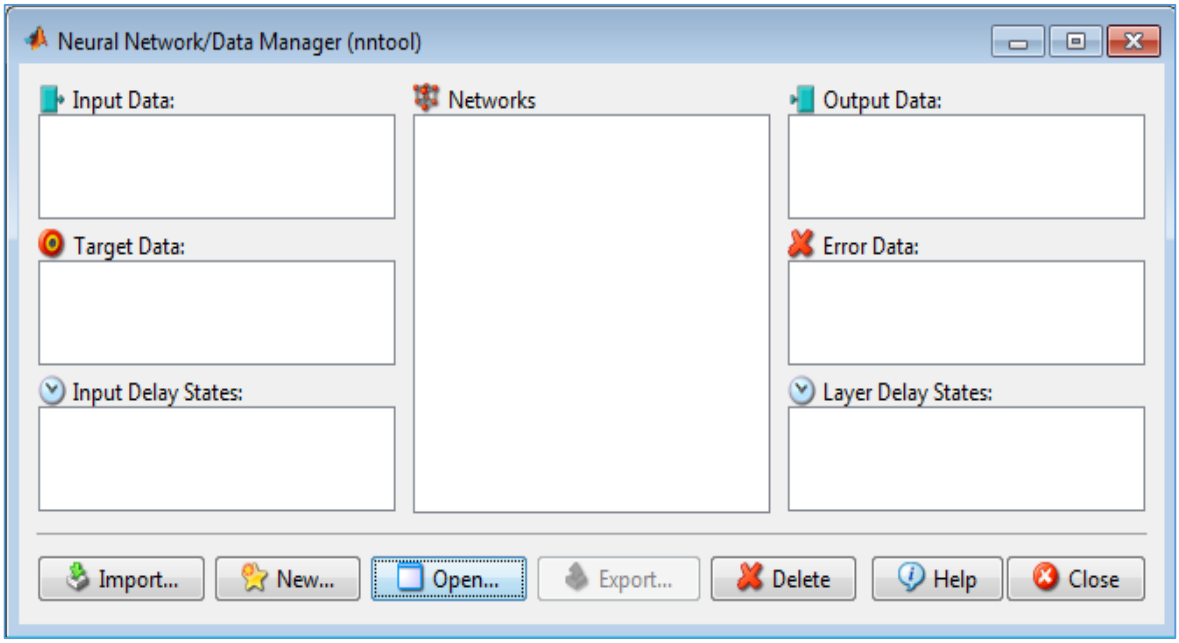


Рис. 11.17. Вікно GUI-інтерфейсу

3. Для формування навчаючої множини і вектора цілей (результатів класифікації) у робочій області GUI-інтерфейсу натиснемо кнопку *New...* у нижній частині вікна та перейдемо до вікна *Create Network or Data*. Для формування даних у вікні *Create Network or Data* необхідно відкрити вкладку *Data*.

4. У полі *Name* вкладки *Data* введемо ім'я змінної *P*, у полі *Value* – вектор значень:

[-1 0 -1 1.5 1 2; -0.5 -1 1 1.5 1.5 1].

Після цього необхідно установити прапорець *Inputs* для вказівки типу даних – вхідні, та натиснути кнопку *Create*. З'явиться вікно повідомленням про створення нової вхідної змінної (рис. 11.18). У вікні повідомлення необхідно натиснути кнопку *OK*.

5. Далі переходимо до створення вектора цілей T: у полі *Name* вкладки *Data* введемо ім'я змінної *T*, у полі *Value* – вектор значень: [0 0 0 1 1 1]. Після цього необхідно установити прапорець *Targets* для вказівки типу даних – Targets/цільові, та натиснути кнопку *Create*.

З'явиться вікно повідомленням про створення нової цільової змінної (рис. 11.19). У вікні повідомлення необхідно натиснути кнопку *OK*.

6. Для створення нейронної мережу переходимо на вкладку *Network* вікна *Create Network or Data* (рис. 11.20). У полі *Name* введемо ім'я створюваної мережі – *networkMy*.

Зі списку Network Type виберемо тип створюваної мережі з прямою передачею сигналу і зі зворотним поширенням помилки: *feed-forward backprop*.

7. У списку *Input data* вкажемо вектор вхідних даних – *P*.

8. У списку *Target data* вкажемо вектор цілей – *T*.

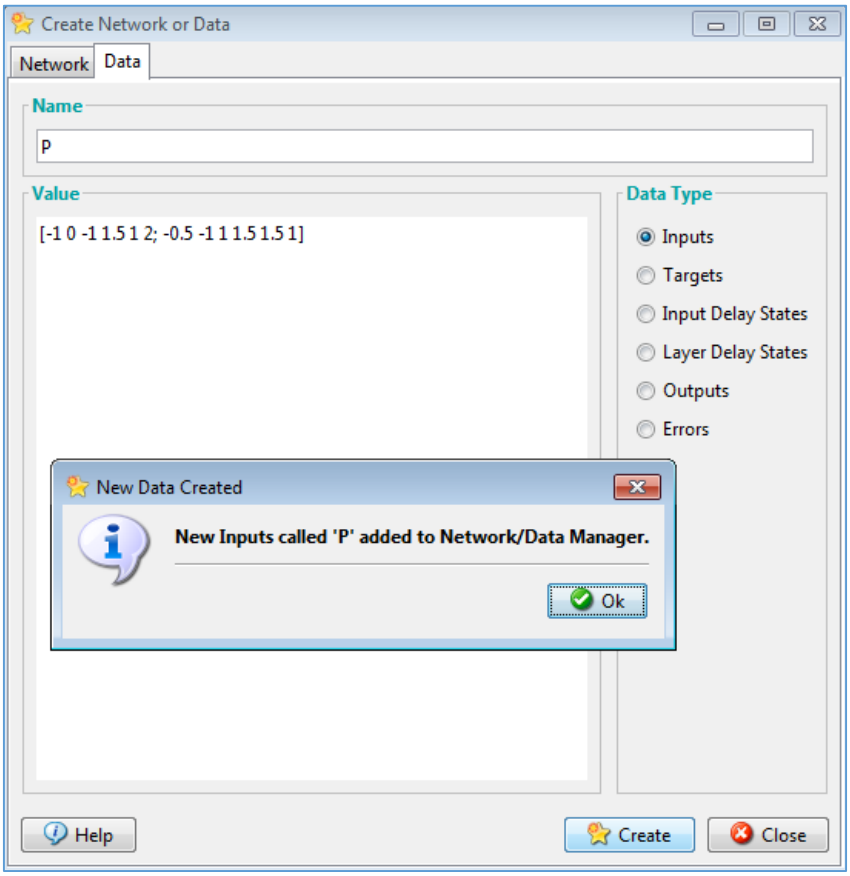


Рис. 11.18. Створення вхідного вектора даних у вікні *Create Network or Data*

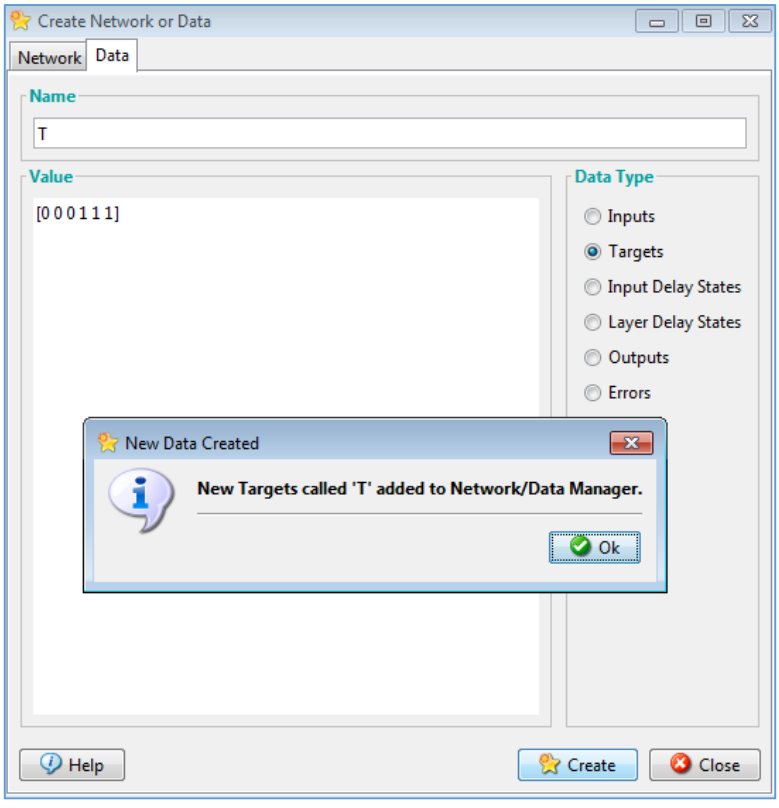


Рис. 11.19. Створення вектора цілей у вікні *Create Network or Data*

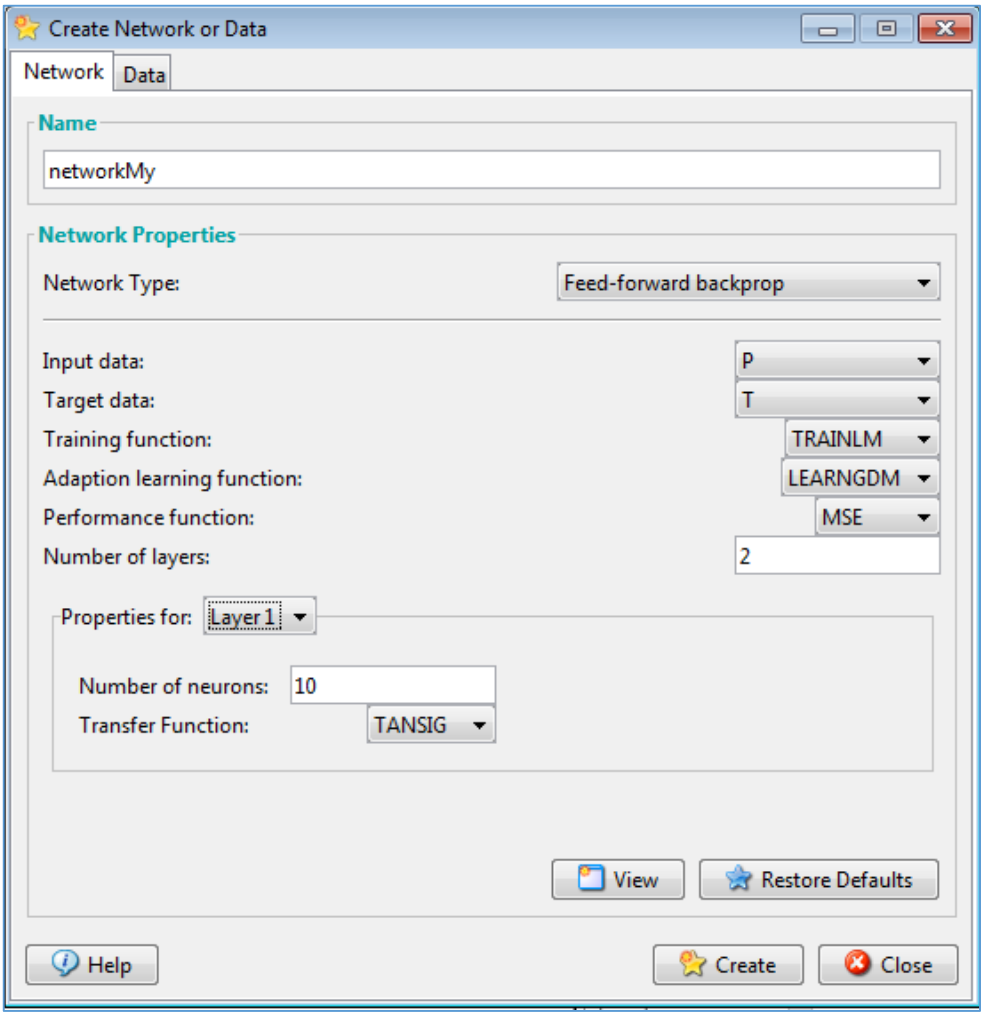


Рис. 11.20. Створення нейронної мережі у вікні *Create Network or Data*

8. Значення перерахованих нижче параметрів залишаємо установленими за замовчуванням:
   a) Training function (функція TRAINLM);
   b) Adaption learning function (функція LEARNGDM);
   c) Performance function (функція MSE);
   d) Number of layers (кількість шарів 2).

9. У списку *Properties for* обираємо перший шар мережі *Layers 1*, у полі *Number of neurons* вказуємо кількість нейронів першого шару: 10. У списку *Transfer Function* значення залишимо за замовчуванням: *TANSIG* (рис. 11.20).

10. Створення мережі завершується натисканням кнопки *Create*. У вікні повідомлення про створення мережі необхідно натиснути кнопку *OK* для його закриття та натиснути кнопку *Close*.

11. У вікні *Neural Network / Data Manager (nntool)* в області *Networks* з'явиться ім'я щойно створеної мережі – *networkMy*. Після виділення імені мережі стають доступні усі кнопки вікна *Neural Network / Data Manager (nntool)* (рис. 11.21).

12. Для виконання ініціалізації мережі необхідно натиснути кнопку *Open…* у нижній частині вікна *Neural Network/Data Manager*. Це призведе до відкриття діалогової панелі *Network: networkMy*. У першій вкладці цього вікна *View*, яка буде відкрита, буде відображена схема нейронної мережі, яка була створена (рис. 11.22).

13. Для введення встановлених діапазонів та ініціалізації ваг необхідно відкрити вкладку *Reinitialize Weights*, в полі *Input Ranges* відображено встановлений діапазон, у разі необхідності можна задати область вихідних значень (*Get from input* – P) і скористатися кнопками *Set Input Ranges* (Встановити діапазони) і *Initialize Weights* (Ініціалізувати ваги) (рис. 11.23). Однак для виконання поставленого завдання це не потрібно.

Якщо потрібно повернутися до колишніх діапазонів, то слід вибрати кнопки *Revert Input Ranges* (Повернути діапазони) і *Revert Weights* (Повернути ваги).

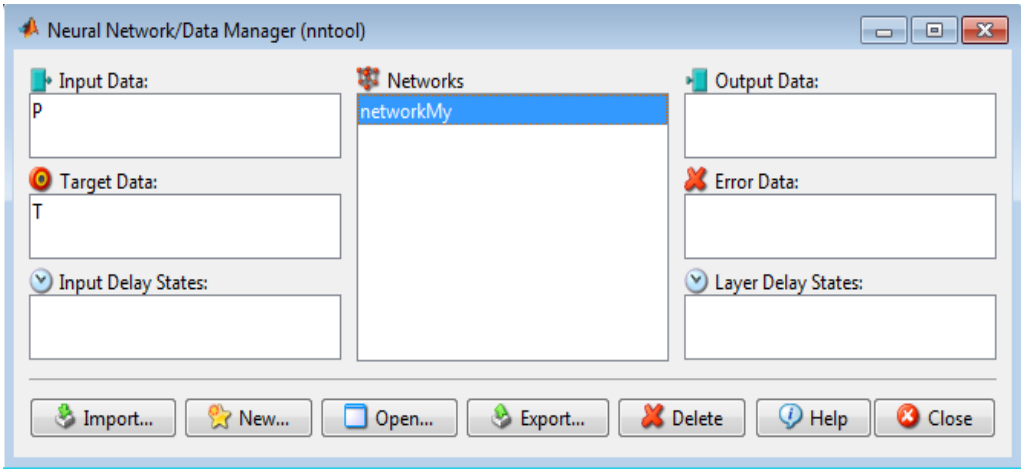


Рис. 11.21. Вікно GUI-інтерфейсу після створення мережі

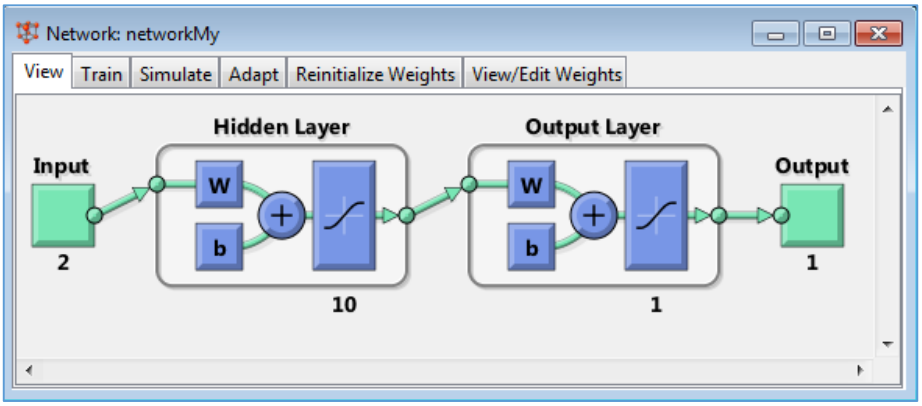


Рис. 11.22. Панель *Network: NetworkMY*

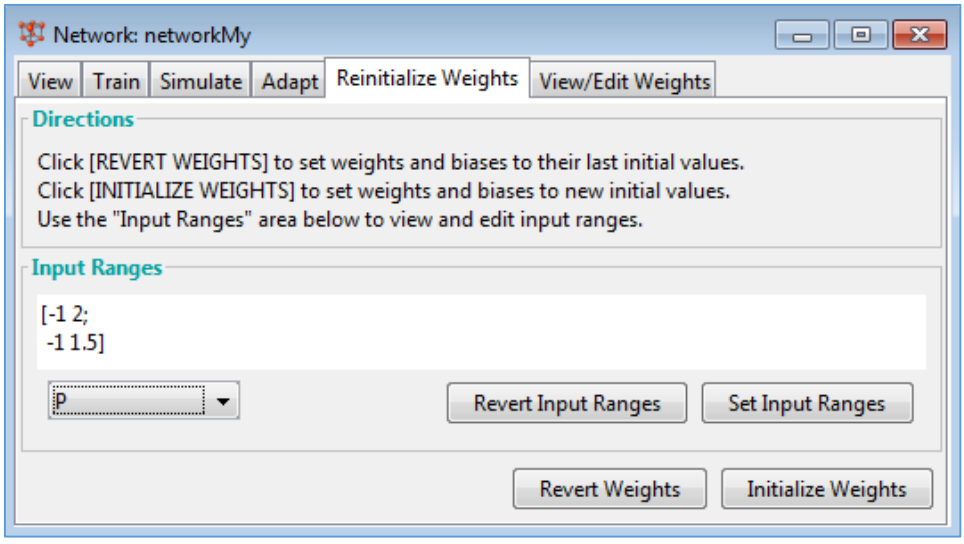


Рис. 11.23. Вкладка налаштування ваг панелі *Network: NetworkMY*

11.3.2. Навчання нейронної мережі

1. Для навчання нейронної мережі необхідно обрати вкладку *Train* панелі *Network: networkMy*. У вікні, що відкриється, на вкладці *Training Info* (*Інформація про навчальні послідовності*), встановлюємо імена вхідних даних – P, та цілей – T (рис. 11.24). Значення у вкладці *Training Parameters* (*Параметри навчання*) залишаємо без змін (рис. 11.25).

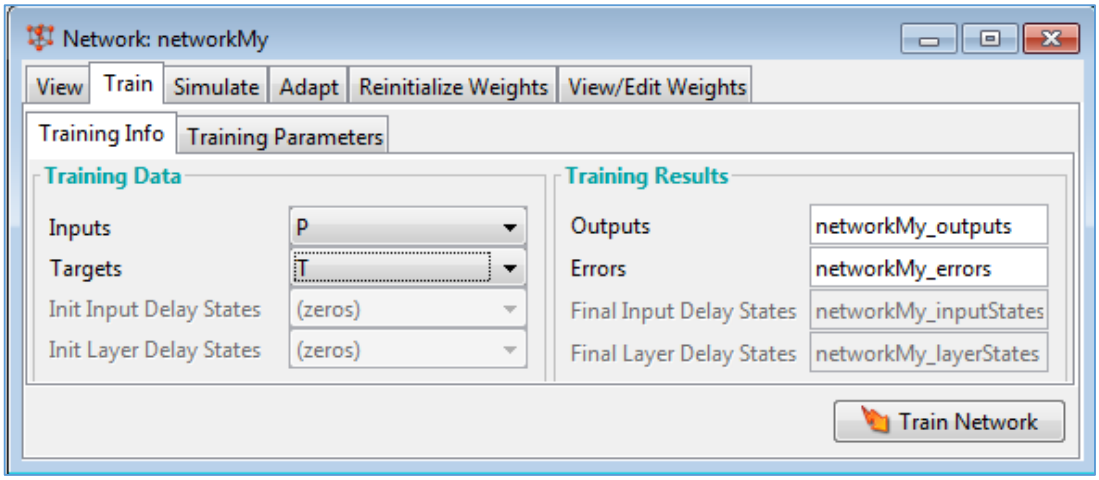


Рис. 11.24. Вкладка Training Info вікна налаштування навчання

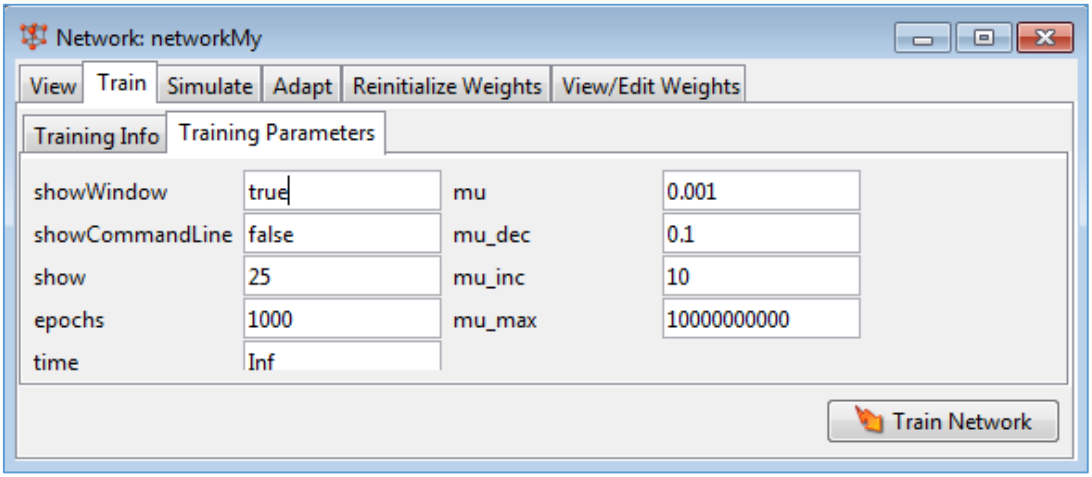


Рис. 11.25. Вкладка Training Parameters вікна налаштування навчання

2. Після здійснення налаштувань навчання натиснути кнопку *Train Network*. Почнеться процес навчання мережі й з'явиться вікно *Neural Network Training*, у якому буде відображатися процес навчання мережі (рис. 11.26). Натиснення кнопок *Performance, Training State, Regression* цього вікна дозволяє вивести результати та оцінити якість побудованої моделі класифікатора.

11.3.3. Робота зі створеною нейронною мережею, класифікація нових об'єктів

1. Для експорту створеної нейронної мережі у робочу область MatLab необхідно натиснути кнопку *Export*... та далі *Select All* (Вибрати все) і *Export*, потім – *Close*.

2. За допомогою імпортованої у робочу область MatLab нейронної мережі здійснимо класифікацію нового об'єкта, який не входив у навчаючу множину, зі значеннями $x_1 = 0{,}5$ і $x_2 = 1{,}3$. Для цього створимо скрипт із наступним кодом:

```
Xnew = [0.5; 1.3]; % задаємо новий об'єкт x
y = networkMy(Xnew); % y - результат класифікації об'єкту x
plotpv(P,T); % виведення об'єктів навчаючої множини
hold on; grid on; % візуалізація результату роботи мережі
scatter(Xnew(1), Xnew(2),60,'red','filled')
```

3. Результатом виконання даного скрипта буде візуалізація об'єктів навчаючої множини та нового об'єкта, якому на графіку відповідає точка, виділена червоним кольором (рис. 11.27).

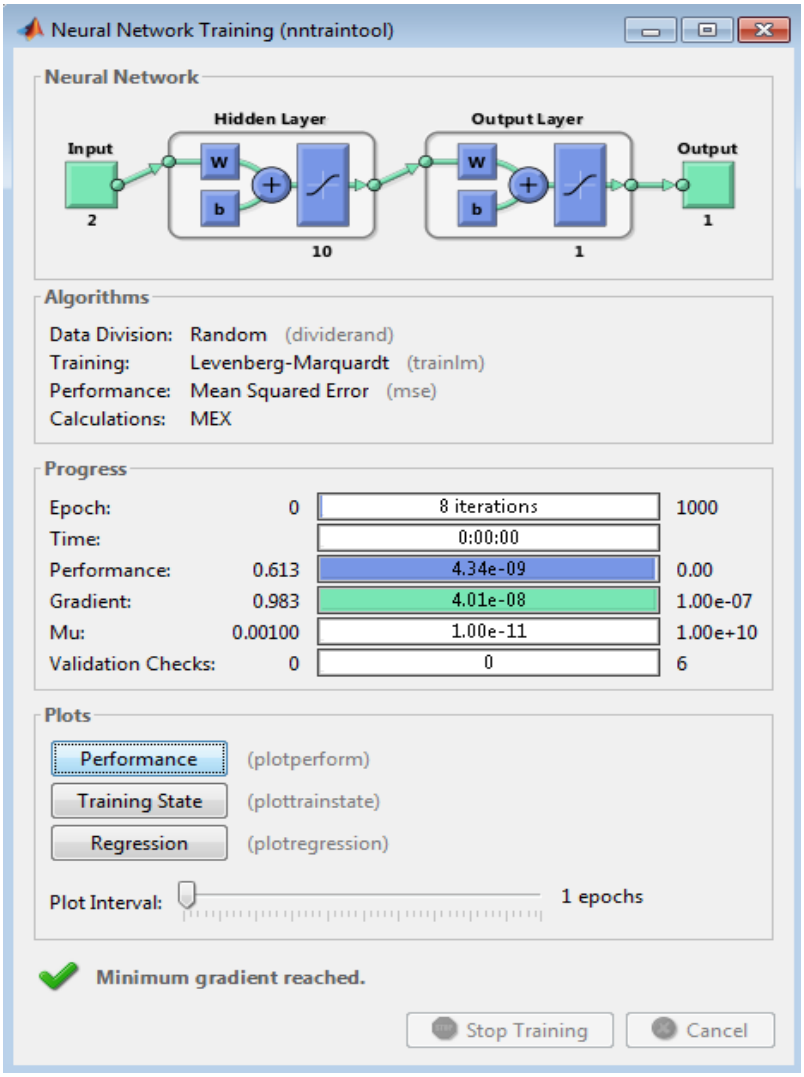


Рис. 11.26. Вікно Neural Network Training

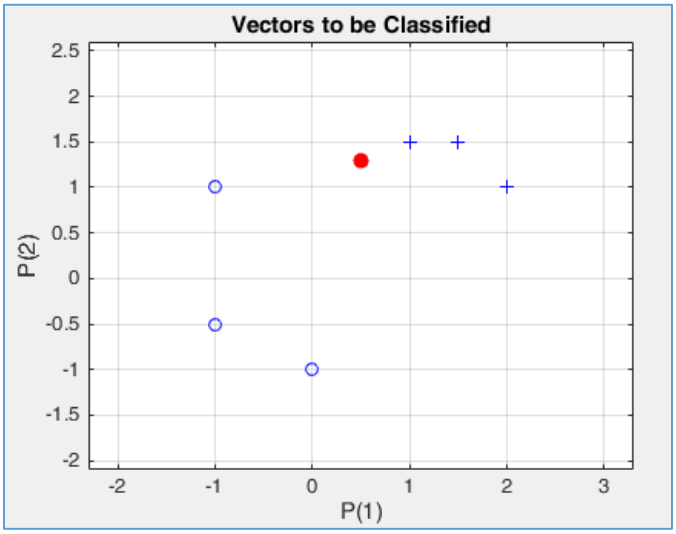


Рис. 11.27. Візуалізація об’єктів 2-х класів навченої множини та нового об’єкта

4. Виведемо результат класифікації нового об’єкта – значення змінної a на екран (рис. 11.28):

```
>> a = sim(networkMy, Xnew)
```

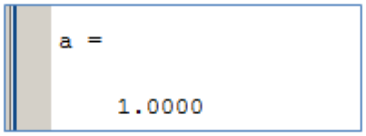


Рис. 11.28. Результат класифікації нового об'єкта

11.4. РОБОТА З МАЙСТРОМ КОНСТРУЮВАННЯ НЕЙРОННИХ МЕРЕЖ В MATLAB: ПРОГНОЗУВАННЯ ЧАСОВОГО РЯДУ

У MatLab існує можливість створення нейронних мереж за допомогою майстра, який дозволяє автоматично підбирати тип нейронної мережі для розв'язання задач класифікації, кластеризації, розпізнавання образів, прогнозування часових рядів.

Майстер дозволяє обрати тип задачі, завантажити необхідні для її розв'язання набори даних, побудувати та здійснити навчання нейронної мережі та використати її для розв'язання поставленої задачі. Розглянемо можливості майстра на прикладі створення нейронної мережі для розв'язання задачі прогнозування часових рядів.

Приклад 5. Ознайомитися з можливостями побудови нейронних мереж по прогнозуванню часового ряду за допомогою майстра MatLab.

1. Для відкриття вікна майстра створення нейронних мереж необхідно ввести команду: >> nnstart
2. У результаті виконання команди буде відкрите вікно створення нейронної мережі *Neural Network Start (nnstart)* (рис. 11.29). Для переходу до створення нейронної мережі для розв'язання задачі прогнозування необхідно натиснути кнопку *Time Series app*.

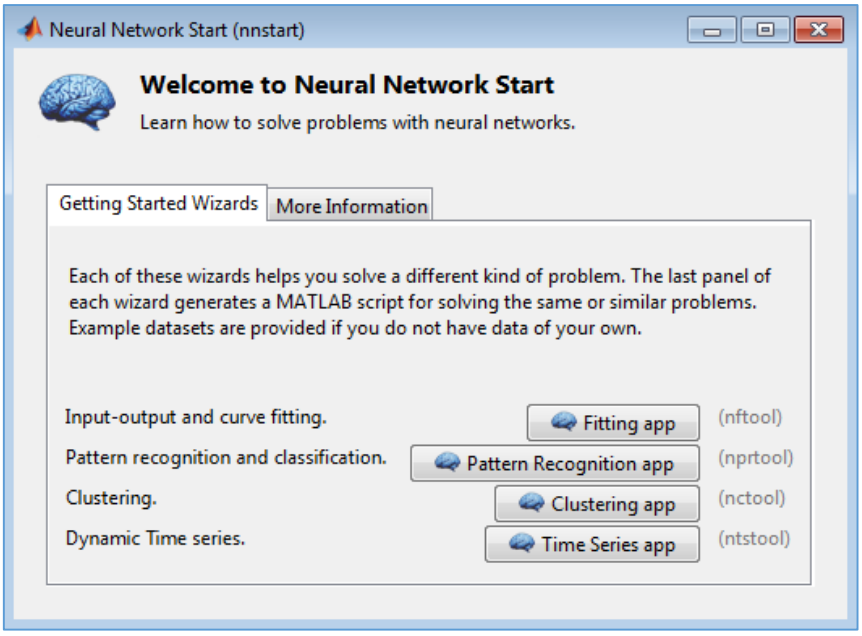


Рис. 11.29. Вікно Neural Network Start

3. Буде відкрите вікно створення нейронної мережі *Neural Time Series (nnstool)* (рис. 11.30). У правій частині вікна знаходиться панель вибору типу нейронної мережі: 1) нелінійна авторегресійна із зовнішнім входом: є простою нейронною мережею прямого поширення з реалізацією часового вікна отримання даних; 2) нелінійна вхід-вихід: є простою нейронною мережею прямого поширення; 3) нелінійна регресійна (генератор). Ставимо прапорець для вибору типу мережі *Nonlinear Autoregressive (NAR)* та натискаємо кнопку *Next.*

4. Буде відкрите вікно вибору даних для створення нейронної мережі *Neural Time Series (nnstool): Select Data.* У лівій частині вікна є можливість завантажити власні дані. Скористаємося можливістю завантаження готового набору даних для ознайомлення з можливостями майстра з налаштування нейронних мереж для часових рядів. Натискаємо кнопку *Load Example Data Set.*

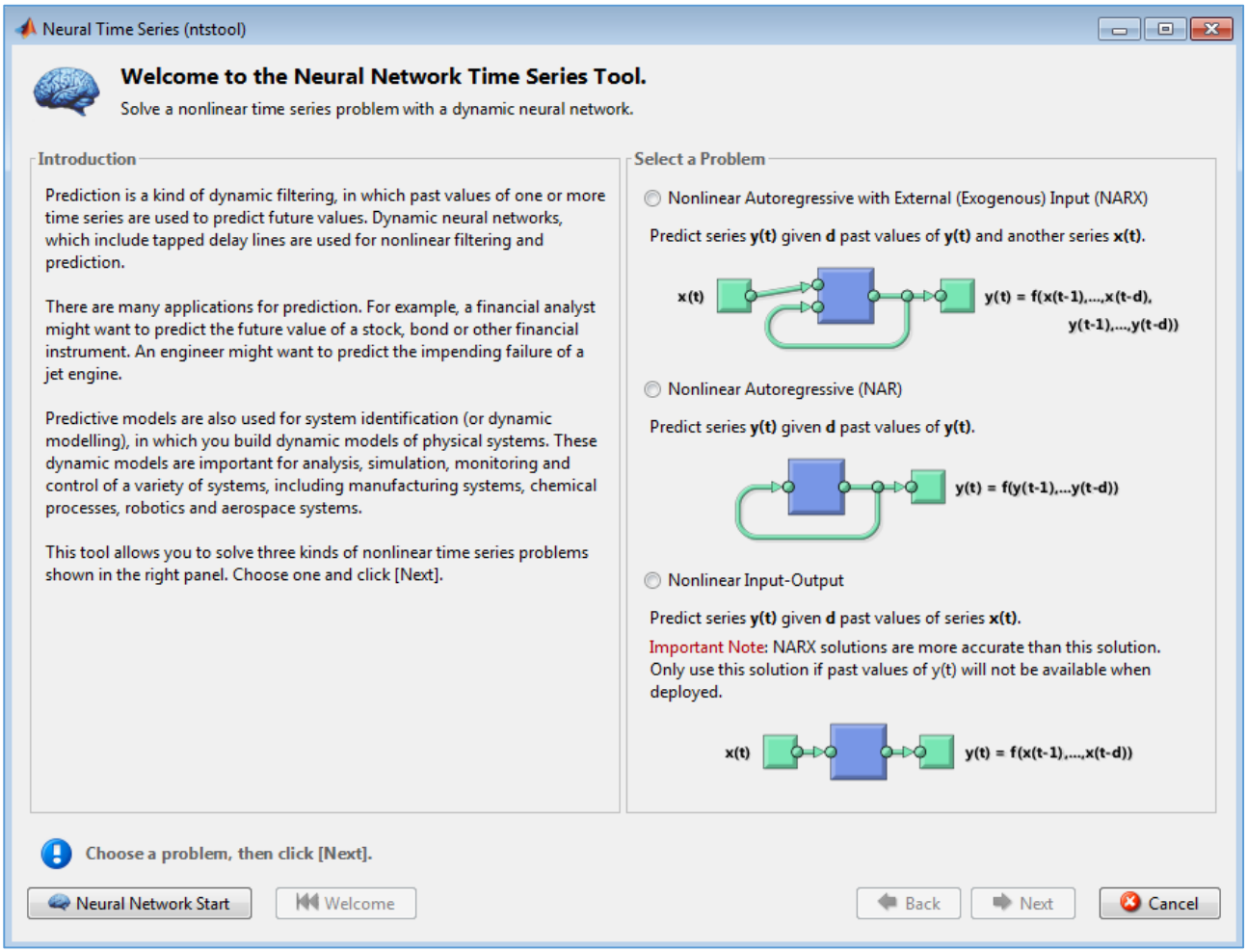


Рис. 11.30. Вікно Neural Time Series

5. У лівій частині вікна, що відкриється, обираємо Data Set *240 Years of Solar Sports*. У правій частині вікна буде відображена характеристика обраного набору даних (рис. 11.31). Після цього ознайомлення з особливостями набору даних натискаємо кнопку *Import*.

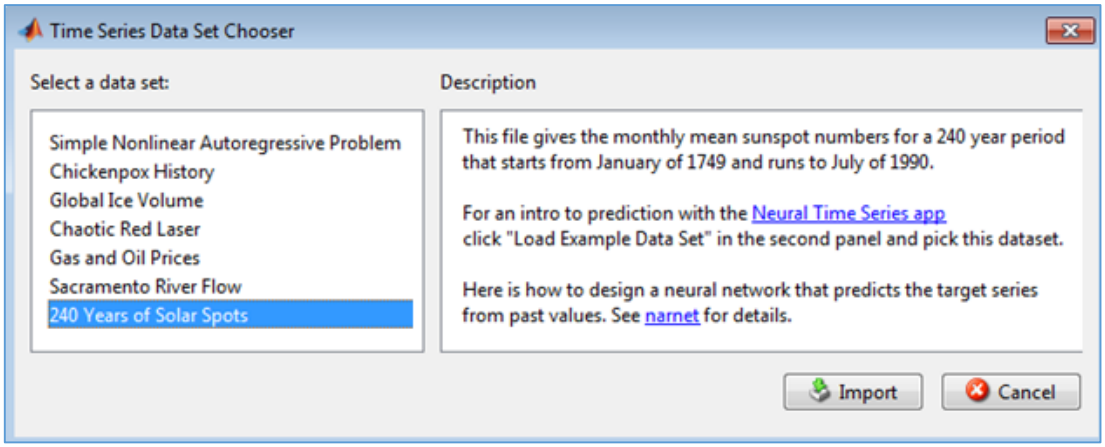


Рис. 11.31. Вікно вибору набору даних

6. У наступному вікні *Neural Time Series (nnstool): Select Data* буде вказано обраний набір даних (рис. 11.32). Цей набір даних використовують для навчання нейронної мережі прогнозування середньомісячної кількості плям на Сонці з урахуванням даних минулих років. Натискаємо кнопку *Next*.

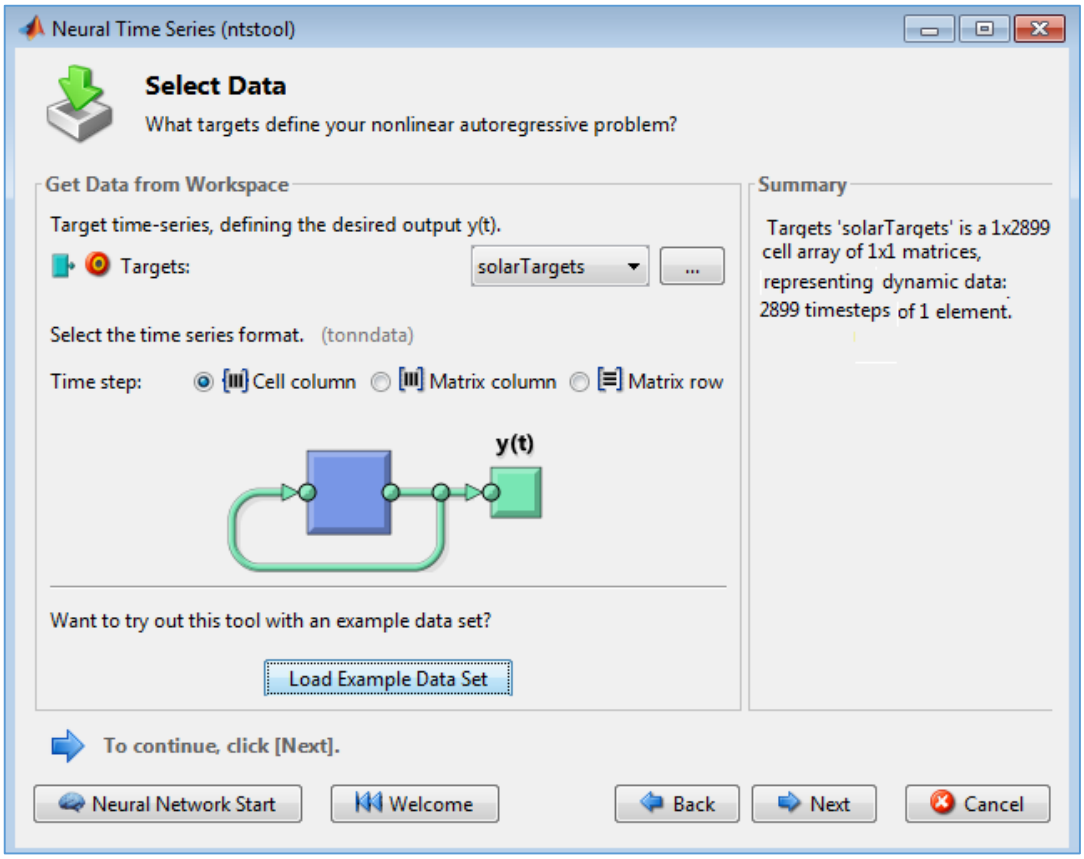


Рис. 11.32. Вікно Neural Time Series: Select Data

7. Далі відкриється вікно, у якому є можливість налаштувати поділ обраного набору даних на навчаючу, тестову та перевірочну множини у відсотковому відношенні (рис. 11.33). Залишаємо ті значення, які установлені по замовчуванню, та натискаємо кнопку *Next*.

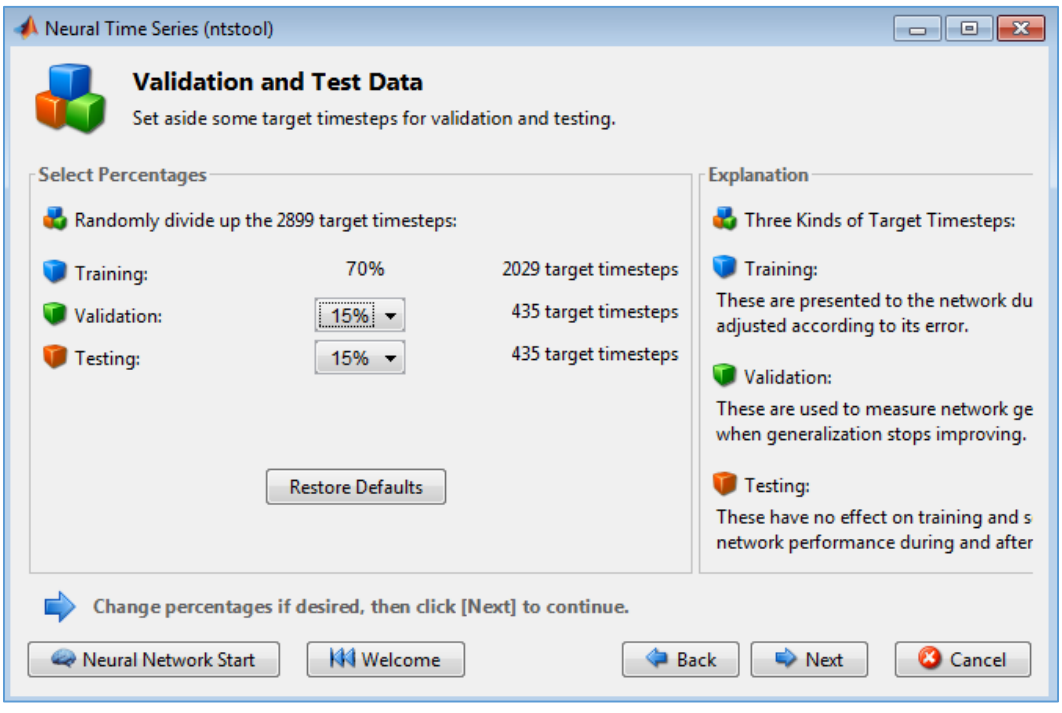


Рис. 11.33. Налаштування поділу набору даних на навчаючу, тестову та перевірочну множини

8. Відкриється вікно, у якому відображена архітектура створеної нейронної мережі прямого поширення та рекомендації, як з нею можна працювати у подальшому (рис. 11.34). У цьому вікні можна змінити кількість нейронів у шарах. А саму створену майстром нейронну мережу прямого поширення можна зробити рекурентною за допомогою команди: >> closeloop(net). Натискаємо кнопку *Next*. Відкриється вікно, у якому можна обрати алгоритм навчання мережі та запустити процес навчання, натиснувши кнопку *Train* (рис. 11.35).

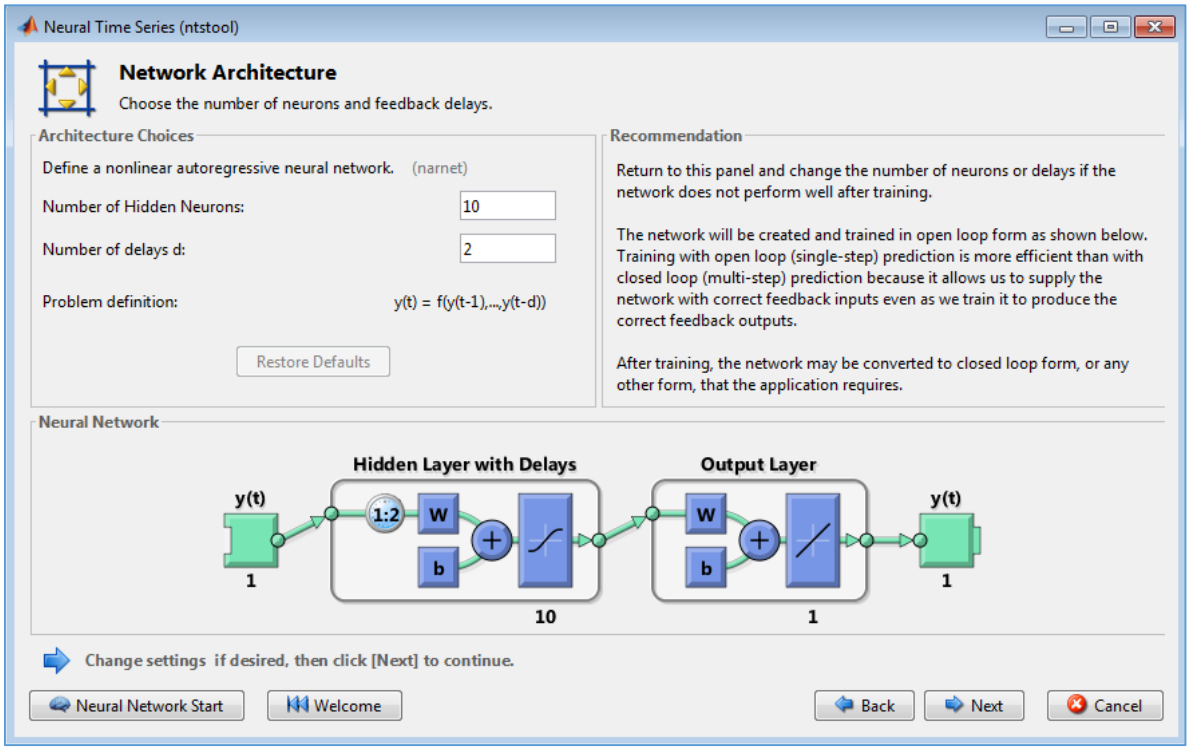


Рис. 11.34. Архітектура створеної нейронної мережі

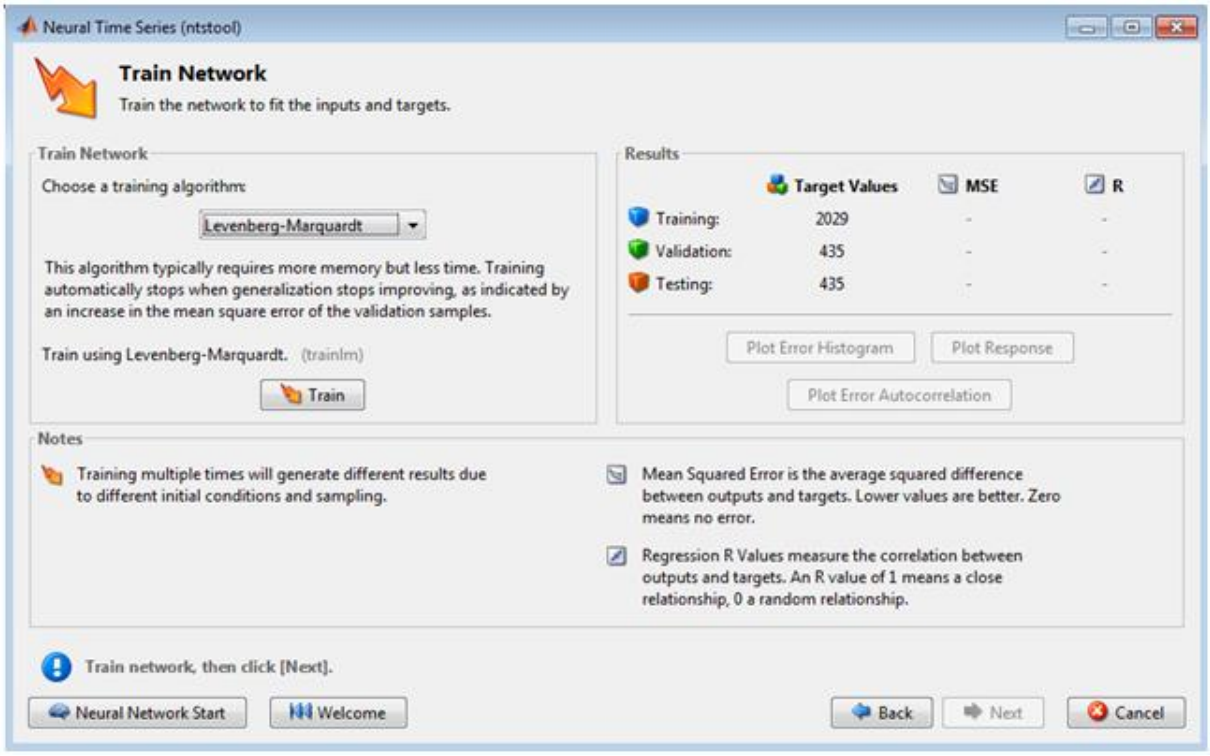


Рис. 11.35. Вікно Train Network

9. Буде запущено процес навчання мережі, результати якого можна переглянути у вікні *Neural Network Training* (рис. 11.36).

У цьому вікні після завершення навчання можна переглянути його характеристики.

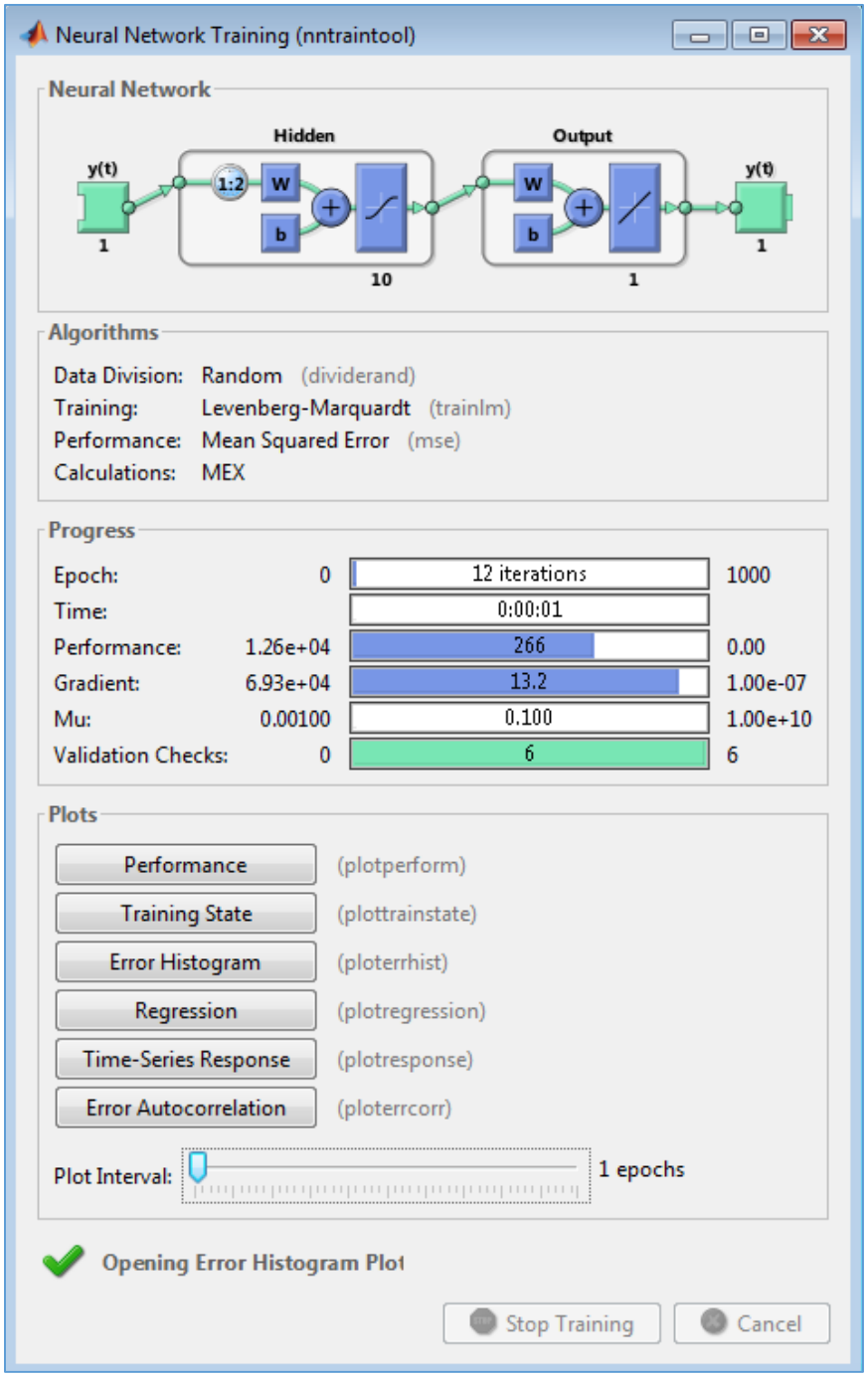


Рис. 11.36. Вікно Neural Network Training

10. Оскільки ми працювали із часовим рядом, вікно *Neural Network Training* має кнопки:

Error Autocorrelation – для виведення корелограми помилок за різними лагами часового ряду (рис. 11.37);

Time-Series Response – для виведення фактичних цільових значень часового ряду та значень часового ряду, розрахованих за побудованою моделлю, на одній осі із відображенням помилок між ними (рис. 11.38).

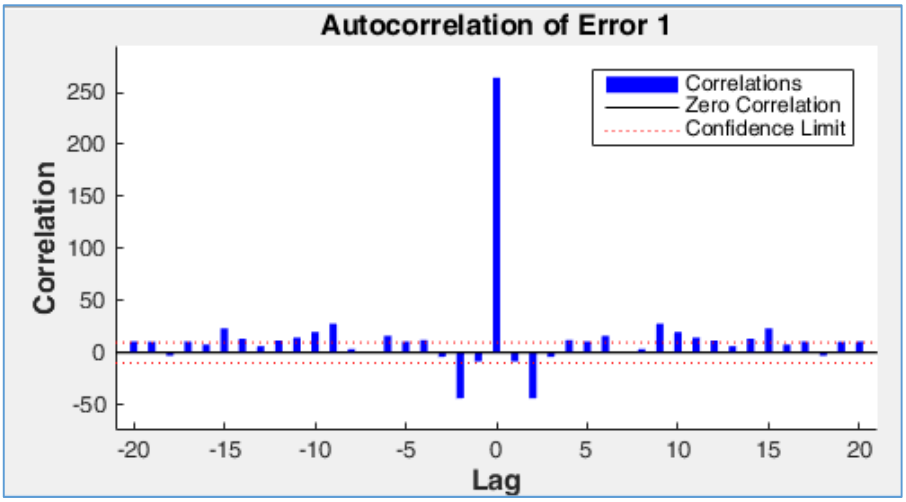


Рис. 11.37. Вікно Error Autocorrelation

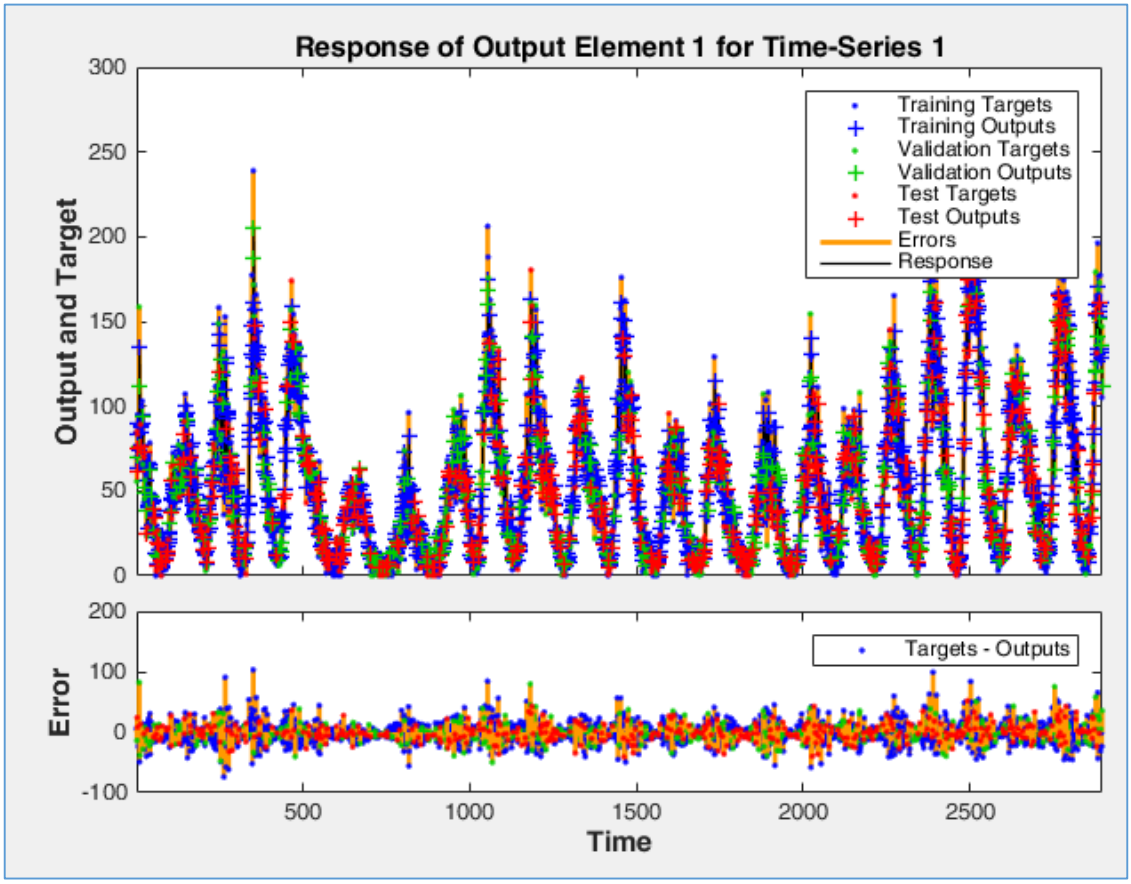


Рис. 11.38. Вікно Time-Series Response

11. Закриваємо вікно *Neural Network Training* і повертаємося до попереднього вікна *Neural Time Series: Train Network*. Після завершення навчання це вікно містить параметри, які характеризують якість побудованої моделі нейронної мережі: середньоквадратичне відхилення MSE та коефіцієнт детермінації R, кнопки для виведення гістограми і корелограми помилок та графіка часового ряду (рис. 11.39).

12. Натискаємо кнопки *Next – Next*. Відкриється вікно *Neural Time Series: Deploy Solution* (рис. 11.40).

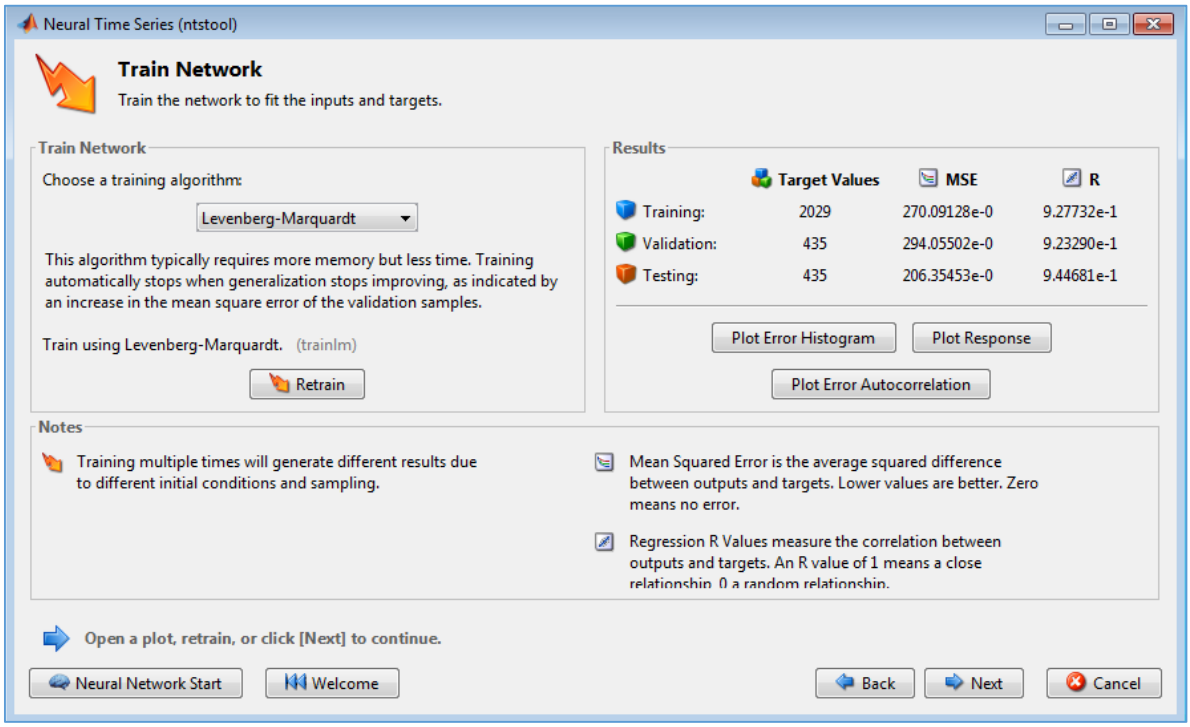


Рис. 11.39. Вікно Neural Time Series: Train Network

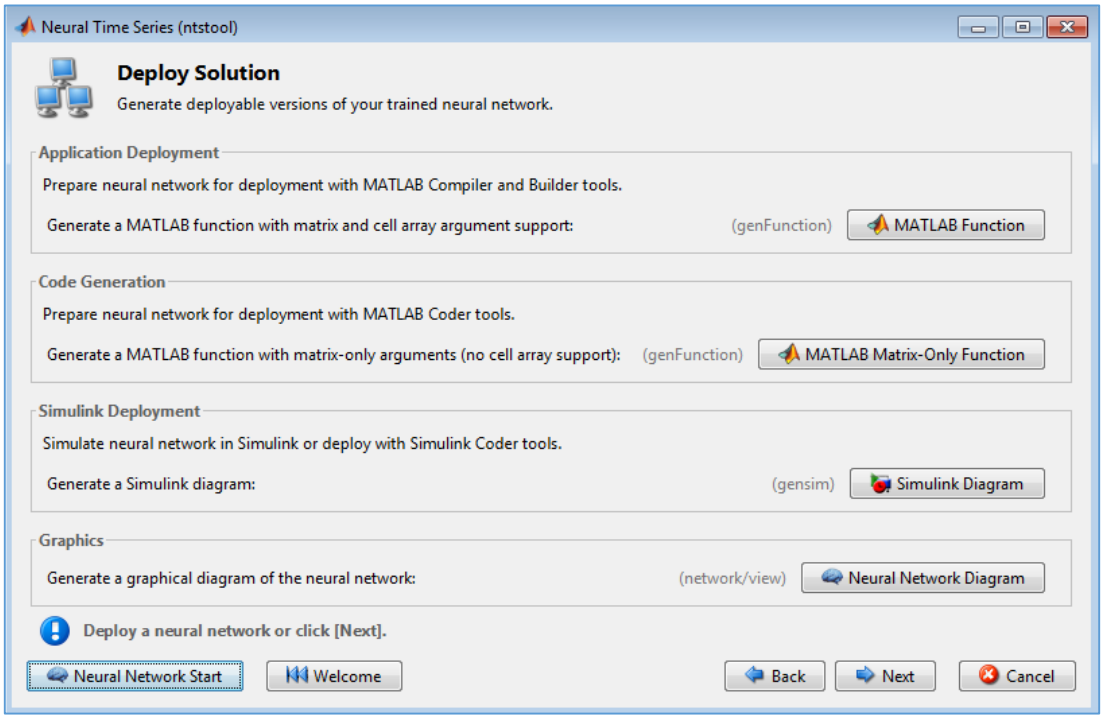


Рис. 11.40. Вікно Neural Time Series: Deploy Solution

12. У останньому вікні є можливість зберегти код, згенерований MatLab при побудові моделі часового ряду, у вигляді функції, натиснувши відповідну кнопку. Після цього можна закрити вікно *Neural Time Series: Deploy Solution* та *Neural Network Start*.

13. Код функції відкриється у вікні редактора MatLab, а далі його можна зберегти та редагувати у разі необхідності і використовувати для отримання прогнозу. Data Set, який було обрано для побудови моделі часового ряду, також буде у робочій області MatLab і його можна зберегти у mat-файлі.

11.5. ЗАВДАННЯ ДЛЯ САМОСТІЙНОЇ РОБОТИ

Завдання 6. Написати програму MatLab, яка здійснює:

1) поділ набору даних класичної задачі класифікації ірисів Фішера на дві множини – навчаючу та тестову;

2) побудову класифікатора за 2-ма ознаками шляхом створення та навчання нейронної мережі на наборі даних, відібраному до навчаючої множини;

3) перевірку результатів роботи класифікатора на наборі даних, відібраному до тестової множини;

4) класифікацію нових об'єктів, які не входять до навчаючої та тестової множин, із використанням побудованого класифікатора;

5) візуалізацію результатів роботи класифікації.

Вхідні дані представлені у файлі *fisheriris.mat*, доступному за посиланням: https://drive.google.com/file/d/10jUJyJfwnn3g7S5zG-6l74dNhIKKUeuo/view?usp=sharing.

Номера ознак n і m, відібраних для класифікації з набору даних та значення ознак нових об'єктів P1 і P2, які необхідно класифікувати, за варіантами представлені у таблиці 11.3.

КОНТРОЛЬНІ ПИТАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ № 11

1. Означення штучної нейронної мережі. Штучний нейрон.
2. Що таке внутрішній стан нейрона, постсинаптична функція?
3. Основні види функцій активації штучного нейрона.
4. Основні етапи побудови нейронної мережі для розв'язання задач Data Mining.
5. Що таке архітектура нейронної мережі?
6. Якими є види нейронних мереж?
7. Повнозв'язна, одношарова та багатошарова нейронні мережі – їх означення.
8. Будова та принцип роботи перцептрона.
9. Основні режими навчання нейронних мереж.
10. Що таке епоха, ітерація навчання нейронних мереж?
11. У чому полягає сутність алгоритму зворотного поширення помилки?
12. Методи навчання нейронних мереж: правило Хеба, правило Хопфілда, дельта-правило.
13. У чому полягають градієнтні методи навчання нейронних мереж?
14. Здійснення класифікації за допомогою нейронної мережі у середовищі MatLab.
15. Створення нейронної мережі в середовищі MatLab у графічному режимі.
16. Робота з майстром створення нейронних мереж у MatLab, прогнозування.

Таблиця 11.3

Дані за варіантами для виконання завдання 1

Варіант	n	m	P1				P2			
			x1	x2	x3	x4	x1	x2	x3	x4
1	1	4	4,69	2,58	1,73	0,14	5,38	2,90	4,24	1,60
2	2	3	4,37	3,24	1,47	0,54	5,37	3,57	6,79	1,75
3	1	2	6,54	3,33	4,37	1,53	6,82	3,33	6,85	2,12
4	1	3	5,69	3,00	1,29	0,45	5,44	2,34	3,72	1,19
5	2	4	5,60	2,44	1,06	0,40	5,39	3,74	6,23	1,93
6	1	3	6,84	2,05	4,08	1,60	5,28	3,02	6,87	2,34
7	2	4	5,07	3,33	1,45	0,47	5,73	3,03	3,19	1,39
8	1	3	5,20	3,55	1,49	0,44	5,73	2,98	5,13	2,46
9	2	4	6,70	2,74	4,07	1,48	5,57	2,99	4,75	1,52
10	1	3	4,65	3,41	1,86	0,38	6,47	2,84	3,95	1,22
11	2	4	4,74	3,38	1,17	0,38	7,13	2,28	4,98	1,69
12	1	3	6,45	2,72	4,40	1,44	7,83	3,06	6,15	1,83
13	1	4	5,26	4,36	1,44	0,14	5,58	3,34	4,38	1,29
14	1	2	4,30	3,42	1,20	0,31	6,23	2,47	4,87	1,43
15	2	3	6,59	3,00	3,46	1,58	5,26	3,52	4,92	1,74

Варіант	n	m	P1				P2			
			x1	x2	x3	x4	x1	x2	x3	x4
16	3	4	4,74	3,93	1,78	0,17	5,90	3,15	4,96	1,36
17	2	4	4,42	3,65	1,43	0,20	5,90	3,34	6,24	1,73
18	1	3	6,68	2,19	3,07	1,52	5,46	3,47	6,23	1,60
19	1	2	4,84	4,25	1,71	0,25	6,41	2,37	3,09	1,69
20	3	4	5,37	4,30	1,34	0,58	5,96	2,33	4,70	1,91
21	3	4	4,90	2,33	4,47	1,45	7,24	2,61	4,54	2,49
22	1	2	5,02	3,23	1,08	0,25	5,53	3,32	4,59	1,64
23	2	4	4,92	4,00	1,67	0,27	5,98	2,26	5,14	1,76
24	1	3	5,76	2,18	4,07	1,74	6,02	3,58	6,75	1,59
25	2	4	4,70	3,99	1,65	0,55	6,08	2,68	3,84	1,71

12. ВИЯВЛЕННЯ ЗВ'ЯЗКІВ І ЗАКОНОМІРНОСТЕЙ. КОРЕЛЯЦІЙНИЙ ТА ДИСПЕРСІЙНИЙ АНАЛІЗИ ДАНИХ

Лабораторна робота № 12

Мета: закріплення знань про базові підходи до виявлення зв'язків між ознаками об'єктів набору даних із застосуванням таблиць спряженості, кореляційного та дисперсійного аналізу даних. Набуття навичок аналізу таблиць спряженості, проведення кореляційного й однофакторного дисперсійного аналізу в програмах MS Excel, MatLab.

Теоретичні знання: аналіз зв'язків у наборі даних. Виявлення наявності зв'язків між ознаками з використанням таблиць спряженості. Кореляційний аналіз даних. Коефіцієнти кореляції Пірсона та Спірмена, їх значущість. Діаграма розсіювання. Однофакторний дисперсійний аналіз даних. Критерій Фішера.

12.1. АНАЛІЗ ЗВ'ЯЗКІВ У НАБОРІ ДАНИХ

12.1.1. Виявлення зв'язків між змінними

Виявлення зв'язків та закономірностей дозволяє розв'язувати такі задачі Data Mining:

1) аналіз зв'язків – задача знаходження залежностей у наборі даних;
2) оцінювання – прогноз неперервних значень ознаки;
3) прогнозування – на основі особливостей наявних даних оцінюються пропущені або майбутні значення ознаки.

Із метою розв'язання вказаних задач застосовують статистичні (дисперсійний, регресійний, кореляційний, факторний аналізи) та кібернетичні методи (нейронні мережі тощо). Серед статистичних методів виділяють:

1) ***параметричні методи*** – методи, які потребують необхідності контролювати відповідність розподілу досліджуваної змінної нормальному закону;
2) ***непараметричні методи*** – методи, які застосовують у випадках, коли розподіл досліджуваної змінної є відмінним від нормального.

Важливу інформацію про сутність явищ та процесів досліджуваної предметної області надає виявлення взаємозв'язків між ознаками об'єктів набору даних. При виявленні взаємозв'язку між ознаками, які характеризують об'єкти аналізованого набору даних, розрізняють:

1) ***незалежні змінні*** – відповідають ознакам, значення яких при проведенні дослідження можна змінювати (впливаючі ознаки);
2) ***залежні змінні*** – відповідають ознакам, значення яких при проведенні дослідження можна тільки вимірювати (результуючі ознаки).

Аналіз зв'язків (англ. *Link Analysis*) між змінними передбачає з'ясування:

1) ***наявності зв'язку*** між змінними;
2) ***тісноти зв'язку***, якщо він існує – наскільки узгодженою та сильною є взаємообумовлена зміна значень двох змінних;
3) ***характеру зв'язку*** – напряму взаємної зміни значень змінних: якщо збільшення значень однієї змінної викликає зростання значень іншої змінної – зв'язок є прямим, а якщо значення іншої змінної при цьому спадають – зв'язок є оберненим;
4) ***форми зв'язку*** – особливості сумісної зміни двох змінних, яку виявляють, аналізуючи стовпчасту діаграму чи таблицю спряженості (якщо хоча б одна змінна є номінальною) або діаграму розсіювання (для порядкових і числових шкал).

Залежно від шкал, в яких представлені змінні, з метою виявлення наявності зв'язку між ознаками застосовують: аналіз таблиць спряженості, кореляційний аналіз даних, дисперсійний аналіз даних (табл. 12.1).

При виявленні зв'язків та закономірностей розрізняють функціональну і стохастичну (статистичну) залежності між змінними, які характеризують об'єкти досліджуваної предметної області.

Функціональну залежність характеризує те, що кожному значенню однієї змінної відповідає цілком певне значення іншої змінної.

Стохастична або ***статистична залежність*** характеризується тим, що кожному значенню однієї змінної відповідає множина можливих значень іншої змінної (певний розподіл іншої змінної).

Статистична залежність проявляє себе у тому, що зміна значень однієї змінної викликає зміну розподілу значень іншої змінної. Це може бути обумовлено впливом неконтрольованих, не врахованих при проведенні дослідження факторів та наявністю похибок.

Таблиця 12.1

Вибір методу перевірки наявності зв'язку між ознаками

Вид аналізу	Кількість змінних, k	Шкала виміру		Закон розподілу	Коефіцієнт / критерій
		Незалежні змінні	Залежна змінна		
Кореляційний аналіз (параметричний)	$k = 2$	Числова шкала (інтервальна, шкала відношень)		Нормальний	Коефіцієнт кореляції Пірсона
Кореляційний аналіз (непараметричний)	$k = 2$	Порядкова та числова шкали (інтервальна, шкала відношень)		Відмінний від нормального	Коефіцієнт рангової кореляції Спірмена
Дисперсійний аналіз (параметричний)	$k \geq 2$	Номінальна та порядкова шкали	Числова шкала (інтервальна, шкала відношень)	Нормальний для залежної змінної	Критерій Фішера
Дисперсійний аналіз (непараметричний)	$k \geq 2$	Номінальна та порядкова шкали	Числова шкала (інтервальна, шкала відношень)	Відмінний від нормального для залежної змінної	Критерій Фрідмана
Аналіз таблиць спряженості (розмірністю $k \times m$)	$k = 2$	Номінальна шкала (k різних значень має 1-ша ознака й m різних значень – 2-га ознака)		–	Критерій згоди Пірсона

12.1.2. Виявлення наявності зв'язку між змінними з використанням таблиць спряженості

Побудова таблиць спряженості – ***кростабуляція*** – та їх аналіз є найбільш простим і розповсюдженим способом виявлення зв'язку між двома категоріальними змінними. За допомогою таблиць спряженості можна виявляти та аналізувати зв'язок і між змінними, представленими у числових шкалах (інтервальній та шкалі відношень), попередньо згрупувавши їх можливі значення у інтервали (табл. 12.2).

Таблиця 12.2

Приклад таблиці спряженості для змінних x_i та x_j

Значення змінної x_j	Значення змінної x_i				$n_j = \sum_{i=1}^{k} n_{ij}$
	i = 1	i = 2	…	i = k	
j = 1	n_{11}	n_{12}	…	n_{1k}	$\sum_i n_{i1}$
j = 2	n_{21}	n_{22}	…	n_{2k}	$\sum_i n_{i2}$
…	…	…	…	…	…
j = m	n_{m1}	n_{m2}	…	n_{mk}	$\sum_i n_{im}$
$n_i = \sum_{j=1}^{m} n_{ij}$	$\sum_j n_{j1}$	$\sum_j n_{j2}$	…	$\sum_j n_{jk}$	$n = \sum_i \sum_j n_{ij}$

Кожен рядок ***таблиці спряженості*** відповідає можливим значенням (інтервалам) однієї змінної, а стовпець – іншої. У комірках таблиці містяться ***частоти*** n_{ij} – число об'єктів, у яких виявлено відповідне поєднання значень змінних. У комірках таблиці спряженості частоти можуть бути абсолютними або відносними (в долях або %).

Основні етапи виявлення зв'язку між змінними x_i та x_j є наступними.

1. За значеннями ознак об'єктів набору даних визначають спостережувані частоти n_{ij} та будують таблицю спряженості для змінних x_i і x_j.

2. Формулюють нульову та альтернативну гіпотези:

H_0: між досліджуваними ознаками статистично значущий зв'язок відсутній;

H_1: між досліджуваними ознаками статистично значущий зв'язок є.

3. Розраховують очікувані при H_0 частоти n'_{ij} за формулою:

$$n'_{ij} = \frac{n_i \cdot n_j}{n}, \tag{12.1}$$

де $n_i = \sum_{j=1}^{m} n_{ij}$ – сума частот по *i*-му стовцю таблиці спряженості,

$n_j = \sum_{i=1}^{k} n_{ij}$ – сума частот по *j*-му рядку таблиці спряженості,

$n = \sum_i \sum_j n_{ij}$ – сума частот усіх комірок таблиці спряженості,

k – кількість значень (інтервалів) змінної x_i, m – кількість значень (інтервалів) змінної x_j.

4. Розраховують емпіричне значення критерію згоди Пірсона χ^2 за формулою, яка для частот, представлених у таблиці спряженості, має вигляд:

$$\chi^2_{емп} = \sum_{i=1}^{k} \sum_{j=1}^{m} \frac{(n_{ij} - n'_{ij})^2}{n'_{ij}}. \tag{12.2}$$

5. Визначають критичне значення критерію Пірсона $\chi^2_{\alpha,l}$ на рівні значущості α для числа ступенів свободи $l = (m-1)\cdot(k-1)$.

6. Порівнюють $\chi^2_{емп}$ та $\chi^2_{\alpha,l}$: якщо $\chi^2_{емп} \le \chi^2_{\alpha,l}$ – нульова гіпотеза про відсутність статистично значущого зв'язку між ознаками приймається з ймовірністю $p = 1-\alpha$. У протилежному випадку, якщо емпіричне значення критерію згоди Пірсона більше за критичне $\chi^2_{емп} > \chi^2_{\alpha,l}$, приймається альтернативна гіпотеза: між ознаками є статистично значимий зв'язок із ймовірністю $p = 1-\alpha$.

У разі виявлення зв'язку між змінними, представленими у числових та порядкових шкалах, аналіз таблиці спряженості дозволяє з'ясувати характер зв'язку. Якщо зростання значень (рангу) однієї ознаки супроводжується зростанням значень (рангу) іншої ознаки, зв'язок буде прямим, а якщо значення (ранг) іншої ознаки будуть спадати при цьому – оберненим.

Здійснюючи аналіз таблиць спряженості із використанням критерію згоди Пірсона доцільно дотримуватися наступних обмежень: $n_{ij} < 5$ повинно бути не більше, ніж для 20% комірок, $n > 40$.

Приклад 1. Є набір даних із 540 виборців на виборах президента, серед яких було 270 жінок та 270 чоловіків. Відомо, що за «кандидата А» проголосувало 240 виборців, а за «кандидата В» – 300 виборців. При цьому 190 чоловіків підтримали «кандидата А», і 80 – «кандидата В». Необхідно побудувати таблицю спряженості та визначити, чи є взаємозв'язок між статтю і політичними перевагами виборців.

1. Для побудови таблиці спряженості визначимо кількість жінок, які підтримали «кандидата А» – від 240 виборців (тих, хто його підримав) необхідно відняти 190 (кількість чоловіків серед тих, хто його підтримав): 240 – 90 = 50. Отже, 50 жінок підтримали «кандидата А». Тоді «кандидата В» підтримало 270 – 50 = 220 жінок. Будуємо таблицю спряженості (табл. 12.3).

Таблиця 12.3

Таблиця спряженості для змінних «стать» та «політичні переваги виборців»

Значення змінної «Стать»	Значення змінної «Політичні переваги виборців»		n_i
	Кандидат А	Кандидат В	
Ж	50	220	270
Ч	190	80	270
n_j	240	300	540

2. Візуальний аналіз вмісту таблиці спряженості дає підстави стверджувати, що між ознаками «стать» та «політичні переваги виборців» є зв'язок, оскільки кандидата А підтримує переважна кількість чоловіків, а кандидата В – переважна кількість жінок.

3. Для достовірного підтвердження зв'язку між ознаками формулюємо нульову та альтернативну гіпотези:

H_0: між досліджуваними ознаками статистично значущий зв'язок відсутній;

H_1: між досліджуваними ознаками є статистично значущий зв'язок.

4. Розраховуємо за формулою 12.1 очікувані при H_0 частоти n'_{ij}:

$$n'_{11} = \frac{240 \cdot 270}{540} = 120 \ , \quad n'_{12} = \frac{300 \cdot 270}{540} = 150 \ ,$$

$$n'_{21} = \frac{240 \cdot 270}{540} = 120 \ , \quad n'_{22} = \frac{300 \cdot 270}{540} = 150 \ .$$

5. Розраховуємо за формулою 12.2 емпіричне значення критерію згоди Пірсона:

$$\chi^2_{емп} = \frac{(n_{11} - n'_{11})^2 + (n_{12} - n'_{12})^2 + (n_{21} - n'_{21})^2 + (n_{22} - n'_{22})^2}{n'_{11} + n'_{12} + n'_{21} + n'_{22}},$$

$$\chi^2_{емп} = \frac{(50-120)^2 + (220-150)^2 + (190-120)^2 + (80-150)^2}{120+120+150+150} = 36{,}2963 .$$

6. Знаходимо на рівні значущості $\alpha = 0{,}05$ для числа ступенів свободи $l = (m-1)\cdot(k-1) = (2-1)\cdot(2-1) = 1$ критичне значення критерію згоди Пірсона, використовуючи функцію MS Excel ХИ2.ОБР.ПХ($\alpha;l$)/ CHISQ.INV.RT($\alpha;l$). Для цього у комірку робочого аркуша необхідно ввести формулу:

=ХИ2.ОБР.ПХ(0,05;1).

Отримуємо $\chi^2_{\alpha,l} = 0{,}0039$.

7. Порівнявши емпіричне $\chi^2_{емп} = 36{,}2963$ та критичне значення критерію Пірсона $\chi^2_{\alpha,l} = 0{,}0039$, маємо $\chi^2_{емп} > \chi^2_{\alpha,l}$. Отже, з ймовірністю 95% можемо відхилити нульову гіпотезу і прийняти альтернативну та стверджувати, що між ознаками «стать» і «політичні переваги виборців» є статистично значущий зв'язок.

12.1.3. Кореляційний аналіз даних

Кореляційний аналіз є сукупністю методів для виявлення статистичної залежності між ознаками набору даних, які дають можливість отримати інформацію про ймовірність появи певних значень однієї змінної під впливом зміни значень інших змінних або іншої змінної. Як числову характеристику ймовірнісного зв'язку величин використовують коефіцієнти кореляції.

Коефіцієнт кореляції набуває значень у інтервалі [-1; 1] (рис. 12.1). Від'ємні значення коефіцієнта свідчать про наявність оберненого зв'язку між змінними: збільшення значень однієї змінної супроводжується зменшенням

значень іншої змінної. Додатні значення коефіцієнта кореляції відповідають прямому зв'язку між змінними: збільшення значень однієї змінної супроводжується збільшенням значень іншої ознаки.

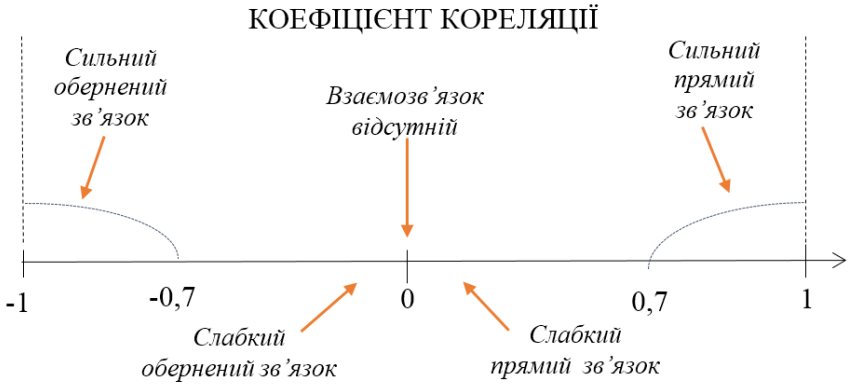


Рис. 12.1. Тіснота та характер зв'язку при різних значеннях коефіцієнта кореляції

Тіснота (сила) кореляційних зв'язків не залежить від їх направленості і визначається емпірично за абсолютним значенням коефіцієнта кореляції (табл. 12.4). Чим ближчим є значення коефіцієнта до 1 або до -1, тим сильнішим є зв'язок. Якщо значення коефіцієнта ближче до 0, зв'язок є слабким.

Кореляційний аналіз дозволяє визначати тісноту та характер зв'язку між досліджуваними величинами й кількісно оцінити ступінь не випадковості їх сумісних змін. Проте наявність кореляційного зв'язку між змінними не є показником існування між ними причинно-наслідкового зв'язку. Їх сумісна змінна може бути опосередкована іншими змінними, з якими характер зв'язку досліджуваних величин є причинно-наслідковим.

Таблиця 12.4

Класифікація тісноти кореляційних зв'язків

Значення коефіцієнта кореляції r (*по модулю*)	Інтерпретація
$\lvert r\rvert \geq 0{,}9$	Дуже сильний зв'язок
$0{,}7 \leq \lvert r\rvert < 0{,}9$	Сильний, тісний зв'язок
$0{,}5 \leq \lvert r\rvert < 0{,}7$	Середній зв'язок
$0{,}3 \leq \lvert r\rvert < 0{,}5$	Помірний зв'язок
$0{,}2 \leq \lvert r\rvert < 0{,}3$	Слабкий зв'язок
$\lvert r\rvert < 0{,}2$	Дуже слабкий зв'язок

Кореляція може бути як лінійною, так і нелінійною. Для визначення тісноти та напряму ***лінійного зв'язку*** двох змінних x_i та x_j розраховують ***коефіцієнт кореляції Пірсона*** (англ. *Pearson's correlation coefficient*) r:

$$r = \frac{\text{cov}\,(x_i, x_j)}{\sigma_{x_i} \cdot \sigma_{x_j}}, \tag{12.3}$$

де σ_{x_i} та σ_{x_j} – стандартні відхилення змінних x_i і x_j, $\text{cov}\,(x_i, x_j)$ – коваріація змінних x_i та x_j.

Коваріація є сумісною варіацією двох змінних x_i та x_j, і розраховується за формулою:

$$\text{cov}(x_i, x_j) = \sum_{k=1}^{n} (x_{ik} - \bar{x}_i) \cdot (x_{jk} - \bar{x}_j), \tag{12.4}$$

де x_{ik} – k-те значення змінної x_i, x_{jk} – k-те значення змінної x_j,

$\bar{x}_i$ та $\bar{x}_j$ – середні значення змінних x_i і x_j,

n – кількість об'єктів у наборі даних.

Коваріація $\text{cov}_в(x_i, x_j)$, визначена за вибірковими значеннями ознак набору даних, є зміщеною точковою оцінкою коваріації генеральної сукупності $\text{cov}(x_i, x_j)$, тобто $\text{cov}_в(x_i, x_j) \neq \text{cov}(x_i, x_j)$. Незміщеною точковою оцінкою коварації є ***виправлена вибіркова коваріація***:

$$\text{cov}(x_i, x_j) = \frac{n}{n-1}\text{cov}_в(x_i, x_j). \quad (12.5)$$

Змінні, між якими досліджується зв'язок із використанням коефіцієнта кореляції Пірсона, повинні бути представлені в інтервальній шкалі або шкалі відношень.

У випадку нелінійного зв'язку між змінними коефіцієнт кореляції Пірсона дає хибні значення. Якщо значення коефіцієнта є близьким до 0, між змінними лінійний статистично значущий зв'язок відсутній, але наявність нелінійного зв'язку не виключена. Для виявлення нелінійних залежностей між змінними використовують інші методи, частіше всього – регресійний аналіз даних.

Дослідження зв'язку між змінними із використанням коефіцієнта кореляції Пірсона є параметричним методом. Тому перед його застосуванням необхідно здійснити перевірку, чи є розподіл змінних близьким до нормального.

До непараметричних методів визначення зв'язку між змінними x_i і x_j відноситься ***коефіцієнт рангової кореляції Спірмена*** r_s:

$$r_s = 1 - \frac{6 \cdot \sum_{k=1}^{n} d_k^2}{n \cdot (n^2 - 1)}, \quad (12.6)$$

де $d_k = r_{ik} - r_{jk}$ – різниця між k-ми рангами змінних x_i та x_j, n – кількість рангів.

Коефіцієнт кореляції Спірмена називається ранговим, тому що для його визначення значення змінних необхідно перетворити у ранги. Для цього значення змінних розміщують у вигляді ранжованого ряду, після чого кожному значенню привласнюється ранг від 1 до n, де n – кількість рангів.

Підтвердження значущості коефіцієнта кореляції здійснюється шляхом перевірки статистичних гіпотез із використанням ***критерію Стьюдента***. Основні етапи перевірки є такими.

1. Формулюють нульову та альтернативну гіпотези:

H_0: $r = 0$, коефіцієнт кореляції генеральної сукупності рівний нулю;

H_1: $r \neq 0$, коефіцієнт кореляції генеральної сукупності відмінний від нуля.

2. Обчислюють емпіричне значення критерію Стьюдента $t_{емп}$:

$$t_{емп} = \frac{|r|}{\sqrt{1-r^2}} \cdot \sqrt{n-2}, \quad (12.7)$$

де r – розраховане за вибірковими даними значення коефіцієнта кореляції, n – кількість об'єктів у наборі даних.

3. Із таблиць або за допомогою функцій програмних засобів визначають критичне значення критерію Стьюдента $t_{кр}$ на рівні значущості α для числа ступенів свободи $l = n - 2$.

4. Порівнюють $t_{емп}$ та $t_{кр}$: якщо $t_{емп} \leq t_{кр}$ – нульова гіпотеза про рівність нулю коефіцієнта кореляції генеральної сукупності (відсутність статистично значимого зв'язку між ознаками) приймається з ймовірністю $p = 1 - \alpha$. У протилежному випадку, якщо емпіричне значення критерію Стьюдента більше за критичне $t_{емп} > t_{кр}$, приймається альтернативна гіпотеза: коефіцієнт кореляції відмінний від нуля із ймовірністю $p = 1 - \alpha$.

При проведенні ***кореляційного аналізу в MS Excel*** використовують такі функції:

1) для розрахунку коефіцієнта кореляції Пірсона:

$$PEARSON\,(масив1;масив2)\ /\ КОРРЕЛ\,(масив1;масив2),$$

де *масив1* – масив значень першої змінної, *масив2* – масив значень другої змінної;

2) для розрахунку критерію Стьюдента:

$$T.INV.2T(\alpha;n-2)\ /\ СТЬЮДЕНТ.ОБР.2X(\alpha;n-2),$$

де α – рівень значущості, $n-2$ – число ступенів свободи, n – кількість об'єктів у наборі даних;

3) для розрахунку вибіркової коваріації двох змінних:

$$COVARIANCE.P(масив1;масив2)\,/\,КОВАРИАЦИЯ.Г\,(масив1;масив2),$$

де *масив1* – масив значень першої змінної, *масив2* – масив значень другої змінної;

4) для розрахунку виправленої вибіркової коваріації двох змінних:

$$COVARIANCE.S(масив1;масив2)\,/\,КОВАРИАЦИЯ.В(масив1;масив2),$$

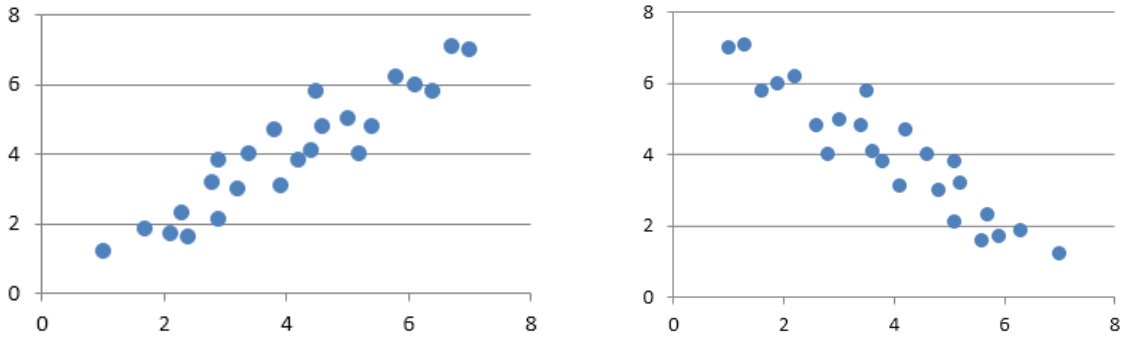
де *масив1* – масив значень першої змінної, *масив2* – масив значень другої змінної.

12.1.4. Діаграма розсіювання

Наочне уявлення про характер кореляційного зв'язку дає ***діаграма розсіювання*** – точкова діаграма, яка використовується для відображення спільного розподілу двох змінних і дозволяє візуально оцінити тісноту зв'язку між досліджуваними ознаками.

Емпіричні правила визначення наявності кореляції (рис. 12.2, рис. 12.3):

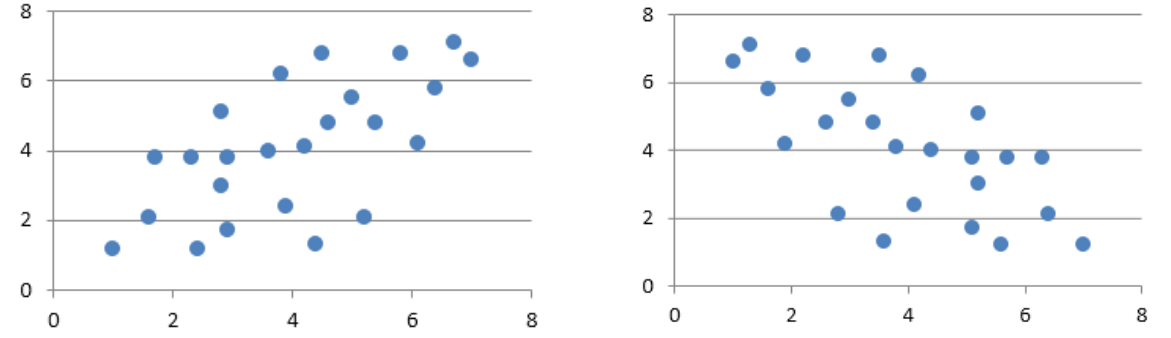
1. Якщо еліпс мінімальної площі, який охоплює усі точки діаграми розсіяння, має достатньо витягнуту форму, між випадковими величинами є зв'язок.



а) прямий зв'язок: r = 0,93 б) обернений зв'язок: r = -0,93

(*між ознаками є сильний лінійний зв'язок*)

Рис. 12.2. Діаграми розсіювання сильного кореляційного зв'язку



а) прямий зв'язок: r = 0,64 б) обернений зв'язок: r = -0,64

(*між ознаками може бути лінійний зв'язок*)

Рис. 12.3. Діаграми розсіювання середнього кореляційного зв'язку

2. Чим більш витягнутою є форма еліпса, тим сильніший зв'язок між ознаками.
3. Якщо більша вісь еліпса утворює із віссю абсцис гострий кут, то зв'язок прямий.
4. Якщо кут між більшою віссю еліпса й віссю абсцис тупий, тоді зв'язок між ознаками обернений.

Приклад 2. Для даних, які характеризують групу студентів із 7 осіб за двома ознаками «кількість пропущених занять» та «підсумковий рейтинг» (табл. 12.5) дослідити зв'язок між ознаками, побудувавши діаграму розсіювання та визначивши: 1) коефіцієнт кореляції Пірсона; 2) коефіцієнт рангової кореляції Спірмена; 3) статистичну значущість розрахованих коефіцієнтів кореляції.

Таблиця 12.5

Інформація про пропуски занять та підсумковий рейтинг студентів

№ з/п	Значення змінної *«кількість пропущених занять»*	Значення змінної *«підсумковий рейтинг»*
1	6	82
2	2	86
3	15	43
4	9	74
5	12	58
6	5	90
7	8	78

1. За даними таблиці 12.4 засобами MS Excel будуємо діаграму розсіювання (рис. 12.4).

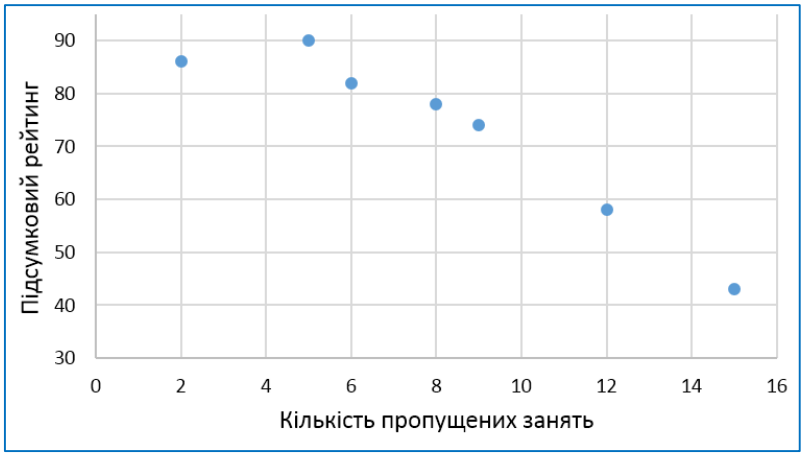


Рис. 12.4. Діаграма розсіювання для змінних «кількість пропущених занять» та «кідсумковий рейтинг»

2. Візуальний аналіз графічного зображення дозволяє стверджувати, що між досліджуваними ознаками існує тісний, сильний лінійний зв'язок, оскільки точки діаграми належать досить витягнутому вздовж більшої осі еліпса. А кут нахилу осі еліпса свідчить про те, що характер зв'язку між ознаками є оберненим. Достовірність та статистична значущість такого зв'язку між ознаками може бути підтверджена шляхом проведення кореляційного аналізу.

3. Сформуємо таблицю MS Excel для визначення коефіцієнта кореляції Пірсона за формулою 12.3. Із цією метою скористаємося функцією MS Excel, увівши в комірку D11 формулу (рис. 12.5): =КОРРЕЛ(C3:C9;D3:D9).

4. Розраховане значення коефіцієнта кореляції Пірсона, рівне r = -0,944, свідчить про наявність сильного оберненого зв'язку між досліджуваними ознаками.

5. Для визначення коефіцієнта рангової кореляції Спірмена необхідно значення кожної змінної перетворити у ранг. Для цього розмістимо значення ознак у порядку їх зростання, поставивши у відповідність кожному значенню ознаки його ранг (рис. 12.6).

6. Сформуємо таблицю MS Excel із рангами кожної змінної і квадратами різниць відповідних рангів та розрахуємо коефіцієнт рангової кореляції Спірмена за формулою 12.6, ввівши у комірку M12 формулу (рис. 12.7):

=1-(6*M10)/(H9*(H9^2-1)).

7. Розраховане значення рангової кореляції Спірмена, рівне r_s = -0,964, також свідчить про наявність сильного оберненого зв'язку між досліджуваними ознаками «Кількість пропущених занять» та «Підсумковий рейтинг».

	A	B	C	D
1				
2		№ п/пр	Кількість пропущених занять, x_1	Підсумковий ретинг, x_2
3		1	6	82
4		2	2	86
5		3	15	43
6		4	9	74
7		5	12	58
8		6	5	90
9		7	8	78
10			*Коефіцієнт кореляції Пірсона*	
11			=КОРРЕЛ(C3:C9;D3:D9)	-0,944

Рис. 12.5. Розрахунок коефіцієнта кореляції Пірсона

Кількість пропущених занять, x_1	Підсумковий ретинг, x_2	Ранг, r_k
2	43	1
5	58	2
6	74	3
8	78	4
9	82	5
12	86	6
15	90	7

Рис. 12.6. Визначення рангів двох змінних

	G	H	I	J	K	L	M
1							
2		№ п/пр	Кількість пропущених занять, x_1	Ранг, r_1	Підсумковий ретинг, x_2	Ранг, r_2	$(d_k=r_{1k}-r_{2k})^2$
3		1	6	3	82	5	4
4		2	2	1	86	6	25
5		3	15	7	43	1	36
6		4	9	5	74	3	4
7		5	12	6	58	2	16
8		6	5	2	90	7	25
9		7	8	4	78	4	0
10						Всього	110
11				*Коефіцієнт рангової кореляції Спірмена*			
12				=1-6*(M10)/(H9*(H9^2-1))			-0,964

Рис. 12.7. Розрахунок коефіцієнта рангової кореляції Спірмена

8. Для визначення значущості розрахованих коефіцієнтів кореляції обчислимо емпіричне значення критерію Стьюдента за формулою 12.7 та критичне значення критерію Стьюдента з використанням функції MS Excel =СТЬЮДЕНТ.ОБР.2Х() (рис. 12.8):

а) розрахунок емпіричного значення критерію Стьюдента $t_{1емп}$ для коефіцієнта кореляції Пірсона: =ABS(D11)*КОРЕНЬ(B9-2)/КОРЕНЬ(1-D11^2);

b) розрахунок емпіричного значення критерію Стьюдента $t_{2емп}$ для коефіцієнта рангової кореляції Спірмена: =ABS(M12)*КОРЕНЬ(H9-2)/КОРЕНЬ(1-M12^2);

c) розрахунок критичного значення критерія Стьюдента $t_{кр}$ на рівні значущості $\alpha = 0{,}01$ для числа ступенів свободи $l = 7 - 2 = 5$: =СТЬЮДЕНТ.ОБР.2Х(0,01;(B9-2)).

Критерій Стьюдента	
Емпіричне значення для коефіцієнта кореляції Пірсона	
6,41	=ABS(D11)*КОРЕНЬ(B9-2)/КОРЕНЬ(1-D11^2)
Емпіричне значення для коефіцієнта рангової кореляції Спірмена	
8,14	=ABS(M12)*КОРЕНЬ(H9-2)/КОРЕНЬ(1-M12^2)
Критичне значення	
4,032	=СТЬЮДЕНТ.ОБР.2Х(0,01;(B9-2))

Рис. 12.8. Розрахунок значень критерія Стьюдента

9. Порівнявши розраховані емпіричні значення критерія Стьюдента $t_{1емп}$ = 14,335 та $t_{2емп}$ =18,203 із критичним значенням критерія Стьюдента $t_{кр}$ = 4,032, ми виявили, що емпіричні значення є більшими за критичне значення.

10. Отриманий результат на рівні значущості $\alpha = 0{,}01$ дозволяє стверджувати, що і коефіцієнт кореляції Пірсона, і коефіцієнт рангової кореляції Спірмена є відмінними від нуля. А отже, між ознаками „кількість пропущених занять" та „підсумковий рейтинг" є статистично значимий сильний лінійний зв'язок із ймовірністю 99%.

12.2. ДИСПЕРСІЙНИЙ АНАЛІЗ ДАНИХ

12.2.1. Базові поняття дисперсійного аналізу даних

Дисперсійний аналіз (англ. *Analysis of Variance, ANOVA*) є сукупністю статистичних методів виявлення впливу на досліджувану результативну ознаку ***факторів*** – незалежних категоріальних ознак, кожна з яких може бути представлена у номінальній або порядковій шкалі.

Суть дисперсійного аналізу полягає в розчленуванні загальної дисперсії залежної змінної на окремі компоненти, зумовлені впливом конкретних факторів – незалежних змінних, та в перевірці гіпотез про значущість впливу цих факторів на досліджувану ознаку.

Якщо вплив факторів на результуючу ознаку перевіряють із використанням F-критерію Фішера, досліджувана ознака повинна бути розподіленою нормально – цей метод є параметричним. У випадку, коли розподіл результуючої ознаки відмінний від нормального, застосовують критерій Фрідмана.

За кількістю факторів – незалежних змінних, які впливають на досліджувану змінну, дисперсійний аналіз може бути однофакторним і багатофакторним.

Однофакторний дисперсійний аналіз досліджує вплив на результативну ознаку одного фактора. Об'єкти досліджуваного набору даних ділять на певну кількість груп, які відрізняються між собою ступенем дії фактора – його рівнем. Якщо між середніми значеннями кожної групи є статистично значуща відмінність, можна стверджувати про наявність впливу фактора на досліджувану ознаку.

Рівень фактора – це таке значення незалежної змінної, яке характеризує конкретний прояв фактора.

Перевірка статистичної значущості впливу фактора на досліджувану ознаку в дисперсійному аналізі базується на порівнянні компоненти дисперсії, обумовленої дією фактора, і компоненти дисперсії, обумовленої випадковими неврахованими чинниками.

Основні етапи однофакторного дисперсійного аналізу з використанням F-критерію Фішера є такими.

1. Формулюють нульову та альтернативну гіпотези:

H_0: фактор не впливає на досліджувану ознаку (відмінність між груповими середніми не є істотною);

H_1: фактор впливає на досліджувану ознаку (відмінність між груповими середніми є статистично значущою).

2. Формують таблицю із вхідними даними, розбивши значення досліджуваної змінної *y* за рівнями фактора, вплив якого досліджується – значенням незалежної змінної *x* (табл. 12.6). Значення досліджуваної змінної у кожній групі розглядають як вибірку нормально розподілених випадкових величин. Кількість об'єктів у групах із різним рівнем фактора може бути різною, а їх сума дорівнює кількості об'єктів у наборі даних.

3. Обчислюють середні значення досліджуваної змінної за формулами:

групові середні: $$\bar{y}_j = \frac{1}{n_j}\sum_{k=1}^{n_j} y_{jk}\,, \tag{12.8}$$

загальна середня: $$\bar{y} = \frac{1}{n}\sum_{j-1}^{m}\sum_{k=1}^{n_j} y_{jk}\,, \tag{12.9}$$

де y_{jk} – k-те значення досліджуванної змінної на j–му рівні фактора, m – кількість рівнів фактора,

n_j – кількість значень досліджуваної ознаки на на j–му рівні фактора,

$n = \sum_{k=1}^{m} n_k$ – кількість об'єктів у наборі даних.

Таблиця 12.6

Вхідні дані для проведення однофакторного дисперсійного аналізу

	Рівні фактора (значення незалежної змінної x)			
	x_1	x_2	…	x_m
Значення досліджуваної змінної y	y_{11}	y_{12}	…	y_{1m}
	y_{21}	y_{22}	…	y_{2m}
	…	…	…	…
	y_{n1}	y_{n2}	…	y_{nm}

4. Знаходять суми квадратів відхилень (англ. *Sum of Squares, SS*) від середніх:

міжгрупову: $$SS_{міжгр} = \sum_{k=1}^{m} n_k \cdot (\bar{y}_k - \bar{y})^2\,, \tag{12.10}$$

внутрішньогрупову: $$SS_{внутр} = \sum_{j=1}^{m}\sum_{k=1}^{n_j} (y_{jk} - \bar{y}_k)^2\,, \tag{12.11}$$

загальну: $$SS_{заг} = \sum_{j=1}^{m}\sum_{k=1}^{n_j} (y_{jk} - \bar{y})^2\,. \tag{12.12}$$

Для розрахованих значень справджується рівність:

$$SS_{заг} = SS_{внутр} + SS_{міжгр}\,.$$

Міжгрупова сума квадратів відхилень обумовлена впливом на досліджувану ознаку фактора, а внутрішньогрупа сума квадратів відхилень – впливом випадкових неврахованих чинників.

5. Розраховують статистичні оцінки компонент дисперсії із врахуванням числа ступенів свободи – середні сум квадратів відхилень (англ. *Mean Square, MS*):

міжгрупову: $$MS^2_{міжгр} = \frac{SS_{міжгр}}{m-1}\,, \tag{12.13}$$

внутрішньогрупову: $$MS^2_{внутр} = \frac{SS_{внутр}}{n-m}\,, \tag{12.14}$$

загальну: $$MS^2_{заг} = \frac{SS_{заг}}{n-1}. \quad (12.15)$$

6. Розраховують емпіричне значення F-критерію Фішера за формулою:

$$F_{емп} = \frac{MS^2_{міжгр}}{MS^2_{внутр}}. \quad (12.16)$$

7. Визначають критичне значення F-критерія Фішера $F_{кр}(\alpha, l_1, l_2)$ на рівні значущості α для ступенів свободи $l_1 = m-1$ та $l_2 = n-1$.

8. Порівнюють $F_{емп}$ та $F_{кр}$: якщо $F_{емп} \le F_{кр}$ – нульова гіпотеза про відсутність впливу фактору на досліджувану ознаку приймається з ймовірністю $p = 1-\alpha$. У протилежному випадку, якщо емпіричне значення критерія Фішера більше за критичне $F_{емп} > F_{кр}$, приймається альтернативна гіпотеза: фактор впливає на досліджувану ознаку із ймовірністю $p = 1-\alpha$.

При проведенні ***однофакторного дисперсійного аналізу*** в MS Excel та MatLab для визначення критичного значення критерія Фішера $F_{кр}$ ***використовують функції***:

MS Excel: $F\ РАСПОБР(\alpha; n_1; n_2) / FINV(\alpha; n_1; n_2)$, $F.ОБР.ПХ(\alpha; n_1; n_2) / F.INV.RT(\alpha; n_1; n_2)$,

MatLab: $finv(1-\alpha, n_1, n_2)$,

де α – рівень значущості, $n_1 = m-1$ та $n_2 = n-1$ – ступені свободи,

m – кількість рівнів фактору, n – кількість об'єктів у наборі даних.

Значно прискорює проведення дисперсійного аналізу з метою виявлення впливу фактора на досліджувану ознаку у середовищах MS Excel та MatLab використання вбудованих засобів:

1) однофакторного дисперсійного аналізу з пакету *Аналіз даних* MS Excel;
2) функції MatLab: *anova1()*.

Вбудовані засоби MS Excel та MatLab при проведенні дисперсійного аналізу для підтвердження достовірності отриманих результатів розраховують ***p-значення*** (англ. ***p-value***) – ймовірність випадкового підтвердження правильності альтернативної гіпотези. Це є ймовірність помилки – відхилення правильної нульової гіпотези.

Чим більшим є емпіричне значення критерія Фішера, тим менше відповідне йому p -значення. Якщо p -значення менше за рівень значущості α, то нульова гіпотеза відхиляється і приймається альтернативна гіпотеза з ймовірністю $1-\alpha$.

12.2.2. Приклад здійснення однофакторного дисперсійного аналізу даних в MS Excel

Приклад 3. Трьом групам студентів промовляли з різною швидкістю (низькою, середньою, високою) десять слів. Довести або спростувати припущення про те, що фактор швидкості пред'явлення слів істотно впливає на показники їх відтворення. Емпіричні дані наведено у таблиці 12.7.

Таблиця 12.7

Показники відтворення слів при різній швидкості їх пред'явлення

№ з/п		Значення ознаки «*Швидкість пред'явлення*» (рівні фактору – значення змінної x)		
		Низька	Середня	Висока
Значення ознаки «*Показник відтворення*» (значення змінної y)	1	7	5	5
	2	8	5	4
	3	7	6	5
	4	6	5	3
	5	7	4	4
	6	5	6	5
	7		4	4
	8			3

1. Формулюємо нульову та альтернативну гіпотези:

H_0: фактор «Швидкість пред'явлення» не впливає на досліджувану ознаку «Показник відтворення» (відмінності в обсязі відтворення слів не є більш вираженими, ніж випадкові відмінності всередині груп);

H_1: фактор «Швидкість пред'явлення» впливає на досліджувану ознаку «Показник відтворення» (відмінності в обсязі відтворення слів є більш вираженими, ніж випадкові відмінності всередині груп).

2. Формуємо таблицю MS Excel із вхідними даними та розраховуємо групові середні досліджуваної змінної y для кожного значення фактора x та загальну середню (рис. 12. 9):

1) групові середні:

$$x_1 = "низька": \quad n_1 = 6, \quad \bar{y}_1 = \frac{1}{n_1}\sum_{k=1}^{n_1} y_{1k} = \frac{7+8+7+9+5+7}{6} = 6{,}67;$$

$$x_2 = "середня": \quad n_2 = 7, \quad \bar{y}_2 = \frac{1}{n_2}\sum_{k=1}^{n_2} y_{2k} = \frac{5+5+6+5+4+6+4}{7} = 5;$$

$$x_3 = "висока": \quad n_3 = 8, \quad \bar{y}_3 = \frac{1}{n_3}\sum_{k=1}^{n_3} y_{3k} = \frac{5+4+5+3+4+5+4+3}{6} = 4{,}13.$$

2) загальна середня ($n = n_1 + n_2 + n_3 = 21$):

$$\bar{y} = \frac{1}{n}\sum_{j-1}^{3}\sum_{k=1}^{n_j} y_{jk} = \frac{1}{21}\left(\sum_{k=1}^{6} y_{1k} + \sum_{k=1}^{7} y_{2k} + \sum_{k=1}^{8} y_{3k}\right) = 5{,}143.$$

	A	B	C	D	E
1					
2		№	Швидкість пред'явлення		
3			Низька	Середня	Висока
4		1	7	5	5
5		2	8	5	4
6		3	7	6	5
7		4	6	5	3
8		5	7	4	4
9		6	5	6	5
10		7		4	4
11		8			3
12		n_i	6	7	8
13		Всього	40	35	33
14		Середні групові	6,67	5	4,13
15				Загальний обсяг	21
16				Загальне середнє	5,14

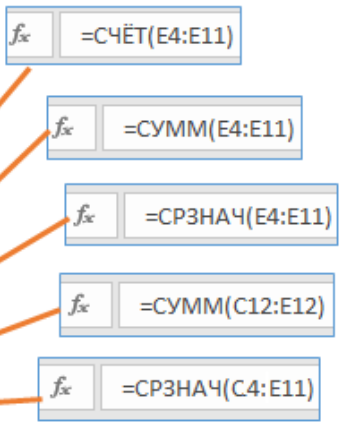


Рис. 12.9. Розрахунки середніх значень у MS Excel

3. Формуємо таблиці MS Excel для розрахунку квадратів різниць значень досліджуваної змінної y із груповими середніми для кожного значення фактора x та квадратів різниць значень досліджуваної змінної y із загальною середньою (рис. 12.10).

3.1. У таблиці *Квадрати різниць по групах* розраховуємо квадрати різниць вхідних значень та середніх у стовпцях – групових середніх:

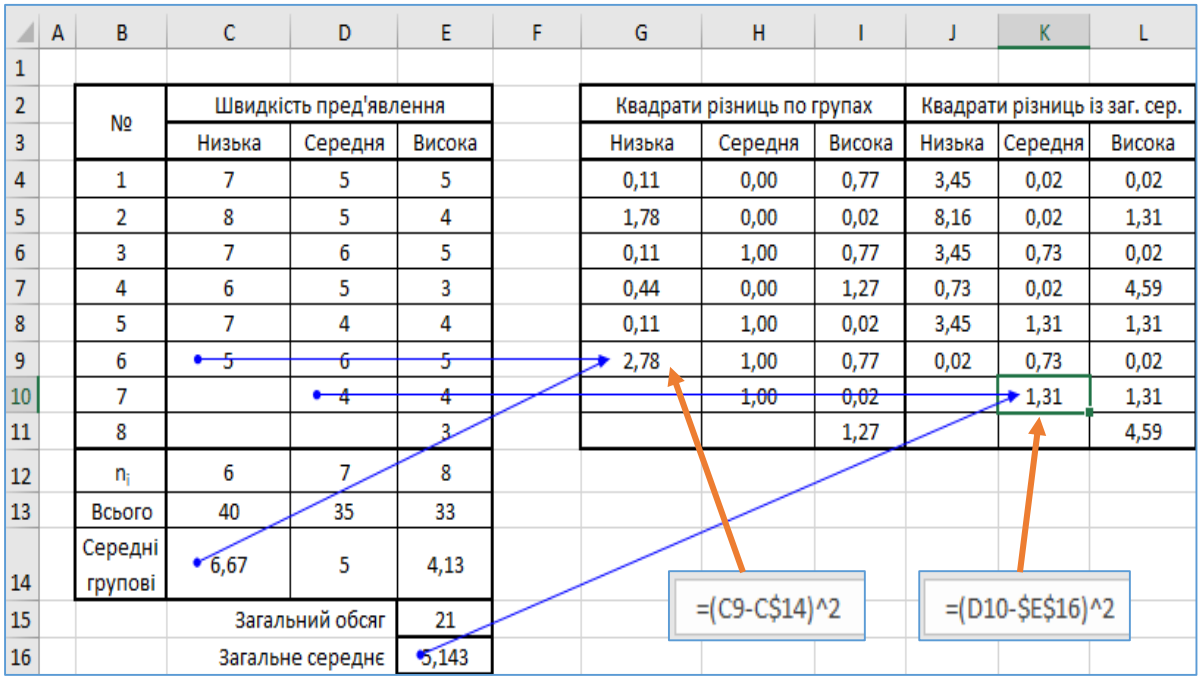
a) у комірці *G4* вводимо формулу *=(C4-C$14)^2* та поширюємо її на стовпець *G5:G9*;

b) у комірці *H4* вводимо формулу *=(D4-D$14)^2*, поширюємо її на стовпець *D5:D10*;

c) у комірці *I4* вводимо формулу *=(E4-E$14)^2*, поширюємо її на стовпець *I5:I11*.

3.2. У таблиці *Квадрати різниць із заг. сер.* розраховуємо квадрати різниць вхідних значень та загальних середніх:

a) у комірці *J4* вводимо формулу *=(C4-$E$16)^2* та поширюємо її на стовпець *J5:J9*;
b) у комірці *K4* вводимо формулу *=(D4-$E$16)^2*, поширюємо її на стовпець *K5:K10*;
c) у комірці *L4* вводимо формулу *=(E4-$E$16)^2*, поширюємо її на стовпець *L5:L11*.



№	Швидкість пред'явлення			Квадрати різниць по групах			Квадрати різниць із заг. сер.		
	Низька	Середня	Висока	Низька	Середня	Висока	Низька	Середня	Висока
1	7	5	5	0,11	0,00	0,77	3,45	0,02	0,02
2	8	5	4	1,78	0,00	0,02	8,16	0,02	1,31
3	7	6	5	0,11	1,00	0,77	3,45	0,73	0,02
4	6	5	3	0,44	0,00	1,27	0,73	0,02	4,59
5	7	4	4	0,11	1,00	0,02	3,45	1,31	1,31
6	5	6	5	2,78	1,00	0,77	0,02	0,73	0,02
7		4	4		1,00	0,02		1,31	1,31
8			3			1,27			4,59
n_i	6	7	8						
Всього	40	35	33						
Середні групові	6,67	5	4,13						
	Загальний обсяг		21						
	Загальне середнє		5,143						

Рис. 12.10. Розрахунки квадратів різниць у MS Excel

4. Формуємо таблиці для обчислення результатів дисперсійного аналізу (рис. 12.11).

4.1. Для розрахунку сум квадратів відхилень:

a) міжгрупової $SS_{міжгруп}$ (формула 12.10): у комірку *E19* вводимо формулу: *=C12*(C14-E16)^2+D12*(D14-E16)^2+E12*(E14-E16)^2*;
b) внутрішньогрупової $SS_{внутр}$ (формула 12.11): у комірку *E20* вводимо формулу *=СУММ(G4:G9;H4:H10;I4:I11)*;
c) загальної $SS_{заг}$ (формула 12.12): у комірку *E21* вводимо формулу *=СУММ(J4:J9;K4:K10;L4:L11)*.

4.2. Для розрахунку ступенів свободи:

a) міжгрупової: у комірку F19 вводимо формулу =3-1 (кількість рівнів фактора мінус 1);
b) внутрішньогрупової: у комірку F20 вводимо формулу =E15-3 (загальний обсяг вибірки мінус 3 рівні фактора);
c) загальної: у комірку F21 вводимо формулу =E15-1 (загальний обсяг вибірки мінус 1).

4.3. Для розрахунку статистичних оцінок дисперсій:

a) міжгрупової $MS_{міжгруп}$ (формула 12.13): у комірку *H19* вводимо формулу = *=E19/F19*;
b) внутрішньогрупової $MS_{внутр}$ (формула 12.14): у комірку *H20* вводимо формулу *=E20/F20*;
c) загальної $MS_{заг}$ (формула 12.15): у комірку *H21* вводимо формулу *=E21/F21*.

4.4. Для розрахунку значень F-критерію Фішера:

a) емпіричного критерію Фішера $F_{емп}$ (формула 12.16): у комірку *C24* вводимо формулу *=H19/H20*;
b) критичного значення критерію Фішера $F_{0.01}$ на рівні значущості 0,01 у комірку *C25* вводимо формулу *=F.ОБР.ПХ(0,01;F19;F20)*;

с) критичного значення критерію Фішера $F_{0.05}$ на рівні значущості 0,05 у комірку *C26* вводимо формулу *=F.ОБР.ПХ(0,05;F19;F20)*.

	A	B	C	D	E	F	G	H
17								
18		Вид дисперсії		Сума квадратів відхилень		Ступені свободи	Статистичні оцінки компонент дисперсії	
19		Міжгрупова		$SS_{міжгруп}$ =	22,3631	2	$MS_{міжгруп}$ =	11,1815
20		Внутрішньогрупова		$SS_{внутр}$ =	14,2083	18	$MS_{внутр}$ =	0,7894
21		Загальна		$SS_{заг}$ =	36,5714	20	$MS_{заг}$ =	1,8286
22								
23		Критерій Фішера						
24		$F_{емп}$ =	14,1655	=H19/H20				
25		$F_{0,01}$ =	6,0129	=F.ОБР.ПХ(0,01;F19;F20)				
26		$F_{0,05}$ =	3,5546	=F.ОБР.ПХ(0,05;F19;F20)				

Рис. 12.11. Розрахунки дисперсій та критерію Фішера в MS Excel

5. Порівнявши емпіричне значення критерію Фішера $F_{емп} = 14{,}3603$ із критичними: $F_{0.01} = 6{,}0129$, $F_{0.05} = 3{,}5546$, було виявлено, що емпіричне значення критерію Фішера є більшим за критичне на рівнях значущості 0,01 та 0,05.

6. Отримані результати дають підстави для відхилення нульової гіпотези та прийняття альтернативної гіпотези, відповідно до якої ми можемо стверджувати, що з ймовірністю 99% вплив фактора «Швидкість пред'явлення» слів на досліджувану ознаку «Показник відтворення» слів є статистично значущим.

Приклад 4. Із використанням пакета *Аналіз даних* MS Excel довести або спростувати припущення про те, що фактор швидкості пред'явлення слів істотно впливає на показники їх відтворення за емпіричними даними, наведеними у таблиці 12.7.

1. Для застосування однофакторного дисперсійного аналізу з пакета *Аналіз даних* MS Excel на вкладці *Дані/Data* у групі Аналіз/*Analysis* необхідно обрати *Аналіз даних/ Data Analysis* та у вікні вибору інструменту аналізу, яке з'явиться, – *Однофакторний дисперсійний аналіз* (рис. 12.12).

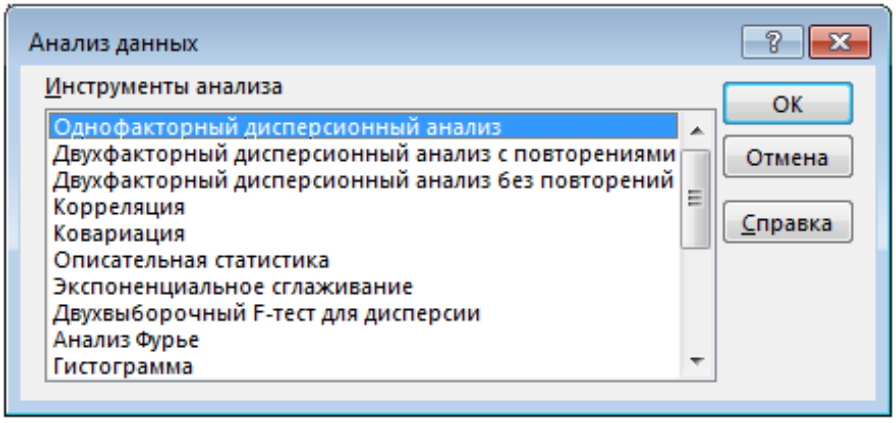


Рис. 12.12. Вікно Аналіз даних / Data Analysis

2. Відкриється вікно *Однофакторий дисперсійний аналіз*, у якому необхідно вказати вхідний інтервал із даними, спосіб групування (у стовпцях чи рядках), рівень значущості та вихідний інтервал (рис. 12.13). Після налаштувань параметрів дисперсійного аналізу натискають кнопку *ОК*.

3. У вказаному при налаштуванні вихідному діапазоні буде виведено результат здійсненого однофакторного дисперсійного аналізу (рис. 12.14).

Тут *SS* – суми квадратів відхилень, *df* – число ступенів свободи;

MS – середнє значення міжгрупової та внутрішньогрупової сум квадратів відхилень;

F – розраховане за емпіричними даними значення критерію Фішера;

p-Значение – визначене для розрахованого *F*;

F критическое – критичне значення критерію Фішера на рівні значущості $\alpha = 0{,}05$.

4. Як бачимо, отримані за допомогою пакета *Аналіз даних* результати співпадають із отриманими у попередньому прикладі: *p-Значение*, рівне 0,0002017, є меншим за рівень значущості $\alpha = 0{,}05$, а розраховане емпіричне значення критерію Фішера 14,166 більше за критичне 3,554, визначене на рівні значущості $\alpha = 0{,}05$. Тому нульова гіпотеза про відсутність впливу фактора швидкості пред'явлення слів на показник відтворення слів відхиляється. Приймається альтернативна гіпотеза – фактор впливає на результативну ознаку з ймовірністю 95%.

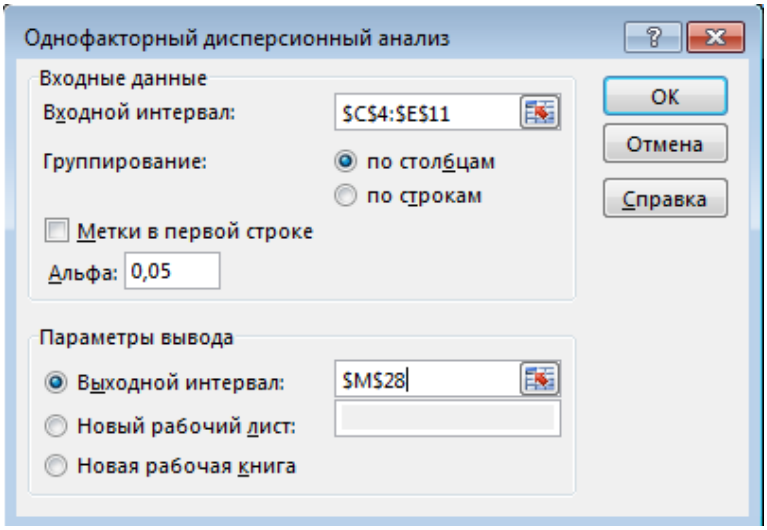


Рис. 12.13. Вікно Однофакторний дисперсійний аналіз

Однофакторный дисперсионный анализ

ИТОГИ

Группы	*Счет*	*Сумма*	*Среднее*	*Дисперсия*
Столбец 1	6	40	6,6667	1,0667
Столбец 2	7	35	5,00	0,6667
Столбец 3	8	33	4,1250	0,6964

Дисперсионный анализ

Источник вариации	*SS*	*df*	*MS*	*F*	*P-Значение*	*F критическое*
Между группами	22,3631	2	11,1815	14,1655	0,0002017	3,55456
Внутри групп	14,2083	18	0,7894			
Итого	36,5714	20				

Рис. 12.14. Виведення результатів дисперсійного аналізу, проведеного з використанням Пакета Аналізу MS Excel

12.2.3. Приклад проведення однофакторного дисперсійного аналізу даних засобами MatLab

Приклад 5. Здійснити однофакторний дисперсійний аналіз для виявлення впливу фактора А на досліджувану ознаку в середовищі MatLab. Вибірки з емпіричними значеннями досліджуваного показника для різних рівнів фактора А наведено у таблиці 12.8.

Таблиця 12.8

Емпіричні дані зі значеннями досліджуваної ознаки на рівнях фактора А

№ з/п	Рівні фактора А			
	А1	А2	А3	А4
1	9	9	7	8
2	8	10	14	16
3	10	10	9	8
4	12	10	16	7

1. Формулюємо нульову та альтернативну гіпотези:

H_0: фактор А не впливає на досліджувану ознаку (між середніми значеннями досліджуваної ознаки на різних рівнях фактора А немає суттєвої відмінності);

H_1: фактор А впливає на досліджувану ознаку (відмінності між середніми значеннями досліджуваної ознаки на різних рівнях фактора А є статистично значущими).

2. У вікні Command Window вводимо команду для створення матриці А, яка буде містити дані зі значеннями досліджуваної ознаки (рис. 12.15):

```
% задаємо матрицю А – вхідний набір даних
>>A=[9 9 7 8; 8 10 14 16; 10 10 9 8; 12 10 16 7];
```

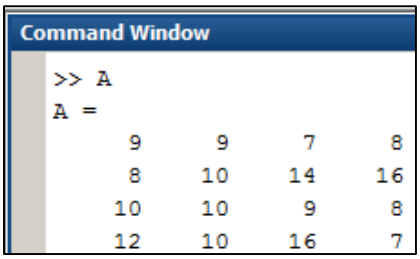


Рис. 12.15. Виведення у вікні Command Window вхідних даних

3. Для здійснення однофакторного дисперсійного аналізу в MatLab застосуємо функцію *anova1()*:

```
% здійснення однофакторного дисперсійного аналізу
>> p=anova1(A)
```

2. Результати здійсненого аналізу відображаються у двох графічних вікнах, які містять:

a) діаграму розкиду групових середніх досліджуваної ознаки на різних рівнях фактора А (рис. 12.16);

b) таблицю з результатами однофакторного дисперсійного аналізу (рис. 12.17).

4. Зробимо аналіз діаграми розмаху середніх за заданими значеннями досліджуваної ознаки на різних рівнях фактора А (рис. 12.16). Візуально різниця між груповими середніми не є дуже великою. Це дозволяє висловити припущення про відсутність суттєвого впливу фактора на досліджувану ознаку.

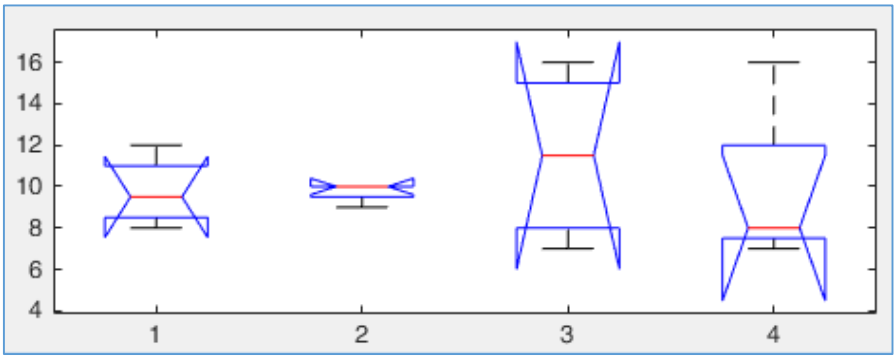


Рис. 12.16. Діаграма розкиду групових середніх досліджуваної ознаки на різних рівнях фактору А

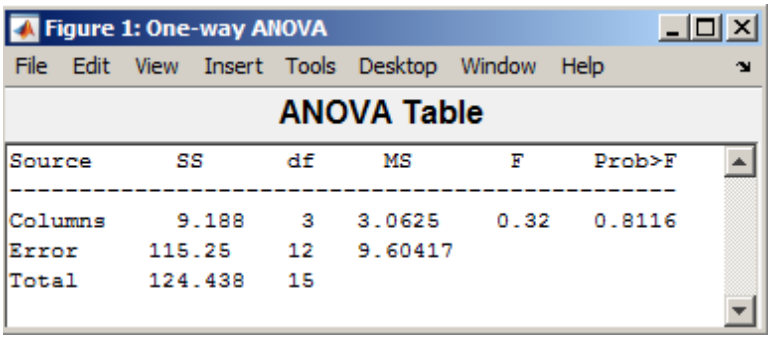


Рис. 12.17. Таблиця з результатами однофакторного дисперсійного аналізу

3. Розглянемо структуру таблиці з результатами однофакторного дисперсійного аналізу (рис. 12.17). Вона містить:

1) вид дисперсії (*Source*): внутрішньогрупова (*Columns*), міжгрупова (*Error*), загальна (*Total*);

2) суми квадратів відхилень від середніх (*SS*) для кожного виду дисперсії;

3) кількість ступенів свободи для кожного виду дисперсії *df*;

4) середнє значення сум квадратів різниць *MS* для міжгрупової та внутрішньогрупової дисперсій;

5) емпіричне значення критерію Фішера $F = 0{,}32$, розраховане за визначеними *MS*;

6) *p-значення* (*Prob>F*): ймовірність відхилення правильної нульової гіпотези, рівна 0,8116.

5. Вихідним аргументом функції *anova1()*, який виводиться у вікні *Command Window*, є *p*-значення (рис. 12.18).

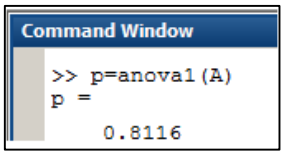


Рис. 12.18. Виведення *p*-значення у вікні Command Window

6. Функція *anova1()* дає можливість отримати більшу кількість вихідних даних, які можна буде використовувати у подальших розрахунках (рис. 12.19):

```
% tbl: масив зі значеннями стандартних для дисперсійного аналізу величин
% stats: структура, з даними для попарного порівняння групових середніх
>> [p,tbl,stats]=anova1(A)
```

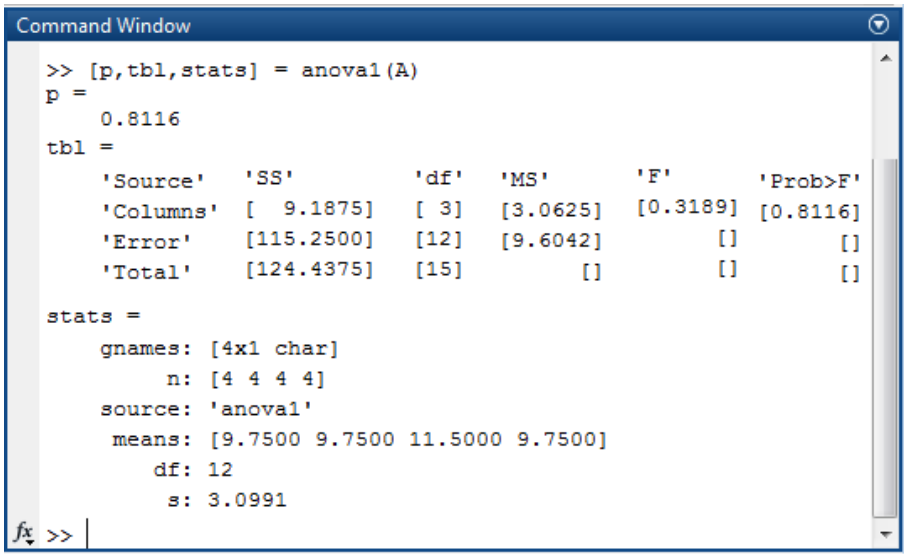


Рис. 12.19. Виведення у вікні Command Window результату

7. Зробимо аналіз отриманих результатів. Ми отримали $p = 0{,}8116$ і $p > 0{,}05$, значить нульова гіпотеза приймається: між середніми значеннями досліджуваної ознаки на різних рівнях фактора А немає суттєвої відмінності. Отже, фактор А не впливає на досліджувану ознаку.

12.3. ЗАВДАННЯ ДЛЯ САМОСТІЙНОЇ РОБОТИ

Завдання 1. З'ясувати наявність зв'яку між ознаками, представленими у таблиці спряженості, побудованій на основі самостійно підібраного набору даних із категоріальними ознаками, що характеризують об'єкти певної предметної області.

Завдання 2. Для даних, які характеризують об'єкти набору даних за двома ознаками (табл. 12.9), дослідити зв'язок між ознаками, визначивши: 1) коефіцієнт кореляції Пірсона; 2) коефіцієнт рангової кореляції Спірмена; 3) статистичну значущість розрахованих коефіцієнтів кореляції.

Завдання 3. Для вхідних даних (табл. 12.10), які є вибірками, що містять швидкість розв'язування задач при різних методиках навчання, перевірити гіпотезу про те, чи існує вплив методики навчання (фактор *A*) на швидкість розв'язування задач із використанням однофакторного дисперсійного аналізу за допомогою: 1) засобів MS Excel; 2) засобів MatLab.

КОНТРОЛЬНІ ПИТАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ № 12

1. Чим відрізняються параметричні та непараметричні статистичні методи виявлення зв'язків і закономірностей між змінними?
2. У чому полягає відмінність між функціональною та стохастичною залежностями між змінними?
3. Як здійснюється виявлення наявності зв'язків між ознаками з використанням таблиць спряженості?
4. Що таке критерій згоди Пірсона? Якою є формула для його визначення.
5. Кореляційний аналіз даних, його сутність.
6. Якими є формули для розрахунку емпіричних значень коефіцієнтів кореляції Пірсона та Спірмена? Як визначають їх значущість?
7. Якими є емпіричні правила дослідження кореляції між змінними за допомогою діаграм розсіювання?
8. Дисперсійний аналіз даних, його сутність. Критерій Фішера.
9. Опишіть основні етапи однофакторного дисперсійного аналізу.
10. Основні підходи до проведення однофакторного дисперсійного аналізу даних у MS Excel та MatLab.

Таблиця 12.9

Набори даних зі значеннями ознак до завдання 2

Варіант	1		2		3		4		5		6	
№ з/пр	X_1	X_2	X_1	X_2	X_1	X_2	X_1	X_2	X_1	X_2	X_1	X_2
1	54	58	7,5	3	4,1	-4,8	5,3	1,7	-3	9,2	5	4
2	23	36	10	2	4,9	-1,7	5,9	4,6	-1	11,4	7,5	3
3	90	100	12,5	1	5,7	1,4	6,5	7,5	1	13,6	10	2
4	79	85	15	0	6,5	4,5	7,1	10,4	3	15,8	12,5	1
5	58	67	17,5	-1	7,3	7,6	7,7	13,3	5	18	15	0
6	83	95	5	-2	2,5	10,7	3,5	16,2	-5	20,2	5	-1
7	67	61	22,5	-3	8,9	13,8	8,9	19,1	9	22,4	20	-2
8	54	58	25	-4	9,7	16,9	9,5	22	11	24,6	22,5	-3
9	28	36	27,5	-5	10,5	20	10,1	24,9	13	26,8	25	-4
10	88	108	30	-6	11,3	23,1	10,7	27,8	15	29	27,5	-5
11	79	85	32,5	-7	12,1	26,2	11,3	30,7	17	31,2	30	-6
12	63	65	35	-8	12,9	29,3	11,9	33,6	19	33,4	32,5	-7
13	81	92	37,5	-9	13,7	32,4	12,5	36,5	21	35,6	35	-8
14	62	61	40	-10	14,5	35,5	13,1	39,4	23	37,8	37,5	-9
15	47	38	42,5	-11	15,3	38,6	13,7	42,3	25	40	40	-10

Варіант	7		8		9		10		11		12	
№ з/пр	x_1	x_2	x_1	x_2	x_1	x_2	x_1	x_2	x_1	x_2	x_1	x_2
1	2	14,4	7,5	-2,8	-9	11,6	7,5	3	3,5	4,2	9,5	-0,7
2	1,5	17,8	9,5	-0,7	-11	13,8	10	2	3	7,6	11,5	1,4
3	1	21,2	11,5	1,4	-13	16	12,5	1	2,5	11	13,5	3,5
4	0,5	24,6	13,5	3,5	-15	18,2	15	0	2	14,4	15,5	5,6
5	0	28	15,5	5,6	-17	20,4	17,5	-1	1,5	17,8	17,5	7,7
6	2,5	31,4	3,5	7,7	-3	22,6	5	-2	2,5	21,2	3,5	9,8
7	-1	34,8	19,5	9,8	-21	24,8	22,5	-3	0,5	24,6	21,5	11,9
8	-1,5	38,2	21,5	11,9	-23	27	25	-4	0	28	23,5	14
9	-2	41,6	23,5	14	-25	29,2	27,5	-5	-0,5	31,4	25,5	16,1
10	-2,5	45	25,5	16,1	-27	31,4	30	-6	-1	34,8	27,5	18,2
11	-3	48,4	27,5	18,2	-29	33,6	32,5	-7	-1,5	38,2	29,5	20,3
12	-3,5	51,8	29,5	20,3	-31	35,8	35	-8	-2	41,6	31,5	22,4
13	-4	55,2	31,5	22,4	-33	38	37,5	-9	-2,5	45	33,5	24,5
14	-4,5	58,6	33,5	24,5	-35	40,2	40	-10	-3	48,4	35,5	26,6
15	-5	62	35,5	26,6	-37	42,4	42,5	-11	-3,5	51,8	37,5	28,7
Варіант	13		14		15		16		17		18	
№ з/пр	x_1	x_2	x_1	x_2	x_1	x_2	x_1	x_2	x_1	x_2	x_1	x_2
1	3	-1,6	1,8	8	4,1	-17,2	7,7	5,7	4	8	5	7
2	1	0,6	6,6	10	3,3	-14,1	9,1	9,6	11	12	0	8
3	-1	2,8	11,4	12	2,5	-11	10,5	13,5	18	16	-5	9
4	-3	5	16,2	14	1,7	-7,9	11,9	17,4	25	20	-10	10
5	-5	7,2	21	16	0,9	-4,8	13,3	21,3	32	24	-15	11
6	-3	9,4	-3	18	2,5	-1,7	3,5	25,2	-3	28	5	12
7	-9	11,6	30,6	20	-0,7	1,4	16,1	29,1	46	32	-25	13
8	-11	13,8	35,4	22	-1,5	4,5	17,5	33	53	36	-30	14
9	-13	16	40,2	24	-2,3	7,6	18,9	36,9	60	40	-35	15
10	-15	18,2	45	26	-3,1	10,7	20,3	40,8	67	44	-40	16
11	-17	20,4	49,8	28	-3,9	13,8	21,7	44,7	74	48	-45	17
12	-19	22,6	54,6	30	-4,7	16,9	23,1	48,6	81	52	-50	18
13	-21	24,8	59,4	32	-5,5	20	24,5	52,5	88	56	-55	19
14	-23	27	64,2	34	-6,3	23,1	25,9	56,4	95	60	-60	20
15	-25	29,2	69	36	-7,1	26,2	27,3	60,3	102	64	-65	21
Варіант	19		20		21		22		23		24	
№ з/пр	x_1	x_2	x_1	x_2	x_1	x_2	x_1	x_2	x_1	x_2	x_1	x_2
1	-2	10,9	5,5	1,4	-8	13,6	5	1,5	5	-2,8	-0,5	-2,8
2	-6,5	13,8	6,5	5,6	-12	15,8	7	-0,5	5,5	0,6	-2,5	-0,7
3	-11	16,7	7,5	9,8	-16	18	9	-2,5	6	4	-4,5	1,4
4	-15,5	19,6	8,5	14	-20	20,2	11	-4,5	6,5	7,4	-6,5	3,5
5	-20	22,5	9,5	18,2	-24	22,4	13	-6,5	7	10,8	-8,5	5,6
6	2,5	25,4	3,5	22,4	4	24,6	3	-8,5	6	14,2	3,5	7,7
7	-29	28,3	11,5	26,6	-32	26,8	17	-10,5	8	17,6	-12,5	9,8
8	-33,5	31,2	12,5	30,8	-36	29	19	-12,5	8,5	21	-14,5	11,9
9	-38	34,1	13,5	35	-40	31,2	21	-14,5	9	24,4	-16,5	14
10	-42,5	37	14,5	39,2	-44	33,4	23	-16,5	9,5	27,8	-18,5	16,1
11	-47	39,9	15,5	43,4	-48	35,6	25	-18,5	10	31,2	-20,5	18,2
12	-51,5	42,8	16,5	47,6	-52	37,8	27	-20,5	10,5	34,6	-22,5	20,3
13	-56	45,7	17,5	51,8	-56	40	29	-22,5	11	38	-24,5	22,4
14	-60,5	48,6	18,5	56	-60	42,2	31	-24,5	11,5	41,4	-26,5	24,5
15	-65	51,5	19,5	60,2	-64	44,4	33	-26,5	12	44,8	-28,5	26,6

Таблиця 12.10

Значення досліджуваної ознаки до завдання 3

№ варіанта		Рівень фактора А				№ варіанта		Рівень фактора А			
		A_1	A_2	A_3	A_4			A_1	A_2	A_3	A_4
1	Значення	9	10	8	9	2	Значення	60	75	60	95
		8	12	10	12			80	66	80	85
		8	16	10	18			75	85	65	100
		9	18	10	8			85	80	60	80
3	Значення	25	29	19	18	4	Значення	6	5	5	18
		28	22	25	30			8	4	4	16
		20	21	30	24			3	10	13	21
		22	18	22	20			2	11	12	20
5	Значення	28	24	23	29	6	Значення	90	80	90	75
		26	28	24	28			85	110	75	120
		21	32	20	27			105	115	120	110
		25	28	24	28			110	90	110	85
7	Значення	10	8	15	12	8	Значення	1	1	3	2
		7	14	12	13			3	2	4	4
		8	6	11	6			2	5	6	4
		6	10	9	8			4	3	4	5
9	Значення	2,2	3,1	2,4	3,3	10	Значення	1,9	2,1	3	3,1
		2,6	3,3	2,6	2,7			1,5	2,2	3,1	3,3
		2,8	3,5	2,8	2,5			1,7	2,5	2,9	2,9
		3,1	3,7	2,6	3,9			1,9	1,8	2,8	2,8
11	Значення	10	9	11	8	12	Значення	22	27	31	29
		12	7	9	8			24	24	30	29
		15	8	8	9			26	26	29	25
		11	6	10	11			25	23	27	30
13	Значення	0,9	1,5	2,2	1,9	14	Значення	12	18	19	21
		0,8	1,7	1,9	1,7			15	19	17	23
		0,5	1,2	2,1	1,4			14	21	22	22
		1,1	1,6	2,2	1,6			11	23	19	23
15	Значення	5,5	7,4	6,3	7,9	16	Значення	10	11,6	7,6	7,2
		5,7	7,6	6,9	8,1			11,2	8,8	10	12
		6,1	7,2	6,5	7,8			8	8,4	12	9,6
		6,5	7,1	6,6	8,1			8,8	7,2	8,8	8
17	Значення	7,2	8	6,4	7,2	18	Значення	25,2	21,6	20,7	26,1
		6,4	9,6	12,8	14,4			23,4	25,2	21,6	25,2
		8	8,8	8	8			18,9	28,8	18	24,3
		9,6	8,8	14,4	6,4			22,5	25,2	21,6	25,2
19	Значення	2,2	3,1	2,4	3,3	20	Значення	6	5	5	18
		2,6	3,3	2,6	2,7			8	4	4	16
		2,8	3,5	2,8	2,5			3	10	13	21
		3,1	3,7	2,6	3,9			2	11	12	20
21	Значення	48	60	48	76	22	Значення	81	72	81	67,5
		64	52,8	64	68			76,5	99	67,5	108
		60	68	52	80			94,5	104	108	99
		68	64	48	64			99	81	99	76,5
23	Значення	2,4	2	2	7,2	24	Значення	1,1	1,1	3,3	2,2
		3,2	1,6	1,6	6,4			3,3	2,2	4,4	4,4
		1,2	4	5,2	8,4			2,2	5,5	6,6	4,4
		0,8	4,4	4,8	8			4,4	3,3	4,4	5,5
25	Значення	2,7	2,9	4,2	4,3	26	Значення	84	105	84	133
		2,1	3,1	4,3	4,6			112	92,4	112	119
		2,4	3,5	4,1	4,1			105	119	91	140
		2,7	2,5	3,9	3,9			119	112	84	112

13. РЕГРЕСІЙНИЙ АНАЛІЗ ДАНИХ. ЛІНІЙНА РЕГРЕСІЯ

Лабораторна робота № 13

Мета: закріплення знань про сутність та етапи проведення регресійного аналізу даних. Набуття навичок здійснення регресійного аналізу в MS Excel, SPSS.

Теоретичні знання: базові поняття регресійного аналізу даних. Методи та етапи побудови регресійної моделі. Лінійна регресійна залежність, проста лінійна регресія. Оцінка загальної якості регресійної моделі: перевірка правильності, адекватності моделі, аналіз статистичної значущості її параметрів та тісноти зв'язку змінних.

13.1. РЕГРЕСІЙНИЙ АНАЛІЗ ДАНИХ

13.1.1. Основні поняття регресійного аналізу

Регресійний аналіз (англ. *Regression analysis*) є сукупністю методів, які призначені для перевірки наявності й встановлення типу зв'язку між залежною змінною (відгуком) та незалежними змінними (факторами) у наборі даних, який містить характеристики об'єктів досліджуваної предметної області.

Основною метою регресійного аналізу даних є теоретичне обґрунтування точкових та інтервальних оцінок параметрів регресійної моделі – рівняння регресійної залежності змінних, а також точковий і інтервальний прогноз значень залежної змінної.

Регресійний аналіз полягає у ***підборі апроксимуючої функції***, яка найкращим чином описує наявну сукупність емпіричних даних:

$$y = f(x_1, x_2, ..., x_m) + \varepsilon, \tag{13.1}$$

де y – залежна змінна ($y \in R$), x_i – незалежні змінні ($x_i \in R, i = \overline{1, m}$),

ε – похибка, обумовлена впливом випадкових чинників, m – кількість незалежних змінних.

При побудові регресійної моделі припускають, що похибка ε у рівнянні регресії є випадковою нормально розподіленою величиною з математичним сподіванням, рівним нулю та сталою дисперсією.

Апроксимація дозволяє установити стохастичну залежність між змінними, які відповідають характеристикам об'єктів набору даних, у вигляді функції шляхом спрощення та згладжування відомих показників. Отримана оцінка функціональної залежності між відгуком та факторами називається ***рівнянням регресії***, або просто ***регресією***.

Побудована ***регресійна модель*** в інтелектуальному аналізі даних використовується для прогнозування значень залежної змінної та виявлення й встановлення типу зв'язку між змінними. Однак виявлений зв'язок між змінними не завжди є причинно-наслідковим. Щоб використовувати регресійну модель для опису причинно-наслідкових зв'язків, аналітик повинен обґрунтувати інтерпретацію таких зв'язків між змінними з опорою на використання при побудові моделі репрезентативно відібраного набору даних значного обсягу в аналізованій предметній області.

Задача регресії, як і задача класифікації, передбачає визначення значень однієї залежної змінної на підставі значень інших незалежних змінних. Однак у задачі класифікації залежна змінна має категоріальний тип і множина її значень є скінченою, на відміну від регресії, де значення незалежних і залежної змінної є дійсними числами.

Залежно від ***кількості незалежних змінних*** розрізняють:

1) ***просту, парну, однофакторну регресію***, яка виявляє залежність залежної змінної від однієї ***незалежної*** змінної;
2) ***множинну, багатофакторну регресію***, яка виявляє залежність залежної змінної від декількох незалежних змінних.

Однофакторна регресійна модель придатна для короткострокових прогнозів. Визначення параметрів множинної (багатофакторною) регресії вимагає трудомістких розрахунків із застосуванням комп'ютерних програмних засобів. Однак отримані результати будуть достовірними та можуть використовуватися для складання довгострокових прогнозів.

Здійснюване на основі регресійного аналізу прогнозування забезпечує, як правило, кращі результати при інтерполяції, ніж при екстраполяції. Тобто визначення значень залежної змінної всередині інтервалів змін емпіричних значень незалежних факторних ознак дає можливість отримати більш достовірні результати порівняно з визначенням значень залежної змінної поза межами цих інтервалів.

Залежно від ***аналітичного математичного виразу***, який описує зв'язок між змінними у рівнянні регресії, виділяють такі види регресії:

1) ***лінійна регресія*** – є простою або множинною залежністю залежної змінної від незалежних змінних, яка виражається лінійною функцією;
2) ***нелінійна регресія*** – є нелінійною залежністю залежної змінної від незалежних змінних (поліноміальною, гіперболічною, степеневою, показниковою, експоненціальною, логарифмічною, логістичною).

Регресійний аналіз даних застосовується для вивчення взаємозв'язків між ознаками об'єктів набору даних у динаміці й дозволяє установити, які незалежні змінні істотно впливають на досліджувану величину – залежну змінну. Це дає можливість аналітику визначити, які ознаки є суттєвими у подальшому аналізі складних систем та процесів із різних сфер навколишньої дійсності, а які можна не враховувати при побудові моделі спостережуваних процесів та явищ.

13.1.2. Етапи та методи регресійного аналізу даних

Послідовність ***етапів регресійного аналізу*** є наступною:

1. ***Постановка задачі*** – полягає у формуванні гіпотез про взаємозв'язки явищ та процесів на основі здійсненого аналізу досліджуваної предметної області.

2. ***Визначення змінних*** – залежної та незалежних, на основі висунутих припущень про фактори, які суттєво впливають на досліджуваний показник. На цьому етапі незалежні змінні потрібно перевірити на ***відсутність мультиколінеарності*** – наявності сильних лінійних взаємозв'язків між ними. Коефіцієнт кореляції між кожною парою незалежних змінних повинен задовольняти нерівність $|r| < 0{,}7$. У випадку виявлення сильних взаємозв'язків між парою незалежних змінних одна зі змінних є зайвою, її необхідно вилучити з подальшого аналізу для уникнення хибних кореляцій.

3. ***Формування набору даних*** зі значеннями за кожною із визначених змінних.

4. ***Вибір регресійної моделі***: формулювання гіпотези про форму зв'язку залежної та незалежних змінних (проста або множинна, лінійна або нелінійна) та вид апроксимуючої функції.

5. ***Визначення функції регресії***: оцінка за наявними емпіричними даними параметрів обраної моделі – розрахунок чисельних значень параметрів рівняння регресії. Оцінка параметрів здійснюється з використанням різних методів: методу найменших квадратів (найпоширеніший), нейронних мереж та інших.

6. ***Оцінка якості регресійної моделі***: перевірка правильності, адекватності моделі, аналіз статистичної значущості її параметрів та тісноти зв'язку змінних. У випадку, якщо якість побудованої моделі не є задовільною, здійснюють побудову іншої моделі. На практиці, як правило, здійснюється побудова декількох моделей, із яких обирається та, що найбільш адекватно описує емпіричні дані.

7. ***Інтерпретація отриманих результатів***: отримані результати регресійного аналізу порівнюються з висунутими гіпотезами, оцінюється їх коректність і правдоподібність.

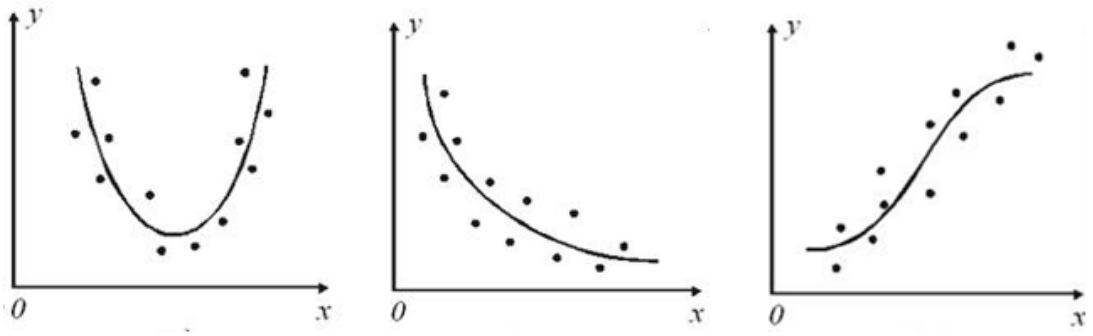
8. ***Застосування побудованої моделі*** у досліджуваній предметній області для управління та прогнозування – передбачення невідомих значень залежної змінної.

На етапах проведення регресійного аналізу аналітик використовує сукупність методів, серед яких розрізняють:

1) ***методи побудови математичних моделей*** досліджуваних систем;
2) ***методи визначення параметрів*** цих моделей;
3) ***методи оцінки якості*** побудованої регресійної моделі.

При побудові регресійної моделі для визначення впливу незалежних змінних на результативний показник – залежну змінну – необхідно підібрати та обґрунтувати рівняння зв'язку між змінними, яке відповідає ***характеру аналітичної стохастичної залежності*** між досліджуваними ознаками. З цією метою використовують:

1) ***графічний метод***: дозволяє висунути припущення про характер зв'язку та вид апроксимуючої функції на основі візуалізації емпіричного набору даних (рис. 13.1);
2) ***порівняння рядів значень змінних набору даних***: дозволяє спостерігати за рівномірністю взаємних змін – якщо зміна факторної ознаки призводить до рівномірної зміни результативної ознаки, то використовують лінійну функцію;
3) ***табличний метод***: спостереження за змінами даних, представлених у таблиці.



а) параболічна залежність б) гіперболічна залежність в) поліноміальна залежність

Рис. 13.1. Використання графічного методу при оцінці характеру аналітичної залежності між двома змінними

Графічний метод дає найбільш наочну картину залежності між змінними. Однак визначення типу моделі за графіком емпіричних даних не є достаньо обґрунтованим. Зазвичай доводиться перевіряти та оцінювати декілька варіантів моделі і обирати більш якісну та адекватну. Зіставлення варіантів різних моделей потребує великих обсягів обчислень, реалізацію яких спрощує використання інтегрованих комп'ютерних пакетів, що підтримують методи регресійного аналізу даних.

Багато залежностей, які описують реальні процеси, є нелінійними, а їх моделювання є досить складною задачею. Випадок лінійної регресійної залежності між змінними є більш дослідженим та простішим для аналізу. На практиці, якщо регресійна модель може бути представлена у вигляді нелінійної функції, що зводиться до лінійної шляхом перетворення змінних (додаток Р), доцільно здійснити ***лінеаризацію моделі***. Це дозволяє знизити розрахункову та аналітичну складність поставленої задачі. При неможливості лінеаризації моделі досліджують нелінійну регресійну залежність, яка вимагає застосування чисельних методів, що реалізують один із методів пошуку екстремуму функції багатьох змінних.

13.1.3. Лінійна регресійна залежність. Проста лінійна регресія

Рівняння ***множинної лінійної регресії*** має вигляд:

$$y = a_0 + a_1 x_1 + ... + a_m x_m + \varepsilon, \tag{13.2}$$

де y – залежна, результативна змінна, x_i – незалежні, факторні змінні ($i = \overline{1, m}$),

a_i – коефіцієнти регресії, a_0 – вільний член рівняння,

ε – похибка, обумовлена впливом випадкових чинників та неврахованих факторів (залишки),

m – кількість незалежних змінних.

Невідомі параметри рівняння лінійної регресії: коефіцієнти a_i та a_0, знаходять за ***методом найменших квадратів***, який дозволяє отримати такі оцінки параметрів, при яких ***залишкова сума квадратів*** (англ. *Residual Sum of Squares, RSS*) – сума квадратів відхилень SS_ε фактичних значень результативної ознаки y_i від розрахованих за регресійною моделлю теоретичних $\hat{y}_i$ є мінімальною (рис. 13.2):

$$SS_\varepsilon = \sum \varepsilon_i^2 = \sum (y_i - \hat{y}_i)^2 \to \min. \tag{13.3}$$

Для знаходження мінімуму функції SS_ε необхідно знайти часткові похідні за кожним із параметрів моделі та прирівняти їх до нуля.

Лінійну множинну регресійну модель зручно представити у матричному вигляді:

$$Y = X \cdot A + E, \tag{13.4}$$

де Y – вектор-стовпець значень залежної змінної, X – матриця значень незалежних змінних,

A – вектор-стовпець коефіцієнтів регресії, E – вектор-стовпець значень випадкової складової.

У матричному вигляді умова знаходження параметрів регресії має вигляд:

$$SS_\varepsilon = \sum \varepsilon_i^2 = E^T E = (Y - XA)^T (Y - XA) \to \min, \tag{13.5}$$

Якщо існує обернена матриця $(X^T X)^{-1}$, формула для розрахунку параметрів лінійної регресії у матричному вигляді є наступною:

$$A=(X^T X)^{-1} X^T Y. \tag{13.6}$$

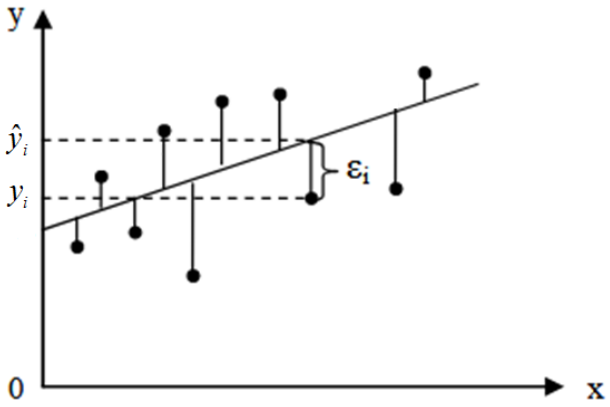


Рис. 13.2. Відхилення фактичних значень результативної змінної від теоретичних, розрахованих за лінійною моделлю

Для однозначного визначення параметрів регресії необхідно, щоб виконувалася рівність:

$$n \geq m+1, \tag{13.7}$$

де n – кількість об'єктів у наборі даних, m – кількість незалежних змінних.

Оцінка параметрів множинної багатофакторної лінійної регресії вимагає трудомістких розрахунків із застосуванням комп'ютерних програмних засобів.

Найпростішим є рівняння ***простої лінійної регресії*** – рівняння прямої виду:

$$y = a_0 + a_1 x + \varepsilon, \tag{13.8}$$

де y – залежна змінна, x – незалежна змінна, a_0 – вільний член рівняння, a_1 – коефіцієнт регресії, ε – похибка, обумовлена впливом випадкових чинників та неврахованих факторів.

Знаходження значень параметрів простої лінійної регресії з використанням методу найменших квадратів зводиться до розв'язання системи лінійних рівнянь:

$$\begin{cases} a_0 \sum_{i=1}^{n} x_i + a_1 \sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} x_i y_i \\ a_0 n + a_1 \sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i \end{cases}, \tag{13.9}$$

де y_i – емпіричні значення залежної змінної, x_i – емпіричні значення незалежної змінної, n – кількість об'єктів набору даних.

Систему рівнянь 13.9 ров'язують методом послідовного виключення змінних або методом визначників. Використання програмних засобів із вбудованими засобами оцінки параметрів регресійної моделі спрощує та прискорює проведення регресійного аналізу даних.

У MS Excel визначення параметрів простої лінійної регресії за методом найменших квадратів здійснюється з використанням функції *ЛИНЕЙН()/LINEST()*.

13.1.4. Оцінка загальної якості регресійної моделі

Після того, як регресійна модель побудована, необхідно оцінити її загальну якість, яка включає:

1) перевірку правильності моделі – чи відповідає побудована модель наявним емпіричним даним;
2) оцінку ступеня апроксимації емпіричних даних рівнянням регресії;
3) перевірку адекватності побудованої моделі – з'ясування значущості регресійної залежності між змінними;
4) аналіз статистичної значущості параметрів моделі;
5) оцінку тісноти зв'язку змінних.

1. ***Перевірка правильності*** побудованої регресійної моделі проводиться з використанням ***показників варіації***, для визначення яких обчислюють:

a) $\hat{y}_i$ – теоретичні значення залежної змінної, розраховані за рівнянням регресії для емпіричних значень незалежної змінної x_i;

b) $\hat{y}_i - \bar{y}$ – варіацію значень залежної змінної $\hat{y}_i$, розрахованих за рівнянням регресії, навколо середнього значення залежної змінної $\bar{y}$, розрахованого за емпіричними даними;

c) $y_i - \hat{y}_i$ – варіацію значень залежної змінної $\hat{y}_i$, розрахованих за рівнянням регресії, навколо фактичних значень залежної змінної y_i.

Визначені показники варіації дозволяють розрахувати суми квадратів відхилень:

1) суму квадратів відхилень від середніх SS_r, що пояснюється регресією, із кількістю ступенів свободи $k_1 = m$:

$$SS_r = \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2 , \tag{13.10}$$

2) залишкову суму квадратів відхилень SS_ε фактичних значень результативної ознаки y_i від розрахованих за регресійною моделлю теоретичних значень $\hat{y}_i$, із кількістю ступенів свободи $k_2 = n - m - 1$:

$$SS_\varepsilon = \sum_{i=1}^{n} (\hat{y}_i - y_i)^2 , \tag{13.11}$$

3) загальну суму квадратів відхилень від середніх $SS_{заг}$ із кількістю ступенів свободи $k_3 = n - 1$:

$$SS_{заг} = \sum_{i=1}^{n} (y_i - \bar{y})^2 , \tag{13.12}$$

де n – кількість об'єктів набору даних, m – кількість незалежних змінних.

Перевірка правильності побудови рівняння регресії здійснюється за ***основним варіаційним рівнянням***:

$$SS_{заг} = SS_r + SS_\varepsilon , \tag{13.13}$$

Якщо це рівняння є правильною рівністю, то рівняння регресії побудовано правильно. Сума квадратів відхилень, що пояснюється регресією, обумовлена впливом на досліджувану ознаку факторів – незалежних змінних, а залишкова сума квадратів відхилень – впливом випадкових неврахованих чинників.

2. Для ***оцінки точності апроксимації*** емпіричних даних рівнянням регресійної моделі розраховують ***коефіцієнт детермінації*** R^2:

$$R^2 = 1 - \frac{SS_\varepsilon}{SS_{заг}} = \frac{SS_r}{SS_{заг}} . \tag{13.14}$$

Значення коефіцієнта детермінації знаходяться у діапазоні від 0 до 1: $0 \le R^2 \le 1$. Для оцінки апроксимації використовують наступні емпіричні правила:

1) $R^2 > 0{,}95$ – висока точність апроксимації;

2) $0{,}6 \le R^2 \le 0{,}95$ – задовільна, прийнятна апроксимація;

3) $R^2 < 0{,}6$ – незадовільна апроксимація.

Чим ближчим є значення коефіцієнта детермінації до 1, тим краще модель описує емпіричні дані. Наприклад, якщо $R^2 = 0{,}97$, то 97 % варіації результативної ознаки пояснюється рівнянням регресії.

У більшості інтегрованих комп'ютерних пакетів коефіцієнт детермінації розраховується автоматично.

3. Для ***перевірки на адекватність*** моделі здійснюють оцінку статистичної значущості рівняння регресії в цілому з використаня ***F-критерію Фішера***.

Основні етапи перевірки регресійної моделі на адекватність є наступними.

3.1. Формулюють нульову та альтернативну гіпотези:

H_0: фактори не впливають на досліджувану ознаку (зв'язок між залежною результативною ознакою та незалежними змінними – факторами – не є істотним);

H_1: фактори впливають на досліджувану ознаку (зв'язок між залежною результативною ознакою та незалежними змінними є статистично значущим).

3.2. Обчислюють дисперсії за формулами:

a) незміщену оцінку дисперсії регресії:

$$\sigma_r^2 = \frac{1}{k_1} SS_r = \frac{1}{m} \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2 ; \tag{13.15}$$

b) незміщену оцінку дисперсії залишків:

$$\sigma_\varepsilon^2 = \frac{1}{k_2} SS_\varepsilon = \frac{1}{n-m-1} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 ; \tag{13.16}$$

c) незміщену оцінку загальної дисперсії:

$$\sigma_{заг}^2 = \frac{1}{k_3} SS_{заг} = \frac{1}{n-1} \sum_{i=1}^{n} (y_i - \bar{y})^2 . \tag{13.17}$$

3.3. Розраховують емпіричне значення критерію Фішера $F_{емп}$ за формулою:

$$F_{емп} = \frac{\sigma_r^2}{\sigma_\varepsilon^2} = \frac{R^2}{1-R^2} \cdot \frac{n-m-1}{m}, \tag{13.18}$$

3.4. На рівні значущості α розраховують критичне значення критерію Фішера $F_{кр}(\alpha, k_1, k_2)$ зі ступенями свободи $k_1 = m$ і $k_2 = n-m-1$. Якщо $F_{емп} < F_{кр}$ – роблять висновок, про відсутність регресійної залежності між залежною та незалежними змінними – ***модель неадекватна***. Якщо $F_{емп} \geq F_{кр}$ – нульова гіпотеза відкидається, роблять висновок, що між залежною та усіма незалежними змінними є статистично значуща регресійна залежність із ймовірністю $p = 1-\alpha$ – ***модель адекватна.***

4. ***Перевірка значущості параметрів рівняння*** лінійної регресії здійснюється шляхом перевірки статистичних гіпотез із використанням ***t-критерію Стьюдента***. Для нелінійної регресії методи оцінки значимості параметрів є іншими.

Основні етапи перевірки значущості параметра a_j рівняння лінійної регресії є наступними.

4.1. Формулюють нульову та альтернативну гіпотези:

H_0: $a_j = 0$, параметр a_j рівняння регресії генеральної сукупності рівний нулю (параметр не є статистично значимим, незалежна змінна x_j не впливає суттєво на результативну ознаку);

H_1: $a_j \neq 0$, параметр a_j рівняння регресії генеральної сукупності відмінний від нуля (параметр є статистично значимим, незалежна змінна x_j істотно впливає на результативну ознаку).

4.2. Обчислюють емпіричне значення критерію Стьюдента $t_{емп}$:

$$t_{емп} = \frac{|a_j|}{m_{a_j}}, \tag{13.19}$$

де a_j – j-й параметр рівняння регресії, m_{a_j} – стандартна похибка j-го параметра.

Для простої лінійної регресії стандартну похибку параметрів обчислюють за формулами:

а) для параметра a_0 – вільного члена:

$$m_{a_0} = \sqrt{\frac{\sigma_\varepsilon^2 \cdot \sum_{i=1}^{n} x_i^2}{n \cdot \sum_{i=1}^{n} (x_i - \bar{x})^2}}, \qquad (13.20)$$

b) для параметра a_1 – коефіцієнта регресії:

$$m_{a_1} = \sqrt{\frac{\sigma_\varepsilon^2}{\sum_{i=1}^{n} (x_i - \bar{x})^2}}, \qquad (13.21)$$

де σ_ε^2 – залишкова дисперсія (формула 13.13), x_i – емпіричні значення незалежної змінної x, $\bar{x}$ – середнє значення незалежної змінної x, n – кількість об'єктів набору даних.

4.3. Визначають критичне значення критерію Стьюдента $t_{кр}$ на рівні значущості α для числа ступенів свободи $l = n - m - 1$.

4.4. Порівнюють $t_{емп}$ та $t_{кр}$: якщо $t_{емп} \le t_{кр}$ – нульова гіпотеза про відсутність статистичної значущості параметра рівняння регресії генеральної сукупності (відсутність статистично значимого зв'язку між відповідними ознаками) приймається з ймовірністю $p = 1 - \alpha$. У протилежному випадку, якщо емпіричне значення критерію Стьюдента більше за критичне $t_{емп} > t_{кр}$, приймається альтернативна гіпотеза: параметр рівняння регресії є статистично значимим із ймовірністю $p = 1 - \alpha$.

5. Показником ***тісноти лінійного зв'язку*** залежної та незалежних змінних є ***коефіцієнт кореляції Пірсона***, який для простої лінійної регресії розраховують за формулою:

$$r_{xy} = \frac{\mathrm{cov}(x, y)}{\sigma_x \sigma_y}, \qquad (13.22)$$

де $\mathrm{cov}(x, y) = \frac{1}{n} \sum (x_i - \bar{x})(y_i - \bar{y})$ – коваріація, сумісна варіація незалежної змінної x та залежної змінної y, σ_x і σ_y – середньо квадратичне відхилення незалежної та залежної змінних.

Чим ближчим буде значення коефіцієнта кореляції до 1 або до -1, тим сильнішим є прямий або обернений лінійний регресійний зв'язок.

Для оцінки тісноти нелінійного зв'язку використовують ***індекс кореляції*** η_{xy}:

$$\eta_{xy} = \sqrt{\frac{SS_r}{SS_{заг}}} = \sqrt{1 - \frac{SS_\varepsilon}{SS_{заг}}} = \sqrt{R^2}, \qquad (13.23)$$

де R^2 – коефіцієнт детермінації.

Індекс кореляції є універсальним показником тісноти зв'язку безвідносно до форми цього зв'язку (лінійної, нелінійної, багатофакторної). Чим ближчим до 1 буде значення η_{xy}, тим вищою буде адекватність моделі та тіснішим зв'язок між змінними.

13.2. ПОБУДОВА РЕГРЕСІЙНОЇ МОДЕЛІ ЗАСОБАМИ MS EXCEL

13.2.1. Здійснення однофакторного лінійного дисперсійного аналізу даних

Завдання 1. Побудувати регресійну модель, яка описує залежність сумарних виробничих витрат від обсягу виробництва. Емпіричні дані наведено у таблиці 13.1.

1. У MS Excel формуємо електронну таблицю, використовуючи дані з таблиці 13.1 та здійснюємо візуалізацію емпіричного набору даних – будуємо емпіричну лінію регресії, обравши тип діаграми – *точкова* (рис. 13.3).

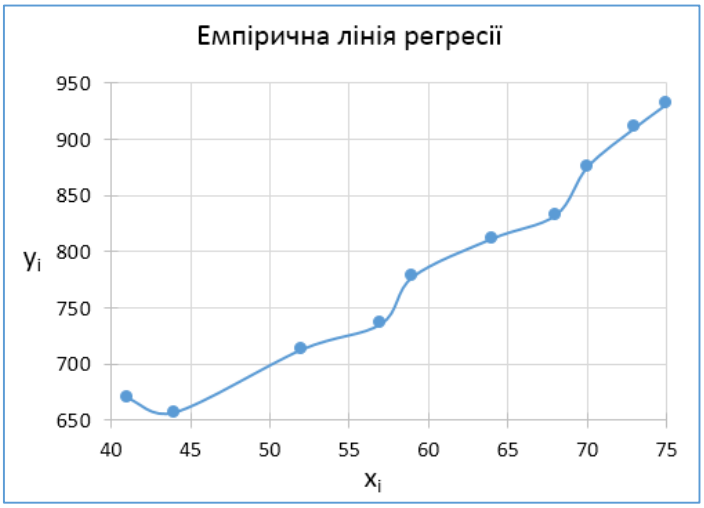
Таблиця 13.1

Дані про сумарні виробничі витрати та обсяги виробництва

Обсяг виробництва x *(тис. од.)*	41	44	52	57	59	64	68	70	73	75
Сумарні виробничі витрати y *(тис. грн)*	670	657	713	736	778	812	833	876	911	932

	A	B	C
1			
2		x_i	y_i
3		41	670
4		44	657
5		52	713
6		57	736
7		59	778
8		64	812
9		68	833
10		70	876
11		73	911
12		75	932

а) таблиця з даними



б) емпірична лінія регресії

Рис. 13.3. Графічне зображення емпіричних даних у MS Excel

2. На основі аналізу графічного зображення емпіричних даних здійснюємо вибір моделі регресії. Оскільки емпірична лінія регресії наближається до прямої лінії, висуваємо гіпотезу про лінійну залежність змінної y від змінної x. Рівняння регресії будемо шукати у вигляді $y = ax + b$.

3. Для знаходження a і b – параметрів рівняння регресії $y = ax + b$, скористаємося функцією MS Excel *ЛИНЕЙН() / LINEST()*, яка повертає параметри лінійного наближення за методом найменших квадратів:

a) підготуємо комірки для коефіцієнтів лінійної регресії, які будуть розраховуватися (рис. 13.4);

	H	I	J	K	L
1		Рівняння регресії: y=ax+b			
2		a	b		
3					

Рис. 13.4. Комірки для розрахунку параметрів лінійної регресії в MS Excel

b) виділяємо комірки *I3:J3*, набираємо формулу *=ЛИНЕЙН(C3:C12;B3:B12) / LINEST(C3:C12;B3:B12)* та, оскільки це функція масиву, натискаємо комбінацію клавіш *Ctrl* + *Shift* + *Enter*;

c) у комірках *I3* та *J3* з'являються розраховані коефіцієнти лінійної регресії (рис. 13.5);

d) отже, шукане рівняння регресії має вигляд $y = 8{,}06x - 305{,}83$.

	H	I	J	K	L
1		Рівняння регресії: y=ax+b			
2		a	b		
3		8,059	305,83		

Рис. 13.5. Розрахунок параметрів лінійної регресії в MS Excel

4. Для ***перевірки правильності*** побудованої регресійної моделі формуємо таблицю MS Excel, розрахувавши (рис. 13.6):

1) суму квадратів відхилень від середніх, що пояснюється регресією (формула 13.15):

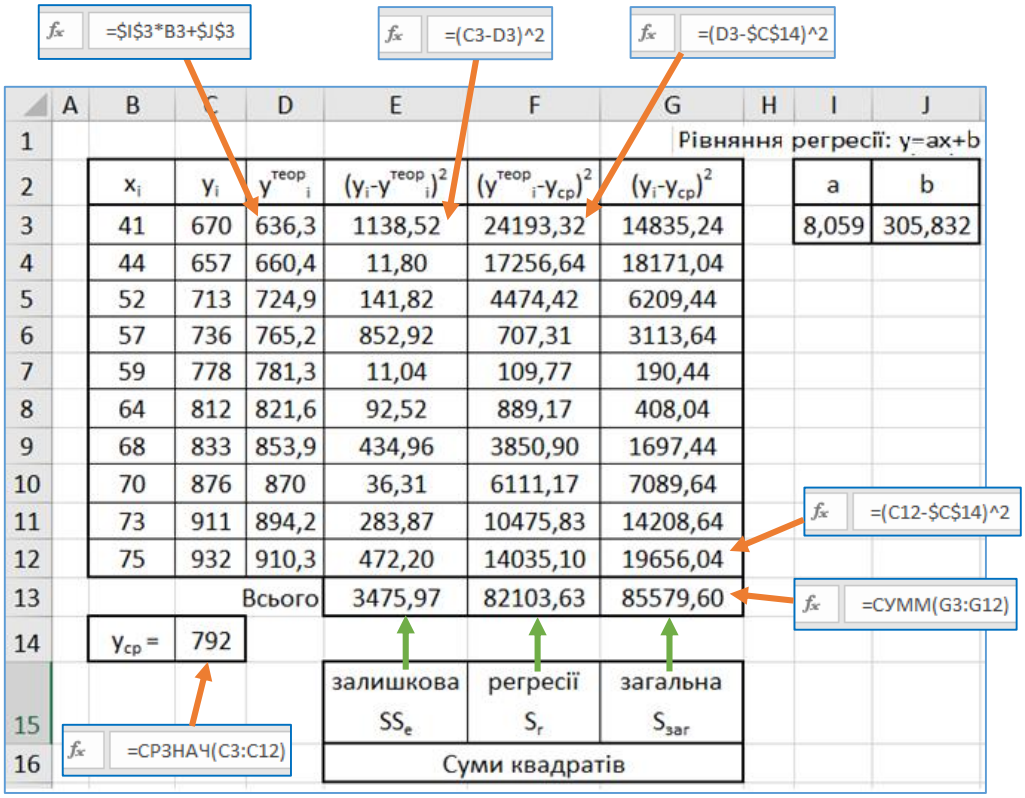
$$SS_r = \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2 = 82103{,}63;$$

2) залишкову суму квадратів відхилень фактичних значень результативної ознаки від значень, розрахованих за регресійною моделлю (формула 13.16):

$$SS_\varepsilon = \sum_{i=1}^{n} (\hat{y}_i - y_i)^2 = 3475{,}97;$$

3) загальну суму квадратів відхилень від середніх (формула 13.17):

$$SS_{заг} = \sum_{i=1}^{n} (y_i - \bar{y})^2 = 85579{,}6.$$



x_i	y_i	$y^{теор}_i$	$(y_i-y^{теор}_i)^2$	$(y^{теор}_i-y_{ср})^2$	$(y_i-y_{ср})^2$
41	670	636,3	1138,52	24193,32	14835,24
44	657	660,4	11,80	17256,64	18171,04
52	713	724,9	141,82	4474,42	6209,44
57	736	765,2	852,92	707,31	3113,64
59	778	781,3	11,04	109,77	190,44
64	812	821,6	92,52	889,17	408,04
68	833	853,9	434,96	3850,90	1697,44
70	876	870	36,31	6111,17	7089,64
73	911	894,2	283,87	10475,83	14208,64
75	932	910,3	472,20	14035,10	19656,04
		Всього	3475,97	82103,63	85579,60
$y_{ср}$ =	792				

Рис. 13.6. Розрахунок сум квадратів відхилень у MS Excel

5. Оскільки основне варіаційне рівняння $SS_{заг} = SS_r + SS_\varepsilon$ (формула 13.18) для побудованої моделі є тотожністю: 85579,6 = 82103,63 + 3475,97, робимо висновок – рівняння регресії побудовано правильно.

6. Для ***оцінки точності апроксимації*** емпіричних даних рівнянням регресії за формулою 13.14 розраховуємо коефіцієнт детермінації, ввівши у комірку *E24* формулу *=F13/G13* (рис. 13.7).

7. Отримане значення коефіцієнта детермінації $R^2 = 0{,}959$ свідчить, що 95,9% варіації результативної ознаки y пояснюється рівнянням регресії. Що свідчить про високу точність апроксимації емпіричних даних рівнянням регресії.

8. Для ***визначення тісноти лінійного зв'язку*** залежної змінної y та незалежної змінної x розрахуємо коефіцієнт кореляції Пірсона за формулою 13.22. Для цього скористаємося функцією MS Excel

PEARSON()/КОРРЕЛ(), ввівши у комірку *E25* формулу *=КОРРЕЛ(B3:B12;C3:C12)* (рис. 13.7). Отримане значення коефіцієнта кореляції r_{xy} = 0,9795 свідчить, що між результативною ознакою y та факторною ознакою x існує тісний лінійний зв'язок.

	C	D	E	F	G	H
23				Формули:		
24		R^2	0,9594	=F13/G13		
25		r_{xy}	0,9795	=КОРРЕЛ(B3:B12;C3:C12)		
26						

Рис. 13.7. Розрахунок коефіцієнтів детермінації та кореляції

9. Для ***перевірки адекватності побудованої моделі*** – статистичної значущості рівняння регресії, формулюємо нульову та альтернативну гіпотези:

H_0: фактор x не впливає на досліджувану ознаку y (зв'язок між залежною та незалежною змінними не є істотним);

H_1: фактор x впливає на досліджувану ознаку y (зв'язок між залежною та незалежною змінними є статистично значущим).

10. Розраховуємо статистичні оцінки дисперсій (рис. 13.8):

1) дисперсію регресії (формула 13.20, $k_1 = m = 1$): $\sigma_r^2 = \frac{1}{k_1} SS_r = 82103{,}626$;

2) залишкову дисперсію (формула 13.21, $k_2 = n - m - 1 = n - 2 = 8$, $n = 10$):

$$\sigma_\varepsilon^2 = \frac{1}{k_2} SS_\varepsilon = 434{,}497;$$

3) загальну дисперсію (формула 13.22, $k_3 = n - 1 = 10 - 1 = 9$):

$$\sigma_{заг}^2 = \frac{1}{k_3} SS_{заг} = 9508{,}844.$$

	J	K	L	M	N	O
16						
17		Кількість об'єктів	n=	10		
18		Кількість змінних	m=	1		
19						
20		Дисперсії		Формули		
21		Залишкова, σ^2_e	434,497	=E13/(M17-M18-1)		
22		Регресії, σ^2_r	82103,626	=F13/M18		
23		Загальна, $\sigma^2_{заг}$	9508,844	=G13/(M17-1)		
24						

Рис. 13.8. Розрахунок дисперсій у MS Excel

11. Для розрахунку значень F-критерію Фішера (рис. 13.9):

1) емпіричного критерію Фішера $F_{емп}$ (формула 13.18): у комірку *E20* вводимо формулу *=L22/L21*;

2) критичного значення критерію Фішера $F_{кр}$ на рівні значущості α = 0,05 зі ступенями свободи $k_1 = m = 1$ і $k_2 = n - m - 1 = 8$ у комірку *E21* вводимо формулу *=F.ОБР.ПХ(0,05;M18;M17-2)*.

	C	D	E	F	G	H
19				Формули:		
20		$F_{емп}$	188,963	=L22/L21		
21		$F_{кр}$	5,318	=F.ОБР.ПХ(0,05;M18;M17-2)		
22						

Рис. 13.9. Розрахунок емпіричного та критичного значень критерію Фішера

12. Порівнявши емпіричне значення критерію Фішера $F_{емп}$ = 188,963 із критичним $F_{кр}$ = 5,318, було виявлено, що емпіричне значення критерію Фішера є більшим за критичне на рівні значущості 0,05.

13. Отриманий результат дає підстави для відхилення нульової гіпотези та прийняття альтернативної гіпотези, відповідно до якої ми можемо стверджувати, що з ймовірністю 95% вплив фактора – незалежної змінної x на досліджувану ознаку y – є статистично значущим. Отже, побудована регресійна модель є адекватною.

13.2.2. Побудова лінійної регресійної моделі за допомогою Пакета аналізу MS Excel

Завдання 2. Побудувати регресійну модель, що описує залежність сумарних виробничих витрат від обсягу виробництва (табл. 13.1) із використанням засобів Пакета аналізу MS Excel.

1. Для вибору інструменту *Регресія* необхідно обрати вкладку *Дані* – групу *Аналіз – Аналіз даних – Регресія*. У вікні, що відкриється, необхідно здійснити наступні налаштування (рис. 13.10):
   - вхідний інтервал Y: *$C$3:$C$12*;
   - вхідний інтервал X: *$B$3:$B$12*;
   - рівень надійності: 95%;
   - вихідний інтервал: *P1*.
2. Після вибору параметрів натискаємо *ОК*.

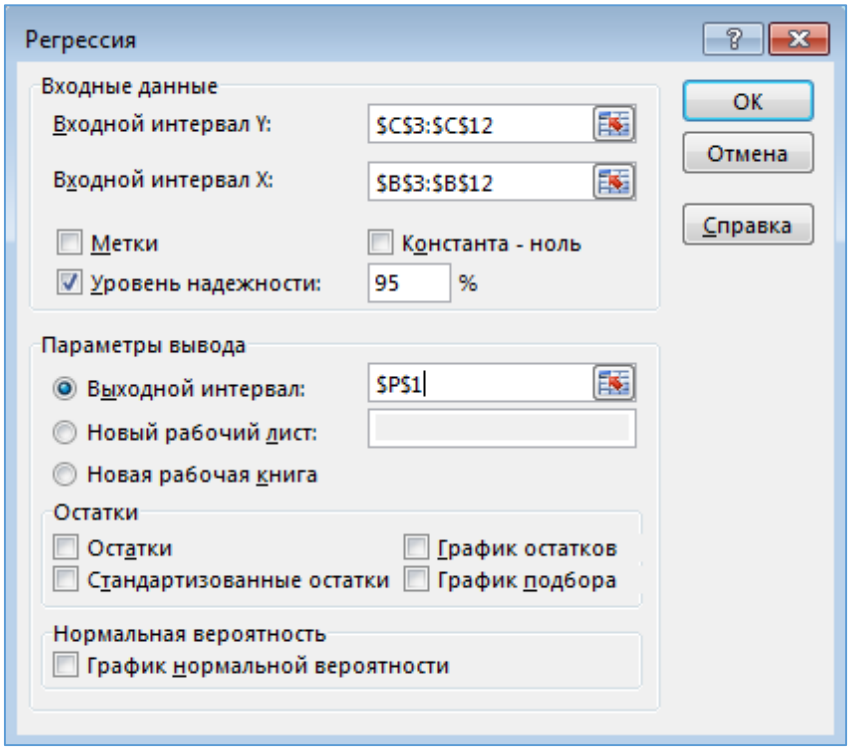


Рис. 13.10. Налаштування параметрів для лінійної регресії в MS Excel

3. У вказаному при налаштуванні вихідному інтервалі буде виведена інформація стосовно проведеного регресійного аналізу. Зробимо її аналіз.

4. Аналіз показників якості побудованої регресійної моделі (рис. 13.11):

a) коефіцієнт кореляції *R* = 0,979 свідчить про наявність тісного лінійного зв'язку між результативною та факторною ознаками;

b) коефіцієнт детермінації *R-квадрат* = 0,959 показує, що 95,9% варіації результативної ознаки обумовлено обраною факторною ознакою, а 4,1% варіації обумовлені факторами, які не включено до моделі.

ВЫВОД ИТОГОВ	
Регрессионная статистика	
Множественный R	0,9795
R-квадрат	0,9594
Нормированный R-квадрат	0,9543
Стандартная ошибка	20,8446
Наблюдения	10

Рис. 13.11. Показники якості побудованої регресійної моделі

5. Зробимо аналіз показників, що характеризують адекватність моделі (рис. 13.12). Для емпіричного значення критерію Фішера F=188,9626 показник значущості F є меншим за рівень значущості $\alpha = 0{,}05$, що свідчить про статистичну значущість, адекватність побудованої моделі регресії.

Отримані результати співпадають із оцінками, які було зроблено у завданні 1 із використанням інших інструментальних засобів MS Excel.

Дисперсионный анализ					
	df	*SS*	*MS*	*F*	*Значимость F*
Регрессия	1	82103,6261	82103,63	188,9626	7,56595E-07
Остаток	8	3475,9739	434,4967		
Итого	9	85579,6			

Рис. 13.12. Результати розрахунків перевірки значущості рівняння регресії

6. Аналіз таблиці коефіцієнтів (рис. 13.13) показує, що параметри рівняння регресії є статистично значимим, оскільки *p*-значення є меншим за рівень значущості $\alpha = 0{,}05$ для емпіричного значення критерію Стьюдента вільного члена a_0 ($t_{емп} = 8{,}504$) і коефіцієнта регресії a_1 ($t_{емп} = 13{,}746$).

Порівняння розрахованих t-критеріїв Стьюдента з критичним, визначеним на рівні значущості $\alpha = 0{,}05$, рівним $t_{кр} = 2{,}306$ також свідчить про те, що параметри побудованої лінійної моделі регресії будуть значущими, оскільки для обох коефіцієнтів $t_{емп} > t_{кр}$.

Критичне значення критерію Стьюдента в MS Excel можна визначити за формулою:

=СТЬЮДЕНТ.ОБР.2Х(α; $n-2$)/T.INV.2T(α; $n-2$),

ввівши у комірку формулу: =СТЬЮДЕНТ.ОБР.2Х(0,05;10-2).

	Коэффициенты	*Станд. ошибка*	*t-статистика*	*P-Значение*
Y-пересечение	305,83190	35,96175	8,50437	0,00003
Переменная X 1	8,05917	0,58628	13,74637	0,00000

Рис. 13.13. Таблиця коефіцієнтів регресії

13.2.3. Побудова нелінійних регресійних залежностей у MS Excel

Завдання 3. Здійснити побудову нелінійних регресійних моделей, що описують залежність сумарних виробничих витрат від обсягу виробництва (табл. 13.1), оцінити їх якість та обрати модель, яка найбільш точно апроксимує емпіричні дані.

1. Діаграму з емпіричною лінією регресії, побудувану у завданні 1 для заданих емпіричних даних (рис. 13.3, б), копіюємо на новий робочий аркуш та робимо п'ять її копій.

2. Для першої копії діаграми виклиємо контекстне меню емпіричної лінії регресії та обираємо *Додати лінію тренда*.

3. У правій частині вікна з'явиться панель *Формат ліній тренда* (рис. 13.14), де обираємо тип лінії регресії – лінійна, прогноз вперед – на 5 періодів, та установлюємо прапорець – показувати рівняння на діаграмі і прапорець – помістити величину достовірності апроксимації R^2. На діаграмі з'явиться рівняння лінійної регресії та розрахований коефіцієнт детермінації R^2 (рис. 13.15).

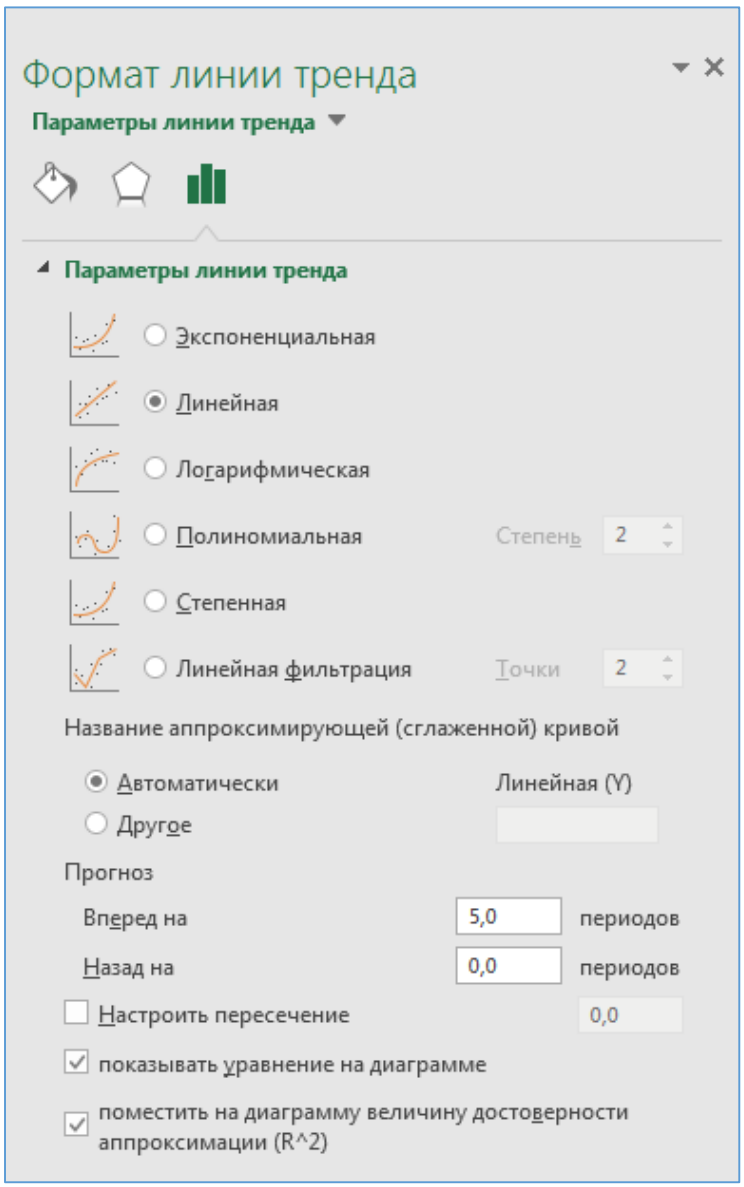


Рис. 13.14. Вікно вибору апроксимуючої функції

4. Розрахований коефіцієнт детермінації свідчить про високий рівень апроксимації емпіричних даних рівнянням лінійної регресії. Оцінка параметрів лінійної регресії та коефіцієнт детермінації співпадають із результатами,

які були отримані у завданнях 1 та 2 для цих емпіричних даних при побудові лінійної регресійної моделі з використанням інших інструментальних засобів MS Excel.

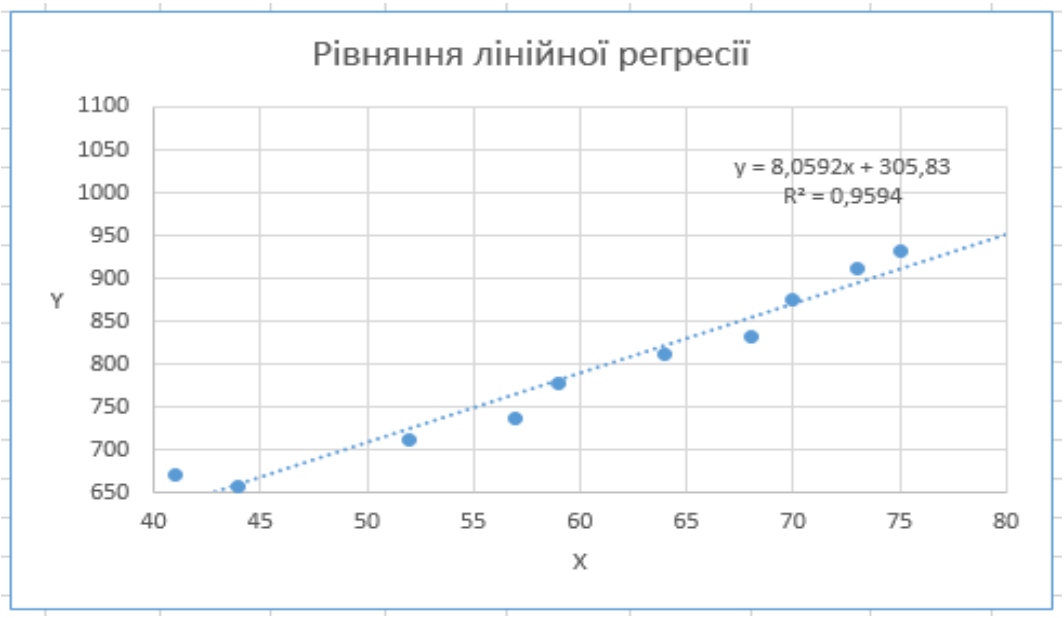


Рис. 13.15. Лінійна регресійна модель

5. Аналогічно на інших копіях діаграм необхідно побудувати експоненціальну, логарифмічну, поліноміальну та степеневу рейгресійні моделі (рис. 13.16–13.19). Розраховані для різних регресійних моделей коефіцієнти детермінації є досить високими і відрізняються не суттєво – знаходяться в межах від 0,94 до 0,99. Тому апроксимація емпіричних даних рівнянням лінійної регресії є доцільною, оскільки саме лінійна регресія має найменшу розрахункову та аналітичну складність.

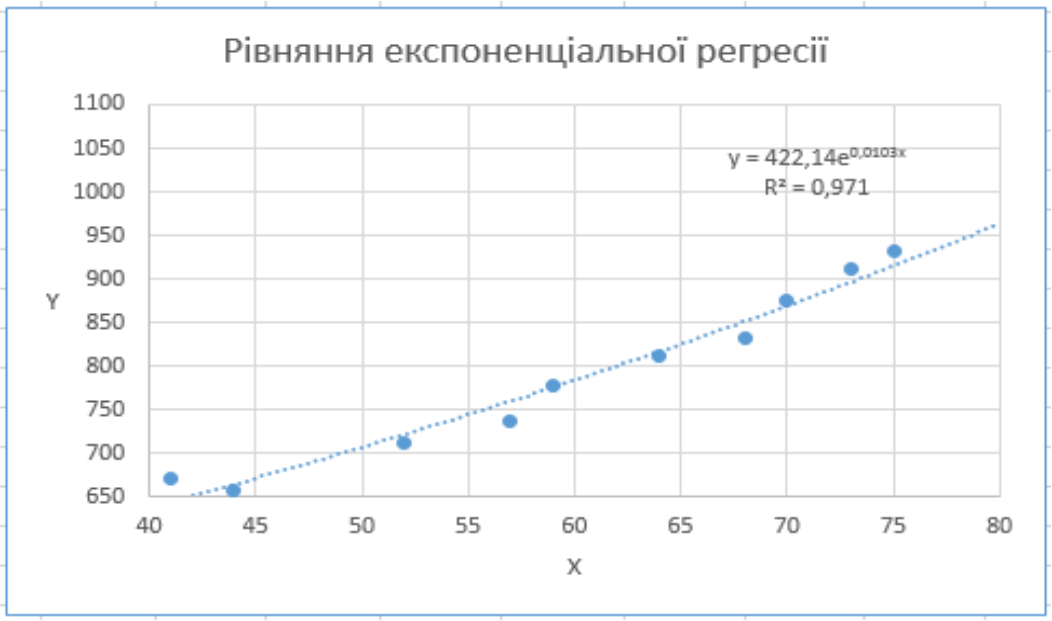


Рис. 13.16. Експоненціальна регресійна модель

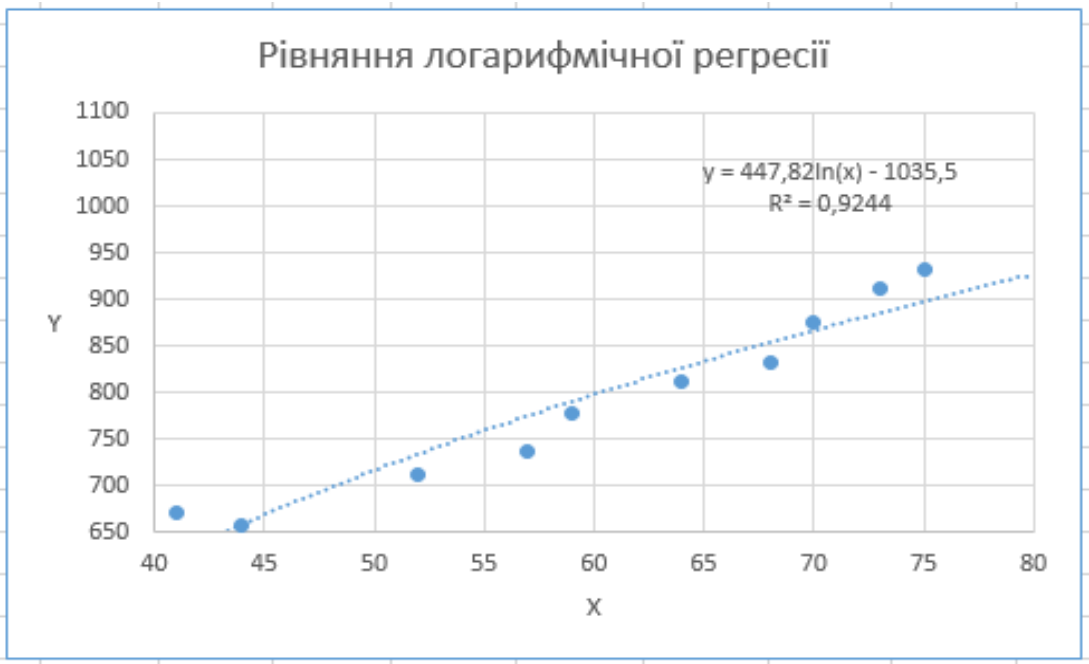


Рис. 13.17. Логарифмічна регресійна модель

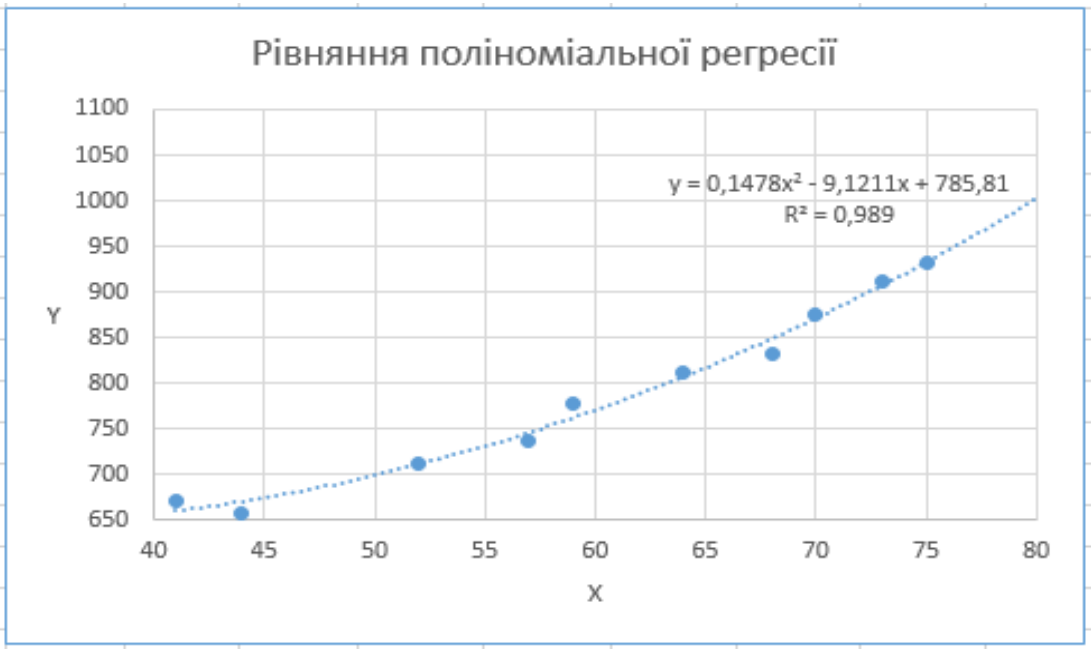


Рис. 13.18. Поліноміальна регресійна модель

6. Серед побудованих моделей найбільш якісною буде поліноміальна регресійна модель (рис. 13.18), оскільки вона має найбільший коефіцієнт детермінації, рівний 0,99. А це значить, що 99% варіації результативної ознаки пояснюється рівнянням поліноміальної регресії. Для перевірки правильності та адекватності поліноміальної регресійної моделі необхідно здійснити розрахунок показників варіації та емпіричного критерію Фішера так, як це було зроблено для лінійної регресійної моделі у завданні 1.

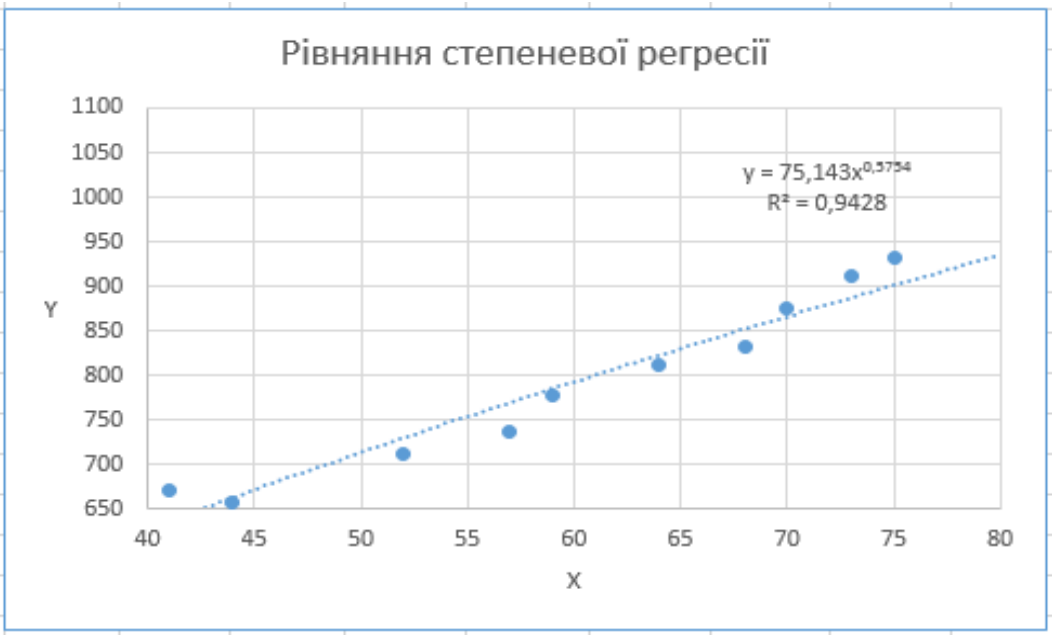


Рис. 13.19. Степенева регресійна модель

13.3. ПРОВЕДЕННЯ РЕГРЕСІЙНОГО АНАЛІЗУ ДАНИХ У SPSS

Завдання 4. У середовщі SPSS провести регресійний аналіз даних, що описують залежність сумарних виробничих витрат від обсягу виробництва (табл. 13.1). Здійснити побудову різних регресійних моделей, оцінити їх якість та обрати модель, яка найбільш точно апроксимує емпіричні дані.

1. У вікні редактора даних SPSS створюємо змінні, робимо їх налаштування та водимо дані (рис. 13.20).

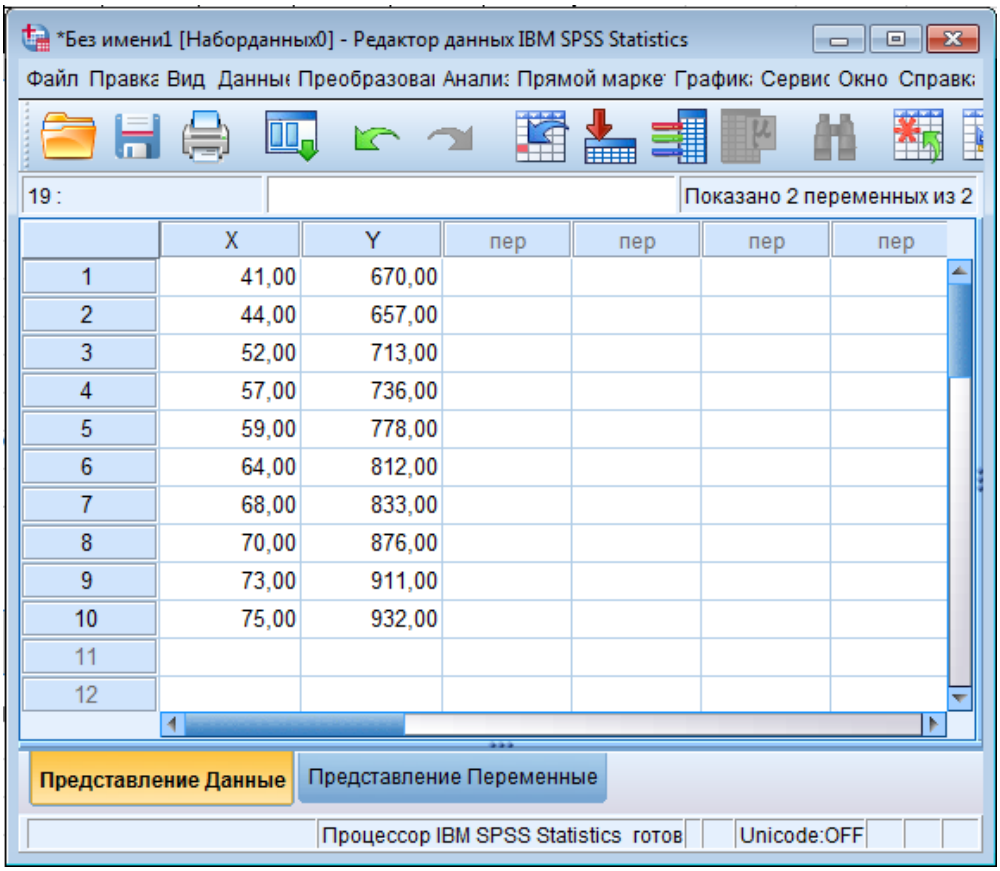


Рис. 13.20. Вікно редактора даних SPSS

2. Далі необхідно обрати пункти меню *Аналіз – Регресія – Лінійна*. У вікні Лінійна регресія, що відкриється, здійснюємо налаштування для побудови лінійної регресійної моделі, вказавши залежну та незалежну змінні (рис. 13.21).

3. Натиснувши кнопку *Статистика*, необхідно здійснити налаштування інформації, яка буде виведена: *оцінка коефіцієнтів регресії* та *Согласие модели* і *Измерение R-квадрат* (рис. 13.22). Після цього натиснути кнопку *Продолжить*, повернутися до вікна *Лінійна регресія* й натиснути кнопку *ОК*.

У разі необхідності можна також вказати інші параметри, які будуть визначені: довірчі інтервали, описові статистики, матриця коваріацій, діагностика колінеарності.

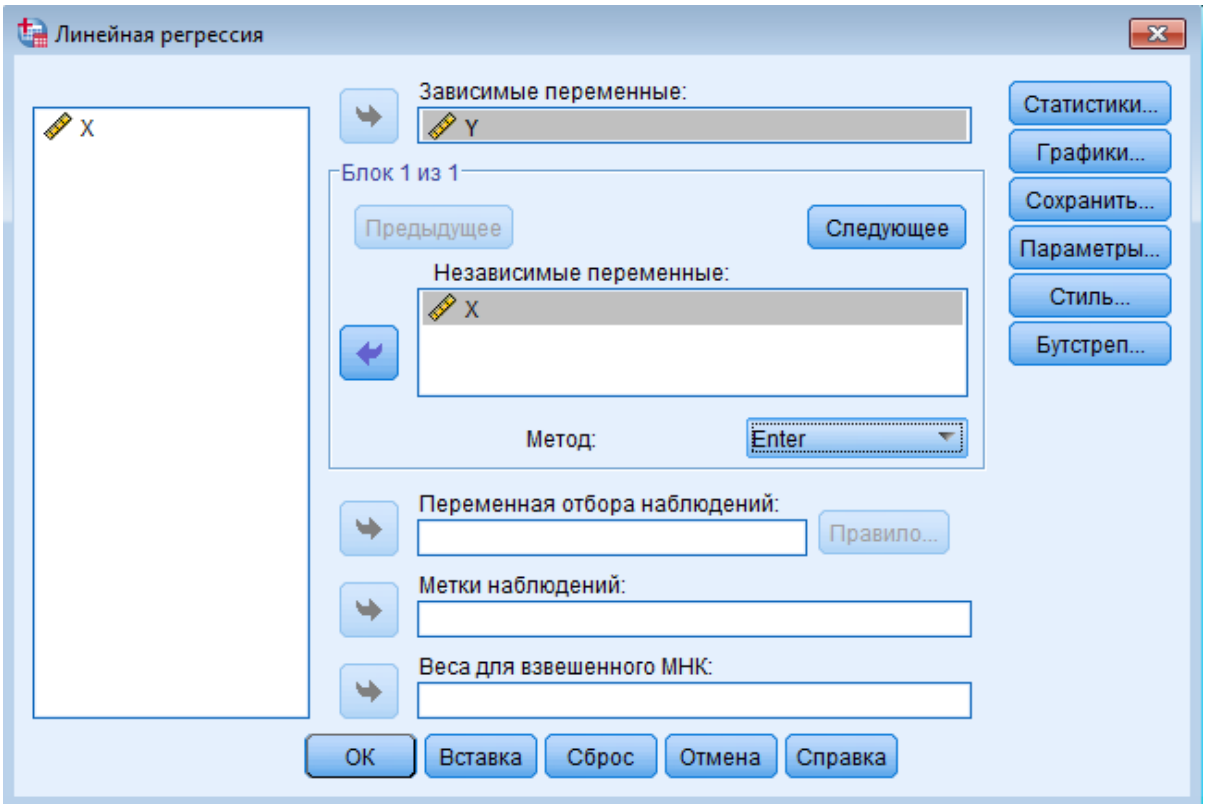


Рис. 13.21. Вікно Лінійна регресія

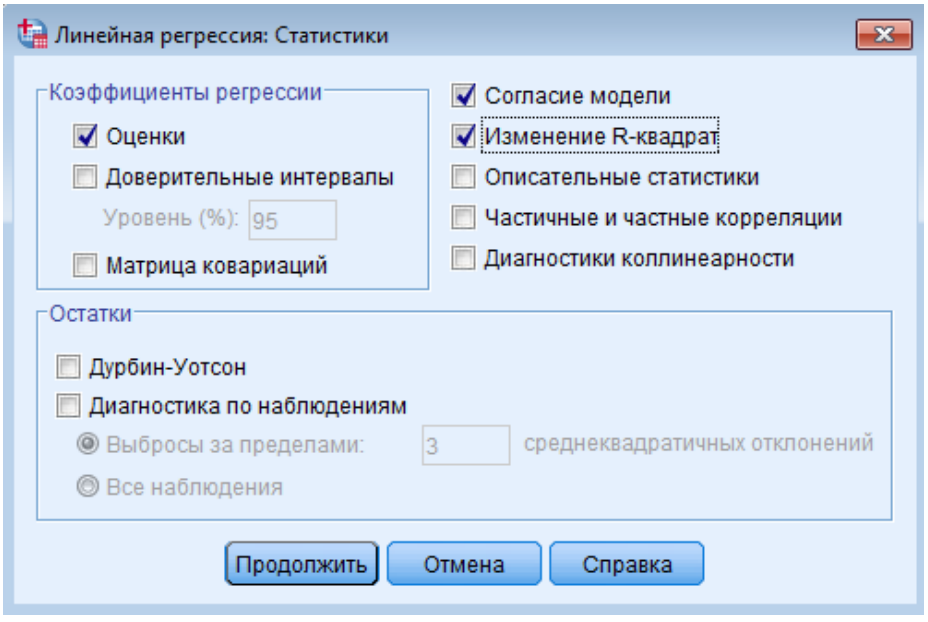


Рис. 13.22. Вікно Лінійна регресія: Статистики

4. У вікні виводу буде виведена інформація про проведений регресійний аналіз та здійснену оцінку його якості (рис. 13.23). Як бачимо, отримані значення збігаються з тими, які було отримано при проведенні регресійного аналізу засобами MS Excel.

Регрессия

[Набор данных0]

Введенные/удаленные переменные[a]

Модель	Введенные переменные	Удаленные переменные	Метод
1	X[b]	.	Enter

a. Зависимая переменная: Y

b. Все требуемые переменные введены.

Сводка для модели[b]

Модель	R	R-квадрат	Скорректированный R-квадрат	Стандартная ошибка оценки	Статистика изменений				
					Изменение R квадрат	Изменение F	ст.св.	ст.св.2	Знач. Изменение F
1	,979[a]	,959	,954	20,8445[illegible]	,959	188,963	1	8	,000

a. Предикторы: (константа), X

b. Зависимая переменная: Y

ANOVA[a]

Модель		Сумма квадратов	ст.св.	Средний квадрат	F	Знач.
1	Регрессия	82103,626	1	82103,626	188,963	,000[b]
	Остаток	3475,974	8	434,497		
	Всего	85579,600	9			

a. Зависимая переменная: Y

b. Предикторы: (константа), X

Коэффициенты[a]

Модель		Нестандартизованные коэффициенты		Стандартизованные коэффициенты	т	Знач.
		B	Стандартная Ошибка	Бета		
1	(Константа)	305,832	35,962		8,504	,000
	X	8,059	,586	,979	13,746	,000

Рис. 13.23. Виведення інформації про проведений регресійний аналіз

5. У вікні редактора даних обраємо пункти меню *Аналіз – Регресія – Підгонка кривих*. У вікні *Підгонка кривих*, що відкриється, необхідно здійснити налаштування для побудови різних регресійних моделей, вказавши залежну та незажну змінні (рис. 13.24), установити прапорці навпроти обраних моделей: лінійної, логарифмічної, квадратичної, експоненціальної та для виведення таблиці дисперсійного аналізу і натиснути кнопку *ОК*.

6. У вікні виводу будуть виведені графіки апроксимуючих функцій (рис. 13.25) та інформація про якість апроксимації обраними функціями емпіричних даних (рис. 13.26–13.29). Для кожної з моделей розраховано коефіцієнти детермінації, здійснено аналіз дисперсій та оцінку параметрів моделі.

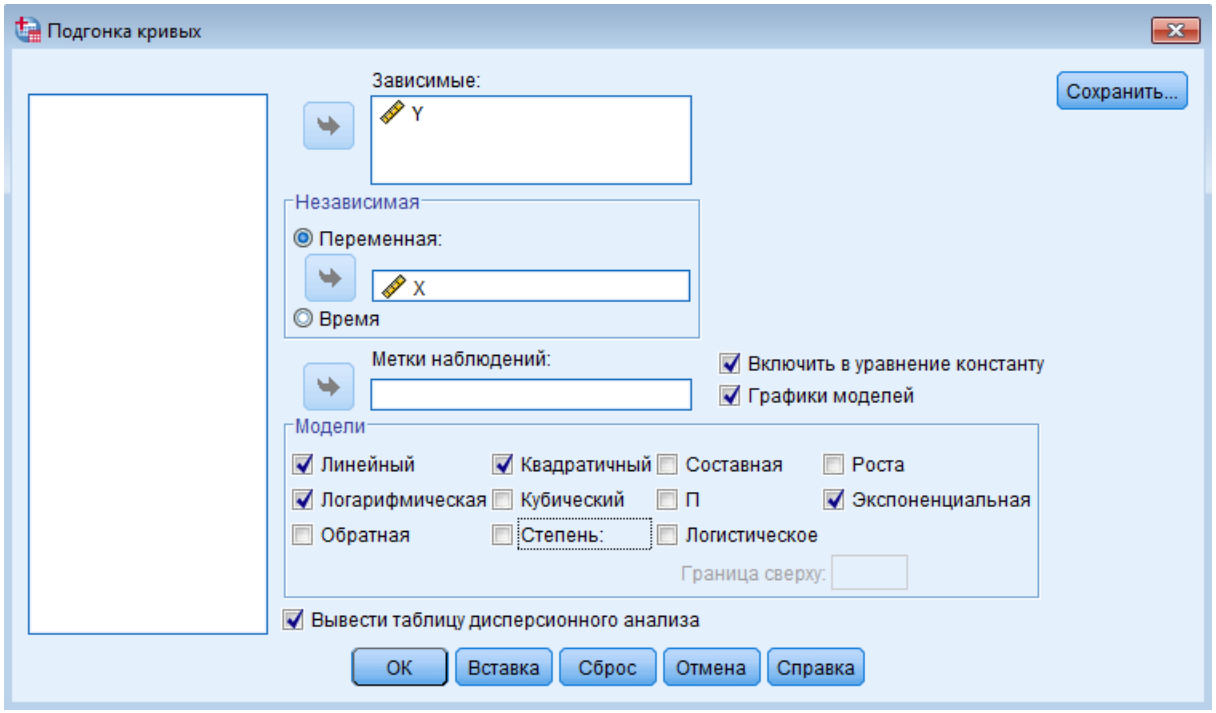


Рис. 13.24. Вікно Підгонка кривих

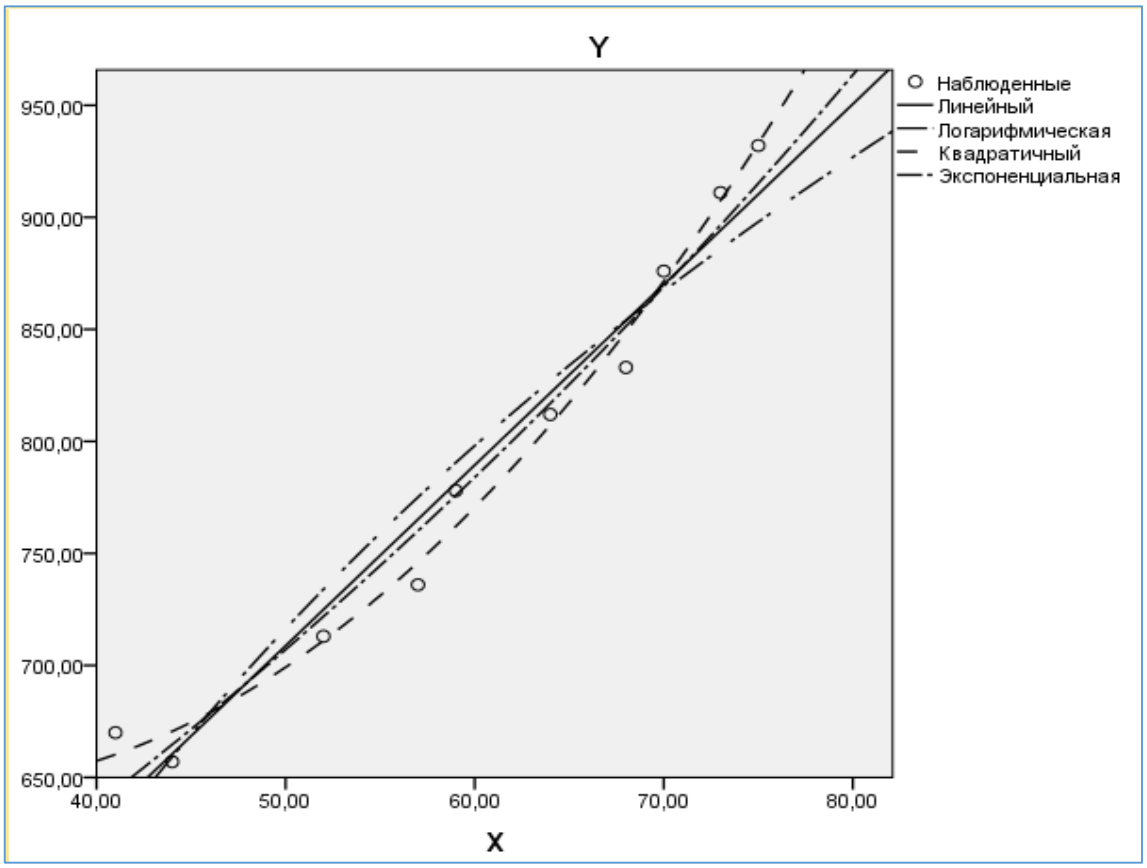


Рис. 13.25. Графіки побудованих апроксимуючих функцій

Линейная

Сводка для модели

R	R-квадрат	Скорректированный R-квадрат	Среднеквадратичная ошибка оценки
,979	,959	,954	20,845

Независимая переменная - это X.

ANOVA

	Сумма квадратов	ст.св.	Средний квадрат	F	Знач.
Регрессия	82103,626	1	82103,626	188,963	,000
Остаток	3475,974	8	434,497		
Всего	85579,600	9			

Независимая переменная - это X.

Коэффициенты

	Нестандартизованные коэффициенты		Стандартизованные коэффициенты		
	B	Стандартная Ошибка	Бета	т	Знач.
X	8,059	,586	,979	13,746	,000
(Константа)	305,832	35,962		8,504	,000

Рис. 13.26. Лінійна регресійна модель

Логарифмическая

Сводка для модели

R	R-квадрат	Скорректированный R-квадрат	Среднеквадратичная ошибка оценки
,961	,924	,915	28,436

Независимая переменная - это X.

ANOVA

	Сумма квадратов	ст.св.	Средний квадрат	F	Знач.
Регрессия	79110,682	1	79110,682	97,835	,000
Остаток	6468,918	8	808,615		
Всего	85579,600	9			

Независимая переменная - это X.

Коэффициенты

	Нестандартизованные коэффициенты		Стандартизованные коэффициенты		
	B	Стандартная Ошибка	Бета	т	Знач.
ln(X)	447,816	45,274	,961	9,891	,000
(Константа)	-1035,487	184,958		-5,598	,001

Рис. 13.27. Логарифмічна регресійна модель

Квадратичная

Сводка для модели

R	R-квадрат	Скорректированный R-квадрат	Среднеквадратичная ошибка оценки
,994	,989	,986	11,622

Независимая переменная - это X.

ANOVA

	Сумма квадратов	ст.св.	Средний квадрат	F	Знач.
Регрессия	84634,049	2	42317,025	313,277	,000
Остаток	945,551	7	135,079		
Всего	85579,600	9			

Независимая переменная - это X.

Коэффициенты

	Нестандартизованные коэффициенты		Стандартизованные коэффициенты		
	B	Стандартная Ошибка	Бета	т	Знач.
X	-9,121	3,983	-1,109	-2,290	,056
X ** 2	,148	,034	2,095	4,328	,003
(Константа)	785,805	112,694		6,973	,000

Рис. 13.28. Квадратична регресійна модель

Экспоненциальная

Сводка для модели

R	R-квадрат	Скорректированный R-квадрат	Среднеквадратичная ошибка оценки
,985	,971	,967	,022

Независимая переменная - это X.

ANOVA

	Сумма квадратов	ст.св.	Средний квадрат	F	Знач.
Регрессия	,135	1	,135	268,223	,000
Остаток	,004	8	,001		
Всего	,139	9			

Независимая переменная - это X.

Коэффициенты

	Нестандартизованные коэффициенты		Стандартизованные коэффициенты		
	B	Стандартная Ошибка	Бета	т	Знач.
X	,010	,001	,985	16,378	,000
(Константа)	422,135	16,311		25,880	,000

Зависимая переменная - это ln(Y).

Рис. 13.29. Експоненціальна регресійна модель

7. Порівнявши отриману інформацію для різних регресійних моделей, робимо висновок, що більш якісною буде квадратична регресійна модель, оскільки рівняння цієї регресії описує найбільший процент варіації результативної ознаки: 99%.

13.4. ЗАВДАННЯ ДЛЯ САМОСТІЙНОЇ РОБОТИ

Завдання 5. Побудувати регресійну модель залежності результативної ознаки y від фактора x. Вхідні дані необхідно сформувати з даних, наведених у таблицях 13.2 та 13.3 відповідно до індивідуального варіанта (табл/ 13.4).

1. Розв'язати задачу, здійснивши розрахунки із використанням функцій MS Excel:

a) побудувати емпіричну лінію регресії;

b) знайти параметри лінійного рівняння регресії;

c) перевірити правильність побудови моделі регресії з використанням основного варіаційного рівняння;

d) перевірити статистичну значущість рівняння регресії з використанням критерію Фішера;

e) установити, яка частка варіації результативної ознаки пояснюється рівнянням регресії, розрахувавши коефіцієнт детермінації;

f) визнатичити тісноту зв'язку між результативною та факторною ознакою з використанням коефіцієнта кореляції Пірсона.

2. Побудувати лінійну регресійну модель засобами Пакета аналізу MS Excel.

3. Побудувати регресійну модель засобами SPSS, здійснивши апроксимацію сукупністю підібраних функцій та визначити, яка з моделей буде більш якісно та адекватно описувати набір емпіричних даних.

КОНТРОЛЬНІ ПИТАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ № 13

1. Що таке регресійний аналіз?
2. Якою є основна мета регресійного аналізу даних?
3. Перерахуйте основні етапи та методи регресійного аналізу даних.
4. Лінійна регресійна залежність. Проста лінійна регресія.
5. Як визначають параметри лінійної регресійної моделі?
6. У чому полягає оцінка загальної якості регресійної моделі?
7. Яким чином здійснюють перевірку правильності побудованої регресійної моделі?
8. Як оцінюють точності апроксимації емпіричних даних рівнянням регресії?
9. Яким чином здійснюють перевірку регресійної моделі на адекватність?
10. Як здійснюється перевірка значущості параметрів рівняння лінійної регресії?
11. Як визначають тісноту зв'язку між факторною та результативними ознаками?
12. Інструментальні засоби та функції MS Excel для проведення регресійного аналізу даних.
13. Інструментальні засоби та функції SPSS для проведення регресійного аналізу.

Таблиця 13.2

Емпіричні дані значень фактора *x* для завдання 4

Значення фактора *x*									
X_0	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9
2,06	2,53	2,17	3,65	3,22	2,16	4,57	2,25	6,15	1,86
2,58	3,54	2,90	3,82	3,87	2,65	5,42	2,98	5,66	1,91
3,14	3,84	3,29	3,76	4,95	3,49	5,29	2,15	7,50	2,14
3,54	3,84	4,13	5,24	5,10	3,16	6,33	2,71	6,90	3,39
4,18	4,22	5,25	5,03	5,98	3,85	7,63	3,70	8,31	3,95
4,78	4,81	4,92	5,52	7,28	4,58	7,53	4,59	8,25	4,30
5,11	6,53	5,79	5,62	6,90	5,33	7,73	4,77	9,39	5,10
5,67	5,82	5,87	6,98	7,54	5,89	8,44	5,34	9,73	5,47
6,02	6,43	6,99	6,91	7,91	6,20	9,49	5,45	9,33	5,97
6,65	7,73	7,04	7,95	8,40	6,39	9,18	6,00	10,50	6,16
7,05	8,19	8,14	7,24	8,14	6,95	10,14	6,25	11,10	6,46
7,52	7,65	8,06	9,27	8,76	7,25	9,94	6,79	11,51	6,07
8,03	9,31	8,57	8,46	9,67	7,80	10,92	8,24	12,42	6,71
8,56	9,26	9,45	10,30	10,28	8,47	11,89	8,51	12,40	7,16
9,03	9,86	9,06	10,72	10,59	9,22	11,14	9,15	13,14	8,81

Таблиця 13.3

Емпіричні дані значень показника y для завдання 4

Значення показника y									
Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7	Y_8	Y_9
7,24	10,89	16,21	12,11	15,21	16,62	10,22	12,50	19,66	14,87
8,02	11,92	17,75	12,30	15,42	17,63	10,58	13,88	20,53	15,78
9,28	12,45	16,39	13,82	16,44	19,22	12,01	15,16	21,31	16,79
10,12	13,27	18,87	14,84	17,93	19,36	12,84	16,06	22,59	18,03
11,12	14,12	19,60	15,86	18,52	20,52	13,28	16,66	23,27	18,29
12,19	15,23	21,21	16,41	19,80	21,95	15,13	17,65	24,44	19,93
13,01	16,07	21,84	17,80	20,76	22,45	15,84	18,46	25,85	20,32
14,12	17,40	23,00	18,61	21,30	23,56	17,08	19,54	26,74	21,18
15,21	18,68	24,44	19,57	22,25	24,90	17,99	20,58	27,36	22,47
16,29	19,46	25,36	21,26	24,14	25,53	18,32	21,77	28,37	23,47
17,01	20,52	25,54	21,08	24,17	26,11	19,49	22,15	29,22	24,07
18,03	21,32	27,14	22,99	25,66	28,02	20,59	23,80	30,50	25,57
19,19	22,58	27,95	23,43	26,50	28,37	21,35	24,79	31,21	27,07
20,21	23,73	28,99	24,63	27,46	29,48	23,20	25,57	32,56	27,62
21,22	25,02	30,80	25,41	29,02	30,42	24,21	27,18	33,66	28,42

Таблиця 13.4

Вибір даних за варіантами

x – незалежна змінна (фактор), y – залежна змінна (відгук)

Варіант	Фактор x	Показник y	Варіант	Фактор x	Показник y
1	X_0	Y_0	17	X_7	Y_6
2	X_1	Y_1	18	X_8	Y_7
3	X_2	Y_2	19	X_9	Y_8
4	X_3	Y_3	20	X_2	Y_9
5	X_4	Y_4	21	X_3	Y_0
6	X_5	Y_5	22	X_4	Y_1
7	X_6	Y_6	23	X_5	Y_2
8	X_7	Y_7	24	X_6	Y_3
9	X_8	Y_8	25	X_7	Y_4
10	X_9	Y_9	26	X_9	Y_3
11	X_1	Y_0	27	X_4	Y_4
12	X_2	Y_1	28	X_5	Y_5
13	X_3	Y_2	29	X_6	Y_6
14	X_4	Y_3	30	X_8	Y_8
15	X_5	Y_4	31	X_9	Y_9
16	X_6	Y_5	32	X_9	Y_8

14. ФАКТОРНИЙ АНАЛІЗ ДАНИХ

Лабораторна робота № 14

Мета: закріплення знань про сутність, основні поняття, методи та етапи проведення факторного аналізу даних. Набуття навичок проведення факторного аналізу засобами пакету SPSS.

Теоретичні знання: основні положення, сутність факторного аналізу даних. Підтверджуючий та дослідницький факторний аналіз. Факторні навантаження, матриця факторних навантажень. Методи та моделі факторного аналізу. Обертання факторів. Основні етапи методу головних компонент. Інструментальні засоби проведення факторного аналізу в SPSS.

14.1. ВИЯВЛЕННЯ ЛАТЕНТНИХ ЗМІННИХ. ФАКТОРНИЙ АНАЛІЗ

14.1.1. Основні положення факторного аналізу даних

Факторний аналіз (англ. *Factor Analysis*) – це сукупність багатовимірних методів, які використовуються для опису зв'язків між змінними набору даних досліджуваної предметної області, з новими змінними, які називають факторами, кількість яких менша за кількість початкових змінних. За рахунок цього відбувається «стиснення» інформації шляхом зменшення розмірності простору ознак.

Метою факторного аналізу є знаходження таких комплексних факторів, які більш повно пояснюють спостережувані зв'язки між наявними змінними. Фактори зазвичай розглядають як широкі поняття, які описують досліджуване явище й важко піддаються безпосередньому виміру.

Основне завдання полягає в тому, щоб, досліджуючи характеристики об'єктів набору даних, виявити невелику кількість прихованих макропараметрів – факторів, якими в основному визначаються відмінності в значеннях вимірюваних параметрів. Між змінними виявляють кореляції та розбивають їх на групи, кожна з яких містить сильно корелюючі між собою змінні, які слабко корелюють зі змінними інших груп.

Це дозволяє скоротити число параметрів та оптимізувати структуру даних, поєднуючи сильно корелюючі між собою змінні в агрегатні макрозмінні – фактори. Основними вимогами при цьому є мінімальна втрата інформації та можливість інтерпретації факторів через змінні, з якими їх пов'язали.

Фактор – це кількісне вираження прихованої (латентної) змінної (агрегатної або макрозмінної), що пояснює взаємні кореляції, існуючі в наборі вихідних змінних. Він є внутрішньою, істотною характеристикою об'єктів досліджуваної предметної області, що інтерпретується як причина взаємопов'язаних значень, які будуть приймати наявні у досліджуваному наборі даних вимірювані змінні.

Виділяють два ***типи факторного аналізу***: підтверджуючий та дослідницький.

Підтверджуючий факторний аналіз (англ. *Confirmatory factor analysis, CFA*) використовується для перевірки гіпотези про існування зв'язку між спостережуваними змінними та їх прихованими структурами – факторами. Така перевірка слугує для спростування чи підтвердження уже існуючого твердження про наявність факторної структури.

Дослідницький факторний аналіз (англ. *Exploratory factor analysis, EFA*) здійснює дослідження структури змінних у наборі даних із метою виявлення факторів. Він зазвичай застосовується на початкових етапах аналізу даних для виявлення внутрішньої структури змінних.

Виявлені у результаті проведеного факторного аналізу узагальнені змінні – фактори – можуть бути використані у подальшому аналізі даних при розв'язанні задач кластерного аналізу та класифікації, побудові регресійних моделей.

14.1.2. Формальна постановка задачі

Формально постановку задачі факторного аналізу можна описати таким чином.

Нехай є n об'єктів набору даних, кожен із яких представлений набором m атрибутів – значень змінних. Тоді вихідна інформація може бути представлена у вигляді матриці даних, у якій рядкам відповідають об'єкти, а стовпцям – змінні, що характеризують ці об'єкти:

$$X = \left| X_1, ..., X_m \right| = \begin{vmatrix} x_{11} & ... & x_{1m} \\ ... & ... & ... \\ x_{n1} & ... & x_{nm} \end{vmatrix}, \qquad (14.1)$$

де X_j – вектор значень j-ї змінної ($j = 1, ..., m$), m – кількість змінних,

x_{ij} – значення j-ї змінної для i–го об'єкта ($i=1,...,n$), n – кількість об'єктів.

Неохідно знайти p нових змінних – факторів F_k ($k=1,...,p$, $p<m$), які беруть участь у поданні всіх змінних X_j і виражають внутрішні властивості досліджуваного явища.

Основними результатами факторного аналізу є набори факторних навантажень і факторних ваг.

Факторні навантаження (англ. *Factor Loadings*) – це значення коефіцієнтів кореляції кожної з вихідних змінних із кожним виявленим фактором. Вони можуть бути представлені у вигляді ***матриці факторних навантажень***, у якій рядкам відповідають вихідні змінні, а стовпцям – фактори:

$$W=\begin{vmatrix} \omega_{11} & \dots & \omega_{1p} \\ \dots & \dots & \dots \\ \omega_{m1} & \dots & \omega_{mp} \end{vmatrix}, \tag{14.2}$$

де ω_{jk} – факторне навантаження j-ї змінної та k-го фактора, яке характеризує тісноту їх зв'язку ($j=1,...,m$, $k=1,...,p$), m – кількість змінних, p – кількість факторів.

Значення факторного навантаження знаходиться в межах [-1; 1]. Додатні значення факторного навантаження вказують на прямий, а від'ємні – на зворотний зв'язок змінної з фактором. Чим тісніший зв'язок змінної з фактором, тим більшим є значення факторного навантаження за модулем.

Аналіз матриці факторних навантажень дає можливість сформулювати висновки про відносну вагу окремої змінної у структурі кожного фактора. Для кожного фактора виявляють та пов'язують із ним ті змінні, з якими він корелює більше всього.

Факторні ваги (англ. *Factor Scores*) – це значення виявлених факторів для кожного з об'єктів набору даних, які можуть бути представлені у вигляді ***матриці факторних ваг***, де рядкам відповідають об'єкти, а стовпцям – фактори:

$$F=\left|F_1,...,F_p\right|=\begin{vmatrix} f_{11} & \dots & f_{1p} \\ \dots & \dots & \dots \\ f_{n1} & \dots & f_{np} \end{vmatrix}, \tag{14.3}$$

де F_k – вектор значень *k*-го фактора ($k=1,...,p$), p – кількість факторів,

f_{ik} – факторна вага: значення *k*-го фактора для i–го об'кта ($i=1,...,n$), n – кількість об'єктів.

Дані про факторні ваги визначають ранжування об'єктів за кожним фактором. Об'єкт із більшим значенням факторної ваги певного фактора має більший ступінь прояву властивостей, обумовлених цим фактором.

Різні методи факторного аналізу використовують різні способи визначення факторних навантажень та факторних ваг.

Серед факторів виділяють:

1) ***загальні фактори*** – латентні макропараметри, що беруть участь у поданні всіх змінних набору даних;

2) ***характерні фактори*** – є специфічними тільки для однієї, «своєї» змінної набору даних.

Виявлення загальних факторів передбачає знаходження залежності між ними та змінними досліджуваного набору даних. Передбачається, що значення вихідних вимірюваних ознак знаходяться у лінійній залежності від виявлених факторів.

Основним об'єктом дослідження є ***кореляційна матриця***, яку використовують для отримання матриці факторних навантажень. Виділення загальних факторів супроводжується стисненням інформації, яке полягає у перетворенні вихідної кореляційної матриці розмірністю $m\times m$ на матрицю факторних навантажень меншої розмірності $m\times p$.

14.1.3. Основні методи та моделі факторного аналізу

До ***основних методів*** факторного аналізу відносять метод головних компонент, метод головних факторів, метод максимальної правдоподібності, центроїдний метод.

Усі ці методи передбачають, що значення вихідних змінних набору даних аналізованої предметної області представлені у числових шкалах, не мають викидів і підпорядковуються багатовимірному нормальному розподілу, а досліджувана залежність факторів зі змінними є лінійною. Необхідно зазначити, що останнім часом було розроблено також методи факторизації категоріальних ознак.

В основі кожного методу факторного аналізу лежить математична модель, яка описує співвідношення між вихідними змінними і загальними факторами.

Кожну змінну можна представити як лінійну комбінацію загальних та характерних факторів:

$$\tilde{X}_j = \omega_{j1}F_1 + ... + \omega_{jp}F_p + U_j, \quad (14.4)$$

де $\tilde{X}_j$ – стандартизоване значення j-ї змінної ($j = 1,...,m$),

F_k – значення k-го загального фактора ($k = 1,..., p$),

ω_{jk} – факторне навантаження, яке характеризує тісноту зв'язку k-го загального фактора та j-ї змінної,

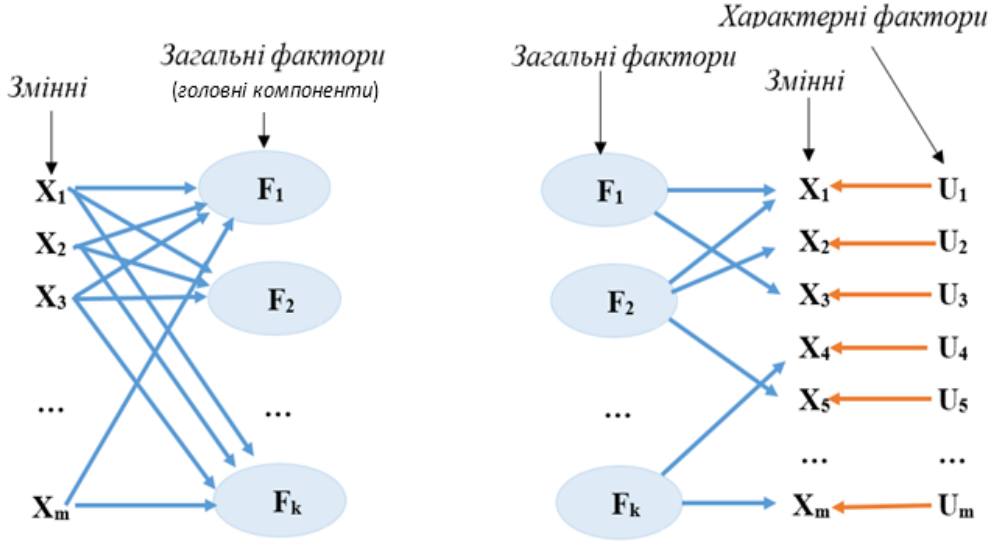
U_j – значення j-го характерного фактора j-ї змінної.

Для знаходження факторних навантажень частіше всього використовують метод головних компонент та метод головних факторів. Метод головних компонент виявляється кращим як метод скорочення даних, у той час як метод головних факторів краще застосовувати з метою відтворення структури взаємозв'язків між змінними.

Різницю у підходах до побудови моделі факторного аналізу за методами головних компонент та головних факторів схематично зображено на рисунку 14.1.

У моделі головних компонент кожна змінна впливає на кожну виділену компоненту – загальний фактор. Цей метод доволяє з'ясувати, як оптимальним чином скоротити простір ознак набору даних шляхом підбору оптимальної кількості компонент (загальних факторів). Факторні навантаження дозволяють для кожної головної компоненти підкреслити деякі змінні більше, інші – менше.

Модель головних факторів націлена на вимірювання прихованої змінної, яку неможливо виміряти. Кожен загальний фактор розглядається через зв'язки, які він викликає у змінних набору даних, а факторні навантаження оптимальним чином цей зв'язок описують.



а) модель головних компонент *б*) модель головних факторів

Рис. 14.1. Порівняння моделей головних компонент і головних факторів

Метод головних компонент (англ. *Principal Component Analysis, PCA*) є найпоширенішим методом виявлення факторів, який припускає, що характерні фактори не корелюють із загальними факторами, а загальні фактори також розглядаються як незалежні нормально розподілені стандартизовані показники.

З математичної точки зору метод головних компонент є ортогональним лінійним перетворенням, яке відображає дані з вихідного простіру ознак у новий простір меншої розмірності. Основна ідея пошуку заснована на припущенні, що чим більша дисперсія вздовж якоїсь осі, тим більше інформації містить значення проекцій на цю

вісь. Шукають вісь із максимальною дисперсією, яка й є першим фактором – першою головною компонентою (рис. 14.2).

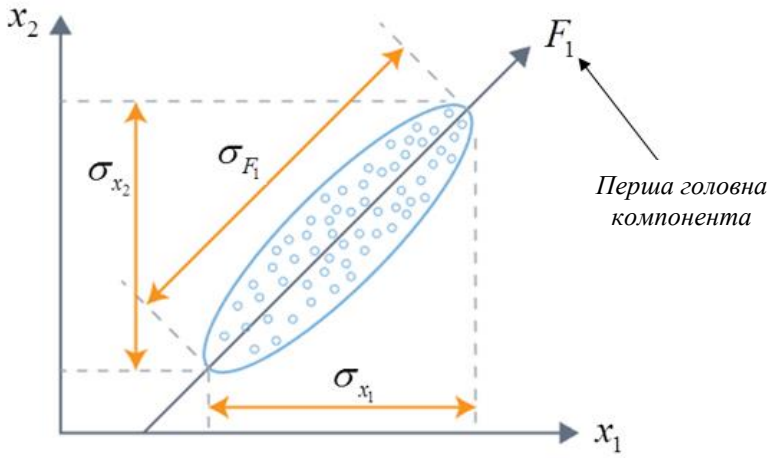


Рис. 14.2. Зменшення розмірності двовимірного простору ознак за допомогою методу головних компонент

Пошук системи взаємно перпендикулярних осей (є припущення про їх незалежність) зводиться до послідовної процедури знаходження такої кількості факторів, які пояснюють якомога більшу частку дисперсії спостережуваних змінних. Для першого фактора F_1 беруть пряму, що проходить через центр координат та діаграму розсіювання: вісь обирають так, щоб сума квадратів відстаней усіх точок до перпендикуляра до цієї прямої була максимальна (ця вісь пояснює максимум дисперсії). Якщо діаграма розсіювання має форму еліпса, то фактор F_1 співпадає з напрямом, у якому об'єкти витягнуті.

Для другого фактора F_2 шукають вісь, перпендикулярну першому фактору: ця вісь пояснює найбільшу частину тієї дисперсії, яка не була пояснена першою віссю. Аналогічно – далі, якщо змінних більше.

На основі матриці вихідних даних X будують матрицю стандартизованих значень $\tilde{X}$ розмірністю $n \times m$ (рядки відповідають об'єктам, стовпці – ознакам набору даних), кожен елемент якої розраховують за формулою:

$$\tilde{x}_{ij} = \frac{x_{ij} - \bar{x}_j}{\sigma_{x_j}}, \tag{14.5}$$

де x_{ij} – значення j -ї змінної для i –го об'єкта, $i = 1,...,n$, n – кількість об'єктів,

$\bar{x}_j$ – середнє значення j -ї змінної, $j = 1,...,m$, m – кількість змінних,

σ_{x_j} – середньоквадратичне відхилення j -ї змінної,

$\tilde{x}_{ij}$ – стандартизоване значення j -ї змінної для i –го об'єкта.

Знаходження матриці стандартизованих значень змінних дозволяє визначити елементи кореляційної матриці *R*. Матриця парних коефіцієнтів кореляції *R* розмірністю $m \times m$ є симетричною відносно головної діагоналі, а кожен її елемент r_{ij} є коефіцієнтом кореляції Пірсона між i -ю та j -ю змінними, рівним одиниці у випадку, коли $i = j$. Тому на головній діагоналі матриці розміщені елементи, які дорівнюють 1.

У матричному вигляді кореляційну матрицю *R* можна представити таким чином:

$$R = \frac{1}{n} \tilde{X}^T \tilde{X}. \tag{14.6}$$

Будуючи факторну модель за методом головних компонент припускають, що стандартизовані змінні $\tilde{X}_j$ є лінійною комбінацією загальних факторів F_k – ***головних компонент***:

$$\tilde{X}_j = \sum_{k=1}^{p} \omega_{jk} F_k \,, \tag{14.7}$$

де ω_{jk} – факторне навантаження j-ї змінної і *k*-го загального фактора.

У матричному вигляді модель головних компонент буде мати вигляд:

$$\tilde{X} = FW^T \,, \tag{14.8}$$

де $\tilde{X}$ – матриця стандартизованих значень вихідних змінних, W – матриця факторних навантажень, F – матриця значень головних компонент – факторних ваг.

Матриця F описує n об'єктів у просторі m головних компонент. При цьому елементи матриці стандартизовані – середнє значення кожної компоненти рівне нулю, дисперсія рівна одиниці. А головні компоненти не корелюють між собою. Тому:

$$\frac{1}{n} F^T F = E \,, \tag{14.9}$$

де E – одинична матриця розмірністю $m \times m$.

Оскільки змінні $\tilde{X}_j$ і головні компоненти F_k стандартизовані, елементи матриці W є коефіцієнтами кореляції між ними – факторними навантаженнями.

Побудова моделі головних компонент націлена на знаходження такої кількості факторів, які пояснюють якомога більшу частку дисперсії спостережуваних параметрів.

За умови ортогональності компонент квадрат факторного навантаження ω_{jk}^2 характеризує внесок *k*-ї компоненти у дисперсію j-ї змінної. Змінна $\tilde{X}_j$ має одиничну дисперсію, яка рівна сумі квадратів факторних навантажень кожної компоненти і цієї змінної:

$$\sigma_j^2 = \sum_{k=1}^{m} \omega_{jk}^2 = 1 \,. \tag{14.10}$$

Повний внесок *k*-ї компоненти F_k у сумарну дисперсію всіх m змінних становить:

$$\sigma_k^2 = \sum_{j=1}^{m} \omega_{jk}^2 \,. \tag{14.11}$$

Оскільки значення змінних стандартизовані і мають одиничну дисперсію, сумарну дисперсію змінних набору даних можна представити як суму дисперсій усіх компонент σ_k^2, рівну кількості змінних:

$$\sigma^2 = \sum_{k=1}^{m} \sigma_k^2 = \sum_{k=1}^{m} \sum_{j=1}^{m} \omega_{jk}^2 = m \,. \tag{14.12}$$

Кореляційна матриця *R* може бути представлена через матрицю факторних навантажень W:

$$R = \frac{1}{n} \tilde{X}^T \tilde{X} = \frac{1}{m} (FW^T)^T FW^T = W(\frac{1}{n} F^T F)W^T = WEW^T \,,$$

$$R = WW^T \,. \tag{14.13}$$

Оскільки матриця *R* є симетричною, існує така ортогональна матриця V, для якої виконується:

$$V^T RV = \Lambda \,, \tag{14.14}$$

де $\Lambda = \begin{vmatrix} \lambda_1 & \dots & 0 \\ \dots & \lambda_k & \dots \\ 0 & 0 & \lambda_m \end{vmatrix}$ – діагональна матриця характеристичних чисел λ_k,

$V = |V_1,...V_k,...,V_m| = \begin{vmatrix} v_{11} & \dots & v_{m1} \\ \dots & \dots & \dots \\ v_{1m} & \dots & v_{mm} \end{vmatrix}$ – ортогональна матриця власних векторів V_k, які відповідають характеристичним числам λ_k.

Пошук головних компонент зводиться до знаходження характеристичних чисел λ_k кореляційної матриці *R* та відповідних їм власних векторів V_k.

Власний вектор матриці – такий вектор, при множенні матриці на який отримується цей же вектор, помножений на певне число, яке називають ***характеристичним числом*** (власним):

$$RV_k = \lambda_k V_k \,, \tag{14.15}$$

де V_k – власні вектори матриці, λ_k – характеристичні числа.

Характеристичні числа знаходять як корені характеристичного рівняння:

$$|\Lambda E - R| = 0 \,. \tag{14.16}$$

Власний вектор V_k, який відповідає характеристичному числу λ_k кореляційної матриці *R*, визначається як відмінний від нуля розв'язок рівняння:

$$(\lambda_k E - R)V_k = 0 \,. \tag{14.17}$$

Оскільки сума діагональних елементів матриці *R* рівна m, маємо:

$$\sum_{k=1}^{m} \lambda_k = m \,. \tag{14.18}$$

Знайдені характеристичні числа будуть рівні:

$$\lambda_k = \sum_{j=1}^{m} \omega_{jk}^2 \,. \tag{14.19}$$

Порівнявши отриманий результат із формулою 14.10, можемо стверджувати, що кожне характеристичне число λ_k характеризує вклад *k*-ї компоненти F_k у сумарну дисперсію всіх m змінних:

$$\lambda_k = \sigma_k^2 = \sum_{j=1}^{m} \omega_{jk}^2 \,. \tag{14.20}$$

Отже, сумарну дисперсію змінних набору даних можна представити як суму дисперсій усіх компонент, які є характеристичними числами кореляційної матриці, що виражаються через факторні навантаження:

$$\sigma^2 = \sum_{k=1}^{m} \lambda_k = \sum_{k=1}^{m} \sum_{j=1}^{m} \omega_{jk}^2 = m \,. \tag{14.21}$$

Питомий вклад кожної компоненти у загальну дисперсію визначають за формулою $\frac{\lambda_k}{m} \cdot 100\%$. Тоді загальний вклад p головних компонент показує відсоток поясненої дисперсії та відповідно втраченої інформації й характеризує ***повноту факторизації***::

$$\frac{1}{m} \sum_{k=1}^{p} \lambda_k \cdot 100\% \,. \tag{14.22}$$

Зазвичай обирають таку кількість головних компонент, вклад яких у сумарну дисперсію є вищим за 60-70%.

Після знаходження власних векторів та характеристичних чисел, які їм відповідають, факторні навантаження можуть бути розраховані за формулою, яка у матричному вигляді є наступною:

$$W = V\Lambda^{-0,5} \,, \tag{14.23}$$

де Λ – діагональна матриця характеристичних чисел,

W – матриця факторних навантажень, V – матриця власних векторів.

Знайдені значення факторних навантажень дозволяють розрахувати значення усіх факторів для кожного об'єкта набору даних:

$$f_{ik} = \frac{1}{\lambda_k} \sum_{j=1}^{m} \omega_{jk} \cdot \tilde{x}_{ij} , \tag{14.24}$$

де f_{ik} – значення *k*-го фактора для i –го об'кта ($i = 1,...,n$),

$\tilde{x}_{ij}$ – стандартизоване значення j -ї змінної для i –го об'єкта ($j = 1,...,m$),

λ_k – характеристичне число, яке відповідає *k*-му фактору ($k = 1,..., p$),

n – кількість об'єктів, m – кількість змінних, p – кількість факторів.

Таким чином, для виявлення факторів необхідно знайти характеристичні числа кореляційної матриці, ранжувати їх у спадному порядку та визначити ті з них, які вносять найбільший вклад у сумарну дисперсію. Завичай кількість таких компонент p буде суттєво меншою за кількість змінних набору даних m . Саме такі компоненти називають ***головними компонентами*** – зальними факторами, які підлягають інтерпретації.

Метод головних факторів або факторизація головної осі (англ. *Principal Axis Factoring, PAF*) припускає, що характерні фактори не корелюють не тільки з загальними факторами, але і між собою. Побудова факторної моделі націлена на пояснення коефіцієнтів кореляції між спостережуваними параметрами.

Якщо усі ознаки X_j стандартизовані (математичне сподівання рівне 0, дисперсія рівна 1), а загальні фактори F_k – незалежні і не пов'язані з характерними факторами U_j, то факторні навантаження ω_{jk} співпадають із коефіцієнтами кореляції між загальними факторами і стандартизованими змінними $\tilde{X}_j$. Тоді загальну дисперсію $\sigma^2_{\tilde{X}_j}$ змінної $\tilde{X}_j$ можна розкласти на дві складові:

$$\sigma_j^2 = \sum_k \omega_{jk}^2 + \sigma_{U_j}^2 , \tag{14.25}$$

де $\sum_k \omega_{jk}^2$ – ***спільність***: частина дисперсії, обумовлена загальними факторами й рівна сумі квадратів факторних навантажень;

$\sigma_{U_j}^2$ – ***специфічність***: частина дисперсії, обумовлена характерними факторами, специфічними для цієї змінної.

При побудові моделі факторного аналізу шукають такі фактори, для яких сумарна спільність є максимальною, а специфічність – мінімальною.

Основна відмінність від методу головних компонент полягає у тому, що у методі головних факторів у кореляційній матриці одиниці на головній діагоналі замінюються оцінками спільностей. Визначення значень характеристичних чисел λ_k та відповідних їм власних векторів кореляційної матриці здійснюється із використанням оціночних значень її діагональних елементів – дисперсій, обумовлених загальними факторами.

14.1.4. Етапи факторного аналізу даних за методом головних компонент. Обертання факторів

1. ***Здійснення стандартизації*** матриці вхідних даних X шляхом побудови центрованої матриці $\tilde{X}$, кожен елемент якої розраховують шляхом відніманням від кожного елемента матриці даних середнього значення відповідної змінної, та ділення знайденої різниці на середньоквадратичне відхилення цієї змінної (формула 14.5).

Після цього середнє значення кожної змінної дорівнює нулю, а дисперсія – одиниці. Тоді загальна дисперсія стандартизованих ознак рівна сумі дисперсій змінних – числу змінних, бо кожна дисперсія дорівнює 1.

2. ***Побудова кореляційної матриці*** *R* шляхом розрахунку попарних коефіцієнтів кореляції Пірсона між відповідними змінними набору даних.

3. ***Визначення значень характеристичних чисел*** λ_k та відповідних їм власних векторів кореляційної матриці.

4. ***Визначення кількості головних компонент – факторів***, достатньої для оптимального представлення даних. Із послідовності характеристичних чисел λ_k обирається p максимальних, які вносять найбільший вклад у загальну дисперсію.

Для визначення оптимальної кількості факторів застосовують спеціальні критерії:

a) ***формальні критерії***: зберігають ті фактори, значення характеристичних чисел яких перевищують 1 (або інше, задане аналітиком число, наприклад, 1,25);

b) ***критерій Кеттелла***: критерій «кам'янистого осипу» – шукають точку, де убування значень характеристичних чисел уповільнюється найбільш сильно.

Остаточне рішення про оптимальну кількість факторів приймається після інтерпретації факторів.

5. ***Обертання факторів*** для пошуку однозначного розв'язку задачі визначення факторів. Необхідність обертання факторів виникає тоді, коли виявленим факторам не вдається дати досить чітку змістовну інтерпретацію.

Пошук простої факторної структури здійснюється за допомогою ортогонального або косокутного обертання. Геометрично ця процедура означає обертання системи координат із метою виявлення напрямків, уздовж яких вихідні змінні максимально змінюються. Обертання дозволяє зробити матрицю факторних навантажень більш контрастною за рахунок збільшення навантажень за одними ознаками і зменшення за іншими. Це сприяє більш виразному виявленню груп ознак, які визначають той або інший фактор.

Є велика кількість методів обертання, найбільш часто вживаним є ортогональне обертання за методом Варімакс (англ. *Varimax*), яке максимізує дисперсії факторних навантажень, роблячи великі значення факторних навантажень більшими, а малі – меншими (рис. 14.3).

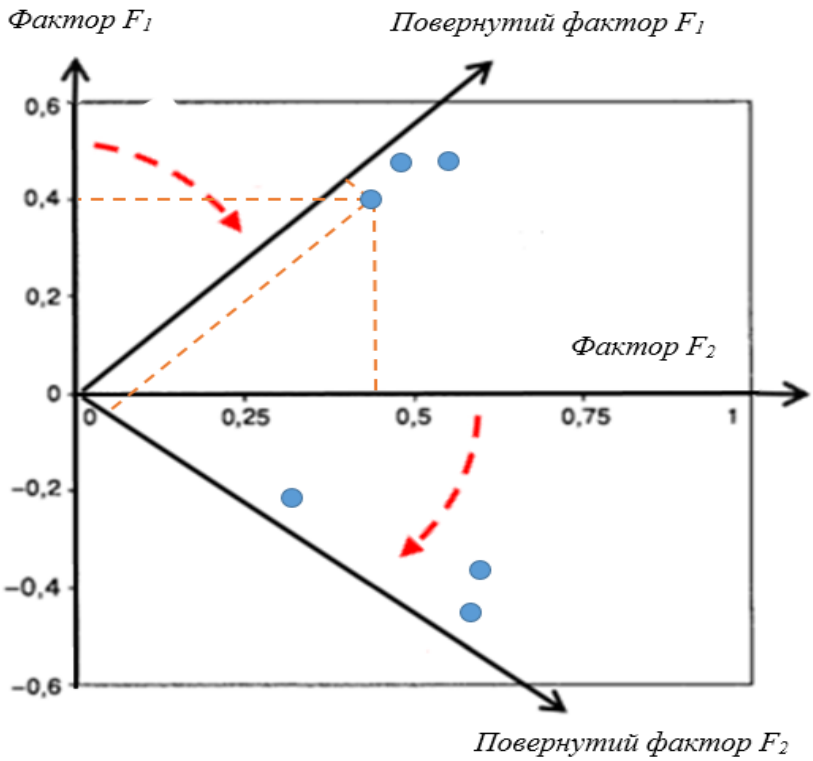


Рис. 14.3. Обертання факторів для виявлення простої факторної структури

Факторні навантаження поверненої матриці розглядають як результат виконання процедури факторного аналізу.

6. ***Інтерпретація виявлених факторів***. Якщо фактори знайдені й витлумачені, то на останньому кроці факторного аналізу окремим об'єктам можна привласнити значення цих факторів – факторні ваги. У такий спосіб кількість змінних, якими представлені об'єкти набору даних, буде зменшена до виявленої кількості факторів, які можна буде використовувати у подальшому аналізі досліджуваної предметної області.

14.1.5. Критерії факторного аналізу

Під час проведення факторного аналізу використовуються такі критерії.

За ***критерієм Кайзера*** головними компонентами вважають такі, для яких характеристичні числа λ_k є більшими за одиницю, а повнота факторизації є не меншою за 70%. За змістом це означає, що цим факторам відповідає дисперсія принаймі однієї змінної.

Критерій адекватності вибірки Кайзера–Мейєра–Олкіна (КМО) використовується для оцінки застосовності факторного аналізу до набору вихідних змінних шляхом перевірки того, настільки повно кореляцію між змінними набору даних можна пояснити іншими змінними – факторами. Значення критерію КМО від 0,5 до 1 свідчать про адекватність факторного аналізу. А значення, менші за 0,5, вказують на те, що факторний аналіз не може бути застосований.

Критерій сферичності Бартлетта є критерієм багатомірної нормальності розподілу та перевіряє гіпотезу про відсутність кореляцій між змінними набору даних. Значення критерію, менше за обраний рівень значущості 0,05 (або 0,01 чи 0,1), вказує на те, що дані цілком прийнятні для проведення факторного аналізу. При більших значеннях критерію гіпотеза відхиляється, тоді застосування факторного аналізу не є доцільним.

14.2. ПРОВЕДЕННЯ ФАКТОРНОГО АНАЛІЗУ В SPSS

Завдання 1. Є набір даних, який містить показники, що характеризують поведінку студентів на заняттях, представлені сукупністю семи змінних, кожна з яких оцінена у 7-бальній шкалі. Для проведення подальшого аналізу необхідно виявити нові змінні – фактори, кількість яких є меншою за кількість вихідних змінних, що пояснюють спостережувані зв'язки між наявними змінними.

Емпіричні дані (табл. 14.1) містяться у файлі *factors.sav*.

Таблиця 14.1

Дані аналізу поведінки студентів на заняттях

Характеристика	№ студента																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Швидкість	3	1	4	3	2	1	6	1	3	1	2	2	2	7	7	4	1	4	6	7
Поведінка	7	5	1	4	4	2	6	6	4	6	5	4	2	4	7	4	3	2	6	6
Активність	3	1	4	4	2	1	7	1	4	1	1	2	1	6	6	5	1	5	6	6
Відсутність	1	7	2	7	7	5	6	3	1	2	6	2	2	7	6	1	2	6	3	3
Уважність	1	4	7	3	5	6	1	2	3	2	4	6	5	5	2	5	4	7	1	2
Впевненість	4	1	4	4	1	1	7	2	4	2	1	1	2	6	7	4	2	4	6	7
Мотивація	1	3	7	3	5	7	2	2	3	2	5	5	7	5	1	3	4	7	2	2

Зауваження. Для факторного аналізу було використано дані 20 студентів за 7 ознаками. Це невеликий за обсягом набір даних, відносно до якого факторний аналіз може застосовуватися тільки з навчальною метою. У цілому факторний аналіз застосовують для аналізу наборів даних із великою кількістю об'єктів та вимірюваних змінних.

14.2.1. Постановка задачі дослідження

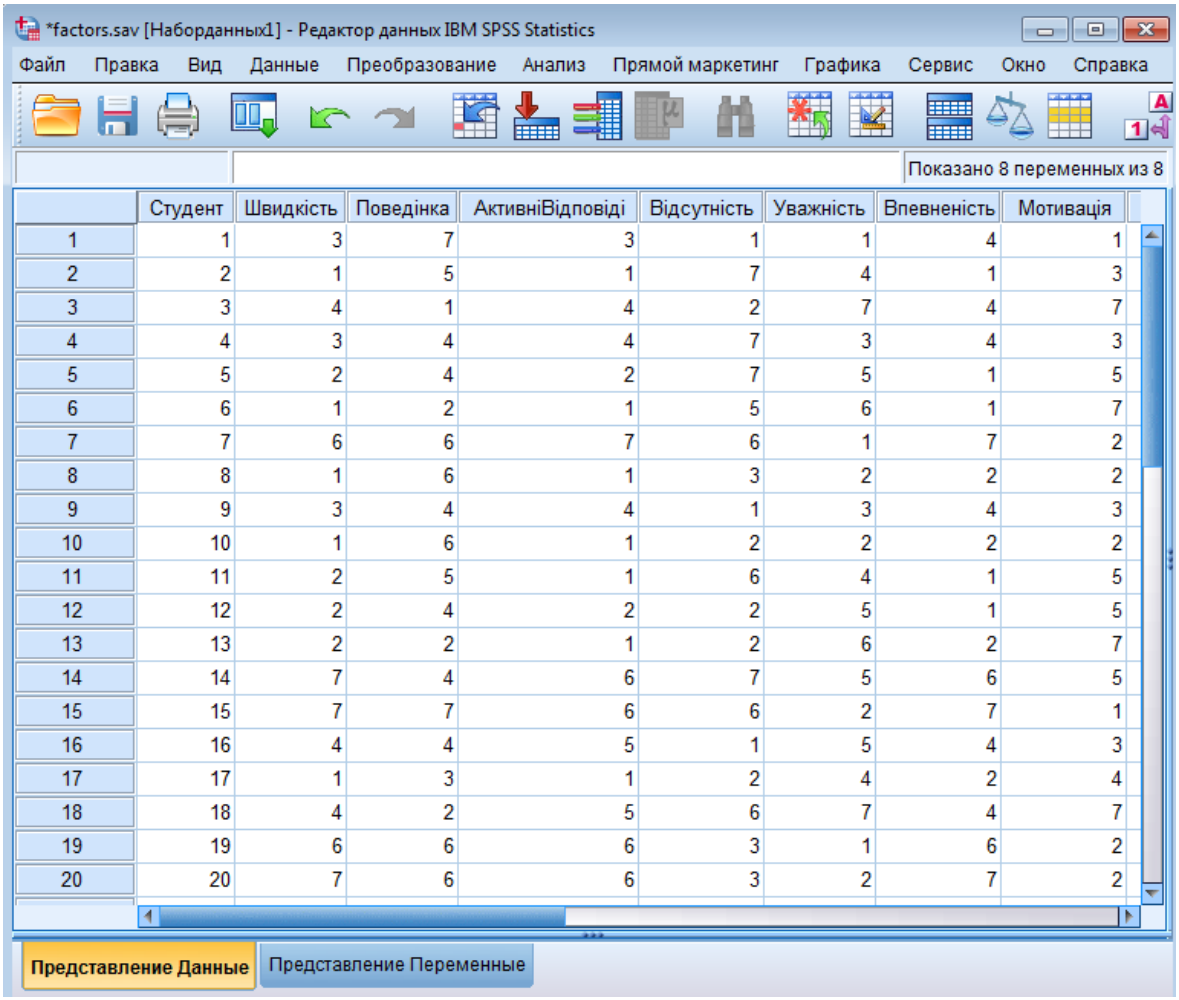
1. Набір даних містить інформацію про активність студентів на заняттях і складається з 7 змінних, кожна з яких може приймати не менше ніж 7 значень. Із метою спрощення процедури подальшого аналізу даних (наприклад, кластерного) доцільно провести факторний аналіз, що дозволить скоротити число досліджуваних змінних та оптимізувати структуру даних.

2. Для розв'язання поставленої задачі необхідно провести дослідницький факторний аналіз, оскільки метою є скорочення простору ознак та виявлення їх факторної структури. Для проведення аналізу буде використано метод головних компонент.

14.2.2. Налаштування параметрів здійснення факторного аналізу засобами SPSS

1. У вікні редактора даних *SPSS* відкриваємо файл із вхідним набором даних *factors.sav* (рис. 14.3).

2. На панелі інструментів необхідно обрати меню *Аналіз/Analyze – Зниження розмірності/Data Reduction – Факторний аналіз/Factor*. Відкриється діалогове вікно *Факторний аналіз/Factor Analysis* (рис. 14.4).



	Студент	Швидкість	Поведінка	АктивніВідповіді	Відсутність	Уважність	Впевненість	Мотивація
1	1	3	7	3	1	1	4	1
2	2	1	5	1	7	4	1	3
3	3	4	1	4	2	7	4	7
4	4	3	4	4	7	3	4	3
5	5	2	4	2	7	5	1	5
6	6	1	2	1	5	6	1	7
7	7	6	6	7	6	1	7	2
8	8	1	6	1	3	2	2	2
9	9	3	4	4	1	3	4	3
10	10	1	6	1	2	2	2	2
11	11	2	5	1	6	4	1	5
12	12	2	4	2	2	5	1	5
13	13	2	2	1	2	6	2	7
14	14	7	4	6	7	5	6	5
15	15	7	7	6	6	2	7	1
16	16	4	4	5	1	5	4	3
17	17	1	3	1	2	4	2	4
18	18	4	2	5	6	7	4	7
19	19	6	6	6	3	1	6	2
20	20	7	6	6	3	2	7	2

Рис. 14.3. Вхідні дані у вікні редактора даних SPSS

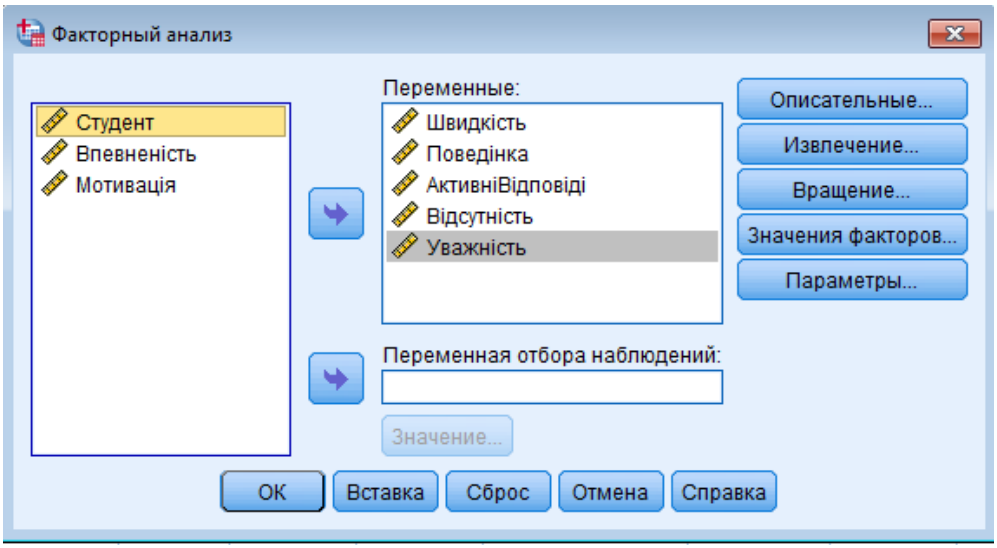


Рис. 14.4. Вікно Факторний аналіз / Factor Analysis

3. У правому полі вікна представлений список усіх змінних, які аналізуються. Із цього поля необхідно обрати масив тих змінних, які беруть участь у факторному аналізі, й перенести їх у поле *Змінні/Variables*. Відбираємо змінні: *Швидкість*, *Поведінка*, *Активні відповіді*, *Відсутність*, *Уважність*, *Впевненість*, *Мотивація*.

4. У полі *Вибір змінних спостережень/Selection Variable* вказують змінні для розбивки аналізованих даних на підгрупи. Наприклад, якщо в це поле помістити змінну «*Стать*», факторний аналіз буде проводитися по двох

масивах даних – окремо для чоловіків і жінок. У задачі, яка розв'язується, поділ аналізованих даних на підгрупи не здійснюється.

5. Для виявлення доцільності проведення факторного аналізу на аналізованому наборі даних необхідно натиснути кнопку *Описові/Descriptives*.

6. У вікні, що відкриється, – *Факторний аналіз: Описові/Factor Analysis: Descriptives* (рис. 14.5) здійснюють налаштування параметрів:

a) у полі *Статистики* необхідно обрати *Початкове рішення*;

b) у полі *Кореляційна матриця* для відображення коефіцієнтів кореляції установлюють прапорець *Коефіцієнти* та *КМО і Критерій сферичності Бартлетта* – для виявлення застосовності факторного аналізу до аналізованого набору даних.

Після вибору параметрів необхідно натиснути кнопку *Продовжити*.

7. Для вибору методу виявлення факторної структури змінних набору даних та визначення їх оптимальної кількості необхідно натиснути кнопку *Вилучення/Extraction*. Відкриється вікно *Факторний аналіз: Виділення факторів/Factor Analysis: Extraction Factors* (рис. 14.6).

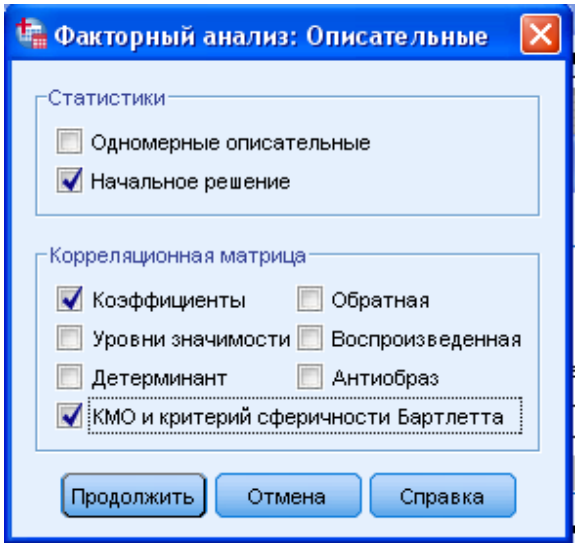


Рис. 14.5. Вікно Факторний аналіз: Описові/Factor Analysis: Descriptives

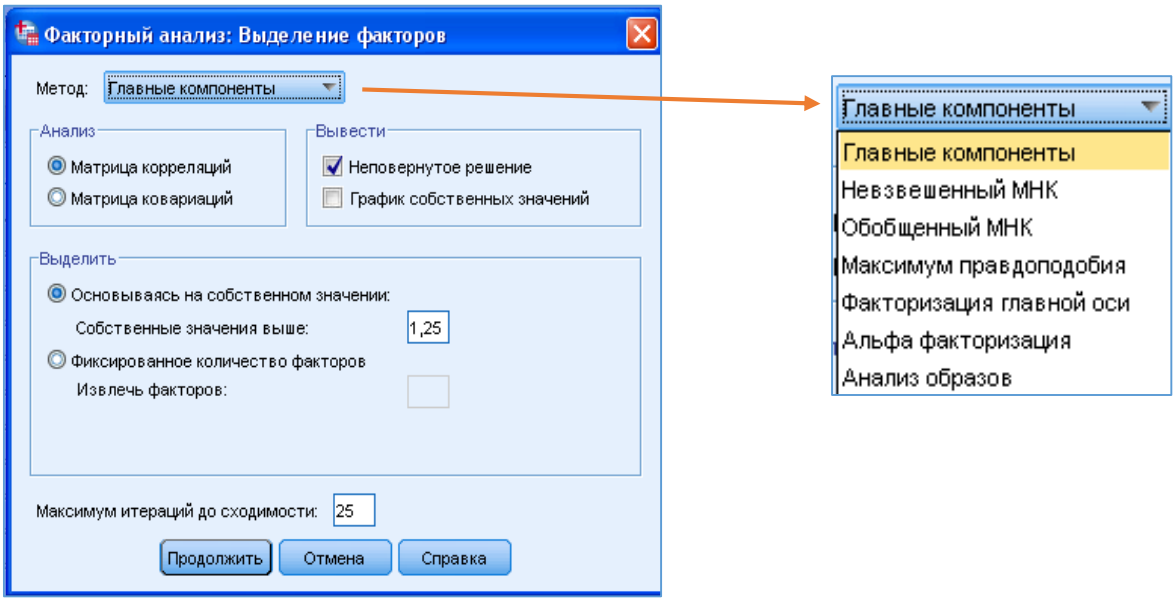


Рис. 14.6. Вікно Факторний аналіз: Виділення факторів/Factor Analysis: Extraction Factors

8. У вікні *Факторний аналіз: Виділення факторів*/ *Factor Analysis: Extraction Factors* здійснюють налаштування:

a) зі списку, що розкривається, обирають метод факторного аналізу: необхідно залишити метод *Головні компоненти* (він установлений по замовчуванню);
b) у полі *Аналіз* вказують *Матриця кореляцій* для виведення її на екран;
c) у полі *Вивести* необхідно прибрати прапорець виведення неповернутого рішення та установити прапорець виведення *Графіка власних значень*;
d) у полі *Виділити* можна задати оптимальну кількість факторів вручну або вказати визначення на основі значень характеристичних чисел: обираємо цю опцію та задаємо, що значення характеристичних чисел повинно бути більше за 1.

Після налаштування вказаних параметрів натискаємо кнопку *Продовжити*.

9. Далі здійснюють налаштування обертання факторів. Для цього необхідно натиснути кнопку *Обертання*. Відкриється вікно *Факторний аналіз: Обертання*/*Factor Analysis: Rotation* (рис. 14.7), у якому вказують:

a) метод обертання факторів – *Варимакс*;
b) виведення на екран повернутої матриці факторних навантажень та графіка навантажень, установивши прапорці *Повернуте рішення* та *Графіки завантажень*.

Після налаштування вказаних параметрів необхідно натиснути кнопку *Продовжити*.

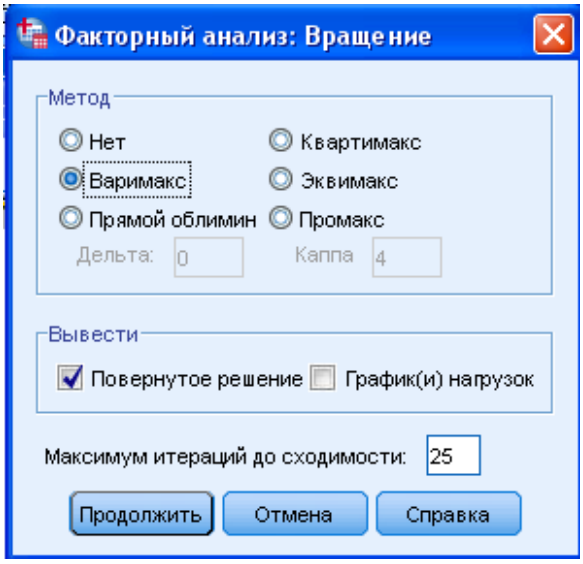


Рис. 14.7. Вікно Факторний аналіз: Обертання/Factor Analysis: Rotation

10. Перед проведенням факторного аналізу необхідно задати створення нових змінних, пов'язаних із виявленими факторами. Для цього натискають кнопку *Значення факторів* й у вікні, що відкриється (рис. 14.8), відмітити *Зберегти як змінні*, метод розрахунку нових змінних залишаємо той, який установлено по замовчуванню – *Регресія*. Після зроблених налаштувань необхідно натиснути кнопку *Продовжити*.

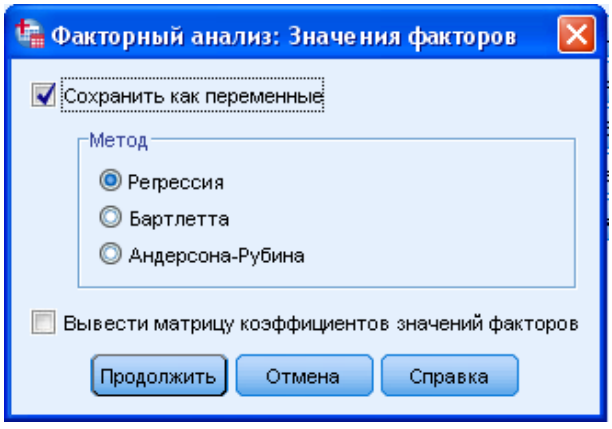


Рис. 14.8. Вікно Факторний аналіз: значення факторів

11. Для запуску процедури виконання факторного аналізу у діалоговому вікні *Факторний аналіз/Factor Analysis* необхідно натиснути кнопку *ОК*. У вікні виведення буде виведено результат факторного аналізу відповідно до зроблених налаштувань.

14.2.3. Аналіз критеріїв факторного аналізу

1. У першій таблиці, виведеній на екран після запуску процедури факторного аналізу, відображено розраховані значення критеріїв КМО й Бартлетта (рис. 14.9), які дозволяють оцінити адекватність проведення факторного аналізу на заданому наборі даних.

КМО и критерий Бартлетта

Мера адекватности выборки Кайзера-Майера-Олкина (КМО).		,584
Критерий сферичности Бартлетта	Примерная Хи-квадрат	168,810
	ст.св.	21
	Знач.	,000

Рис. 14.9. Таблиця значень критерію КМО та критерію Бартлетта

2. Критерій КМО надає можливість визначити, наскільки повно побудована факторна модель описує структуру характеристик студентів, представлених змінними аналізованого набору даних. Він може приймати значення, представлені в інтервалі від нуля – факторна модель абсолютно незастосовна, до одиниці – факторна модель ідеально описує структуру змінних набору даних. Результати факторного аналізу можуть вважатися дійсними, якщо значення критерію КМО більше 0,5.

Розраховане для заданого набору даних значення критерію КМО рівне 0,584, що свідчить про прийнятність побудованої факторної моделі.

3. Критерій Бартлетта перевіряє гіпотезу про відсутність кореляційної залежності між змінними досліджуваного набору даних. Виявлено, що розрахована значущість цієї гіпотези становить 0,000 (рис. 14.9). Це означає, що нульова гіпотеза може бути відхилена й прийнята альтернативна гіпотеза – кореляційні зв'язки між змінними існують. Тому можливе їх групування на підставі тісноти кореляцій.

4. Розраховані значення критеріїв КМО та Бартлетта дозволяють зробити висновок про придатність заданого набору даних для проведення факторного аналізу.

14.2.4. Виявлення кореляційної залежності

1. У результаті проведеного факторного аналізу було побудовано кореляційну матрицю, які містить попарні коефіцієнти кореляцій між змінними набору даних (рис. 14.10). Здійснимо аналіз розрахованих значень коефіцієнтів кореляцій.

Корреляционная матрица

		Швидкість	Поведінка	АктивніВідповіді	Відсутність	Уважність	Впевненість	Мотивація
Корреляция	Швидкість	1,000	,291	,945	,180	-,260	,942	-,275
	Поведінка	,291	1,000	,236	,031	-,911	,369	-,918
	АктивніВідповіді	,945	,236	1,000	,149	-,260	,942	-,299
	Відсутність	,180	,031	,149	1,000	,100	,057	,129
	Уважність	-,260	-,911	-,260	,100	1,000	-,433	,930
	Впевненість	,942	,369	,942	,057	-,433	1,000	-,442
	Мотивація	-,275	-,918	-,299	,129	,930	-,442	1,000

Рис. 14.10. Таблиця Кореляційна матриця

2. Значення коефіцієнта кореляції між змінною *Уважність* та змінною *Впевненість* дорівнює -0,433. Абсолютна величина коефіцієнта 0,433 свідчить про достатній ступінь взаємозв'язку між змінними. Знак «мінус» говорить про те, що ці змінні пов'язані оберненим зв'язком: впевненість у собі дещо заважає уважності. Тому при об'єднанні вони потрапляють у різні групи.

3. У одну групу мають бути об'єднані змінні, які мають високий ступінь прямого взаємозв'язку: наприклад, *Впевненість* і *Швидкість* – значення коефіцієнтів кореляції між змінними становить 0,942.

4. Досить слабо пов'язана з усіма іншими змінними змінна *Відсутність*. Цю змінну взагалі доцільно виключити з подальшого аналізу.

14.2.5. Визначення оптимального числа компонент на основі аналізу розрахованих характеристичних чисел

1. Оптимальне число компонент факторної моделі визначається за допомогою розрахованих характеристичних чисел. Значення цих показників наведено в таблиці *Пояснення сукупної дисперсії/Total Variance Explained*, що виводиться на екран серед інших результатів факторного аналізу (рис. 14.11). Здійснимо аналіз даних таблиці *Пояснення сукупної дисперсії*.

Объясненная совокупная дисперсия

Компонент	Начальные собственные значения			Суммы квадратов загрузок вращения		
	Всего	% дисперсии	Суммарный %	Всего	% дисперсии	Суммарный %
1	3,826	54,650	54,650	2,964	42,343	42,343
2	1,980	28,284	82,934	2,841	40,591	82,934
3	,962	13,742	96,676			
4	,108	1,542	98,218			
5	,078	1,108	99,326			
6	,037	,524	99,850			
7	,010	,150	100,000			

Метод выделения факторов: метод главных компонент.

Рис. 14.11. Таблиця Пояснення сукупної дисперсії/Total Variance Explained

2. У першому стовпці таблиці *Компонент/Component* вказується номер компоненти. У третьому стовпці *% дисперсії /Variance* вказано відсоток поясненої кожним компонентом дисперсії. Чим більшим є процент поясненої компонентою дисперсії, тим більш значимою вона є.

3. У четвертому стовпці таблиці *Сумарний %/Cumulative %* наведено відсоток інформації, збереженої в разі групування змінних за допомогою кількості компонент, яка рівна номеру компоненти (рядку таблиці) – сумарний відсоток поясненої дисперсії. Наприклад, якщо число факторів у факторній моделі дорівнює чотирьом, вихідна інформація буде збережена на 98,218%.

4. У другому стовпці таблиці *Всього/Total* наведено розраховані для кожної компоненти значення *Характеристичних чисел/Eigenvalues*. У проведеному факторному аналізі була задана умова: значення *Характеристичних чисел* має бути більшим за 1. Максимальне число компонент факторної моделі, у яких цей показник перевищує одиницю, становить 2. Це означає, що оптимальне число факторів у факторній моделі становить 2. Перший фактор пояснює 42,34% сумарної дисперсії, другий – 82,93%.

5. Як видно з отриманих даних, факторна модель, яка складається з двох факторів, зберігає 82,934% вхідної інформації. Враховуючи, що в ході факторного аналізу число змінних скоротиться в 3,5 рази (з 7 до 2), а втрата інформації становить менше 17%, використання побудованої факторної моделі є доцільним.

14.2.6. Визначення оптимального числа компонент на основі аналізу графіка значень характеристичних чисел

1. Оптимальне число компонент факторної моделі можна визначити за допомогою аналізу *Графіка власних значень/Screen plot* (рис. 14.12).

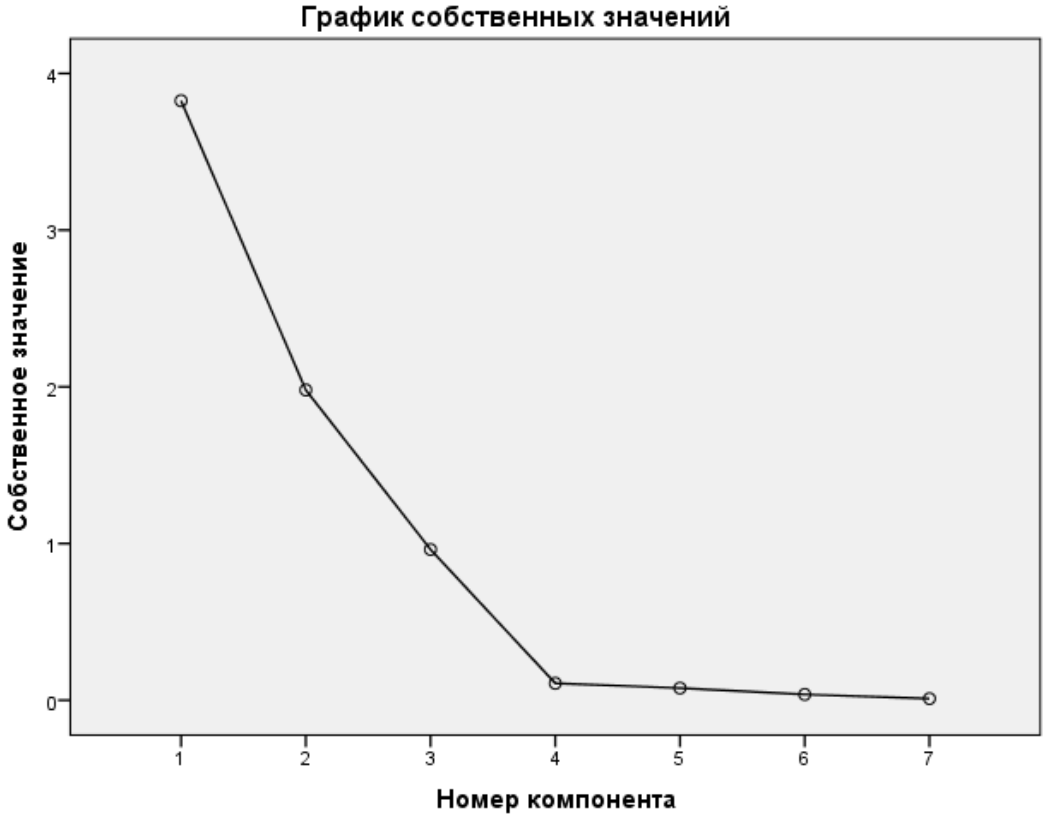


Рис. 14.12. Графічне визначення кількості компонент факторної моделі

2. Побудований графік відображає залежність між значеннями характеристичних чисел та номерами компонент, які їм відповідають. При зміні номера компоненти з 4 до 7 цей графік є практично лінійною функцією, а при зміні номера компоненти з 4 до 2 відбувається «перелом» графіка (*Графік кам'янистого осипу*). Включати в модель рекомендують таку кількість факторів, яка відповідає номеру компоненти у місці перелому графіка. Це означає, що оптимальне число компонент факторної моделі – факторів, дорівнює двом.

3. Результати графічного методу підтвердили результати розрахункового методу визначення ооптимальної кількісті факторів факторної моделі.

14.2.7. Побудова факторної моделі

1. Як результат факторного аналізу на екран виводиться таблиця з елементами повернутої матриці компонент факторної моделі – *Повернена матриця компонент/Rotated Component Matrix* (рис. 14.13). Аналіз її вмісту дозволяє пояснити виявлені фактори за ступенем факторного навантаження.

2. Повернена матриця компонент містить коефіцієнти кореляції, що характеризують зв'язки між змінними аналізованого набору даних і виявленими у побудованій факторній моделі факторами (компонентами). З кожним фактором поєднують ті змінні, які мають з ним найбільш тісний зв'язок – найбільше за модулем значення коефіцієнта кореляції.

3. Аналіз значень коефіцієнтів кореляції між виявленими факторами та змінними набору даних показує, що з першим фактором – *компонентом 1*, більше за все корелюють змінні *Поведінка*, *Уважність*, *Мотивація*. Крім того, змінна *Поведінка* пов'язана із фактором оберненим зв'язком – коефіцієнт кореляції є від'ємним. Це не викликає подиву, оскільки мінімальні бали при оцінці цієї ознаки отримали ті студенти, які ретельно розбиралися із завданнями.

4. З другим фактором – *компонентом 2*, більше всього корелюють змінні *Швидкість*, *Активні відповіді*, *Впевненість*. Тут виявилася така закономірність: активно відповідали на питання ті ж студенти, які були більш впевнені та виконували завдання більш швидко. Активність та швидкість є різними проявами впевненості в собі, що не є гарантом відсутності помилок.

5. Змінна *Відсутність* слабо пов'язана з кожним із виявлених факторів. Тому її доцільно виключити з аналізу і не враховувати при їх змістовій інтерпретації.

Повернутая матрица компонентов[a]

	Компонент	
	1	2
Швидкість	-,165	,967
Поведінка	-,946	,120
АктивніВідповіді	-,158	,965
Відсутність	,195	,284
Уважність	,966	-,120
Впевненість	-,327	,919
Мотивація	,969	-,138

Метод выделения факторов: метод главных компонент.
Метод вращения: варимакс с нормализацией Кайзера.

a. Вращение сошлось за 3 итераций.

Рис. 14.13. Таблиця Повернена матриця компонент/ Rotated Component Matrix

14.2.8. Графічне представлення факторних навантажень

Графік *Діаграма компонент у повернутому просторі/Component plot in rotated space* (рис. 14.14) містить ту ж інформацію, що й таблиця *Повернута матриця компонентів/Rotated component matrix*. Наприклад, *Поведінка*, *Уважність*, *Мотивація* мають високе факторне навантаження за першим фактором й мале факторне навантаження за другим фактором. Відповідно вони розташовані близько до горизонтальної осі й далеко від вертикальної осі.

Рис. 14.14. Графік Діаграма компонент у повернутому просторі/ Component plot in rotated space

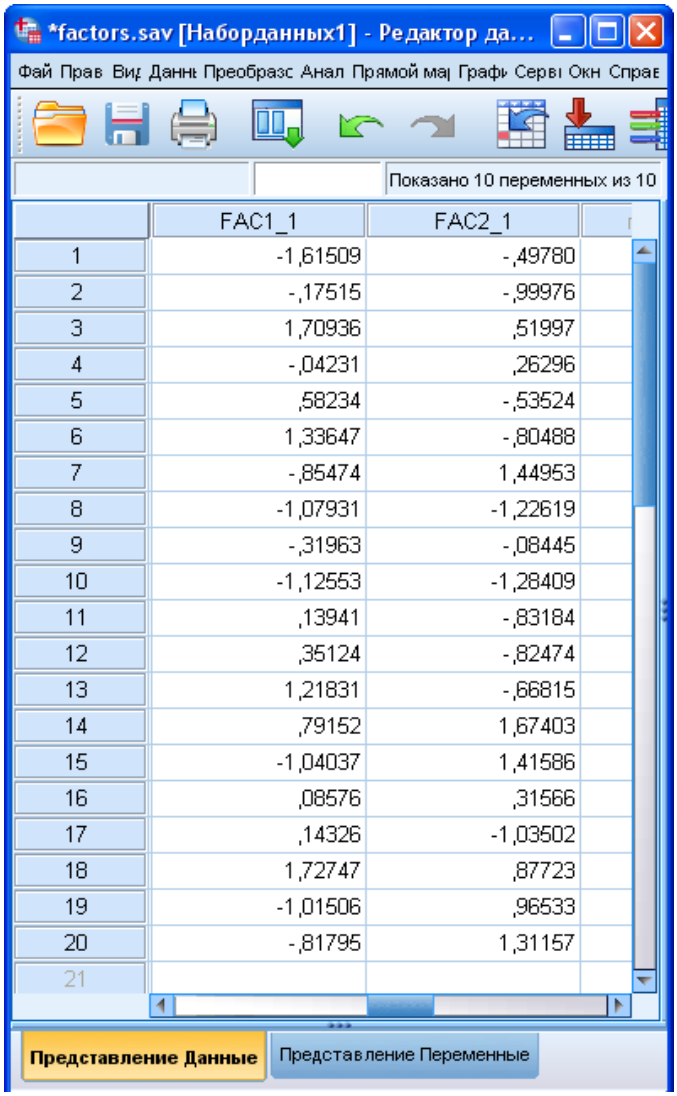
14.2.9. Інтерпретація результатів

Інтерпретація результатів полягає в установлені сутності кожного фактора та відповідно до цього визначенні його назви. З урахування здійсненого попереднього аналізу маємо:

1) перший фактор (*компонент 1*) зібрав старанних студентів, які були більш мотивовані, більш уважні та більше часу приділяли виконанню завдань під час занять;
2) другий фактор (*компонент 2*) зібрав студентів, які були впевнені у собі, перші давали відповідь на поставлені питання та швидко виконували поставлені завдання.

14.2.10. Збереження факторів як нових змінних

1. У результаті проведеного за методом головних компонент факторного аналізу було створено нові змінні, що відповідають виявленим факторам, та розраховано їх значення для кожного об'єкта набору даних – факторні ваги. Ці змінні та їх значення з'являються у вікні редактора даних поруч зі значеннями вихідних змінних набору даних (рис. 14.15).



	FAC1_1	FAC2_1
1	-1,61509	-,49780
2	-,17515	-,99976
3	1,70936	,51997
4	-,04231	,26296
5	,58234	-,53524
6	1,33647	-,80488
7	-,85474	1,44953
8	-1,07931	-1,22619
9	-,31963	-,08445
10	-1,12553	-1,28409
11	,13941	-,83184
12	,35124	-,82474
13	1,21831	-,66815
14	,79152	1,67403
15	-1,04037	1,41586
16	,08576	,31566
17	,14326	-1,03502
18	1,72747	,87723
19	-1,01506	,96533
20	-,81795	1,31157
21		

Рис. 14.15. Нові факторні змінні у вікні редактора даних SPSS

2. Перейшовши на вкладку *Представлення – Змінні* редактора даних, можна відредагувати назви цих змінних відповідно до їх установленої сутності (рис. 14.16).

3. У подальшому аналізі об'єкти набору даних будуть представлені новими змінними, кількість яких є меншою за вихідну кількість змінних набору даних у 3,5 рази. Крім зменшення розрахункової складності обчислень, виявлена більш проста факторна структура змінних, яка дозволяє надати нову змістову інтерпретацію явищам досліджуваної предметної області.

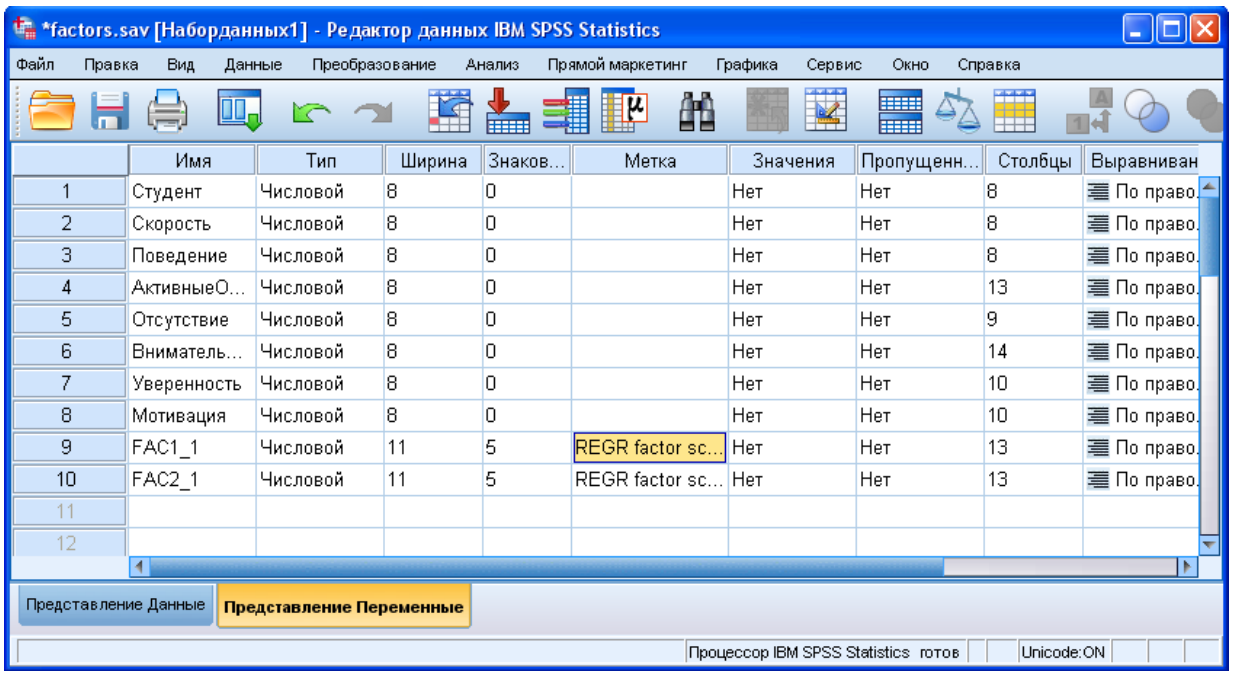


Рис. 14.16. Вкладка Представлення Змінні у вікні редактора даних SPSS

14.3. ЗАВДАННЯ ДЛЯ САМОСТІЙНОЇ РОБОТИ

Завдання 2. На основі проведеного факторного аналізу визначити наступне.

2.1. Як краще назвати нові факторні змінні?

2.2. Як варто інтерпретувати результати факторного аналізу, якщо значення критерію КМО становить *A*, а значення величини Значущість/*Significance* для критерію Бартлетта – *B*?

2.3. Якою є оптимальна кількість факторів факторної моделі, якщо в таблиці *Пояснення сукупної дисперсії/Total Variance Explained* мінімальне значення характеристичних чисел, що перевищує одиницю, перебуває в *C*-му рядку, а в стовпці *Сумарний%/Cumulative%* – міститься значення, рівне *D*?

Значення *A, B, C, D* відповідно до варіанта необхідно обрати з таблиці 14.2.

КОНТРОЛЬНІ ПИТАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ № 14

1. У чому полягає сутність та головна мета факторного аналізу даних?
2. Що є фактором у факторному аналізі даних?
3. У яких випадках застосовують підтверджуючий факторний аналіз?
4. Коли застосовують дослідницький факторний аналіз?
5. Що таке факторне навантаження та факторна вага?
6. Чим відрізняються загальні фактори від характерних?
7. Основні методи факторного аналізу даних.
8. Яким вимогам повинні задовольняти змінні набору даних для їх відповідності основним методам факторного аналізу?
9. Якою є залежність між вихідними змінними та факторами у факторній моделі?
10. У чому полягає сутність методу головних компонент?
11. Основна відмінність методу головних факторів від методу головних компонент.
12. Як здійснюється визначення оптимальної кількості компонент факторної моделі розрахунковим і графічним способами?
13. Якими є основні критерії факторного аналізу?
14. Яким чином здійснюється обертання факторів і з якою метою воно проводиться?
15. З якою метою і яким чином компоненти факторної моделі зберігаються як нові змінні у вихідному файлі даних SPSS?

Таблиця 14.2

Індивідуальні варіанти з даними до завдання 2

Варіант	A	B	C	D
1	0,742	0,02	5	74,206
2	0,125	0,01	3	82,640
3	0,954	0,001	2	45,187
4	0,561	0,006	4	78,924
5	0,311	0,01	3	99,821
6	0,421	0,61	5	45,158
7	0,658	0,02	4	56,004
8	0,579	0,01	5	19,125
9	0,778	0,001	3	84,614
10	0,211	0,006	2	95,419
11	0,651	0,01	4	74,245
12	0,425	0,51	4	74,238
13	0,789	0,36	5	12,579
14	0,998	0,02	3	95,438
15	0,451	0,01	5	56,145
16	0,561	0,001	3	31,445
17	0,198	0,006	2	42,198
18	0,846	0,01	4	65,821
19	0,954	0,45	5	57,936
20	0,742	0,71	3	77,864
21	0,123	0,02	3	81,640
22	0,321	0,60	5	48,158
23	0,828	0,03	4	59,004
24	0,671	0,01	4	75,245
25	0,591	0,001	3	33,115

СПИСОК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ

Базові

1. Черняк І. О. Інтелектуальний аналіз даних / О. І. Черняк, П. В. Захарченко. – К. : Знання, 2014. – 599 с.
2. Актуальні проблеми Data Mining : навч. посіб. для студентів факультету комп'ютерних наук та кібернетики / О. О. Марченко, Т. В. Россада. – К. : КНУ ім. Т. Шевченка, 2017. – 150 с.
3. Барсегян А. А. Анализ данных и процессов: уч. пособие / А. А. Барсегян, М. С. Куприянов, И. И. Холод, М. Д. Тесс, С. И. Елизаров. – СП. : БХВ-Петербург, 2009. – 512 с.
4. Гороховатський В. О., Творошенко І. С. Методи інтелектуального аналізу та оброблення даних : навч. посіб. / В. О. Гороховатський, І. С. Творошенко. – Харків : ХНУРЕ, 2021. – 92 с.
5. Ланде Д. В., Субач І. Ю., Бояринова Ю. Є. Основи теорії і практики інтелектуального аналізу даних у сфері кібербезпеки : навч. посіб. – К. : ІСЗЗІ КПІ ім. І. Сікорського, 2018. – 297 с.
6. Сирота А. А. Методы и алгоритмы анализа данных и их моделирование в MATLAB : уч. пособие / А. А. Сирота. – СПб. : БХВ-Петербург, 2016. – 384 с.
7. Брантон С. Л., Куц Дж. Н. Анализ данных в науке и технике / пер. с англ. А. А. Слинкина. – М. : ДМК Пресс, 2021. – 542 с.
8. Ковалева М. А., Волошин С. Б. Анализ даных : уч. пособие / М. А. Ковалева, С. Б. Волошин. – М. : Мир науки, 2019. – 129 с.
9. Інтелектуальний аналіз даних. Комп'ютерний практикум : навч. посіб. / О. О. Сергеєв-Горчинський, Г. В. Іщенко. – К. : КПІ ім. І. Сікорського, 2018. – 73 с.
10. Statistics and Analysis of Scientific Data / Massimiliano Bonamente. – New York : Springer Science+Business Media LLC, 2017. – 323 p.
11. Брюс П., Брюс Є. Практическая статистика для специалистов Data Science / П. Брюс, Є. Брюс. – СПб. : БХИ-Петербург, 2018. – 304 с.
12. Бахрушин В. Є. Інтелектуальний аналіз даних : підручник / В. Є. Бахрушин. – К. : Знання, 2014. – 599 с.
13. Лупан І. В. Комп'ютерні статистичні пакети: навч.-метод. посіб. / І. В. Лупан, О. В. Авраменко. – Кіровоград, 2010. – 218 с.
14. Інтелектуальний аналіз даних : практикум / М. Т. Фісун, І. О. Кравець, П. П. Казмірчук, С. Г. Ніколенко. – Л. : «Новий світ-2000», 2016. – 162 с.
15. Методи статистичного аналізу даних: навч. посіб. / Г. Г. Швачич, В. С. Коноваленков, О. В. Соболенко та ін. – Дніпропетровськ : РМетФУ, 2017. – 178 с.
16. Наследов А. IBM SPSS Statistics 20 и AMOS: профессиональный статистический анализ данных. – СПб. : Питер, 2013. – 416 с.
17. Дьяконов В. П. MATLAB. Полный самоучитель. – М. : ДМК Пресс, 2012. – 768 с.
18. Доррер М. Г. Моделирование нейронных сетей в системе MatLab : лабораторный практикум / М. Г. Доррер. – Красноярск, 2021. – 98 с.

Допоміжні

1. Федорончак Т. В. Методичні вказівки до лабораторних з дисципліни «Інтелектуальний аналіз даних» / Т. В. Федорончак. – Запоріжжя : ЗНТУ, 2016. – 60 с.
2. Ліщина Н. М., Ящук А. А. Інтелектуальний аналіз даних : методичні вказівки до виконання лабораторних робіт / Н. М. Ліщина, А. А. Ящук. – Луцьк : Луцький НТУ, 2015. – 102 с.
3. Технології штучного інтелекту-2. Комп'ютерні технології інтелектуального аналізу даних : метод. вказівки до виконання практикуму для студ. спец. «Автоматизоване управління технологічними процесами» / укл. Д. О. Ковалюк. – К. : НТУУ КПІ, 2014. – 26 с.
4. Интеллектуальный анализ данных (Введение в Data Mining) : учеб. пособие / А. А. Шумейко, С. Л. Сотник. – Днепропетровск, 2012. – 212 с.
5. Data Mining. Concepts and Techniques / Jiawei Han, Micheline Kamber, Jian Pei. – 3rd ed., 2012. – 744.
6. Интеллектуальный анализ данных : учебное пособие для студентов / Г. Ю. Чернышова. Саратовский гос. соц.-экономический университет. – Саратов, 2012. – 92 с.
7. IBM SPSS Statistics Base 22: руководство пользователя по работе с модулем «Statistic Base» в IBM SPSS Statistics 22.0. – 218 с.
8. Василенко О. А. Математично-статистичні методи аналізу у прикладних дослідженнях: навч. посіб. / О. А. Василенко, І. А. Сенча. – Одеса : ОНАЗ ім. О. С. Попова, 2011. – 166 с.
9. Паянок Т. М., Задорожня Т. М. Статистичний аналіз даних : навч. посіб. / Т. М. Паянок, Т. М. Задорожня. – Ірпінь : Університет ДФСУ, 2020. – 312 с.
10. Особенности системы MatLAB для решения задач вычислительной математики: уч. пособие / Е. А. Кочегурова. – Томск : Из-во Томского политехнического университета, 2013. – 110 с.
11. Єгоршин О. О. Довідник з математичної статистики з прикладами обчислень у MatLab : навч.-практ. посіб. Ч. 2 / О. О. Єгоршин, Л. М. Малярець, Б. В. Сінкевич. – Харків : Вид. ХНЕУ, 2009. – 508 с.

12. Лабораторний практикум з дисципліни «Статистичне моделювання та прогнозування» / укл. О. В. Раєвнєва, І. В. Чанкіна, Л. А. Гольтяєва. – Х. : Вид. ХНЕУ ім. С. Кузнеця, 2014. – 68 с.

13. Методичні вказівки до лабораторних та самостійних робіт із дисципліни «Математична статистика» / упоряд.: О. І. Василик, М. В. Карташов, В. П. Кнопова та ін. – К. : ВПЦ «Київський університет», 2014. – 84 с.

14. Козирєва О. В., Федорова В. О. Статистика : навч. посіб. / О. В. Козирєва, В. О. Федорова. – Х. : Видавництво Іванченка, 2021. – 187 с.

ДОДАТКИ

Додаток А

Зразок оформлення титульного аркуша звіту про виконання лабораторної роботи

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет імені Петра Могили
Факультет комп'ютерних наук
Кафедра інтелектуальних інформаційних систем

ЗВІТ
про виконання лабораторної роботи № 2

НАБОРИ ДАНИХ. ШКАЛИ. ПОПЕРЕДНЯ ОБРОБКА ДАНИХ

з дисципліни «Інтелектуальний аналіз даних»

Студента 301 групи ____Ткаченко О.І._____
(прізвище, ім'я, по батькові)
Викладач: канд.пед.н. доцент Болюбаш Н.М.___
(посада, прізвище, ініціали)

Миколаїв – 2023

Додаток Б

Системні змінні, команди командного режиму MatLab

1. Основні системні змінні (константи)	
i або j	Уявна одиниця (корінь квадратний з -1)
pi	Число $\pi = 3{,}1415926...$
eps	Погрішність операцій над числами з плаваючою точкою (2^-52)
realmin	Найменше число з плаваючою точкою (2^-1022)
realmax	Найбільше число з плаваючою точкою (2^1023)
inf	Значення машинної нескінченності
ans	Змінна, що зберігає результат останньої операції і може бути використана для відображення останнього результату у Командному вікні
NaN	Вказівка на нечисловий характер даних (Not_a_Number)
2. Команди управління вікном командного режиму	
clc	Очищає екран і розміщає курсор у лівому верхньому куті порожнього екрану
echo filename on	Включає режим виведення на екран тексту Script-файлу (файлу-сценарію)
echo filename off	Виключає режим висновку на екран тексту Script-файлу
echo fi1ename	Міняє режим виведення на протилежний
echo on all	Включає режим виведення на екран тексту всіх m-файлів
echo off all	Відключає режим виведення на екран тексту всіх m-файлів
more on	Включає режим посторінкового виведення (корисний при перегляді великих m-файлів)
more off	Відключає режим посторінкового виведення (у цьому випадку для перегляду більших фалів треба користуватися лінійкою прокручування)
diary file_name.txt	Веде запис на диск усіх команд у рядках введення та отриманих результатів у вигляді текстового файлу з зазначеним ім'ям
diary off	Призупиняє запис у файл
diary on	Починає запис у файл
clear name1, name2, …	Використовується для знищення певних змінних чи функцій з робочої області пакету
clear all	Знищення усіх змінних або функцій з робочої області пакету
help	Видає назви всіх доступних файлів допомоги
help filename	Допомога у роботі з окремим файлом із зазначеним ім’ям
3. Команди редактора Command Window MatLab	
Клавіші	Призначення
$\rightarrow$ або $\leftarrow$	Переміщення курсору на один символ вправо або вліво
$Ctrl+\rightarrow$	Переміщення курсору вправо на одне слово
$Ctrl+\leftarrow$	Переміщення курсору вліво на одне слово
Home	Переміщення курсору в початок рядка
End	Переміщення курсору в кінець рядка
$\uparrow i \downarrow$	Перегляд попередніх команд вперед або назад для підстановки в рядок формул
Del	Видалення символу праворуч від курсору
Backspace	Видалення символу ліворуч від курсору
$Ctrl+k$	Видалення символів до кінця рядка
Esc	Очищення рядка вводу
Ins	Включення/виключення режиму вставки
PgUp	Перегортання сторінок сесії вгору
PgDown	Перегортання сторінок сесії вниз

Додаток В

Оператори MatLab

Кожному оператору в MatLab відповідає певна функція. Повний список операторів можна одержати, використовуючи команду *help ops*.

1. Арифметичні оператори та функції MatLab		
Оператор	Опис	Функція
+	Додавання	plus
+	Унарний плюс	uplus
–	Віднімання	Minus
–	Унарний мінус	uminus
*	Матричне множення	mtimes
.*	Поелементне множення масивів	times
^	Піднесення матриці до степеня	mpower
.^	Поелементне піднесення масиву до степеня	power
/	Ділення матриць зліва направо	mrdivide
\	Ділення матриць справа наліво (обернене ділення матриць)	mldivide
./	Поелементне ділення масивів зліва направо	rdivide
.\	Поелементне ділення масивів справа наліво	ldivide
2. Оператори відношення та їх функції		
Оператор	Назва	Функція
= =	Дорівнює	eq
~ =	Не дорівнює	ne
<	Менше	lt
>	Більше	gt
< =	Менше або дорівнює	le
> =	Більше або дорівнює	ge
3. Логічні оператори і функції		
Оператор	Назва	Функція
&	Логічне І	and
|	Логічне АБО	or
~	Логічне НІ	not
	Виключаюче АБО	Xor

4. Робота логічних операторів					
x	*y*	*x&y* *and(x,y)*	*x|y* *or(x,y)*	*~x* *not(x)*	*xor(x,y)*
0	0	0	0	1	0
0	1	0	1	1	1
1	0	0	1	0	1
1	1	1	1	0	0

Додаток Г

Спеціальні символи, пріоритет виконання операцій в MatLab

1. Спеціальні символи		
Символ	Опис	
:	Дві крапки слугують для формування підвекторів та підматриць із векторів та матриць, а також для створення числових послідовностей	
( )	Круглі дужки застосовуються для задання порядку виконання операцій в арифметичних виразах, вказівки аргументів функції та вказівки індексів елементу вектора або матриці	
[ ]	Круглі дужки застосовуються для формування векторів та матриць	
{ }	Фігурні дужки призначені для формування масивів комірок	
.	Десяткова крапка застосовується для відокремлення дробової частини числа від цілої, а також для виділення полів структур	
..	Дві крапки означають перехід по дереву каталогів на один рівень угору	
…	Три і більше крапок у кінці рядка означають продовження цього рядка	
;	Використовується для розділення рядків матриць (у круглих дужках), а також для заборони виведення на екран результатів обчислень (у кінці операторів)	
,	Кома застосовується для розділення індексів елементів матриці і аргументів функції, а також для розділення операторів у рядку	
%	Знак процента ставиться перед текстовими коментарями (їх MatLab ігнорує)	
!	Знак оклику свідчить про введення команди операційної системи	
=	Знак дорівнює є символом привласнення значень у арифметичних виразах	
‘	Апостроф є символом транспонування; текст, взятий у апострофи, представляється як вектор символів з компонентами, які є ASCII-кодами символів	
[,]	Горизонтальна конкатенація матриць	
[;]	Вертикальна конкатенація матриць	
2. Пріоритет виконання операцій (*наведено у порядку їх зниження, може бути змінено з допомогою дужок*)		
1	( )	Круглі дужки
2	.’	Транспонування
	‘	Транспонування з комплексним спряженням
	^	Піднесення до степеня,
	.^	По елементне піднесення до степеня
3	+	Унарний плюс
	-	унарний мінус
	~	Логічне заперечення
4	.*, *	Множення
	./, .\, /, \	Ділення
5	+ і -	Додавання та віднімання
6	<, <=, >, >=, ++,~=	Операції відношення
7	&	Логічне і
8	|	Логічне або

Додаток Д

Деякі елементарні математичні функції MatLab

1. Тригонометричні функції	
sin, cos, tan, cot	Тригонометричні
asin, acos, atan, acot	Обернені тригонометричні
sinh, cosh, tanh, coth	Гіперболічні функції
asinh, acosh, atanh, acoth	Обернені гіперболічні
sec, csc	Секанс і косеканс
asec, acsc	Обернені функції секанса і косеканса
sech, csch	Гіперболічний секанс і косеканс
asech, acsch	Обернені гіперболічні секанс і косеканс
2. Логарифмічні функції	
exp	Експонента
log	Натуральний логарифм
log10	Десятковий логарифм
log2	Логарифм за основою два
3. Піднесення до степеня	
x^y	Піднесення числа *x* до степеня *y*
pow2(n)	Підносить 2 у степінь n
sqrt	Корінь квадратний
nextpow2	Повертає степінь *n* виразу 2^n
4. Функції комплексного аргументу	
abs	Модуль
angle	Фаза
conj	Комплексно-спряжене число
imag	Уявна частина комплексного числа
real	Дійсна частина комплексного числа
cplxpair	Сортування на комплексно-спряжені пари
5. Функції округлення та обчислення остатку від ділення	
fin(x)	Округлення до найближчого цілого в сторону нуля
floor(x)	Округлення до найближчого цілого в сторону від'ємної нескінченості
ceil(x)	Округлення до найближчого цілого в сторону додатної нескінченості
round(x)	Округлення до найближчого цілого
mod(x)	Остаток від цілочисельного ділення з врахуванням знаку
rem(x)	Остаток від цілочисельного ділення по модулю
sign(x)	Визначення знаку числа

Додаток Е

Функції MatLab для роботи з матрицями

Ім'я	Призначення	Приклад	Результат
ones	Формує одиничну матрицю із заданим числом рядків та стовбців	>>M = ones(2,3)	M = 1 1 1 1 1 1
zeros	Формує матрицю нулів із заданим числом рядків та стовбців	>>M = zeros(2,3)	M = 0 0 0 0 0 0
rand	Формує матрицю випадкових значень в діапазоні [0;1] із заданим числом рядків та стовбців	>>M = rand(2,3)	M = 0.9501 0.6068 0.8913 0.2311 0.4860 0.7621
min	Знаходить найменший елемент матриці у стовбцях у рядках	M = 0.9501 0.6068 0.8913 0.2311 0.4860 0.7621 >>m1 = min(M,[],1) >>m2 = min(M,[],2)	m1 = 0.2311 0.4860 0.7621 m2 = 0.6068 0.2311
max	Знаходить найбільший елемент у стовбцях матриці у рядках матриці	M = 0.9501 0.6068 0.8913 0.2311 0.4860 0.7621 >>m1 = max(M,[],1) >>m2 = max(M,[],2)	m1 = 0.9501 0.6068 0.8913 m2 = 0.9501 0.7621
sum	Сумує елементи матриці у стовпцях у рядках	>> m = sum(M,1) >>m = sum(M,2)	m = 1.1812 1.0928 1.6534 m = 2.4482 1.4792

Додаток Є

Основні формули нормалізації та стандартизації даних

Формули наведено для здійснення нормалізації значень змінної x, представленої набором: $x_1, x_2, ..., x_n$, де n – кількість об'єктів, x' – нормалізовані значення змінної x.

№	Формула	Характеристика формули
1	min-max нормалізація $x_i' = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}}, \ i \in \{1,..., n\}$	Область нормалізованих значень: [0, 1]. Рекомендується використовувати, якщо значення початкових даних рівномірно заповнюють область дослідження. Для деяких методів прогнозування формула неефективна у випадку рівності значень нулю або їх зосередження біля кінців відрізка [0, 1]
	$x_i' = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}} \cdot (B - A) + A$ $x \in [x_{\min}, x_{\max}], \ x' \in [A, B], i \in \{1,..., n\}$	Різновидом min-max нормалізації є формула перетворення до діапазону [A, B]
2	max-min нормалізація $x_i' = \frac{x_{\max} - x_i}{x_{\max} - x_{\min}}, \ i \in \{1,..., n\}$	Аналогічна першій формулі, але дозволяє зворотньо пропорційно розвернути шкалу, що зручно у випадках, коли більшість характеристик мають максимізуватися, а дана характеристика – мінімізуватися. Недоліки ті ж самі
3	z- нормалізація (стандартизація) $x_i' = \frac{x_i - \bar{x}}{\sigma_x}, \ i \in \{1,..., n\}$	Відрізняється тим, що стандартизовані значення є безрозмірними, знаходяться з дисперсією та середньоквадратичним відхиленням, рівними 1, на відрізку $\left[\frac{x_{\min} - \bar{x}}{\sigma_x}; \frac{x_{\max} - \bar{x}}{\sigma_x}\right]$ переважно в околі нуля. У перетворенні використовується середнє значення $\bar{x}$ та стандартне (середньоквадратичне) відхилення σ_x змінної x. *Корисно*: коли невідомі максимальні та мінімальні значення або є домінуючі аномалії (значення у певних проміжках)
4	$x_i' = \frac{2(x_i - x_{\min})}{x_{\max} - x_{\min}} - 1, \ i \in \{1,..., n\}$	Область нормалізованих значень: [-1, 1]. Формула зручна для використання при прогнозуванні із застосуванням нейронних мереж, в яких активаційною функцією є гіперболічний тангенс. Має всі ті ж переваги й недоліки, що і перетворення 1 та 2
5	$x_i' = \frac{1}{1 + e^{-x_i}}$	Область нормалізованих значень: [0, 1]. Використовується рідко, здебільшого для підсилення реакції на зміни значень в околі нуля
6	Масштабування $x_i' = \lambda \cdot x_i, \ \lambda \neq 0, \lambda = const$ $i \in \{1,..., n\}$	Зміна значень змінної x шляхом множення на константу λ

Додаток Ж

Статистичні функції MatLab

Функція	Призначення
mean (A)	Повертає середнє арифметичне значення елементів масиву, якщо А – вектор, або повертає вектор-рядок, що містить середні арифметичні значення елементів кожного стовпця, якщо А – матриця
mean(A,dim)	Повертає середнє значення елементів по стовпцях або по рядках матриці залежно від значення dim (dim=l по стовпцях й dim=2 по рядках відповідно)
median(A)	Повертає медіану, якщо А – вектор; або вектор-рядок медіан для кожного стовпця, якщо А – матриця
median(A,dim)	Повертає значення медіан для стовпців або рядків матриці залежно від значення скаляра dim (dim=l по стовпцях й dim=2 по рядках відповідно)
std(X)	Повертає стандартне (середньоквадратичне) відхилення елементів масиву, якщо X - вектор. Якщо X – матриця, то std(X) повертає вектор-рядок, що містить стандартне відхилення елементів кожного стовпця
std(X,flag)	Повертає те ж значення, що й std(X), якщо flag=0, а якщо flag=1, функція std(X,1) повертає середньоквадратичне відхилення
std(X,flag,dim)	Повертає стандартне (середньоквадратичне) відхилення по рядках (dim=2) або по стовпцях (dim=1) матриці X залежно від значення змінної dim
min(A)	Повертає мінімальний елемент, якщо А – вектор або повертає вектор-рядок, що містить мінімальні елементи кожного стовпця, якщо А – матриця
min(A,B)	Повертає масив того ж розміру, що й А і В, кожен елемент якого є мінімальний з відповідних елементів цих масивів
min(A,[ ],dim)	Повертає найменший елемент по стовпцях або по рядках матриці залежно від значення скаляра dim: min(A,[ ],1) повертає мінімальні елементи кожного стовпця матриці А, а min(A,[ ],0) повертає мінімальні елементи кожного рядка матриці А
[C,I] = min(A)	Крім мінімальних значень повертає вектор індексів цих елементів
max(A)	Повертає максимальний елемент, якщо А – вектор або повертає вектор-рядок, що містить максимальні елементи кожного стовпця, якщо А – матриця
var(X)	Повертає дисперсію елементів масиву X

Додаток З

Значення константи S функції plot(X,Y,S) в MatLab

Функція *plot(X,Y,S)* аналогічна *plot(X,Y)*, проте тип лінії графіка можна задавати за допомогою строкової констранти *S*. Значеннями константи *S* можуть бути наступні символи:

Колір лінії	
y	Жовтий
m	Фіолетовий
c	Блакитний
r	Червоний
g	Зелений
b	Синій
w	Білий
k	Чорний
Тип точки	
.	Точка
o	Окружність
x	Хрест
+	Плюс
*	Зірочка
s	Квадрат
d	Ромб
v	Трикутник (униз)
	Трикутник (нагору)
<	Трикутник (уліво)
>	Трикутник (вправо)
p	
Тип лінії	
-	Суцільна
:	Подвійний пунктир
-.	Штрих-пунктир
--	Штрихова

Приклад простої програми для побудови графіків трьох функцій з різним стилем представлення кожної з них:

```
>> x= -2*pi:0.1:2*pi;
>> y1=sin(x); y2=sin(x).^2; y3=sin(x).^3;
>> plot(x,y1,'-m', x,y2,'-. +r', x,y3,'-ok')
```

Додаток К

Закони розподілу випадкової величини

К.1. Закони розподілу неперервної випадкової величини

Для неперервної випадкової величини x:

$$f(x) = F'(x),\ F(x) = \int f(x)dx,$$

де $f(x)$ – щільність ймовірності, $F(x)$ – функція розподілу.

К.1.1. Нормальний розподіл

Нормальний закон розподілу є найбільш поширеним.

Щільність ймовірності: $f(x) = \dfrac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$,

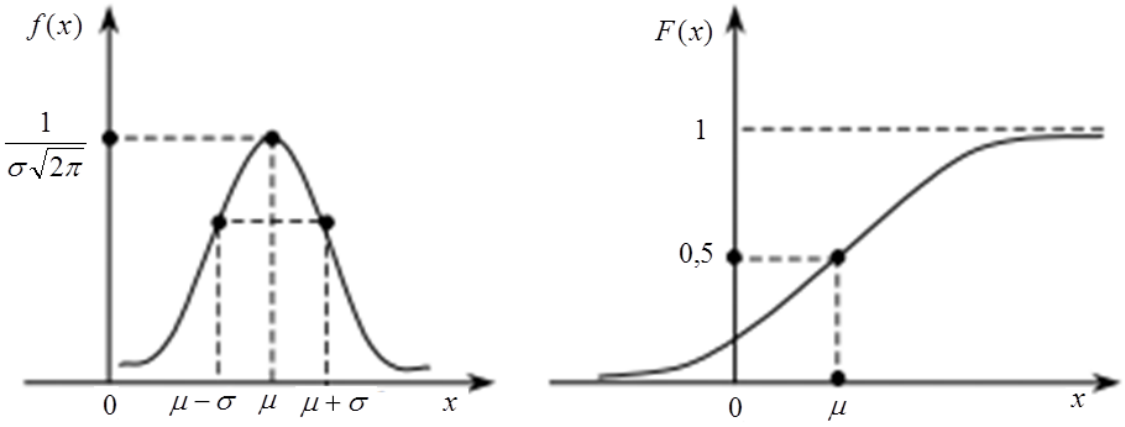
де μ – математичне сподівання, σ – стандартне відхилення.

Для нормального розподілу медіана, мода та математичне сподівання співпадають.

Функція розподілу: $F(x) = \int\limits_{-\infty}^{+\infty} f(x)dx$.

Інтеграл функції розподілу не виражається через елементарні функції, її значення знаходять за таблицями. Ймовірність попадання у інтервал рівна: $P(x_1 \le x \le x_2) = \Phi(\dfrac{x_2-\mu}{\sigma}) - \Phi(\dfrac{x_1-\mu}{\sigma})$,

де $\Phi(z) = \dfrac{1}{\sqrt{2\pi}} \int\limits_{-\infty}^{z} e^{\frac{-z^2}{2}} dz$ – функція Лапласа.



а) графік щільності ймовірності б) графік функції розподілу

Рис. К.1. Графіки функцій номального розподілу

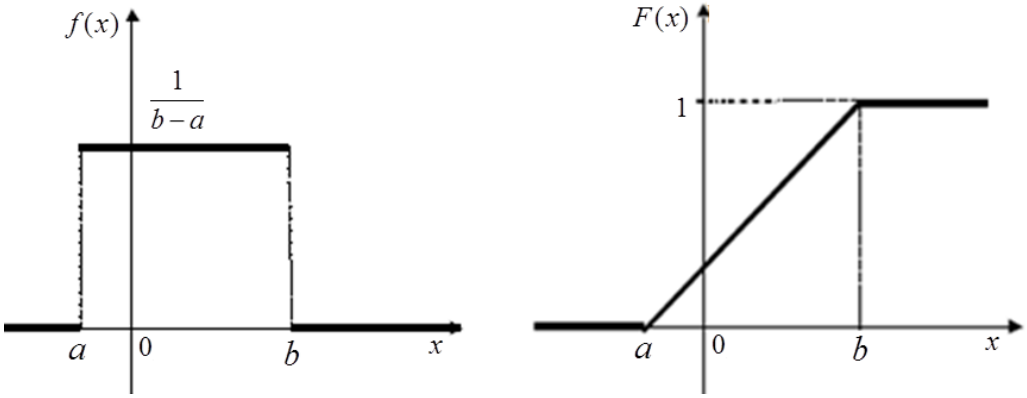
К.1.2. Рівномірний розподіл

Щільність ймовірності: $f(x) = \begin{cases} 0, & x < a; \\ \dfrac{1}{b-a}, & a \le x \le b; \\ 0, & x > b. \end{cases}$,

де $[a; b]$ – інтервал розподілу,

$\mu = \dfrac{b+a}{2}$ – математичне сподівання, $\sigma = \dfrac{b-a}{2\sqrt{3}}$ – стандартне відхилення.

Функція розподілу: $F(x)=\begin{cases} 0, & x<a; \\ \dfrac{x-a}{b-a}, & a\le x\le b; \\ 1, & x>b. \end{cases}$



а) графік щільності ймовірності б) графік функції розподілу

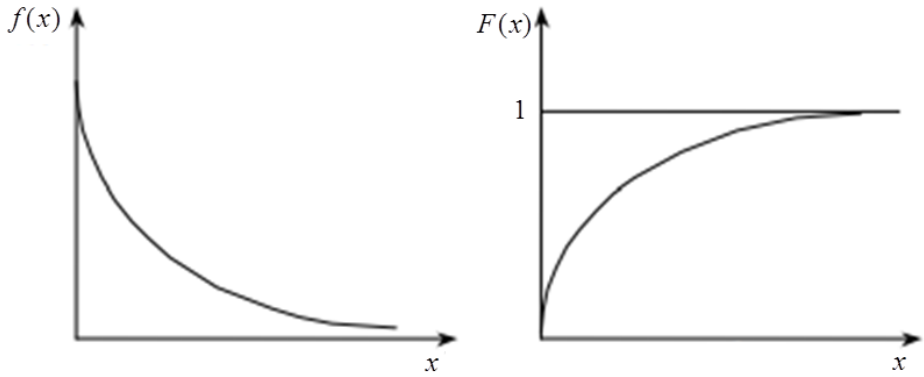
Рис. К.2. Графіки функцій рівномірного розподілу

К.1.3. Показниковий (експоненціальний) розподіл

Щільність ймовірності: $f(x)=\begin{cases} 0, & x\le 0 \\ \lambda e^{-\lambda x}, & x>0 \end{cases}$,

де $\mu=\dfrac{1}{\lambda}$ – математичне сподівання, $\sigma=\dfrac{1}{\lambda}$ – стандартне відхилення.

Функція розподілу: $F(x)=\begin{cases} 1-e^{-\lambda x}, & x\ge 0, \\ 0, & x<0. \end{cases}$



а) графік щільності ймовірності б) графік функції розподілу

Рис. К.3. Графіки функцій показникового розподілу

К.2. Закони розподілу дискретної випадкової величини

Закони розподілу дискретної випадкової величини мають місце, коли проводиться n незалежних випробувань, у кожному з яких певна подія може відбутися з однією й тією ж ймовірністю p, незалежною від інших випробувань. Появу події називають успіхом, а число появ події у серії випробувань є випадкова величина x, яка може приймати цілі додатні значення від 0 до n. Ймовірність успіху – p, тоді ймовірність невдачі дорівнюватиме $q=1-p$.

К.2.1. Біноміальний розподіл

Дискретна випадкова величина x має біноміальний розподіл – розподіл Бернуллі, якщо ймовірність набуття нею конкретних значень k – ймовірність випадання точно k успішних випадків при n випробуваннях, рівна: $P(x=k)=C_n^k \cdot p^k \cdot q^{n-k}$, де $C_n^k = \frac{n!}{k!(n-k)!}$ – біноміальний коефіцієнт, $k \in [0,n]$.

Функція ймовірностей: $f(x)=C_n^x \cdot p^x \cdot q^{n-x}$, де $x \in N,\ x \in [0,n]$.

$\mu = np$ – математичне сподівання, $\sigma = \sqrt{npq}$ – стандартне відхилення.

Функція розподілу: $F(x) = \begin{cases} 0, & при\ x \le 0 \\ \sum_{x=1}^{n} C_n^x \cdot p^x \cdot q^{n-x}, & при\ 0 < x \le n \\ 1, & при\ x > n \end{cases}$.

К.2.2. Розподіл Пуассона

Дискретна випадкова величина x має розподіл Пуассона тоді, коли ймовірність набуття нею конкретних значень k – ймовірність випадання точно k успішних випадків при n випробуваннях, рівна:

$$P(x=k)=\frac{\lambda_k}{k!}e^{-\lambda}, \text{ де } \lambda = np .$$

Функція ймовірностей: $f(x)=\frac{\lambda^x}{x!}e^{-\lambda}$, де $x \in N,\ x \in [0,n]$.

$\mu = \lambda$ – математичне сподівання, $\sigma = \lambda$ – стандартне відхилення.

Функція розподілу: $F(x) = \begin{cases} 0, & при\ x \le 0 \\ \sum_{x=1}^{n} \frac{\lambda^x}{x!} e^{-\lambda}, & при\ 0 < x \le n \\ 1, & при\ x > n \end{cases}$.

Дискретна випадкова величина x має розподіл Пуассона тоді, коли n велике, а ймовірність появи події у окремому випробуванні незначна.

Додаток Л

Параметри функції clusterdata() в MatLab

<table>
<tr><td colspan="3">Синтаксис функції:
clusterdata(X, 'distance', 'linkage', 'cutoff', 'maxclust', 'criterion')</td></tr>
<tr><td rowspan="6">Параметри</td><td>X</td><td>Матриця вхідних даних</td></tr>
<tr><td>distance</td><td>Задає спосіб визначення міри близькості між об’єктами (відстані для метричних даних), по замовчуванню – відстань Евкліда</td></tr>
<tr><td>linkage</td><td>Задає метод зв’язку кластерів при об’єднанні</td></tr>
<tr><td>cutoff</td><td>Задає поріг несумісності для об’єднання об’єктів у кластери</td></tr>
<tr><td>maxclust</td><td>Задає максимальну кількість кластерів</td></tr>
<tr><td>criterion</td><td>Задає критерій для об’єднання об’єктів у кластери: по замовчуванню inconsistent (несумісність)</td></tr>
<tr><td colspan="2">Значення параметру distance</td><td>Назва міри близькості</td></tr>
<tr><td>1</td><td>euclidean</td><td>Відстань Евкліда</td></tr>
<tr><td>2</td><td>seuclidean</td><td>Стандартизована відстань Евкліда (поділена на середньоквадратичне відхилення відповідних змінних)</td></tr>
<tr><td>3</td><td>cityblock</td><td>Манхетенська відстань (міських кварталів)</td></tr>
<tr><td>4</td><td>minkowski</td><td>Відстань Мінковського</td></tr>
<tr><td>5</td><td>cosine</td><td>Відстань косинус</td></tr>
<tr><td>6</td><td>correlation</td><td>Відстань кореляції</td></tr>
<tr><td>7</td><td>hamming</td><td>відстань Хеммінга</td></tr>
<tr><td colspan="2">Значення параметру linkage</td><td>Назва методу зв’язку кластерів</td></tr>
<tr><td>1</td><td>single</td><td>Метод найближчого сусіда</td></tr>
<tr><td>2</td><td>complete</td><td>Метод найдальшого сусіда</td></tr>
<tr><td>3</td><td>average</td><td>Метод середнього зв’язку</td></tr>
<tr><td>4</td><td>centroid</td><td>Центроїдний метод (має сенс тільки якщо 'distance' дорівнює 'euclidean')</td></tr>
<tr><td>5</td><td>ward</td><td>Метод Уорда</td></tr>
<tr><td colspan="2">Значення параметру maxclust</td><td>Ціле число – максимальне число кластерів, що формуються</td></tr>
</table>

Додаток М

Синтаксис та параметри функції kmeans() в MatLab

Приклади застосування функції *kmeans()*:

IDX = kmeans(X,k) – розділяє множину об’єктів заданих матрицею вхідних даних X на k кластерів, *IDX* – вектор-стовпець, що вказує належність об’єкта до певного кластера (індекс кластера);

[IDX,C] = kmeans(X,k) – повертає ще матрицю C координат центрів ваги кластерів – центроїдів;

[IDX,C, sumd] = kmeans(X,k) – повертає ще вектор *sumd* сум відстаней від об’єктів кожного кластера до його центроїда;

[IDX,C, sumd,D] = kmeans(X,k) – повертає ще матрицю *D* відстаней від усіх об’єктів до центроїдів – центрів кожного кластера (розмірністю $n \times k$).

У таблиці М.1 наведено додаткові параметри функції *kmeans()*.

Таблиця М.1

Значення параметрів функції *kmeans*()

<table>
<tr><td colspan="3">Синтаксис функції:
kmeans(*X*, *k*, '*distance*', '*start*', '*replicates*, '*maxiter*', '*emptyaction*', '*display*')</td></tr>
<tr><td rowspan="7">Параметри</td><td>*X*</td><td>Матриця вхідних даних</td></tr>
<tr><td>*k*</td><td>Кількість кластерів, на які необхідно розділити множину об’єктів, заданих матрицею *X*</td></tr>
<tr><td>*distance*</td><td>Спосіб визначення мір близькості, яку мінімізує функція kmeans</td></tr>
<tr><td>*start*</td><td>Метод побудови початкових центрів кластерів</td></tr>
<tr><td>*replicates*</td><td>Кількість повторень розв’язку задачі кластеризації з різними початковими центрами кластерів (повертається розв’язок з мінімальною величиною *sumd*), задається неявно, якщо у якості параметру обрано 3-вимірний масив</td></tr>
<tr><td>*maxiter*</td><td>Максимальна кількість ітерацій (по замовчуванню 100)</td></tr>
<tr><td>*emptyaction*</td><td>Дія у випадку, якщо у кластері не виявляється ні однієї точки</td></tr>
<tr><td>*display*</td><td>Рівень виведення повідомлень</td></tr>
<tr><td colspan="2">Значення параметру *distance*</td><td>Назва міри близькості</td></tr>
<tr><td>2</td><td>*sqeuclidean*</td><td>Квадрат відстані Евкліда</td></tr>
<tr><td>3</td><td>*cityblock*</td><td>Манхетенська відстань
(міських кварталів)</td></tr>
<tr><td>5</td><td>*cosine*</td><td>Відстань косинус</td></tr>
<tr><td>6</td><td>*correlation*</td><td>Відстань кореляції</td></tr>
<tr><td colspan="2">Значення параметру
start</td><td>Назва методу</td></tr>
<tr><td>1</td><td>*sample*</td><td>*k* центрів кластерів обираються з матриці *X* випадковим чином (по замовчуванню)</td></tr>
<tr><td>2</td><td>*uniform*</td><td>*k* центрів кластерів обираються випадковим чином (рівномірно) з діапазону зміни *X* (не може застосовуватися, якщо була задана відстань Хеммінга)</td></tr>
<tr><td>3</td><td>*cluster*</td><td>Початкові центри кластерів обираються за результатами розв’язання задачі кластеризації для 10% випадково обраних з X об’єктів з параметром *start* рівним *sample*</td></tr>
<tr><td>4</td><td>*матриця*</td><td>Матриця розміром $k \times n$ центрів кластерів у явному вигляді. У цьому випадку замість параметру *k* функції *kmeans* вводиться пустий масив [ ]. Функція *kmeans* визначить кількість кластерів як першу розмірність матриці. Можна також задати 3-вимірний масив, припускаючи довжину 3-ї розмірності як значення параметру *replicates*</td></tr>
</table>

Значення параметру *emptyaction*		Назва дії
1	*error*	Розглядає появу порожнього кластера як помилку (за замовчуванням)
2	*drop*	Видаляє усі порожні кластери, відповідні значення у параметрах *C* і *D* установлюються в *NaN*[1].
3	*singleton*	Створює новий кластер з одного об'єкту, найбільш віддаленого від центра його кластера
Значення параметру *display*		Рівень повідомлень
1	*off*	Ніяких повідомлень немає
2	*iter*	Видає інформацію про кожну ітерацію: її номер, стадію процесу оптимізації, кількість переміщених об'єктів і загальну суму відстаней
3	*final*	Виведення фінальної інформації
4	*notify*	Видається тільки попередження й повідомлення про помилки (за замовчуванням)

Приклади виклику функції *k-means* із додатковими параметрами:

```
[IDX,C, sumd,D] = kmeans(X,k,'display','iter','distance','correlation')
[IDX,C, sumd,D] = kmeans(X,[],'start',[1 2; 3 4]);
```

[1] NaN (англ. Not-a-Number) –„не число", особливий стан числа із плаваючою комою, який може виникнути, коли попередня операція завершилася з невизначеним результатом або число не відповідає необхідним умовам

Додаток Н

Синтаксис та параметри функції fcm() в MatLab

Функція *fcm()* визначає для кожного об’єкта вхідних даних ступінь належності до певного кластера й ітераційно просуває центри кластерів у правильному напрямку.

Синтаксис функції fcm(): [C, U, J] = fcm(data, k, options), де

1) вхідні параметри:
   a) *data* – множина даних для кластеризації: матриця вхідних даних;
   b) *k* – кількість кластерів (більше 1) ;
2) результуючі дані, які функція повертає:
   a) *C*– матриця кінцевих центрів ваги кластерів – їх центроїдів, кожен рядок містить координати центрів ваги;
   b) *U* – кінцева матриця нечіткого розбиття об’єктів на *k* кластерів;
   c) *J* – значення цільової функції на кожному етапі ітерації;
3) додаткові параметри:
   a) *options(1)* – показник ступеня належності для матриці нечіткого розбиття *U* (коефіцієнт нечіткого розбиття, по замовчуванню рівний 2);
   b) *options(2)* – максимальна кількість ітерацій (по замовчуванню 100);
   c) *options(3)* – мінімально припустима зміна значень цільової функції: параметр зупинки (по замовчуванню рівна 1*e*-5);
   d) *options(4)* – відображення інформації на кожному кроці ітерації (по замовчуванню 1).

Приклад виклику функції *fcm* із додатковими параметрами:

[C, U, J] = fcm(data, k, [2, 80, 1*e*-4, 1])

Процес кластеризації закінчується, коли досягнуто максимальну кількість ітерацій або значення цільової функції між двома послідовними ітераціями менше заданої величини точності – параметра зупинки.

Додаток П

Критичні значення критерію Ірвіна

Обсяг вибірки, n	Критерій Ірвіна, $\lambda_{кр}$		
	$\alpha = 0{,}1$	$\alpha = 0{,}05$	$\alpha = 0{,}01$
2	2,33	2,77	3,64
3	1,79	2,17	2,90
4	1,58	1,92	2,60
5	1,45	1,77	2,46
6	1,37	1,67	2,30
7	1,31	1,60	2,22
8	1,26	1,55	2,14
9	1,22	1,50	2,09
10	1,18	1,46	2,04
11	1,15	1,43	2,00
12	1,13	1,40	1,97
13	1,11	1,38	1,94
14	1,09	1,36	1,91
15	1,08	1,34	1,89
20	1,03	1,27	1,80
25	0,99	1,23	1,74
30	0,96	1,20	1,70
35	0,93	1,17	1,66
40	0,91	1,15	1,63
45	0,89	1,13	1,61
50	0,88	1,11	1,59
60	0,86	1,08	1,56
70	0,84	1,06	1,53
80	0,83	1,04	1,51
90	0,82	1,03	1,49
100	0,81	1,02	1,47
200	0,75	0,95	1,38
300	0,72	0,91	1,33
500	0,69	0,88	1,28
1000	0,65	0,83	1,22

Додаток Р

Лінеаризація рівнянь регресії

Шукане рівняння: $Y = A_0 + A_1 X$

Вид функції	Перетворення змінних	Параметри рівняння регресії	Перехід до початкових змінних
Степенева $y = a_0 \cdot x^{a_1}$	$X = \ln x$ $Y = \ln y$ $A_0 = \ln a_0$ $A_1 = a_1$	$A_1 = \frac{\overline{XY} - \bar{Y} \cdot \bar{X}}{\overline{X^2} - \bar{X}^2}$ $A_0 = \bar{Y} - A_1 \cdot \bar{X}$	$\tilde{y} = e^Y$, $x = e^X$, $a_0 = e^{A_0}$, $a_1 = A_1$
Показникова $y = a_0 \cdot a_1^x$	$X = x$ $Y = \ln y$ $A_0 = \ln a_0$ $A_1 = \ln a_1$	$A_1 = \frac{\overline{xY} - \bar{Y} \cdot \bar{x}}{\overline{x^2} - \bar{x}^2}$ $A_0 = \bar{Y} - A_1 \cdot \bar{x}$	$\tilde{y} = e^Y$, $x = X$, $a_0 = e^{A_0}$, $a_1 = e^{A_1}$
Обернена $y = \frac{1}{a_0 + a_1 x}$	$X = x$ $Y = 1/y$ $A_0 = a_0$ $A_1 = a_1$	$A_1 = \frac{\overline{xY} - \bar{Y} \cdot \bar{x}}{\overline{x^2} - \bar{x}^2}$ $A_0 = \bar{Y} - A_1 \cdot \bar{x}$	$\tilde{y} = 1/Y$, $x = X$, $a_0 = A_0$, $a_1 = A_1$
Напівлогарифмічна $y = a_0 + a_1 \ln x$	$X = \ln x$ $Y = y$ $A_0 = a_0$ $A_1 = a_1$	$A_1 = \frac{\overline{Xy} - \bar{y} \cdot \bar{X}}{\overline{X^2} - \bar{X}^2}$ $A_0 = \bar{y} - A_1 \cdot \bar{X}$	$\tilde{y} = Y$, $x = e^X$, $a_0 = A_0$, $a_1 = A_1$
Гіперболічна $y = a_0 + \frac{a_1}{x}$	$X = 1/x$ $Y = y$ $A_0 = a_0$ $A_1 = a_1$	$A_1 = \frac{\overline{Xy} - \bar{y} \cdot \bar{X}}{\overline{X^2} - \bar{X}^2}$ $A_0 = \bar{y} - A_1 \cdot \bar{X}$	$\tilde{y} = Y$, $x = 1/X$, $a_0 = A_0$, $a_1 = A_1$
Експоненціальна $y = a_0 e^{a_1 \cdot x}$	$X = x$ $Y = \ln y$ $A_0 = \ln a_0$ $A_1 = a_1$	$A_1 = \frac{\overline{xY} - \bar{Y} \cdot \bar{x}}{\overline{x^2} - \bar{X}^2}$ $A_0 = \bar{Y} - A_1 \cdot \bar{x}$	$\tilde{y} = Y$, $x = X$, $a_0 = e^{A_0}$, $a_1 = A_1$

ДЛЯ НОТАТОК

ДЛЯ НОТАТОК

ДЛЯ НОТАТОК

Навчальне видання

Надія Миколаївна
БОЛЮБАШ

ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ

Навчальний посібник

Редактор *О. Михайлова*
Комп'ютерна верстка, дизайн обкладинки *К. Гросу-Грабарчук*
Друк *С. Волинець.* Фальцювально-палітурні роботи *О. Мішалкіна.*

Підписано до друку 14.12.2023.
Формат 60х84$^{1}/_{8}$. Папір офсет.
Гарнітура «Times New Roman». Друк ризограф.
Ум. друк. арк. 37,2. Обл.-вид. арк. 15,1.
Тираж 100 пр. Зам. № 6600.

Видавець і виготовлювач: ЧНУ ім. Петра Могили.
54003, м. Миколаїв, вул. 68 Десантників, 10.
Тел.: 8 (0512) 50–03–32, 8 (0512) 76–55–81, e-mail: rector@chmnu.edu.ua.
Свідоцтво суб'єкта видавничої справи ДК № 6124 від 05.04.2018.