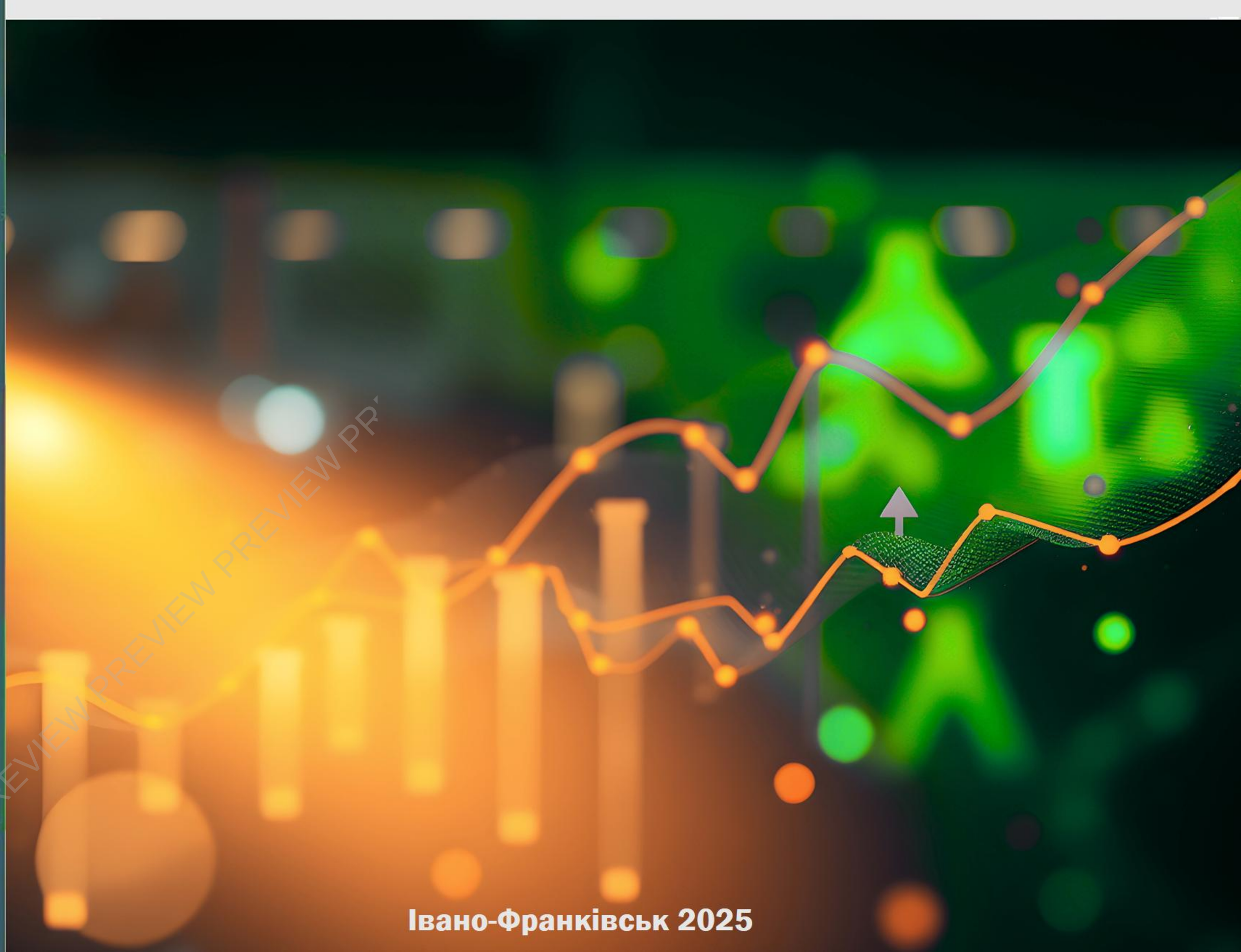
Міністерство освіти і науки України
Карпатський національний університет імені Василя Стефаника

Віктор Гусак
Дмитро Господарьов
Володимир Лущак

СТАТИСТИКА МАЛИХ ВИБІРОК У БІОЛОГІЇ І МЕДИЦИНІ З ОСНОВАМИ ПРОГРАМУВАННЯ В *PYTHON* і *R*

Посібник

Друге видання



Івано-Франківськ 2025

Гусак Віктор Васильович

**кандидат біологічних наук,
доцент**

Господарьов Дмитро Валерійович

**кандидат біологічних наук,
доцент**

Лущак Володимир Іванович

**доктор біологічних наук,
професор**

Міністерство освіти і науки України
Карпатський національний університет
імені Василя Стефаника

Віктор Гусак
Дмитро Господарьов
Володимир Лущак

СТАТИСТИКА МАЛИХ ВИБІРОК У БІОЛОГІЇ І МЕДИЦИНІ З ОСНОВАМИ ПРОГРАМУВАННЯ В *PYTHON* І *R*

Посібник

Друге видання

Івано-Франківськ
Видавець Голіней О. В.
2025

УДК 519.25(075.8)
С 78

Схвалено до друку Вченою радою Факультету природничих наук Карпатського національного університету імені Василя Стефаника (протокол № 3 від 18 листопада 2025 року)

Рецензенти:
Осипчук Михайло Михайлович – професор, доктор фізико-математичних наук, професор кафедри математичного та функціонального аналізу Карпатського національного університету імені Василя Стефаника
Бандура Андрій Іванович – професор, доктор фізико-математичних наук, професор кафедри фізико-математичних наук Івано-Франківського національного технічного університету нафти і газу

Відповідальний редактор Гусак В. В., кандидат біологічних наук, доц.

С 78 **Гусак В. В., Господарьов Д. В., Лущак В. В.** Статистика малих вибірок у біології і медицині з основами програмування в *Python* і *R*. Друге видання. Івано-Франківськ : Голіней О. В., 2025. 262 с.

ISBN 978-617-8470-47-0

У навчальному посібнику описані та пояснені основні методи статистичного аналізу даних, зокрема тих, які отримуються у дослідженнях в галузях медицини та біології. Найбільше уваги приділено методам параметричної і непараметричної статистики, дисперсійного, кореляційного та регресійного аналізів. Подані критерії оцінки ймовірності різниці між середніми, в тому числі і для множинних порівнянь. Проаналізовані основні комп'ютерні програми, які часто використовуються для обробки даних. Теоретичний матеріал супроводжується великою кількістю професійно спрямованих прикладів і задач для самостійної роботи.

Навчальний посібник адресується студентам, аспірантам, науковим працівникам, викладачам, які ведуть дослідження в різних галузях медицини і біології.

Друге видання, виправлене та розширене. Видання містить оновлені приклади кодів Python і R із QR для завантаження, нові статистичні тести, а також розширені табличні дані. У посібнику подано 108 математичних формул, 56 кодів Python і R для обчислення статистичних показників, 12 рисунків і 22 таблиці, бібліографічний список містить 29 найменувань.

УДК 519.25(075.8)

ISBN 978-617-8470-47-0

ЗМІСТ

ТЛУМАЧНИЙ СЛОВНИК ТЕРМІНІВ

Асиметрія (англ. *Skewness, γ_1*) – показник, який характеризує ступінь несиметричності розподілу. Це відношення центрального моменту третього порядку μ_3 до куба середнього квадратичного відхилення σ^3: $\gamma_1 = \mu_3/\sigma^3$.

Варіанта, випадкова величина (англ. *Random variable; x_i*) – окреме випадкове значення варіаційної ознаки, якого набуває ця ознака в ряді розподілу.

Варіаційний ряд (англ. *Variation range*) – послідовність кількісних показників проявів станів певної ознаки (варіант), розташованих у черговості їх зростання чи зменшення.

Вибіркова сукупність, або вибірка (англ. *Sample*) – це сукупність об'єктів, вибраних випадковим чином з генеральної сукупності. Власне цей набір даних може бути підданий далі статистичній обробці.

Генеральна сукупність (англ. *Population)* – сукупність всіх об'єктів, що підлягають дослідженню. Обсяг генеральної сукупності, тобто число об'єктів дослідження може бути досить великим. Часто буває неможливо дослідити всі об'єкти генеральної сукупності.

Дисперсія (англ. *Variance, σ^2*) – міра відхилення випадкових значень від середньої значення величини для вибірки або генеральної сукупності.

Довірчий інтервал (англ. *Confidence interval, Δx*) – це інтервал, у межах якого з заданою ймовірністю можна чекати значення

оцінюваного параметру. Застосовується для повнішої оцінки точності у порівнянні з точковою оцінкою.

Ексцес (англ. *Kurtosis, γ_2*) – ступінь загостреності (згладженості) кривої емпіричного розподілу. Це характеристика, що обчислюється за наступною формулою: $\gamma_2=\mu_4/\sigma^4-3$, де де μ_4 – центральний момент четвертого порядку, σ^2 – дисперсія.

Індекс **кореляції** (англ. *Correlation index*) – умовна величина, розрахована лише по відношенню до певної кривої. Її значення може бути доведене до 1, якщо в як криву, що описує зв'язок, взяти параболу, в якій кількість параметрів доведена до кількості одиниць спостереження. Ця величина використовується для вимірювання щільності криволінійного зв'язку і визначається аналогічно до коефіцієнта кореляції. Індекс кореляції приймає значення від 0 до 1. Певного знака він не має, оскільки на різних відрізках кривої напрям зв'язку може змінюватись.

Закон розподілу (англ. *Distribution law*) – це будь-яке співвідношення, що встановлює зв'язок між можливими значеннями випадкової величини та відповідними ймовірностями.

Квантиль (англ. *Quantile*) – значення, яке задана випадкова величина не перевищує з фіксованою ймовірністю, що є порядком квантиля. Квантилі відсікають в межах ряду певну частину його членів. Тобто, квантиль розподіл значень – це таке число x_p, що значення p-ї частини сукупності менше або рівне x_p. Наприклад, квантиль 0,25 (також називається 25-м

процентилем або нижнім квартилем) змінної – це таке значення (x_p), коли 25% (p) значень змінної попадають нижче даного значення. Значення 50-ого квантилю є медіаною.

Квартиль (англ. *Quartile*) – це квантиль порядку p = 0,25 або p = 0,75.

Коефіцієнт асиметрії (англ. *Skewness*) – це безрозмірна характеристика симетричності статистичного ряду.

Коефіцієнт варіації (англ. *Coefficient of variation*) – відношення стандартного відхилення до абсолютного значення математичного сподівання випадкової величини.

Коефіцієнт детермінації (англ. *Determination coefficient, r-squared,* R^2) – статистичний показник, що використовується в статистичних моделях як міра залежності варіації залежної змінної від варіації незалежних змінних. Вказує наскільки отримані спостереження підтверджують модель.

Коефіцієнт ексцесу (англ. *Coefficient of kurtosis*) – числова характеристика розподілу ймовірностей дійсної випадкової величини, що характеризує «крутість», тобто, стрімкість підвищення кривої розподілу у порівнянні з нормальною кривою.

Кореляційний аналіз (англ. *Correlation analysis*) – метод дослідження взаємозалежності ознак у генеральній сукупності, які є випадковими величинами.

Кореляція (англ. *Correlation*) – це взаємозалежність двох або декількох випадкових величин.

Крива регресії (англ. *Regression curve*) – для двох випадкових величин X і Y це крива, яка відображає залежність умовного математичного сподівання випадкової величини Y за умови $X = x$ для кожної змінної x. Якщо крива регресії Y по X являє собою пряму лінію, то регресію називають «простою лінійною».

Крива розподілу (англ. *Distribution curve*) – це крива, що зображає щільність розподілу $f(x)$ випадкової величини.

Медіана (англ. *Median*) – значення x_i, розміщене посередині ряду значень, що розставлені від найменшого числа до найбільшого. Геометрично, медіана – це абсциса точки, у якій площа, що обмежена кривою розподілу, ділиться навпіл. Це квантиль порядку $p=0,5$.

Метод найменших квадратів (англ. *Method of least squares*, *МНК*) – метод оцінки параметрів моделі на підставі експериментальних даних, що містять випадкові помилки. В основі методу лежать наступні міркування: при заміні точного (невідомого) параметра моделі приблизними значенням необхідно мінімізувати суму квадратів різниць між експериментальними даними і теоретичними (обчисленими за допомогою запропонованої моделі). Це дозволяє розрахувати параметри моделі за допомогою МНК з мінімальною похибкою у вказаному вище розумінні.

Непараметричні критерії (англ. *Nonparametric tests*) – це критерії перевірки гіпотез про розподіли сукупностей. Це критерій Колмогорова-Смирнова, Манна–Вітні і багато інших.

Непараметричні критерії позбавлені обмежень, які присутні при застосуванні параметричних критеріїв. Ці критерії дозволяють вирішити деякі важливі завдання, які супроводжують дослідження в біології: виявлення відмінностей у рівні досліджуваної ознаки, оцінка зсуву значень досліджуваної ознаки, виявлення відмінностей у розподілах ознак.

Об'єм вибірки **(**англ. *Sample size***)** – число вибіркових одиниць у вибірці.

Параметричні критерії (англ. *Parametric tests*) – це критерії, які використовуються в задачах перевірки параметричних гіпотез і включають у свій розрахунок показники розподілу, наприклад, середні, дисперсії тощо. Це такі відомі класичні критерії, як t-критерій Стьюдента, F-критерій Фішера тощо. Параметричні критерії дозволяють прямо оцінити рівень основних параметрів генеральних сукупностей, різниці середніх і відмінності в дисперсіях. Ці критерії здатні виявити тенденції зміни ознаки при переході від умови до умови, оцінити взаємодію двох і більше факторів у впливі на зміни ознаки. Параметричні критерії вважаються дещо більш потужними, ніж непараметричні, за умови, що ознака виміряна інтервальною шкалою і нормально розподілена. Однак з інтервальною шкалою можуть виникнути певні проблеми, якщо дані представлені не в стандартизованих оцінках.

Потужність критерію (англ. *Power of a test*) – ймовірність відхилити нульову гіпотезу, коли вона хибна.

Промахи (англ. *Outliers*) – це спостереження у вибірці, які відрізняються від інших за величиною настільки, що виникає припущення про їхню приналежність іншій сукупності або отримання в результаті похибки вимірювань.

Регресійний аналіз, регресія (англ. *Regression analysis* і *regression*) – статистичний метод, використовуваний для дослідження відносин між двома величинами. На відміну від суворої функційної залежності *y=f(x)* у регресійній моделі одному й тому ж значенню величини x можуть відповідати кілька значень величини y, іншими словами, при фіксованому значенні x величина y має деякий випадковий розподіл.

Розмах вибірки (англ. *Range*) – різниця між найбільшим і найменшим значеннями кількісної ознаки у вибірці.

Розподіл (англ. *Probability distribution*) – функція, яка визначає ймовірність того, що випадкова величина прийме якесь задане значення або належатиме заданому проміжку значень.

- ***нормальний*** (англ. *Normal distribution, Laplace-Gauss distribution*) – розподіл ймовірностей неперервної випадкової величини X такий, що щільність розподілу ймовірностей має вигляд: $f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$

Репрезентативність – головна властивість вибіркових сукупностей характеризувати генеральну сукупність з відповідною точністю та достатньою надійністю.

Репрезентативністю позначають і ступінь відповідності вибіркових показників генеральним параметрам.

Середня арифметична величина (англ. *Arithmetic mean*) – це частка від ділення суми всіх варіант сукупності на їх загальну кількість.

Середня арифметична зважена (англ. *Weighted arithmetic mean*) **–** це величина, яка обчислюється із значень варіювальної ознаки з урахуванням ваг. Її застосовують у тих випадках, коли значення ознаки представлені у вигляді варіаційного ряду розподілу, в якому чисельність одиниць по варіантах не однакова, а також при розрахунку середньої зі середніх при різному обсязі сукупності. Зважування в даному випадку здійснюється за частотами, які показують скільки разів повторюється та або інша варіанта.

Середня арифметична проста **(**англ. *Simple arithmetic mean*) **–** це величина, яка застосовується в тих випадках, коли відомі дані про окремі значення ознаки та їх число в сукупності, тобто розраховується у разі, коли є незгруповані індивідуальні значення ознаки. В статистичній практиці вона застосовується, як правило, для розрахунку середніх рівнів ознак, представлених у вигляді абсолютних показників.

Середня величина (англ. *Mean*) – це узагальнений показник, що характеризує типовий рівень варіювальної ознаки в якісно однорідній сукупності.

Систематична похибка, систематична помилка **(**англ. *Systematic error***)** – компонента помилки результату, яка залишається постійним чи закономірно змінюється в ході отримання результатів перевірки для однієї ознаки.

Стандартне відхилення (англ. *Standard deviation*) – додатний квадратний корінь із значення дисперсії.

Статистична гіпотеза (англ. *Statistical hypothesis*) – це будь-яке припущення щодо вигляду або параметрів невідомого закону розподілу. У конкретній ситуації статистичну гіпотезу формулюють як припущення на певному рівні статистичної значущості про властивості генеральної сукупності за оцінками вибірки. Статистичну гіпотезу прийнято позначати літерою H: (Hypothesis).

Статистичні дані (англ. *Statistical data*) **–** це сукупність показників, отриманих внаслідок статистичного спостереження або обробки даних.

Статистичний критерій (англ. *Statistical test*) — строге математичне правило, за яким приймається або відкидається та або інша статистична гіпотеза. Побудовою критерію є вибір відповідної функції на основі результатів спостережень (ряду емпірично набутих значень ознаки), яка служить для виявлення міри розбіжності між емпіричними значеннями і гіпотетичними.

Стандартна помилка середньої арифметичної величини (англ. *Standard error of the mean*) – стандартна помилка, що вказує на точність, з якою показник вибірки – середнє арифметичне

– представляє (репрезентує) середнє арифметичне для генеральної сукупності.

Ступінь свободи (англ. *Degree of freedom*) – у загальному випадку число доданків (параметрів) мінус число обмежень, що накладаються на них.

Функція розподілу (англ. *Distribution function*) – функція, що задає для будь-якого значення x ймовірність того, що випадкова величина X менша або дорівнює x.

Центральний момент q-го порядку (англ. *Central moment of order q*) – математичне сподівання *q-го степеня* центрованої випадкової величини.

Щільність розподілу (англ. *Probability density function*) – перша похідна функції розподілу неперервної випадкової величини ($f(x)=dF(x)/dx$) в тих точках, де вона існує.

ВСТУП

Математична статистика вже давно і успішно використовується у біології і медицині. Втім, завжди існують початківці, для яких доцільність певних способів обробки, аналізу, представлення даних, а також вибір тестів для статистичних порівнянь не очевидні. Різнобій у подачі числових даних та їх обробки часто зустрічається навіть у рецензованих наукових статтях, а також дисертаціях. Це є свідченням того, що для частини тих, хто сьогодні займається наукою, статистична обробка залишається «незвіданою територією», або неоднозначною частиною знань.

Під час керівництва студентською науковою роботою ми виявили низку проблем, які найчастіше виникають перед початківцями. Перша проблема психологічна – страх перед усім, що стосується математики як дуже складної для розуміння дисципліни. Насправді, коло завдань, які стоять перед дослідником щодо обробки даних, набагато вужче, ніж таке для спеціалістів-математиків. Для найпростішого статистичного аналізу потрібні лишень базові, по суті шкільні знання математики. На сьогодні, більшість необхідних обрахунків не потрібно проводити вручну. Існує багато комп’ютерних статистичних програм, які виконують всі розрахунки за долі секунди. Завдання дослідника – знати алгоритми обрахунку і критерії вибору тестів для порівнянь. Завдяки цьому можна правильно «пояснити» машині, що саме рахувати, і розуміти той результат, який видається на екран монітора. Звісно, для глибокого статистичного аналізу потрібні спеціальні ґрунтовні знання з

окремих математичних дисциплін. Інколи це стає незалежною метою – зрозуміти біологічне явище з математичної точки зору, створити модель процесу. Такий підхід вже приніс багато користі і продовжує успішно розвиватись. Існують навіть спеціалізовані періодичні видання, які зосереджуються саме на біометриці (або біометрії) – науці, яка застосовує статистику, а також інші розділи математики для розв'язання біологічних проблем.

Друга проблема початківців – у формальному відношенні до статистики і надмірній вірі у «всемогутність» статистичних програм. Дійсно, сучасні комп'ютерні програми дозволяють заощадити багато часу, проте вони є тільки допоміжним інструментом. Алгоритм обробки даних, їх аналіз і подача залежать від рішення самого дослідника. При правильному застосуванні статистичний аналіз суттєво допомагає зрозуміти досліджуване явище, при неправильному – ускладнює розуміння і сприйняття даних, а іноді призводить до невірних висновків. Тому за використанням комп'ютерних програм має бути глибоке усвідомлення тих операцій, які виконуються з введеними в програму числами, і того, про що свідчитиме результат. Приємно згадувати, що колись (ці часи ще застали авторів даного посібника) доводилось самостійно складати алгоритми розрахунків на програмованому калькуляторі чи комп'ютері або й без них. У цьому випадку дослідник мусів розуміти саму суть обробки даних.

На сьогодні, цілком природним є бажання лідера дослідницької команди (студентської чи аспірантської), щоб кожен у ній був самостійним у статистиці, міг коректно і правдиво представляти

свою частину результатів. Тому метою даного посібника є не тільки ознайомлення дослідників-початківців з найбільш розповсюдженими методами статистичного аналізу, а також попередження можливих помилок.

РОЗДІЛ 1. СУКУПНІСТЬ, ВИБІРКА І ТИПИ ДАНИХ

1.1. Генеральні та вибіркові сукупності

В більшості випадків питання статистичної обробки даних виникає тоді, коли дослідникові необхідно чисельно охарактеризувати явище. Так, одноразове визначення активності алкогольдегідрогенази у культурі пекарських дріжджів не дає бачення про процеси, які аналізуються. Ця активність залежить від дуже багатьох чинників. Тому повторне визначення активності для цієї самої культури або культури дріжджів, вирощених у подібних умовах, буде відрізнятися. Іншими словами, активність алкогольдегідрогенази у дріжджів буде варіювати. Для оцінки цієї варіації потрібно провести бодай декілька незалежних визначень, або повторів. Середнє значення активності, обраховане на основі значень повторів, а також показники варіації вже є інформативнішими. Набір значень, який ми отримали в результаті незалежних вимірювань, вважатиметься *вибіркою*, а окремі значення *варіантами*. Власне цей набір даних може бути підданий надалі статистичній обробці. Проте вибірка – це не тільки значення, отримані в кількох незалежних вимірюваннях. Частіше під вибіркою розуміють також набір значень, отриманих після вимірювань, зроблених для групи об’єктів, наприклад, для кількох культур дріжджів у нашому випадку. Такою групою можуть бути листки або насіння різних дерев, пацієнти з різними синдромами, риби одного виду тощо.

Будь-яка група, незалежно від її розміру, в статистиці називається *сукупністю*. Об'єкти, які входять у сукупність, мають певні ознаки, які відрізняють їх від інших об'єктів. Розрізняють генеральні та вибіркові сукупності. *Генеральною сукупністю* є всі об'єкти, які відносяться до категорії, що цікавить дослідника. Наприклад, всі мухи виду *Drosophila melanogaster*, всі листки дуба, всі дафнії Івано-Франківської області тощо. В окремих випадках є можливість вивчити всю генеральну сукупність (наприклад, коли вивчаємо ріст всіх студентів одного курсу, або вміст гемоглобіну для всіх в місті хворих на певну рідкісну хворобу). Проте дослідник не може вивчити повністю великі генеральні сукупності. Уявлення про генеральну сукупність можна скласти за її частиною – вибірковою сукупністю. *Вибіркова сукупність*, або *вибірка* – це частина сукупності, відібрана за певними правилами для дослідження з генеральної сукупності.

Для того, щоб за вибіркою скласти правильне уявлення про генеральну сукупність, вона має бути репрезентативною, тобто правильно відображати кількісні та якісні співвідношення генеральної сукупності. Репрезентативність з'ясовується лише тоді, коли необхідно охарактеризувати усю велику сукупність особин, що цікавить дослідника, на основі вивчення лише якоїсь її частини і коли уже відібрана відповідна вибірка. При цьому можливі певні відхилення вибіркових показників від параметрів генеральної сукупності. Такі відхилення називаються похибками репрезентативності. Похибки репрезентативності виникають тому, що вибіркова сукупність не точно відповідає своїй генеральній

сукупності за тими чи іншими ознаками. Ці похибки виникають також внаслідок помилок, яких можна позбутися або звести до мінімального рівня завдяки правильній організації досліду. Охарактеризуємо ці помилки. *Методичні помилки* виникають внаслідок застосування неправильної методики добору та обробки експериментального матеріалу, а також недосконалого проведення експериментів. *Помилки точності* виникають внаслідок помилок первинної реєстрації фактів, вимірювання неперевіреними або зіпсованими приладами, розрахунками з недостатньою або надлишковою точністю (неправильне округлення проміжних даних). *Помилки уваги* – це описки, прорахунки, пропуски даних, переплутування результатів досліджень, друкарські помилки. *Помилки типовості* з'являються головним чином на початкових етапах досліджень, які пов'язані з добором до групи нетипових для генеральної сукупності об'єктів. Це особливо небезпечний вид помилок. Їх можна припуститися несвідомо через нерозуміння правил добору об'єктів до групи (рандомізації). Іншою причиною появи цих помилок (найнебезпечнішою) є тенденційність дослідника, бажання за будь-яку ціну підтвердити свої припущення. Єдиний принцип, який береться в основу відбору об'єктів у вибірку – принцип випадковості. Для реалізації цього принципу, дослідник створює такі умови відбору, щоб у кожного представника генеральної сукупності була однакова ймовірність потрапити у вибірку.

1.2. Уявлення про малу вибірку

Одне з основних питань математичної статистики: якою повинна бути мінімальна необхідна кількість інформації для отримання достатньої статистично коректної правильності результату?

Всесвітньо відомий вчений Рональд Фішер у своїй книзі «The Design of Experiments» (1935) вказав на те, що мінімальне число зразків (повторів експерименту) не може бути меншим, ніж **4**. В іншому випадку, неминуче виникає систематична помилка (систематична помилка, або зсув (bias) – це систематичне (невипадкове, однонапрямлене) відхилення результатів від дійсних значень). Розрізняють декілька основних типів цих помилок. Зсув, зумовлений відбором, виникає, коли порівнювані групи розрізняються не лише за ознакою, яка вивчається, але й за іншими чинниками, що впливають на результат. Зсув, зумовлений вимірюванням, виникає тоді, коли в порівнюваних групах використовуються різні методи вимірювання. Зсув, зумовлений чинниками, які втручаються, виникає, коли один чинник пов'язаний з іншим і ефект одного спотворює ефект іншого.

Дослідники на практиці найчастіше мають справу з малою вибіркою, коли кількість варіант є меншою за 30 ($4 \leq n \leq 30$). Розробка теорії малої вибірки належить англійському статистові Вільяму Сілі Ґоссету (William Sealy Gosset), який у 1908 році опублікував свою працю «Біометрика» під псевдонімом «Стьюдент». Крім того, дослідження, які стосуються малих вибірок, пов'язані також з іменами Колмогорова А.М., Ноймана Дж. І Вальда А. Так, Колмогоров А.М. запропонував критерій достатності

статистики при обмеженому числі спостережень. Дж. фон Нойман створив новий напрямок у математичній статистиці, основне положення якого говорить: «Завдання статистики – виявляти загальний характер поведінки об’єкту в умовах невизначеності». Вальд А. розробив розділ статистики, який називається послідовним аналізом. За ним, необхідний обсяг вибірки, визначається в процесі самих випробувань. Ідеї Колмогорова, Ноймана і Вальда в частині малих вибірок розвинені у багатьох роботах, бібліографію яких можна знайти у фундаментальних працях із математичної статистики.

1.3. Типи даних

У багатьох випадках дослідник має справу з числами. Багато показників, таких як концентрація речовин, оптична густина, розміри, маса, можуть бути виміряні з великою точністю і мати певне числове значення. В інших випадках показники є цілими числами – кількість листків на деревах, кількість відкладених яєць або лялечок комах. Інші ознаки – стать, стан (норма або мутація), колір. Часто, коли технічно неможливо зробити вимірювання і отримати певне числове значення у початківців опускаються руки. Проте багато ознак, які на перший погляд не виражаються числами, можна «оцифрувати», статистично обробити, а отже, отримати значущу інформацію про явище.

Ознаки поділяють на якісні та кількісні. Кількісні ознаки можна виміряти, порахувати і виразити в тих чи інших одиницях вимірювання. За якісними ознаками об’єкти можна поділити на чіткі

категорії. Якісна ознака може мати декілька станів. Так, вовна тварини може бути чорною, білою, коричневою, рудою; колір очей – чорним, карим, сірим, зеленим, блакитним тощо. Деякі якісні ознаки мають два стани, наприклад, стать, ген (мутантний або нормальний), обстежуваний (здоровий чи хворий). Такі якісні ознаки називаються альтернативними. У свою чергу, кількісні ознаки поділяють на неперервні, подані дійсними числами (наприклад, зріст, маса тіла, концентрація речовин), та дискретні, подані цілими числами (наприклад, кількість тварин в певній дослідній групі). Якісні ознаки поділяють на категоріальні та порядкові або рангові. Категоріальні ознаки – це, наприклад, стать, вікові групи. Якщо категорій тільки дві – мутантний або нормальний, присутній або відсутній, живий – мертвий, і т. ін., то ознаки називають дихотомічними. Інколи доводиться мати справу з ознаками, які можна описати за допомогою фізичних величин. Ступінь розвитку таких ознак суб'єктивно оцінюється описом «краще» або «гірше», «більше» або «менше». У таких випадках об'єктові присвоюють ранг – умовне числове значення, яке описує ступінь розвитку ознаки. Тому такі ознаки називаються ранговими.

Розрізняють також декілька шкал вимірювання, за якими класифікують типи даних. Так, виділяють шкалу найменувань, або номінальну (статі, фрукти), порядкову (оцінки успішності – «задовільно», «добре», «відмінно»), інтервальну шкалу (в цій шкалі виражаються такі вимірювання як температура або час), та шкалу відношень (відсотки, долі одиниці).

1.4. Структура даних

Дані можуть бути первинними і вторинними. Первинні дані – це результати безпосередніх вимірювань. Вторинні дані утворюються агрегацією первинних даних. Наприклад, необхідно дізнатись масу тіла людини та з'ясувати рівень глюкози у неї в крові. Одноразове зважування дає досить точний результат і його, як правило, не повторюють. Методика визначення концентрації глюкози в крові складніша і дає менш точний результат. Тому роблять декілька паралельних визначень, за результатами яких обраховується середнє арифметичне. Паралельні вимірювання є первинними даними, а усередненний результат – вторинним.

Якщо один і той самий об'єкт вимірюється двічі (у різних експериментальних умовах), то отримані дані утворюють пари. Пари даних отримуються також у тому випадку, коли кожному об'єктові однієї вибірки відповідає цілком певний об'єкт з іншої вибірки. Такі дані є попарно зв'язані.

1.5. Заокруглення даних

Сучасні калькулятори і комп'ютери дозволяють проводити дуже точні розрахунки. Дуже часто значення середніх виходять числами, які мають багато знаків після коми. Зрозуміло, що в реальності ми не можемо отримати таке число. До якого знаку можна округлювати середні значення так, щоб не втратити точність з одного боку і не ввести колег в оману – з іншого? Існує декілька правил, які дозволяють коректно заокруглити пораховане на комп'ютері значення середньої величини без втрати точності:

1. Якщо вихідні вимірювання виконують до десятої долі одиниці, то всі наступні розрахунки округлюються до десятої долі. Тобто заокруглення в розрахунках відповідає точністю визначень вихідних вимірювань. Так, при подачі середнього арифметичного значення і стандартного відхилення, треба враховувати точність вихідних даних. Необхідно також враховувати точність вимірювань приладів, за допомогою яких дослідник отримує результати.

2. Заокруглюють тільки кінцеве значення обчислень, а всі проміжні обчислення проводять з усіма доступними знаками.

3. Результат вимірювання заокруглюється до того ж десяткового розряду, яким закінчується заокруглене значення абсолютної похибки. Наприклад, результат 5,6342, похибка ±0,11. Результат заокруглюють до 5,63 (5,63±0,11).

4. У переважній більшості випадків заокруглення проводять до трьох значущих цифр. Значущими цифрами числа вважаються всі цифри від першої зліва, яка не дорівнює нулю, до останньої цифри справа. При цьому нулі, які записані у вигляді множника 10^n, не враховуються. Так, число 15,0 має три значущі цифри; число 40 – дві значущі цифри; число 127,20 – п’ять значущих цифр; 0,515×10 – три значущі цифри; 0,0066 – дві значущі цифри. Наприклад, потрібно записувати 312 замість 312,3; 12,4 замість 12,41; 1,12 замість 1,121; 0,323 замість 0,3231. Чому саме заокруглюють до трьох значущих цифр? При такому заокругленні максимальна похибка заокруглення знаходиться в межах від 0,06 до 0,5%, що є досить точним показником. Якщо б ми заокруглювали за однією або

двома значущими цифрами, то максимальна похибка заокруглення знаходилась би в межах від 6 до 50 % і від 0,6 до 5 %, відповідно.

5. Розрізняють записи наближених чисел за кількістю значущих цифр.

Приклад 1. Розрізняють числа 2,3 і 2,30. Запис 2,3 означає, що точні тільки цілі і десяті частки, справжнє значення числа може бути, наприклад, 2,33 і 2,28. Запис 2,30 означає, що точні й соті частки: справжнє значення числа може бути 2,303 і 2,298, але не 2,41 і не 2,382.

Приклад 2. Запис 373 означає, що всі цифри точні: якщо за останню цифру ручатися не можна, то число має бути записане як $3{,}7\times10^2$.

Приклад 3. Якщо в числі 4720 точні лише дві перші цифри, воно має записуватись як $4{,}7\times10^3$.

6. При заокругленні даних слід скористатись *правилом арифметичного заокруглення*:

- якщо за останньою цифрою, яку зберігають, слідують цифри 0, 1, 2, 3 і 4, то її залишають такою ж самою (*заокруглення з недостачею*). Наприклад, число 1,562 заокруглюють до 1,56;

- якщо за останньою цифрою, яку зберігають, стоять цифри 6, 7, 8, 9, то цю цифру збільшують на одну одиницю (*заокруглення з надлишком*). Наприклад, число 1,566 заокруглюють до 1,57;

- якщо цифра, що відкидається, дорівнює 5, а наступні за нею цифри невідомі або нулі, то останню цифру, що зберігається, не змінюють, якщо вона парна, і збільшують на одиницю, якщо вона

непарна. Наприклад, число 1234,50 заокруглюють до 1234, а число 8765,50 – до 8766;

- якщо цифра, що відкидається дорівнює 5, але за нею слідують відмінні від нуля цифри, то останню цифру, що залишається, збільшують на одиницю. Наприклад, число 1,5554 заокруглюють до 1,56.

РОЗДІЛ 2. ПОКАЗНИКИ ВАРІАЦІЇ

2.1. Середні величини та медіана

Одним з найважливіших статистичних параметрів є значення середньої величини. Воно відображує найбільш типове значення для ознаки. Втім, середня величина – це абстрактний показник. Часто вона набуває значень, які можуть жодного разу не зустрітись серед вихідних спостережень. Наприклад, уявімо вибірку з чотирьох рослин, які мають довжину стебла: 2, 4, 6 та 8 см. Середнє арифметичне буде дорівнювати 5 см – значенню, яке було відсутнє у вибірці. Варіація ознаки завжди має певні межі. Не виключено, що всі рослини даного виду, у певному віці, за відсутності хвороб і впливу довкілля мають довжину стебла не менше 2 см і не більше 8 см. Середнє з цих двох крайніх значень буде так само становити 5 см. Отже, на основі середньої величини можна міркувати не тільки про властивості окремої вибірки, але і про генеральну сукупність. Розрізняють декілька типів середніх величин: середню арифметичну, середню квадратичну, середню кубічну і середню гармонійну. Загальна формула для обчислень більшості середніх величин наступна:

$$\overline{x} = \left(\frac{1}{n}\sum_{i=1}^{n} x_i^m\right)^{\frac{1}{m}} \qquad \textbf{(1)}$$

де x_i – значення варіанти, n – кількість варіант. Значення m вказує на тип середньої: якщо m = +1, то обчислюється середня арифметична величина; –1 – середня гармонійна; +2 – середня квадратична; +3 – середня кубічна величина.

У більшості досліджень використовується середня арифметична величина, яка може бути простою або зваженою.

Просту середню арифметичну величину отримують шляхом додавання всіх отриманих значень і поділу цієї суми на число значень. Якщо набір з *n* досліджень змінної «*x*» зобразити як «x_1, x_2, x_3, ..., x_n», то формула для обчислення простої середньої арифметичної величини «$\bar{x}$» (її також позначають як «M_x») буде мати наступний вигляд:

$$\bar{x} = \frac{x_1 + x_2 + x_3 + x_n}{n} \text{ або } \bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}, \quad \textbf{(2)}$$

де *n* – число досліджень, Σ – сума значень варіант, *i* – порядковий номер значення, x_i – певне значення у вибірці.

Приклад 4. Потрібно обчислити середнє арифметичне значення активності ферменту супероксиддисмутази в зябрах карася сріблястого, коли були отримані наступні дані: 54,3; 68,2; 55,6; 60,0; 51,4 (Од/мг білка).

Для цього використаємо формулу **(2)**:

$$\bar{x} = \frac{54{,}3 + 68{,}2 + 55{,}6 + 60{,}0 + 51{,}4}{5} = 57{,}9 \text{ (Од/мг білка).}$$

Середнє арифметичне значення можна знайти за допомогою встановлених стандартних бібліотек *Python* (*Statistics*, *NumPy*) або через написання формули для його обчислення.

Код 1 з використанням бібліотеки ***stats*** наступний:

```
import stats
numbers = [54.3, 68.2, 55.6, 60.0, 51.4]
mean = stats.mean(numbers)
print("Середнє арифметичне: ", mean)
```

Код 2 з використанням бібліотеки ***statistics*** наступний:

```
import statistics as st
numbers = [54.3, 68.2, 55.6, 60.0, 51.4]
mean = st.mean(numbers)
print("Середнє арифметичне: ", mean)
```

Код 3 з використанням бібліотеки ***numpy*** наступний:

```
import numpy as np
numbers = [54.3, 68.2, 55.6, 60.0, 51.4]
mean = np.mean(numbers)
print("Середнє арифметичне: ", mean)
```

Код 4 без використання бібліотек Python наступний:

```
def calculate_mean(numbers):
    total = sum(numbers)
    count = len(numbers)
    mean = total / count
    return mean

# Приклад використання
numbers = [54.3, 68.2, 55.6, 60.0, 51.4]
mean_value = calculate_mean(numbers)
print("Середнє арифметичне: ", mean_value)
```

У цьому прикладі (Код 4) функція *calculate_mean* отримує список чисел *numbers*. Вона обчислює суму всіх чисел у списку, потім визначає кількість чисел за допомогою функції *len*, і нарешті обчислює середнє значення, поділивши суму на кількість. Потім значення середнього виводиться за допомогою функції *print*. В цьому прикладі середнє арифметичне значення списку [54.3, 68.2, 55.6, 60.0, 51.4] буде 57,9.

Код R для обчислення середнього арифметичного значення:

```
mean(c(54.3, 68.2, 55.6, 60.0, 51.4))
```

Зважену середню арифметичну величину використовують у тих випадках, коли варіаційний ряд є досить великим ($n > 30$) і окремі його значення повторюються, а також тоді, коли треба об'єднати середні арифметичні декількох груп. У першому випадку для обчислення зваженої середньої використовують наступну формулу:

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i f_i}{n}, \quad \textbf{(3)}$$

де x_i – значення варіант з i-тої групи, f_i – кількість варіант в i-тій групі.

Приклад 5. Експериментально визначали плодючість дафній і отримали наступні значення: 8, 11, 23, 9, 8, 12, 17, 13, 13, 8, 11, 23, 11, 8, 16, 23, 20, 21, 21, 9, 11, 17 та 13 нащадків. За формулою **(3)** можна отримати зважену середню арифметичну величину плодючості дафній:

$$\bar{x} = \frac{8\times4+11\times4+23\times3+9\times2+12\times1+17\times2+13\times3+16\times1+20\times1+21\times2}{23} = 14$$

(особин).

Середня арифметична кількох однорідних груп обчислюється за подібною формулою:

$$\bar{x} = \frac{\sum \bar{x_i} n_i}{\sum n_i}, \quad \textbf{(4)}$$

де n_i – кількість значень в i-тій групі;

$\bar{x}_i$ – середнє арифметичне значення в *i*-тій групі.

Приклад 6. Вміст гемоглобіну в крові дорослих чоловіків (n_1 = 30) дорівнював 69,8%. Цей показник для іншої групи чоловіків того ж віку (n_2 = 20) склав 64,9%. Потрібно визначити середню арифметичну величину з цих двох середніх. Для вибірок однакового розміру ($n_1 = n_2$) $\bar{x} = (69{,}8 + 64{,}9)/2 = 67{,}4$ %. Якщо розмір однієї вибірки становить 30, а іншої – 20 осіб, то в такому випадку використовується формула **(4)**:

$$\bar{x} = \frac{69{,}8 \times 30 + 64{,}9 \times 20}{30 + 20} = 67{,}8\ \%.$$

Формула зваженої середньої використовується не тільки для полегшення обрахунків при повторюваності варіант або об'єднання середніх, а також для обчислення середніх у тих випадках, коли кожний результат не є рівнозначним і залежить від якоїсь умови (ваги).

Приклад 7. Плодові мушки не вилуплюються з лялечок одночасно. Вилуплення займає декілька днів. Так, на дев'ятий день після відкладення яєць у пробірці вилупилось 2 особини, на десятий – 6, на 11-ий – 10, на 12-ий – 16, на 13-ий – 11, на 14-ий – 5, на 15-ий – 2. Порахуємо зважене середнє, використовуючи як n_i – кількість особин, які вилупились за певний день, а як x_i – день від початку відкладання яєць:

$$\bar{x} = \frac{9 \times 2 + 10 \times 6 + 11 \times 10 + 12 \times 16 + 13 \times 11 + 14 \times 5 + 15 \times 2}{2 + 6 + 10 + 16 + 11 + 5 + 2} = 12.$$

В даному випадку значення зваженої середньої буде вказувати на день, коли вилупилось найбільше особин.

Іншою важливою величиною, яка характеризує вибірку, є **медіана** (**Ме**). **Медіаною** називають значення x_i, розміщене посередині ряду значень, що розставлені в порядку від найменшого до найбільшого. Такий ряд інакше називається варіаційним. Так, якщо всі отримані в ході експерименту значення в дослідній групі виписати в ряд у порядку їх збільшення, то медіаною буде вважатись те значення, яке стоїть в цьому ряді посередині. Порядковий номер, або ранг значення, яке є медіаною для ряду з непарною кількістю значень можна встановити за формулою:

$$Me = x^*_{\frac{n+1}{2}}, \qquad \textbf{(5)}$$

де x^*_i – i-тий елемент варіаційного ряду.

Так, якщо ми маємо ряд з п’яти значень, розміщених в порядку зростання, то медіаною буде 3-тє за порядком значення. Якщо кількість чисел має парне значення, то медіаною буде середнє арифметичне між значеннями, які мають порядкові номери $n/2$ та $(n + 2)/2$. Тобто, половина значень у вибірці буде більша або рівна медіані, а інша – менша або рівна медіані.

Медіану використовують замість середньої арифметичної величини в тих випадках, коли варіаційний ряд є асиметричним. Якщо побудувати криву розподілу для групи з таких даних, то її пік буде зміщеним, на відміну від кривої нормального розподілу. Медіану також використовують тоді, коли дані є дискретними, або тоді коли ми не маємо чіткої «верхньої межі» ряду даних. Наприклад, «плодова мушка виходила з теплової коми впродовж шести і більше хвилин». Зрозуміло, що не має сенсу чекати, доки «оживе» муха, яка, можливо, вже померла.

Проте ми можемо просто порахувати кількість тих мух, які «оживали» більше шести хвилин. Середнє значення з таких даних ми порахувати не зможемо, але медіану – так. Нижче розглянемо два приклади обчислення медіан у вибірці:

Приклад 8.

I. *Випадок з парним числом значень у вибірці*

Припустимо, що ми маємо наступні значення дискретної ознаки:

15, 1, 4, 11, 3, 10, 7, 16, 13, 5, 16, 9, 6, 5.

Цей ряд складається з 14 чисел. Перша дія при обчисленні медіани – ранжування, тобто розміщення значень в порядку зростання:

Номер	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Значення	**1**	**3**	**4**	**5**	**5**	**6**	**7**	**9**	**10**	**11**	**13**	**15**	**16**	**16**

Медіана буде знаходитись між сьомим та восьмим значеннями (виділені сірим кольором) і чисельно дорівнювати середньому значенню між ними:

$$Me = (7 + 9) / 2 = 8.$$

Графічно це буде виглядати наступним чином:

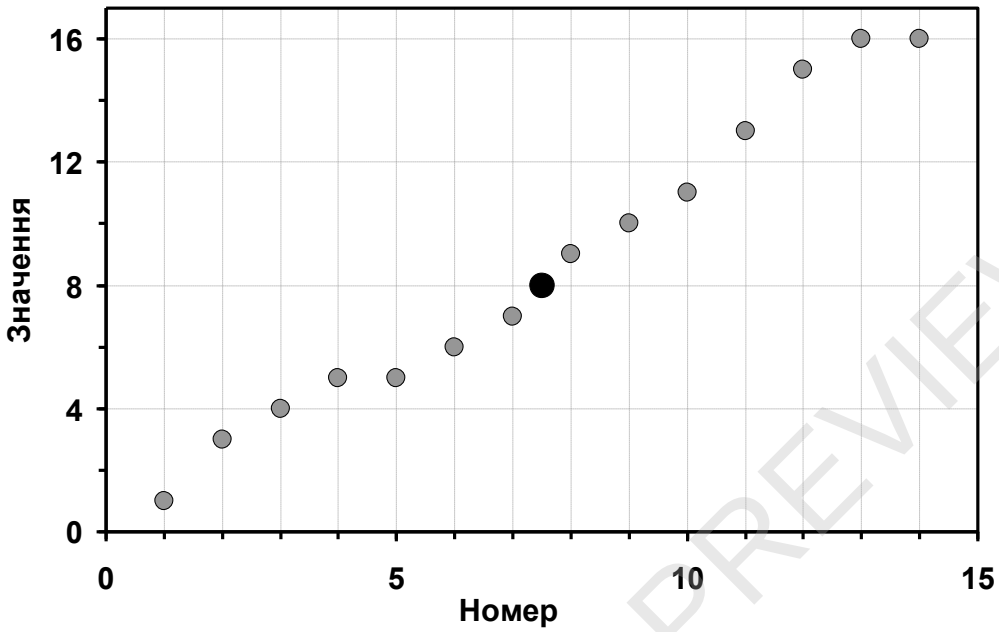


Рис. 1. Графічний метод представлення медіани

Значення медіани на рис. 1 позначене чорною точкою. Бачимо, що по обидва боки від медіани розміщена однакова кількість даних – по сім значень.

II. *Випадок з непарним числом значень у вибірці*

Візьмемо іншу вибірку, яка містить наступні значення:

9, 15, 5, 1, 11, 4, 16, 13, 10, 5, 16, 6, 3.

На відміну, від попередньої вибірки, тут ми маємо непарне число значень. Знову розміщуємо значення в порядку зростання:

Номер	1	2	3	4	5	6	7	8	9	10	11	12	13
Значення	**1,**	**3,**	**4,**	**5,**	**5,**	**6,**	**9,**	**10,**	**11,**	**13,**	**15,**	**16,**	**16**

Медіана дорівнюватиме 9, тому що саме це значення є центральним у наведеному вище ряді. Медіану, як і середнє

арифметичне значення можна знайти за допомогою встановлених стандартних бібліотек *Python* (stats, statistics, numpy) або через написання формули для її обчислення.

Код 1 з використанням бібліотеки ***stats*** наступний:

```
import stats
numbers = [9, 15, 5, 1, 11, 4, 16, 13, 10, 5, 16, 6,
3]
median = stats.median(numbers)
print("Медіана: ", median)
```

Код 2 з використанням бібліотеки ***statistics*** наступний:

```
import statistics
numbers = [9, 15, 5, 1, 11, 4, 16, 13, 10, 5, 16, 6, 3]
median = statistics.median(numbers)
print("Медіана: ", median)
```

Код 3 з використанням бібліотеки ***numpy*** наступний:

```
import numpy
numbers = [54.3, 68.2, 55.6, 60.0, 51.4]
median = numpy.median(numbers)
print("Середнє арифметичне: ", median)
```

Приклад коду Python (код 4) для обчислення медіани без використання бібліотек:

```
def calculate_median(numbers):
    sorted_numbers = sorted(numbers)
    count = len(sorted_numbers)
    middle_index = count // 2

    if count % 2 == 0:
        median = (sorted_numbers[middle_index - 1] +
sorted_numbers[middle_index]) / 2
    else:
        median = sorted_numbers[middle_index]

    return median

# Приклад використання
numbers = [9, 15, 5, 1, 11, 4, 16, 13, 10, 5, 16, 6,
3]
median_value = calculate_median(numbers)
print("Медіана: ", median_value)
```

У цьому прикладі (Код 4) функція *calculate_median* отримує список чисел *numbers*. Вона сортує список за порядком зростання за допомогою функції *sorted*. Потім вона визначає кількість чисел у списку за допомогою функції *len* і знаходить середній індекс. Якщо кількість чисел непарна, медіаною є число з середини відсортованого списку. Якщо кількість чисел парна, медіаною є середнє арифметичне двох чисел з середини. Нарешті, функція повертає обчислене значення медіани. В даному прикладі медіана списку [9, 15, 5, 1, 11, 4, 16, 13, 10, 5, 16, 6, 3] буде 9.

Приклад коду R для обчислення медіани:

```
# Введення даних
numbers <- c(4, 6, 2, 8, 9, 1, 5)

# Обчислення медіани
med <- median(numbers)
print(paste("Медіана: ", med))
```

У цьому прикладі дані введені у вектор *numbers*, після чого використовується функція *median*, щоб обчислити медіану. Результат виводиться на екран за допомогою функції *print*.

Іншою величиною, яка характеризує вибірку, є **мода (mode)**. Мода є однією з основних мір центральної тенденції в статистиці і є значенням, яке найчастіше зустрічається в масиві даних. Вона може бути розглянута як найтиповіше або представницьке значення вибірки. Мода – це число, яке має найбільшу кількість повторень серед даних. Це може бути корисно при описі набору даних, бо вона дозволяє знайти типове значення вибірки.

Мода може бути знайдена в будь-якому наборі даних, якщо є принаймні один повторюваний елемент. Якщо кілька елементів мають однакову кількість повторень, то вибіркова мода вважається будь-яким з цих елементів. Якщо немає жодного повторюваного елемента, то мода вибірки не існує.

Мода є додатковою мірою розподілу вибірки разом з медіаною та середнім значенням. Наприклад, якщо розподіл вибірки має декілька мод, це може вказувати на бімодальний або мультимодальний розподіл.

Мода може бути використана при вивченні медичних даних. Наприклад, при вивченні даних про здоров'я населення можна

використовувати моду для визначення найбільш поширених хвороб або медичних проблем серед пацієнтів. Це може допомогти медичним установам зосередитися на популярних хворобах та забезпечити належне лікування для пацієнтів з цими хворобами.

Моду, на відміну від середнього арифметичного значення і медіани, можна знайти за допомогою бібліотек *Python* statistics і scipy або через написання формули для її обчислення.

Код 1 з використанням бібліотеки ***statistics*** наступний:

```
import statistics
numbers = [9, 15, 5, 1, 11, 4, 16, 13, 10, 5, 16, 6, 3]
mode = statistics.mode(numbers)
print("Мода: ", mode)
```

Код 2 з використанням бібліотеки ***scipy*** наступний:

```
import scipy.stats as st
numbers = [9, 15, 5, 1, 11, 4, 16, 13, 10, 5, 16, 6, 3]
mode = st.mode(numbers)
print("Мода: ", mode)
```

Приклад коду Python (код 3) для обчислення всіх варіантів наявності моди у вибірці:

```
import statistics
numbers = [9, 15, 5, 1, 11, 4, 16, 13, 10, 5, 16, 6, 3]
mode = statistics.multimode(numbers)
if len(mode) == len(numbers):
    print("Data series does not contain MODE!")
else:
    print(f"Mode: {mode}\n")
```

Код 3, на відміну від інших, дає змогу обчислити всі можливі варіанти моди у ряді даних. Він також обчислює моди не тільки для

чисел, але й для даних, виражених словами. В результаті на екран виводиться значення модальності. Якщо існує більше ніж одна мода або її немає, цей код також виводить повідомлення про це. Тому ми рекомендуємо використовувати саме цей код.

Код R для обчислення Моди:

```
    find_mode <- function(x) {
      unique_values <- unique(x)
      tab <- sapply(unique_values, function(val) sum(x
== val))
      max_tab <- max(tab, na.rm = TRUE)

      if (is.na(max_tab) || max_tab == 1) {
        return("немає моди")
      }

      mode_values <- unique_values[tab == max_tab]
      return(mode_values)
    }

    # Define a list of data
    data <- c("Інгібітор", "Активатор", "Інактиватор")

    # Find the mode
    mode_result <- find_mode(data)
cat(mode_result, "\n")
```

Результатом цього коду буде наступне значення:

```
[1] «немає моди»
```

Іноді виникає необхідність у більш дрібному поділі статистичного ряду. Тому, крім медіани виділяють **процентилі** або **квантилі** (*P*), **квартилі** Q (Q_1, Q_2, Q_3), **квінтилі** q1,…, q4 (1/5 ряду)

і **децилі** d1, d2,…, d9 (1/10 ряду). Найчастіше у біології для характеристики ряду використовуються процентилі і квартилі.

Процентилі є характеристиками, які поділяють ряд спостережень на 100 частин. Для цього потрібно 99 процентилей. Процентиль характеризує значення, яке досягається заданим відсотком загальної кількості даних в варіаційному ряді.

P-ий процентиль – це така величина, що *P* % даних є меншими цієї величини і (100 – *P*) % є більшими або рівними. Для того, щоб визначити *P*-й процентиль, необхідно виконати наступне:

1) представити результати спостережень у вигляді варіаційного ряду;

2) обчислити номер *P*-го процентиля у варіаційному ряді:

$$i = \frac{P}{100} \times n \qquad \textbf{(6)}$$

де *P* – значення процентиля;

n – об’єм вибірки;

i –процентиль в ряді спостережень;

3) визначити значення *P*-го процентиля:

а) якщо *i* – ціле число, то *P*-й процентиль є середньою величиною *i*-го і (*i+1*)-го спостережень у варіаційному ряді;

б) якщо *i* неціле число, то *P*-ий процентиль можна визначити за допомогою інтерполяції або округлення.

Приклад 9. Визначити 30-й процентиль для наступного ряду спостережень: 14, 12, 19, 23, 5, 13, 28, 17.

Спочатку із вищевказаних даних формуємо варіаційний ряд:

5, 12, 13, 14, 17, 19, 23, 28

$$i = \frac{30}{100} \times 8 = 2{,}4.$$

Оскільки *i* не є цілим числом, то номер 30-го процентиля в цьому варіаційному ряді визначається як ціла частина від значення 2,4+1=3,4, тобто 3. Отже, 30-м процентилем є значення **13**.

Приклад коду Python для обчислення 30-го процентиля:

```
import numpy
data = input("Введіть дані, розділені комами: ")
data_list = [float(x) for x in data.split(",")]
p30 = numpy.percentile(data_list, 30)
print("30-й процентиль: ", p30)
```

Цей код спочатку запитує користувача ввести дані, розділені комами. Потім він перетворює цей рядок на список чисел і визначає 30-й процентиль цих чисел за допомогою функції *numpy.percentile()*. Нарешті, він виводить 30-й процентиль на екран. Для коректної роботи цього коду потрібно спочатку імпортувати бібліотеку NumPy *(import numpy)*.

Приклад коду R для обчислення 30-го процентиля:

```
data <- c(5, 12, 13, 14, 17, 19, 23, 28)
p30 <- quantile(data, 0.3)
print(p30)
```

Квартилі – це характеристики, які поділяють ряд спостережень на 4 частини. Для цього необхідні 3 квартилі, які позначаються Q_1, Q_2, Q_3.

Q_1 є 25-м процентилем, тобто $Q_1 = P_{25}$.

Q_2 є 50-м процентилем, тобто $Q_2 = P_{50}$, або медіаною.

Q_3 – це 75-й процентиль, тобто $Q_3 = P_{75}$.

Приклад 10. Визначити Q_1, Q_2, Q_3 для наступної вибірки:

109 121 122 129 106 116 125 114.

Спочатку формуємо варіаційний ряд:

106 109 114 116 121 122 125 129.

Позаяк $Q_1 = P_{25}$, то визначаємо 25-й процентиль. Для n = 8 маємо:

$$i = \frac{25}{100} \times 8 = 2$$

А що як i – ціле число, то P_{25} визначається як середнє другого і третього значень у варіаційному ряді:

$$P_{25} = 109 + 114 = 111{,}5 ;$$

Отже, $Q_1 = 111{,}5$.

$Q_2 = P_{50}$ і є також медіаною. Так як n – парне число, то:

$$Q_2 = 116 + 121 = 118{,}5;$$

$Q_3 = P_{75}$, то визначаємо 75-й процентиль:

$$i = \frac{75}{100} \times 8 = 6$$

Через те, що i – ціле число, квартиль P_{75} визначається як середнє шостого і сьомого значень у варіаційному ряді:

$$Q_3 = 122 + 125 = 123{,}5$$

Приклад коду Python для обчислення квартилів.

Для визначення квартилів в Python можна використовувати функцію *numpy.percentile()*. Для цього спочатку потрібно отримати від користувача список даних, який міститиме всі значення вибірки. Далі, за допомогою функції *numpy.percentile()* можна обчислити квартилі.

```
    import numpy as np

    # Отримання списку даних від користувача
    data = input("Введіть значення вибірки через кому:
").split(",")
    data = [int(x) for x in data]

    # Обчислення квартилів
    q1 = np.percentile(data, 25)
    q2 = np.percentile(data, 50)
    q3 = np.percentile(data, 75)

    # Виведення результатів
    print("Квартиль 1:", q1)
    print("Квартиль 2:", q2)
    print("Квартиль 3:", q3)
```

Приклад коду R для обчислення квартилів.

Для визначення квартилів в R можна використати функцію *quantile()*. Квартилі позначаються як 0,25; 0,5 та 0,75 квантилі відповідно.

```
    data <- c(109, 121, 122, 129, 106, 116, 125, 114)
    q1 <- quantile(data, 0.25)
    q2 <- quantile(data, 0.5)
    q3 <- quantile(data, 0.75)
    cat("Квартиль 1:", q1, "\n")
    cat("Квартиль 2:", q2, "\n")
    cat("Квартиль 3:", q3, "\n")
```

Це виведе результат у наступному форматі:

```
Квартиль 1: [значення квартилю 1]
Квартиль 2: [значення квартилю 2]
Квартиль 3: [значення квартилю 3]
```

П’ять базових показників, а саме мінімальне значення (x_{min}), нижній квартиль (Q_1), медіана ($Me=Q_2$), верхній квартиль (Q_3) і максимальне значення (x_{max}), дають достатнє уявлення про особливості ще не оброблених наборів даних. В *R* наявна спеціальна функція *fivenum* для обрахунку згаданих п’яти показників для ряду даних. Код може виглядати наступним чином:

```
x <- c(87.4, 12.3, 69.3, 34.5, 5.7, 27.2, 94.1, 1.9)

# П’ятірка чисел Тюкі
fivenum(x)

# Виправлений варіант з правильними лапками
names <- c("Мінімальне значення",
           "Перший квартиль",
           "Медіана (Другий квартиль)",
           "Третій квартиль",
           "Максимальне значення")

fivenum_df <- data.frame(fivenum(x), row.names = names,
                         fix.empty.names = FALSE)
fivenum_df
```

2.2. Стандартне відхилення, дисперсія та коефіцієнт варіації

У попередньому підрозділі ми бачили, що середні величини дозволяють охарактеризувати вибірку одним числом. Проте значення середньої арифметичної величини не показує мінливості ознаки. У вже згаданому прикладі з довжиною стебла рослин середня арифметична буде одна і та сама для вибірок зі значеннями: 3, 8, 5, 4, 7 та 5, 5, 5, 6, 6 см. Проте видно, що у першій вибірці значення істотно відрізняються одне від одного. Варіація параметрів

сама по собі є важливою характеристикою живого організму чи групи організмів. У багатьох випадках мала варіація означає те, що показник суворо контролюється організмом і його зміна може, наприклад, зменшувати шанс виживання.

У 1894 році один із засновників біометрії Карл Пірсон ввів термін стандартне відхилення (середнє квадратичне відхилення або середня квадратична похибка вимірювань, яку в англомовній літературі позначають як SD – *standard deviation*). Для вибірки середнє квадратичне відхилення обчислюється за наступною формулою:

$$s = \sqrt{\frac{\sum (x_i - \bar{x})^2}{n-1}} \qquad \textbf{(7)}$$

або $s = \sqrt{\frac{\sum x_i^2 - \frac{(\sum x_i)^2}{n}}{n-1}}$, **(8)**

де $\bar{x}$ – середнє арифметичне, x_i – i-те значення, n – розмір вибірки. З цієї формули видно, що при обчисленні стандартного відхилення враховується різниця між окремою варіантою серії досліджень та їх середньої арифметичної величини: $(x_i - \bar{x})$. При цьому кожне відхилення підноситься до квадрату для знищення ефекту взаємної компенсації додатних та від'ємних доданків.

Крім того, слід згадати і те, що стандартне відхилення є квадратним коренем вибіркової дисперсії:

$$s^2 = \frac{\sum (x_i - \bar{x})^2}{n-1} \quad . \qquad \textbf{(9)}$$

Дисперсія є важливим показником варіації і вказує на те, як задалеко одна від одної розміщені варіанти сукупності.

Приклад 11. Активність лактатдегідрогенази в печінці коропа становила: 1,40; 1,47; 1,08; 1,23; 1,12; 1,00 Од/мг білка. Обчислимо стандартне відхилення, використовуючи формулу **(8)**:

$$s = \sqrt{\frac{1{,}40^2 + 1{,}47^2 + 1{,}08^2 + 1{,}23^2 + 1{,}12^2 + 1{,}00^2 - \frac{(1{,}40 + 1{,}47 + 1{,}08 + 1{,}23 + 1{,}12 + 1{,}00)^2}{6}}{6-1}}$$

$$= \sqrt{\frac{1{,}96 + 2{,}16 + 1{,}17 + 1{,}51 + 1{,}25 + 1 - \frac{7{,}3^2}{6}}{5}} = \sqrt{\frac{0{,}17}{5}} = \sqrt{0{,}034} = 0{,}18$$

(Од/мг білка).

Приклад коду Python для обчислення дисперсії і стандартного відхилення вибірки, використовуючи бібліотеку statistics:

```
import statistics
data = [1.40, 1.47, 1.08, 1.23, 1.12, 1.00]

# Обчислення дисперсії
variance = statistics.variance(data)
variance_round = round(variance, 3)
print("Дисперсія: ", variance_round)

# Обчислення стандартного відхилення
standard_deviation = statistics.stdev(data)
standard_deviation_round = round(standard_deviation,
3)
print("Стандартне відхилення: ",
standard_deviation_round)
```

У цьому прикладі використовується модуль *statistics*, який надає функціональність для обчислення статистичних показників, включаючи дисперсію та стандартне відхилення. Функція *statistics.variance* використовується для обчислення дисперсії на основі даних. Вона приймає список чисел і повертає вибіркову

дисперсію. Функція *statistics.stdev* використовується для обчислення стандартного відхилення на основі даних. Вона також приймає список чисел і повертає вибіркове стандартне відхилення. В даному прикладі результати обчислень виводяться за допомогою функції *print*. Ми також застосували функцію *round* до значень дисперсії та стандартного відхилення, щоб округлити їх до третього знаку після коми. У результаті виведення ви побачите округлені значення дисперсії та стандартного відхилення для заданих даних:

```
Дисперсія:  0.035
Стандартне відхилення:  0.186
```

Приклад коду R для обчислення дисперсії і стандартного відхилення малої вибірки:

```
# Введення даних
x <- c(1.4, 1.47, 1.08, 1.23, 1.12, 1.0)

# Обчислення дисперсії
var_x <- var(x, na.rm = TRUE)

# Обчислення стандартного відхилення
sd_x <- sd(x, na.rm = TRUE)

# Виведення результатів
cat("Дисперсія: ", var_x, "\n")
cat("Стандартне відхилення: ", sd_x, "\n")
```

У цьому коді ми спочатку вводимо дані вектора x, потім використовуємо функції *var()* і *sd()* для обчислення дисперсії та стандартного відхилення, відповідно. Параметр *na.rm = TRUE* вказує на те, що будь-які значення *NA* у векторі мають бути

проігноровані при обчисленні. Нарешті, за допомогою функції *cat()* ми виводимо результати.

Слід зауважити і те, що в дослідницькій роботі інколи потрібно порівняти варіацію ознак, значення яких виражаються в різних одиницях вимірювань. Для порівняння варіацій ознак, які виражені в різних одиницях вимірювань, використовують коефіцієнт варіації C*v*, величину якого виражають у відсотках:

$$Cv = \frac{s}{\bar{x}} \times 100\%, \qquad \textbf{(10)}$$

де s – стандартне відхилення, $\bar{x}$ – середнє арифметичне значення.

2.3. Варіація і розподіл

Давньогрецький філософ Геракліт казав «не можна двічі увійти в одну і ту саму річку». Будь-який дослідник природи може підтвердити це досвідом: важко отримати ідентичні цифри при повторі експерименту. Те саме стосується й спостережень. Наприклад, листки одного дерева будуть відрізнятись за площами, плодові мушки однієї лінії – за масою, культури бактерій одного штаму – за активністю амілази чи іншого ферменту. Варіація – характерна особливість будь-якого біологічного об’єкту. Середні значення і показники варіації власне несуть дослідникові інформацію про об’єкт чи явище. Вони також є предметом для подальших підрахунків, наприклад, при статистичних порівняннях.

Варіація не є безмежною. Відомо, що нормальний рівень гемоглобіну в сироватці крові буде відповідати якомусь діапазону значень, так само, як і активність ферментів у тканинах, значення кров’яного тиску тощо. Для будь-якої ознаки, яка варіює, можна знайти найменше і найбільше значення. Це робиться шляхом

розстановки значень у порядку зростання, або, інакше кажучи, ранжуванням. Якщо проаналізувати ряди значень, отриманих після вимірювань, то можна помітити що певна частина значень не буде сильно відрізнятись. Так, ріст більшості дорослих людей знаходиться у межах від 160 см до 180 см. Цю ситуацію можна змоделювати і представити наочно. Уявімо, що ми маємо вибірку з 1000 осіб одного віку. Нехай 500 з них має ріст 160–180 см, 200 чоловік – 150–159 см, інші 200 – 181–190 см, 50 чоловік матимуть ріст від 145 до 149 см, і решта 50 – від 191 до 200 см. Це буде виглядати наступним чином:

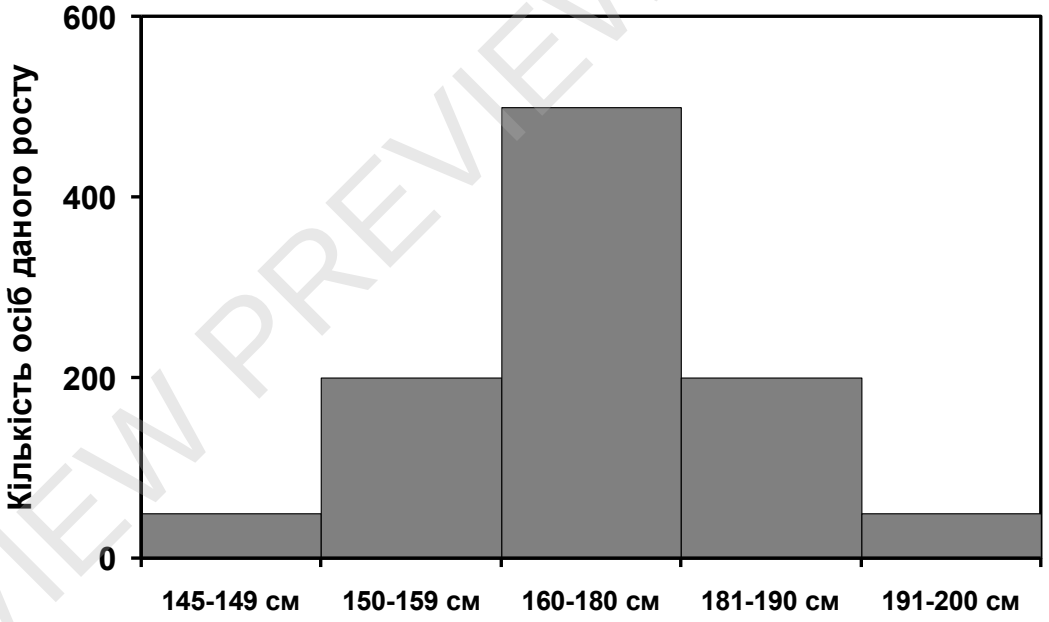


Рис. 2. Стовпчаста діаграма, яка показує варіацію осіб за ростом

Варто зауважити, що інтервали у нашій моделі не рівні, що не зображено на Рис. 2. Для того, щоб показати цю характеристику розподілу, графік можна зобразити наступним чином:

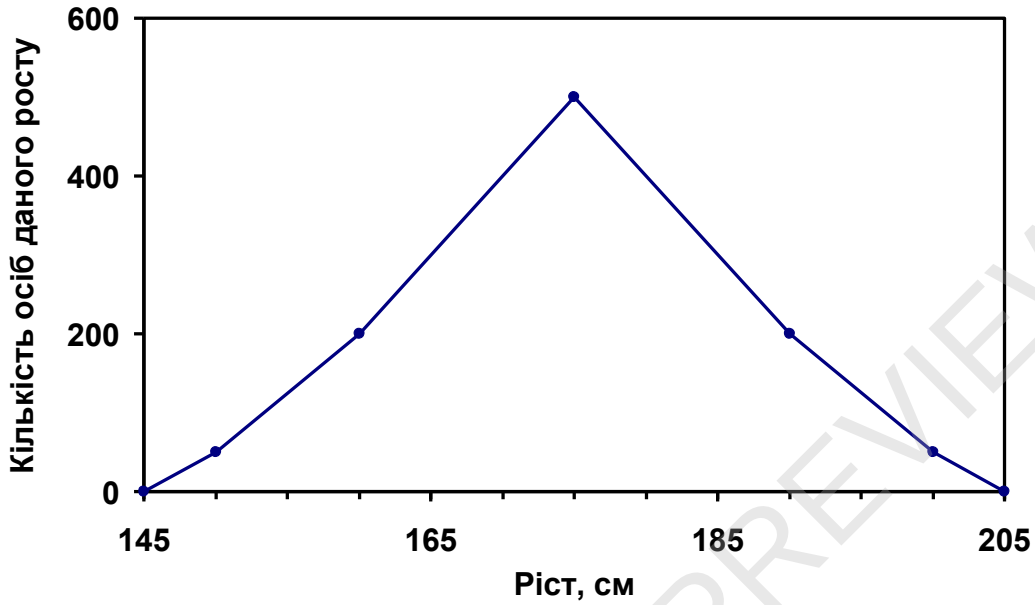


Рис. 3. Полігон частот, який відображує варіацію осіб за ростом

Ще більш наочна і точніша модель представлена нижче (рис. 4). Вона включає дані щодо росту 992 осіб з межами варіації від 145 до 200 см.

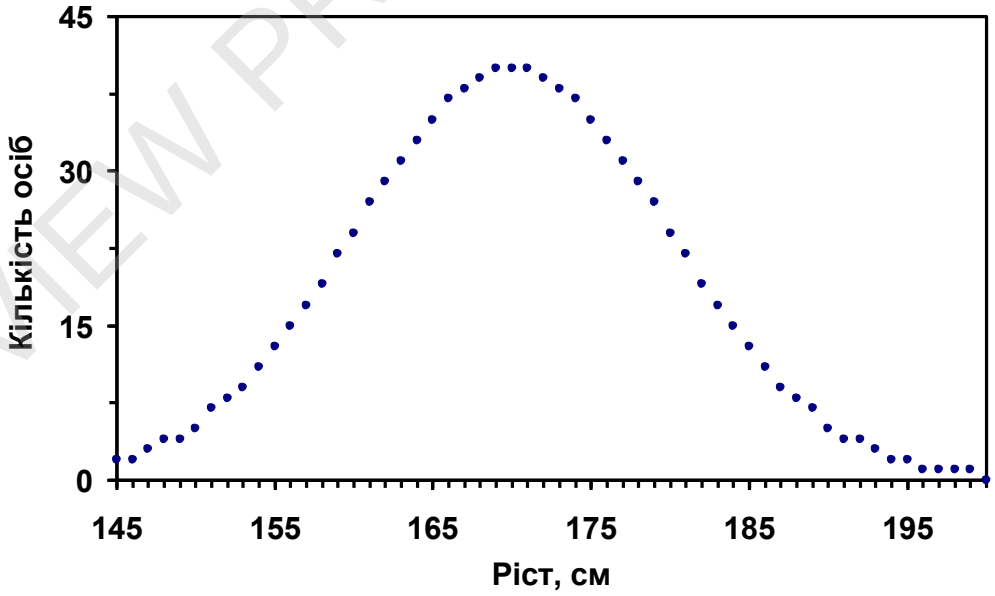


Рис. 4. Розподіл осіб за ростом в генеральній сукупності

Цей графік вже нагадує розподіл ймовірностей випадкової величини, який є предметом математичної статистики.

Крива, зображена на рис. 3, є крива, яка подібна на графік щільності нормального розподілу, що описується рівнянням:

$$f(x) = \frac{1}{s\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2s^2}} , \quad \textbf{(11)}$$

де e – ірраціональна стала, яка приблизно дорівнює **2,7183**, π – число «пі» (3,1416), μ – математичне сподівання, s – стандартне відхилення.

РОЗДІЛ 3. ПОХИБКИ ОЦІНЮВАННЯ ПАРАМЕТРІВ ВИБІРКИ

3.1. Помилка середньої арифметичної величини

У розділі 2 ми моделювали варіацію зросту в дорослих людей. Припустимо, що розподіл росту в дорослих людей є нормальним. В статистиці подібні функції використовується не тільки для опису варіації. У нашій моделі ми використовували групу з 992 особин. В окремих випадках така велика група може бути генеральною сукупністю. Дослідник часто має справу з невеликою кількістю спостережень – вибіркою. Вибірка формується з генеральної сукупності шляхом випадкового відбору. У вказаному нами прикладі ми можемо відібрати декілька осіб зі ростом 182, 174, 155, 168, 176 та 194 см. Середнє значення для цієї вибірки дорівнюватиме приблизно 175 см, тоді як середнє для всієї сукупності було задане як 170 см. Візьмемо іншу вибірку зі значень, які утворюють криву на рис. 3 в розділі 2: наприклад, це індивідууми з ростом 145, 163, 154, 174, 185 та 160 см. Середнє значення буде дорівнювати 163,5 см. Отже, ми бачимо, що обидві вибірки будуть по-різному представляти генеральну сукупність. Для встановлення тих меж, в яких знаходитиметься середнє арифметичне генеральної сукупності, використовують стандартну помилку середньої арифметичної величини (англомовний термін – SEM чи SE, абревіатура від *standard error of the mean*). Її можна обчислити за формулою:

$$m = \frac{s}{\sqrt{n}} \qquad \textbf{(12)}$$

Підставивши формулу **(8)** у формулу **(12)**, отримаємо:

$$m = \sqrt{\frac{n\sum x_i^2 - (\sum x_i)^2}{n^2(n-1)}}. \quad \textbf{(13)}$$

На відміну від стандартного відхилення *s*, стандартна помилка середньої арифметичної величини m (її ще позначають як $S_{\bar{x}}$) не є характеристикою, що описує мінливість ознаки у вибірці. Натомість, стандартна помилка вказує на точність, з якою показник вибірки – середнє арифметичне – представляє (репрезентує) середнє арифметичне для генеральної сукупності.

Приклад 12. Розрахуємо стандартну помилку середнього для наведених вище вибірок, взятих зі сукупності осіб з різним ростом в межах від 145 до 200 см. Використавши формулу **(13)** для першої з наведених вище вибірок, отримаємо:

$$m = \sqrt{\frac{6\cdot(182^2+174^2+155^2+168^2+176^2+194^2)-(182+174+155+168+176+194)^2}{6^2(6-1)}} \approx$$

5,36, для другої відповідно ≈ 5,82.

Після заокруглення значень запис мав би виглядати наступним чином: 175 ± 5 та 164 ± 6. В обох випадках ми бачимо, що середнє арифметичне значення для групи, з якої ми брали обидві вибірки, потрапляє в діапазон середнє арифметичне (для вибірки) ± похибка середнього. Так, середнє арифметичне для групи з 992 чоловік різного росту було задане як 170 см (див. Розділ 2). Для першої вибірки ми отримаємо це значення, якщо віднімемо стандартну похибку – 175 – 5 = 170. Для другої вибірки середнє значення для генеральної сукупності вийде при додаванні стандартної похибки – 164 + 6 = 170 см. Звісно, це не є жорстким правилом, і середнє значення для вибірки може бути віддаленим від середнього для генеральної сукупності на значно більшу відстань, ніж значення

стандартної похибки для вибірки. На цю відстань впливатиме репрезентативність і розмір вибірки, а також відповідність варіації ознаки нормальному розподілу.

Приклад коду Python для обчислення середнього арифметичного значення, стандартного відхилення і похибки середнього арифметичного значення вибірки:

```
import statistics

data = [5.1, 3.5, 1.4, 0.2, 4.9, 3.0, 1.4, 0.2, 4.7, 3.2]

# Обчислення середнього арифметичного значення
mean = statistics.mean(data)
print("Середнє арифметичне: ", mean)

# Обчислення стандартного відхилення
std_dev = statistics.stdev(data)
print("Стандартне відхилення: ", std_dev)

# Обчислення похибки середнього арифметичного значення
n = len(data)
std_error = std_dev / (n**0.5)
print("Похибка середнього: ", std_error)
```

У цьому коді використовується модуль *statistics*, який надає функціонал для обчислення статистичних мір, таких як середнє арифметичне значення та стандартне відхилення.

Функція *statistics.mean* використовується для обчислення середнього арифметичного значення вибірки. Вона приймає список чисел та повертає їх середнє арифметичне.

Функція *statistics.stdev* використовується для обчислення стандартного відхилення вибірки. Вона також приймає список чисел та повертає відхилення.

Для обчислення похибки середнього арифметичного значення, використовується формула *std_error = std_dev / (n**0.5)*, де *std_dev* – стандартне відхилення, а *n* – кількість елементів у вибірці.

В результаті ми отримаємо середнє арифметичне значення, стандартне відхилення та похибку середнього арифметичного значення для даної вибірки.

Нижче наводимо приклад коду R для обчислення середнього арифметичного значення, стандартного відхилення і похибки середнього арифметичного значення. Ми можемо використати вибірки з ростом людей, які використовували вище. Їх можна ввести вручну, або скопіювати з будь-якої електронної таблиці:

```
#Ручне введення, в якому вибірки називаємо sample1 та
sample2:
sample1 <- c(182, 174, 155, 168, 176, 194)
sample2 <- c(145, 163, 154, 174, 185, 160)

#Середні значення для вибірки sample1:
mean(sample1)

#Стандартне відхилення для цієї ж вибірки:
sd(sample1)
```

В більшості програм, зокрема Microsoft Excel або Google Sheets, немає спеціальної функції для обрахунку стандартної похибки середнього. Це також справедливо для основних статистичних

пакетів *R – base* та *stats*. Проте, у *R* дуже легко створити цю функцію самостійно, знаючи наведену вище формулу:

```
se <- function(x) {sd(x)/sqrt(length(x))}
```

Тут, за допомогою функції *sd()* ми рахуємо стандартне відхилення (standard deviation). Функція *length()* рахує розмір вибірки (в буквальному перекладі – «довжину»), а функція *sqrt()* призначена для отримання квадратного кореня (square root). *Sqrt(length(x))* означає взяття квадратного кореня з числа варіант. Далі за допомогою створеної функції рахуємо стандартну похибку середнього для вибірки sample1:

```
se(sample1)
```

Якщо розрахунків небагато, то нескладно декілька разів вручну набрати (або скопіювати) код, який ми прописали у функції вище, замінивши *x* на назву вибірки:

```
sd(sample1)/sqrt(length(sample1))
```

Якщо нам потрібно мати низку показників для вибірки (або вибірок) у формалізованому вигляді, то ми можемо створити для цього спеціальну функцію (назвемо її, наприклад, *multmeas* {від multiple measures}):

```
multmeas <- function(x) {
      c(mean(x), sd(x), se(x), fivenum(x))
}
```

Задаємо назви показників:

```
names <- c(«Середнє арифметичне»,
           «Стандартне відхилення»,
           «Стандартна похибка»,
           «Мінімальне значення»,
           «Перший квартиль»,
           «Медіана»,
           «Третій квартиль»,
           «Максимальне значення»)
```

Зрештою, оформлюємо розрахунки у вигляді таблиці (порівняйте з кодом вище для розрахунку максимального і мінімального значень, і квартилів):

```
data.frame(multmeas(sample1), row.names = names,
                                fix.empty.names =
FALSE)
```

3.2. Довірчий інтервал

Похибка репрезентативності середнього арифметичного вказує те, наскільки середнє арифметичне для вибірки близьке до середнього арифметичного для генеральної сукупності. Часто інформативнішим показником є інтервал, в який потрапляє значення параметру розподілу (середнього арифметичного, дисперсії) із заданою ймовірністю – **довірчий інтервал (Δ*x*)**. Так, у наведеному вище прикладі, де аналізувався зріст 992 осіб, крайні значення становили 145 та 200 см. Візьмемо ще одну випадкову вибірку із зазначених 992 осіб: люди зі зростом 163, 150, 146, 193, 177 та 191 см. Середнє арифметичне для цієї вибірки становитиме 170 см, стандартна похибка – близько 8 см, стандартне відхилення – близько 20 см. В діапазон 170±20 см попадає близько 96,6% значень, які

використовуються у нашій моделі. Для дослідника часто достатньо знати інтервал, в який вкладається не менше як 95% значень генеральної сукупності. В окремих випадках важливо знати інтервал, в якому знаходиться не менше як 97,5% або 99% значень. Відсоток граничних значень – 5, 2,5 або ж 1% називається довірчим рівнем і позначається грецькою літерою α (або літерою p, якщо розраховується в долях одиниці). Існує також інше трактування довірчого рівня, яке ми розглянемо в наступних розділах. Довірчий інтервал можна розрахувати, виходячи з функції розподілу. Для розрахунку довірчого інтервалу використовують не заданий, як на рис. 3 (розділ 2), розподіл, а ідеалізований, в якому середнє значення дорівнює 0, а сам графік функції стає симетричним до осі ординат, на якій відкладається ймовірність випадкової події. Одиниці, які відкладаються на осі абсцис – стандартне відхилення – одне, два, три і так далі (рис. 5).

На графіку (рис. 5) наведені ймовірності (у долях одиниці) потрапляння у вибірку значень, які відхиляються від середнього на n-ну кількість стандартних відхилень. Варто зазначити, що на рис. 2 та 3 показана крива нормального розподілу.

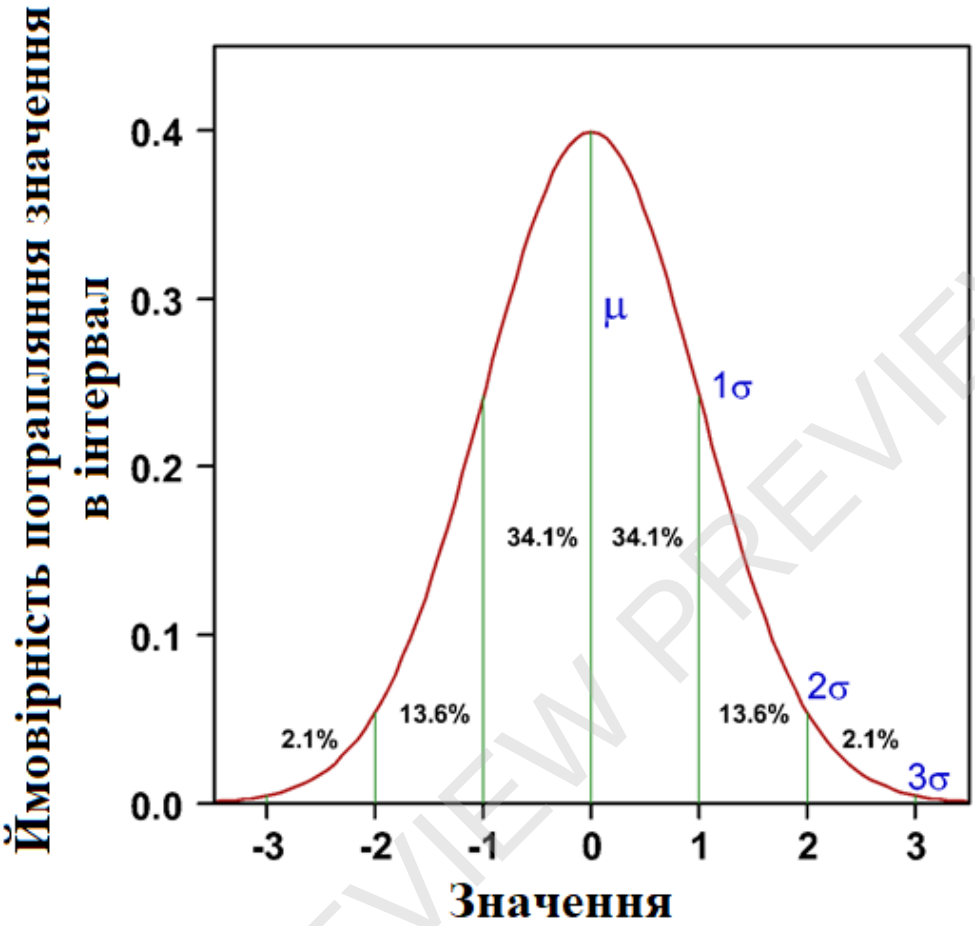


Рис. 5. Крива нормального розподілу для значень абстрактного показника. Крива намальована в R за допомогою функції *dnorm*. Для побудови був заданий розмір сукупності 1000 варіант, середнє арифметичне – 0, стандартне відхилення – 1. На графіку μ – середнє арифметичне для сукупності, σ – стандартне відхилення для сукупності.

Для того, щоб отримати графік на рис. 3, ми задали 170 (см) як середнє та 10 (см) – стандартне відхилення. Додатково ми помножили кожну з отриманих ймовірностей на кількість осіб – 992. Графік на рис. 4 являє собою криву нормального розподілу для випадку, коли середнє арифметичне дорівнює 0. Підраховано, що при такому розподілі 95% значень будуть знаходитись в діапазоні середнє арифметичне ± 1,96 стандартних відхилень, або інакше $\mu \pm 1{,}96 \times \sigma$. 95% Довірчий інтервал для середнього значення

(математичного сподівання) нормального розподілу з відомим середньоквадратичним відхиленням σ обчислюється за формулою:

$$\overline{x} \pm 1{,}96 \times \sigma / \sqrt{n} \qquad \textbf{(14)}$$

незалежно від розміру вибірки.

Якщо ж σ невідоме, то використовується формула:

$$\overline{x} \pm t_P(f) \times s / \sqrt{n} \qquad \textbf{(15)}$$

де $t_P(f)$ – це квантиль порядку $\frac{1+p}{2}$ розподілу Стьюдента з df ступенями свободи. Значення коефіцієнту Стьюдента для деяких значень p і df наведені в табл. 1.

Число ступенів свободи (часто позначається, як df – **d**egrees of **f**reedom) дорівнюватиме $n - 1$, де n – розмір вибірки.

Таблиця 1. Значення t при рівні статистичної значущості (p)

Число ступенів свободи *df*	**Рівень статистичної значущості *p***				
	0,1 (10%)	**0,05 (5%)**	**0,02 (2%)**	**0,01(1%)**	**0,001 (0,1%)**
3	2,35	3,18	4,54	5,84	12,9
4	2,13	2,78	3,75	4,60	8,61
5	2,02	2,57	3,37	4,03	6,86
6	1,94	2,45	3,14	3,71	5,96
7	1,90	2,37	3,00	3,50	5,41
8	1,86	2,31	2,90	3,36	5,04
9	1,83	2,26	2,82	3,25	4,78
10	1,81	2,23	2,76	3,17	4,59
11	1,80	2,20	2,72	3,11	4,44
12	1,78	2,18	2,68	3,06	4,32
13	1,77	2,16	2,65	3,01	4,22
14	1,76	2,15	2,62	2,98	4,14
15	1,75	2,13	2,60	2,95	4,07
16	1,75	2,12	2,58	2,92	4,02
17	1,74	2,11	2,57	2,90	3,97
18	1,73	2,10	2,55	2,88	3,92
19	1,73	2,09	2,54	2,86	3,88
20	1,73	2,09	2,53	2,85	3,85
21	1,72	2,08	2,52	2,83	3,82
22	1,72	2,07	2,51	2,82	3,79
23	1,71	2,07	2,50	2,81	3,77
24	1,71	2,06	2,49	2,80	3,75

25	1,71	2,06	2,49	2,79	3,73
26	1,71	2,06	2,48	2,78	3,71
27	1,70	2,05	2,47	2,77	3,69
28	1,70	2,05	2,47	2,76	3,67
29	1,70	2,05	2,46	2,76	3,66
30	1,70	2,04	2,46	2,75	3,65

При розрахунку довірчого інтервалу ми також маємо бути впевнені, що дані, які аналізуються, підпорядковуються нормальному розподілу, як наприклад дані у використовуваному нами прикладі щодо росту 992 осіб. Якщо дані не підпорядковуються нормальному розподілу (наприклад, смертність організмів, плодючість, кількісні характеристики поведінкових реакцій), то для оцінки особливостей вибірки розраховують медіану, квартилі, а також процентилі, які відповідають 99% та 1% значень.

Приклад 13. Розрахуємо $\overline{x}$ і Δx для даних з активності супероксиддисмутази в печінці карася сріблястого – 154, 153, 125, 136, 142 і 146 Од/мг білка. Для цього використаємо формули **(2)**, **(13)** і **(15)**:

$$\overline{x} = \frac{154+153+125+136+142+146}{6} = 143 \text{ (Од./мг білка).}$$

$$m = \sqrt{\frac{6(154^2+153^2+125^2+136^2+142^2+146^2)-856^2}{36(6-1)}} = \sqrt{\frac{736356-732736}{180}} = 4{,}48$$

(Од/мг білка).

$$\Delta x = 2{,}45 \times 4{,}48 = 11$$

Отже, в результаті обчислень ми отримали наступне значення активності ферменту: $\overline{x} \pm \Delta x = 143 \pm 11$ (Од/мг білка).

Приклад коду Python для обчислення довірчого інтервалу:

```
    import statistics
    import math
    from scipy import stats

    # Запитуємо користувача про дані
    data = input("Введіть дані, розділені комами (неціла
частина через крапку): ")

    # Перетворюємо рядок в список чисел
    data = [float(x) for x in data.split(",")]

    # Обчислюємо середнє значення
    mean = statistics.mean(data)
    mean_round = round(mean, 2)

    # Обчислюємо стандартне відхилення
    stdev = statistics.stdev(data)

    # Задаємо рівень довіри
    confidence_level = 0.95

    # Обчислюємо критичне значення t-розподілу Стьюдента
    n = len(data)
    t_value = abs(stats.t.ppf((1 - confidence_level) /
2, n - 1))
    t_value_round = round(t_value, 2)

    # Обчислюємо довірчий інтервал
    lower = mean - (t_value * stdev / math.sqrt(n))
    lower_round = round(lower, 2)
    upper = mean + (t_value * stdev / math.sqrt(n))
    upper_round = round(upper, 2)

    # Виводимо результат
    print("Середнє арифметичне значення:", mean_round)
    print("Значення Стьюдента:", t_value_round)
    print("Довірчий інтервал для середнього значення:",
(lower_round, upper_round))
```

Користувачеві запропоновано ввести дані у форматі, розділеному комами. Після цього код обчислює середнє значення, стандартне

відхилення, критичне значення t-розподілу Стьюдента та довірчий інтервал. Крім того, додано функцію *round*, за допомогою якої провели заокруглення результатів до 2-го знаку після коми. Отримаємо такий результат:

```
Введіть дані, розділені комами (неціла частина через крапку): >? 154,153,125,136,142,146
Середнє арифметичне значення: 142.67
Значення Стьюдента: 2.57
Довірчий інтервал для середнього значення: (131.14, 154.19)
```

Код R для обчислення довірчого інтервалу з надійністю 95% для введених даних користувачем:

```
    # Введення даних
    x <- c(154, 153, 125, 136, 142, 146)

    # Обчислення середнього значення та стандартного відхилення
    mean_x <- mean(x)
    sd_x <- sd(x)

    # Обчислення кількості спостережень
    n <- length(x)

    # Обчислення довірчого інтервалу
    t_value <- qt((1 + 0.95) / 2, n - 1)
    se <- sd_x / sqrt(n)
    lower <- mean_x - t_value * se
    upper <- mean_x + t_value * se

    # Виведення результатів
    cat("Довірчий інтервал: ", round(lower, 2), "до", round(upper, 2))
```

Результатом виконання цього коду є:

```
Довірчий інтервал: 131.14 до 154.19
```

3.3. Неузгодженості у записах при використанні стандартної похибки середнього

Дуже часто в статтях і дисертаціях зустрічається запис «*M* ± *m*», який, як припускається, пояснює тип представлення даних. Автори під *M* розуміють середню арифметичну величину, а під *m* – стандартну похибку середнього. Втім, якщо немає деталізації, то лишається не зрозумілим, що саме дослідник вкладає у символи *M* та *m*. Для того, щоб запис не залишав сумнівів, ми рекомендуємо в публікаціях давати повне пояснення на кшталт «Всі значення представлені у вигляді середньої ± стандартне відхилення» (у англомовній літературі *Mean* ± *Standart Deviation* (*M* ± *SD*)) чи «Вибіркові характеристики представлені у вигляді середньої ± помилка середньої». (у англомовній літетарурі *Mean* ± *Standart Error of Mean* (*M* ± *SEM* або *M* ± *SE*)).

РОЗДІЛ 4. АНАЛІЗ ДАНИХ, ЯКІ ВИПАДАЮТЬ В ХОДІ ДОСЛІДЖЕНЬ (ПРОМАХИ І СИСТЕМАТИЧНІ ПОХИБКИ)

Значення, які потрапляють у вибірку, можуть дуже істотно відрізнятись за величиною (неоднорідні дані). Візьмемо наш приклад з ростом 992 осіб. Вибірка може виглядати по-різному: може включати осіб з ростом 145, 147, 154 та 199 см, або 145, 165, 181 та 193 см. Звичайно, що у першій вибірці значення 199 см виглядає сумнівним, але зрозуміло, що помилку у вимірюванні зросту важко допустити. В таких випадках кажуть, що вибірка не є репрезентативною, тобто якимось чином у неї потрапили тільки люди низького росту.

Коли дослідник отримує неоднорідні дані, то завжди перед ним стоїть питання: які ж дані слід брати до уваги – одні чи інші? В такому випадку часто самовільно викидаються ті дані, які дослідник вважає невдалими – занадто великими, чи занадто малими. Але такий підхід є неправильним і часто приводить до хибності отриманих результатів.

Щоб таких помилок не було треба скористатись статистичними критеріями. Розглянемо деякі із них.

4.1. Критерій Шовене

Цей критерій для виключення промахів є досить простим. Він використовується у тому випадку, коли відомо, що розподіл генеральної сукупності є нормальним. Для обчислення цього критерію, крім значень середнього арифметичного і стандартного

відхилення, потрібно знайти величину коефіцієнта *u* (різниця між найбільшою (або найменшою) варіантою і середнім арифметичним, поділена на стандартне відхилення):

$$u = \frac{x_{\max} - \bar{x}}{s} \quad \textbf{(16)}$$

або

$$u = \frac{\bar{x} - x_{\min}}{s} \quad \textbf{(17)}$$

Після цього обчислення отриманий коефіцієнт (*u*) співставляють з критичним значенням $u_{кр}$ для *n* значень. Якщо $u \geq u_{кр}$, то це значення (x_{max} чи x_{min}) виключають із подальших обрахунків. Величини $u_{кр}$ наведені в табл. 2.

Таблиця 2. Величини коефіцієнта $u_{кр}$

n	$u_{кр}$	n	$u_{кр}$	n	$u_{кр}$
4	1,534	**13**	2,070	**22**	2,278
5	1,645	**14**	2,100	**23**	2,295
6	1,732	**15**	2,128	**24**	2,311
7	1,803	**16**	2,154	**25**	2,326
8	1,863	**17**	2,178	**26**	2,341
9	1,915	**18**	2,200	**27**	2,355
10	1,960	**19**	2,222	**28**	2,369
11	2,000	**20**	2,241	**29**	2,382
12	2,037	**21**	2,260	**30**	2,394

Якщо у варіаційному ряді є декілька величин, що різко відрізняються від інших, то можна застосовувати інший підхід. За наведеною вище таблицею залежно від числа варіант визначають $u_{кр}$ і обчислюють $\bar{x} + s \times u_{кр}$ та $\bar{x} - s \times u_{кр}$. Якщо значення, яке вважають ймовірним промахом, не входить в інтервал цих обчислень, воно дійсно є промахом і відкидається.

Приклад 14. Використавши критерій Шовене, перевіримо наступні дані: 14,8; 14,2; 14,8; 33,6; 14,1 на наявність промахів.

$\bar{x}$ і s для цієї вибірки будуть дорівнювати, відповідно, 18,3 і 8,6.

Оскільки кількість варіант дорівнює 5, то, використовуючи табл. 2, отримаємо $u_{кр} = 1{,}68$. Далі обчислюємо:

$$\bar{x} + u_{кр} = 18{,}3+8{,}6\times1{,}68=32{,}7$$

Оскільки 33,6>32,7, то значення варіанти 33,6 є промахом і його виключається з наступних обчислень. Тобто, заново обчислюють $\bar{x}$ і s та в кінцевому варіанті як характеристики вибірки використовують обчислені заново параметри.

Код Python для перевірки вибірки на наявність промахів за критерієм Шовене:

```
data = list(map(float, input("Enter the data separated
by a space: ").split()))

# Number of data
n_d = len(data)

# Sorting out a range of data
sort_d = sorted(data)

# Minimum value
min_d = min(data)

# Maximum value
max_d = max(data)

# Arithmetic mean
mean_d = sum(data) / n_d

# Standard deviation
st_d = (sum((i - mean_d) ** 2 for i in data) / (n_d -
1)) ** 0.5
```

```
# Chauvenet’s for minimum value
sh_min = (mean_d - min_d) / st_d

# Chauvenet’s for maximum value
sh_max = (max_d - mean_d) / st_d

# Critical values of Chauvenet’s
sh_crits = {
    4: 1.534,
    5: 1.645,
    6: 1.732,
    7: 1.803,
    8: 1.863,
    9: 1.915,
    10: 1.960,
    11: 2.000,
    12: 2.037,
    13: 2.070,
    14: 2.100,
    15: 2.128,
    16: 2.154,
    17: 2.178,
    18: 2.200,
    19: 2.222,
    20: 2.241,
    21: 2.260,
    22: 2.278,
    23: 2.295,
    24: 2.311,
    25: 2.326,
    26: 2.341,
    27: 2.355,
    28: 2.369,
    29: 2.382,
    30: 2.394
}

sh_crit = sh_crits[n_d]

print(
    f" Number of data: {n_d}\n"
    f" Sorting out a range of data: {sort_d}\n"
    f" Critical Chauvenet’s value: {sh_crit}"
)
if sh_min > sh_crit:
```

```
    print(f" Chauvenet's test: minimum value: {min_d}
is an outlier,\n"
          f" since the calculated Chauvenet's value
{round(sh_min, 3)}\n"
          f" is greater than its critical value
{sh_crit}")
else:
    print(f" Chauvenet's test: minimum value: {min_d}
is NOT an outlier")
if sh_max > sh_crit:
    print(f" Chauvenet's test: maximum value: {max_d}
is an outlier,\n"
          f" since the calculated Chauvenet's value
{round(sh_max, 3)}\n"
          f" is greater than its critical value
{sh_crit}")
else:
    print(f" Chauvenet's test: maximum value: {max_d}
is NOT an outlier")
```

В цьому алгоритмі спочатку обчислюються середнє арифметичне значення *mean* та стандартне відхилення *sd*, які необхідні для обчислення промахів за критерієм Шовене.

Результатом виконання цього коду за вказаним прикладом є:

```
Мінімальне значення:  14.1
Максимальне значення:  33.6
Шовене-статистика для мінімального значення: 0.491
Шовене-статистика для максимального значення: 1.788
Критичне значення: 1.680
Мінімальне значення не є промахом
Максимальне значення є промахом
```

Для перевірки даних на наявність промахів за критерієм Шовене в R немає готового коду. Тому можна скористатись наступним алгоритмом:

```
# Chauvenet's test (min / max)

# Вхідні дані
data <- c(14.8, 14.2, 14.8, 33.6, 14.1)

# Кількість спостережень
n_d <- length(data)

if (n_d < 4) {
  stop("Chauvenet's test не застосовують для n < 4")
}

# Сортування
sort_d <- sort(data)

# Мінімум і максимум
min_d <- min(data)
max_d <- max(data)

# Середнє та стандартне відхилення
mean_d <- mean(data)
st_d <- sd(data)

# Chauvenet's для min і max
sh_min <- (mean_d - min_d) / st_d
sh_max <- (max_d - mean_d) / st_d

# Таблиця критичних значень (для n = 4…30)
sh_crits <- list(
    `4` = 1.534, `5` = 1.645, `6` = 1.732, `7` =
1.803,
    `8` = 1.863, `9` = 1.915, `10` = 1.960, `11` =
2.000,
    `12` = 2.037, `13` = 2.070, `14` = 2.100, `15` =
2.128,
    `16` = 2.154, `17` = 2.178, `18` = 2.200, `19` =
2.222,
    `20` = 2.241, `21` = 2.260, `22` = 2.278, `23` =
2.295,
    `24` = 2.311, `25` = 2.326, `26` = 2.341, `27` =
2.355,
    `28` = 2.369, `29` = 2.382, `30` = 2.394
```

```
  )

  # Вибір критичного значення
  sh_crit <- sh_crits[[as.character(n_d)]]

  if (is.null(sh_crit)) {
    stop("Немає табличного Chauvenet's значення для n
=", n_d)
  }

  # Вивід результатів
  cat("Кількість даних:", n_d, "\n")
  cat("Відсортовані дані:", sort_d, "\n")
  cat("Середнє значення:", round(mean_d, 3), "\n")
  cat("Стандартне відхилення:", round(st_d, 3), "\n")
  cat("Критичне  Chauvenet's  значення:",  sh_crit,
"\n\n")

  if (sh_min > sh_crit) {
    cat("Min =", min_d, "→ OUTLIER (", round(sh_min,
3), ">", sh_crit, ")\n")
  } else {
    cat("Min =", min_d, "→ не є промахом\n")
  }

  if (sh_max > sh_crit) {
    cat("Max =", max_d, "→ OUTLIER (", round(sh_max,
3), ">", sh_crit, ")\n")
  } else {
    cat("Max =", max_d, "→ не є промахом\n")
  }
```

В результаті виконання цього R-коду ми отримаємо результат:

```
Min = 14.1 → не є промахом
Max = 33.6 → OUTLIER ( 1.788 > 1.645 )
```

4.2. Q-критерій Діксона

При його використанні отримані результати вимірювань записують у варіаційний ряд за збільшенням:

$$x_1, x_2, \ldots x_n \ (x_1 < x_2 < \ldots < x_n)$$

Цей критерій визначається як:

$Q = \frac{x_2 - x_1}{x_n - x_1}$ (для мінімального значення) **(18)**

або $Q = \frac{x_n - x_{n-1}}{x_n - x_1}$ (для максимального значення). **(19)**

Проте розрахунок за цими формулами буде коректним тільки для $n \epsilon \{3,\ldots,7\}$. При $n \epsilon \{8,\ldots,10\}$ в знаменнику повинна стояти різниця між ймовірним промахом і значенням, найближчим до максимального (або мінімального) значення:

$Q = \frac{x_2 - x_1}{x_{n-1} - x_1}$ (для мінімального значення); **(20)**

$Q = \frac{x_n - x_{n-1}}{x_n - x_2}$ (для максимального значення). **(21)**

Значення Q порівнюють з критичним значенням критерію, наведеним у табл. 3. Якщо критичне значення менше дослідного, то ймовірний промах підтверджується. При цьому як рівень статистичної значущості p беруть 0,10, а не 0,05. Зазвичай на промахи перевіряють мінімальне і максимальне значення. Після відкидання промахів знову повторюють обчислення.

Таблиця 3. Критичні значення $Q_{кр}$ при різних рівнях статистичної значущості p і числі вимірювань n

n	***Рівень статистичної значущості p***	
	0,05	**0,01**
3	0,970	0,994
4	0,829	0,926
5	0,710	0,821
6	0,625	0,740
7	0,568	0,680
8	0,526	0,634
9	0,493	0,598
10	0,466	0,568
11	0,444	0,542
12	0,426	0,522
13	0,410	0,503
14	0,396	0,488

15	0,384	0,475
16	0,374	0,463
17	0,365	0,452
18	0,356	0,442
19	0,349	0,433
20	0,342	0,425
21	0,337	0,418
22	0,331	0,411
23	0,326	0,404
24	0,321	0,399
25	0,317	0,393
26	0,312	0,388
27	0,308	0,384
28	0,305	0,380
29	0,301	0,376
30	0,298	0,372

Приклад 15. Використавши критерій Діксона, перевіримо на наявність промахів наступні дані: 14,8; 14,2; 14,8; 33,6; 14,1. $\bar{x}$ і s для цієї вибірки будуть дорівнювати, відповідно, 18,3 і 8,6.

1) із поданих даних формуємо варіаційний ряд:

14,1; 14,2; 14,8; 14,8; 33,6

2) за формулою **(21)** обчислюємо Q:

$$Q = \frac{33,6 - 14,8}{33,6 - 14,1} = 0,964$$

Порівнюючи Q, яке ми отримали в результаті обчислень (0,964), з критичним значенням $Q_{кр}$ (0,642) при $n=5$ і рівні статистичної значущості $p<0,10$ (табл. 3), значення 33,6 є промахом.

Код Python для перевірки вибірки на наявність промахів за критерієм Діксона:

```
    data = list(map(float, input("Enter the data
separated by a space: ").split()))

    # Number of data
    n_d = len(data)

    # Sorting out a range of data
    sort_d = sorted(data)

    # Minimum value
    min_d = min(data)

    # Maximum value
    max_d = max(data)

    # Arithmetic mean
    mean_d = sum(data) / n_d

    # Standard deviation
    st_d = (sum((i - mean_d) ** 2 for i in data) / (n_d
- 1)) ** 0.5

    # Dixon's for the minimum value
    d_min = (sort_d[1] - min_d) / (max_d - min_d)

    # Dixon's for maximum value
    d_max = (max_d - sort_d[-2]) / (max_d - min_d)

    p_05 = 0.05

    # Dixon's critical values
    d_crits = {
        4: 0.829,
        5: 0.710,
        6: 0.625,
        7: 0.568,
        8: 0.526,
        9: 0.493,
        10: 0.466,
        11: 0.444,
        12: 0.426,
        13: 0.410,
```

```
        14: 0.396,
        15: 0.384,
        16: 0.374,
        17: 0.365,
        18: 0.356,
        19: 0.349,
        20: 0.342,
        21: 0.337,
        22: 0.331,
        23: 0.326,
        24: 0.321,
        25: 0.317,
        26: 0.312,
        27: 0.308,
        28: 0.305,
        29: 0.301,
        30: 0.298
    }

    d_crit = d_crits[n_d]

    print(
        f"Number of data: {n_d}\n"
        f"Sorting out a range of data: {sort_d}\n"
        f"Dixon's critical value: {d_crit}"
    )
    if d_min > d_crit:
        print(f"Dixon's Q test: minimum value: {min_d}
is an outlier of the p < {p_05}")
    else:
        print(f"Dixon's Q test: minimum value: {min_d}
is not an outlier of the p < {p_05}")
    if d_max > d_crit:
        print(f"Dixon's Q test: maximum value: {max_d}
is an outlier of the p < {p_05}")
    else:
        print(f"Dixon's Q test: maximum value: {max_d}
is not an outlier of the p < {p_05}")
```

Результатом виконання цього коду за вказаним прикладом є:

```
Мінімальне значення:  14.1
Максимальне значення:  33.6
Q-статистика для мінімального значення: 0.143
Q-статистика для максимального значення: 0.964
Критичне значення: 0.642
Мінімальне значення не є промахом
```

Максимальне значення є промахом

Для перевірки даних на наявність промахів за критерієм Діксона в R-програмі можна скористатись наступним кодом:

```
# Вхідні дані
data <- c(14.8, 14.2, 14.8, 33.6, 14.1)

# Number of data
n_d <- length(data)

# Sorting the data
sort_d <- sort(data)

# Minimum and maximum values
min_d <- min(data)
max_d <- max(data)

# Arithmetic mean
mean_d <- mean(data)

# Standard deviation (unbiased)
st_d <- sd(data)

# Dixon's Q for minimum value
d_min <- (sort_d[2] - min_d) / (max_d - min_d)

# Dixon's Q for maximum value
d_max <- (max_d - sort_d[n_d - 1]) / (max_d - min_d)

# Significance level
p_05 <- 0.05

# Dixon's critical values (α = 0.05)
d_crits <- list(
  `4` = 0.829,
  `5` = 0.710,
  `6` = 0.625,
  `7` = 0.568,
  `8` = 0.526,
  `9` = 0.493,
```

```
  `10` = 0.466,
  `11` = 0.444,
  `12` = 0.426,
  `13` = 0.410,
  `14` = 0.396,
  `15` = 0.384,
  `16` = 0.374,
  `17` = 0.365,
  `18` = 0.356,
  `19` = 0.349,
  `20` = 0.342,
  `21` = 0.337,
  `22` = 0.331,
  `23` = 0.326,
  `24` = 0.321,
  `25` = 0.317,
  `26` = 0.312,
  `27` = 0.308,
  `28` = 0.305,
  `29` = 0.301,
  `30` = 0.298
)

# Select critical value
d_crit <- d_crits[[as.character(n_d)]]

# Output
cat("Number of data:", n_d, "\n")
cat("Sorted data:", sort_d, "\n")
cat("Dixon's critical value:", d_crit, "\n")

if (d_min > d_crit) {
  cat("Тест Діксона Q: мінімальне значення:", min_d, "є
промахом при p <", p_05, "\n")
} else {
  cat("Тест Діксона Q: мінімальне значення:", min_d,
"не є промахом при p <", p_05, "\n")
}

if (d_max > d_crit) {
  cat("Тест Діксона Q: максимальне значення:", max_d,
"є промахом при p <", p_05, "\n")
} else {
  cat("Тест Діксона Q: максимальне значення:", max_d,
"не є промахом при p <", p_05, "\n")
}
```

Код бере вектор числових даних, обчислює їх кількість, сортує, знаходить мінімум, максимум, середнє та стандартне відхилення. Далі він розраховує статистики Діксона Q для найменшого та найбільшого значення, щоб перевірити їх на можливість бути викидами. Зі списку критичних значень Dixon's Q для рівня значущості 0,05 вибирається те, яке відповідає розміру вибірки. Потім отримані значення `d_min` і `d_max` порівнюються з критичним значенням, і програма виводить висновок, чи є мінімальне або максимальне число викидом за тестом Діксона.

4.3. Ізоляційний ліс

Ізоляційний ліс (ІЛ, IF) ізолює спостереження шляхом випадкового розбиття простору ознак; аномалії легше ізолювати, і тому вони отримують вищі оцінки за аномалії. Він непараметричний, добре масштабується і працює з простими наборами ознак, що робить його гарним методом за замовчуванням для виявлення аномалій. Для невеликих вибірок він все ще може бути ефективним, якщо (а) зберегти модель простою і (б) доповнити її надійною попередньою обробкою і консервативним порогом.

Інженерія ознак (з урахуванням часу, але легка):

- Залишок від біжучої медіани (надійне видалення тренду).
- Перша різниця (фіксує різкі стрибки).
- Надійний z-рахунок за допомогою MAD (Median Absolute Deviation) для величин, що не залежать від масштабу.

Ці три характеристики дозволяють IF виявити (i) стрибки відносно локального тренду, (ii) раптові стрибкоподібні зміни та (iii)

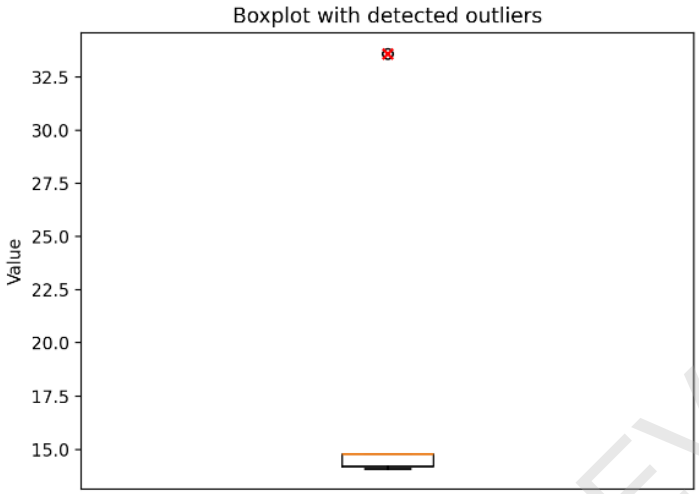
незвично великі значення відносно глобальної мінливості – без використання важких моделей часових рядів.

Наступний код можна використовувати в Python для перевірки даних на наявність викидів за допомогою тесту IF:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.ensemble import IsolationForest

def detect_outliers_iforest(series, random_state=42,
plot=True):
    x = pd.Series(series, dtype=float)
    n = len(x)

# Robust features
    if n >= 9:
        w = max(5, (n // 4) | 1)
    else:
        w = max(3, n | 1)
    trend = x.rolling(window=w, center=True,
min_periods=1).median()
    resid = x - trend
    diff1 = x.diff().fillna(0.0)
    med = x.median()
    mad = np.median(np.abs(x - med))
    z_mad = 0.6745 * (x - med) / (mad + 1e-12)

    X = np.c_[resid.values, diff1.values,
z_mad.values]

    if n < 8 or np.allclose(mad, 0.0):
        is_outlier = np.abs(z_mad.values) > 3.0
        score = np.abs(z_mad.values)
        res = pd.DataFrame({
            "index": np.arange(n),
            "value": x.values,
            "score": score,
            "is_outlier": is_outlier
        })
```

```
    else:
        q1, q3 = x.quantile([0.25, 0.75])
        iqr = q3 - q1
        lower, upper = q1 - 1.5 * iqr, q3 + 1.5 * iqr
        cont_guess = np.mean((x < lower) | (x >
upper))
        contamination = float(np.clip(cont_guess,
0.01, 0.5))
        iso = IsolationForest(n_estimators=200,

contamination=contamination,

random_state=random_state)
        iso.fit(X)
        scores = -iso.score_samples(X)
        preds = iso.predict(X)
        res = pd.DataFrame({
            "index": np.arange(n),
            "value": x.values,
            "score": scores,
            "is_outlier": (preds == -1)
        })

# Plot boxplot
    if plot:
        plt.figure()
        plt.boxplot(x, vert=True, showfliers=True)
        mask = res["is_outlier"].values
        if mask.any():
            plt.scatter(np.repeat(1, mask.sum()),
x[mask], marker='x', color="red", zorder=3)
        plt.title("Boxplot with detected outliers")
        plt.ylabel("Value")
        plt.xticks([])
        plt.show()

    return res

# Example
data = [14.1, 14.2, 14.8, 14.8, 33.6]
print(detect_outliers_iforest(data))
```

Результат:



	index	value	score	is_outlier
0	0	14.1	0.786917	False
1	1	14.2	0.674500	False
2	2	14.8	0.000000	False
3	3	14.8	0.000000	False
4	4	33.6	21.134333	**True**

Повернуті результати включають:

score: вищий означає «більш аномальний»;

is_outlier: булевий прапорець, використовуючи вивчений поріг.

Ставтеся до позначених точок як до кандидатів на помилки («промахи»), а не як до істини в останній інстанції; завжди проводьте перехресну перевірку зі знаннями предметної області або необробленими журналами/примітками приладів.

Пропоновані значення за замовчуванням (для невеликих вибірок, n ≈ 10-50):

Ізоляційний ліс: ntrees=200, підвибірка за замовчуванням, фіксована випадкова вибірка для відтворюваності. Надійне масштабування за допомогою MAD з малим епсилоном, щоб уникнути ділення на нуль.

Переваги та застереження:

- Стійкий до негаусівського шуму, потребує мінімального налаштування.

- При дуже малих n (n<8) оцінки можуть бути чутливими; зберігайте запасний варіант і переглядайте результати вручну.

- Якщо присутня сильна сезонність, розгляньте можливість додавання сезонних залишків перед IF.

Наступний код можна використовувати в R-програмі для перевірки даних на наявність промахів за допомогою тесту IF:

```
suppressPackageStartupMessages(library(isotree))

detect_outliers_iforest <- function(x, seed = 42, plot
= TRUE) {
  x <- as.numeric(x)
  n <- length(x)
  if (n >= 9) {
    k <- max(5, floor(n / 4))
    if (k %% 2 == 0) k <- k + 1
  } else {
    k <- ifelse(n >= 3, 3, 1)
  }
  trend <- if (k > 1) stats::runmed(x, k = k, endrule
= "median") else rep(median(x), n)
  resid <- x - trend
  diff1 <- c(0, diff(x))
  med <- median(x)
  mad_val <- stats::mad(x, constant = 1)
  z_mad <- 0.6745 * (x - med) / (mad_val + 1e-12)
  X <- cbind(resid = resid, diff1 = diff1, z_mad =
z_mad)
  if (n < 8 || abs(mad_val) < .Machine$double.eps) {
    is_outlier <- abs(z_mad) > 3
    score <- abs(z_mad)
    res <- data.frame(
      index = seq_len(n) - 1L,
      value = x,
```

```
      score = score,
      is_outlier = is_outlier
    )
  } else {
    qs <- stats::quantile(x, probs = c(0.25, 0.75))
    iqr <- qs[2] - qs[1]
    lower <- qs[1] - 1.5 * iqr
    upper <- qs[2] + 1.5 * iqr
    cont_guess <- mean(x < lower | x > upper)
    contamination <- max(0.01, min(0.5, cont_guess))
    set.seed(seed)
    iso <- isotree::isolation.forest(X, ntrees = 200)
    scores <- as.numeric(predict(iso, X))
    thr <- stats::quantile(scores, probs = 1 -
contamination)
    is_outlier <- scores > thr
    res <- data.frame(
      index = seq_len(n) - 1L,
      value = x,
      score = scores,
      is_outlier = is_outlier
    )
  }
# Plot boxplot
  if (plot) {
    boxplot(x, main = "Boxplot with detected
outliers", ylab = "Value")
    mask <- res$is_outlier
    if (any(mask)) {
      points(rep(1, sum(mask)), x[mask], pch = 4, col
= "red")
    }
  }
  return(res)
}
# Example
data <- c(14.1, 14.2, 14.8, 14.8, 33.6)
print(detect_outliers_iforest(data))
```

РОЗДІЛ 5. ПЕРЕВІРКА ВИБІРКИ НА НОРМАЛЬНІСТЬ РОЗПОДІЛУ ДАНИХ

5.1. Загальні уявлення про критерії перевірки вибірки на нормальний розподіл даних

Для визначення закону розподілу певної величини, перш за все, потрібно віднести її або до дискретної, або неперервної. Більшість величин, які вимірюються у експериментальній біології, вважаються неперервними.

Дослідники на практиці найчастіше зустрічаються з малою вибіркою ($4 \le n \le 30$) і міркувати про нормальність розподілу даних є досить важко.

За міжнародним стандартом **ISO 5479-97 (**The International Organization for Standardization) вважається, що при кількості вимірювань $n<10$ перевірити гіпотезу про вид розподілу результатів вимірювань неможливо. При числі даних $10<n<30$ також важко судити про вид розподілу, тому для перевірки відповідності даних нормальному розподілу використовують певний статистичний критерій (складовий критерій *d*, критерій Шапіро-Вілка, критерій Колмогорова, критерій Андерсона–Дарлінга, критерій Пірсона χ^2 та інші). За міжнародним стандартом **ISO 5479-97** для перевірки даних на нормальний розподіл слід використовувати складовий критерій *d*, *W*-критерій Шапіро-Вілка, критерії перевірки на симетричність і на значення ексцесу (див. пункт 5.4), критерій Андерсона–Дарлінга. Цей стандарт не рекомендує критерій χ^2 і подібних до нього, бо вони

підходять тільки для згрупованих даних. Проте критерій χ^2 може бути використаний для великих вибірок ($n>100$).

Перш ніж перевіряти нормальність розподілу даних за наведеними вище критеріями, можна також перевірити, чи виконується така умова:

$$\bar{x} \approx Me.$$

Якщо між ними різниця 10-20%, то розподіл даних є відмінним від нормального. Проте якщо медіана і середнє арифметичне значення подібні – це все одно ще не свідчитиме про нормальний розподіл даних.

5.2. Критерій Андерсона–Дарлінга

Критерій Андерсона–Дарлінга є статистичним методом перевірки на нормальний розподіл вибірки. Цей тест є одним із найчутливіших тестів для перевірки на нормальний розподіл, особливо для малих вибірок.

Ідея критерію Андерсона–Дарлінга полягає у порівнянні фактичного розподілу вибірки з теоретичним нормальним розподілом. Числове значення статистики тесту є мірою відстані між емпіричною функцією розподілу вибірки та теоретичною функцією розподілу, тобто якщо це значення більше критичного значення, то можна стверджувати, що вибірка не має нормального розподілу.

Критичні значення визначаються в залежності від рівня значущості тесту та розміру вибірки. Рівень значущості (p-значення) вказує на ймовірність того, що вибірка не є нормально

розподіленою, а якщо це значення менше заданого рівня значущості, то можна стверджувати, що вибірка має нормальний розподіл.

Якщо p-значення менше вибраного рівня значущості (зазвичай 0,05), то нульову гіпотезу про нормальний розподіл можна відхилити і встановити, що вибірка не належить до нормального розподілу. Навпаки, якщо p-значення більше вибраного рівня значущості, то немає достатніх доказів, щоб відхилити нульову гіпотезу, тому вибірку можна вважати прийнятною для нормального розподілу.

Послідовність дій виконання тесту Андерсона–Дарлінга:

1. Сформулювати нульову і альтернативну гіпотези.

Нульова гіпотеза (H0): вибірка походить з нормального розподілу.

Альтернативна гіпотеза (H1): вибірка не походить з нормального розподілу.

2. Обчислити середнє значення (m) та стандартне відхилення (s) для малої вибірки.

3. Відсортувати значення вибірки в порядку зростання.

4. Обчислюємо стандартизоване значення випадкової величини (z), яке виражається в одиницях стандартного відхилення від середнього значення. Воно обчислюється як різниця між спостереженням і середнім значенням вибірки, поділеним на стандартне відхилення. Значення z можуть бути використані для порівняння значень з різних розподілів, оскільки вони мають однакову шкалу.

$$Z = (x_i - m) / s \quad \textbf{(22)}$$

5. Обчислюємо значення функції нормального розподілу для заданих значень z, використовуючи стандартну нормальну функцію розподілу:

$$F(z) = \int_{-\infty}^{z} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt \quad (23)$$

Отже, F є кумулятивною функцією або функцією розподілу стандартного нормального розподілу. Це значення представляє ймовірність, що випадкова змінна буде менше або дорівнюватиме z при заданому розподілі з конкретними параметрами. Зазвичай F лежить у діапазоні від 0 до 1.

6. Обчислити масив (f_{compl}), що містить ймовірності того, що значення вибірки більші за відповідні значення нормальної випадкової величини з такими ж самими середнім і стандартним відхиленням, як вибірка:

$$f_{compl} = 1 - F \quad (24)$$

7. Обчислюємо міру відхилення емпіричної функції розподілу (w) від теоретичної функції розподілу. Кожен елемент вектору w обчислюється як логарифм двох ймовірностей: ймовірності, що спостереження знаходиться лівіше або правіше поточного значення, у порівнянні з теоретичною функцією розподілу. Значення w більші для вибірок, які відхиляються від теоретичного розподілу, і менші для вибірок, які краще відповідають теоретичному розподілу:

$$w = (2 \times n_i - 1) / n \times (ln(F) + ln(f_{compl})), \quad (25)$$

де n_i – порядеовий номер значення із масиву даних.

8. Обчислюємо критерій Андерсона–Дарлінга за формулою:

$$A^2 = -10 - (w_1 + \ldots + w_n) \quad (26)$$

9. Порівнюємо наше значення критерію A^2 із критичним ($A^2_{крит}$) із відповідним значенням рівнів значущості α (табл. 4).

Табл. 4. Критичні значення критерію Андерсона–Дарлінга

Рівень (α)	**n < 15**	**15 ≤ n < 25**	**n ≥ 25**
0,10	0,632	0,760	0,873
0,05	0,752	0,906	1,038
0,025	0,873	1,065	1,223
0,01	1,029	1,223	1,472

Якщо $A^2 < A^2_{крит}$, то на рівні значущості α немає достатніх доказів для відхилення нульової гіпотези про те, що вибірка з нормального розподілу. Якщо $A^2 \geq A^2_{крит}$, то на рівні значущості α можна відхилити нульову гіпотезу на користь альтернативної гіпотези про те, що вибірка не походить з нормального розподілу.

5.3. Статистичний критерій *W* (критерій Шапіро-Вілка)

Цей критерій є одним з найбільш чутливих щодо перевірки даних на їхній нормальний розподіл. Його застосовують при 10≤n<30.

Перевірку гіпотези про те, що дані мають нормальний розподіл здійснюють в наступній послідовності:

1) результати досліджень розміщують у вигляді послідовності:

$$x_1 \leq x_2 \leq x_n,$$

де *n* – число досліджень;

2) обчислюють значення величини *SS* (сума квадратів – *Sum of Squares*):

$$SS = \sum_{i=1}^{n} {x_i}^2 - \frac{1}{n}\left(\sum_{i=1}^{n} x_i\right)^2 ; \qquad \textbf{(27)}$$

3) обчислюють значення величини b за формулою:

$$b = \sum_{i=1}^{k} a_{n-i+1}\,(x_{n-i+1} - x_i)\,, \qquad \textbf{(28)}$$

де значення коефіцієнтів a_{n-i+1} для $i = 1...k$ можна взяти із табл. 5.

Таблиця 5. Значення коефіцієнтів a_{n-i+1}

i	***n***						
	10	**11**	**12**	**13**	**14**	**15**	**16**
1	0,5739	0,5601	0,5475	0,5359	0,5251	0,5150	0,5056
2	0,3291	0,3315	0,3325	0,3325	0,3318	0,3306	0,3290
3	0,2141	0,2260	0,2347	0,2412	0,2460	0,2495	0,2521
4	0,1224	0,1429	0,1586	0,1707	0,1802	0,1878	0,1939
5	0,0399	0,0695	0,0922	0,1099	0,1240	0,1353	0,1447
6			0,0303	0,0539	0,0727	0,0880	0,1005
7					0,0240	0,0433	0,0593
8							0,0196
i	**17**	**18**	**19**	**20**	**21**	**22**	**23**
1	0,4968	0,4886	0,4808	0,4734	0,4643	0,4590	0,4542
2	0,3273	0,3253	0,3232	0,3211	0,3185	0,3156	0,3126
3	0,2540	0,2553	0,2561	0,2565	0,2578	0,2571	0,2563
4	0,1988	0,2027	0,2059	0,2085	0,2119	0,2131	0,2139
5	0,1524	0,1587	0,1641	0,1686	0,1736	0,1764	0,1787
6	0,1109	0,1197	0,1271	0,1334	0,1399	0,1443	0,1480
7	0,0725	0,0837	0,0932	0,1013	0,1092	0,1150	0,1201
8	0,0359	0,0496	0,0612	0,0711	0,0804	0,0878	0,0941
9		0,0163	0,0303	0,0422	0,0530	0,0618	0,0696
10				0,0140	0,0263	0,0368	0,0459
11						0,0122	0,0228
12							0,0000
i	**24**	**25**	**26**	**27**	**28**	**29**	**30**
1	0,4493	0,4450	0,4407	0,4366	0,4328	0,4291	0,4254
2	0,3098	0,3069	0,3043	0,3018	0,2992	0,2968	0,2944
3	0,2554	0,2543	0,2533	0,2522	0,2510	0,2499	0,2487
4	0,2145	0,2148	0,2151	0,2152	0,2151	0,2150	0,2148
5	0,1807	0,1822	0,1836	0,1848	0,1857	0,1864	0,1870
6	0,1512	0,1539	0,1563	0,1584	0,1601	0,1616	0,1630
7	0,1245	0,1283	0,1316	0,1346	0,1372	0,1395	0,1415
8	0,0997	0,1046	0,1089	0,1128	0,1162	0,1192	0,1219
9	0,0764	0,0823	0,0876	0,0923	0,0965	0,1002	0,1036
10	0,0539	0,0610	0,0672	0,0728	0,0778	0,0822	0,0862
11	0,0321	0,0403	0,0476	0,0540	0,0598	0,0650	0,0697
12	0,0107	0,0200	0,0284	0,0358	0,0424	0,0483	0,0537
13		0,0000	0,0094	0,0178	0,0253	0,0320	0,0381
14				0,0000	0,0084	0,0159	0,0227
15						0,0000	0,0076

Якщо n – парне, то $k = \frac{n}{2}$, а якщо n – непарне, то $k = \frac{n-1}{2}$ (в цьому випадку x_{k+1} не використовується для обчислень).

4) знаходять значення W-критерію за формулою:

$$W = \frac{b^2}{SS} \; ; \qquad \textbf{(29)}$$

5) при певному рівні статистичної значущості (зазвичай $p \leq 0,05$) перевіряють виконання умови:

$$W \geq W_{кр} \; , \qquad \textbf{(30)}$$

де $W_{кр}$ – критичне значення критерію, що взяте із табл. 6.

Таблиця 6. Критичні значення *W*-критерію

n	**Statistical significance level *p***	
	0,01	**0,05**
10	0,781	0,842
11	0,792	0,850
12	0,805	0,859
13	0,814	0,866
14	0,825	0,874
15	0,835	0,881
16	0,844	0,887
17	0,851	0,892
18	0,858	0,897
19	0,863	0,901
20	0,868	0,905
21	0,873	0,908
22	0,878	0,911
23	0,881	0,914
24	0,884	0,916
25	0,888	0,918
26	0,891	0,920
27	0,894	0,923
28	0,896	0,924
29	0,898	0,926
30	0,900	0,927

Якщо умова **(30)** виконується, то говорять про нормальний розподіл даних.

5.4. Коефіцієнт асиметрії та ексцесу

В математичній статистиці під *асиметрією (skewness)* розуміють показник, який характеризує ступінь несиметричності розподілу, а *ексцес (kurtosis)* – ступінь загостреності (згладженості) кривої розподілу ймовірностей випадкової величини, яку будують за результатами вимірювань (спостережень), у порівнянні з функцією нормального розподілу даних.

Перевірку гіпотези про те, що дані мають нормальний розподіл, використовуючи коефіцієнти асиметрії та ексцесу, здійснюють в такій послідовності:

1. Обчислюють коефіцієнт асиметрії за формулою:

$$As = \frac{\sum_{i=1}^{n}(x_i - \overline{x})^3}{ns^3}\,. \qquad \textbf{(31)}$$

Його величина може бути додатною (для правосторонньої асиметрії) і від'ємною (для лівосторонньої асиметрії).

2. Обчислюють показник ексцесу за формулою:

$$Ex = \frac{\sum_{i=1}^{n}(x_i - \overline{x})^4}{ns^4} - 3\,. \qquad \textbf{(32)}$$

Якщо показник ексцесу більший за нуль, то розподіл є гостровершинним із відхиленням від нормального розподілу, а якщо менший за нуль – то плосковершинним із відхиленням від нормального розподілу, наприклад розподіл Стьюдента (рис. 6).

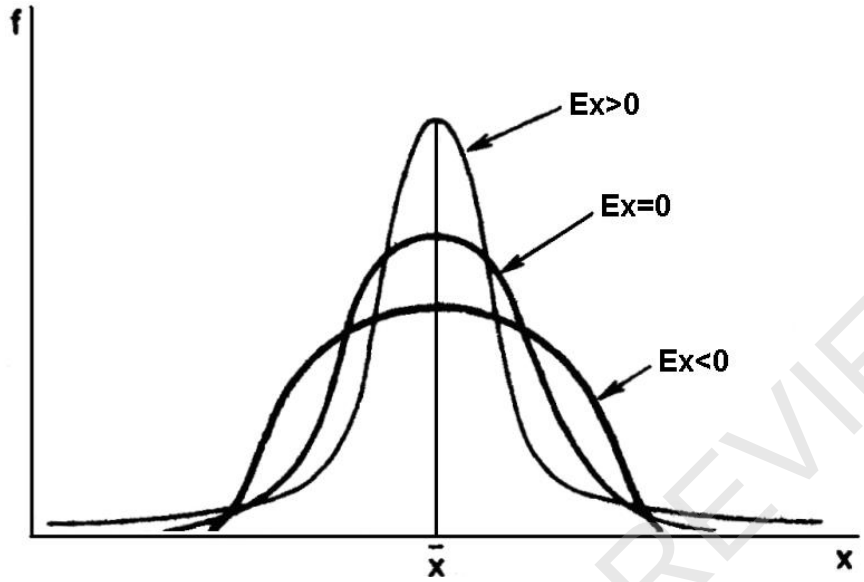


Рис 6. Ексцес розподілу даних

3. Обчислюють середні квадратичні відхилення коефіцієнту асиметрії та ексцесу:

$$s_{As} = \sqrt{\frac{6(n-1)}{(n+1)(n+3)}}; \quad \textbf{(33)}$$

$$s_{Ex} = \sqrt{\frac{24n(n-2)(n-3)(n-5)}{(n-1)^2(n+3)(n+5)}}. \quad \textbf{(34)}$$

4. Розраховують показники t_{As} і t_{Ex}:

$$t_{As} = \frac{|As|}{s_{As}}; \quad \textbf{(35)}$$

$$t_{Ex} = \frac{|Ex|}{s_{Ex}}. \quad \textbf{(36)}$$

Якщо показники t_{As} і t_{Ex} дорівнюють або більші за *3*, то говорять про статистично істотну відмінність емпіричного розподілу від нормального.

В табл. 7 представлені середні квадратичні відхилення коефіцієнту асиметрії та ексцесу для різних значень *n*, починаючи з 10.

Таблиця 7. Середні квадратичні відхилення коефіцієнту асиметрії та ексцесу

n	s_{As}	s_{Ex}
10	0,615	2,063
11	0,598	2,256
12	0,582	2,425
13	0,567	2,573
14	0,553	2,704
15	0,540	2,821
16	0,528	2,926
17	0,516	3,021
18	0,506	3,107
19	0,495	3,186
20	0,486	3,258
21	0,477	3,324
22	0,468	3,385
23	0,460	3,441
24	0,452	3,494
25	0,445	3,543
26	0,438	3,588
27	0,431	3,630
28	0,424	3,670
29	0,418	3,708
30	0,412	3,743

Проте слід врахувати і те, що коефіцієнти асиметрії та ексцесу слугують не стільки для перевірки нормальності, скільки для виявлення відхилень розподілу, який досліджується, від нормального.

Наведемо приклад перевірки даних вибірки на нормальний розподіл або перевіримо наскільки розподіл даних відрізняється від нормального.

Приклад 19. В результаті досліджень активності каталази в печінці карася сріблястого ми отримали наступні дані: 117, 115, 135, 121, 145, 123, 147, 127, 127, 144 Од/мг білка (наведені власні дані). Перевіримо дані за допомогою наведених вище критеріїв (*складовий*

критерій d, критерій Шапіро-Вілка, коефіцієнти ексцесу та асиметрії) на нормальність їх розподілу.

В підрозділі 5.1 згадувалося, що перед перевіркою даних вибірки на нормальний розподіл інколи перевіряється рівність між медіаною та середньою арифметичною величиною. Для цього слід розташувати всі дані в порядку їх зростання:

115, 117, 121, 123, 127, 127, 135, 144, 145, 147

Знаходимо медіану (підрозділ 2.1):

$$Me=127.$$

Обчислюємо середнє арифметичне значення $\bar{x}$ за формулою **(2)**:

$$\bar{x}=\frac{117+115+135+121+145+123+147+127+127+144}{10}=130 \text{ (Од/мг білка).}$$

Різниця між Me і $\bar{x}$ становить 2%, а, отже, ці дані можуть мати нормальний розподіл. Перевіряємо дані за вище вказаними критеріями на нормальний розподіл.

<u>За допомогою критерію Андерсона–Дарлінга</u>

Для перевірки цієї вибірки на нормальний розподіл за допомогою тесту Андерсона–Дарлінга виконаємо наступні кроки:

1. Сформулюємо нульову і альтернативну гіпотези:

H0: Вибірка походить з нормального розподілу.

H1: Вибірка не походить з нормального розподілу.

2. Обчислення середнього значення та стандартного відхилення:

$\bar{x}$ = (115 + 117 + 121 + 123 + 127 + 127 + 135 + 144 + 145 + 147) / 10 = 130

$s = \text{sqrt}((1 / (10 – 1)) \times ((115 – 130)^2 + (117 – 130)^2 + (121 – 130)^2 + (123 – 130)^2 + (127 – 130)^2 + (127 – 130)^2 + (135 – 130)^2 + (144 – 130)^2 + (145 – 130)^2 + (147 – 130)^2)) = 12,37$ (sqrt – корінь квадратний)

3. Сортування вибірки:

115, 117, 121, 123, 127, 127, 135, 144, 145, 147

4. Обчислення значень z:

$z_1 = (115 – 130) / 12,37 = -1,21$

$z_2 = (117 – 130) / 12,37 = -1,05$

$z_3 = (121 – 130) / 12,37 = -0,73$

$z_4 = (123 – 130) / 12,37 = -0,51$

$z_5 = (127 – 130) / 12,37 = -0,16$

$z_6 = (127 – 130) / 12,37 = -0,16$

$z_7 = (135 – 130) / 12,37 = 0,41$

$z_8 = (144 – 130) / 12,37 = 1,13$

$z_9 = (145 – 130) / 12,37 = 1,21$

$z_{10} = (147 – 130) / 12,37 = 1,51$

5. Обчислення значень функції розподілу стандартного нормального розподілу:

$F_1 = 0,1134$

$F_2 = 0,1401$

$F_3 = 0,2353$

$F_4 = 0,3039$

$F_5 = 0,4357$

$F_6 = 0,4357$

$F_7 = 0,6615$

$F_8 = 0,8704$

$F_9 = 0,8773$

$F_{10} = 0,9332$

6. Обчислення значень f_{compl}:

$f_{compl1} = 1 – F_1 = 0,8866$

$f_{compl2} = 1 – F_2 = 0,8599$

$f_{compl3} = 1 – F_3 = 0,7647$

$f_{compl4} = 1 – F_4 = 0,6961$

$f_{compl5} = 1 – F_5 = 0,5643$

$f_{compl6} = 1 – F_6 = 0,5643$

$f_{compl7} = 1 – F_7 = 0,3385$

$f_{compl8} = 1 – F_8 = 0,1296$

$f_{compl9} = 1 – F_9 = 0,1227$

$f_{compl10} = 1 – F_{10} = 0,0668$

7. Обчислення значень w:

$w_1 = (2 \times 1 – 1) / 10 \times (\ln(F_1) + \ln(f_{compl10})) = -0,488$

$w_2 = (2 \times 2 – 1) / 10 \times (\ln(F_2) + \ln(f_{compl9})) = -1,219$

$w_3 = (2 \times 3 – 1) / 10 \times (\ln(F_3) + \ln(f_{compl8})) = -1,745$

$w_4 = (2 \times 4 – 1) / 10 \times (\ln(F_4) + \ln(f_{compl7})) = -1,592$

$w_5 = (2 \times 5 – 1) / 10 \times (\ln(F_5) + \ln(f_{compl6})) = -1,263$

$w_6 = (2 \times 6 – 1) / 10 \times (\ln (F_6) + \ln (f_{compl5})) = -1,543$

$w_7 = (2 \times 7 – 1) / 10 \times (\ln (F_7) + \ln(f_{compl4})) = -1,001$

$w_8 = (2 \times 8 – 1) / 10 \times (\ln (F_8) + \ln(f_{compl3})) = -0,611$

$w_9 = (2 \times 9 – 1) / 10 \times (\ln (F_9) + \ln(f_{compl2})) = -0,479$

$w_{10} = (2 \times 10 – 1) / 10 \times (\ln(F_{10}) + \ln(f_{compl1})) = -0,360$

8. Обчислення суми всіх значень w:

$w = w_1 + w_2 + … + w_{10} = -10,301$

9. Обчислення значення A^2:

$A^2 = -10 – (-10{,}301) = 0{,}301$

10. Порівняємо отримане значення статистики тесту Андерсона–Дарлінга з критичним значенням. Оскільки 0,301 є меншим за критичне значення (0,752) за α=0,05, то можна стверджувати, що вибірка 117, 115, 135, 121, 145, 123, 147, 127, 127, 144 може бути з нормального розподілу на рівні значущості α = 0,05.

Код Python для перевірки вибірки на нормальний розподіл даних за критерієм Андерсона–Дарлінга:

```
import numpy as np
from scipy.stats import norm

# User data entry
data = input("Enter the data separated by a space: ").split(" ")
data = np.array([float(i.strip()) for i in data])

# Calculating the statistics of the Anderson-Darling test
def anderson_darling(data):
    n = len(data)
    mean = np.mean(data)
    std_dev = np.std(data, ddof=1)
    data_sorted = np.sort(data)
    z = (data_sorted - mean) / std_dev
    F = norm.cdf(z)
    f_compl = 1 - F
    w = (2 * np.arange(1, n + 1) - 1) / n * (np.log(F) + np.log(f_compl[::-1]))
    A_squared = -n - np.sum(w)
    return A_squared

# Calculating critical values of the Anderson-Darling test for significance levels 5%, 2.5, 1 i 0,5%
```

```
critical_values = {
    8: [0.752, 0.873, 1.029],
    15: [0.906, 1.065, 1.223],
    25: [1.038, 1.223, 1.472],
}
# Displaying test results
A_squared = anderson_darling(data)
A_squared_round = round(A_squared, 3)
n = len(data)
cv_idx = 0
if n >= 8 and n < 15:
    cv_idx = 0
elif n >= 15 and n < 25:
    cv_idx = 1
else:
    cv_idx = 2

if cv_idx == 0:
    crit_vals = critical_values[8]
elif cv_idx == 1:
    crit_vals = critical_values[15]
else:
    crit_vals = critical_values[25]
crit_val = np.interp(A_squared, np.array([8, 15, 25]),
crit_vals[:3])
p_value = np.exp(-1.2337141 - 0.03255817 * A_squared +
0.000975417 * A_squared ** 2)
# p-value approximation
p_value_round = round(p_value, 3)
print("Anderson-Darling test result:")
print("Test statistics =", A_squared_round)
print("Critical   for   5%,   2.5,   1%   levels   of
significance:", crit_vals)
print("Critical value for 5% significance level =",
crit_val)
if A_squared > crit_val:
    print("The  sample  is  not  drawn  from  a  normal
distribution ")
else:
    print("The   sample   is   drawn   from   a   normal
distribution ")
print("p =", p_value_round)
```

При виконанні цього коду спочатку відбувається введення даних користувачем у вигляді списку чисел через пробіл. Після цього виконується тест Андерсона–Дарлінга за допомогою функції **anderson_darling()**. Виведення результатів тесту здійснюється за допомогою використання атрибутів **A_squared_round**, **crit_vals, crit_val** та **p_value_round**.

Нарешті, умовний оператор перевіряє, чи перевищує статистика тесту критичне значення, і виводить відповідну інформацію про те, чи є вибірка з нормального розподілу.

Результат виконання цього коду:

```
Введіть дані через пробіл: 115 117 121 123 127 127
135 144 145 147
Результат тесту Андерсона-Дарлінга:
Статистика тесту = 0.406
Критичне значення для 5%, 2.5, 1% рівнів значущості:
[0.752, 0.873, 1.029]
Критичне значення для 5% рівня значущості = 0.752
Вибірка походить з нормального розподілу
p-значення = 0.287
```

Статистика тесту (0,406) є числом, яке представляє відстань між емпіричною функцією розподілу вибірки та теоретичною функцією розподілу нормального розподілу. Чим менше це число, тим більше вибірка відповідає нормальному розподілу. Критичні значення (0,752, 0,873, 1,029) представляють межі між рівнями значущості 5%, 2,5% та 1%. Якщо статистика тесту менша критичного значення, то на рівні значущості, що відповідає цьому критичному значенню, можна відкинути гіпотезу про те, що вибірка не походить з нормального розподілу. Наприклад, якщо статистика тесту менша критичного значення на рівні значущості 5%, то ми можемо

відкинути гіпотезу про те, що вибірка не походить з нормального розподілу на рівні значущості 5%. У цьому конкретному випадку статистика тесту досить низька, а отримані критичні значення та p-значення вказують на те, що вибірка може бути з нормального розподілу.

Код R для перевірки вибірки на нормальний розподіл даних за критерієм Андерсона–Дарлінга:

```
# Створення вектора з вибірковими даними
x <- c(115, 117, 121, 123, 127, 127, 135, 144, 145, 147)

# Встановлення пакету для критерію Андерсона-Дарлінга
install.packages("nortest")
library(nortest)

# Перевірка вибірки за критерієм Андерсона-Дарлінга
ad.test(x)
```

Результатом виконання коду буде виведення статистики критерію Андерсона–Дарлінга, його критичного значення та p-значення, що дає змогу зробити висновок про те, чи є вибірка нормально розподіленою:

```
Anderson-Darling normality test
data:  x
A = 0.40623, p-value = 0.2824
```

<u>За допомогою критерію Шапіро-Вілка:</u>

1. Розміщуємо дані у порядку зростання:

115, 117, 121, 123, 127, 127, 135, 144, 145, 147.

2. Обчислюємо значення величини *SS* за формулою (**27):**

$SS=115^2+117^2+121^2+123^2+127^2+127^2+135^2+144^2+145^2+147^2 -$

$(115+117+121+123+127+127+135+144+145+147)^2/10 = 1277.$

3. Обчислюємо значення величини *b* за формулою (**28**):

$$b = a_{10}(y_{10}-y_1)+ a_9(y_9-y_2)+ a_8(y_8-y_3)+ a_7(y_7-y_4)+ a_6(y_6-y_5).$$

Коефіцієнти a_i беремо із табл. 7:

a_{10} =0,5769; a_9 =0,3291; a_8 =0,2141; a_7 = 0,1224; a_6=0,0399.

B = 0,5769(147-115)+ 0,3291(145-117) + 0,2141(144-121) + 0,1224(135-123) + 0,0399(127-127) = 34,1.

4. За формулою **(29)** обчислюємо значення *W*-критерію:

$$W=1163/1277=0,911.$$

5. Знаходимо критичне значення $W_{кр}$ при n=10 та рівні статистичної значущості p=0,05 (табл. 8):

$$W_{кр}=0,842.$$

В даному випадку виконується нерівність **(30)**, бо 0,911>0,842. Тому можна говорити про те, що отримані нами дані підпорядковуються *нормальному розподілу.*

Код Python для перевірки вибірки на нормальний розподіл даних за критерієм Шапіро-Вілка:

```
from scipy.stats import shapiro

# Data
data = input("Enter the data separated by space: ")
data = [float(x) for x in data.split()]

# criterion calculation W
```

```
stat, p = shapiro(data)

# output of results
print('stat=%.3f, p=%.3f' % (stat, p))
if p > 0.05:
    print('The data is normally distributed')
else:
    print('Data is not normally distributed')
```

У цьому коді ми використовуємо масив **data**, а функція **shapiro()** повертає дві величини: **stat** – значення статистики критерію **W** та **p** – рівень значущості тесту. Рівень значущості може бути використаний для визначення того, чи мають дані нормальний розподіл. У прикладі, якщо p > 0,05, ми вважаємо, що дані мають нормальний розподіл. Інакше, якщо p <= 0,05, ми вважаємо, що дані не мають нормального розподілу.

Результатом виконання цього коду за вказаним прикладом є:

```
Введіть дані через кому: >? 117, 115, 135, 121, 145, 123, 147, 127, 127, 144
stat=0.904, p=0.242
Дані мають нормальний розподіл
```

Для перевірки даних на нормальний розподіл за критерієм Шапіро-Вілка в R можна скористатись наступним алгоритмом:

```
# вхідні дані

data <- c(117, 115, 135, 121, 145, 123, 147, 127, 127, 144)

# обчислення критерію W

result <- shapiro.test(data)

# вивід результатів

print(result)
```

У цьому прикладі ми використовуємо масив **data**, а функція **shapiro.test()** повертає результат тесту на нормальність, який містить значення статистики критерію W та рівень значущості тесту. Результат може бути використаний для визначення того, чи мають дані нормальний розподіл. У прикладі, якщо **p-value > 0.05**, ми вважаємо, що дані мають нормальний розподіл. Інакше, якщо **p-value <= 0.05**, ми вважаємо, що дані не мають нормального розподілу.

Результатом виконання цього коду за вказаним прикладом є:

```
Shapiro-Wilk normality test
data:  data
W = 0.90391, p-value = 0.2417
```

Цей результат вказує на те, що дані у вибірці мають нормальний розподіл.

<u>За допомогою коефіцієнту асиметрії та ексцесу</u>

1. Первинні дані та допоміжні величини оформлюємо у вигляді таблиці:

	x_i	$(x_i-\bar{x})$	$(x_i-\bar{x})^2$	$(x_i-\bar{x})^3$	$(x_i-\bar{x})^4$
	115	-15,1	228,01	-3442,95	51988,6
	117	-13,1	171,61	-2248,09	29450,0
	121	-9,10	82,81	-753,571	6857,50
	123	-7,10	50,41	-357,911	2541,17
	127	-3,10	9,61	-29,791	92,3521
	127	-3,10	9,61	-29,791	92,3521
	135	4,90	24,01	117,649	576,480
	144	13,9	193,21	2685,62	37330,1
	145	14,9	222,01	3307,95	49288,4
	147	16,9	285,61	4826,81	81573,1
Σ	**1301**		**1276,9**	**4075,92**	**259790**
$\bar{x}$	**130,1**				

2. За допомогою формули **(8)** обчислюємо середнє квадратичне відхилення:

$$s = \sqrt{\frac{1276,9}{10-1}} = 11,91.$$

3. Обчислюємо коефіцієнти асиметрії та ексцесу за формулами **(31)** і **(32)**, відповідно:

$$As = \frac{4075,92}{20 \cdot 11,91^3} = 0,1206;$$

$$Ex = \frac{259790}{20 \cdot 11,91^4} - 3 = -2,3547.$$

4. Розраховуємо показники t_{As} і t_{Ex} за формулами **(35)** і **(36)** та даними з табл. 9, відповідно:

$$t_{As} = \frac{|0,1206|}{0,615} = 0,196;$$

$$t_{Ex} = \frac{|-2,3547|}{2,063} = 1,141.$$

Отже, показники t_{As} і t_{Ex} менші за *3*, що говорить про статистично незначущу відмінність емпіричного розподілу від нормального.

Зауваження!!! Проте багато біологічних показників не мають нормального розподілу і тому коректніше стверджувати, що розподіли мало відрізняються від нормального. Що ж робити в такому випадку, коли масив даних не має нормального розподілу? Тоді дані, які ми отримали, можна трансформувати, скориставшись, наприклад, перетвореннями:

- логарифмування ($z = log_{10}y$);
- отримання з даних кореня квадратного ($z = \sqrt{y}$);
- зворотне перетворення ($z = 1/y$);
- піднесення до квадрату ($z = y^2$);
- логіт-перетворення ($z = \ln \frac{p}{1-p}$).

В результаті таких перетворень розподіл вихідних даних може стати нормальним, що буде підґрунтям застосування параметричних методів обробки даних. Але в цих методах перетворень є і свої недоліки. Найперше те, що при перетворенні даних перетворюються і одиниці їх вимірювань, гублячи свій фізичний зміст. По-друге, результати аналізу перетворених даних важко інтерпретувати.

РОЗДІЛ 6. ПОРІВНЯННЯ ДВОХ І БІЛЬШЕ ГРУП МІЖ СОБОЮ

6.1. Вибір статистичного критерію

На початку обробки результатів виникає питання: який саме статистичний критерій вибрати для порівняння генеральних сукупностей на основі вибірок? Ключова проблема – вибрати параметричний чи непараметричний критерій? У вітчизняних підручниках зі статистики рідко наводяться чіткі умови такого вибору.

Перечислимо умови вибору критеріїв.

Умови вибору параметричних критеріїв:

1. Дані представлені рядом із неперервних величин (використовується неперервна числова шкала вимірювань).

2. Сукупності даних мають нормальний розподіл. Наприклад, дисперсійний аналіз і його частковий випадок, критерій Стьюдента, що базуються на порівнянні середніх арифметичних величин і дисперсій.

Умови вибору непараметричних критеріїв:

1. Дані представлені рядом із дискретних величин (використовується шкала на основі категорій, порядкових чисел рангів, імен, введених дослідником).

2. Обмеження параметричних критеріїв не виконуються або не можуть бути перевірені на даній вибірці. Наприклад, дані мають розподіл, відмінний від нормального.

Слід зауважити і те, що при виборі критерію потрібно враховувати їхні обмеження. Так, наведемо деякі з цих обмежень:

1. Рівна або нерівна кількість спостережень у вибірках допускається для даного критерію? (Наприклад, критерій Манна–Вітні допускає нерівну кількість спостережень, критерій Стьюдента для двох незалежних вибірок допускає нерівну кількість спостережень, а критерій Стьюдента для двох залежних вибірок (парний тест) – ні).

2. Яка мінімальна кількість спостережень необхідна для розрахунку даного критерію? Для багатьох непараметричних критеріїв мінімальна кількість спостережень складає 5-6. Також слід мати на увазі і те, що при малій кількості вибірки у ряді випадків в стандартний критерій вводять поправки.

Найбільш характерними помилками є:

- використання параметричних методів для аналізу даних, що не мають нормального розподілу;
- використання методів, що мають застосовуватись для незалежних вибірок, при аналізі парних даних.

Наведемо підсумкову таблицю з використання того чи іншого критерію (табл. 8).

В цьому посібнику ми зупинимось детальніше на параметричних критеріях, які найчастіше використовуються дослідниками при порівнянні двох і більше дослідних груп, та на їхніх непараметричних аналогах.

Таблиця 8. Вибір статистичного критерію для обчислення

Розподіл величин	Досліджуються		
	дві групи	більше двох груп	взаєзв'язок за двома ознаками
Нормальний (застосовувати параметричні критерії)	Критерій Стьюдента, критерій Велча, критерій Тюкі, критерій Шеффе	Критерій Стьюдента з поправкою Бонферроні, Дисперсійний аналіз (ANOVA), критерій Ньюмена-Коулса, критерій Даннета, критерій Шеффе, критерій Тюкі, тест F Снедекора	Лінійна регресія, кореляційний аналіз за Пірсоном або метод Бленда-Алтмана
Відмінний від нормального або невідомий (застосовувати непараметричні критерії)	χ^2-критерій, критерії Манна–Вітні, Вілкоксона, Колмогорова – Смірнова, Вальда-Вольфовіца, точний критерій Фішера	Критерій Краскла-Уоліса, медіанний критерій	Коефіцієнти рангової кореляції Спірмена або Кендала

В табл. 9 наведені параметричні критерії і їхні непараметричні критерії-аналоги, на яких ми зупинимось детальніше в наступних підрозділах.

Таблиця 9. Параметричні критерії та їхні непараметричні критерії-аналоги

Кількість груп для порівняння	Критерії	
	Параметричні	Непараметричні
Дві незалежні групи	t-критерій Стьюдента для незалежних вибірок Критерій Велча	*U*-критерій Манна–Вітні
Дві залежні групи	Парний критерій Стьюдента	*W*-критерій Вілкоксона
Три і більше груп	критерій Ньюмена-Коулса, критерій Тюкі, критерій Дункана, критерій Даннета	Критерій Данна

6.2. Порівняння двох груп між собою

6.2.1. Непарний та парний критерії Стьюдента

Критерій Стьюдента надзвичайно популярний у біологічних дослідженнях. Він використовується у більш, ніж в половині біологічних та медичних досліджень. Проте часто дослідники забувають те, що цей критерій має певні обмеження. Так, його потрібно використовувати тільки:

- для порівняння двох груп, а не декілька груп попарно;
- групи, що відібрані для порівняння мають мати нормальний розподіл даних.

Перед тим, як перейти до опису даного критерію, потрібно розглянути питання однорідності дисперсій.

Для перевірки гіпотези однорідності дисперсій є ряд як параметричних, так і непараметричних критеріїв. Серед параметричних слід виділити наступні: критерій Бартлетта (*Bartlett's test*), критерій Кокрена (*Cochran's test*), критерій Хартлі (*Hartley's test*), критерій Левене (*Levene's test*), критерій Фішера (*Fisher's test*). Серед непараметричних виділяють: критерій Ансарі-Бредлі (*Ansary-Bradley's test*), критерій Муда (*Mood's test*), критерій Зігеля-Тюкі (*Siegel-Tukey test*), критерій Кейпена (*Capon's test*), критерій Клотца (*Klotz's test*). Ми розглянемо тільки два параметричних критерії: *Кокрена* (для вибірок, що містять одинакову кількість даних) та *Фішера* (для вибірок, що містять різну кількість даних), а також один непараметричний критерій *Зігеля-Тюкі*.

Якщо вибірки мають нормальний розподіл даних і містять однакову кількість даних ($n_1=n_2$), то рівність дисперсій сукупностей

можна проаналізувати за допомогою ***критерію Кокрена G*** (в літературі можна зустріти Кочрена, Кохрена). Ми рекомендуємо використовувати саме цей критерій, який, по-перше, є досить простим у застосуванні, по-друге, є чутливішим за інші критерії, по-третє, саме його рекомендують використовувати в тих випадках, коли одна із вибіркових дисперсій є значно більшою за інші.

Спочатку обчислюють відношення максимальної оцінки дисперсії до суми оцінок всіх дисперсій:

$$G = \frac{s^2_{max}}{\sum_{i=1}^{n} s_i^2}. \qquad \textbf{(37)}$$

Після цього значення G порівнюють із критичним значенням критерію Кокрена $G_{кр}$ (табл. 10). Розподіл величини G залежить від числа ступенів свободи df $(df=n–1)$, кількості членів у кожній з вибірок n і рівня статистичної значущості p.

Якщо виконується нерівність $G < G_{кр}$, то дисперсії вибірок можна вважати рівними.

Таблиця 10. Критичні точки розподілу критерію Кокрена (df – число ступенів свободи, L – кількість груп для порівняння) при статистичній значущості p

L	**Рівень статистичної значущості $p<0,05$**						
	df *(n-1)*						
	3	**4**	**5**	**6**	**7**	**8**	**9**
2	0,9392	0,9057	0,8772	0,8534	0,8332	0,8159	0,8010
3	0,7977	0,7457	0,7070	0,6770	0,6531	0,6333	0,6167
4	0,6839	0,6287	0,5894	0,5598	0,5365	0,5175	0,5018
5	0,5981	0,5440	0,5063	0,4783	0,4564	0,4387	0,4241
6	0,5321	0,4803	0,4447	0,4184	0,3980	0,3817	0,3682
7	0,4800	0,4307	0,3972	0,3726	0,3536	0,3384	0,3259
8	0,4377	0,3910	0,3594	0,3362	0,3185	0,3043	0,2927
9	0,4027	0,3584	0,3285	0,3067	0,2901	0,2768	0,2659

10	0,3733	0,3311	0,3028	0,2823	0,2666	0,2541	0,2439
11	0,3482	0,3080	0,2811	0,2616	0,2468	0,2350	0,2254
12	0,3264	0,2880	0,2624	0,2440	0,2299	0,2187	0,2096
13	0,3074	0,2707	0,2463	0,2286	0,2152	0,2046	0,1960
14	0,2907	0,2554	0,2321	0,2153	0,2025	0,1924	0,1841
15	0,2758	0,2419	0,2195	0,2034	0,1912	0,1815	0,1737
16	0,2624	0,2298	0,2083	0,1929	0,1812	0,1719	0,1644
17	0,2504	0,2190	0,1983	0,1835	0,1722	0,1633	0,1561
18	0,2395	0,2092	0,1892	0,1750	0,1641	0,1556	0,1487
19	0,2296	0,2003	0,1810	0,1672	0,1568	0,1486	0,1419
20	0,2205	0,1921	0,1735	0,1602	0,1502	0,1422	0,1358
21	0,2122	0,1847	0,1667	0,1538	0,1441	0,1364	0,1302
22	0,2045	0,1778	0,1604	0,1479	0,1385	0,1311	0,1251
23	0,1974	0,1715	0,1545	0,1425	0,1333	0,1262	0,1204
24	0,1908	0,1656	0,1491	0,1374	0,1286	0,1216	0,1160
25	0,1846	0,1601	0,1441	0,1328	0,1242	0,1174	0,1120
26	0,1789	0,1550	0,1395	0,1284	0,1201	0,1135	0,1082
27	0,1735	0,1503	0,1351	0,1244	0,1162	0,1099	0,1047
28	0,1685	0,1458	0,1311	0,1206	0,1127	0,1064	0,1014
29	0,1638	0,1416	0,1272	0,1170	0,1093	0,1033	0,0984
30	0,1593	0,1377	0,1236	0,1137	0,1061	0,1003	0,0955

Якщо вибірки містять різну кількість даних ($n_1 \neq n_2$), то рівність дисперсій сукупностей можна проаналізувати за допомогою ***критерію Фішера F*** (в літературі можна зустріти Фішера-Снедекора), який визначають за такою формулою:

$$F = \frac{s_1^{\,2}}{s_2^{\,2}}\ , \qquad \textbf{(38)}$$

де $s_1 > s_2$, F – значення критерію Фішера, яке порівнюють із його критичним значенням ($F_{кр}$) при рівні статистичної значущості p і числі ступенів свободи: $df_1 = n_1 - 1$, $df_2 = n_2 - 1$ (n_1, n_2 – об'єми вибірок), причому df_1 – число ступенів свободи більшої дисперсії, а df_2 – число ступенів свободи меншої дисперсії (табл. 11).

Таблиця 11. Критичні значення $F_{кр}$ при рівні статистичної значущості $p<0,05$

df_2 ($df_м$)	df_1 ($df_в$)									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
3	10,1	9,55	9,28	9,12	9,01	8,94	8,89	8,85	8,81	8,79
4	7,71	6,94	6,59	6,39	6,26	6,16	6,09	6,04	6,00	5,96
5	6,61	5,79	5,41	5,19	5,05	4,95	4,88	4,82	4,77	4,74
6	5,99	5,14	4,76	4,53	4,39	4,28	4,21	4,15	4,10	4,06
7	5,59	4,74	4,35	4,12	3,97	3,87	3,79	3,73	3,68	3,64
8	5,32	4,46	4,07	3,84	3,69	3,58	3,50	3,44	3,39	3,35
9	5,12	4,26	3,86	3,63	3,48	3,37	3,29	3,23	3,18	3,14
10	4,96	4,10	3,71	3,48	3,33	3,22	3,14	3,07	3,02	2,98
11	4,84	3,98	3,59	3,36	3,20	3,09	3,01	2,95	2,90	2,85
12	4,75	3,89	3,49	3,26	3,11	3,00	2,91	2,85	2,80	2,75
13	4,67	3,81	3,41	3,18	3,03	2,92	2,83	2,77	2,71	2,67
14	4,60	3,74	3,34	3,11	2,96	2,85	2,76	2,70	2,65	2,60
15	4,54	3,68	3,29	3,06	2,90	2,79	2,71	2,64	2,59	2,54
16	4,49	3,63	3,24	3,01	2,85	2,74	2,66	2,59	2,54	2,49
17	4,45	3,59	3,20	2,96	2,81	2,70	2,61	2,55	2,49	2,45
18	4,41	3,55	3,16	2,93	2,77	2,66	2,58	2,51	2,46	2,41
19	4,38	3,52	3,13	2,90	2,74	2,63	2,54	2,48	2,42	2,38
20	4,35	3,49	3,10	2,87	2,71	2,60	2,51	2,45	2,39	2,35
21	4,32	3,47	3,07	2,84	2,68	2,57	2,49	2,42	2,37	2,32
22	4,30	3,44	3,05	2,82	2,66	2,55	2,46	2,40	2,34	2,30
23	4,28	3,42	3,03	2,80	2,64	2,53	2,44	2,37	2,32	2,27
24	4,26	3,40	3,01	2,78	2,62	2,51	2,42	2,36	2,30	2,25
25	4,24	3,39	2,99	2,76	2,60	2,49	2,40	2,34	2,28	2,24
26	4,23	3,37	2,98	2,74	2,59	2,47	2,39	2,32	2,27	2,22
27	4,21	3,35	2,96	2,73	2,57	2,46	2,37	2,31	2,25	2,20
28	4,20	3,34	2,95	2,71	2,56	2,45	2,36	2,29	2,24	2,19
29	4,18	3,33	2,93	2,70	2,55	2,43	2,35	2,28	2,22	2,18
30	4,17	3,32	2,92	2,69	2,53	2,42	2,33	2,27	2,21	2,16

Якщо статистичні дані не розподілені за нормальним законом, то перевірка гіпотези про рівність генеральних дисперсій здійснюється за *критерієм Зігеля-Тюкі*. Для цього спочатку формулюються гіпотези:

H_0 – дисперсії двох генеральних сукупностей рівні, тобто $s^2_1=s^2_2$;

H_1 – дисперсії двох генеральних сукупностей не рівні, тобто $s^2_1 \neq s^2_2$.

Перевірка дисперсій виконується за даними двох вибірок у такій послідовності:

1) формується об'єднана вибірка;

2) даним об'єднаної вибірки присвоюються ранги (порядкові номери) за правилом: найменшому значенню присвоюється ранг 1, двом найбільшим – ранги 2 і 3; наступним двом найменшим – ранги 4 і 5; наступним найбільшим – ранги 6 і 7 і т. д. Якщо кількість елементів вибірки непарна, то її центральний елемент (тобто медіана) не отримує ніякого рангу;

3) розраховуються суми рангів елементів вихідних вибірок R_1 і R_2;

4) розраховується величина Z, що має приблизно нормальний розподіл:

$$Z = \frac{2R_1 - n_1(n_1+n_2+1)+1}{\frac{n_2}{3}\times\sqrt{n_1(n_1+n_2+1)}}, \quad \textbf{(39)}$$

де n_1, n_2 – об'єми вибірок. При цьому R_1 – сума рангів меншої за об'ємом вибірки. Якщо $2R_1 > n_1(n_1+n_2+1)+1$, Z обчислюється за формулою:

$$Z = \frac{2R_1 - n_1(n_1+n_2+1)-1}{\frac{n_2}{3}\times\sqrt{n_1(n_1+n_2+1)}} \quad \textbf{(40)}$$

5) у випадку, коли перевіряються вибірки різних об'ємів, обчислюється скорегована нормальна випадкова величина Z' за формулою:

$$Z' = Z + \left(\frac{1}{10n_1} - \frac{1}{10n_2}\right)\left(Z^3 - 3Z\right) \quad \textbf{(41)}$$

6) обирається рівень значущості α;

7) За допомогою таблиці значень функції нормального розподілу або вбудованої функції Excel NORMSDIST знаходиться ймовірність *P(Z)* або P(Z′). Якщо 2P(Z)>α (або 2P(Z′)>α), то гіпотеза H_0 про рівність дисперсій приймається.

Слід зауважити також, що для перевірки правильності присвоєння рангів можна скористатися формулами:

$$R_1 + R_2 = \frac{(n_1+n_2)(n_1+n_2+1)}{2} \quad \textbf{(42)}$$

у випадку парної кількості елементів об'єднаної вибірки;

а у випадку непарної кількості елементів об'єднаної вибірки формула наступна:

$$R_1 + R_2 = (n_1 + n_2)\left(\frac{n_1+n_2+1}{2} - 1\right) \quad \textbf{(43)}$$

Приклад 20. В результаті експерименту отримали дані, які не розподілені за нормальним законом:

Група 1: 182 132 94 167 91 126 118 112 201 116

Група 2: 101 97 126 111 161 95 127 141 121 134

Перевіряємо гіпотезу про рівність дисперсій за *критерієм Зігеля-Тюкі*. Сформулюємо гіпотези: H_0 – дисперсії вибірок рівні; H_1 – дисперсії не рівні. Перевіримо справедливість гіпотези H_0 за критерієм Зігеля-Тюкі. Спочатку отримуємо об'єднану вибірку, присвоїмо її елементам ранги і знайдемо їх суму. Результати розрахунків оформимо у вигляді таблиці. Для зручності підкреслимо елементи першої вибірки:

Елементи об’єднаної вибірки	Сортована об’єднана вибірка	Ранги елементів об’єднаної вибірки	Ранги елементів першої вибірки	Ранги елементів другої вибірки
182	91	1	1	
132	94	4	4	
94	95	5		5
167	97	8		8
91	101	9		9
126	111	12		12
118	112	13	13	
112	116	16	16	
201	118	17	17	
116	121	20		20
101	126	19	19	
97	126	18		18
126	127	15		15
111	132	14	14	
161	134	11		11
95	141	10		10
127	161	7		7
141	167	6	6	
121	182	3	3	
134	201	2	2	
		Суми	**95**	**115**

Розрахуємо за формулою **(39)** значення Z, враховуючи, що $n_1=n_2=10$:

$$Z = \frac{2 \times 95 - 10(10 + 10 + 1) + 1}{\frac{10}{3} \times \sqrt{10(10 + 10 + 1)}} \approx -0{,}393$$

Оберемо рівень значущості $\alpha=0{,}05$. За допомогою вбудованої функції Excel NORMSDIST знаходимо ймовірність P(Z):

P(Z) = NORMSDIST (−0,393; 0; 1; TRUE) = 0,3469.

Оскільки $2P(Z)=2\times0{,}3469=0{,}6938>\alpha=0{,}05$, то гіпотеза H_0 про рівність генеральних дисперсій приймається. **Отже, дисперсії вибірок однакові.**

Код Python для виконання тесту Зігеля-Тюкі:

```
import numpy as np
from scipy.stats import rankdata, ranksums

group1 = [182, 132, 94, 167, 91, 126, 118, 112, 201,
116]
group2 = [101, 97, 126, 111, 161, 95, 127, 141, 121,
134]

# Create a dictionary where the keys are the elements
and the values are the corresponding ranks
ranks_dict1 = {
    182: 3, 132: 14, 94: 4, 167: 6, 91: 1, 126: 19, 118:
17, 112: 13, 201: 2, 116: 16
}

ranks_dict2 = {
    95: 5, 97: 8, 101: 9, 111: 12, 121: 20, 126: 18,
127: 15, 134: 11, 141: 10, 161: 7
}

# Find the ranks of the elements of the first and second
groups using the dictionary
ranks_group1 = [ranks_dict1[value] for value in group1]
ranks_group2 = [ranks_dict2[value] for value in group2]

# Calculating the sum of group ranks
sum_ranks_group1 = np.sum(ranks_group1)
sum_ranks_group2 = np.sum(ranks_group2)

# Calculating the value Z
n1 = len(group1)
n2 = len(group2)
Z = ((2 * sum_ranks_group1 - n1 * (n1 + n2 + 1) + 1) /
((n2 / 3) * np.sqrt(n1 * (n1 + n2 + 1))))

# Level of significance
alpha = 0.05

# Testing the hypothesis
```

```
p_value = ranksums(group1, group2).pvalue
is_rejected = p_value < alpha

# Display of results
print("Rank groups 1:", ranks_group1)
print("Rank groups 2:", ranks_group2)
print("Sum of group ranks 1:", sum_ranks_group1)
print("Sum of group ranks 2:", sum_ranks_group2)
print("Significance Z:", Z)
print("p-value:", p_value)
print("The hypothesis of equality of variances is
accepted:", not is_rejected)
```

Результат виконання цього коду:

```
Ранги групи 1: [3, 14, 4, 6, 1, 19, 17, 13, 2, 16]
Ранги групи 2: [9, 8, 18, 12, 7, 5, 15, 10, 20, 11]
Сума рангів групи 1: 95
Сума рангів групи 2: 115
Значення Z: -0.39333736882514186
p-значення: 0.7337299956962472
Гіпотеза про рівність дисперсій приймається: True
```

Код R для виконання тесту Зігеля-Тюкі:

```
group1 <- c(182, 132, 94, 167, 91, 126, 118, 112, 201,
116)
group2 <- c(101, 97, 126, 111, 161, 95, 127, 141, 121,
134)

# Create a dictionary where the keys are the elements
and the values are the corresponding ranks
ranks_dict1 <- list(`182` = 3, `132` = 14, `94` = 4,
`167` = 6, `91` = 1, `126` = 19, `118` = 17, `112` =
13, `201` = 2, `116` = 16)
ranks_dict2 <- list(`95` = 5, `97` = 8, `101` = 9, `111`
= 12, `121` = 20, `126` = 18, `127` = 15, `134` = 11,
`141` = 10, `161` = 7)

# Find the ranks of the elements of the first and second
groups using the dictionary
```

```
ranks_group1 <- sapply(group1, function(x)
ranks_dict1[[as.character(x)]])
ranks_group2 <- sapply(group2, function(x)
ranks_dict2[[as.character(x)]])

# Calculating the sum of group ranks
sum_ranks_group1 <- sum(ranks_group1)
sum_ranks_group2 <- sum(ranks_group2)

# Calculating the value Z
n1 <- length(group1)
n2 <- length(group2)
Z <- ((2 * sum_ranks_group1 - n1 * (n1 + n2 + 1) + 1) /
((n2 / 3) * sqrt(n1 * (n1 + n2 + 1))))

# Level of significance
alpha <- 0.05

# Hypothesis testing with an approximate p-value
p_value <- wilcox.test(group1, group2, exact =
FALSE)$p.value
is_rejected <- p_value < alpha

# Display of results
cat("Rank groups 1:", ranks_group1, "\n")
cat("Rank groups 2:", ranks_group2, "\n")
cat("Sum of group ranks 1:", sum_ranks_group1, "\n")
cat("Sum of group ranks 2:", sum_ranks_group2, "\n")
cat("Value Z:", Z, "\n")
cat("p-value:", p_value, "\n")
cat("The hypothesis of equality of variances is
accepted:", !is_rejected, "\n")
```

Результат виконання цього коду:

```
Ранги групи 1: 3 14 4 6 1 19 17 13 2 16
Ранги групи 2: 9 8 18 12 7 5 15 10 20 11
Сума рангів групи 1: 95
Сума рангів групи 2: 115
Значення Z: -0.3933374
p-значення: 0.7622821
Гіпотеза про рівність дисперсій приймається: TRUE
```

Тепер опишемо детальніше тести Стьюдента для незалежних та залежних даних. Чим вони відрізняються?

Якщо дані, які ми хочемо порівняти, були отримані на різних об'єктах (наприклад, визначаємо активність одного і того ж ферменту на карасях в контрольній групі, і групі, що піддавалась впливу йонів кобальту. Для цього дослідження потрібно як мінімум два карасі). Тут ми використаємо непарний тест Стьюдента для незалежних даних (*Normal Student's t-test*).

Якщо дані були отримані на одному об'єкті (наприклад, кількість еритроцитів до і після впливу йонів кобальту на організм карася сріблястого), то ми будемо використовувати парний тест Стьюдента для залежних даних (*Paired Student's t-test*). Тест для залежних даних потужніший, бо використовуються величини, що належать одному і тому ж об'єкту.

Для обчислення значення статистики критерію Стьюдента використовують наступну формулу:

$$t = \frac{\left|\bar{x}_1 - \bar{x}_2\right|}{\sqrt{\left[\frac{\sum (x_i - \bar{x}_1)^2 + \sum (x_i - \bar{x}_2)^2}{n_1 + n_2 - 2}\right] \times \left(\frac{n_1 + n_2}{n_1 \times n_2}\right)}} \qquad \textbf{(44)}$$

Оскільки в дослідженнях часто використовують шість-вісім повторів (n=6-8), то саме цю формулу слід використовувати для обчислення незалежних даних.

Число ступенів свободи *df* визначають при цьому за наступними формулами:

1)при $n_1 = n_2$:

$$df = n - 1 + \frac{2n-2}{\frac{s_1^{\,2}}{s_2^{\,2}} + \frac{s_2^{\,2}}{s_1^{\,2}}} \quad ; \qquad \textbf{(45)}$$

2) при $n_1 \neq n_2$:

$$df = \left(\frac{s_1^{\,2}}{n_1} - \frac{s_2^{\,2}}{n_2} \right)^2 \Bigg/ \left[\frac{\left(s_1^{\,2}/n_1\right)^2}{n_1+1} + \frac{\left(s_2^{\,2}/n_2\right)^2}{n_2+1} \right] - 2 \ . \qquad \textbf{(46)}$$

Вищевказану формулу (49) для обчислення статистики критерію Стьюдента доцільно використовувати при непарному тесті Стьюдента для незалежних даних. Якщо ми маємо справу із залежними даними, то в такому випадку при парному тесті Стьюдента слід використовувати при числі ступенів свободи *df = n-1* наступну формулу:

$$t = \frac{|\overline{d}|}{s_{\overline{d}}} \ , \qquad \textbf{(47)}$$

де $\overline{d} = \frac{\sum d}{n}$ – середнє значення змін між даними до і після експерименту;

$s_{\overline{d}} = \frac{s_d}{\sqrt{n}}$ – стандартна помилка змін між даними до і після експерименту;

$s_d = \sqrt{\frac{\sum (d - \overline{d})^2}{n-1}}$ - вибіркове стандартне відхилення змін *d*.

Наведемо приклади, де можна застосувати формулу **(44)** для незалежних даних і формулу **(47)** для залежних даних та поетапне обчислення в цих випадках *t-критерію*.

Приклад 20 (*Normal Student's t-test*). В печінці карася сріблястого визначали активність лактатдегідрогенази, причому одна із дослідних груп тварин піддавалась дії йонів кобальту в

концентрації 100 мг Co^{2+}/л води. Отримали наступні результати (використано власні дані):

Контрольна група		Група риб, що піддавалась дії йонів кобальту в концентрації 100 мг Co^{2+}/л води	
№ риби	**Активність ЛДГ, Од./мг білка**	**№ риби**	**Активність ЛДГ, Од./мг білка**
1	1,88	6	2,29
2	2,26	7	2,46
3	2,09	8	2,51
4	2,08	9	2,27
5	2,07		

Відомо, що показники активності лактатдегідрогенази в печінці карася сріблястого розподіляються нормально.

Найкраще скористатись оформленням обчислень у вигляді таблиці:

Показник, що обраховується	**Контрольна група**	**Група риб, що піддавалась дії йонів кобальту в концентрації 100 мг Co^{2+}/л води**
***n* (в даному випадку, дорівнює кількості риб)**	$n_1 = 5$	$n_2 = 4$
Число ступенів свободи *df=n-1*	$df_1 = 4$	$df_2 = 3$
$\sum x_i$	1,88+2,26+2,09+2,08+ 2,07 = **10,38**	2,29+2,46+2,51+2,27 = 9,53
$\bar{x} = \frac{\sum x_i}{n}$	$\bar{x}_1$ = 10,38 / 5 = **2,08**	$\bar{x}_2$ = 9,53 / 4 = 2,38
$x_i - \bar{x}$	1,88 – 2,08 = **- 0,20** 2,26 – 2,08 = **0,18** 2,09 – 2,08 = **0,01** 2,08 – 2,08 = **0** 2,07 – 2,08 = **- 0,01**	2,29 – 2,38 = - 0,09 2,46 – 2,38 = 0,08 2,51 – 2,38 = 0,13 2,27 – 2,38 = - 0,11
$(x_i - \bar{x})^2$	$(- 0,20)^2$ = **0,040** $0,18^2$ = **0,032** $0,01^2$ = **0,0001** 0^2 = **0** $(- 0,01)^2$ = **0,0001**	$(- 0,09)^2$ = 0,008 $(- 0,08)^2$ = 0,006 $0,13^2$ = 0,017 $(- 0,11)^2$ = 0,012
$\sum (x_i - \bar{x})^2$	$\sum (x_i - \bar{x}_1)^2$ = **0,07252**	$\sum (x_i - \bar{x}_2)^2$ = 0,04347
Дисперсія $s^2 = \frac{\sum (x_i - \bar{x})^2}{df}$	**0,018**	0,014

Критерій Фішера-Снедекора: $F = \frac{s_1^2}{s_2^2}$	**F=1,29** **Оскільки за табличними даними при df_1=4, df_2=3 та p<0,05 $F_{кр}$=9,12, то виконується умова $F<F_{кр}$. Дисперсії однорідні (у випадку неоднорідності для наступних обчислень можна скористатись критерієм Стьюдента за формулою (44))**
$\sqrt{\left[\frac{\sum(x_i-\bar{x}_1)^2+\sum(x_i-\bar{x}_2)^2}{n_1+n_2-2}\right]\times\left(\frac{n_1+n_2}{n_1\times n_2}\right)}$	$\sqrt{\left[\frac{0{,}072+0{,}043}{5+4-2}\right]\times\left(\frac{5+4}{5\times 4}\right)}$ **= 0,08635**
$\bar{x}_2-\bar{x}_1$	**0,3065**
$t=\frac{\bar{x}_2-\bar{x}_1}{\sqrt{\left[\frac{\sum(x_i-\bar{x}_1)^2+\sum(x_i-\bar{x}_2)^2}{n_1+n_2-2}\right]\times\left(\frac{n_1+n_2}{n_1\times n_2}\right)}}$	**3,5494**

Для визначення однорідностей чи відмінностей між групами необхідно звернутися до таблиці Стьюдента (табл. 1). Беручи до уваги число ступенів свободи $df=n_1 + n_2 - 2$ та рівень статистичної значущості p знаходимо табличне значення t. За табличними даними для статистичної значущості $p=0,05$ і при $df = 7$ значення $t_{кр}=2,37$. В даному випадку $t=3,5494$, оскільки $t > t_{кр}$, це говорить про істотну різницю між досліджуваними рядами при рівні статистичної значущості $p<0,05$. Також за допомогою вбудованої функції Excel *TTECT* можна визначити однорідності чи відмінності між групами.

Алгоритм виконання Normal Student's t-test в Python:

```
import numpy as np
from scipy.stats import t

def student_t_test(x1, x2):

# Calculate the means and variances for each sample
    mean_x1 = np.mean(x1)
    r_mean_x1 = round(mean_x1, 2)
```

```
    mean_x2 = np.mean(x2)
    r_mean_x2 = round(mean_x2, 2)
    numerator = (mean_x2 - mean_x1)
    sum_sq_diff_x1 = sum((xi - mean_x1) ** 2 for xi in
x1)
    sum_sq_diff_x2 = sum((xi - mean_x2) ** 2 for xi in
x2)
    denominator    =    np.sqrt(((sum_sq_diff_x1    +
sum_sq_diff_x2) / (len(x1) + len(x2) - 2)) * ((len(x1)
+ len(x2)) / (len(x1) * len(x2))))
    t_statistic = numerator / denominator
    var_x1 = np.var(x1, ddof=1)
    r_var_x1 = round(var_x1, 2)
    var_x2 = np.var(x2, ddof=1)
    r_var_x2 = round(var_x2, 2)
    print("Group average 1:", r_mean_x1)
    print("Group average 2:", r_mean_x2)
    print("Group variance 1:", r_var_x1)
    print("Group variance 2:", r_var_x2)

# Calculate the degrees of freedom and p-value
    df = len(x1) + len(x2) - 2
    p_value = (1 - t.cdf(abs(t_statistic), df)) * 2

    return t_statistic, p_value

# Example:
data_x1 = [1.88, 2.26, 2.09, 2.08, 2.07]
data_x2 = [2.29, 2.46, 2.51, 2.27]

# Performing a Student's t-test
result_t, p_value = student_t_test(data_x1, data_x2)

# Displaying test results

print(f"  Test  statistics:  {result_t:.4f},  p-value:
{p_value:.4f}")

if p_value < 0.05:
    print("There is statistically significant evidence
that the difference between the means of the two samples
is statistically significant.")
else:
    print("There  is  no  statistically  significant
evidence to support that the difference between the
means of the two samples is statistically significant.")
```

Функція **student_t_test()** приймає два аргументи: **x1** та **x2**, які є списками або масивами значень відповідних вибірок. У тілі функції обчислюється статистика тесту Стьюдента, яка використовується для обчислення p-значення та подальшої інтерпретації результатів тесту. Результати тесту виводяться на екран, разом зі статистикою тесту та відповідним p-значенням. За допомогою умовного оператора **if** тестується статистична значущість результатів залежно від того, чи є p-значення меншим за деякий пороговий рівень значущості (зазвичай 0,05). Якщо p-значення менше за цей поріг, то вважається, що різниця між середніми значеннями двох вибірок статистично значуща. Інакше, різниця не є статистично значущою.

Результатом виконання попереднього коду є:

```
Середнє групи 1: 2.08
Середнє групи 2: 2.38
Дисперсія групи 1: 0.02
Дисперсія групи 2: 0.01
Статистика тесту: 3.5494, p-значення: 0.0094
Є статистично значущі докази на користь того, що різниця між середніми значеннями двох вибірок статистично значуща.
```

Алгоритм виконання тесту Стьюдента у R:

```
# Data
x1 <- c(1.88, 2.26, 2.09, 2.08, 2.07)
x2 <- c(2.29, 2.46, 2.51, 2.27)

# Performing Student's t-test
result <- t.test(x1, x2)

# Displaying test results
cat("Test statistics: ", result$statistic, "\n")
cat("p-value: ", result$p.value, "\n")

if (result$p.value < 0.05) {
  cat("Statistically significant evidence that the difference between the means of the two samples is statistically significant.\n")
} else {
  cat("There is no statistically significant evidence to support that the difference between the means of the two samples is statistically significant.\n")
}
```

У цьому прикладі ми створюємо дві вибірки **x1** та **x2** з використанням заздалегідь визначених значень. Далі викликається функція **t.test()**, якій передаються ці дві вибірки. Результати тесту зберігаються у змінній **result**. Далі виводяться результати тесту на екран за допомогою функції **cat()**. Змінна **result$statistic** містить значення статистики тесту Стьюдента, а **result$p.value** містить відповідне p-значення. За допомогою умовного оператора **if** тестується статистична значущість результатів залежно від того, чи є p-значення меншим за деякий пороговий рівень значущості (зазвичай 0,05). Якщо p-значення менше за цей поріг, то вважається, що різниця між середніми значеннями двох вибірок статистично значуща. Інакше, різниця не є статистично значущою.

Результатом виконання попереднього R-коду буде:

```
x1 <- c(1.88, 2.26, 2.09, 2.08, 2.07)
x2 <- c(2.29, 2.46, 2.51, 2.27)
Статистика тесту:  -3.5494
p-значення:  0.0094
Статистично значущі докази на користь того, що різниця між середніми значеннями двох вибірок статистично значуща.
```

Приклад 21 (*Paired Student's t-test*). В плазмі крові карасів визначали активність лактатдегідрогенази до і після дії на них йонів кобальту в концентрації 100 мг Co^{2+}/л води. Отримали наступні результати:

№ риби	Активність ЛДГ, Од/мг білка	
	до впливу йонів кобальту	*після впливу йонів кобальту*
1	1,33	**1,87**
2	1,12	**1,68**
3	1,23	**1,54**
4	1,15	**1,63**
5	1,13	**1,52**
6	**1,19**	**1,60**

Всі обчислення оформляємо у вигляді таблиці:

Показник, що обраховується	Результати обчислень
n **(в даному випадку, дорівнює кількості риб)**	**6**
d	**1,87-1,33=0,54** **1,68-1,12=0,56** **1,54-1,23=0,31** **1,63-1,15=0,48** **1,52-1,13=0,37** **1,60-1,19=0,41**
$\bar{d} = \frac{\sum d}{n}$	**0,44**
$s_d = \sqrt{\frac{\sum (d - \bar{d})^2}{n-1}}$	**0,17**
$s_{\bar{d}} = \frac{s_d}{\sqrt{n}}$	**0,069**
$t = \frac{\bar{d}}{s_{\bar{d}}}$	**6,38**

df=n-1	**5**

Із табл. 1 знаходимо критичне значення $t_{0,01}$ при рівні статистичної значущості $p<0,01$ і ступенів свободи $df=5$. Воно рівне 4,03, тобто менше за отримане нами (6,38). Таким чином, збільшення активності ЛДГ в плазмі крові карасів після впливу на них йонів кобальту в концентрації 100 мг Co^{2+}/л води істотне при рівні статистичної значущості $p<0,05$.

Код Python для перевірки відмінностей між двома залежними групами за Paired Student's t-test:

```
import numpy as np
from scipy.stats import ttest_rel

# Data
x = np.array(input("Enter the data for the first sample, separated by spaces: ").split(), dtype=float)
y = np.array(input("Enter the data for the second sample, separated by spaces: ").split(), dtype=float)
if len(x) == len(y):
    # Paired Student's t-test
    result = ttest_rel(x, y)
    if result.pvalue < 0.05:
        print(
            " There is statistically significant evidence that the difference between the means of the two samples is statistically significant.")
    else:
        print(
            " There is no statistically significant evidence to support that the difference between the means of the two samples is statistically significant.")
    # Displaying test results
    print("Test statistics:", result.statistic)
    print("p-value:", result.pvalue)
```

```
else:
    print("Error: different amount of data in two samples.")
```

Цей код перевіряє, чи мають два ряди однакову довжину, використовуючи вбудовану функцію **len()**. Якщо довжини двох рядів не збігаються, то виводиться повідомлення про помилку, а парний t-тест Стьюдента не виконується. Якщо довжини збігаються, то можна продовжувати з виконанням парного t-тесту Стьюдента. У цьому прикладі ми використовували функцію **ttest_rel()** з пакету **scipy.stats** для виконання парного t-тесту Стьюдента в Python. Ми спочатку ввели дані для двох залежних вибірок за допомогою функції **input()**, після чого конвертували їх у масиви **NumPy** за допомогою методу **array()** та параметра **dtype=float**. Далі викликали функцію **ttest_rel()** з цими масивами як вхідними параметрами, і результати тесту зберігали в змінну **result**. Потім ми вивели результати тесту на екран, включаючи значення статистики t-тесту та відповідне p-значення. За допомогою умовного оператора **if**, ми тестували статистичну значущість результатів залежно від того, чи є p-значення меншим за деякий критичний рівень значущості (зазвичай 0,05). Якщо p-значення менше за цей поріг, то вважається, що різниця між середніми значеннями двох вибірок статистично значуща. В іншому випадку, немає достатніх доказів для того, щоб стверджувати, що різниця між середніми значеннями двох вибірок статистично значуща.

Результатом виконання цього коду за вказаним прикладом є:

```
Введіть дані для першої вибірки, розділені пробілами: >? 1.33 1.12 1.23 1.15 1.13 1.19
Введіть дані для другої вибірки, розділені пробілами: >? 1.87 1.68 1.54 1.63 1.52 1.60
```

```
Є статистично значущі докази на користь того, що різниця між середніми значеннями двох вибірок статистично значуща.
Статистика тесту: -11.463947182283425
p-значення: 8.848365547682244e-05
```

Код R для перевірки відмінностей між двома залежними групами за Paired Student's t-test:

```
# Data
x <- c(1.33, 1.12, 1.23, 1.15, 1.13, 1.19)
y <- c(1.87, 1.68, 1.54, 1.63, 1.52, 1.60)

# Checking for the same amount of data in two samples
if (length(x) != length(y)) {
  stop("Error: Different amount of data in the two samples.")
}

# Performing Student's paired t-test
result <- t.test(x, y, paired=TRUE)

# Displaying test results
cat("Test statistics: ", result$statistic, "\n")
cat("p-value: ", result$p.value, "\n")

if (result$p.value < 0.05) {
  cat("There is statistically significant evidence that the difference between the means of the two samples is statistically significant.\n")
} else {
  cat("There is no statistically significant evidence to support that the difference between the means of the two samples is statistically significant.\n")
}
```

У цьому коді ми використовуємо функцію **t.test()** для виконання парного t-тесту Стьюдента. Спочатку ми вводимо дані для двох залежних вибірок. Далі ми перевіряємо, чи мають два ряди однакову довжину, використовуючи вбудовану функцію **length()**. Якщо

довжини двох рядів не збігаються, то виводиться повідомлення про помилку, а парний t-тест Стьюдента не виконується. Якщо довжини збігаються, то ми викликаємо функцію **t.test()** з цими векторами. Функція **t.test()** приймає два вектори даних та параметр **paired=TRUE**, що вказує на виконання парного t-тесту Стьюдента для залежних вибірок. Результатом є список з різними характеристиками тесту, включаючи статистику **t** та **p**-значення. Нарешті, ми виводимо результати тесту, включаючи статистику **t** та **p**-значення, та використовуємо умовний оператор для визначення, чи є статистично значущі різниці між середніми значеннями двох вибірок за рівнем значущості 0,05.

Результатом виконання цього коду за вказаним прикладом є:

```
Статистика тесту:  -11.46395
p-значення:  8.848366e-05
Є статистично значущі докази на користь того, що
   різниця між середніми значеннями двох вибірок
   статистично значуща.
```

Критерій Стьюдента, як уже було сказано вище, призначений для порівняння двох груп, проте на практиці він використовується для оцінки великої кількості груп через попарне їх порівняння. При цьому спрацьовує **ефект багатьох порівнянь**. Наприклад, досліджували вплив йонів важких металів А і Б на активність каталази. Дослідження проводили на трьох групах:

група А – група, що отримала іони металу А;

група Б – група, що отримала іони металу Б;

група С – група, що не отримала жодного препарату.

За допомогою критерію Стьюдента проводили 3 парних порівняння: групу А порівнювали з групою В, групу Б – з групою В і наостанок А з Б. Отримавши достатньо високе значення *t* в кожному із трьох порівнянь, повідомили, що «*p<0,05*». Це означає, що ймовірність помилкового висновку про існування відмінностей не перевищує 5%. Але це неправильно, бо ймовірність помилки значно перевищує 5%. Чому так? В дослідженні був прийнятий 5% рівень значущості. Отже, ймовірність помилитися при порівнянні груп А і В – 5%. Проте, слід взяти до уваги, що ми помилимося в 5% випадків при порівнянні груп Б і В і в 5% випадків – при порівнянні груп А і Б. В загальному випадку ця ймовірність рівна:

$$P' = 1 - (1 - 0{,}05)^k, \qquad \textbf{(48)}$$

де k – число порівнянь.

Тому в даному випадку ймовірність помилитися хоча би в одному із порівнянь складає приблизно 3×5=15%. Якщо порівнюються чотири групи між собою, то ми отримуємо шість пар. Тоді при рівні статистичної значущості *p<0,05* ми отримуємо значення 5×6 = 30%. І коли дослідник, виявивши таким чином «ефективний» препарат, говорить про ймовірність помилки 5%, то насправді ця ймовірність складає 30%.

Отже, із всього вищесказаного можна зробити наступні висновки:

1. Критерій Стьюдента може бути використаний для перевірки гіпотези про різницю середніх значень тільки для двох груп.
2. Якщо критерій Стьюдента був використаний для перевірки різниці між декількома групами, то істинний рівень

значущості можна отримати, перемноживши рівень значущості на число можливих порівнянь.

3. Якщо експеримент передбачає велику кількість груп потрібно використати дисперсійний аналіз.

Проте для порівняння більш, ніж трьох груп попарно між собою критерій Стьюдента можна використовувати, ввівши *поправку Бонферроні*, але цю поправку слід використовувати тільки тоді, коли порівнюється невелика кількість груп між собою (до 8). Проте найчастіше для порівняння великої кількості груп між собою використовують критерій Ньюмена-Коулса (див. далі по тексту), який має значно більшу чутливість у порівнянні з критерієм Стьюдента з поправкою Бонферроні.

6.2.2. *U*-критерій Манна–Вітні як непараметричний аналог непарного критерію Стьюдента

U-критерій Манна–Вітні (*Mann–Whitney U-test* або *Wilcoxon–Mann–Whitney test,* або *Wilcoxon rank-sum test*)

Цей критерій являє собою непараметричну альтернативу *t*-критерію для незалежних вибірок. Він підходить для порівняння малих вибірок: у кожній з вибірок має бути не менше 4 значень ознаки. Допускається, щоб в одній вибірці було 3 значення, але в другій тоді повинно бути не менше п’яти.

Алгоритм обчислення *U*-критерію Манна–Вітні наступний:

1. Дані двох груп об’єднують і впорядковують по збільшенню. Ранг 1 отримує найменше із всіх значень, ранг 2 – наступне тощо. Найбільший ранг отримує найбільше значення серед двох груп.

Якщо значення збігаються, їм надають один і той самий середній ранг (наприклад, якщо два значення знаходяться на 3-му і 4-му місцях, то вони отримують ранг 3,5).

2. Підраховують окремо суму рангів (T), що припали на частку елементів першої вибірки, і окремо – на частку елементів другої вибірки.

3. За формулою обчислюють значення U-критерію Манна–Вітні:

$$U = n_1 \times n_2 + \frac{n_x(n_x+1)}{2} - T_x, \quad \textbf{(49)}$$

де n_1 і n_2 – об'єми вибірок, що порівнюються;

n_x – об'єм вибірки, в якої більша сума рангів (T_x).

T_x – більша сума рангів.

4. Обчислене значення U порівнюють із критичним значенням (табл. 14). Відмінності тим вищі, чим менше значення U.

Таблиця 12. Критичні значення U-критерію Манна–Вітні ($P<0,05$)

Розмір меншої вибірки	Розмір більшої вибірки															
	5	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**
3	0	1	1	2	2	3	3	4	4	5	5	6	6	7	7	8
4	1	2	3	4	4	5	6	7	8	9	10	11	11	12	13	13
5	2	3	5	6	7	8	9	11	12	13	14	15	17	18	19	20
6		5	6	8	10	11	13	14	16	17	19	21	22	24	25	27
7			8	10	12	14	16	18	20	22	24	26	28	30	32	34
8				13	15	17	19	22	24	26	29	31	34	36	38	41
9					17	20	23	26	28	31	34	37	39	41	45	48
10						23	26	29	33	36	39	42	45	48	52	55
11							30	33	37	40	44	47	51	55	58	62
12								37	41	45	49	53	57	61	65	69
13									45	50	54	59	63	67	72	76
14										55	59	64	67	74	78	83
15											64	70	75	80	85	90

16			75	81	86	92	98
17				87	93	99	105
18					99	106	112
19						113	119
20							127

Приклад 22. Перевіримо наявність істотних відмінностей між двома групами за даними із прикладу 20, використовуючи *U*-критерій Манна–Вітні.

За U-критерієм Манна–Вітні:

1. Надаємо ранги даним та обчислюємо суми рангів (*T*) для груп:

Контрольна група		**Група риб, що піддавалась дії йонів кобальту в концентрації 100 мг Co^{2+}/л води**	
Ранг	Активність ЛДГ, Од/мг білка	Ранг	**Активність ЛДГ, Од/мг білка**
1	1,88	7	2,29
5	2,26	8	2,46
4	2,09	9	2,51
3	2,08	6	2,27
2	2,07		
T_1=15		**T_2=30**	

2. Порівнюємо суми рангів груп між собою. В нашому випадку $T_2 > T_1$.

3. За формулою **(49)** обчислюємо значення *U*-критерію Манна–Вітні:

$$U = 5 \times 4 + \frac{4(4+1)}{2} - 30 = 0.$$

Порівнюємо отримане нами значення критерію *U* із його табличним значенням $U_{кр}$. За табл. 14 при порівнянні груп чисельністю 4 і 5 отримуємо, що при $p<0,05$ $U_{кр}$ рівне 1. Ми отримали, що $U=0$, тому гіпотеза про однорідність вибірок

відкидається (відмінності істотні при цих рівнях статистичної значущості).

Алгоритм виконання Mann-Whitney U-test в Python:

```
from scipy.stats import mannwhitneyu

# User data entry
group1 = [float(x) for x in input("Enter the data for
the first sample, separated by spaces: ").split()]
group2 = [float(x) for x in input("Enter the data for
the second sample, separated by spaces: ").split()]

# application of the Mann-Whitney test
stat, p = mannwhitneyu(group1, group2)

# output of results
print('stat=%.3f, p=%.3f' % (stat, p))

# interpretation of results
alpha = 0.05
if p > alpha:
    print('No statistically significant differences
between groups ')
else:
    print('There are statistically significant
differences between the groups')
```

Цей код використовує бібліотеку **scipy** для виконання тесту Манна–Вітні і виводить результати на екран. Щоб застосувати цей код до своїх власних даних, просто внесіть значення **group1** та **group2** на свої власні дані.

Результатом виконання попереднього коду є:

```
    Введіть дані для першої вибірки, розділені пробілами:
>? 1.88 2.26 2.09 2.08 2.07
    Введіть дані для другої вибірки, розділені пробілами:
>? 2.29 2.46 2.51 2.27
    stat=0.000, p=0.016
    Є статистично значущі різниці між групами
```

Алгоритм виконання тесту Манна–Вітні *у R:*

```
# Data
group1 <- c(1.88, 2.26, 2.09, 2.08, 2.07)
group2 <- c(2.29, 2.46, 2.51, 2.27)
# application of the Mann-Whitney test
result <- wilcox.test(group1, group2)
# output of results
print(result)
# interpretation of results
alpha <- 0.05
if(result$p.value > alpha) {
  print('No statistically significant differences
between groups ')
} else {
  print('There are statistically significant
differences between the groups')
}
```

Цей код використовує функцію **wilcox.test()** для виконання тесту Манна–Вітні і виводить результати на екран. Щоб застосувати цей код до своїх власних даних, просто замініть значення **group1** та **group2** на свої власні дані.

Результатом виконання попереднього R-коду є:

```
> # введення даних
> group1 <- c(1.88, 2.26, 2.09, 2.08, 2.07)
> group2 <- c(2.29, 2.46, 2.51, 2.27)

Wilcoxon rank sum exact test
data:  group1 and group2
W = 0, p-value = 0.01587
alternative hypothesis: true location shift is not
equal to 0
> # інтерпретація результатів
[1] «Є статистично значущі різниці між групами»
```

Підведемо короткий підсумок по критеріях, які можна застосувати при порівнянні двох незалежних груп. Отже, при

застосуванні параметричного (t-критерій Стьюдента для незалежних вибірок) та непараметричного (U-критерій Манна–Вітні) критеріїв для одних і тих самих даних (приклади 19 і 21) ми отримали однакові результати про те, що середні значення для виборок істотно *відрізняються одне від одного при заданих рівнях значущості p. Тому ці критерії ми пропонуємо застосовувати разом, якщо дослідник має сумніви у чутливості того чи іншого критерію через малу вибірку (n<15).*

6.2.3. *W*-критерій Вілкоксона: непараметричний аналог парного критерію Стьюдента

Сам алгоритм для обчислення *W*-критерію Вілкоксона (*Wilcoxon's test*) наступний:

1. Спочатку потрібно обчислити зміни показників до і після певного впливу. Відкидаються пари, в яких зміни дорівнюють нулю.

2. Ці зміни розміщують в порядку зростання і присвоюють їм ранги (див. алгоритм для обчислення U-критерію Манна–Вітні).

3. Присвоюють кожному рангу знак у відповідності з напрямом змін: якщо значення збільшилось – «+», а якщо зменшилось – «–».

4. Обчислюють суму знакових рангів *W* (див. приклад 22).

5. Порівнюють величину *W* із табличним значенням $W_{кр}$ (табл. 13). Якщо виконується нерівність $W \geq W_{кр}$, то дані між собою до і після впливу певної речовини істотно відрізняються при певному рівні значущості *p*.

Таблиця 13. Критичні значення *W*-критерію Вілкоксона (двосторонній варіант)

n	*W*	*p*	*n*	*W*	*p*
5	15	0,062	13	65	0,022
6	21	0,032		57	0,048
	19	0,062	14	73	0,020
7	28	0,016		63	0,050
	24	0,046	15	80	0,022
8	32	0,024		70	0,048
	28	0,054	16	88	0,022
9	39	0,020		76	0,050
	33	0,054	17	97	0,020
10	45	0,020		83	0,050
	39	0,048	18	105	0,020
11	52	0,018		91	0,048
	44	0,054	19	114	0,020
12	58	0,020		98	0,050
	50	0,052	20	124	0,020
				106	0,048

Приклад 23. Перевіримо наявність істотних відмінностей між двома групами за даними із прикладу 21, використовуючи *W*-критерій Вілкоксона.

1. Надаємо знакові ранги даним та обчислюємо *W*:

№ риби	Активність ЛДГ, Од/мг білка		Величина змін	Ранг змін	Знаковий ранг змін
	до впливу йонів кобальту	*після впливу йонів кобальту*			
1	1,33	1,87	0,54	5	+5
2	1,12	1,68	0,56	6	+6
3	1,23	1,54	0,31	1	+1
4	1,15	1,63	0,48	4	+4
5	1,13	1,52	0,37	2	+2
6	1,19	1,60	0,41	3	+3
					W=21

2. Порівнюємо отримане нами значення критерію *W* із його табличним значенням $W_{кр}$. За табл. 18 при чисельності групи *n=6* отримуємо, що $W_{кр}$ знаходиться в межах 19-21. Ми отримали, що *W=21*, тому гіпотеза про рівність вибірок відкидається.

Алгоритм виконання W-критерію Вілкоксона в Python:

```
from scipy.stats import wilcoxon

# Дані (парні вимірювання: група1 = "до", група2 =
"після")
group1 = [1.33, 1.12, 1.23, 1.15, 1.13, 1.19]
group2 = [1.87, 1.68, 1.54, 1.63, 1.52, 1.60]

# Обчислюємо різниці
differences = [b - a for a, b in zip(group1, group2)]

# Ранжуємо за абсолютними значеннями
ranked_diff = sorted(differences, key=abs)

# Формуємо "signed ranks"
signed_ranks = [
    (ranked_diff.index(diff) + 1) if diff > 0 else -
(ranked_diff.index(diff) + 1)
    for diff in differences
]

print("Significant ranks of change:", signed_ranks)

# Обчислюваний W = сума знакових рангів
W_manual = sum(signed_ranks)
print("W (manual):", W_manual)

# Використовуємо правильний парний тест Вілкоксона
statistic,   p_value   =   wilcoxon(group2,   group1,
zero_method="wilcox", alternative="two-sided")

print("W (scipy wilcoxon):", statistic)
print("p-value:", p_value)

# Перевірка значущості
alpha = 0.05
if p_value < alpha:
    print("Є статистично значуща різниця між групами.")
else:
```

```
    print("Статистично значущої різниці між групами немає.")
```

Цей код використовує бібліотеку **SciPy** для застосування тесту Вілкоксона та обчислення p-значення. Обчислюється статистика Вілкоксона (**W**) і **p**-значення, яке використовується для того, щоб визначити, чи є статистично значуща різниця між групами. Якщо **p**-значення менше за **alpha**, то є статистично значуща різниця між групами. В іншому випадку немає статистично значущої різниці між групами.

Результатом виконання попереднього коду є:

```
Знакові ранги змін: [5, 6, 1, 4, 2, 3]
W-статистика: 21
p-значення: 0.03125
Є статистично значуща різниця між групами.
```

Алгоритм виконання тесту Вілкоксона (**W**) *у R:*

```
# Data
group1 <- c(1.87, 1.68, 1.54, 1.63, 1.52, 1.60)
group2 <- c(1.33, 1.12, 1.23, 1.15, 1.13, 1.19)

# Conducting a Wilcoxon signed-rank test
wilcox.test(group1, group2, paired = TRUE)
```

Результат виконання цього коду буде містити такі параметри, як статистика тесту, p-значення:

```
# Створення вибірок для порівняння
group1 <- c(1.87, 1.68, 1.54, 1.63, 1.52, 1.60)
group2 <- c(1.33, 1.12, 1.23, 1.15, 1.13, 1.19)

        Wilcoxon signed rank exact test
data:  group1 and group2
W = 21, p-value = 0.03125
```

```
    alternative hypothesis: true location shift is not
equal to 0
```

Отже, при застосуванні параметричного (парний критерій Стьюдента) та непараметричного (W-критерій Вілкоксона) критеріїв для залежних даних (приклади 20 і 22) ми отримали однакові результати про те, що вибірки між собою істотно відрізняються при заданих рівнях значущості p. Тому ці критерії ми також пропонуємо застосовувати разом, якщо дослідник сумнівається у чутливості того чи іншого критерію через малу кількість повторів (n<15).

6.3. Порівняння трьох і більше груп між собою: доцільність використання параметричних чи непараметричних критеріїв

6.3.1. Критерій Ньюмена-Коулса

Для порівняння трьох і більше груп між собою можна використати ряд критеріїв. Один із них – це *критерій Ньюмена-Коулса (Newman-Keuls test).*

Алгоритм визначення значення цього критерію наступний:

1. Спочатку потрібно за допомогою дисперсійного аналізу перевірити нульову гіпотезу про рівність всіх середніх, тобто:

1.1. Обчислюємо міжгрупове число ступенів свободи ($df_м$):

$$df_м = m - 1 \ , \qquad \textbf{(50)}$$

де m – число груп, що порівнюються.

1.2. Обчислюємо внутрішньогрупове число ступенів свободи ($df_в$):

$$df_в = N - m \ , \qquad \textbf{(51)}$$

де $N = \sum n_i$.

1.3. Визначаємо внутрішньогрупову дисперсію ($s_в^2$):

$$s_в^2 = \frac{\sum s_i^2 n_i}{N} \ , \qquad \textbf{(52)}$$

де $s_i^2 = \frac{\sum (x_i - \bar{x}_i)^2}{n_i - 1}$ – вибіркова дисперсія в *i*-тій групі;

n_i – об'єм групи.

1.4. Обчислюємо міжгрупову дисперсію ($s_м^2$):

$$s_м^2 = \frac{\sum n_i (\bar{x}_i - \bar{x})^2}{df_м} \ , \qquad \textbf{(53)}$$

де $\bar{x}$ – вибіркове середнє об'єднаної вибірки, яке можна обчислити за формулою:

$$\bar{x} = \frac{\sum n_i \bar{x}_i}{N} \qquad \textbf{(54)}$$

1.5. Обчислюємо критерій *F* за формулою:

$$F = \frac{s_м^2}{s_в^2} \qquad \textbf{(55)}$$

1.6. Використовуючи табл. 14 для $df_м$ і $df_в$, враховуючи рівень статистичної значущості $p < 0{,}05$, знаходимо критичне значення $F_{кр}$.

Якщо $F > F_{кр}$, то гіпотеза про рівність середніх значень вибірок відхиляється.

1.7. Якщо гіпотеза про рівність всіх середніх значень відкидається, то ці дані впорядковують за зростанням і порівнюються попарно, щоразу обчислюючи значення критерію Ньюмена-Коулса, використовуючи формулу:

$$q = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{s_в^2\left(\frac{1}{n_1} + \frac{1}{n_2}\right)}}, \qquad \textbf{(56)}$$

де $\bar{x}_1$ і $\bar{x}_2$ – середні значення, які потрібно порівняти між собою;

$s_в^2$ – внутрішньогрупова дисперсія;

n_1 і n_2 – об'єми вибірок відповідних груп.

1.8. Обчислене значення q порівнюється з критичним значенням $q_{кр}$ (табл. 14). Критичне значення $q_{кр}$ залежить від статистичної значущості p (ймовірність помилково виявити відмінності хоча б в одній з усіх пар, що порівнюються, тобто справжній рівень статистичної значущості), числа ступенів свободи $df = N–m$ (N – сума чисельності всіх груп, m – число груп) і величини l, яка називається інтервалом порівняння. Інтервал порівняння визначається наступним чином. Якщо середні значення, що порівнюються, стоять відповідно на j-му і i-му місці в упорядкованому ряді, то інтервал порівняння $l=j – i + 1$. Наприклад, при порівнянні 4-го і 1-го членів цього ряду l=4–1+1 = 4, при порівнянні 2-го і 1-го l=2 – 1 + 1 = 2.

Якщо $q>q_{кр}$, то групи істотно відрізняються між собою при заданих рівнях значущості p.

Примітка. *Цей критерій залежить від послідовності порівнянь, тому їх потрібно проводити в певній послідовності, яка задається двома правилами*:

1. Якщо середні значення величин розмістити від найменшого до найбільшого (від 1 до m), то спочатку потрібно порівняти найбільше значення з найменшим, тобто m з 1, потім m з 2-м, 3-м і так далі. Потім передостаннє ($m–1$) – з 1-м, 2-м і так далі. Наприклад, у

випадку чотирьох груп порядок порівнянь наступний: 4-1, 4-2, 4-3, 3-1, 3-2, 2-1.

2. Якщо будь-які середні величини не відрізняються, то всі середні значення, що лежать між ними, також не відрізняються. Наприклад, якщо не виявлено відмінностей між 3-м і 1-м середніми, то не потрібно порівнювати 3-є з 2-м і 2-е з 1-м.

Таблиця 14. Критичне значення $q_{кр}$ при статистичній значущості $p<0,05$

df	**Інтервал порівняння *l***								
	2	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
6	3,461	4,339	4,896	5,305	5,628	5,895	6,122	6,319	6,493
7	3,344	4,165	4,681	5,060	5,359	5,606	5,815	5,998	6,158
8	3,261	4,041	4,529	4,886	5,167	5,399	5,597	5,767	5,918
9	3,199	3,949	4,415	4,756	5,024	5,244	5,432	5,595	5,739
10	3,151	3,877	4,327	4,654	4,912	5,124	5,305	5,461	5,599
11	3,113	3,820	4,256	4,574	4,823	5,028	5,202	5,353	5,487
12	3,082	3,773	4,199	4,508	4,751	4,950	5,119	5,265	5,395
13	3,055	3,735	4,151	4,453	4,690	4,885	5,049	5,192	5,318
14	3,033	3,702	4,111	4,407	4,639	4,829	4,990	5,131	5,254
15	3,014	3,674	4,076	4,367	4,595	4,782	4,940	5,077	5,198
16	2,998	3,649	4,046	4,333	4,557	4,741	4,897	5,031	5,150
17	2,984	3,628	4,020	4,303	4,524	4,705	4,858	4,991	5,108
18	2,971	3,609	3,997	4,277	4,495	4,673	4,824	4,956	5,071
19	2,960	3,593	3,977	4,253	4,469	4,645	4,794	4,924	5,038
20	2,950	3,578	3,958	4,232	4,445	4,620	4,768	4,896	5,008
24	2,919	3,532	3,901	4,166	4,373	4,541	4,684	4,807	4,915
30	2,888	3,486	3,845	4,102	4,302	4,464	4,602	4,720	4,824

Наведемо приклад використання цього критерію для порівняння дослідних груп між собою та з контрольною групою.

Приклад 24. В результаті досліджень впливу йонів нікелю на активність супероксиддисмутази в зябрах карася сріблястого було отримано наступні результати (наведені власні дані):

Групи риб	Активність СОД, Од/мг білка
Контроль	51,6; 48,2; 69,4; 104; 92,0; 87,9
10 мг/л Ni^{2+}	68,5; 78,5; 78,2; 74,5; 76,7; 74,1
25 мг/л Ni^{2+}	55,8; 41,4; 56,2; 65,8; 42,0; 60,0
50 мг/л Ni^{2+}	41,4; 43,7; 37,9; 42,4; 27,3; 46,4

Порівняймо дослідні групи між собою та контрольною групою за допомогою критерію Ньюмена-Коулса.

1. Попередні обчислення можна оформити у вигляді таблиці:

Показник	Контроль	10 мг/л Ni^{2+}	25 мг/л Ni^{2+}	50 мг/л Ni^{2+}
x_i	51,6 48,2 69,4 104 92,0 87,9	68,5 78,5 78,2 74,5 76,7 74,1	55,8 41,4 56,2 65,8 42,0 60,0	41,4 43,7 37,9 42,4 27,3 46,4
n	6	6	6	6
m		4		
$df_м = m-1$		3		
$df_в = m(n-1)$		20		
$N = \sum n_i$		24		
$\bar{x}_i = \frac{\sum x_i}{n}$	$\bar{x}_1 = 75,5$	$\bar{x}_2 = 75,1$	$\bar{x}_3 = 53,5$	$\bar{x}_4 = 39,9$
$s_i^2 = \frac{\sum (x_i - \bar{x}_i)^2}{n-1}$	518	13,7	97,0	45,6
$s_в^2 = \frac{\sum s_i^2 n_i}{N}$		169		
$\bar{x} = \frac{\sum n_i \bar{x}_i}{N}$		61		
$s_м^2 = \frac{\sum n_i (\bar{x}_i - \bar{x})^2}{df_м}$		1821		
$F = \frac{s_м^2}{s_в^2}$		10,8		
***$F_{кр}$* (табл. 16)**		3,10		
***F*>*$F_{кр}$*, 10,8>3,10. Гіпотеза про рівність середніх значень груп даних відхиляється**				

2.1. За формулою **(56)**, порівнюючи контрольну групу із тією групою, яка найбільш від неї відрізняється (в даному випадку з

групою риб, що експоновані до йонів нікелю концентрацією 50 мг/л Ni^{2+}), обчислюємо *q*:

$$q_1 = \frac{75{,}5 - 39{,}9}{\sqrt{169\left(\frac{1}{6} + \frac{1}{6}\right)}} = 4{,}74.$$

2.2. При статистичній значущості *p<0,05*, ступенях свободи *df = 20* (*df = N – m*) та інтервалі порівняння *l = 4* критичне значення критерію $q_{кр}$ дорівнює 3,96 (табл. 16).

2.3. Умова $q>q_{кр}$ виконується, оскільки 4,74>3,96. Тому групи даних «Контроль» і «50 мг/л Ni^{2+}» між собою істотно відрізняються при статистичній значущості *p<0,05*.

3.1. Порівнюємо між собою групи даних «Контроль» і «25 мг/л Ni^{2+}»:

$$q_2 = \frac{75{,}5 - 53{,}5}{\sqrt{169\left(\frac{1}{6} + \frac{1}{6}\right)}} = 2{,}93.$$

3.2. При *p<0,05*, *df = 20* (*df = N–m*) і *l=3* критичне значення критерію $q_{кр}$ дорівнює 3,58 (табл. 19).

3.3. Умова $q>q_{кр}$ не виконується, оскільки 2,93<3,58. Тому групи даних «Контроль» і «25 мг/л Ni^{2+}» між собою не відрізняються, а, отже, не відрізняються між собою також і групи даних: «Контроль» та «10 мг/л Ni^{2+}», «10 мг/л Ni^{2+}» та «25 мг/л Ni^{2+}».

4.1. Порівнюємо між собою групи даних «10 мг/л Ni^{2+}» і «50 мг/л Ni^{2+}»:

$$q_3 = \frac{75{,}1 - 39{,}9}{\sqrt{169\left(\frac{1}{6} + \frac{1}{6}\right)}} = 4{,}69.$$

4.2. При *p < 0,05*, *df = 20* (*df = N – m*) і *l = 3* критичне значення критерію $q_{кр}$ дорівнює 3,58 (табл. 16).

4.3. Умова $q>q_{кр}$ виконується, оскільки 4,69>3,58. Тому групи даних: «10 мг/л Ni^{2+}» і «50 мг/л Ni^{2+}» між собою істотно відрізняються при $p<0,05$.

5.1. Порівнюємо між собою групи даних: «25 мг/л Ni^{2+}» і «50 мг/л Ni^{2+}»:

$$q_4 = \frac{53,5 - 39,9}{\sqrt{169\left(\frac{1}{6}+\frac{1}{6}\right)}} = 1,81.$$

5.2. При $p<0,05$, $df = 20$ ($df = N–m$) і $l = 2$ критичне значення критерію $q_{кр}$ дорівнює 2,95 (табл. 16).

5.3. Умова $q>q_{кр}$ не виконується, оскільки 1,81<2,95. Тому групи даних: «25 мг/л Ni^{2+}» і «50 мг/л Ni^{2+}» між собою не відрізняються.

Отже, в результаті обчислень згідно з критерієм Ньюмена-Коулса наступні групи **«Контроль»** і **«50 мг/л Ni^{2+}»**, **«10 мг/л Ni^{2+}»** і **«50 мг/л Ni^{2+}»** істотно відрізняються між собою при $p<0,05$.

6.3.2. Критерій Тюкі (Tukey's test)

Тест Тюкі є методом, який використовується для порівняння середніх значень між більш ніж двома групами, які були піддані однофакторному дисперсійному аналізу. Основна ідея полягає в порівнянні середніх значень кожної пари груп і обчисленні різниці між ними.

Тест Тюкі обчислює верхню межу, яка є мінімальною різницею між двома середніми значеннями, щоб вважати цю пару відмінними на заданому рівні значущості. Формула для обчислення верхньої межі (q) має вигляд:

$$q = Q\sqrt{\frac{MSE}{n}} \quad , \qquad \textbf{(57)}$$

де Q – критичне значення статистики Тюкі для заданого рівня довіри і кількості груп; MSE – середньоквадратичне відхилення всередині груп, яке знаходять за допомогою дисперсійного аналізу (ANOVA); n – загальна кількість спостережень (загальна кількість даних у групах, які порівнюються між собою).

Якщо різниця між середніми значеннями будь-якої пари груп більша за верхню межу, то ці групи вважаються статистично значущими.

Тест Тюкі враховує загальний рівень значущості для всіх порівнянь, тому його вважають точнішим, ніж інші методи множинних порівнянь. Він може бути застосований до будь-якої кількості груп і не потребує рівних вибірок для кожної групи.

Код Python для виконання порівнянь між групами за критерієм Тюкі:

```
import numpy as np
from statsmodels.stats.multicomp import pairwise_tukeyhsd

group1 = input("Enter the data for group 1 (separated by commas): ")
group1 = list(map(float, group1.split(",")))
group2 = input("Enter the data for group 2 (separated by commas): ")
group2 = list(map(float, group2.split(",")))
group3 = input("Enter the data for group 3 (separated by commas): ")
group3 = list(map(float, group3.split(",")))
group4 = input("Enter the data for group 4 (separated by commas): ")
```

```
group4 = list(map(float, group4.split(",")))

# Combine data into a single array
data = np.concatenate([group1, group2, group3, group4])

# Creating an array with group labels
labels = ['group1'] * len(group1) + ['group2'] *
len(group2) + ['group3'] * len(group3) + ['group4'] *
len(group4)

# Calling the pairwise_tukeyhsd() function to get
results
tukey_results = pairwise_tukeyhsd(data, labels)

# Displaying the results
print(tukey_results)
```

Результат виконання цього коду для вищевказаних даних в Python:

```
Введіть дані для групи 1 (через кому):
51.6,48.2,69.4,104,92.0,87.9
Введіть дані для групи 2 (через кому):
68.5,78.5,78.2,74.5,76.7,74.1
Введіть дані для групи 3 (через кому):
55.8,41.4,56.2,65.8,42.0,60.0
Введіть дані для групи 4 (через кому):
41.4,43.7,37.9,42.4,27.3,46.4
```

Multiple Comparison of Means – Tukey HSD, FWER=0.05

group1	group2	meandiff	p-adj	lower	upper	reject
group1	group2	-0.4333	0.9999	-21.4201	20.5535	False
group1	group3	-21.9833	0.038	-42.9701	-0.9965	True
group1	group4	-35.6667	0.0006	-56.6535	-14.6799	True
group2	group3	-21.55	0.0428	-42.5368	-0.5632	True
group2	group4	-35.2333	0.0007	-56.2201	-14.2465	True
group3	group4	-13.6833	0.2914	-34.6701	7.3035	False

Отже, є відмінності між наступними групами: 1 і 3, 1 і 4, 2 і 3 та 2 і 4.

Код R для виконання порівнянь між групами за критерієм Тюкі:

```
# ANOVA + Tukey HSD Test

# Дані для 4 груп
group1 <- c(51.6, 48.2, 69.4, 104, 92.0, 87.9)
group2 <- c(68.5, 78.5, 78.2, 74.5, 76.7, 74.1)
group3 <- c(55.8, 41.4, 56.2, 65.8, 42.0, 60.0)
group4 <- c(41.4, 43.7, 37.9, 42.4, 27.3, 46.4)

# Об’єднання даних у один вектор
data <- c(group1, group2, group3, group4)

# Групові мітки
labels <- factor(rep(c("group1", "group2", "group3",
"group4"),
                     times = c(length(group1),
length(group2),
                               length(group3),
length(group4))))

# Проведення однофакторного ANOVA
anova_model <- aov(data ~ labels)

# Тест Тюкі для множинних порівнянь
tukey_results <- TukeyHSD(anova_model)

# Вивід результатів
print(summary(anova_model))    # результат ANOVA
print(tukey_results)           # результат Tukey HSD
```

Цей кодСтворює 4 групи з вашими числами, Збирає всі дані в один масив (data). Додає факторні мітки (labels), які вказують, до якої групи належить кожне спостереження. Виконує ANOVA (aov), щоб перевірити, чи є значущі відмінності між групами.

Якщо різниця є, функція TukeyHSD() покаже попарні порівняння між усіма групами..

Результат виконання тесту:

```
Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = data ~ labels)

$labels
                      diff       lwr         upr     p adj
group2-group1  -0.4333333 -21.42014  20.5534780 0.9999282
group3-group1 -21.9833333 -42.97014  -0.9965220 0.0379948
group4-group1 -35.6666667 -56.65348 -14.6798553 0.0006439
group3-group2 -21.5500000 -42.53681  -0.5631887 0.0428442
group4-group2 -35.2333333 -56.22014 -14.2465220 0.0007345
group4-group3 -13.6833333 -34.67014   7.3034780 0.2913511
```

Отже, є відмінності між наступними групами: 1 і 3, 1 і 4, 2 і 3 та 2 і 4.

6.3.3. Критерій Дункана для порівняння груп між собою

Тест Дункана – це статистичний тест, що базується на порівнянні середніх значень груп за допомогою множинних порівнянь середніх значень.

Він базується на наступних формулах:

1. Розрахунок середнього значення (M) за формулою (2) для кожної групи, що порівнюється.

2. Розрахунок загального середнього значення ($M_{заг.}$) для всіх груп:

$$M_{заг.} = \sum x_i \ / \ N, \qquad \textbf{(58)}$$

де $\sum x_i$ – сума значень груп, що порівнюються, N – кількість значень у всіх групах.

3. Розрахунок міжгрупової дисперсії ($\sigma_{між}$):

$$\sigma_{між} = ((M_1-M_{заг.})^2 + \ldots + (M_n-M_{заг.})^2) / df_м, \quad \textbf{(59)}$$

де $((M_1-M_{заг.})^2 + \ldots + (M_n-M_{заг.})^2)$ – сума квадратів різниць між середніми значеннями кожної групи та загальним середнім значенням;

$df_м$ – кількість ступенів свободи міжгрупової дисперсії, яка розраховується як (*кількість груп – 1*).

4. Розрахунок внутрішньогрупової дисперсії ($\sigma_{вн}$):

$$\sigma_{вн} = ((x_1-M)^2 + \ldots + (x_n-M)^2) / df_в , \quad \textbf{(60)}$$

де $((x_1-M)^2 + \ldots + (x_n-M)^2)$ – сума квадратів різниць між кожним значенням в групі та середнім значенням групи;

$df_в$ – кількість ступенів свободи внутрішньогрупової дисперсії, яка розраховується як: загальна кількість спостережень – кількість груп (наприклад, якщо у вас 24 спостереження, розподілені на 4 групи, то $df_в = 24 – 4 = 20$).

5. Обчислити статистику критерію Дункана (F_{stat}) за формулою:

$$F_{stat} = \sigma_{між} / \sigma_{вн} \quad \textbf{(61)}$$

6. Обчислити критичне значення критерію Дункана ($F_{кр.}$) за допомогою таблиці критичних значень F-розподілу з рівнем значущості α та ступенями свободи для $\sigma_{між}$ та $\sigma_{вн}$ дисперсій. Для обчислення критичного значення критерію Дункана (*Fкр.*) за допомогою таблиці критичних значень F-розподілу необхідно виконати кілька кроків:

а) знайти ступені свободи для міжгрупової дисперсії ($\sigma_{між}$) та внутрішньогрупової дисперсії ($\sigma_{вн}$);

б) знайти критичне значення *F* для міжгрупової дисперсії в таблиці критичних значень F-розподілу за рівнем значущості α та ступенями свободи $df_м$ та $df_в$ (табл. 13).

Зауваження: Якщо точних даних з табл. 13 немає, ви можете скористатися наближеними значеннями або скористатися різними онлайн-калькуляторами або статистичними пакетами, що надають цю інформацію автоматично.

7. Порівняти значення F_{stat} та $F_{кр.}$. Якщо F_{stat} більше $F_{кр.}$, то можна зробити висновок, що є статистично значущі відмінності між середніми значеннями груп.

8. Для подальшого встановлення різниць між середніми значеннями груп застосовують тест Дункана, в якому обчислюються різниці між середніми значеннями груп та їх стандартні помилки. Розрахунок значення статистики тесту Дункана (tD):

$$tD = (M_1 – M_2) / SD, \quad \textbf{(62)}$$

де M_1 і M_2 – середні значення двох груп, які порівнюються;

а *SD* – стандартна помилка розрахована, як квадратний корінь з [(σ$_{вн.}$ / кількість значень у групі 1) + (σ$_{вн.}$ / кількість значень у групі 2)].

6. Порівнюємо *tD* з критичним значенням ($tD_{крит.}$) *t*-розподілу з рівнем значущості α та ступенями свободи, обчисленими на основі внутрішньогрупової дисперсії σ$_{вн.}$.

Якщо значення *tD* більше критичного значення, то робиться висновок про статистичну значущість різниці між середніми значеннями груп.

Тест Дункана має високу потужність, а це означає, що він здатний виявляти навіть невеликі статистичні різниці між групами. Крім того, він не потребує збільшення рівня довіри для порівняння більшої кількості груп, як це може відбуватися у інших тестах.

Однак, тест Дункана також має свої обмеження. Наприклад, він може бути менш точним, якщо розмір груп неоднаковий, або якщо в групах є висока мінливість. Також він може бути менш ефективним, якщо кількість груп дуже велика, бо порівняння всіх можливих пар груп може зайняти багато часу та зусиль.

Код Python для виконання порівнянь між групами за критерієм Дункана:

```
import numpy as np
from scipy.stats import f, t

# data for several groups
groups = [
np.array([51.6, 48.2, 69.4, 104, 92.0, 87.9]),
np.array([68.5, 78.5, 78.2, 74.5, 76.7, 74.1]),
np.array([55.8, 41.4, 56.2, 65.8, 42.0, 60.0]),
np.array([41.4, 43.7, 37.9, 42.4, 27.3, 46.4]),
]

# number of groups and number of observations in each
group
k = len(groups)
n = len(groups[0])

# overall average for all groups
GM = np.mean(np.concatenate(groups))

# sum of squares of deviations of observations from the
mean value for all groups
SS_t = np.sum([(group - GM)**2 for group in groups])
```

```
# internal sum of squares of deviations of observations
from the mean in each group
SS_w = np.sum([(group - np.mean(group))**2 for group in
groups])

# total number of degrees of freedom and internal number
of degrees of freedom
df_t = k * n - 1
df_w = k * (n - 1)

# estimation of variance between groups and estimation
of variance within groups
MS_b = SS_t / df_t
MS_w = SS_w / df_w

# statistical significance F
F_stat = MS_b / MS_w

# critical importance of statistics F
alpha = 0.05
F_crit = f.ppf(1 - alpha, k - 1, df_w)

# Calculate differences between group means and their
standard errors
group_means = [np.mean(group) for group in groups]
SE_means = np.sqrt(MS_w / n)

# comparison of group averages
for i in range(k):
    for j in range(i + 1, k):
        diff = np.abs(group_means[i] - group_means[j])
        t_stat = diff / (SE_means * np.sqrt(2))
        df_diff = df_w
        t_crit = t.ppf(1 - alpha / 2, df_diff)

        if t_stat > t_crit:
            print(f"Difference between group averages
{i + 1} та {j + 1} is statistically significant with
p<0.05.")
        else:
            print(f"Difference between group averages
{i + 1} та {j + 1} is not statistically significant.")
```

Результат виконання тесту:

```
Різниця між середніми значеннями груп 1 та 2 не є
статистично значущою.
```

```
Різниця між середніми значеннями груп 1 та 3 є
статистично значущою з p<0.05.
Різниця між середніми значеннями груп 1 та 4 є
статистично значущою з p<0.05.
Різниця між середніми значеннями груп 2 та 3 є
статистично значущою з p<0.05.
Різниця між середніми значеннями груп 2 та 4 є
статистично значущою з p<0.05.
Різниця між середніми значеннями груп 3 та 4 не є
статистично значущою.
```

Код R для виконання порівнянь між групами за критерієм Дункана:

```
group1 <- c(51.6, 48.2, 69.4, 104, 92.0, 87.9)
group2 <- c(68.5, 78.5, 78.2, 74.5, 76.7, 74.1)
group3 <- c(55.8, 41.4, 56.2, 65.8, 42.0, 60.0)
group4 <- c(41.4, 43.7, 37.9, 42.4, 27.3, 46.4)

install.packages("agricolae")
library(agricolae)

# Merge data groups into a single vector
y <- c(group1, group2, group3, group4)

# A vector indicating which group each value belongs to
trt <- rep(1:4, each=6)

# Calculate the internal variance of errors and the
number of degrees of freedom
MSerror <- var(y)
DFerror <- length(y) - length(unique(trt)) - 1

# Run Duncans test
duncan.test(y, trt, MSerror=MSerror, DFerror=DFerror,
console=TRUE)
```

Результат виконання тесту наступний:

```
Duncan's new multiple range test
for y

Mean Square Error:  384.6178

trt,  means

         y       std r       se  Min   Max    Q25   Q50    Q75
1 75.51667 22.768260 6 8.006433 48.2 104.0 56.050 78.65 90.975
2 75.08333  3.705356 6 8.006433 68.5  78.5 74.200 75.60 77.825
3 53.53333  9.846556 6 8.006433 41.4  65.8 45.450 56.00 59.050
4 39.85000  6.751815 6 8.006433 27.3  46.4 38.775 41.90 43.375

Alpha: 0.05 ; DF Error: 19

Critical Range
       2        3        4
23.69890 24.87093 25.61328

Means with the same letter are not significantly
different.

         y groups
1 75.51667      a
2 75.08333      a
3 53.53333     ab
4 39.85000      b
```

В результатах тесту вказані як проміжні обчислення відмінностей між групами за тестом Дункана, так і наведені самі відмінності між ними. Відмінності позначаються буквами «a”, «b” та іншими. В нашому випадку Група 1 відрізняється від Груп 3 і 4, а Група 2 – також від груп 3 і 4 із значущістю 0,05.

6.3.4. Критерій Даннета: порівняння декількох груп з контрольною

Для порівняння груп з контрольною можна використати як критерій Ньюмена-Коулса, так і критерій Тюкі, порівнюючи лише дослідні групи з контрольною. Проте за наявності контрольної групи користуються спеціальним критерієм – *критерієм Даннета (Dunnett's test).*

Критерій Даннета можна обчислити наступним чином:

1. Спочатку середні значення для всіх груп впорядковують по абсолютній величині їх відмінності від контрольної групи.

2. Контрольну групу порівнюють з іншими, починаючи з тієї групи, яка найбільш відрізняється від неї.

3. Статистику критерію Даннета q' обчислюють за формулою:

$$q' = \frac{\left|\bar{x}_к - \bar{x}_i\right|}{\sqrt{s_в^2\left(\frac{1}{n_к} + \frac{1}{n_i}\right)}}, \qquad \textbf{(63)}$$

де $\bar{x}_к$ і $\bar{x}_i$ - середнє значення для контрольної та дослідної груп, відповідно;

$s_в^2$ - внутрішньогрупова дисперсія;

$n_к$ і n_i - об’єм контрольної та дослідної груп, відповідно.

4. Для того, щоб знайти критичне значення критерію Даннета $q_{кр}'$ (табл. 15), потрібно взяти до уваги величину *m*, яка є сталою і дорівнює числу груп, включаючи контрольну, ступінь свободи $df = N - m$ (N – сума чисельності всіх груп, m – число груп) і величини *l*, яка називається інтервалом порівняння (див. детальніше в пункті 6.3.1).

Якщо $q' > q_{кр}'$, то говорять про істотну відмінність між дослідною та контрольною групами при рівні статистичної значущості $p<0,05$.

Таблиця 15. Критичні значення $q_{кр}'$ при рівні статистичної значущості $p<0,05$

df	**Інтервал порівняння *l***								
	2	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
6	2,45	2,86	3,10	3,26	3,39	3,49	3,57	3,64	3,71
7	2,36	2,75	2,97	3,12	3,24	3,33	3,41	3,47	3,53
8	2,31	2,67	2,88	3,02	3,13	3,22	3,29	3,35	3,41
9	2,26	2,61	2,81	2,95	3,05	3,14	3,20	3,26	3,32
10	2,23	2,57	2,76	2,89	2,99	3,07	3,14	3,19	3,24
11	2,20	2,53	2,72	2,84	2,94	3,02	3,08	3,14	3,19
12	2,18	2,50	2,68	2,81	2,90	2,98	3,04	3,09	3,14
13	2,16	2,48	2,65	2,78	2,87	2,94	3,00	3,06	3,10
14	2,14	2,46	2,63	2,75	2,84	2,91	2,97	3,02	3,07
15	2,13	2,44	2,61	2,73	2,82	2,89	2,95	3,00	3,04
16	2,12	2,42	2,59	2,71	2,80	2,87	2,92	2,97	3,02
17	2,11	2,41	2,58	2,69	2,78	2,85	2,90	2,95	3,00
18	2,10	2,40	2,56	2,68	2,76	2,83	2,89	2,94	2,98
19	2,09	2,39	2,55	2,66	2,75	2,81	2,87	2,92	2,96
20	2,09	2,38	2,54	2,65	2,73	2,80	2,86	2,90	2,95
24	2,06	2,35	2,51	2,61	2,70	2,76	2,81	2,86	2,90
30	2,04	2,32	2,47	2,58	2,66	2,72	2,77	2,82	2,86

Приклад 25. Порівняймо дослідні групи із контрольною у прикладі 23 за критерієм Даннета.

1. Початкові дані оформляємо у вигляді таблиці:

Показник	Контроль	10 мг/л Ni^{2+}	25 мг/л Ni^{2+}	50 мг/л Ni^{2+}
x_i	51,6 48,2 69,4 104 92,0 87,9	68,5 78,5 78,2 74,5 76,7 74,1	55,8 41,4 56,2 65,8 42,0 60,0	41,4 43,7 37,9 42,4 27,3 46,4
n	6	6	6	6
m		4		
N		24		
$\overline{x}_i$	75,5	75,1	53,5	39,9
s_i^2	518	13,7	97,0	45,6
$s_в^2$		169		

2. Порівнюємо з контрольною групою ту групу, яка найбільш від неї відрізняється (в даному випадку з групою риб, які зазнали дії йонів нікелю з концентрацією 50 мг/л) **(63)**:

$$q' = \frac{75,5 - 39,9}{\sqrt{169\left(\frac{1}{6} + \frac{1}{6}\right)}} = \frac{35,6}{7,51} = 4,74.$$

Число ступенів свободи *df* дорівнює 20, а інтервал порівняння *l=4*. За табл. 17 знаходимо критичне значення $q_{кр}'$. Воно дорівнює 2,54. Оскільки обчислене нами значення більше за критичне, то дослідна група «50 мг/л Ni^{2+}» істотно відрізняється від контрольної при рівні статистичної значущості $p<0,05$. Продовжуємо порівняння.

3. Порівнюємо з контрольною групою групу риб, підданих дії йонів нікелю з концентрацією 25 мг/л Ni^{2+}:

$$q' = \frac{75,5 - 53,5}{\sqrt{169\left(\frac{1}{6} + \frac{1}{6}\right)}} = \frac{22,0}{7,51} = 2,93.$$

Число ступенів свободи *df* дорівнює 20, а інтервал порівняння *l=3*. За табл. 20 знаходимо критичне значення $q_{кр}'$. Воно дорівнює *2,38*. Оскільки обчислене нами значення *q=2,93* більше за критичне (*2,38*), то дослідна група «25 мг/л Ni^{2+}» істотно відрізняється від контрольної при рівні статистичної значущості *p<0,05*. Продовжуємо порівняння.

4. Порівнюємо з контрольною групою групу риб, експонованих до йонів нікелю концентрацією 10 мг/л Ni^{2+}:

$$q' = \frac{75{,}5 - 75{,}1}{\sqrt{169\left(\frac{1}{6} + \frac{1}{6}\right)}} = \frac{0{,}4}{7{,}51} = 0{,}05.$$

Обчислене нами значення *q=0,05* менше за критичне (*2,09*) при *df=20* і *l=2*, тому дослідна група не відрізняється від контрольної.

Отже, критерій Даннета є «м'якшим» порівняно з критерієм Ньюмена-Коулса, про що свідчать величини його критичних значень (табл. 16 і 17), а також результати наших обчислень (приклади 23 і 24).

Код Python для перевірки відмінностей між дослідними групами і контрольною групою за тестом Даннета:

```
import numpy as np

# Initial data in the form of a table
data = {
    'C': [51.6, 48.2, 69.4, 104, 92.0, 87.9],
    'D1': [68.5, 78.5, 78.2, 74.5, 76.7, 74.1],
    'D2': [55.8, 41.4, 56.2, 65.8, 42.0, 60.0],
    'D3': [41.4, 43.7, 37.9, 42.4, 27.3, 46.4]
}

# Calculate the average values for each group
means = {group: np.mean(values) for group, values in
data.items()}

# Number of groups
m = len(data)
print('Number of groups:', m)

# Total number of observations
N = sum(len(values) for values in data.values())
print(' Total amount of data in groups:', N)

# Find the overall average
mean_total = round(sum(len(data[group]) *
(means[group]) for group in data)/ N, 2)
print(' Average value of the experimental groups:',
mean_total)

# Find the number of degrees of freedom for the between-
group variance
df_intra = m - 1

# Find the between-group variance (SSB)
SSB = round(sum(len(data[group]) * (means[group] -
mean_total)**2 for group in data) / df_intra, 2)
print(' Between-group variance:', SSB)

def calculate_mean(data_dict):
    means = {}
    for group, values in data_dict.items():
        mean = round(sum(values) / len(values), 2)
        means[group] = mean
```

```
    return means

means = calculate_mean(data)
print('The average values of the groups', means)

# Determination of group variances
def calculate_variance(data_dict):
    variances = {}
    for group, values in data_dict.items():
        mean = means.get(group, 0)  # Use get to get
the value from the dictionary or 0 if the key does not
exist.
        variance = sum((x - mean) ** 2 for x in values)
/ (len(values) - 1)
        variances[group] = variance
    return variances

variances = calculate_variance(data)

# Find the within-group variance (SSW)
SSW = round(sum(variances[group] * len(data[group]) for
group in data) / N, 2)
print(' Within-group variance:', SSW)

# Find the number of degrees of freedom for the within-
group variance
df_w = N - m

def find_critical_q(df_w, m):
    critical_values = {
        2: [2.45, 2.86, 3.10, 3.26, 3.39, 3.49, 3.57,
3.64, 3.71],
        3: [2.36, 2.75, 2.97, 3.12, 3.24, 3.33, 3.41,
3.47, 3.53],
        4: [2.31, 2.67, 2.88, 3.02, 3.13, 3.22, 3.29,
3.35, 3.41],
        5: [2.26, 2.61, 2.81, 2.95, 3.05, 3.14, 3.20,
3.26, 3.32],
        6: [2.23, 2.57, 2.76, 2.89, 2.99, 3.07, 3.14,
3.19, 3.24],
        7: [2.20, 2.53, 2.72, 2.84, 2.94, 3.02, 3.08,
3.14, 3.19],
        8: [2.18, 2.50, 2.68, 2.81, 2.90, 2.98, 3.04,
3.09, 3.14],
        9: [2.16, 2.48, 2.65, 2.78, 2.87, 2.94, 3.00,
3.06, 3.10],
```

```
        10: [2.14, 2.46, 2.63, 2.75, 2.84, 2.91, 2.97,
3.02, 3.07],
        15: [2.13, 2.44, 2.61, 2.73, 2.82, 2.89, 2.95,
3.00, 3.04],
        20: [2.09, 2.38, 2.54, 2.65, 2.73, 2.80, 2.86,
2.90, 2.95],
        24: [2.06, 2.35, 2.51, 2.61, 2.70, 2.76, 2.81,
2.86, 2.90],
        30: [2.04, 2.32, 2.47, 2.58, 2.66, 2.72, 2.77,
2.82, 2.86]
    }
    return critical_values[df_w][m]

# Find the critical value of the Dunnett test qcr'
critical_q = find_critical_q(df_w, m - 2)
print("The critical value of the Dunnett criterion
qcr':", critical_q)

# Comparing the experimental and control groups
for group, values in data.items():
    if group == 'C':
        continue
    q = round(abs(means[group] - means['C']) /
(np.sqrt((SSW * (1/len(data['C']) +
1/len(data['C'])))), 2)
    print(f" Dunnett's criterion value for the group
'{group}':", q)
    if q > critical_q:
        print(f" Group '{group}' differs from the
control at the level of statistical significance
p<0.05.")
    else:
        print(f" Group '{group}' does not differ from
the control.")
```

Результатом виконання цього коду за вказаним прикладом є:

```
Кількість груп: 4
 Загальна кількість даних у групах: 24
 Середнє значення в експериментальних групах: 61.0
 Міжгрупова дисперсія: 1824.29
 Середні значення по групах {'C': 75.52, 'D1': 75.08,
'D2': 53.53, 'D3': 39.85}
 Within-group variance: 168.67
Критичне значення критерію Даннета qкр': 2.54
```

```
 Значення критерію Даннета для групи 'D1': 0.06
 Група "D1" не відрізняється від контролю.
 Значення критерію Даннета для групи 'D2': 2.93
 Група "D2" відрізняється від контролю на рівні статистичної значущості p<0.05.
 Dunnett's criterion value for the group 'D3': 4.76
Група "D3" відрізняється від контролю на рівні статистичної значущості p<0.05.
```

Код R для перевірки відмінностей між дослідними групами і контрольною групою за тестом Даннета:

```
library(multcomp)

perform_dunnett_test <- function() {
  n_groups <- as.integer(readline("Enter the number of groups (N): "))
  group_names <- character(n_groups)
  group_data <- data.frame()

  for (i in 1:n_groups) {
    group <- readline(paste("Enter the data for the group ", i, " (through coma). First group - control: ", sep = ""))
    group <- as.numeric(strsplit(group, ",")[[1]])  # Updated this line to extract the first element
    group <- na.omit(group)  # Remove NA values if any

    group_names[i] <- readline(paste("Enter the name of the group ", i, ": ", sep = ""))

    if (length(group_data$Value) > 0) {
      group_data <- rbind(group_data, data.frame(Group = rep(group_names[i], length(group)), Value = group))
    } else {
      group_data <- data.frame(Group = rep(group_names[i], length(group)), Value = group)
    }
  }

  # Converting the "Group" variable to a factor
```

```
  group_data$Group <- factor(group_data$Group)

  # Set the linear model
  lm_model <- lm(Value ~ Group, data = group_data)

  # Perform a Dunnett test comparing the first group
with all other groups
  dunnett_result <- glht(lm_model, linfct = mcp(Group =
"Dunnett"))

  print(summary(dunnett_result))
}

perform_dunnett_test()
```

Результат виконання цього коду за вказаним прикладом:

```
     Simultaneous Tests for General Linear
Hypotheses

Multiple Comparisons of Means: Dunnett Contrasts

Fit: lm(formula = Value ~ Group, data = group_data)

Linear Hypotheses:
            Estimate Std. Error t value Pr(>|t|)
d1 - c == 0  -0.4333     7.4981  -0.058   0.9999
d2 - c == 0 -21.9833     7.4981  -2.932   0.0217 *
d3 - c == 0 -35.6667     7.4981  -4.757   <0.001 ***
---
Signif. Codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’
   0.1 ‘ ’ 1
(Adjusted p values reported – single-step method)
```

Отже, котрольна група відрізняється від груп d2 і d3, відповідно, з $p<0,05$ і $p<0,001$.

6.3.5. Непараметричний критерій Данна для порівняння декількох груп між собою

Як аналог до вищевказаних параметричних критеріїв (Ньюмена-Коулса, Тюкі та Даннета) можна використати непараметричний **критерій Данна (Dunn's test)**. Чому ми зупинились саме на цьому критерії? Перш за все це пов'язано із тим, що його можна використовувати для порівняння вибірок як рівного, так і різного об'єму, а також порівнювати контрольну групу із дослідними, що досить зручно.

Критерій Данна визначається за формулою:

$$Q = \frac{\left|\overline{R}_A - \overline{R}_B\right|}{\sqrt{\frac{N(N+1)}{12}\left(\frac{1}{n_A} + \frac{1}{n_B}\right)}}, \qquad \textbf{(64)}$$

де $\overline{R}_A$ і $\overline{R}_B$ – середні ранги двох вибірок А і В, що порівнюються ($\overline{R}_{A,B} = \sum R_{A,B}/n_{A,B}$). Для обчислення середнього рангу значення порівнюваних вибірок об'єднуються в один ряд і розміщуються від меншого до більшого в порядку зростання; найменше значення буде мати ранг 1. Потім отримані ранги вже сумуються окремо для кожної вибірки і діляться на кількість значень у вибірці.

N_A і n_B – об'єми вибірок, що порівнюються;

N – загальний об'єм всіх вибірок, що порівнюються.

Критичні значення $Q_{кр}$ наведені в табл. 16. Порівняння вибірок проводиться за таким самим алгоритмом, як це робиться при використанні критерію Ньюмена-Коулса. Якщо $Q > Q_{кр}$, то групи даних істотно відрізняються між собою при вибраному рівні статистичної значущості p.

Таблиця 16. Критичні значення $Q_{кр}$ для попарного порівняння груп

Кількість вибірок, які порівнюються *k*	Рівень статистичної значущості *p*	
	0,05	**0,01**
2	1,960	2,576
3	2,394	2,936
4	2,639	3,144
5	2,807	3,291
6	2,936	3,403
7	3,038	3,494
8	3,124	3,570
9	3,197	3,635
10	3,261	3,692
11	3,317	3,743
12	3,368	3,789
13	3,414	3,830
14	3,456	3,868
15	**3,**494	3,902

При порівнянні дослідної групи із контрольною формула для Q залишається незмінною, тільки критичне значення $Q_{кр}$ потрібно взяти із табл. 17.

Таблиця 17. Критичні значення $Q_{кр}$ для порівняння з контрольною групою

Кількість вибірок, які порівнюються *k*	Рівень статистичної значущості *p*	
	0,05	**0,01**
2	1,960	2,576
3	2,242	2,807
4	2,394	2,936
5	2,498	3,024
6	2,576	3,091
7	2,639	3,144
8	2,690	3,189
9	2,735	3,227
10	2,773	3,261
11	2,807	3,291
12	2,838	3,317
13	2,866	3,342
14	2,891	3,364
15	2,914	3,384

Приклад 26. Порівняємо дані з прикладу 23 за критерієм Данна.

Початкові дані оформлюємо у вигляді таблиці:

Показник	Контроль		10 мг/л Ni^{2+}		25 мг/л Ni^{2+}		50 мг/л Ni^{2+}	
	$A_{сод}$, Од/мг білка	Ранг	$A_{сод}$, Од/мг білка	Ранг	$A_{сод}$, Од/мг білка	Ранг	$A_{сод}$, Од/мг білка	Ранг
x_i (R_i)	51,6	10	68,5	15	55,8	11	41,4	3,5
	48,2	9	78,5	21	41,4	3,5	43,7	7
	69,4	16	78,2	20	56,2	12	37,9	2
	104	24	74,5	18	65,8	14	42,4	6
	92,0	23	76,7	19	42,0	5	27,3	1
	87,9	22	74,1	17	60,0	13	46,4	8
$\sum R$	104		110		58,5		27,5	
$\overline{R} = \sum R / n$	17,3		18,3		9,75		4,58	
n	6		6		6		6	
N	24							

2.1. За формулою **(64)**, порівнюючи контрольну групу із тією групою, яка найбільше від неї відрізняється (в даному випадку з групою риб, які перебували у воді з концентрацією йонів нікелю 50 мг/л Ni^{2+}), обчислюємо Q:

$$Q = \frac{17{,}3 - 4{,}58}{\sqrt{\frac{24(24+1)}{12}\left(\frac{1}{6}+\frac{1}{6}\right)}} = 3{,}116.$$

2.2. При рівні статистичної значущості $p<0{,}05$ та кількості вибірок $k=4$, $Q_{кр}=2{,}394$ (табл. 17).

2.3. Умова $Q>Q_{кр}$ виконується тому, що 3,116>2,394. Тому групи даних: «Контроль» і «50 мг/л Ni^{2+}» істотно між собою відрізняються при $p<0{,}05$.

3.1. Порівнюємо між собою групи даних: «Контроль» і «25 мг/л Ni^{2+}»:

$$Q = \frac{17{,}3 - 9{,}75}{\sqrt{\frac{24(24+1)}{12}\left(\frac{1}{6}+\frac{1}{6}\right)}} = 1{,}850.$$

3.2. Умова $Q>Q_{кр}$ не виконується, оскільки 1,850<2,394. Тому групи даних: «Контроль» і «25 мг/л Ni^{2+}» між собою не відрізняються, а, отже, не відрізняються між собою також і групи даних: «Контроль» та «10 мг/л Ni^{2+}».

4.1. Порівнюємо між собою групи даних: «10 мг/л Ni^{2+}» і «50 мг/л Ni^{2+}»:

$$Q=\frac{18,3-4,58}{\sqrt{\frac{24(24+1)}{12}\left(\frac{1}{6}+\frac{1}{6}\right)}}=3,361.$$

4.2. Умова $Q>Q_{кр}$ виконується, бо 3,361>2,394. Тому групи даних: «10 мг/л Ni^{2+}» і «50 мг/л Ni^{2+}» істотно між собою відрізняються при $p<0,05$.

5.1. Порівнюємо між собою групи даних: «10 мг/л Ni^{2+}» і «25 мг/л Ni^{2+}»:

$$Q=\frac{18,3-9,75}{\sqrt{\frac{24(24+1)}{12}\left(\frac{1}{6}+\frac{1}{6}\right)}}=2,095.$$

5.2. Умова $Q>Q_{кр}$ не виконується, бо 2,095<2,394. Тому групи даних: «10 мг/л Ni^{2+}» і «25 мг/л Ni^{2+}» між собою статистично не відрізняються.

6.1. Порівнюємо між собою групи даних: «25 мг/л Ni^{2+}» і «50 мг/л Ni^{2+}»:

$$Q=\frac{9,75-4,58}{\sqrt{\frac{24(24+1)}{12}\left(\frac{1}{6}+\frac{1}{6}\right)}}=1,266.$$

6.2. Умова $Q>Q_{кр}$ не виконується, оскільки 1,266<2,394. Тому групи даних: «25 мг/л Ni^{2+}» і «50 мг/л Ni^{2+}» між собою статистично не відрізняються.

Код Python для перевірки відмінностей між дослідними групами за непараметричним тестом Данна:

```
import numpy as np
import pandas as pd
from scipy import stats
import scikit_posthocs as sp

# User data entry
group1 = pd.Series(input("Enter the data for group 1,
separated by spaces: ").split())
group2 = pd.Series(input("Enter the data for group 2,
separated by spaces: ").split())
group3 = pd.Series(input("Enter the data for group 3,
separated by spaces: ").split())
group4 = pd.Series(input("Enter the data for group 4,
separated by spaces: ").split())

# Converting data into numerical format
group1 = group1.astype(float)
group2 = group2.astype(float)
group3 = group3.astype(float)
group4 = group4.astype(float)

# Conducting the Dunn criterion
data = np.array([group1, group2, group3, group4])
dunn_results = sp.posthoc_dunn(data, p_adjust='holm')

# Displaying the results
print("Results of the Dunn's test:", dunn_results)
```

Результат виконання коду:

```
Введіть дані для групи 1, розділені пробілами: >? 51.6
48.2 69.4 104 92.0 87.9
Введіть дані для групи 2, розділені пробілами: >? 68.5
78.5 78.2 74.5 76.7 74.1
Введіть дані для групи 3, розділені пробілами: >? 55.8
41.4 56.2 65.8 42.0 60.0
Введіть дані для групи 4, розділені пробілами: >? 41.4
43.7 37.9 42.4 27.3 46.4
Результати критерію Данна:
          1         2         3         4
```

```
   20  1.000000   0.806455   0.189535   0.008927
2   0.806455   1.000000   0.141885   0.004530
3   0.189535   0.141885   1.000000   0.411137
4   0.008927   0.004530   0.411137   1.000000
```

Після введення даних користувачем код проводить критерій Данна для порівняння між групами. Результати представлені у вигляді матриці, де кожен елемент вказує на значущість різниці між двома групами. Значення менші за 0,05 вказують на відмінності між дослідними групами.

Висновки. Результати обчислень за непараметричним критерієм Данна (приклад 25) повністю відтворили результати обчислень за параметричним критерієм Стьюдента-Ньюмена-Коулса (приклад 23). Тому ми також рекомендуємо використання непараметричного критерію Данна для порівняння груп даних з малим числом повторів ($n<15$) поряд з вище розглянутими параметричними критеріями.

РОЗДІЛ 7. ВЗАЄМОЗВ'ЯЗКИ МІЖ ГРУПАМИ: КОРЕЛЯЦІЙНО-РЕГРЕСІЙНИЙ АНАЛІЗ

7.1. Кореляційний аналіз

Кореляційний аналіз – метод дослідження взаємозалежності ознак у генеральній сукупності, які є випадковими величинами з нормальним характером розподілу. Основними вимогами до застосування кореляційного аналізу є достатня кількість спостережень, сукупності факторних і результативних показників, а також їх кількісне вимірювання і відображення в інформаційних джерелах. Застосування кореляційного аналізу тісно пов'язане з регресійним аналізом, тому його часто називають кореляційно-регресійним. Головні завдання кореляційного аналізу:

– визначення наявності та форми зв'язку;

– вимірювання щільності (сили) зв'язку;

– виявлення впливу факторів на результативну ознаку.

Напрямок та сила взаємозв'язку між числовими ознаками відображаються у математичному показнику, який називається коефіцієнтом кореляції *r*. Для оцінки ступеня лінійності зв'язку між двома кількісними ознаками застосовують коефіцієнт кореляції Пірсона.

Проте в біохімічних та інших біологічних дослідженнях найчастіше використовують саме коефіцієнт кореляції Пірсона, який ми опишемо детальніше.

В табл. 18 та на рис. 6 показано умовну відповідність між величиною коефіцієнту кореляції і ступенем лінійності зв'язку (так

звана «шкала Чеддока», складена американським математиком Робертом Е. Чеддоком). Як бачимо тільки при *r=1* зв'язок між досліджуваними ознаками є функціональним, який описується лінійним рівнянням (див. рис. 6). Шкала Чеддока і рис. 6 вказують на ступінь лінійності зв'язку між ознаками, які порівнюються.

Таблиця 18. Величина коефіцієнту кореляції і ступінь лінійності зв'язку за «шкалою Чеддока»

Коефіцієнт кореляції	Ступінь лінійності зв'язку	Англійський відповідник
1,00	Зв'язок функціональний	Functional relationship
0,90 – 0,99	Дуже сильний	Very strong
0,70 – 0,89	Сильний	Strong
0,50 – 0,69	Значний	Significant
0,30 – 0,49	Помірний	Moderate
0,10 – 0,29	Слабкий	Weak
0,00	Зв'язок відсутній	Relationship is absent

У біології, а особливо в медицині кореляційний взаємозв'язок вважають наявним, коли r ≥ 0,7 (тобто «сильна» і «дуже сильна» кореляція за «шкалою Чеддока»). Якщо коефіцієнт кореляції дорівнює нулю, то можна говорити, що величини незалежні за умови нормального розподілу.

Кореляційні зв'язки можна вивчати на якісному рівні з діаграм розсіяння емпіричних значень змінних X і Y (рис. 6) і відповідним чином їх інтерпретувати. Так, наприклад, якщо підвищення рівня однієї змінної супроводжується підвищенням рівня іншої, то йдеться про додатну кореляцію або прямий зв'язок ($0 < r \leq 1$, рис. 7а, б). Якщо ж зростання однієї змінної супроводжується зниженням значень іншої, то маємо справу з від'ємною кореляцією або

зворотним зв'язком ($-1 \le r < 0$ рис. 7г, г). Якщо $r = 0$, то говорять про відсутність кореляційних взаємозв'язків між ознаками.

Коефіцієнт кореляції r для вибірки є точковою оцінкою генерального коефіцієнту кореляції – параметра ρ. Як для будь-якої випадкової величини, значення r може змінюватись при повторних дослідженнях вибірок, взятих з тієї самої генеральної сукупності. Тому коефіцієнт кореляції для вибірки має статистичну помилку, яку можна обчислити за формулою:

$$s_r = \sqrt{(1 - r^2)/(n - 2)}, \quad \textbf{(65)}$$

де s_r – статистична помилка вибіркового коефіцієнту кореляції; r – вибірковий коефіцієнт кореляції; n – число вивчених об'єктів, в яких виміряні дві ознаки.

Довірчий інтервал для коефіцієнту кореляції двох ознак, які нормально розподілені, буде охоплювати:

$$r - ts_r \le \rho \le r + ts_r, \quad \textbf{(66)}$$

де $r - ts_r$ – нижня, $r + ts_r$ – верхня межа довірчого інтервалу; t – табличне значення критерію Стьюдента, який залежить від прийнятого рівня довіри та ступенів свободи ($df = n - 2$).

При аналізі зв'язку перевіряється нульова гіпотеза: в генеральній сукупності зв'язок між ознаками відсутній (H_0: $\rho = 0$). При нормальному розподілі обох ознак, для перевірки нульової гіпотези використовують критерій Стьюдента t: $t = r/s_r$, де r – коефіцієнт кореляції, s_r – стандартна помилка коефіцієнта кореляції.

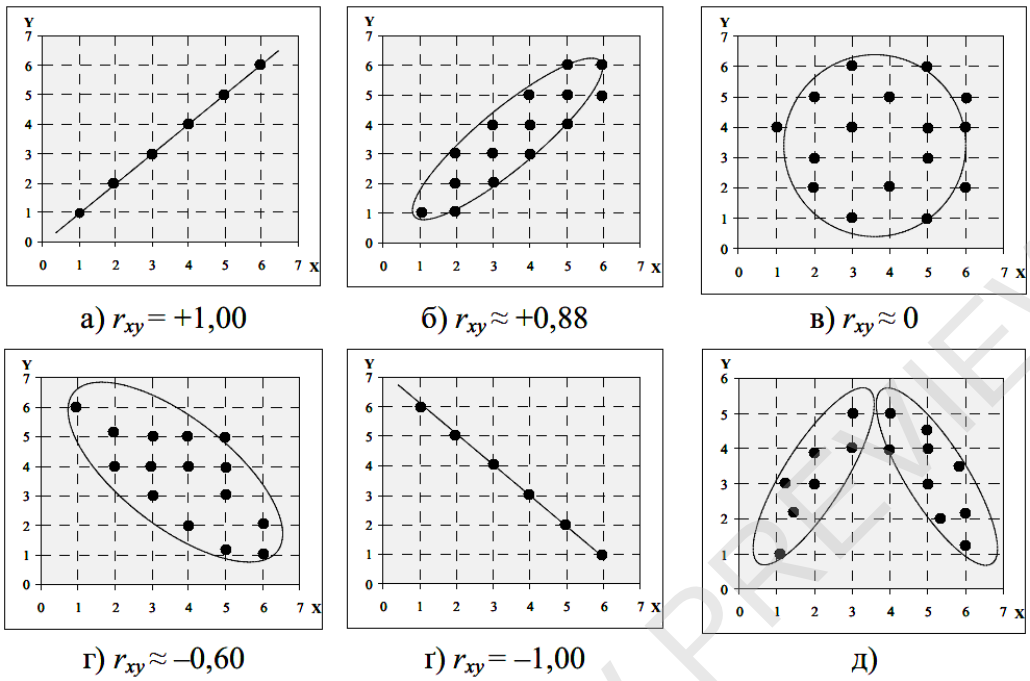


Рис 7. Діаграми розсіяння емпіричних значень змінних X і У:

а) зв'язок описується рівнянням $y=ax+b$, де a – тангенс кута нахилу лінії регресії до вісі абсцис ($a>0$), а b – ордината у точці перетину лінії регресії з віссю ординат; б) сильна позитивна кореляція; в) нульова кореляція (кореляція відсутня); г) помірна негативна кореляція; ґ) зв'язок описується рівнянням $y=-ax+b$ $(a>0)$; д) нелінійна кореляція.

Основними етапами кореляційного аналізу є наступні:

1. Оформляємо отримані нами дані за двома ознаками у вигляді таблиці:

	x_i	y_i	$x_i - \bar{x}$	$y_i - \bar{y}$	$(x_i - \bar{x})(y_i - \bar{y})$	$(x_i - \bar{x})^2$	$(y_i - \bar{y})^2$
Сума	$\sum$	$\sum$			$\sum$	$\sum$	$\sum$
Середні значення	$\bar{x}$	$\bar{y}$					

2. Підставимо отримані дані з вказаної вище таблиці в наступну формулу:

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}} \quad \textbf{(67)}$$

3. Отримавши коефіцієнт кореляції *r*, перевіряємо його статистичну значущість за допомогою *t* критерію Стьюдента. При цьому висувається гіпотеза про те, що *r=0*:

$$t_r = \frac{|r|}{\sqrt{1-r^2}} \sqrt{n-2} \quad \textbf{(68)}$$

Якщо обчислене значення $t_r > t_{кр}$ (табл. 1) при рівні статистичної значущості $p<0,05$ і числі ступенів свободи $df=n-2$, то гіпотеза про відсутність зв'язків між ознаками ($r=0$) відкидається.

4. Беручи до уваги результати обчислень із таблиці, враховуючи знак коефіцієнта кореляції та тісноту зв'язку за табл. 20, можна говорити про взаємозв'язки між досліджуваними ознаками.

5. Для лінійної залежності коефіцієнт детермінації рівний квадрату коефіцієнту кореляції:

$$R^2 = r^2 \times 100\% \text{ ,} \quad \textbf{(69)}$$

де *r* – коефіцієнт кореляції.

Цей показник вказує наскільки зміни величини *X* приводять до змін величини *Y*. Так, при *r* = 0,9 близько 81% ($0,9^2 \times 100\% \approx 81\%$) змін однієї ознаки визначається змінами іншої, в 19% випадків збіг чи незбіг варіацій двох ознак є чисто випадковими.

Приклад 27. За наведеними даними потрібно вказати на наявність чи відсутність взаємозв'язків між активністю

лактатдегідрогенази (X_i) та вмістом лактату (Y_i) в плазмі крові карася сріблястого:

***Xᵢ*:** 6,28; 6,89; 7,34; 7,92; 8,26; 8,74; 8,39; 8,34; 8,74; 9,72; 14,0; 15,6; 17,7; 18,5; 20,1; 22,9; 24,8; 31,3; 36,2; 39,9.

***Yᵢ*:** 5,11; 5,82; 6,96; 7,39; 7,07; 7,73; 7,81; 7,56; 8,00; 8,45; 8,77; 9,01; 9,13; 9,45; 9,77; 10,1; 10,6; 10,8; 11,3; 12,4.

Оформлюємо отримані нами дані за двома ознаками у вигляді таблиці:

X_i	Y_i	$x_i - \bar{x}$	$y_i - \bar{y}$	$(x_i - \bar{x})(y_i - \bar{y})$	$(x_i - \bar{x})^2$	$(y_i - \bar{y})^2$
6,28	5,11	-9,80	-3,55	34,79	96,04	12,60
6,89	5,82	-9,19	-2,84	26,10	84,46	8,07
7,34	6,96	-8,74	-1,70	14,86	76,39	2,89
7,92	7,39	-8,16	-1,27	10,36	66,59	1,61
8,26	7,07	-7,82	-1,59	12,43	61,15	2,53
8,74	7,73	-7,34	-0,93	6,82	53,88	0,86
8,39	7,81	-7,69	-0,85	6,54	59,14	0,72
8,34	7,56	-7,74	-1,10	8,51	59,91	1,21
8,74	8,00	-7,34	-0,66	4,84	53,88	0,44
9,72	8,45	-6,36	-0,21	1,34	40,45	0,04
14,0	8,77	-2,08	0,11	-0,23	4,33	0,01
15,6	9,01	-0,48	0,35	-0,17	0,23	0,12
17,7	9,13	1,62	0,47	0,76	2,62	0,22
18,5	9,45	2,42	0,79	1,91	5,86	0,62
20,1	9,77	4,02	1,11	4,46	16,16	1,23
22,9	10,1	6,82	1,44	9,82	46,51	2,07
24,8	10,6	8,72	1,94	16,92	76,04	3,76
31,3	10,8	15,2	2,14	32,53	231,0	4,58
36,2	11,3	20,1	2,64	53,06	404,0	6,97
39,9	12,4	23,8	3,74	89,01	566,4	13,99
Сума	321,6	173,2			334,7	2005	64,54
Сер.знач.	**16,08**	**8,66**				

Обчислюємо коефіцієнт кореляції за формулою **(67)**:

$$r = \frac{334,7}{\sqrt{2005 \times 64,54}} = 0,93.$$

За формулою **(68)** перевіряємо статистичну значущість коефіцієнта кореляції *r* за допомогою *t* критерію Стьюдента:

$$t_r = \frac{0{,}93}{\sqrt{1-0{,}93^2}}\sqrt{20-2} = 10{,}73.$$

Число ступенів свободи у цьому випадку: *df=n–2=20–2=18*. Знаходимо $t_{0,05}$ по табл. 1 для визначення критерію Стьюдента. Умова $t_r > t_{0,05}$, бо 10,73>2,10. Тому між даними спостерігається дуже сильна позитивна кореляція, істотна при рівні статистичної значущості $p<0,05$.

Обчислюємо за формулою **(69)** коефіцієнт детермінації:

$$R^2 = 0{,}93^2 \times 100\% = 86\%.$$

Отже, за результатами кореляційного аналізу можна зробити висновок про те, що взаємозв'язок між активністю лактатдегідрогенази і вмістом лактату статистично значущий. Коефіцієнт детермінації свідчить, що лінійна залежність адекватно (на 86%) описує реальну залежність між величинами. Можливо можна підібрати кращу форму залежності, бо наочно за формою графіка важко визначити форму цієї залежності (див. Рис. 8).

Рис. 8. Кореляційне поле залежності активності лактатдегідрогенази і вмісту лактату в плазмі крові карася сріблястого

Тому в наступному підрозділі визначимо вид регресії, яка найадекватніше описує цю залежність.

Код Python для обчислення коефіцієнта кореляції і детермінації:

```
from scipy.stats import pearsonr

# дві групи даних
group1 = [6.28, 6.89, 7.34, 7.92, 8.26, 8.74, 8.39,
8.34, 8.74, 9.72, 14.0, 15.6, 17.7, 18.5, 20.1, 22.9,
24.8, 31.3, 36.2, 39.9]
group2 = [5.11, 5.82, 6.96, 7.39, 7.07, 7.73, 7.81,
7.56, 8.00, 8.45, 8.77, 9.01, 9.13, 9.45, 9.77, 10.1,
10.6, 10.8, 11.3, 12.4]

# обчислення коефіцієнта кореляції Пірсона
corr, p_val = pearsonr(group1, group2)

# виведення результатів
print("Коефіцієнт кореляції: ", corr)
print("Коефіцієнт детермінації: ", corr**2 * 100)
print("p-значення: ", p_val)
```

У цьому прикладі, змінні x та y містять дві групи даних, для яких ми хочемо обчислити коефіцієнт кореляції Пірсона. Функція **pearsonr()** повертає кореляційний коефіцієнт, коефіцієнт детермінації та p-значення для тесту на статистичну значущість кореляції. Результати виводяться за допомогою функції **print()**.

Результат виконання програми:

```
Коефіцієнт кореляції:  0.9301225577116229
Коефіцієнт детермінації:  86,51279723640113
p-значення:  2.922157014462498e-09
```

Код R для обчислення коефіцієнту кореляції і детермінації:

```
# Створення векторів даних
Group1 <- c(6.28, 6.89, 7.34, 7.92, 8.26, 8.74, 8.39,
8.34, 8.74, 9.72, 14.0, 15.6, 17.7, 18.5, 20.1, 22.9,
24.8, 31.3, 36.2, 39.9)
Group2 <- c(5.11, 5.82, 6.96, 7.39, 7.07, 7.73, 7.81,
7.56, 8.00, 8.45, 8.77, 9.01, 9.13, 9.45, 9.77, 10.1,
10.6, 10.8, 11.3, 12.4)

# Обчислення коефіцієнта кореляції та коефіцієнта
детермінації
correlation_test <- cor.test(Group1, Group2)
correlation_coef <- correlation_test$estimate
determination <- summary(lm(Group1 ~ Group2))$r.squared
p_value <- correlation_test$p.value

# Виведення результатів
cat(paste("Correlation coefficient: ",
round(correlation_coef, digits = 3), "\n"))
cat(paste("Coefficient of determination: ",
round(determination, digits = 3), "\n"))
cat(paste("p-value: ", p_value, "\n"))
```

Результат виконання:

```
Correlation coefficient:  0.93
Coefficient of determination:  0.865
p-value:  2.92215701446257e-09
```

7.2. Парний регресійний аналіз

Коефіцієнт кореляції вказує лишень на ступінь зв’язку у варіації двох змінних величин. Проте він не дає змогу судити про те, як кількісно змінюється одна величина при зміні іншої. Для цього існує інший метод – це регресійний аналіз.

Цей аналіз можна використовувати для виявлення взаємозв’язку між фактором, що впливає на об’єкт (X), і фактором, що змінюється (Y).

Розрізняють лінійні і нелінійні регресії.

Рівняння лінійної парної регресії наступне:

$$y = a + bx + \varepsilon \ , \qquad \textbf{(70)}$$

де y – значення фактору Y;

a – вільний член;

b – коефіцієнт регресії;

x – незалежна змінна;

ε – випадкова помилка.

При центрованості помилок ε вільний член a можна визначити за формулою:

$$a = \bar{y} - b\bar{x} \ , \qquad \textbf{(71)}$$

де $\bar{x}$ і $\bar{y}$ – середні значення фактору X і Y у вибірках з n спостережень.

Коефіцієнт регресії b обчислюють за формулами **(72)** або **(73)**:

$$b = \frac{\sum_{i=1}^{n} x_i y_i - \frac{1}{n}\sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{\sum_{i=1}^{n} x_i^{\,2} - \frac{1}{n}\left(\sum_{i=1}^{n} x_i\right)^2} ; \qquad \textbf{(72)}$$

$$b = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2} \ . \qquad \textbf{(73)}$$

Нелінійний взаємозв’язок між даними може описуватись різними функціями:

Гіперболічною: $y = a + \frac{b}{x}$ **(74)**

Показниковою (експоненційною): $y = ab^x$ **(75)**

Напівлогарифмічною: $y = a + b\ln x$ **(76)**

Логарифмічною (степеневою): $\ln y = a + b\ln x$ **(77)**

Зворотною: $y = \frac{1}{a + bx}$ **(78)**

Параболічною (Поліномна модель другого порядку):

$$y = ax^2 + bx + c \quad \textbf{(79)}$$

Кубічною (Поліномна модель третього порядку):

$$y = ax^3 + bx^2 + cx + d \quad \textbf{(80)}$$

і поліномними моделями вищих порядків. Однак можлива і складніша залежність, що не описується наведеними формулами.

Часто перед дослідниками постає ряд питань, а саме: яке рівняння регресії використати для опису своїх даних, яке з них найадекватніше описує дані з найменшими похибками і помилками тощо? Тому ми зупинимось на обчисленні коефіцієнтів регресій для різних рівнянь, порівнянні рівнянь між собою, виборі оптимального виду рівняння регресії, обчисленні похибок цих рівнянь.

Вирішення завдання побудови якісного рівняння регресії, що відповідає емпіричним даним і меті дослідження, є достатньо складним і багатоступеневим процесом. Його можна розбити на три етапи:

1) вибір формули рівняння регресії;

2) визначення параметрів вибраного рівняння;

3) аналіз якості рівняння і перевірка його адекватності емпіричним даним.

Вибір формули, зазвичай, здійснюється за графіком реальних статистичних даних у вигляді точок в декартовій системі координат *(діаграма розсіювання).* Проте нерідко виникають ситуації, коли розміщення точок приблизно відповідає декільком функціям і необхідно вибрати з них найкращу. На практиці невідомо, яка

модель вірна, і часто підбирають таку модель, яка найбільше відповідає реальним даним. Ознаками «доброї» моделі є:

1. *Простота.* Модель повинна бути максимально простою. Ця властивість визначається тим фактом, що модель не зображає дійсність ідеально, а слугує її спрощенням.

2. *Максимальна відповідність.* Рівняння тим краще, чим більшу частину діапазону залежної змінної воно може пояснити.

3. *Прогнозні якості.* Модель може бути визнана якісною, якщо отримані на її основі прогнози підтверджуються реальністю.

Для обчислення коефіцієнтів регресійних рівнянь рекомендується використовувати метод найменших квадратів (МНК), який був запропонований на початку XIX ст. Лежандром і Ґаусом. МНК полягає в тому, що теоретичні значення лінії регресії *y* повинні бути отримані таким чином, щоб сума квадратів відхилень від цих даних емпіричних величин даних була мінімальною, тобто:

$$\Sigma(y_i - y_x)^2 \rightarrow min, \quad \textbf{(81)}$$

де y_x – теоретичне значення результативної ознаки.

Якщо у вас є набір точок даних (x_1, y_1), (x_2, y_2), ..., (x_n, y_n), метод найменших квадратів намагається знайти лінію або функцію *y=f(x)*, яка мінімізує різницю між фактичними значеннями *y* і прогнозованими значеннями *f(x)*. Формально, метод найменших квадратів мінімізує суму квадратів похибок (залишків):

$$S = \sum_{i=1}^{n} (y_i - f(x_i))^2 \quad \textbf{(82)}$$

де y_1 – фактичні значення; f(xi) – прогнозовані дані на основі моделі.

Основні етапи обчислень:

1. Знаходимо коефіцієнти рівнянь регресії

1.1. Рівняння лінійної регресії

Для обчислення коефіцієнтів *a* і *b* рівняння лінійної регресії **(71)** необхідно розв'язати нормальні рівняння методу найменших квадратів:

$$\begin{cases} a\,n + b\sum x_i = \sum y_i \\ a\sum x_i + b\sum (x_i)^2 = \sum y_i x_i \end{cases}. \qquad \textbf{(83)}$$

Із цієї системи можна знайти коефіцієнти *a* і *b*:

$$a = (\Sigma y_i \, \Sigma (x_i)^2 - \Sigma y_i \; x_i \, \Sigma \; x_i)/(n \, \Sigma (x_i)^2 - (\Sigma x_i)^2); \qquad \textbf{(84)}$$

$$b = (n \, \Sigma y_i \; x_i - \Sigma x_i \, \Sigma y_i) / (n \, \Sigma (x_i)^2 - (\Sigma x_i)^2). \qquad \textbf{(85)}$$

1.2. Лінійне рівняння з логарифмуванням факторної ознаки (напівлогарифмічна регресія)

Для обчислення коефіцієнтів *a* і *b* рівняння прямої з логарифмуванням факторної ознаки **(76)** необхідно розв'язати наступну систему рівнянь:

$$\begin{cases} an + b\sum \ln x_i = \sum y_i \\ a\sum \ln x_i + b\sum (\ln x_i)^2 = \sum y_i \ln x_i \end{cases}. \qquad \textbf{(86)}$$

Із цієї системи можна знайти коефіцієнти *a* і *b*:

$$a = (\Sigma y_i \, \Sigma (\ln x_i)^2 - \Sigma y_i \ln x_i \, \Sigma \ln x_i)/(n \, \Sigma (\ln x_i)^2 - (\Sigma \ln x_i)^2), \qquad \textbf{(87)}$$

$$b = (n \, \Sigma y_i \ln x_i - \Sigma \ln x_i \, \Sigma y_i) / (n \, \Sigma (\ln x_i)^2 - (\Sigma \ln x_i)^2. \qquad \textbf{(88)}$$

1.3. Лінійне рівняння з логарифмуванням факторної і результативної ознак (логарифмічна регресія)

Для обчислення коефіцієнтів *a* і *b* рівняння прямої з логарифмуванням факторної ознаки **(77)** необхідно розв'язати наступну систему рівнянь:

$$\begin{cases} an+b\sum \ln x_i=\sum \ln y_i \\ a\sum \ln x_i+b\sum (\ln x_i)^2=\sum \ln y_i \ln x_i \end{cases}. \quad \textbf{(89)}$$

Із цієї системи можна знайти коефіцієнти *a* і *b*:

$$a = (\Sigma \ln y_i \, \Sigma(\ln x_i)^2 - \Sigma \ln y_i \ln x_i \, \Sigma \ln x_i)/(n\,\Sigma(\ln x_i)^2 - (\Sigma \ln x_i)^2), \quad \textbf{(90)}$$

$$b = (n\,\Sigma \ln y_i \ln x_i - \Sigma \ln x_i \, \Sigma \ln y_i) / (n\,\Sigma(\ln x_i)^2 - (\Sigma \ln x_i)^2 . \quad \textbf{(91)}$$

1.4. Рівняння гіперболічної регресії

Нормальні рівняння методу найменших квадратів для гіперболи **(79)** такі:

$$\begin{cases} an+b\sum (1/x_i)=\sum y_i \\ a\sum (1/x_i)+b\sum (1/x_i^{\,2})=\sum (y_i/x_i) \end{cases}. \quad \textbf{(92)}$$

Результатом розв'язування системи нормальних рівнянь є наступні рівняння:

$$a = (\Sigma y_i \, \Sigma(1/x_i)^2 - \Sigma(y_i/x_i)\,\Sigma(1/x_i))/(n\,\Sigma(1/x_i)^2 - (\Sigma(1/x_i))^2), \quad \textbf{(93)}$$

$$b = (n\,\Sigma(y_i/x_i) - \Sigma(1/x_i)\,\Sigma y_i)/(n\,\Sigma(1/x_i)^2 - (\Sigma(1/x_i))^2). \quad \textbf{(94)}$$

1.5. Рівняння показникової кривої

Для обчислення коефіцієнтів *a* і *b* рівняння **(80)** необхідно розв'язати таку систему рівнянь:

$$\begin{cases} n\ln a+\ln b\sum x_i=\sum y_i \\ \ln a\sum x_i+\ln b\sum x_i^{\,2}=\sum x_i \ln y_i \end{cases}. \quad \textbf{(95)}$$

Із цієї системи можна знайти коефіцієнти *a* і *b*:

$$ln\ a = (\Sigma\ ln\ y_i\ \Sigma\ x_i^2 - \Sigma\ x_i\ ln\ y_i\ \Sigma\ x_i) / (n\ \Sigma\ x_i^2 - (\Sigma\ x_i)^2) \quad \textbf{(96)}$$

$$ln\ b = (n\ \Sigma\ x_i\ ln\ y_i - \Sigma\ x_i\ \Sigma\ ln\ y_i) / (n\ \Sigma\ x_i^2 - (\Sigma\ x_i)^2) \quad \textbf{(97)}$$

1.5. Рівняння параболічної регресії

Для відшукання коефіцієнтів *a, b і c* рівняння **(84)** необхідно розв'язати наступну систему рівнянь:

$$\begin{cases} \sum y_i = an + b\sum x_i + c\sum x_i^2 \\ \sum y_i x_i = a\sum x_i + b\sum x_i^2 + c\sum x_i^3 \\ \sum y_i x_i^2 = a\sum x_i^2 + b\sum x_i^3 + c\sum x_i^4 \end{cases} . \quad \textbf{(98)}$$

Ці та інші види регресій подані в кодах Python і R (див. далі).

Наступним кроком є обчислення помилок видів регресій, що дає змогу вибрати найбільш підходящу модель.

Обчислюємо величини *середньої помилки апроксимації* $\bar{\varepsilon}$ (Mean Absolute Percentage Error – MAPE):

$$\bar{\varepsilon} = \frac{1}{n} \cdot \sum \frac{|y_i - y_x|}{y_i} \cdot 100\% \quad , \quad \textbf{(99)}$$

де y_i – емпіричне значення результативної ознаки (*результативною* називається *ознака*, яка змінюється під впливом факторної *ознаки*), причому $y_i > 0$;

y_x – теоретичне значення результативної ознаки.

Значення середньої помилки апроксимації не має перевищувати 10-15%.

Перевірку адекватності регресійної моделі можна провести за допомогою кореляційного аналізу. Тіснота кореляційного зв'язку між *x* і *y* визначається за допомогою *теоретичного кореляційного відношення* (*індекс кореляції*) з рівнянь **(100)** або **(101)**:

$$\eta = \sqrt{\frac{\sum (y_x - \bar{y})^2}{\sum (y_i - \bar{y})^2}} \quad \textbf{(100)}$$

$$\eta = \sqrt{1 - \frac{\sum (y_i - y_x)^2}{\sum (y_i - \bar{y})^2}} \quad \textbf{(101)}$$

Теоретичне кореляційне відношення може знаходитися в межах від 0 до 1. Чим ближче кореляційне відношення до 1, тим тісніший зв'язок між ознаками.

Коли декілька рівнянь адекватно прогнозують значення, то в такому випадку придатнішим рівнянням регресії є те, яке характеризується найбільшим фактичним значенням *F-критерію Фішера*, обчисленим за формулою:

$$F = S_y^2 / S^2_{зал} \quad , \quad \textbf{(102)}$$

де

$$S_y^2 = (\Sigma y_i^2 - ((\Sigma y_i)^2 / n)) / (n - 1) \quad \textbf{(103)}$$

$$S^2_{зал} = \Sigma (y_i - y_x)^2 / (n - 2) \quad \textbf{(104)}$$

Обчислюємо похибки і помилки, оскільки, чим менші величини похибок і помилок, тим надійніше рівняння описує досліджуваний взаємозв'язок.

Абсолютну похибку Δ (Standard Error of Regression чи Standard Error of Estimate or Root Mean Squared Error – RMSE) обчислюємо за формулою:

$$\Delta = \pm \sqrt{(1/n) \sum ((y_i - y_x)/y_x)^2} \times 100 \quad \textbf{(105)}$$

Відносну похибку δ (Relative Root Mean Squared Error – RRMSE чи Normalized RMSE – nRMSE) знаходимо за формулою:

$$\delta = \pm \sqrt{1/n \sum (y_i - y_x)^2} \quad \textbf{(106)}$$

Також обчислюємо системну o_p (Bias) і випадкову помилки o_δ (Residual Standard Deviation):

$$o_p = (1/n)\Sigma((y_i - y_x)/y_x) \times 100 \quad \textbf{(107)}$$

$$o_\delta = \pm\sqrt{(1/n)\sum\left\{((y_i - y_x)/y_x)\times 100 - o_p\right\}^2} \quad \textbf{(108)}$$

де n – кількість спостережень, y_i – і-те спостереження (досліджувані значення), y_x – теоретичне значення результативної ознаки.

Формула (**107**) розраховує систематичну похибку між досліджуваними значеннями y_i та довірчою величиною y_x. Ця похибку виражається у відсотках від довірчої величини. Цей показник дозволяє оцінити, наскільки відхилення від довірчої величини впливають на досліджувані результати. Чим менше значення o_p, тим менша відносна помилка. Якщо o_p дорівнює 0, це означає, що досліджувані значення точно збігаються з довірчою величиною.

6. На основі фактичних значень *F*-критерію Фішера, похибок робимо загальний висновок про адекватність того чи іншого рівняння регресії. Детальніше це розглянуто далі по тексту.

Приклад 27. За наведеними даними (приклад 26) потрібно встановити: форму зв’язку між активністю лактатдегідрогенази (X_i) та вмістом лактату (Y_i) в плазмі крові карася сріблястого, параметри рівняння регресії та тісноту взаємозв’язку.

Для цього прикладу ми обмежимося кількістю розрахунків за типами регресій, зупинившись на описанні тільки лінійної, напівлогорифмічної, гіперболічної та експонеційної видах регресії.

Збільшена кількість обчислень для вибору оптимального типу взаємозалежностей наведена в коді Python і R (наведено нижче).

За методом найменших квадратів знаходимо коефіцієнти a і b ймовірних типів рівнянь регресії, будуємо графіки рівнянь регресії та здійснюємо перевірку значущості рівняння регресії.

1.1. Рівняння лінійної регресії

1.1.1. Формуємо таблицю з первинних даних та обчислень допоміжних величин для обчислення коефіцієнтів *a* і *b* даного рівняння:

Алдг X_i	**Лактат** Y_i	$(x_i)^2$	$y_i\, x_i$	Y_x	$y_i - y_x$	$(y_i - y_x)^2$	y_i^2
6,28	5,11	39,438	32,091	7,027	-1,917	3,673	26,112
6,89	5,82	47,472	40,100	7,128	-1,308	1,712	33,872
7,34	6,96	53,876	51,086	7,203	-0,243	0,059	48,442
7,92	7,39	62,726	58,529	7,300	0,090	0,008	54,612
8,26	7,07	68,228	58,398	7,357	-0,287	0,082	49,985
8,74	7,73	76,388	67,560	7,437	0,293	0,086	59,753
8,39	7,81	70,392	65,526	7,379	0,431	0,186	60,996
8,34	7,56	69,556	63,050	7,370	0,190	0,036	57,154
8,74	8,00	76,388	69,920	7,437	0,563	0,317	64,000
9,72	8,45	94,478	82,134	7,600	0,850	0,722	71,403
14,0	8,77	196,000	122,78	8,314	0,456	0,208	76,913
15,6	9,01	243,360	140,56	8,581	0,429	0,184	81,180
17,7	9,13	313,290	161,60	8,931	0,199	0,039	83,357
18,5	9,45	342,250	174,83	9,065	0,385	0,148	89,303
20,1	9,77	404,010	196,38	9,332	0,438	0,192	95,453
22,9	10,1	524,410	231,29	9,799	0,301	0,091	102,01
24,8	10,6	615,040	262,88	10,116	0,484	0,235	112,36
31,3	10,8	979,690	338,04	11,200	-0,400	0,160	116,64
36,2	11,3	1310,44	409,06	12,017	-0,717	0,514	127,69
39,9	12,4	1592,01	494,76	12,634	-0,234	0,055	153,76
Σ x_i	**Σ y_i**	**$\Sigma(x_i)^2$**	**$\Sigma y_i\, x_i$**	**Σy_x**	**$\Sigma(y_i - y_x)$**	**$\Sigma(y_i - y_x)^2$**	**Σ y_i^2**
321,62	**173,23**	**7179,441**	**3120,564**	**173,228**	**0,001784**	**8,708**	**1565,0**

За допомогою вказаної вище таблиці та формул **(84)** і **(85)** отримуємо коефіцієнти регресії:

$$\boldsymbol{a} = 5{,}9791;$$

$$\boldsymbol{b} = 0{,}1668.$$

Тоді рівняння регресії матиме наступний вигляд:

$$\boldsymbol{y = 0{,}167\,x + 5{,}979}.$$

За формулою **(99)** обчислюємо величину *середньої помилки апроксимації* $\bar{\varepsilon}$:

$$\bar{\varepsilon} = \frac{1}{20} \times 1{,}379 \times 100\% = 6{,}89\%.$$

Отже, апроксимація даних наведеним вище рівнянням регресії є статистично значущою, оскільки 6,89% < 15%.

За допомогою формул **(100)** і **(69)** обчислюємо величини *індекс кореляції* та *коефіцієнт детермінації*:

$$\eta = \sqrt{1 - \frac{8{,}708}{64{,}562}} = 0{,}93;$$

$$R^2 = 0{,}93^2 = 0{,}86 \times 100 = 86\%.$$

За формулою **(102)** обчислюємо фактичне значення *F-критерію Фішера*. При цьому порівнюємо загальну дисперсію S_y^2 **(103)** із залишковою $S^2_{зал}$ **(104)**:

$$\boldsymbol{S}_{\boldsymbol{y}}^{\boldsymbol{2}} = (1565{,}0 - 1500{,}43) / 19 = 3{,}3984;$$

$$\boldsymbol{S}^{\boldsymbol{2}}_{\boldsymbol{зал}} = 8{,}708 / 18 = 0{,}4838.$$

Тоді за формулою **(102)**:

$$\boldsymbol{F} = 3{,}3984 / 0{,}4838 = \mathbf{7{,}02}.$$

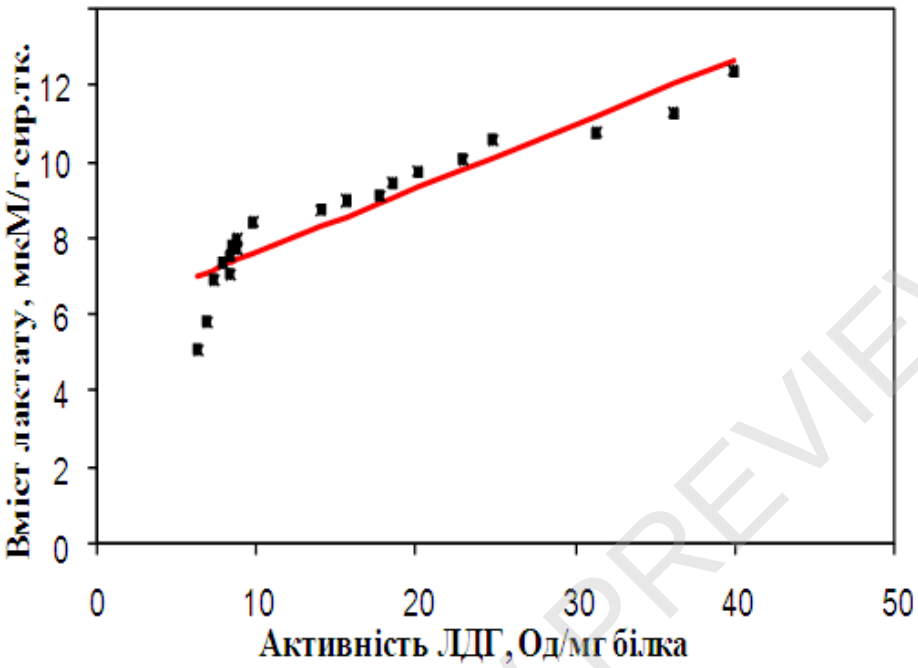


Рис. 9. Лінійна регресія

При рівні статистичної значущості $p<0,05$ $\boldsymbol{F_{кр}} = 2,22$. $\boldsymbol{F>F_{кр}}$ – лінійне **рівняння регресії адекватно описує фактичний взаємозв'язок між вмістом лактату і активністю** ЛДГ. При цьому значення $F=7,02$ вказує на те, що рівняння лінії в сім разів краще описує цей взаємозв'язок, ніж середнє значення залежної змінної.

Отже, за результатами регресійного аналізу можна зробити висновок про те, що отримане лінійне рівняння, яке за експериментальними даними має вигляд: $\boldsymbol{y = 0,167\ x + 5,979}$ *(рис. 9: пряма лінія) в 7,02 рази краще описує зміни залежної змінної (вміст лактату), ніж середнє значення аргументу.*

1.2. Лінійне рівняння з логарифмуванням факторної ознаки (напівлогарифмічне)

1.2.1. Формуємо таблицю з первинних даних та обчислень допоміжних величин для розрахунку коефіцієнтів *a* і *b* даного рівняння рівняння:

$A_{ЛДГ}$ X_i	Лактат Y_i	$\ln x_i$	$(\ln x_i)^2$	$y_i \ln x$	Y_x	$y_i - y_x$	$(y_i - y_x)^2$	y_i^2
6,28	5,11	1,837	3,376	9,389	6,382	-1,272	1,617	26,112
6,89	5,82	1,930	3,725	11,233	6,658	-0,838	0,702	33,872
7,34	6,96	1,993	3,973	13,874	6,847	0,113	0,013	48,442
7,92	7,39	2,069	4,282	15,293	7,074	0,316	0,100	54,612
8,26	7,07	2,111	4,458	14,928	7,199	-0,129	0,017	49,985
8,74	7,73	2,168	4,700	16,758	7,367	0,363	0,132	59,753
8,39	7,81	2,127	4,524	16,612	7,245	0,565	0,319	60,996
8,34	7,56	2,121	4,499	16,035	7,228	0,332	0,110	57,154
8,74	8,00	2,168	4,700	17,343	7,367	0,633	0,400	64,000
9,72	8,45	2,274	5,172	19,217	7,684	0,766	0,587	71,403
14	8,77	2,639	6,965	23,145	8,772	-0,002	0,000	76,913
15,6	9,01	2,747	7,547	24,753	9,095	-0,085	0,007	81,180
17,7	9,13	2,874	8,257	26,236	9,471	-0,341	0,117	83,357
18,5	9,45	2,918	8,513	27,573	9,603	-0,153	0,023	89,303
20,1	9,77	3,001	9,004	29,317	9,851	-0,081	0,006	95,453
22,9	10,1	3,131	9,804	31,624	10,239	-0,139	0,019	102,01
24,8	10,6	3,211	10,31	34,035	10,477	0,123	0,015	112,36
31,3	10,8	3,444	11,86	37,191	11,171	-0,371	0,138	116,64
36,2	11,3	3,589	12,88	40,556	11,605	-0,305	0,093	127,69
39,9	12,4	3,686	13,59	45,711	11,895	0,505	0,255	153,76
Σx_i	**Σy_i**	**$\Sigma \ln x_i$**	**$\Sigma(\ln x_i)^2$**	**$\Sigma y_i \ln x_i$**	**Σy_x**	**$\Sigma(y_i - y_x)$**	**$\Sigma(y_i - y_x)^2$**	**Σy_i^2**
321,62	**173,23**	**52,039**	**142,140**	**470,823**	**173,230**	**-0,000257**	**4,671**	**1565,0**

За формулами **(87) і (88)** отримаємо, відповідно:

a = (173,23 × 142140 – 470,823 × 52,039) / (20 × 142,140 – $52{,}039^2$) = **0,903,**

b = (20 × 470,823 – 52,039 × 173,23) / (20 × 142,140 – $52{,}039^2$) = **2,982**.

Рівняння регресії в даному випадку наступне:

$$y = 0{,}903 + 2{,}982 \ln x_i.$$

За формулою **(99)** обчислюємо величину *середньої помилки апроксимації* $\bar{\varepsilon}$:

$$\bar{\varepsilon} = \frac{1}{20} \times 1{,}002 \times 100\% = 5{,}01\%.$$

Отже, апроксимація даних наведеним вище рівнянням регресії є статистично значущою, оскільки 5,01%<15%.

За допомогою формул **(101)** і **(69)** обчислюємо величини *індекс кореляції* та *коефіцієнт детермінації*:

$$\eta = \sqrt{1 - \frac{4{,}671}{64{,}562}} = 0{,}96;$$

$$R^2 = 0{,}96^2 = 0{,}92 \times 100\% = 96\%.$$

За формулою **(102)** обчислюємо фактичне значення *F-критерію Фішера*. При цьому порівнюємо загальну дисперсію S_y^2 **(103)** зі залишковою дисперсією $S^2_{зал}$ **(104).**

$$\boldsymbol{S}_y^2 = (1565{,}0 - 1500{,}43) / 19 = 3{,}3984;$$

$$\boldsymbol{S}^2_{зал} = 4{,}671 / 18 = 0{,}2595.$$

Тоді

$$\boldsymbol{F} = 3{,}3984 / 0{,}2595 = \mathbf{13{,}1}.$$

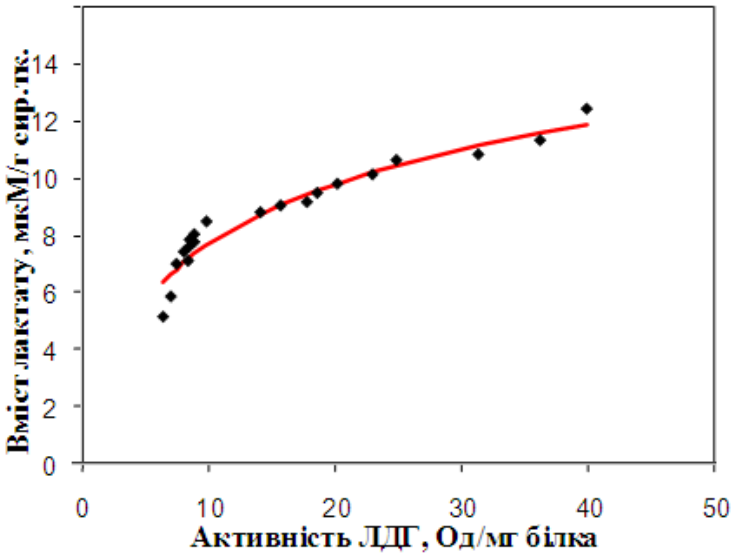


Рис. 10. Логарифмування факторної ознаки

При рівні статистичної значущості $p=0,05$ $\boldsymbol{F>F_{кр}}$=2,22. Тому рівняння регресії адекватно описує фактичні зміни вмісту лактату від змін активності ЛДГ. При цьому значення $F=13,1$ вказує на те, що напівлогарифмічне рівняння в 13 разів краще описує цей взаємозв'язок, ніж середнє значення залежної змінної.

Отже, за результатами регресійного аналізу можна зробити висновок про те, що лінійне рівняння з логарифмуванням факторної ознаки, яке за результатами експерименту має вигляд: $\boldsymbol{y = 0,903 + 2,982\ lnx}$ *(Рис. 10: крива) в 13,1 рази краще описує зміни залежної змінної (вміст лактату), ніж середнє значення аргументу.*

1.3. Рівняння гіперболічної регресії

1.3.1. Формуємо таблицю з первинних даних та обчислень допоміжних величин для обчислення коефіцієнтів *a* і *b* даного рівняння:

Акт. ЛДГ X_i	**Лактат Y_i**	$1/x_i$	$(1/x_i)^2$	y_i/x_i	Y_x	y_i-y_x	$(y_i-y_x)^2$	y_i^2
6,28	5,11	0,159	0,025	0,814	5,693	-0,583	0,340	26,112
6,89	5,82	0,145	0,021	0,845	6,266	-0,446	0,198	33,872
7,34	6,96	0,136	0,019	0,948	6,627	0,333	0,111	48,442
7,92	7,39	0,126	0,016	0,933	7,032	0,358	0,128	54,612
8,26	7,07	0,121	0,015	0,856	7,243	-0,173	0,030	49,985
8,74	7,73	0,114	0,013	0,884	7,513	0,217	0,047	59,753
8,39	7,81	0,119	0,014	0,931	7,319	0,491	0,241	60,996
8,34	7,56	0,120	0,014	0,906	7,290	0,270	0,073	57,154
8,74	8,00	0,114	0,013	0,915	7,513	0,487	0,237	64,000
9,72	8,45	0,103	0,011	0,869	7,982	0,468	0,219	71,403
14	8,77	0,071	0,005	0,626	9,259	-0,489	0,239	76,913
15,6	9,01	0,064	0,004	0,578	9,557	-0,547	0,299	81,180
17,7	9,13	0,056	0,003	0,516	9,865	-0,735	0,541	83,357
18,5	9,45	0,054	0,003	0,511	9,965	-0,515	0,265	89,303
20,1	9,77	0,050	0,002	0,486	10,139	-0,369	0,136	95,453
22,9	10,1	0,044	0,002	0,441	10,387	-0,287	0,082	102,01
24,8	10,6	0,040	0,00	0,427	10,522	0,078	0,006	112,36
31,3	10,8	0,032	0,00	0,345	10,862	-0,062	0,004	116,64
36,2	11,3	0,028	0,00	0,312	11,038	0,262	0,069	127,69
39,9	12,4	0,025	0,00	0,311	11,142	1,258	1,582	153,76

Σx_i	Σy_i	$\Sigma 1/x_i$	$\Sigma(1/x_i)^2$	$\Sigma y_i / x_i$	Σy_x	$\Sigma(y_i - y_x)$	$\Sigma(y_i - y_x)^2$	Σy_i^2
321,62	**173,23**	**1,723**	**0,185**	**13,455**	**173,22**	**0,0147**	**4,847**	**1565,0**

Знаходимо коефіцієнти a і b за формулами **(93)** і **(94)**, відповідно:

$$\boldsymbol{a} = (173{,}23 \times 0{,}185 - 13{,}455 \times 1{,}723) / (20 \times 0{,}185 - 1{,}723^2) = \mathbf{12{,}16},$$

$$\boldsymbol{b} = (20 \times 13{,}455 - 1{,}723 \times 173{,}23)/(20 \times 0{,}185\text{-}1{,}723^2) = \mathbf{-\ 40{,}61}.$$

В результаті отримаємо рівняння регресії, що має вигляд:

$$\boldsymbol{y = 12{,}16 - 40{,}61/x}.$$

За формулою **(99)** обчислюємо величину *середньої помилки апроксимації* $\bar{\varepsilon}$:

$$\bar{\varepsilon} = \frac{1}{20} \times 1{,}16 \times 100\% = 5{,}80\%.$$

Отже, апроксимація даних наведеним вище рівнянням регресії є статистично значущою, оскільки 5,80%<15%.

За допомогою формул **(101)** і **(69)** обчислюємо величини *індексу кореляції* та *коефіцієнта детермінації*:

$$\eta = \sqrt{1 - \frac{4{,}847}{64{,}562}} = 0{,}96;$$

$$R^2 = 0{,}96^{2=} 0{,}92 \times 100\% = 92\%.$$

За формулою **(102)** обчислюємо фактичне значення *F-критерію Фішера*. При цьому порівнюємо загальну дисперсію S_y^2 **(103)** із залишковою $S^2_{зал}$ **(104)**:

$$\boldsymbol{S}_y^2 = (1565{,}0 - 1500{,}43) / 19 = 3{,}3984;$$

$$\boldsymbol{S}^2_{зал} = 4{,}847 / 18 = 0{,}2693.$$

Тоді

$$\boldsymbol{F} = 3{,}3984 / 0{,}2693 = \mathbf{12{,}62}.$$

При рівні статистичної значущості $p=0,05$ $\boldsymbol{F>F_{кр}}=2,22$. Тому рівняння регресії адекватно описує фактичний взаємозв'язок між вмістом лактату і активністю ЛДГ. При цьому значення $F=14,0$ вказує на те, що рівняння гіперболи в 14 разів краще описує цей взаємозв'язок, ніж середнє значення залежної змінної.

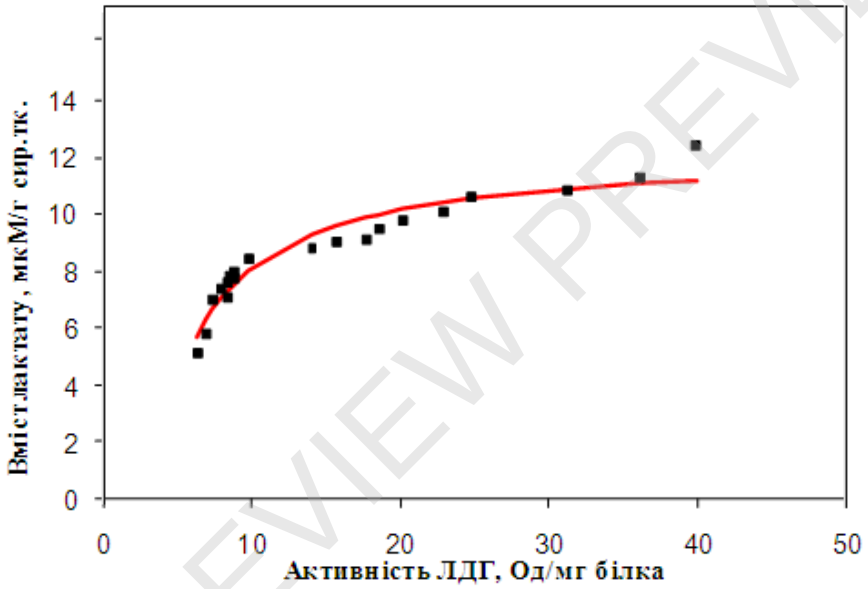


Рис. 11. Рівняння гіперболи

Отже, за результатами регресійного аналізу можна зробити висновок про те, що рівняння гіперболи, яке за результатами експерименту має вигляд: $\boldsymbol{y = 12,16 - 40,61/x}$ *(Рис. 11: крива) в 12,62 рази краще описує зміни залежної змінної (вміст лактату), ніж середнє значення аргументу.*

1.4. Показникове рівняння кривої

1.4.1. Формуємо таблицю з первинних даних та обчислень допоміжних величин для обчислення коефіцієнтів *a* і *b* даного рівняння:

Акт. ЛДГ X_i	**Вміст лактату Y_i**	$ln\ y_i$	x_i^2	$x_i\ ln\ y_i$	Y_x	$y_i – y_x$	$(y_i – y_x)^2$	y_i^2
6,28	5,11	1,631	39,438	10,244	7,039	-1,929	3,720	26,112
6,89	5,82	1,761	47,472	12,135	7,120	-1,300	1,690	33,872
7,34	6,96	1,940	53,876	14,241	7,181	-0,221	0,049	48,442
7,92	7,39	2,000	62,726	15,841	7,259	0,131	0,017	54,612
8,26	7,07	1,956	68,228	16,155	7,306	-0,236	0,056	49,985
8,74	7,73	2,045	76,388	17,874	7,372	0,358	0,128	59,753
8,39	7,81	2,055	70,392	17,245	7,324	0,486	0,236	60,996
8,34	7,56	2,023	69,556	16,871	7,317	0,243	0,059	57,154
8,74	8,00	2,079	76,388	18,174	7,372	0,628	0,394	64,000
9,72	8,45	2,134	94,478	20,744	7,509	0,941	0,885	71,403
14	8,77	2,171	196,00	30,399	8,139	0,631	0,398	76,913
15,6	9,01	2,198	243,36	34,294	8,388	0,622	0,386	81,180
17,7	9,13	2,212	313,29	39,145	8,727	0,403	0,163	83,357
18,5	9,45	2,246	342,25	41,551	8,859	0,591	0,349	89,303
20,1	9,77	2,279	404,01	45,814	9,130	0,640	0,410	95,453
22,9	10,1	2,313	524,41	52,957	9,624	0,476	0,227	102,01
24,8	10,6	2,361	615,04	58,549	9,974	0,626	0,392	112,36
31,3	10,8	2,380	979,69	74,480	11,27	-0,472	0,223	116,64
36,2	11,3	2,425	1310,4	87,778	12,36	-1,061	1,126	127,69
39,9	12,4	2,518	1592,0	100,46	13,253	-0,853	0,727	153,76
Σx_i	**Σy_i**	**$\Sigma\ lny_i$**	**Σx_i^2**	**$\Sigma x_i\ ln\ y_i$**	Σy_x	**$\Sigma(y_i – y_x)$**	**$\Sigma(y_i – y_x)^2$**	**Σy_i^2**
321,62	**173,23**	**42,728**	**7179,4**	**724,95**	**172,53**	**0,7034**	**11,634**	**1565,0**

Знаходимо коефіцієнти *a* і *b* *за формулами* **(96)** і **(97)**, відповідно:

ln a = (42,728 × 7179,4 – 724,95 × 321,62) / (20 × 7179,4 – $(321,62)^2$) = 1,8332

Звідси, беручи до уваги те, що основа натурального логарифму ***e = 2,7182*** отримаємо ***a* = 6,254**.

Ln b = (20 × 724,95 – 321,62 × 42,728) / (20 × 7179,4 – $(321,62)^2$) = 0,01885

Тому ***b*** = **1,019**.

В результаті отримуємо наступне рівняння регресії:

$$y = 6,254 \times 1,019^{x}.$$

За формулою **(99)** обчислюємо величину *середньої помилки апроксимації* $\bar{\varepsilon}$:

$$\bar{\varepsilon} = \frac{1}{20} \times 1{,}640 \times 100\% = 8{,}20\%.$$

Отже, апроксимація даних наведеним вище рівнянням регресії є статистично значущою, оскільки 8,20%<15%.

За допомогою формул **(101)** і **(69)** обчислюємо величини *індексу кореляції* та *коефіцієнта детермінації*:

$$\eta = \sqrt{1 - \frac{11{,}634}{64{,}562}} = 0{,}91;$$

$$R^2 = 0{,}91^2 \times 100\% = 0{,}83 \times 100\% = 83\%.$$

За формулою **(102)** обчислюємо фактичне значення *F-критерію Фішера*. При цьому порівнюємо загальну дисперсію S_y^2 **(103)** із залишковою $S^2_{зал}$ **(104)**:

$$\boldsymbol{S}_y^2 = (1565{,}0 - 1500{,}43) / 19 = 3{,}3984;$$

$$\boldsymbol{S}^2_{зал} = 11{,}634 / 18 = 0{,}6463.$$

Тоді

$$\boldsymbol{F} = 3{,}3984 / 0{,}6463 = \mathbf{5{,}26}.$$

При рівні статистичної значущості $P=0{,}05$ $\boldsymbol{F} > \boldsymbol{F}_{кр} = 2{,}22$. Тому лінійне рівняння показникової кривої адекватно описує фактичний вміст лактату відносно активності ЛДГ. При цьому значення $F=5{,}26$ вказує на те, що рівняння показникової кривої лише в 5 раз краще описує цей взаємозв’язок, ніж середнє значення залежної змінної.

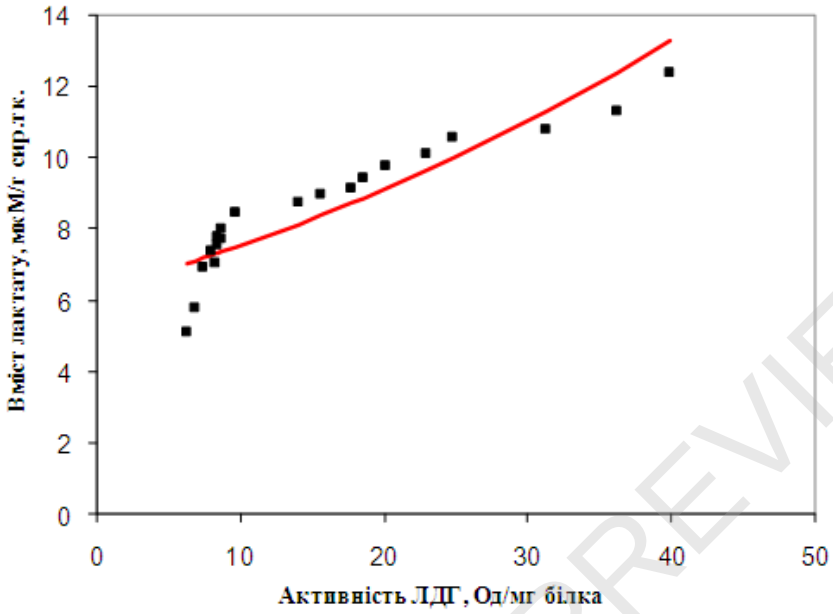


Рис. 12. Показникові рівняння

Отже, за результатами регресійного аналізу можна зробити висновок про те, що показникове рівняння, яке за результатами експерименту має вигляд: $y = 6{,}254 \times 1{,}019^{x}$ *(Рис. 12: крива) лише в* ***5,26*** *раз краще описує зміни залежної змінної (вміст лактату), ніж середнє значення аргументу.*

Обчислення похибок і помилок для вищевказаних рівнянь регресії

Позначивши через δ_1, δ_2, δ_3, δ_4 абсолютні похибки рівнянь лінійної напівлогарифмічної, гіперболічної та показникової регресії, відповідно, за формулою **(106)** обчислюємо їхні величини:

$$\delta_1 = \pm\sqrt{1/20 \times 0{,}001784} = \pm 0{,}0094;$$

$$\delta_2 = \pm\sqrt{1/20 \times 0{,}000257} = \pm 0{,}0036;$$

$$\delta_3 = \pm\sqrt{1/20 \times 0{,}014675} = \pm 0{,}0271;$$

$$\delta_4 = \pm\sqrt{1/20 \times 0{,}703429} = \pm 0{,}1875.$$

Первинні дані та обчислення допоміжних коефіцієнтів рівнянь наведені в табл. 19, позначивши через $Z_{i1} = (y_i - y_x)_1 / y_{x1}$, $Z_{i2} = (y_i - y_x)_2 / y_{x2}$, $Z_{i3} = (y_i - y_x)_3 / y_{x3}$, $Z_{i4} = (y_i - y_x)_4 / y_{x4}$.

Таблиця 19. Допоміжні величини для обчислення похибок і помилок рівнянь регресії

$(y_i - y_x)_1$	$(y_i - y_x)_2$	$(y_i - y_x)_3$	$(y_i - y_x)_4$	y_{x1}	y_{x2}	y_{x3}	y_{x4}	Z_{i1}	Z_{i2}	Z_{i3}	Z_{i4}
-1,9166	-1,2717	-0,5830	-1,9287	7,0266	6,3817	5,6930	7,0387	-0,2728	-0,1993	-0,1024	-0,2740
-1,3084	-0,8381	-0,4455	-1,3000	7,1284	6,6581	6,2655	7,1200	-0,1835	-0,1259	-0,0711	-0,1826
-0,2434	0,1133	0,3331	-0,2205	7,2034	6,8467	6,6269	7,1805	-0,0338	0,0165	0,0503	-0,0307
0,0898	0,3165	0,3579	0,1307	7,3002	7,0735	7,0321	7,2593	0,0123	0,0447	0,0509	0,0180
-0,2869	-0,1288	-0,1732	-0,2359	7,3569	7,1988	7,2432	7,3059	-0,0390	-0,0179	-0,0239	-0,0323
0,2931	0,3627	0,2168	0,3577	7,4369	7,3673	7,5132	7,3723	0,0394	0,0492	0,0289	0,0485
0,4314	0,5646	0,4906	0,4862	7,3786	7,2454	7,3194	7,3238	0,0585	0,0779	0,0670	0,0664
0,1898	0,3324	0,2697	0,2430	7,3702	7,2276	7,2903	7,3170	0,0258	0,0460	0,0370	0,0332
0,5631	0,6327	0,4868	0,6277	7,4369	7,3673	7,5132	7,3723	0,0757	0,0859	0,0648	0,0851
0,8496	0,7658	0,4683	0,9405	7,6004	7,6842	7,9817	7,5095	0,1118	0,0997	0,0587	0,1252
0,4557	-0,0021	-0,4891	0,6305	8,3143	8,7721	9,2591	8,1395	0,0548	-0,0002	-0,0528	0,0775
0,4288	-0,0848	-0,5466	0,6217	8,5812	9,0948	9,5566	8,3883	0,0500	-0,0093	-0,0572	0,0741
0,1985	-0,3414	-0,7355	0,4035	8,9315	9,4714	9,8655	8,7265	0,0222	-0,0360	-0,0746	0,0462
0,3851	-0,1532	-0,5147	0,5911	9,0649	9,6032	9,9647	8,8589	0,0425	-0,0160	-0,0517	0,0667
0,4382	-0,0805	-0,3695	0,6402	9,3318	9,8505	10,139	9,1298	0,0470	-0,0082	-0,0364	0,0701
0,3012	-0,1394	-0,2865	0,4762	9,7988	10,239	10,386	9,6238	0,0307	-0,0136	-0,0276	0,0495
0,4843	0,1229	0,0776	0,6258	10,1157	10,477	10,522	9,9742	0,0479	0,0117	0,0074	0,0627
-0,3999	-0,3712	-0,0625	-0,4722	11,1999	11,171	10,862	11,272	-0,0357	-0,0332	-0,0058	-0,0419
-0,7173	-0,3049	0,2619	-1,0613	12,0173	11,605	11,038	12,361	-0,0597	-0,0263	0,0237	-0,0859
-0,2344	0,5050	1,2579	-0,8528	12,6344	11,895	11,142	13,253	-0,0186	0,0425	0,1129	-0,0643
								ΣZ_{i1}	ΣZ_{i2}	ΣZ_{i3}	ΣZ_{i4}
								-0,0245	-0,0117	-0,0019	0,1117

Z_1^2	Z_2^2	Z_3^2	Z_4^2
0,0744	0,0397	0,0105	0,0751
0,0337	0,0158	0,0051	0,0333
0,0011	0,0003	0,0025	0,0009
0,0002	0,0020	0,0026	0,0003
0,0015	0,0003	0,0006	0,0010
0,0016	0,0024	0,0008	0,0024
0,0034	0,0061	0,0045	0,0044
0,0007	0,0021	0,0014	0,0011
0,0057	0,0074	0,0042	0,0073
0,0125	0,0099	0,0034	0,0157
0,0030	0,0000	0,0028	0,0060
0,0025	0,0001	0,0033	0,0055
0,0005	0,0013	0,0056	0,0021
0,0018	0,0003	0,0027	0,0045
0,0022	0,0001	0,0013	0,0049
0,0009	0,0002	0,0008	0,0024
0,0023	0,0001	0,0001	0,0039

***Абсолютна похибка* (105)**

$\Delta_1 = \pm \sqrt{(1/20) \cdot 0{,}153188457 \cdot 100} = \pm$ **8,752 %;**

$\Delta_2 = \pm \sqrt{(1/20) \cdot 0{,}0916958 \cdot 100} = \pm$ **6,771 %;**

$\Delta_3 = \pm \sqrt{(1/20) \cdot 0{,}0653363 \cdot 100} = \pm$ **5,716 %.**

$\Delta_4 = \pm \sqrt{(1/20) \cdot 0{,}184182 \cdot 100} = \pm$ **9,596 %.**

***Системна помилка* (107)**

$o_{p1} = (1/20) \cdot (-0{,}0245) \cdot 100 =$ **- 0,1227;**

$o_{p2} = (1/20) \cdot (-0{,}0117) \cdot 100 =$ **- 0,0587;**

$o_{p3} = (1/20) \cdot (-0{,}0019) \cdot 100 =$ **- 0,0095;**

$o_{p4} = (1/20) \cdot 0{,}1117 \cdot 100 =$ **0,5585.**

0,0013	0,0011	0,0000	0,0018
0,0036	0,0007	0,0006	0,0074
0,0003	0,0018	0,0127	0,0041
Σ	Σ	Σ	Σ
0,1532	0,0917	0,0653	0,1842

Для обчислення випадкової помилки (o_δ) допоміжні дані треба занести до таблиці (табл. 20).

Таблица 20. Допоміжні дані для обчислення випадкової помилки (o_δ)

$\{((y_i - y_x)/y_x)_1 \times 100 - o_{p1}\}^2$	$\{((y_i - y_x)/y_x)^2 \times 100 - o_{p2}\}^2$	$\{((y_i - y_x)/y_x)_3 \times 100 - o_{p3}\}^2$	$\{((y_i - y_x)/y_x)_4 \times 100 - o_{p4}\}^2$
737,323	394,746	104,664	781,748
332,388	156,971	50,4255	354,063
10,6043	2,93420	25,3621	13,1747
1,83173	20,5480	26,0006	1,54088
14,2629	2,99675	5,67073	14,3493
16,5113	24,8220	8,38124	18,4394
35,6412	61,6396	45,0626	36,9596
7,27795	21,6964	13,7524	7,63485
59,1967	74,7710	42,1034	63,3058
127,715	100,502	34,5340	143,177
31,4005	0,00118	27,8001	51,6672
26,2135	0,76359	32,6056	46,9598
5,50197	12,5726	55,4368	16,5254
19,1052	2,36142	26,5819	37,3782
23,2198	0,57606	13,2075	41,6575
10,2166	1,69765	7,55672	19,2688
24,1070	1,51732	0,55827	32,6703
11,8902	10,6534	0,31980	22,5418
34,1743	6,59603	5,67510	83,6127
3,00228	18,5232	127,663	48,9077
Σ	**Σ**	**Σ**	**Σ**
1531,58	916,89	653,36	1835,58

Таким чином, використовуючи формулу **(108)** та дані із табл. 20, обчислюємо випадкові помилки наведених вище рівнянь регресії:

$$o_{\delta 1} = \pm\sqrt{(1/20)\times 1531{,}58} = \pm 8{,}75;$$

$$o_{\delta 2} = \pm\sqrt{(1/20)\times 916{,}89} = \pm 6{,}77;$$

$$o_{\delta 3} = \pm\sqrt{(1/20)\times 653{,}36} = \pm 5{,}72;$$

$$o_{\delta 4} = \pm\sqrt{(1/20)\times 1835{,}58} = \pm 9{,}58.$$

Загальний висновок. Взаємне порівняння рівнянь регресії (табл. 21) показує, що найкращі результати дають рівняння регресії, які описуються рівнянням гіперболи і лінійними рівнянням з логарифмуванням факторної ознаки (напівлогарифмічне). Ці рівняння показали найвищі значення *F*-критерію Фішера та найменші помилки та похибки. З всіма показниками гіберболічна взаємозалежність досліджуваних ознак є найбільш правдоподібною.

Таблиця 21. Взаємне порівняння рівнянь регресії

Вид рівняння регресії	**Дисперсія**		***F*-критерій**		$\bar{\varepsilon}$ %	**η**	R^2	**Похибка рівняння**		**Помилка**	
	зага-льна	зали-шкова	*F*	$F_{кр}$				абсо-лютн. **Δ**	відносна **δ, %**	система-тична $\mathbf{o_р}$	випад-кова $\mathbf{o_δ}$
Лінійне $y=0{,}167x+5{,}979$	3,398	0,484	7,0	2,2	8,09	0,93	0,86	±0,0094	±8,752	- 0,1227	±8,75
Напівлогорифмічне $y=0{,}903+2{,}982\ln x$		0,259	13,1		6,07	0,96	0,92	±0,0036	±6,771	- 0,0587	±6,77
Гіперболічне $y=12{,}16–40{,}61/x$		0,242	12,62		5,80	0,96	0,92	±0,0271	±5,716	- 0,0095	±5,72
Експоненційне $y = 6{,}254 \times 1{,}019^x$		0,646	5,25		9,72	0,91	0,83	±0,1875	±9,596	0,5585	±9,58

Код Python для вибору графіка регресії і його виведення на екран, а також обчислення коефіцієнтів Фішера і детермінації наведений нижче. Код будує та порівнює кілька регресійних моделей на тих самих даних, де x – незалежна змінна, y – залежна, а *X = x.reshape(-1,1)* підготовлено для сумісності зі *sklearn*. Для неупередженої оцінки якості застосовано вкладену крос-перевірку: внутрішня (*inner_cv=3*) через GridSearchCV добирає гіперпараметри, зокрема степінь полінома, а зовнішня (outer_*cv=5*)

формує крос-валідаційні прогнози *yhat_cv*, на яких і рахуються метрики.

До набору моделей входять лінійна та напівлогарифмічна регресії (остання реалізована власним sklearn-сумісним естіматором ($y=a+b\ln x$)), поліноміальна регресія зі степенем, обраним на внутрішній CV, а також аналітичні форми, що оцінюються через *curve_fit*: гіпербола ($y=a+b/x$), експонента ($y=Ae^{Bx}$) і степенева ($y=Ax^B$) (обидві лінеаризуються лог-перетворенням), та логістична крива ($y = \frac{a}{1+be^{-cx}}$). Для кожної моделі зберігаються як крос-валідаційні прогнози, так і прогнози повної підгонки *yhat_full*.

Окрема функція *evaluate* обчислює ключові показники: (R^2), MAE, RMSE, відносний RMSE у %, MAPE у %, систематичне зміщення (Bias) і випадкову складову помилки (Random) у відсотках, а також модуль кореляції між фактом і прогнозом. Для повної підгонки додатково рахується F-критерій за відношенням дисперсій, де потрібна кількість параметрів моделі *p*. Допоміжна функція *scale* масштабує будь-які метрики до [0,1], інвертуючи їх для величин, які треба мінімізувати.

Щоб вибрати «найкращу» модель, код утворює зважений композитний бал: для кожної метрики береться нормований показник, множиться на вагу з *weights* і підсумовується. Метрики з природою «чим більше – тим краще” (наприклад, (R^2), кореляція, F) максимізуються, а такі як MAE, RMSE, MAPE, RRMSE, Bias і Random – мінімізуються через інверсію шкали. Модель із найбільшим сумарним балом оголошується переможцем; далі виводиться таблиця з усіма метриками для прозорого порівняння.

Наприкінці будуються два типи графіків: окремий – із даними та гладкою кривою найкращої моделі, доповнений підписом з її рівнянням і стислим рядком метрик; і сітка з усіма моделями, де кожна панель містить емпіричні точки, відповідну криву, рівняння та ті самі ключові показники, що дає наочне уявлення про форму підгонки та якість прогнозу.

```
import numpy as np
import matplotlib.pyplot as plt
import warnings
import math

from sklearn.base import clone, BaseEstimator,
RegressorMixin
from sklearn.model_selection import KFold,
GridSearchCV
from sklearn.pipeline import make_pipeline
from sklearn.preprocessing import PolynomialFeatures
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score,
mean_absolute_error, mean_squared_error
from scipy.optimize import curve_fit

# Data
x = np.array([6.28, 6.89, 7.34, 7.92, 8.26, 8.74,
8.39, 8.34, 8.74, 9.72, 14.0, 15.6, 17.7, 18.5, 20.1,
22.9, 24.8, 31.3, 36.2, 39.9])
y = np.array([5.11, 5.82, 6.96, 7.39, 7.07, 7.73,
7.81, 7.56, 8.00, 8.45, 8.77, 9.01, 9.13, 9.45, 9.77,
10.1, 10.6, 10.8, 11.3, 12.4])

X = x.reshape(-1, 1)

#Config
# Outer/inner folds for nested CV
outer_cv = 5
inner_cv = 3
random_state = 42
```

```
# Composite-score weights (tune as needed)
weights = {
    "R2": 3.0, "Corr_th": 2.0, "F_value": 2.0,
    "MAE": 1.5, "RMSE": 1.5, "RRMSE_%": 1.0, "MAPE_%":
2.0,
    "Bias_op_%": 1.0, "Random_oδ_%": 1.0
}


# Helpers
def average_approx_error(y, yhat):
    """MAPE / Average Approximation Error in %,
requires y_i > 0."""
    y = np.asarray(y); yhat = np.asarray(yhat)
    mask = y > 0
    return np.mean(np.abs((yhat[mask] - y[mask]) /
y[mask])) * 100.0


def evaluate(y, yhat_cv, yhat_full, p):
    """
    Compute metrics.
    - Uses cross-validated predictions (yhat_cv) for
R2/MAE/RMSE/RRMSE/MAPE/Bias/Random/Corr.
    - Uses full-fit predictions (yhat_full) to compute
F-statistic (requires parameter count p).
    """
    n = len(y)
    mask = np.isfinite(yhat_cv)
    y_cv = y[mask]; yhat_cv = yhat_cv[mask]

    r2 = r2_score(y_cv, yhat_cv)
    r2_full = np.nan if yhat_full is None else
r2_score(y, yhat_full)
    mae = mean_absolute_error(y_cv, yhat_cv)
    rmse = np.sqrt(mean_squared_error(y_cv, yhat_cv))
    rrmse = rmse / np.mean(y_cv) * 100.0
    mape = average_approx_error(y_cv, yhat_cv)
    op = np.mean((y_cv - yhat_cv) / y_cv) * 100.0
# systematic bias, %
    od = np.sqrt(np.mean(((y_cv - yhat_cv) / y_cv *
100.0 - op) ** 2))  # random error, %
    corr = np.abs(np.corrcoef(y_cv, yhat_cv)[0, 1])
# correlation of y and yhat
```

```
    # F-statistic on the full fit (only for parametric
models)
    if np.isnan(p) or yhat_full is None:
        F = np.nan
    else:
        Sy2 = (np.sum(y**2) - (np.sum(y)**2) / n) / (n
- 1)          # variance of y
        St2 = np.sum((y - yhat_full) ** 2) / (n - p)
# residual variance
        F = Sy2 / St2
# Fisher's F

    return {"R2_full": r2_full, "R2": r2, "MAE": mae,
"RMSE": rmse, "RRMSE_%": rrmse,
            "MAPE_%": mape, "Bias_op_%": op,
"Random_oδ_%": od,
            "Corr_th": corr, "F_value": F}


def scale(arr, reverse=False):
    """
    Min-max scale to [0,1].
    If reverse=True, larger original values become
smaller scores (for minimize-type metrics).
    """
    arr = np.array(arr, dtype=float)
    finite = np.isfinite(arr)
    if finite.sum() <= 1:
        s = np.zeros_like(arr); s[finite] = 1.0
        return s
    lo, hi = np.nanmin(arr), np.nanmax(arr)
    if np.isclose(hi, lo):
        s = np.ones_like(arr)
    else:
        s = (arr - lo) / (hi - lo)
    s[~finite] = 0.0
    return 1.0 - s if reverse else s


def nested_cv_predict_estimator(base_estimator,
param_grid, X, y,
                                outer_cv=5,
inner_cv=3, scoring="r2",
                                random_state=42):
    """
    Nested CV for parametric estimators.
```

```
    - Inner CV (GridSearchCV) selects hyperparameters
(e.g., polynomial degree).
    - Outer CV produces unbiased predictions for
metrics.
    - Returns (CV predictions, full-fit predictions,
refit best estimator).
    """
    kf_outer = KFold(n_splits=outer_cv, shuffle=True,
random_state=random_state)
    yhat_cv = np.empty(y.shape[0], dtype=float)

    for tr, te in kf_outer.split(X):
        if param_grid:
            searcher = GridSearchCV(base_estimator,
param_grid, cv=inner_cv, scoring=scoring)
            searcher.fit(X[tr], y[tr])
            est = searcher.best_estimator_
        else:
            est = clone(base_estimator).fit(X[tr],
y[tr])
        yhat_cv[te] = est.predict(X[te])

    # Refit on all data (for plotting and F-statistic)
    if param_grid:
        final_search = GridSearchCV(base_estimator,
param_grid, cv=inner_cv, scoring=scoring)
        final_search.fit(X, y)
        best_est = final_search.best_estimator_
    else:
        best_est = clone(base_estimator).fit(X, y)

    yhat_full = best_est.predict(X)
    return yhat_cv, yhat_full, best_est


# curve_fit models + manual CV
def hyperbolic(x, a, b): return a + b / x
def logistic(x, a, b, c): return a / (1.0 + b *
np.exp(-c * x))


def cv_curvefit(x, y, fit_func, pred_func, p,
outer_cv=5):
    """
    Manual CV for models trained with curve_fit.
```

```
    Returns (CV predictions, full-fit predictions,
fitted params, param count p).
    """
    kf = KFold(n_splits=outer_cv, shuffle=True,
random_state=random_state)
    yhat_cv = np.empty_like(y, dtype=float)
    for tr, te in kf.split(x):
        try:
            params = fit_func(x[tr], y[tr])
            yhat_cv[te] = pred_func(params, x[te])
        except Exception:
            yhat_cv[te] = np.nan
    # Full fit
    try:
        params_full = fit_func(x, y)
        yhat_full = pred_func(params_full, x)
    except Exception:
        params_full, yhat_full = None, None
    return yhat_cv, yhat_full, params_full, p


# Fit/predict pairs for specific analytic forms
def fit_hyperbolic(x_tr, y_tr):
    with warnings.catch_warnings():
        warnings.simplefilter("ignore")
        pars, _ = curve_fit(hyperbolic, x_tr, y_tr,
maxfev=50000)
    return pars


def pred_hyperbolic(pars, x_te): return
hyperbolic(x_te, *pars)


def fit_logistic(x_tr, y_tr):
    with warnings.catch_warnings():
        warnings.simplefilter("ignore")
        p0 = [y_tr.max(), 1.0, 0.1]
        pars, _ = curve_fit(logistic, x_tr, y_tr,
p0=p0, bounds=(0, np.inf), maxfev=80000)
    return pars


def pred_logistic(pars, x_te): return logistic(x_te,
*pars)
```

```
def fit_exponential(x_tr, y_tr):   # y = A * exp(B x)
    lr = LinearRegression().fit(x_tr.reshape(-1,1),
np.log(y_tr))
    A, B = np.exp(lr.intercept_), lr.coef_[0]
    return np.array([A, B])


def pred_exponential(pars, x_te):  return pars[0] *
np.exp(pars[1] * x_te)


def fit_power(x_tr, y_tr):         # y = A * x^B
    lr = LinearRegression().fit(np.log(x_tr).reshape(-
1,1), np.log(y_tr))
    A, B = np.exp(lr.intercept_), lr.coef_[0]
    return np.array([A, B])


def pred_power(pars, x_te): return pars[0] * (x_te **
pars[1])


# Semi-log estimator (sklearn-compatible)
class SemiLogEstimator(BaseEstimator, RegressorMixin):
    """
    Semi-log model: y = a + b * ln(x).
    Implemented as an sklearn-compatible wrapper
around LinearRegression so it can be cloned.
    """
    def __init__(self):
        self._lr = LinearRegression()

    def fit(self, X, y):
        X = np.asarray(X).reshape(-1, 1)
        if np.any(X <= 0):
            raise ValueError("SemiLogEstimator
requires X > 0 for log transform.")
        self._lr.fit(np.log(X), y)
        return self

    def predict(self, X):
        X = np.asarray(X).reshape(-1, 1)
        return self._lr.predict(np.log(X))

    def get_params(self, deep=True):
```

```
        # No hyperparameters to expose
        return {}

    def set_params(self, **params):
        # Nothing to set
        return self


# Build models
models = {}

# Linear
lin = LinearRegression()
cv_pred, full_pred, est =
nested_cv_predict_estimator(lin, {}, X, y, outer_cv,
inner_cv)
models["Linear"] = {"yhat_cv": cv_pred, "yhat_full":
full_pred, "p": 2, "est_or_params": est}

# Semi-logarithmic: y = a + b ln(x)
semi = SemiLogEstimator()
cv_pred, full_pred, est =
nested_cv_predict_estimator(semi, {}, X, y, outer_cv,
inner_cv)
models["Semi-logarithmic"] = {"yhat_cv": cv_pred,
"yhat_full": full_pred, "p": 2, "est_or_params": est}

# Polynomial (degree chosen by GridSearchCV)
poly_est =
make_pipeline(PolynomialFeatures(include_bias=False),
LinearRegression())
poly_grid = {"polynomialfeatures__degree": [2, 3, 4]}
cv_pred, full_pred, best_poly =
nested_cv_predict_estimator(poly_est, poly_grid, X, y,
outer_cv, inner_cv)
deg =
best_poly.get_params()["polynomialfeatures__degree"]
models[f"Polynomial(deg={deg})"] = {"yhat_cv":
cv_pred, "yhat_full": full_pred, "p": deg + 1,
"est_or_params": best_poly}

# Hyperbolic: y = a + b/x
cv_pred, full_pred, pars, p = cv_curvefit(x, y,
fit_hyperbolic, pred_hyperbolic, p=2,
outer_cv=outer_cv)
```

```
models["Hyperbolic"] = {"yhat_cv": cv_pred,
"yhat_full": full_pred, "p": p, "est_or_params": pars}

# Exponential: y = A * exp(B x)
cv_pred, full_pred, pars, p = cv_curvefit(x, y,
fit_exponential, pred_exponential, p=2,
outer_cv=outer_cv)
models["Exponential"] = {"yhat_cv": cv_pred,
"yhat_full": full_pred, "p": p, "est_or_params": pars}

# Power: y = A * x^B
cv_pred, full_pred, pars, p = cv_curvefit(x, y,
fit_power, pred_power, p=2, outer_cv=outer_cv)
models["Power"] = {"yhat_cv": cv_pred, "yhat_full":
full_pred, "p": p, "est_or_params": pars}

# Logistic: y = a / (1 + b e^{-c x})
cv_pred, full_pred, pars, p = cv_curvefit(x, y,
fit_logistic, pred_logistic, p=3, outer_cv=outer_cv)
models["Logistic"] = {"yhat_cv": cv_pred, "yhat_full":
full_pred, "p": p, "est_or_params": pars}

# Metrics & selection
for name in list(models.keys()):
    m = models[name]
    models[name]["metrics"] = evaluate(y,
m["yhat_cv"], m["yhat_full"], m["p"])

maximize = ["R2", "Corr_th", "F_value"]  # higher is
better
minimize = ["MAE", "RMSE", "RRMSE_%", "MAPE_%",
"Bias_op_%", "Random_oδ_%"]  # lower is better

names = list(models.keys())
scores = np.zeros(len(names))
for i, nm in enumerate(names):
    s = 0.0
    for k in maximize:
        s += weights[k] *
scale([models[n]["metrics"][k] for n in names])[i]
    for k in minimize:
        s += weights[k] *
scale([models[n]["metrics"][k] for n in names],
reverse=True)[i]
    scores[i] = s
```

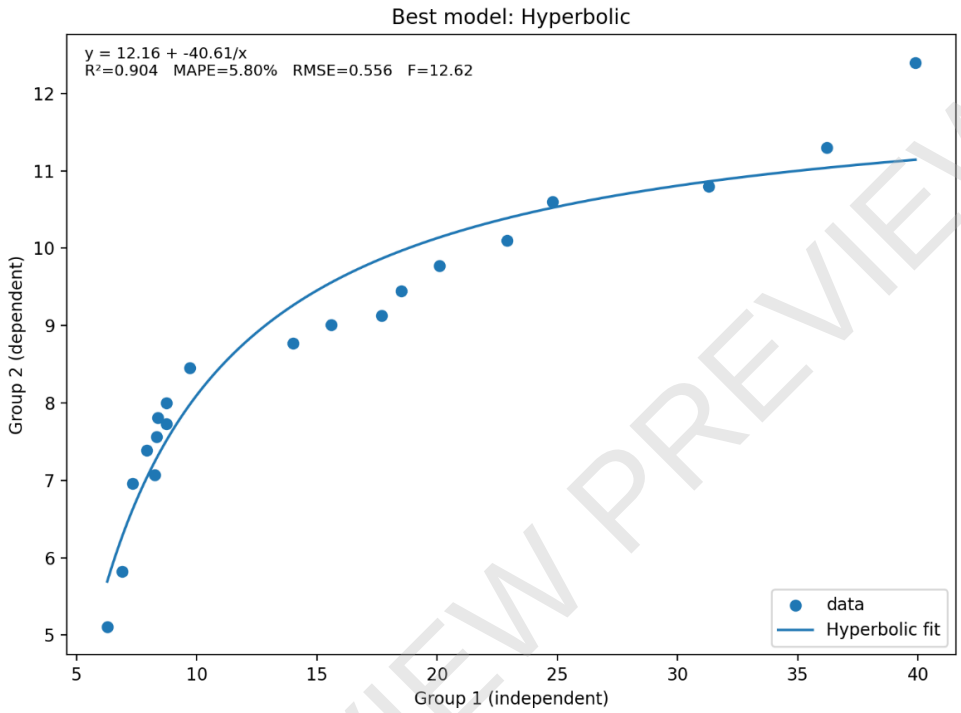
```
best_idx = int(np.argmax(scores))
best_name = names[best_idx]
print("Best model:", best_name)

# Metrics table
hdr = ("Model\tR2\tCV-
R2\tCorr\tF\tMAE\tRMSE\tRRMSE(%)\tMAPE(%)\t|Bias|(%)\t
Random(%)")
print(hdr)
for nm in names:
    m = models[nm]["metrics"]
    F = m['F_value']
    F_out = "NaN" if np.isnan(F) else f"{F:.2f}"
    line =
f"{nm}\t{m['R2_full']:.4f}\t{m['R2']:.4f}\t{m['Corr_th
']:.4f}\t{F_out}\t" \

f"{m['MAE']:.4f}\t{m['RMSE']:.4f}\t{m['RRMSE_%']:.2f}\
t" \

f"{m['MAPE_%']:.2f}\t{abs(m['Bias_op_%']):.2f}\t{m['Ra
ndom_oδ_%']:.2f}"
    print(line)

# Plots
# Common X-grid for smooth curves
xx = np.linspace(x.min(), x.max(), 400)


def predict_curve(name, xx):
    """
    Produce smooth predictions for plotting from the
fitted object/params.
    """
    obj = models[name]["est_or_params"]
    if name.startswith("Polynomial"):
        return obj.predict(xx.reshape(-1,1))
    if name == "Linear":
        return obj.predict(xx.reshape(-1,1))
    if name == "Semi-logarithmic":
        return obj.predict(xx.reshape(-1,1))
    if name == "Hyperbolic" and obj is not None:
        return hyperbolic(xx, *obj)
    if name == "Exponential" and obj is not None:
        return pred_exponential(obj, xx)
    if name == "Power" and obj is not None:
```

```
        return pred_power(obj, xx)
    if name == "Logistic" and obj is not None:
        return pred_logistic(obj, xx)
    return None


def _fmt_F(v):
    """Format F; show '-' if NaN/inf."""
    return "-" if (v is None or not np.isfinite(v))
else f"{v:.2f}"


def model_equation(name):
    """
    Build a readable equation string with fitted
parameters for the given model.
    Uses 4 significant digits for compactness.
    """
    obj = models[name]["est_or_params"]
    try:
        if name == "Linear":
            a = float(obj.intercept_)
            b = float(obj.coef_.ravel()[0])
            return f"y = {a:.4g} + {b:.4g}·x"
        elif name == "Semi-logarithmic":
            a = float(obj._lr.intercept_)
            b = float(obj._lr.coef_.ravel()[0])
            return f"y = {a:.4g} + {b:.4g}·ln(x)"
        elif name.startswith("Polynomial"):
            linreg =
obj.named_steps["linearregression"]
            poly =
obj.named_steps["polynomialfeatures"]
            a = float(linreg.intercept_)
            coefs = linreg.coef_.ravel()
            terms = [f"{a:.4g}"]
            for pwr, c in enumerate(coefs, start=1):
                sign = " + " if c >= 0 else " - "

terms.append(f"{sign}{abs(float(c)):.4g}·x^{pwr}")
            return "y = " + "".join(terms)
        elif name == "Hyperbolic" and obj is not None:
            a, b = obj
            return f"y = {a:.4g} + {b:.4g}/x"
        elif name == "Exponential" and obj is not
None:
```

```
            A, B = obj
            return f"y = {A:.4g}·e^({B:.4g}·x)"
        elif name == "Power" and obj is not None:
            A, B = obj
            return f"y = {A:.4g}·x^{B:.4g}"
        elif name == "Logistic" and obj is not None:
            a, b, c = obj
            return f"y = {a:.4g}/(1 + {b:.4g}·e^(-
{c:.4g}·x))"
    except Exception:
        pass
    return "(equation unavailable)"


def metrics_line(name):
    """
    Build a compact metrics line for annotation: R², 
MAPE (%), RMSE, and F.
    Note: R² is CV-based; F is computed on the full 
fit.
    """
    m = models[name]["metrics"]
    return f"R²={m['R2']:.3f}  
MAPE={m['MAPE_%']:.2f}%   RMSE={m['RMSE']:.3g}  
F={_fmt_F(m['F_value'])}"


def annotate_equation_and_metrics(ax, name, loc=(0.02, 
0.98), fontsize=8):
    """
    Write the equation and key metrics (incl. F) in 
the plot corner.
    """
    eq = model_equation(name)
    ml = metrics_line(name)
    ax.text(loc[0], loc[1],
            f"{eq}\n{ml}",
            transform=ax.transAxes, va="top", 
ha="left",
            fontsize=fontsize,
            bbox=dict(boxstyle="round,pad=0.3", 
fc="white", ec="none", alpha=0.75))


# Plot: BEST model only
plt.figure(figsize=(8, 6))
```

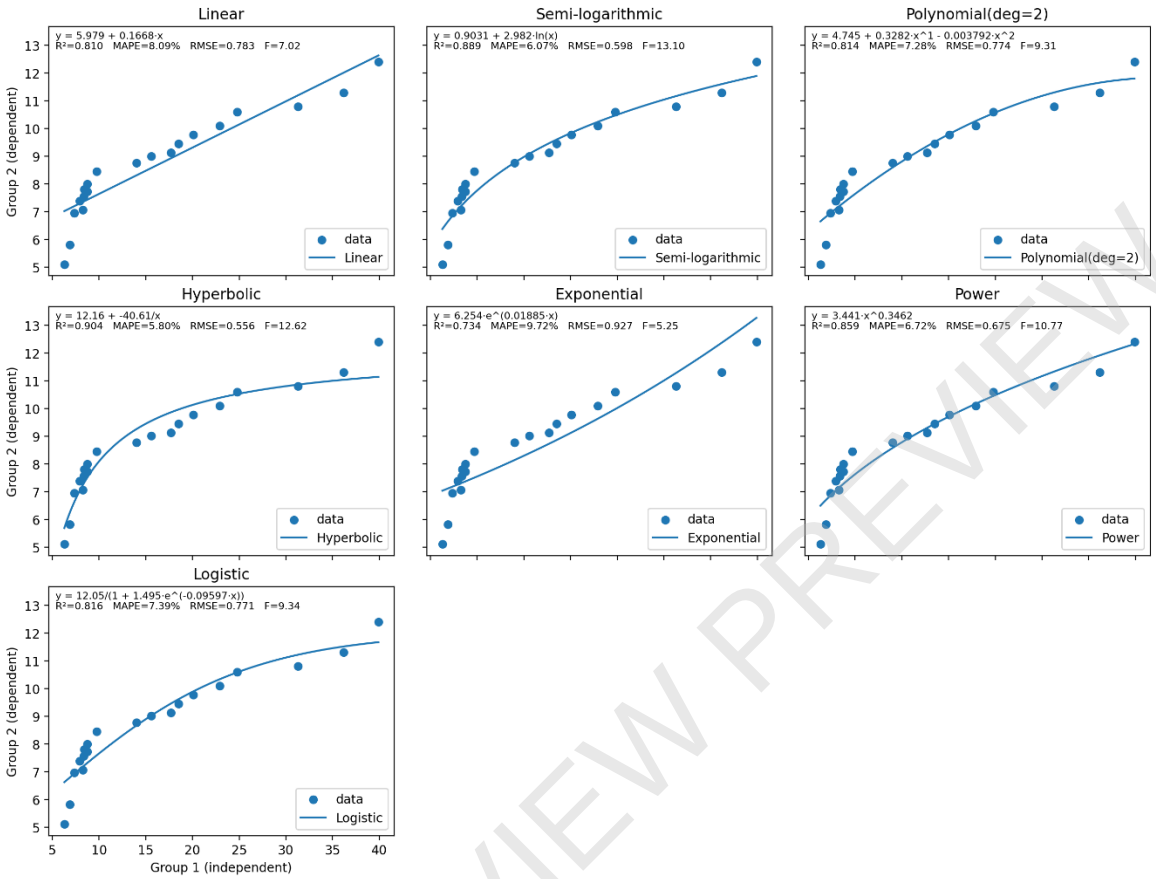
```
plt.scatter(x, y, label="data")
yy_best = predict_curve(best_name, xx)
if yy_best is not None:
    plt.plot(xx, yy_best, label=f"{best_name} fit")
plt.title(f"Best model: {best_name}")
plt.xlabel("Group 1 (independent)")
plt.ylabel("Group 2 (dependent)")
annotate_equation_and_metrics(plt.gca(), best_name,
fontsize=9)
plt.legend()
plt.tight_layout()
plt.show()

# Plot: grid of per-model fits with equation + metrics
(incl. F)
n_models = len(names)
ncols = 3
nrows = math.ceil(n_models / ncols)

fig, axes = plt.subplots(nrows, ncols, figsize=(12,
4*nrows), sharex=True, sharey=True)
axes = np.array(axes).reshape(-1)

for i, nm in enumerate(names):
    ax = axes[i]
    ax.scatter(x, y, label="data")
    yy = predict_curve(nm, xx)
    if yy is not None:
        ax.plot(xx, yy, label=nm)
    ax.set_title(nm)
    annotate_equation_and_metrics(ax, nm, fontsize=8)
    ax.legend(loc="lower right")

# Hide any unused axes
for j in range(i+1, len(axes)):
    fig.delaxes(axes[j])

# Axis labels
for ax in axes[-ncols:]:
    ax.set_xlabel("Group 1 (independent)")
for r in range(0, len(axes), ncols):
    axes[r].set_ylabel("Group 2 (dependent)")

plt.tight_layout()
plt.show()
```

Результат виконання цього коду наступний:




```
Best model: Hyperbolic
```

Mo	R2	CV-R2	Corr	F	MAE	RMSE	RRMSE(%)	MAPE(%)	Bias(%)	Random(%)
Li	0.8651	0.8102	0.9012	7.02	0.5925	0.7828	9.04	8.09	1.19	12.90
Sl	0.9277	0.8893	0.9431	13.10	0.4412	0.5979	6.90	6.07	0.68	9.59
Po	0.9039	0.8142	0.9024	9.31	0.5433	0.7744	8.94	7.28	1.39	11.84
Hy	0.9249	0.9041	0.9509	12.62	0.4700	0.5563	6.42	5.80	0.47	6.94
Ex	0.8194	0.7337	0.8730	5.25	0.7522	0.9271	10.70	9.72	0.82	14.08
Po	0.9121	0.8589	0.9297	10.77	0.4806	0.6749	7.79	6.72	0.48	10.87
Lo	0.9042	0.8159	0.9033	9.34	0.5470	0.7710	8.90	7.39	1.05	11.97



Регресійний аналіз за вказаним прикладом можна також провести і у R за алгоритмом:

```
suppressPackageStartupMessages({
  library(stats)
  library(ggplot2)
  library(gridExtra)
})

set.seed(42)

#         Data
x <- c(6.28, 6.89, 7.34, 7.92, 8.26, 8.74, 8.39, 8.34,
8.74, 9.72,
       14.0, 15.6, 17.7, 18.5, 20.1, 22.9, 24.8, 31.3,
36.2, 39.9)
y <- c(5.11, 5.82, 6.96, 7.39, 7.07, 7.73, 7.81, 7.56,
8.00, 8.45,
```

```
       8.77, 9.01, 9.13, 9.45, 9.77, 10.1, 10.6, 10.8,
11.3, 12.4)

X <- matrix(x, ncol = 1)

#        Config
outer_cv <- 5
inner_cv <- 3
random_state <- 42

weights <- c(
  R2 = 3.0, Corr_th = 2.0, F_value = 2.0,
  MAE = 1.5, RMSE = 1.5, `RRMSE_%` = 1.0, `MAPE_%` =
2.0,
  `Bias_op_%` = 1.0, `Random_oδ_%` = 1.0
)

#        Helpers
average_approx_error <- function(y, yhat) {
  mask <- y > 0
  mean(abs((yhat[mask] - y[mask]) / y[mask])) * 100
}

r2_score <- function(y, yhat) {
  ss_res <- sum((y - yhat)^2)
  ss_tot <- sum((y - mean(y))^2)
  1 - ss_res / ss_tot
}

evaluate <- function(y, yhat_cv, yhat_full, p) {
  n <- length(y)
  mask <- is.finite(yhat_cv)
  y_cv <- y[mask]; yhat_cv <- yhat_cv[mask]

  r2 <- r2_score(y_cv, yhat_cv)
  r2_full <- if (any(is.na(yhat_full))) NA_real_ else
r2_score(y, yhat_full)
  mae <- mean(abs(y_cv - yhat_cv))
  rmse <- sqrt(mean((y_cv - yhat_cv)^2))
  rrmse <- rmse / mean(y_cv) * 100
  mape <- average_approx_error(y_cv, yhat_cv)
  op <- mean((y_cv - yhat_cv) / y_cv) * 100
  od <- sqrt(mean((((y_cv - yhat_cv) / y_cv) * 100 -
op)^2))
  corr <- abs(cor(y_cv, yhat_cv))
```

```
  F_value <- NA_real_
  if (!is.na(p) && all(is.finite(yhat_full))) {
    Sy2 <- (sum(y^2) - (sum(y)^2) / n) / (n - 1)
    St2 <- sum((y - yhat_full)^2) / (n - p)
    F_value <- Sy2 / St2
  }

  list(R2_full = r2_full, R2 = r2, MAE = mae, RMSE =
rmse, `RRMSE_%` = rrmse,
       `MAPE_%` = mape, `Bias_op_%` = op, `Random_oδ_%`
= od,
       Corr_th = corr, F_value = F_value)
}

scale01 <- function(arr, reverse = FALSE) {
  a <- as.numeric(arr)
  finite <- is.finite(a)
  if (sum(finite) <= 1) {
    s <- rep(0, length(a)); s[finite] <- 1; return(s)
  }
  lo <- min(a[finite]); hi <- max(a[finite])
  if (isTRUE(all.equal(hi, lo))) {
    s <- rep(1, length(a))
  } else {
    s <- (a - lo) / (hi - lo)
  }
  s[!finite] <- 0
  if (reverse) 1 - s else s
}

#       CV utilities
kfold_indices <- function(n, k, seed = 42) {
  set.seed(seed)
  idx <- sample.int(n)
  folds <- split(idx, rep(1:k, length.out = n))
  lapply(folds, sort)
}

nested_cv_predict_estimator    <-    function(fit_fun,
pred_fun, X, y,
                                          param_grid    =
NULL,
                                          outer_cv  =  5,
inner_cv = 3,
                                          seed = 42) {
  X <- as.matrix(X)
```

```
  n <- nrow(X)
  folds_outer <- kfold_indices(n, outer_cv, seed)
  yhat_cv <- rep(NA_real_, n)

  for (fold in folds_outer) {
    tr <- setdiff(seq_len(n), fold)
    te <- fold

    if (!is.null(param_grid) && length(param_grid) > 0)
{
      scores <- c()
      for (param in param_grid) {
        inner_folds    <-    kfold_indices(length(tr),
inner_cv, seed + 1)
        yhat_inner <- rep(NA_real_, length(tr))
        for (infold in inner_folds) {
          te2 <- tr[infold]
          tr2 <- setdiff(tr, te2)
          model  <-  fit_fun(X[tr2,  ,  drop  =  FALSE],
y[tr2], param)
          yhat_inner[match(te2, tr)] <- pred_fun(model,
X[te2, , drop = FALSE])
        }
        scores    <-    c(scores,    r2_score(y[tr],
yhat_inner))
      }
      best_param <- param_grid[[which.max(scores)]]
      model <- fit_fun(X[tr, , drop = FALSE], y[tr],
best_param)
    } else {
      model <- fit_fun(X[tr, , drop = FALSE], y[tr],
NULL)
      best_param <- NULL
    }
    yhat_cv[te]  <-  pred_fun(model,  X[te,  ,  drop  =
FALSE])
  }

  if (!is.null(param_grid) && length(param_grid) > 0) {
    scores <- c()
    for (param in param_grid) {
      inner_folds <- kfold_indices(n, inner_cv, seed +
2)
      yhat_inner <- rep(NA_real_, n)
      for (infold in inner_folds) {
        te2 <- infold
```

```
        tr2 <- setdiff(seq_len(n), te2)
        model <- fit_fun(X[tr2, , drop = FALSE], y[tr2],
param)
        yhat_inner[te2]  <-  pred_fun(model,  X[te2,  ,
drop = FALSE])
      }
      scores <- c(scores, r2_score(y, yhat_inner))
    }
    best_param <- param_grid[[which.max(scores)]]
    best_model <- fit_fun(X, y, best_param)
  } else {
    best_model <- fit_fun(X, y, NULL)
    best_param <- NULL
  }
  yhat_full <- pred_fun(best_model, X)

  list(yhat_cv = yhat_cv, yhat_full = yhat_full,
       best_model    =    best_model,    best_param    =
best_param)
}

#        Model fit/predict
# Linear: y = a + b x
fit_linear <- function(X, y, param) {
  X <- as.matrix(X)
  df <- data.frame(y = y, x = as.numeric(X[,1]))
  lm(y ~ x, data = df)
}
pred_linear <- function(model, X) {
  X <- as.matrix(X)
  as.numeric(predict(model,  newdata  =  data.frame(x  =
as.numeric(X[,1]))))
}

# Semi-log: y = a + b ln(x)
fit_semilog <- function(X, y, param) {
  X <- as.matrix(X)
  if (any(X[,1] <= 0)) stop("SemiLog requires X > 0")
  df <- data.frame(y = y, x = as.numeric(X[,1]))
  lm(y ~ log(x), data = df)
}
pred_semilog <- function(model, X) {
  X <- as.matrix(X)
  as.numeric(predict(model,  newdata  =  data.frame(x  =
as.numeric(X[,1]))))
}
```

```
# Polynomial: degree hyperparameter (2,3,4)
fit_poly <- function(X, y, degree) {
  X <- as.matrix(X)
  d <- if (is.null(degree)) 2 else degree
  df <- data.frame(y = y, x = as.numeric(X[,1]))
  for (p in 2:d) df[[paste0("x", p)]] <- df$x^p
  form   <-   as.formula(paste("y   ~",   paste(c("x",
paste0("x", 2:d)), collapse = " + ")))
  lm(form, data = df)
}
pred_poly <- function(model, X) {
  X <- as.matrix(X)
  df <- data.frame(x = as.numeric(X[,1]))
  d <- length(coef(model)) - 1
  if (d >= 2) for (p in 2:d) df[[paste0("x", p)]] <-
df$x^p
  as.numeric(predict(model, newdata = df))
}

# Hyperbolic: y = a + b/x
fit_hyperbolic <- function(X, y, param) {
  X <- as.matrix(X)
  df <- data.frame(x = as.numeric(X[,1]), y = y)
  nls(y ~ a + b / x, data = df,
      start = list(a = mean(y), b = 1),
      control = nls.control(maxiter = 200))
}
pred_hyperbolic <- function(model, X) {
  X <- as.matrix(X)
  predict(model,    newdata    =    data.frame(x    =
as.numeric(X[,1])))
}

# Exponential: y = A * exp(B x)
fit_exponential <- function(X, y, param) {
  X <- as.matrix(X)
  lx <- as.numeric(X[,1])
  fit <- lm(log(y) ~ lx)
  A <- exp(coef(fit)[1]); B <- coef(fit)[2]
  list(A = A, B = B)
}
pred_exponential <- function(model, X) {
  X <- as.matrix(X)
  with(model, A * exp(B * as.numeric(X[,1])))
}
```

```
# Power: y = A * x^B
fit_power <- function(X, y, param) {
  X <- as.matrix(X)
  lx <- log(as.numeric(X[,1])); ly <- log(y)
  fit <- lm(ly ~ lx)
  A <- exp(coef(fit)[1]); B <- coef(fit)[2]
  list(A = A, B = B)
}
pred_power <- function(model, X) {
  X <- as.matrix(X)
  with(model, A * (as.numeric(X[,1])^B))
}

# Logistic: y = a / (1 + b * exp(-c x))
fit_logistic <- function(X, y, param) {
  X <- as.matrix(X)
  df <- data.frame(x = as.numeric(X[,1]), y = y)
  start <- list(a = max(y), b = 1, c = 0.1)
  nls(y ~ a / (1 + b * exp(-c * x)),
      data = df, start = start,
      algorithm = "port",
      lower = c(0, 0, 0),
      control = nls.control(maxiter = 500))
}
pred_logistic <- function(model, X) {
  X <- as.matrix(X)
  predict(model,    newdata    =    data.frame(x    =
as.numeric(X[,1])))
}

#       Build models
models <- list()

# Linear
res     <-     nested_cv_predict_estimator(fit_linear,
pred_linear, X, y,
                                    outer_cv          =
outer_cv, inner_cv = inner_cv,
                                    seed              =
random_state)
models[["Linear"]]  <-  list(yhat_cv  =  res$yhat_cv,
yhat_full = res$yhat_full,
                             p  =  2,  est_or_params  =
res$best_model)
```

```
# Semi-log
res     <-     nested_cv_predict_estimator(fit_semilog,
pred_semilog, X, y,
                                        outer_cv           =
outer_cv, inner_cv = inner_cv,
                                        seed               =
random_state)
models[["Semi-logarithmic"]]    <-    list(yhat_cv    =
res$yhat_cv, yhat_full = res$yhat_full,
                                          p       =       2,
est_or_params = res$best_model)

# Polynomial (degree via inner CV)
poly_degrees <- list(2, 3, 4)
res <- nested_cv_predict_estimator(fit_poly, pred_poly,
X, y,
                                        param_grid         =
poly_degrees,
                                        outer_cv           =
outer_cv, inner_cv = inner_cv,
                                        seed               =
random_state)
deg_best <- length(coef(res$best_model)) - 1
models[[paste0("Polynomial(deg=", deg_best, ")")]] <-
  list(yhat_cv    =    res$yhat_cv,    yhat_full    =
res$yhat_full,
       p = deg_best + 1, est_or_params = res$best_model)

# Reuse outer folds for plain CV models below
folds    <-    kfold_indices(length(y),    outer_cv,
random_state)

# Hyperbolic
cv_pred <- rep(NA_real_, length(y))
for (fold in folds) {
  tr <- setdiff(seq_along(y), fold); te <- fold
  m  <-  try(fit_hyperbolic(X[tr,,drop=FALSE],  y[tr],
NULL), silent = TRUE)
  if (inherits(m, "try-error")) { cv_pred[te] <- NA }
else {
    cv_pred[te]          <-          pred_hyperbolic(m,
X[te,,drop=FALSE])
  }
}
m_full <- try(fit_hyperbolic(X, y, NULL), silent = TRUE)
```

```
y_full   <-   if   (inherits(m_full,   "try-error"))
rep(NA_real_, length(y)) else pred_hyperbolic(m_full,
X)
models[["Hyperbolic"]]  <-  list(yhat_cv  =  cv_pred,
yhat_full = y_full,
                                  p = 2, est_or_params =
if (inherits(m_full, "try-error")) NULL else m_full)

# Exponential
cv_pred <- rep(NA_real_, length(y))
for (fold in folds) {
  tr <- setdiff(seq_along(y), fold); te <- fold
  m <- fit_exponential(X[tr,,drop=FALSE], y[tr], NULL)
  cv_pred[te] <- pred_exponential(m, X[te,,drop=FALSE])
}
m_full <- fit_exponential(X, y, NULL)
y_full <- pred_exponential(m_full, X)
models[["Exponential"]]  <-  list(yhat_cv  =  cv_pred,
yhat_full = y_full,
                                   p = 2, est_or_params =
m_full)

# Power
cv_pred <- rep(NA_real_, length(y))
for (fold in folds) {
  tr <- setdiff(seq_along(y), fold); te <- fold
  m <- fit_power(X[tr,,drop=FALSE], y[tr], NULL)
  cv_pred[te] <- pred_power(m, X[te,,drop=FALSE])
}
m_full <- fit_power(X, y, NULL)
y_full <- pred_power(m_full, X)
models[["Power"]] <- list(yhat_cv = cv_pred, yhat_full
= y_full,
                          p   =   2,   est_or_params   =
m_full)

# Logistic
cv_pred <- rep(NA_real_, length(y))
for (fold in folds) {
  tr <- setdiff(seq_along(y), fold); te <- fold
  m   <-   try(fit_logistic(X[tr,,drop=FALSE],   y[tr],
NULL), silent = TRUE)
  if (inherits(m, "try-error")) { cv_pred[te] <- NA }
else {
    cv_pred[te] <- pred_logistic(m, X[te,,drop=FALSE])
  }
```

```
}
m_full <- try(fit_logistic(X, y, NULL), silent = TRUE)
y_full   <-   if   (inherits(m_full,   "try-error"))
rep(NA_real_, length(y)) else pred_logistic(m_full, X)
models[["Logistic"]]   <-   list(yhat_cv   =   cv_pred,
yhat_full = y_full,
                                p = 3, est_or_params = if
(inherits(m_full, "try-error")) NULL else m_full)

#        Metrics & selection
for (nm in names(models)) {
  m <- models[[nm]]
  models[[nm]]$metrics   <-   evaluate(y,   m$yhat_cv,
m$yhat_full, m$p)
}

maximize <- c("R2", "Corr_th", "F_value")
minimize  <-  c("MAE",  "RMSE",  "RRMSE_%",  "MAPE_%",
"Bias_op_%", "Random_oδ_%")

names_vec <- names(models)
scores <- numeric(length(names_vec))

for (i in seq_along(names_vec)) {
  s <- 0
  for (k in maximize) {
    s  <-  s  +  weights[k]  *  scale01(sapply(models,
function(mm) mm$metrics[[k]]))[i]
  }
  for (k in minimize) {
    s  <-  s  +  weights[k]  *  scale01(sapply(models,
function(mm) mm$metrics[[k]]), reverse = TRUE)[i]
  }
  scores[i] <- s
}

best_idx <- which.max(scores)
best_name <- names_vec[best_idx]
cat("Best model:", best_name, "\n")

#        Equations & labels
model_equation <- function(name) {
  obj <- models[[name]]$est_or_params
  sdig <- function(z) formatC(z, digits = 4, format =
"fg")
  tryCatch({
```

```
    if (name == "Linear") {
      a <- coef(obj)[1]; b <- coef(obj)[2]
      paste0("y = ", sdig(a), " + ", sdig(b), "·x")
    } else if (name == "Semi-logarithmic") {
      a <- coef(obj)[1]; b <- coef(obj)[2]
      paste0("y = ", sdig(a), " + ", sdig(b), "·ln(x)")
    } else if (startsWith(name, "Polynomial")) {
      co <- coef(obj)
      a <- co[1]
      terms <- paste0(ifelse(co[-1] >= 0, " + ", " -
"),
                      sdig(abs(co[-1])),      "·x^",
seq_along(co[-1]))
      paste0("y = ", sdig(a), paste0(terms, collapse =
""))
    } else if (name == "Hyperbolic" && !is.null(obj)) {
      a <- coef(obj)["a"]; b <- coef(obj)["b"]
      paste0("y = ", sdig(a), " + ", sdig(b), "/x")
    } else if (name == "Exponential" && !is.null(obj))
{
      A <- obj$A; B <- obj$B
      paste0("y = ", sdig(A), "·e^(", sdig(B), "·x)")
    } else if (name == "Power" && !is.null(obj)) {
      A <- obj$A; B <- obj$B
      paste0("y = ", sdig(A), "·x^", sdig(B))
    } else if (name == "Logistic" && !is.null(obj)) {
      a <- coef(obj)["a"]; b <- coef(obj)["b"]; c <-
coef(obj)["c"]
      paste0("y = ", sdig(a), "/(1 + ", sdig(b), "·e^(-
", sdig(c), "·x))")
    } else "(equation unavailable)"
  }, error = function(e) "(equation unavailable)")
}

metrics_line <- function(name) {
  m <- models[[name]]$metrics
  paste0("R²=", sprintf("%.3f", m$R2),
         "    MAPE=", sprintf("%.2f", m$`MAPE_%`), "%
RMSE=", sprintf("%.3g", m$RMSE),
         "         F=",   ifelse(is.finite(m$F_value),
sprintf("%.2f", m$F_value), "\u2013"))
}

#        Plotting
xx <- seq(min(x), max(x), length.out = 400)
```

```
predict_curve <- function(name, xx) {
  obj <- models[[name]]$est_or_params
  xx <- as.numeric(xx)
  if (startsWith(name, "Polynomial")) {
    df <- data.frame(x = xx)
    d <- length(coef(obj)) - 1
    if (d >= 2) for (p in 2:d) df[[paste0("x", p)]] <-
df$x^p
    as.numeric(predict(obj, newdata = df))
  } else if (name == "Linear") {
    as.numeric(predict(obj, newdata = data.frame(x =
xx)))
  } else if (name == "Semi-logarithmic") {
    as.numeric(predict(obj, newdata = data.frame(x =
xx)))
  } else if (name == "Hyperbolic" && !is.null(obj)) {
    predict(obj, newdata = data.frame(x = xx))
  } else if (name == "Exponential" && !is.null(obj)) {
    with(obj, A * exp(B * xx))
  } else if (name == "Power" && !is.null(obj)) {
    with(obj, A * (xx^B))
  } else if (name == "Logistic" && !is.null(obj)) {
    predict(obj, newdata = data.frame(x = xx))
  } else {
    rep(NA_real_, length(xx))
  }
}

# --- Best model plot (з формулою та метриками)
yy_best <- predict_curve(best_name, xx)
p_best <- ggplot() +
  geom_point(aes(x, y), data = data.frame(x = x, y =
y)) +
  geom_line(aes(x, y), data = data.frame(x = xx, y =
yy_best)) +
  labs(title = paste0("Best model: ", best_name),
       x = "Group 1 (independent)", y = "Group 2
(dependent)") +
  annotate("text", x = min(x), y = max(y),
           hjust = 0, vjust = 1,
           label = paste(model_equation(best_name),
"\n", metrics_line(best_name)),
           size = 3.2, alpha = 0.8)
print(p_best)

# Таблиця метрик
```

```
fmtF      <-      function(v)      ifelse(is.finite(v),
sprintf("%.2f", v), "NaN")
header            <-            c("Model","R2_full","CV-
R2","Corr","F","MAE","RMSE","RRMSE(%)","MAPE(%)","|Bia
s|(%)","Random(%)")
cat(paste(header, collapse = "\t"), "\n")
for (nm in names_vec) {
  m <- models[[nm]]$metrics
  line                                                <-
sprintf("%s\t%.4f\t%.4f\t%.4f\t%s\t%.4f\t%.4f\t%.2f\t%
.2f\t%.2f\t%.2f",
                  nm,
                  ifelse(is.na(m$R2_full),         NaN,
m$R2_full),
                  m$R2, m$Corr_th, fmtF(m$F_value),
                  m$MAE,      m$RMSE,      m$`RRMSE_%`,
m$`MAPE_%`,
                  abs(m$`Bias_op_%`), m$`Random_oδ_%`)
  cat(line, "\n")
}
# --- Грід з усіма валідними моделями
plot_list <- list()
for (nm in names_vec) {
  yy <- predict_curve(nm, xx)
  if (all(is.na(yy))) next
  p <- ggplot() +
    geom_point(aes(x, y), data = data.frame(x = x, y =
y)) +
    geom_line(aes(x, y), data = data.frame(x = xx, y =
yy)) +
    labs(title = nm, x = "Group 1 (independent)", y =
"Group 2 (dependent)") +
    annotate("text", x = min(x), y = max(y),
             hjust = 0, vjust = 1,
             label  =  paste(model_equation(nm),  "\n",
metrics_line(nm)),
             size = 3, alpha = 0.8)
  plot_list[[nm]] <- p
}
valid_plots    <-    Filter(function(p)    inherits(p,
"ggplot"), plot_list)
if (length(valid_plots) > 0) {
  do.call(grid.arrange, c(valid_plots, ncol = 3))
} else {
  warning("No valid plots to arrange.")
}
```

РОЗДІЛ 8. ПРОГРАМИ ДЛЯ СТАТИСТИЧНОЇ ОБРОБКИ ДАНИХ

Які програми варто використовувати для статистичної обробки даних? Це питання, мабуть, виникає перед кожним дослідником, бо в наш час є досить багато таких програм як платних, так і доступних на безкоштовній основі. Для різного роду статистичного аналізу можна використовувати стандартні пакети, що входять до складу електронних таблиць (*Excel, Quattro Pro*), математичні пакети загального призначення (*Mathcad, Maple*) і спеціалізовані програмні продукти (*СПП*). Міжнародний ринок нараховує більш, ніж 1000 пакетів, які здатні вирішувати задачі статистичного аналізу даних в середовищі операційних систем Windows, Linux, Macintosh тощо. Ми зупинимось на характеристиці тільки найвідоміших СПП, які, на нашу думку, заслуговують на особливу увагу.

R є потужною мовою програмування і середовищем для статистичного аналізу та обробки даних. Ось кілька корисних функцій R, які можуть бути використані для статистичної обробки даних: *read.csv():* функція для завантаження даних з CSV файлу; *summary():* використовується для отримання статистичної інформації про числові змінні, такі як середнє значення, мінімум, максимум, медіана тощо; *str():* показує типи даних та кількість спостережень у кожному стовпці; *aggregate():* дозволяє проводити агрегацію даних, наприклад, обчислення суми, середнього арифметичного тощо, групуючи їх за певними категоріями; *ggplot2:* пакет для візуалізації даних за допомогою граматики графіків, що

дозволяє швидко створювати красиві та інформативні графіки; *dplyr:* пакет, який надає функції для роботи з даними, такі як *filter()* для фільтрації даних, *mutate()* для додавання нових змінних, або summarize() для створення зведених таблиць; *tidyr:* цей пакет допомагає перетворювати та організовувати дані у форматі «довгий» (long) або «широкий» (wide) за допомогою функцій *gather()* і *spread()*; *lm():* використовується для побудови моделей лінійної регресії, дозволяючи оцінити параметри моделі та зробити прогнози на основі вхідних змінних; *t.test():* функція для проведення стюдентівського t-тесту для порівняння середніх двох груп; *cor():* дозволяє обчислити матрицю кореляцій між числовими змінними. Це лише декілька прикладів корисних функцій в **R** для статистичної обробки даних. **R** має багато додаткових пакетів та функцій, які дозволяють виконувати різноманітний аналіз та візуалізацію даних, залежно від потреб дослідника.

Python має багато корисних функцій для статистичної обробки даних. Ось декілька з них:

NumPy: NumPy – це бібліотека для обчислення чисельних операцій з масивами. Вона має багато статистичних функцій, таких як обчислення середнього, медіани, дисперсії, кореляції тощо.

Pandas: Pandas – це бібліотека для аналізу даних. Вона має зручні функції для групування, агрегації, сортування, фільтрації та інших операцій над даними.

SciPy: SciPy – це бібліотека для наукових обчислень, яка містить багато функцій для статистичного аналізу, включаючи розподіли ймовірностей, статистичні тести, оптимізацію та інші.

StatsModels: StatsModels – це бібліотека для статистичного моделювання, яка дозволяє виконувати регресійний аналіз, аналіз дисперсії, часові ряди та інші статистичні методи.

Matplotlib і Seaborn: Matplotlib і Seaborn – це бібліотеки для візуалізації даних. Вони дозволяють побудувати графіки, діаграми розподілу, розсіювання та інші статистичні візуалізації.

Scikit-learn: Scikit-learn – це бібліотека для машинного навчання, але вона також містить функції для роботи з даними, такі як розподіл даних на тренувальний і тестовий набори, нормалізація, кодування категоріальних змінних тощо.

SciPy.stats: модуль scipy.stats містить розподіли ймовірностей, функції для обчислення статистичних показників та статистичних тестів.

Math: вбудований модуль math містить різноманітні математичні функції такі, як обчислення логарифмів, експоненти, факторіалів тощо.

Collections: вбудований модуль collections містить класи, які допомагають обробляти контейнери даних, накшталт Counter для підрахунку елементів або defaultdict для створення словників із значеннями.

Ці функції, разом з іншими, надають широкий спектр інструментів для статистичної обробки даних у **Python**. Залежно від вашої потреби ви можете використовувати різні бібліотеки і модулі для виконання різних завдань статистичного аналізу.

IBM SPSS (Statistical Package for Social Science). Один з найпопулярніших статистичних пакетів (https://www.ibm.com/spss).

Відрізняється гнучкістю, потужністю, його можна застосовувати для всіх видів статистичних обчислень, які використовуються в біомедицині. Недавно вийшла 28-а англомовна версія. SPSS має зручні графічні засоби (більш, ніж 50 типів діаграм), а також розвинуті засоби підготовки звітів. Аналітичні параметри показуються на екрані у вигляді простих і зрозумілих меню і діалогових вікон. Нова контекстно-орієнтована довідкова система містить покрокові інструкції для найважливіших операцій.

STATGRAPHICS. Ця програма включає більш, ніж 250 статистичних процедур. Кожній групі процедур відповідає власне меню. Результати виводяться в табличній формі або на зручних для відтворення графіках. Остання версія програми V19 збагачена діалоговою системою вводу даних і вибору методів аналізу (сайт http://www.statgraphics.com). Модуль Statistical Advisor, що коротко пояснює суть будь-якого проведеного аналізу, допомагає в інтерпретації результатів. Окрім того, ця програма містить в собі порадник, який пояснює, що означає та чи інша обчислена величина і що потрібно дослідникові робити далі. Таким чином, STATGRAPHICS є достатньо корисним програмним продуктом, доступним для молодого дослідника**.**

SIGMAPLOT. Остання 16.0 версія програми має непоганий інтуїтивно зрозумілий інтерфейс. Окрім того на сайті програми https://grafiti.com/ подана не тільки демо-версія цієї програми, але й інструкція для використання цієї програми у форматі PDF, і досить показова відеопрезентація.

MINITAB. Статистичний пакет MINITAB в даний час випускається у версії 16. Із сайту виробника http://www.minitab.com можна взяти повнофункціональний пробний варіант програми, який працює 30 днів. Це достатньо зручний у роботі програмний пакет, що має зрозумілий інтерфейс, досить добрі можливості з візуалізації результатів роботи, має детальну довідку. MINITAB має добре продуманий розділ описової статистики, який керується за допомогою зручного меню. Команди, які часто використовуються, можна запустити, набравши першу букву їхньої назви.

PRISM. Ця програма створена спеціально для біомедичних досліджень і містить основні статистичні функції, що часто використовуються. На сайті http://www.graphpad.com крім можливості завантажити демо-версію Prism, можна отримати довідник у форматі PDF по біомедичній статистиці.

Безкоштовна програма – MYNOVA. Це досить спеціалізована безкоштовна програма для біомедичних досліджень. Вона виділяється зрозумілим інтерфейсом і простотою обчислень, дуже проста у використанні. Проте ця програма не може конкурувати із такими славнозвісними спеціалізованими пакетами, як IBM SPSS чи STATGRAPHICS.

Безкоштовна програма – STATOPTIMA. Це спеціалізована безкоштовна програма для біомедичних досліджень, а саме для обчислень у малих вибірках. Цю програму розробив один із авторів цього посібника для поширення серед молодих дослідників та покращенню розуміння статистичного аналізу. StatOptima містить статистичні тести і коди до них, які були розглянуті у цьому

посібнику. Відповідно до законодавства України ця програма була сертифікована і захищена авторським правом.

За відповідним QR-кодом  вона додається до посібника.

На якій же програмі зосередити свою увагу? Перш за все потрібно завантажити їх демо-версії, а потім вже робити свій вибір.

Що стосується можливих рекомендацій, то вони наступні:

- ***ми рекомендуємо всі проміжні обчислення проводити в програмі MS Excel або LibreOffice Calc***, де не тільки зосереджений потужний функціонал, але й ці програми дають змогу досліднику з легкістю вводити свої формули і зберігати їх для подальшого використання;

- якщо потрібний загальноприйнятий професійний пакет із потужним функціоналом, то можна скористатись програмами R, Python, IBM SPSS або STATGRAPHICS;

- для користувачів, що у своїх дослідженнях застосовують стандартні статистичні методи, можна рекомендувати безкоштовну програму STATOPTIMA чи MYNOVA.

УЗАГАЛЬНЕННЯ

Розглянувши статистичні показники, які необхідні для правильного опису експериментальних даних, на завершення нами складена узагальнювальна таблиця (алгоритм) послідовності етапів статистичної обробки даних, застосування певних формул, критеріїв чи методів для цих обчислень (табл. 22). Ця таблиця не є панацеєю статистичної обробки даних, проте, на нашу думку, вона допоможе досліднику в осмисленні послідовності етапів обробки даних.

Таблиця 22. Етапи статистичної обробки даних

Етап роботи	**Показник, який потрібно визначити**	**Формула, критерій або метод для використання**	**Ст.**
I. ХАРАКТЕРИСТИКИ ВИБІРКИ			
1	$\bar{x}$ (M)	$\bar{x} = \frac{\sum x_i}{n}$	**29**
2	s	$s = \sqrt{\frac{\sum (x_i - \bar{x})^2}{n-1}}$ **або** $s = \sqrt{\frac{\sum x_i^{\,2} - \frac{(\sum x_i)^2}{n}}{n-1}}$	**46**
3	m	$m = \frac{s}{\sqrt{n}}$ **або** $m = \sqrt{\frac{n\sum x_i^{\,2} - (\sum x_i)^2}{n^2(n-1)}}$	**53**
4	*Промахи*	Критерій Шовене Q-критерій Діксона Ізоляційний ліс	**66** **72** **79**
За наявності промахів у вибірці, їх викидаємо і повторюємо обчислення за пунктами 1-3			
6	*Нормальний розподіл даних при n≥10*	$\bar{X} \approx Me$ Критерій Андерсона–Дарлінга Критерій Шапіро-Вілка Коефіцієнт асиметрії та ексцесу	**86** **86** **89** **92**
Якщо масив даних істотно відрізняється від нормального розподілу, то для опису даних потрібно використати **медіану** або **збільшити кількість повторів**			
7	Δx *при* $p<0,05$	$\Delta x = t_P(f) \times m$	**62**

Результати обчислень записуємо у вигляді $\bar{x} \pm s$ або $\bar{x} \pm m$ або $\bar{x} \pm \Delta x$			
II. ПОРІВНЯННЯ ВИБІРОК МІЖ СОБОЮ			
8	*Однорідність дисперсій* (тільки при порівнянні двох вибірок між собою)	Критерій Кокрена G Критерій Фішера F	**111** **112**
9	*Дві незалежні вибірки*	t-критерій Стьюдента для незалежних вибірок *U*-критерій Манна–Вітні	**120** **133**
		або	
9	*Дві залежні вибірки*	Парний критерій Стьюдента *W*-критерій Вілкоксона	**121** **138**
		або	
9	*Більше двох вибірок, які порівнюються між собою*	Критерій Ньюмена-Коулса Критерій Тюкі Критерій Дункана Критерій Данна	**142** **148** **152** **168**
		або	
9	*Більше двох вибірок, які порівнюються із контрольною вибіркою*	Критерій Даннета Критерій Данна	**159** **168**
III. ВЗАЄМОЗВ'ЯЗКИ МІЖ ГРУПАМИ: КОРЕЛЯЦІЙНО-РЕГРЕСІЙНИЙ АНАЛІЗ			
9	*Коефіцієнти рівнянь регресії*	Метод МНК / Метод найменших квадратів	**183**
10	$\bar{\varepsilon}$	$\bar{\varepsilon} = \frac{1}{n} \cdot \sum \frac{\lvert y_i - y_x \rvert}{y_i} \cdot 100\%$	**188**
11	η	$\eta = \sqrt{1 - \frac{\sum (y_i - y_x)^2}{\sum (y_i - \bar{y})^2}}$	**189**
12	S_y^2	$S_y^2 = (\Sigma y_i^2 - ((\Sigma y_i)^2 / n)) / n - 1$	**189**
13	$S^2_{зал}$	$S^2_{зал} = \Sigma (y_i - y_x)^2 / (n - 2)$	**189**
14	F	$F = S_y^2 / S^2_{зал}$	**189**
15	δ	$\delta = \pm \sqrt{1/n \sum (y_i - y_x)^2}$	**189**
16	Δ	$\Delta = \pm \sqrt{(1/n) \sum ((y_i - y_x)/y_x)^2} \times 100$	**189**
17	o_p	$o_p = (1/n) \Sigma ((y_i - y_x)/y_x) \times 100$	**190**
18	o_δ	$o_\delta = \pm \sqrt{(1/n) \sum \{((y_i - y_x)/y_x) \times 100 - o_p\}^2}$	**190**
На основі фактичних значень *F*-критерію Фішера, похибок та помилок робимо загальний висновок про адекватність того чи іншого рівняння регресії і використовуємо його для опису даних			

ЗАВДАННЯ ДЛЯ САМОСТІЙНОГО ВИКОНАННЯ

Завдання 1.

Розрахуйте середнє арифметичне і стандартну помилку середньої арифметичної величини для вибірки зі значеннями 24, 27, 32, 30, 35, 28, 30.

Завдання 2.

Обчислити середнє арифметичне, дисперсію, стандартне відхилення, стандартну похибку середнього для вибірок зі значеннями:

Вибірка **X:** 34, 45, 25, 56, 31, 29, 49

Вибірка **У:** 106, 118, 96, 105, 122, 99, 94.

Завдання 3.

Розрахувати середнє арифметичне, медіану, стандартне відхилення, стандартну похибку середнього, довірчий інтервал для середнього арифметичного (p = 0.025, нормальний розподіл) для наступних показників:

1. Контроль – 25, 28, 31, 21, 35, 17. Дослід – 35, 39, 31, 41, 29, 43, 49, 26.

2. Контроль – 0,74, 0,95, 0,65, 0,82, 0,51. Дослід – 0,54, 0,32, 0,41, 0,67.

Завдання 4.

Питома активність фосфофруктокінази в нирках чотирьох коропів, виловлених в ставку села Сорочинці становила 41, 36, 32 та 27 міліодиниць/мг білка. Активність цього ж ферменту, в нирках коропів, виловлених зі ставка села Крипаківці становила 14, 25, 6 і 19 міліодиниць/мг білка.

Обрахуйте для цих даних середні арифметичні значення, стандартні помилки середньої арифметичної величини, 10-ий, 25-ий, 50-ий, 75-ий та 90-ий процентилі. Намалюйте за результатами спостереження графік, який має унаочнювати відмінність між активністю фосфофруктокінази в нирках сорочинських та крипаківських коропів.

Завдання 5.

За допомогою методу А дослідникам у 6 незалежних експериментах вдалось синтезувати у безклітинній системі синтезу білка поліпептидні ланцюги завдовжки 48, 55, 37, 41, 53 та 45 амінокислот, відповідно. За допомогою методу Б в подібній серії експериментів вдалось синтезувати ланцюги завдовжки 23, 42, 36, 28, 32 та 25 амінокислот.

Знайдіть:

а) скільки амінокислот в середньому буде містити поліпетидний ланцюг при синтезі методом А і методом Б (результат округліть до цілих);

б) медіани, дисперсії, стандартні відхилення, стандартні похибки середнього, перший та третій квартилі для двох наборів даних (округлення – до тисячних).

Дайте відповідь на запитання:

1). У скільки разів метод А є ефективнішим за метод Б?

2). На скільки відсотків метод А є ефективнішим за метод Б?

3). Скільки відсотків від середнього значення становить його стандартна похибка для кожного набору даних?

4). У скільки разів і на скільки відсотків відрізняються значення першого квартилю набору даних для методу А від такого для методу Б?

Завдання 6.

У зразках А, Б і В визначали рівень карбонільних груп білків. У низці повторів отримали наступні набори значень: для зразка А – 1,26; 1,34; 1,42 та 1,29 нмоль динітрофенілгідразонів/мг білка; для зразка Б – 1,45; 1,53; 1,49; 1,58; 1,41; 1,61 та 1,43 нмоль динітрофенілгідразонів/мг білка; для зразка В – 1,74; 1,63; 1,54; 1,47 та 1,68 нмоль динітрофенілгідразонів/мг білка.

Знайдіть:

а) середнє, стандартне відхилення, стандартну похибку середнього, медіану, перший та третій квартиль для трьох зразків;

б) на скільки відсотків відрізняються середні значення зразків Б і В від середнього для зразка А.

Побудуйте графік (у Microsoft Excel), де будуть показані середні та межі, в яких знаходиться більшість значень у генеральній сукупності (стандартні відхилення) для кожного зразка.

Завдання 7.

Вміст глутатіону в чотирьох контрольних культурах клітин лінії HepG2 становив 13, 3, 7 та 9 пмоль/клітину. В 3-ох культурах клітин лінії HEK293T цей вміст становив 7, 14, та 24 пмоль/клітину. В 5-ти культурах лінії 3T3 цей вміст був рівний 1, 2, 7, 3 і 12 пмоль/клітину. Обрахуйте для цих даних середні арифметичні значення, стандартні помилки середньої арифметичної величини, 10-ий, 25-ий, 50-ий, 75-ий та 90-ий процентилі. Намалюйте за результатами спостережень графік, який має унаочнювати вміст глутатіону в різних лініях культивованих клітин.

Завдання 8.

При оцінці рухової активності плодової мушки (ґрунтується на негативному геотаксисі – рефлекторному русі тварини вгору («від землі») одразу після струшування додолу) у 5 незалежних повторах були отримані наступні дані для двох ліній (у секундах): 2, 75, 79, 82, 5 – для лінії X; 14, 17, 98, 21, 94 – для лінії Y.

Наскільки відрізняються середні значення рухової активності двох ліній?

Наскільки відрізняються медіани рухової активності двох ліній?

Напишіть алгоритм побудови в Microsoft Excel графіка, на якому можна було б відобразити значення всіх повторів для двох ліній і, одночасно, медіану та середнє значення.

Намалюйте такий графік у зошиті.

Завдання 9.

Вчені дослідили вплив різних концентрацій фенобарбіталу (ФБ) на питому активність глутатіон-S-трансферази у комарів *Aedesaegypti*, які переносять тропічні хвороби. В першому повторі досліду контроль (без обробки фенобарбіталом) мав 59 одиниць/мг білка; особини, оброблені 0,1 мкмоль/л ФБ, мали 45 одиниць/мг білка; 0,25 мкмоль/л – 37; 0,75 мкмоль/л – 31; 1,00 мкмоль/л – 31 одинця/мг білка. Другий повтор: 0,00 – 54; 0,10 – 40; 0,25 – 32; 0,75 – 26; 1,00 – 24 одиниці/мг білка. Третій повтор: 0,00 – 49; 0,10 – 35; 0,25 – 27; 0,75 – 21; 1,00 – 17 одиниць/мг білка.

Унаочніть дані трьох експериментів у одному графіку так, щоб продемонструвати залежність активності глутатіон-S-трансферази комарів *Aedesaegypti* від різних концентрацій фенобарбіталу.

Завдання 10.

Знайдіть середнє, стандартне відхилення, медіану, 25-й і 75-й процентилі для наступних даних: 291, 204, 360, 245, 234, 212, 253, 248, 226, 241, 223, 214. Чи можна вважати, що вибірка взята із сукупності з нормальним розподілом? Обґрунтуйте свою відповідь.

Завдання 11.

Концентрація молочної кислоти в крові чотирьох однорічних самців мишей лінії C57BL/6J становила 49, 69, 51 та 63 мкмоль/л. Для трьох самців лінії BALB/c ця концентрація становила 34, 46, та 58 мкмоль/л. У 6-ти мишей лінії MRL цей показник дорівнював 25, 29, 31, 36 і 19 мкмоль/л.

Обрахуйте для цих даних середні арифметичні значення і довірчі інтервали (розподіл Стьюдента, $\alpha = 0.025$). Намалюйте за результатами спостережень графік, який має унаочнювати концентрацію молочної кислоти в крові однорічних мишей (самців), а також варіативність цього показника.

Завдання 12.

Дослідник провів 5 незалежних повторів досліду і виявив, що в 1-ому повторі за присутності чинника N в середовищі для плодової мушки зі 178 лялечок вилупилось 108 дорослих особин, в 2-ому повторі зі 102 – 69, в 3-ому з 215 – 125, в 4-ому з 93– 62, в 5-ому зі 147 – 89. За відсутності чинника з усіх лялечок завжди вилуплювались дорослі особини.

Напишіть рекомендацію (з обрахунками) щодо того, як порівняти отримані дані та як було б найкраще, на вашу думку, їх представити. Намалюйте ваш варіант графіка.

Завдання 13.

Під час досліджень довжини десяти паростків лохини були отримані такі результати (см): 11,1; 12,1; 13,2; 12,9; 14,5; 13,7; 12,5; 9,8; 12,5;

12,8. Обчисліть середнє арифметичне значення та показники варіації вибірки.

Завдання 14.

Відомо, що для здорової людини pH крові є нормально розподіленою величиною із середнім значенням 7,4 і стандартним відхиленням 0,2. Визначте діапазон значень цього параметра (використайте правило трьох сигм).

Завдання 15.

Середній зріст немовлят в нормально розподіленій популяції новонароджених становить 51,4 см. За даними вибірки, наданої одним з пологових будинків, середній зріст новонароджених хлопчиків становить 51,8 см, вибіркова дисперсія 2,1 $см^2$, обсяг вибірки $n = 25$.

Чи можна припустити, що середній зріст в популяції новонароджених хлопчиків більший ніж 51,4 см при рівні значущості $\alpha = 0,05$.

Завдання 16.

Дослідниця спостерігала за темпом утворення лялечок лінії Dahomey («дагомі») *Drosophila melanogaster* на стандартному поживному середовищі. Лялечки почали утворюватися на 4-ту добу після відкладання яєць. Так, на 4-ту добу на стінках бутлі з'явилося 5 лялечок, на 5-ту – додалося ще 10; на 6-ту – 24, на 7-му – 56, на 8-му

– 12, на 9-ту – 5, а на 10-ту добу кількість лялечок вже не змінювалась.

За скільки діб, в середньому, заляльковуються особини лінії Dahomey?

В яких межах коливається цей час?

Намалюйте графік залежності лялькування від часу у вигляді: а) кривої за отриманими значеннями (без додаткових підрахунків) та б) накопичувальної (кумулятивної) кривої.

Завдання 17.

Студенти вивчили дію різних концентрацій хлориду алюмінію на вміст заліза в дафніях *Daphnia pulex*. В першому повторі досліду в ємностях, де було 0,001 ммоль/л $AlCl_3$, дафнії містили 10 пг Fe/особину; 0,01 ммоль/л –14 пг Fe/особину; 0,1ммоль/л – 32 пг Fe/особину; 1ммоль/л – 31 пг Fe/особину; 10ммоль/л – 17 пг Fe/особину, 100 ммоль/л – 5пг Fe/особину. Другий повтор: 0,001 – 20; 0,01 – 34; 0,1 – 45; 1 – 33; 10 – 17; 100 – 8 пг Fe/особину. Третій повтор: 0,001 – 27; 0,01 – 21; 0,1 – 52; 1 – 38; 10 – 11; 100 – 8 пг Fe/особину.

Унаочніть дані трьох експериментів у одному графіку так, щоб продемонструвати залежність вмісту заліза в *Daphnia pulex* від дію різних концентрацій хлориду алюмінію.

Завдання 18.

В контрольній групі плодових мушок, яких вирощували на середовищі з 2,5% сахарози та 5% дріжджів на 150-ту годину

заляльковувалось 100% яєць в чотирьох повторах досліду. З тих особин, яких вирощували на середовищі з 15% сахарози і 5% дріжджів, на 150 годину заляльковувалось 56% всіх личинок в першому повторі, 49% – в другому, 14% – в третьому і четвертому, 29% – у п'ятому і 19% – у шостому. З тих особин, яких вирощували на середовищі з 10% сахарози і 5% дріжджів, на 150 годину заляльковувалось 34% всіх личинок в першому повторі, 48% – в другому і третьому, 47% – в четвертому і 29% – у п'ятому.

Намалюйте за результатами спостереження графік, який має демонструвати вплив співвідношення між компонентами дієти на швидкість лялькування. Покажіть на графіку медіани.

Завдання 19.

В результаті експерименту отримані наступні значення величин: 3, 6, 8, 11, 6, 10, 7, 9, 7, 3, 4, 8, 2, 7, 9, 4, 9, 11, 7, 8, 4, 10, 5, 6, 7. Побудуйте полігон частот і статистичну функцію розподілу. Знайдіть середню арифметичну величину, дисперсію, моду і медіану.

Завдання 20.

Довжина тіла у 30 особин карася сріблястого (в см): 14,6; 14,6; 13,1; 13,3; 14,6; 11,2; 16,0; 12,3; 12,2; 14,8; 14,1; 11,2; 15,1; 14,7; 12,3; 14,3; 14,3; 15,1; 15,6; 13,3; 14,1; 15,3; 13,8; 13,9; 15,3.

Побудуйте варіаційний ряд і полігон розподілу, знайдіть середнє арифметичне, варіаційний розмах, дисперсію, стандартне відхилення, коефіцієнт варіації, коефіцієнт асиметрії і медіану.

Завдання 21.

Використавши критерії Шовене, Діксона, Романовського і Ірвіна, перевірте наступні дані: 12,6; 13,2; 14,8; 28,6; 12,1 на наявність промахів.

Завдання 22.

Під час досліджень довжини десяти паростків лохини були отримані такі результати (см): 13,2; 12,2; 11,1; 12,9; 15,4; 12,7; 12,2; 9,6; 11,5; 27,8. Перевірте вибірку на нормальність розподілу та наявність промахів.

Завдання 23.

Було вивчено загальний вміст азоту в плазмі крові мишей у віці 37 і 180 днів.

Результати виражені в грамах на 100 $см^3$ плазми.

У віці 37 днів: 0,98; 0,83; 0,99; 0,86; 0,90; 0,81; 0,94; 0,92; 0,87.

У віці 180 днів: 1,20; 1,18; 1,33; 1,21; 1,20; 1,07; 1,13; 1,12.

Встановіть ймовірність відмінностей між вибірками.

Завдання 24.

Відомо дані про активність амілази в слюні дітей (група 1) і людей старечого віку (група 2). Група 1 – 34,4; 85,0; 96,2; 102; 103; 63,2; 69,1; 83,1; група 2 – 5,01; 2,12; 11,1; 8,01; 4,02; 2,03; 6,03; 8,04. Враховуючи, що дані розподілені за нормальним законом, перевірте гіпотезу про рівність дисперсій.

Завдання 25.

В результаті експерименту отримали дані, які не розподілені за нормальним законом. Перевірте гіпотезу про рівність дисперсій.

Група 1: 281 232 114 177 91 176 218 112 201 116

Група 2: 201 127 226 211 261 195 157 241 171 234

Завдання 26.

Порівняйте за тестом Стьюдента значення активності гліцеральдегід-3-фосфатдегідрогенази в нирках мишей в контрольній та дослідній (схильність до утворення карциноми нирок) групах, отримані у 5 незалежних експериментах.

Контроль: 131, 130, 120, 167, 133 мкмоль утвореного НАДН за хвилину на мг білка.

Дослід: 156, 164, 177, 164, 176 мкмоль утвореного НАДН за хвилину на мг білка.

Обчисліть середні арифметичні значення, стандартні помилки середньої арифметичної величини, стандартне відхилення, медіани та значення *p* (згідно з тестом). Намалюйте графік, який демонструє результати експерименту з відображеним статистичним аналізом (стандартна похибка середнього, позначення різниці, якщо різниця дійсно значуща).

Завдання 27.

При дослідженні розмірів плодів ожини сорту «Натчез» в двох вибірках з різних місць були отримані наступні дані (в мм):

перша вибірка – 33, 35, 34, 36, 38, 33, 34, 35, 33, 32, 35;

друга вибірка – 35, 36, 33, 35, 35, 36, 33, 32, 30, 35, 33.

Визначте за критерієм Стьюдента – чи відрізняються ці вибірки, або ж вони належать до однієї сукупності?

Використовувати рівень значущості $\alpha = 0{,}05$.

Завдання 28.

Порівняйте вплив препарату на медіанну тривалість життя шести популяцій плодової мушки:

Контроль: 34, 32, 44, 24, 33, 42 доби.

Препарат А: 28, 27, 25, 26, 24, 31 доба.

Обчисліть середні значення, стандартні помилки середньої арифметичної величини, медіани та значення *p* (згідно з тестом Стьюдента). На основі цього намалюйте графік.

Завдання 29.

Оцініть ефективність препарату, який підвищує витривалість (відстань, яку можна подолати без перепочинку) п’ятьох спортсменів-бігунів. До вживання препарату:

Спортсмен A – 23 км, N – 35 км, F – 33 км, G – 21 км, W – 36 км.

Через місяць після початку вживання препарату та відсутності додаткових тренувань:

Спортсмен A – 25 км, N – 39 км, F – 34 км, G – 30 км, W – 43 км.

Обчисліть середні арифметичні значення для кожної групи, стандартні помилки середньої арифметичної величини, значення *p* (згідно з тестом Стьюдента). Дайте відповідь на запитання: чи

істотно і на скільки відсотків препарат змінює витривалість спортсменів-бігунів.

Завдання 30.

В експерименті оцінювалась дія двох речовин на фермент амілазу. Результати для проб наступні:

Активність амілази без дії будь-яких речовин (контрольна група): проба А – 68, проба Б – 54, проба В – 63, проба Г – 57 Од./мг білка.

Речовина Х: проба А – 79, проба Б – 67, проба В – 81, проба Г – 73 Од./мг білка.

Речовина Y: проба А – 44, проба Б – 50, проба В – 52, проба Г – 49 Од./мг білка.

Обчисліть середні арифметичні значення для контрольної та експериментальних груп, стандартні помилки середньої арифметичної величини, значення *p* (згідно з тестом Стьюдента). Дайте відповідь на запитання: на скільки відсотків в середньому речовини змінюють активність амілази.

Завдання 31.

Порівняйте середнє значення вибірки А (значення – 0,119, 0,143, 0,135) із середнім значенням вибірки Б (значення – 0,154, 0,143, 0,139) за допомогою t-тесту Велча (модифікований тест Стьюдента). Для цього:

а) розрахуйте фактичне значення t-критерію;

б) знайдіть критичне значення критерію для рівня значущості 0,05 за допомогою програми Microsoft Excel або аналогічної (у Microsoft Excel це можна зробити за допомогою формули T.INV.2T);

в) порівняйте критичне і фактичне значення χ^2, і зробіть висновок про справдження або заперечення гіпотези H_0.

Додатково знайдіть точне значення p.

Завдання 32.

В результаті досліджень активності лактатдегідрогенази в печінці коропа отримали наступні дані: 107, 109, 115, 121, 125, 123, 147, 117, 117, 112 Од/мг білка. Перевірте дані за допомогою *складового критерію d, критерію Шапіро-Вілка, коефіцієнтів ексцесу та асиметрії* на нормальність їх розподілу.

Завдання 33.

Перевірте наявність значущих відмінностей між двома групами, використовуючи *U*-критерій Манна–Вітні за наступними даними:

Вибірка **X:** 68 63 69 75 61 60

Вибірка **Y:** 30 27 28 23 28 29

Завдання 34.

Розрахуйте *t* фактичне за формулою *t*-тесту Велча для порівняння двох вибірок: вибірка А має значення 35, 46, 39, 40. Вибірка Б має значення – 29, 23, 26, 26. Чи значущо відрізняються середні для двох вибірок, якщо *t* критичне для рівня значущості 0,05 та ступенів свободи 3 становить 3,183?

Завдання 35.

Перевірте наявність ймовірних відмінностей між двома групами, використовуючи *W*-критерій Вілкоксона, за наступними даними:

Вибірка **X:** 10,5; 10,9; 10,5; 10,7; 10,9

Вибірка **Y:** 16,9; 15,7; 15,2; 16,7; 15,8

Завдання 36.

Порівняйте середні арифметичні величини двох незалежних вибірок методом Стьюдента за рівнем значущості $\alpha = 0,05$.

Вибірка **X:** 10,5; 10,9; 10,5; 11,4; 11,9; 10,7; 10,9; 11,5; 12,8; 11,2; 12,3; 12,7

Вибірка **Y:** 11,7; 10,6; 16,9; 15,7; 11,7; 12,6; 15,2; 12,4; 12,4; 12,2; 16,7; 15,8

Завдання 37.

В результаті досліджень впливу йонів кобальту на активність каталази в печінці коропа було отримано наступні результати:

Групи риб	**Активність каталази, Од/мг білка**
Контроль	53,9; 51,5; 71,4; 88; 95,0; 88,9
5 мг/л Co^{2+}	70,5; 78,5; 80,2; 76,5; 78,7; 76,1
10 мг/л Co^{2+}	57,8; 43,4; 56,2; 63,8; 42,0; 58,0
20 мг/л Co^{2+}	41,4; 41,7; 38,9; 40,4; 27,3; 44,4

Порівняйте дослідні групи між собою та контрольною групою за допомогою критерію Ньюмена-Коулса.

Завдання 38.

Порівняйте дослідні групи із контрольною за критерієм Даннета (дані взяти із прикладу 34).

Завдання 39.

Порівняйте дослідні групи із контрольною за непараметричним критерієм Данна (дані взяти із прикладу 34).

Завдання 40.

Досліджувалася взаємозв'язок між висотою голови *x* і довжиною 3-го членика вусика в *Drosophila melanogaster*. Для цього за допомогою окуляр мікрометра отримані наступні дані по *x* і *y* (в розподілах окуляр-мікрометра). Що ви можете сказати про взаємозв'язок ознак?

X: 16 17 16 16 17 17 18 19 19 18 18 18 16 17 16 16 16 18 16 14 16 15 18 16 17

Y: 30 32 33 34 33 34 34 37 37 36 36 36 36 34 32 32 32 35 34 31 33 32 36 34 34

Завдання 41.

У харіуса були виміряні довжина голови *x* і довжина грудного плавця *y* у мм:

X: 68 63 69 75 53 61 50 49 60 45 43 56 54 43 49

Y: 40 33 38 45 31 35 30 27 38 28 23 32 30 28 29

Визначте коефіцієнт кореляції між *x* і *y*. Побудуйте лінійну регресію.

Завдання 42.

Значенням фактору/змінної X: 10, 25, 50, 100, 250, 500 відповідають значенням показника Y: 742, 636, 619, 415, 149, 37.

Знайдіть:

а) найкращу модель регресії для опису залежності Y від X;

б) коефіцієнти рівняння регресії згідно із запропонованою моделлю;

в) коефіцієнт детермінації для запропонованої моделі;

г) передбачене значення Y для значення X – 184.

Завдання 43.

За наведеними даними вкажіть на наявність чи відсутність взаємозв'язків між активністю АМФ-дезамінази *X* та вмістом ІМФ *Y* у білих м'язах карася сріблястого:

X: 6,28; 6,89; 7,34; 7,92; 8,26; 8,74; 8,39; 8,34; 8,74; 9,72; 14,0; 15,6; 17,7; 18,5.

Y: 5,11; 5,82; 6,96; 7,39; 7,07; 7,73; 7,81; 7,56; 8,00; 8,45; 8,77; 9,01; 9,13; 9,45.

Завдання 44.

Порівняйте середні арифметичні величини двох залежних вибірок методом Стьюдента за рівнем значущості $\alpha = 0,05$.

Вибірка **X:** 11,5; 11,9; 11,5; 12,4; 13,9; 11,7; 11,9; 11,5; 11,8; 11,4; 126; 13,7

Вибірка **Y:** 13,7; 13,6; 19,9; 18,7; 14,7; 15,6; 18,2; 15,4; 15,4; 15,2; 19,7; 18,8

Завдання 45.

Значенням фактору/змінної X: 20, 32, 60, 250, 450, 600 відповідають значенням показника Y: 40, 150, 421, 652, 684, 725.

Знайдіть:

а) найкращу модель регресії для опису залежності Y від X;

б) коефіцієнти рівняння регресії згідно із запропонованою моделлю;

в) коефіцієнт детермінації для запропонованої моделі.

Завдання 46.

Порівняйте середні арифметичні величини двох залежних вибірок методом Стьюдента за рівнем значущості $\alpha = 0,05$.

Вибірка **X:** 12,3; 11,9; 10,8; 11,4; 11,2; 10,3; 11,9; 11,2; 12,9; 12,2; 12,4; 12,1

Вибірка **Y:** 21,7; 20,1; 26,9; 25,1; 21,3; 22,4; 25,1; 22,4; 22,3; 22,2; 26,7; 25,4

Завдання 47.

Під час досліджень довжини дванадцяти паростків ожини, які були вирощені методом мікроклонального розмноження рослин, отримали наступні дані (см): 3,2; 2,2; 2,1; 2,9; 5,4; 2,7; 2,2; 4,6; 2,5; 2,8. Перевірте вибірку на нормальність розподілу та наявність промахів.

Завдання 48.

В результаті експерименту отримали наступні значення величин: 4, 8, 10, 6, 10, 7, 9, 7, 3, 4, 8, 2, 7, 9, 4, 9, 11, 7. Побудуйте полігон частот

і статистичну функцію розподілу. Знайдіть середню арифметичну величину, дисперсію, моду і медіану.

Завдання 49.

В результаті експерименту отримали наступні активності супероксиддисмутази: 88, 101, 52, 102, 115, 114, 112, 78, 74, 68, 45, 188, 142, 142, 145. Перевірте дані на наявність промахів, побудуйте полігон частот і статистичну функцію розподілу. Знайдіть середню арифметичну величину, дисперсію, моду і медіану.

Завдання 50.

В результаті експерименту отримали наступні активності каталази: 188, 121, 55, 102, 117, 114, 112, 80, 74, 75, 45, 198, 142, 143, 143. Перевірте дані на наявність промахів, побудуйте полігон частот і статистичну функцію розподілу. Знайдіть середню арифметичну величину, дисперсію, моду і медіану.

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

Українською

Майборода Р. Комп'ютерна статистика: підручник. К.: ВПЦ «Київський університет», 2019.

Кальченко В.В., Мурашковська В.П., Ткач Ю.М. Математичні обчислення засобами пакету R — програмування. Чернігів: ЧНТУ, 2017.

Загальна статистика, моделювання

Agresti A. An Introduction to Categorical Data Analysis (3rd ed.). Wiley, 2018.

Agresti A. Categorical Data Analysis (3rd ed.). Wiley, 2013.

Efron B., Hastie T. Computer Age Statistical Inference. CUP, 2016.

Faraway J.J. Extending the Linear Model with R (2nd ed.). Chapman & Hall/CRC, 2016.

Gibbons J.D., Chakraborti S. Nonparametric Statistical Inference (5th ed.). CRC Press, 2010.

Hastie T., Tibshirani R., Friedman J. The Elements of Statistical Learning (2nd ed.). Springer, 2009.

James G., Witten D., Hastie T., Tibshirani R. An Introduction to Statistical Learning (2nd ed.). Springer, 2021.

McElreath R. Statistical Rethinking (3rd ed.). CRC Press, 2023.

Gelman A. et al. Bayesian Data Analysis (3rd ed.). Chapman & Hall/CRC, 2013.

Burnham K.P., Anderson D.R. Model Selection and Multimodel Inference (2nd ed.). Springer, 2002.

Lang T. “Twenty Statistical Errors Even YOU Can Find in Biomedical Research Articles”, Croat Med J, 2004.

Біостатистика / медицина / епідеміологія

Daniel W.W., Cross C.L. Biostatistics: A Foundation for Analysis in the Health Sciences (11th ed.). Wiley, 2018.

Rosner B. Fundamentals of Biostatistics (9th ed.). Cengage, 2022.

Motulsky H. Intuitive Biostatistics (4th ed.). OUP, 2017.

Bland M. An Introduction to Medical Statistics (4th ed.). OUP, 2015.

Lash T.L., VanderWeele T.J., Haneuse S., Rothman K.J. Modern Epidemiology (4th ed.). Wolters Kluwer, 2021.

Chow S.-C., Liu J.P. Design and Analysis of Clinical Trials (3rd ed.). Wiley, 2014.

R

R Core Team. R: A Language and Environment for Statistical Computing. R Foundation, 2024.

Wickham H. Advanced R (2nd ed.). Chapman & Hall/CRC, 2019.

Wickham H., Grolemund G. R for Data Science (2nd ed.). O’Reilly, 2023.

Python / інструменти аналізу даних

McKinney W. Python for Data Analysis (3rd ed.). O’Reilly, 2022.

VanderPlas J. Python Data Science Handbook (2nd ed.). O’Reilly, 2023.

Géron A. Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow (3rd ed.). O'Reilly, 2022.

Langtangen H.P. A Primer on Scientific Programming with Python (5th ed.). Springer, 2016.

Downey A.B. Think Stats (2nd ed.). O'Reilly, 2014.

Література прикладного характеру

Sokal R.R., Rohlf F.J. Biometry (4th ed.). W.H. Freeman, 2012.

Zuur A.F., Ieno E.N., Walker N., Saveliev A.A., Smith G.M. Mixed Effects Models and Extensions in Ecology with R. Springer, 2009.

Навчальне видання

Гусак Віктор Васильович
Господарьов Дмитро Валерійович
Лущак Володимир Іванович

СТАТИСТИКА МАЛИХ ВИБІРОК У БІОЛОГІЇ І МЕДИЦИНІ З ОСНОВАМИ ПРОГРАМУВАННЯ В *PYTHON* І *R*

НАВЧАЛЬНИЙ ПОСІБНИК
Друге видання

Редактор Гусак В. В.

Віддруковано згідно з наданим оригінал-макетом замовника

Підписано до друку 04.12.2025 р. Формат 60x84/16
Гарнітура Times New Roman. Умовн. друк. арк. 15,23
Наклад 100 прим. Зам. №4 від 04.12.2025 р.

Видавець і виготовлювач
ФОП Голіней О. В.
м. Івано-Франківськ, вул. Галицька 128
+380664816601, gsm1502@ukr.net

Свідоцтво про внесення до Державного реєстру видавців
ДК №7970 від 20.10.2023

ISBN 978-617-8470-47-0