МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ДНІПРОВСЬКА ПОЛІТЕХНІКА»

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра системного аналізу та управління

К.С. Хабарлак, Т.В. Хом’як

АНАЛІЗ ДАНИХ ТА ЗНАНЬ

Конспект лекцій
для здобувачів ступеня бакалавра
зі спеціальності 124 Системний аналіз
(F4 Системний аналіз та наука про дані)

Дніпро
НТУ «ДП»
2025

Хабарлак К.С.

Аналіз даних та знань [Електронний ресурс] : конспект лекцій для здобувачів ступеня бакалавра спеціальності 124 Системний аналіз (F4 Системний аналіз та наука про дані) / уклад: К.С. Хабарлак, Т.В. Хом'як ; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – Дніпро : НТУ «ДП», 2025. – 112 с.

Укладачі:
К.С. Хабарлак, доктор філософії;
Т.В. Хом'як, канд. фіз.-мат. наук, доц.

Затверджено науково-методичною комісією зі спеціальності F4 Системний аналіз та наука про дані (протокол № 3 від 12.05.2025) за поданням кафедри системного аналізу та управління (протокол № 5 від 07.05.2025).

У конспекті лекцій з курсу «Аналіз даних та знань» подано теоретичні основи первинного аналізу, регресійних моделей та методів класифікації, A/B-тестування, перевірки значущості статистичних гіпотез, множинного тестування, методів зменшення розмірності. Матеріали лекційних занять наведено згідно з робочою програмою дисципліни «Аналіз даних та знань» спеціальності 124 Системний аналіз за першим (бакалаврським) рівнем вищої освіти.

Орієнтовано на активізацію навчальної діяльності здобувачів та закріплення практичних навичок у засвоєнні дисципліни «Аналіз даних та знань».

Відповідальний за випуск завідувач кафедри системного аналізу та управління Т.А. Желдак, канд. техн. наук, доц.

Зміст

Вступ

У сучасному світі, де люди, цифрові технології та пристрої Інтернету речей безперервно генерують неймовірні обсяги даних, аналіз даних перетворюється на важливу компетенцію, що визначає успіх бізнесу, наукових досліджень та суспільного розвитку. Крім того, аналіз даних стимулює інновації, сприяючи створенню нових продуктів, оптимізації логістичних ланцюгів та підвищенню економічної ефективності. Таким чином, аналіз даних не просто перетворює гігантські масиви інформації на корисні висновки, але й стає ключовим інструментом для формування конкурентоспроможної стратегії в умовах цифрової економіки. Сьогодні ця галузь активно розвивається, створюючи нові можливості для фахівців у сфері даних та підвищуючи попит на спеціалістів, які вміють поєднувати технічні навички з аналітичним мисленням.

У даному конспекті лекцій з курсу «Аналіз даних та знань» подано теоретичні основи аналізу даних. У межах курсу будуть розглянуті наступні теми:

1. Первинний аналіз даних.
2. Візуалізація даних.
3. Побудова та підгонка лінійної регресії.
4. Бутстрап. Довірчі інтервали.
5. Передбачення за допомогою регресії. Факторні змінні. Перехресний контроль.
6. A/B тестування.
7. Дисперсійний аналіз та багаторукий бандит.
8. Лінійні, метричні та ймовірнісні методи класифікації.
9. Методи зменшення розмірності даних.

Завдяки вивченню зазначених тем курсу, здобувач ознайомиться з теоретичними основами аналізу даних, включаючи первинний аналіз, візуалізацію, бутстрап, довірчі інтервали, побудову та підгонку лінійної регресії з факторними змінними, передбачення за допомогою регресії з використанням перехресного контролю, A/B-тестування, дисперсійний аналіз та методи множинного тестування, а також лінійні, метричні та ймовірнісні методи класифікації та методи зменшення розмірності даних. Це дозволить здобувачу збирати, оброблювати та аналізувати великі масиви даних; будувати, оцінювати та застосовувати регресійні та класифікаційні моделі для передбачення на нових, невідомих вхідних даних, а також ефективно використовувати методи зменшення розмірності для оптимізації аналітичних процесів.

Мета дисципліни – сформувати у здобувачів вищої освіти: 1) практичні навички попередньої обробки, аналізу та візуалізації даних провідними методами за допомогою мови програмування Python; 2) вміння будувати моделі машинного навчання, що відповідають задачі, та навчати їх; 3) здобути навички роботи із бібліотеками Python, зокрема scikit-learn, SciPy, Pandas, NumPy, Matplotlib для

машинного навчання та обробки даних. Знання та навички, отримані в курсі, будуть корисними для подальшого працевлаштування здобувача.

Завдання курсу:

- навчитися проводити первинний аналіз даних, розраховувати статистичні оцінки та візуалізувати дані за допомогою мови програмування Python;
- навчитись застосовувати машинне навчання, методи регресії та класифікації до практичних задач;
- отримати практичні навички проведення A/B тестування, множинного тестування та підтвердження статистичних гіпотез;
- опанувати роботу із мовою програмування Python для аналізу даних та із прикладними бібліотеками, scikit-learn, SciPy, Pandas, NumPy, Matplotlib.

Дисциплінарні результати навчання:

1. Вміти збирати та видобувати дані з різних джерел, структурувати та зберігати дані. Знати сучасні способи розвідувального та первинного аналізу даних, вміти застосовувати їх. Способи відбору даних.
2. Вміти розраховувати статистичні оцінки, візуалізувати розподіл даних. Оцінювати довірчий інтервал оцінки.
3. Знати сучасні методи та моделі машинного навчання, вміти навчати їх, здійснювати інтерполяцію та екстраполяцію даних.
4. Знати та вміти застосовувати основні способи візуалізації даних та їх розподілу.
5. Знати способи виявлення проблем в даних, зокрема пропущені та аномальні значення, мультиколінеарність, спотворюючі змінні. Вміти усувати їх різними способами. Вміти зменшувати просторову розмірність даних.
6. Застосовувати методи збору та аналізу даних взаємодії з користувачем (зацікавленість в веб-сторінках, кліки, конверсія). Вміти проводити A/B тест та застосовувати методи множинного тестування, здійснювати перевірку статистичної значущості їх результатів.

Лекція 1 – Первинний аналіз даних

Дані надходять із численних джерел: показань датчиків, подій, тексту, фотографій та відео. Інтернет речей (Internet of Things, IoT) викидає потоки інформації. Значна частина даних не структурована: фотографії є набором пікселів, при цьому кожен піксель містить інформацію про колір у форматі RGB (червоний, зелений, синій). Тексти складаються з послідовностей словникових та несловникових символів, часто розбитих на розділи, підрозділи тощо. По суті, головне завдання науки про дані полягає в тому, щоб переробляти цей потік сирих даних в інформацію, що має практичне значення. Щоб застосувати охоплені у цьому курсі статистичні поняття, неструктуровані сирі дані необхідно переробити в структуровану форму. Однією з форм структурованих даних, що зустрічаються, є таблиця з рядками і стовпцями.

Первинний аналіз даних є ключовим етапом статистичного дослідження, що дозволяє отримати перше уявлення про структуру, закономірності та особливості набору інформації. Цей етап охоплює обчислення основних статистичних мір, таких як середнє, медіана, дисперсія, стандартне відхилення, міжквартильний розмах (IQR) та інші, що дають змогу оцінити центральну тенденцію, розподіл та варіабельність даних. Важливо враховувати, що кожен з цих показників має свої підходи до інтерпретації: наприклад, медіана менш чутлива до викидів, ніж середнє, а IQR ефективно виявляє аномальні значення. Первинний аналіз не лише допомагає виявити помилки або аномалії в даних, але й формує основу для подальших статистичних моделей, вибираючи адекватні методи аналізу залежно від характеру розподілу та мети дослідження.

1.1 Типи прямокутних даних

Є два базові типи структурованих даних: числовий і категоріальний. Числові дані мають дві форми: безперервну, як, наприклад, швидкість вітру чи тривалість часу, і дискретну, як, наприклад, виникнення дії. Категоріальні дані приймають лише фіксований набір значень, як, наприклад, тип екрану телевізора (плазма, LCD, LED тощо) чи назву міста (Дніпро, Київ тощо). Двійкові дані є важливим випадком категоріальних даних. Ці дані приймають лише одне з двох значень, таких як 0/1, так/ні або істина/брехня. Ще один корисний тип категоріальних даних – порядкові дані, в яких категорії впорядковані; їх прикладом є числовий рейтинг (1, 2, 3, 4 чи 5).

Навіщо нам взагалі вникати у таксономію типів даних? Виявляється, що для цілей аналізу даних і передбачуваного моделювання тип даних грає значну роль, допомагаючи визначати спосіб візуалізації, аналізу даних чи статистичної моделі. По суті, у обчислювальних системах науки про дані, такі як R та Python, ці типи даних використовуються для покращення обчислювальної продуктивності. Ще важливіше, що тип змінної визначає те, яким чином обчислювальна система трактуватиме обчислення для цієї змінної.

Числові дані	**Категоріальні**

• Дані, що виражаються на чисельній шкалі
• **Неперервні**
  • Дані, можуть приймати будь-яке значення на інтервалі.
  • *Синоніми:* інтервал, число із плаваючою комою, чисельна величина.
• **Дискретні**
  • Дані, котрі можуть приймати лише цілочисельні значення, так як кількості.
  • *Синоніми:* ціле число, кількість, лічильна величина.
	• Дані, що можуть приймати лише конкретну множину значений, що представляють множину можливих категорій
	• *Синоніми:* перерахування, пронумеровані та номінальні дані, фактори.
	• **Двійкові**
	  • Окремий випадок категорійних даних з двома категоріями значень, наприклад, 0/1, істина/неправда.
	  • *Синоніми:* дихотомічна, логічна, індикаторна, булева величина.
	• **Порядкові**
	  • Категорійні дані, що мають явно вказаний порядок.
	  • *Синонім:* впорядкований фактор.

Розробники обчислювальних систем та програмісти баз даних можуть поставити запитання: а навіщо нам в аналітиці взагалі потрібен поділ на категоріальні та порядкові дані? Зрештою, категорії є просто колекцією текстових (чи числових) значень, і опорна база даних автоматично працює зі своїми внутрішнім уявленням. Однак чітка ідентифікація даних як категоріальні, на відміну від текстових, дійсно пропонує ряд переваг:

- знання про те, що дані є категоріальними, може служити сигналом для обчислювальної системи про те, яким чином повинні поводитися статистичні процедури, такі як створення графіка або підгонка моделі;
- зберігання та індексація даних можуть бути оптимізовані (як у реляційній базі даних);
- підтримка можливих значень, що приймаються конкретною категоріальною змінною, цільовою обчислювальною системою (як, наприклад, структура даних enum).

Прямокутні дані – це загальний термін для двомірної матриці, в якій рядки позначають записи (випадки), а стовпці – ознаки (змінні). Кадр даних – це специфічний формат, властивий мовам R та Python. Вихідні дані не завжди надходять у такій формі: неструктуровані дані (наприклад, текст) необхідно обробити і привести до такого виду, щоб їх можна було подати як безліч ознак прямокутних даних. Дані в реляційних базах даних повинні бути вилучені та поміщені в одну єдину таблицю для більшості завдань з аналізу даних та моделювання.

Терміни прямокутних даних

- Кадр даних (data frame)
    - Прямокутні дані (електронна таблиця) – це базова структура даних для статистичних моделей та моделей, що автоматично навчаються.
- Ознака
    - Стовбець в таблиці зазвичай зветься ознакою.
    - *Синоніми: атрибут, вхід, провісник, предиктор, змінна.*
- Вихід
    - Багато проектів науки про дані мають на меті передбачення результату форматі так/ні. Ознаки іноді використовуються для передбачення результату експерименту або статистичного дослідження.
    - *Синоніми:* результат, залежна змінна, відгук, ціль, вихід.
- Записи
    - Рядок у таблиці зазвичай називається записом.
    - *Синоніми:* випадок, приклад, прецедент, екземпляр, спостереження, шаблон, патерн, зразок.

1.2 Типовий формат даних

У табл. 1.1 показана суміш вимірюваних або кількісних даних (наприклад, тривалість та ціна) та категоріальних даних (наприклад, категорія та валюта). Як згадувалося раніше, спеціальної формою категоріальної змінної є двійкова змінна (так/ні чи 0/1). Таку змінну можна побачити у правому стовпці таблиці 1.1 – «змагальність» – індикаторна змінна, що показує, чи були торги змагальними (було кілька претендентів чи ні). Ця індикаторна змінна також є змінною результату, коли сценарій у тому, щоб передбачити змагальність чи незмагальність аукціону.

Традиційні таблиці бази даних мають чи кілька стовпців, іменованих індексом. Він значно підвищує ефективність деяких запитів до бази даних. У Python при використанні бібліотеки pandas основною прямокутною структурою

даних є об'єкт DataFrame, що містить таблицю даних. За замовчуванням для об'єкта DataFrame створюється автоматичний цілісний індекс, який ґрунтується на порядку проходження рядків таблиці. Pandas також має можливість задавати багаторівневі/ієрархічні індекси з метою підвищення ефективності деяких операцій.

Табл. 1.1. Приклад кадру даних (data frame).

Категорія	Валюта	Рейтинг продавця	Тривалість	День закриття	Ціна закриття	Ціна відкриття	Змагальність
Music/ Movie/ Game	US	3249	5	Понеділок	0,01	0,01	0
Music/ Movie/ Game	US	3249	5	Понеділок	0,01	0,01	0
Automotive	US	3115	7	Вівторок	0,01	0,01	0
Automotive	US	3115	7	Вівторок	0,01	0,01	0
Automotive	US	3115	7	Вівторок	0,01	0,01	0
Automotive	US	3115	7	Вівторок	0,01	0,01	0
Automotive	US	3115	7	Вівторок	0,01	0,01	1
Automotive	US	3115	7	Вівторок	0,01	0,01	1

1.3 Непрямокутні структури даних

Крім прямокутних даних, існують і інші структури даних.

Дані часового ряду записують послідовні виміри однієї й тієї ж змінної. Ці дані є сирим матеріалом для методів статистичного прогнозування, і вони також є ключовим компонентом даних, вироблених пристроями Інтернету-речей.

Просторові структури даних, що використовуються в картографічній та геопросторовій аналітиці, складніші та варіабельніші, ніж прямокутні структури даних. В об'єктному поданні центральною частиною даних є об'єкт (наприклад, будинок) та його просторові координати, а в польовій проекції основна увага приділяється малим одиницям простору та значенню відповідної метрики (яскравості пікселя, наприклад).

Графові (або мережеві) структури даних використовуються для подання фізичних, соціальних та абстрактних зв’язків. Наприклад, граф соціальної мережі, такий як Facebook або LinkedIn може представляти зв'язки між людьми в мережі.

Сполучені дорогами центри розподілу є прикладом фізичної мережі. Графові структури широко застосовуються в деяких типах завдань, таких як оптимізація мережі та рекомендаційні системи.

Кожен із цих типів даних має у науці про дані свою спеціалізовану методологію. У центрі уваги цього курсу знаходяться прямокутні дані – основний структурний елемент у передбачуваному моделюванні.

1.4 Оцінки центрального положення

Змінні з вимірюваними чи кількісними даними можуть мати тисячі чітко помітних значень. Базовий крок у розвідуванні даних полягає в отриманні «типового значення» для кожної ознаки (змінної): оцінки того, де розташована більшість даних (тобто їх центральна тенденція).

На перший погляд, завдання узагальнення даних виглядає досить тривіальною: треба просто взяти середнє арифметичне даних. Насправді незважаючи на те, що середнє обчислюється досить просто і його вигідно використовувати, воно не завжди буває найкращим підходом до обчислення центрального значення. З цієї причини у статистиці було розроблено та популяризовано кілька альтернативних оцінок середнього значення.

Оцінки центрального положення

- Середнє (mean)
    - Сума всіх значень, поділена на кількість значень.
    - *Синонім:* середнє арифметичне.
- Середнє зважене
    - Сума добутків всіх значень на їх ваги, поділена на суму ваг.
    - *Синонім:* середнє арифметичне зважене.
- Медіана (median)
    - Таке значення, що половина відсортованих даних знаходиться вище та нижче даного значення.
    - *Синонім:* 50-й перцентиль.
- Зважена медіана
    - Таке значення, що половина суми ваг знаходиться вище та нижче відсортованих даних.
- Середнє усічене
    - Середнє число всіх значень після відкидання фіксованої кількості граничних значень.
    - *Синонім:* обрізане середнє.

- Робастний – нечутливий до граничних значень.
    - *Синонім:* стійкий.
- Викид – значення даних, котре сильно відрізняється від більшості даних.

    Синонім: граничне, екстремальне чи аномальне значення.

Середньою базовою оцінкою центрального положення є середнє значення, або середнє арифметичне:

$$\text{Середнє} = \overline{x} = \frac{\sum_{i=1}^{n} x_i}{n} \tag{1.1}$$

N (або n) позначає загальну кількість записів чи спостережень. У статистиці позначення N використовується з великої літери, якщо воно позначає популяцію, і малої n, якщо воно означає вибірку з популяції. У науці про дані ця відмінність перестала бути важливою, і тому можна побачити і те й інше.

Різновидом середнього є середнє усічене, яке обчислюється шляхом відкидання фіксованого числа сортованих значень з кожного кінця послідовності і потім взяття середнього арифметичного значення, що залишилися. Усічене середнє усуває вплив граничних значень:

$$\text{Усічене середнє} = \overline{x} = \frac{\sum_{i=p+1}^{n-p} x_{(i)}}{n-2p} \tag{1.2}$$

Ще один вид середнього значення – це середньозважене значення, яке обчислюється шляхом множення кожного значення даних x_i хі на свою вагу w_i поділу їх суми на суму ваг:

$$\text{Середнє зважене} = \overline{x}_w = \frac{\sum_{i=1}^{n} w_i x_i}{\sum_{i}^{n} w_i} \tag{1.3}$$

1.4.1 Медіана та робастні оцінки

Медіана – це число, розташоване в сортованому списку даних рівно посередині. Якщо є парне число даних, то серединним значенням є те, що не знаходиться в наборі даних фактично, а є середнім арифметичним двох значень, які поділяють сортовані дані на верхню та нижню половини.

Порівняно із середнім, у якому використовуються абсолютно всі спостереження, медіана залежить лише від значень у центрі сортованих даних. Хоча це може виглядати як недолік, оскільки середнє значення набагато чутливіше до даних, існує багато прикладів, у яких медіана є більш відповідною метрикою центрального положення. Скажімо, ми хочемо подивитись типові доходи домогосподарств в округах, розташованих на узбережжі озера Вашингтон у Сіетлі. При порівнянні округу Медіна з округом Уіндермір використання середнього значення дало б різні результати, тому що в Медіні живе Білл Гейтс. Якщо ж ми використовуватимемо медіану, то вже не буде мати значення,

наскільки багатим є Білл Гейтс, – позиція серединного спостереження залишиться тією ж.

З тих же причин, з яких використовується середнє зважене, можна обчислити і зважену медіану. Як і з медіаною, ми спочатку виконуємо сортування даних, незважаючи на те, що з кожним значенням даних пов'язана вага. На відміну від середнього зваженого зважена медіана це таке значення, в якому сума ваг дорівнює для нижньої і верхньої половин сортованого списку. Як і медіана, зважена медіана робастна до викидів.

Медіана називається робастною оцінкою центрального становища, оскільки вона не перебуває під впливом викидів (граничних випадків), які можуть спотворити результати.

Викид – це будь-яке значення, яке сильно дистанційоване від інших значень у наборі даних. Точне визначення викиду є дещо суб'єктивним, незважаючи на те, що в прикладних пакетах використовуються деякі правила. Викид як такий робить значення даних нерепрезентативним (як у попередньому прикладі з Біллом Гейтсом) чи помилковим. Разом з тим, викиди часто є результатом помилок даних, таких як змішування даних з різними одиницями вимірювання (кілометри з метрами) або погані показання від датчика. Коли викиди є результатом неправильних даних, середнє значення показуватиме погану оцінку центрального становища, тоді як медіана буде як і раніше допустимою. У будь-якому випадку викиди повинні бути виявлені і зазвичай заслуговують на подальше розслідування.

Медіана не єдина робастна оцінка центрального становища. Насправді з метою запобігання впливу викидів широко використовується і середнє усічене. Наприклад, усічення нижніх і верхніх 10% даних (загальноприйнятий вибір) забезпечить захист від викидів у всіх, крім найменших, наборах даних. Середнє усічене може вважатися компромісом між медіаною і середнім: воно робастно до граничних значень даних, але використовує більше даних для розрахунку оцінки центрального положення.

Окремий випадок – виявлення аномалій

На відміну від типового аналізу даних, де викиди іноді інформативні, а іноді – прикра перешкода, у виявленні аномалій цільовими об'єктами є викиди, і значний масив даних переважно служить для визначення "норми", з якою співміряються аномалії.

Інші робасні метрики центрального положення

У статистиці була розроблена маса інших оцінювачів, так званих естіматорів, центрального положення переважно з метою розробки більш робастних інструментів оцінки, ніж середнє, і більш ефективних (тобто здатних краще виявляти невеликі різниці в центральному положенні між наборами даних). Ці методи є потенційно корисними для невеликих наборів даних. Разом про те

вони ледь дають додаткові вигоди за умов великих і навіть помірно розмірних наборів даних.

1.4.2 Приклад оцінки центрального положення в Python

У табл. 1.2 показані перші кілька рядків із набору даних, що містить відомості про чисельність населення та рівень вбивств (в одиницях вбивств на 100 тис. осіб на рік) по кожному штату.

Табл. 1.2. Дані про населення та рівень вбивств в США.

№	Штат	Населення	Рівень вбивств (на 100 тис. осіб)	Абрев.
1	Alabama	4779736	5.7	AL
2	Alaska	710231	5.6	AK
3	Arizona	6392017	4.7	AZ
4	Arkansas	2915918	5.6	AR
5	California	37253956	4.4	CA
6	Colorado	5029196	2.8	CO
7	Connecticut	3574097	2.4	CT
8	Delaware	897934	5.8	DE
9	Texas	29145505	4.1	TX
10	New York	19297729	3.5	NY

Прочитаємо кадр даних, розрахуємо середнє, усічене середнє, медіану та зважене середнє.

Код

```
import pandas as pd
import numpy as np
from scipy.stats import trim_mean

state = pd.read_csv('state.csv')
print(state['Населення'].mean())
print(trim_mean(state['Населення'], 0.1))
print(state['Населення'].median())

print(np.average(state['Рівень вбивств (на 100 тис.
осіб)'],weights=state['Населення']))
```

Вивід

```
10999631.9
9004016.5
4904466.0
4.149384641680601
```

Середнє більше середнього усіченого, яке більше медіани. Це викликано тим, що середнє усічене виключає найбільші та найменші штати (trim=0.1) відкидає по 1% з кожного кінця). Якщо ми захочемо вирахувати середньостатистичний рівень убивств у країні, то маємо використовувати зважене середнє чи медіану, щоб врахувати різну чисельність населення у штатах.

Зважене середнє значення доступне за допомогою пакета NumPy. Для обчислення середнього та медіани на Python ми можемо використовувати методи кадр даних пакета pandas. Для усіченого середнього значення потрібна функція trim_mean з бібліотеки scipy.stats.

1.5 Оцінки варіабельності

Центральне положення – це лише одна з розмірностей в узагальненні ознаки. Друга розмірність – варіабельність, іменована також дисперсністю, показує, чи згруповані значення даних щільно, чи вони розкидані. В основі статистики лежить варіабельність: її вимір, зменшення, розрізнення випадкової варіабельності від реальної, ідентифікація різних джерел реальної варіабельності та прийняття рішень за умов її присутності.

Так само як і у випадку центрального положення, яке можна виміряти різними способами (середнє, медіана тощо), існують різні способи виміряти варіабельність.

Терміни оцінок варіабельності

- Відхилення
  - Різниця між значенням спостережень та оцінкою центрального положення.
  - *Синоніми:* помилки, похибки, залишки.
- Дисперсія
  - Сума квадратичних відхилень від середнього, поділена на $n - 1$, де n – кількість значень даних.
  - *Синоніми:* середньоквадратичне відхилення, середньоквадратична помилка.
- Стандартне відхилення
  - Квадратний корінь з дисперсії.
  - *Синоніми:* норма l_2, *євклідова норма.*
- Середнє абсолютне відхилення
  - Середнє абсолютних значень відхилень від середнього.
  - *Синоніми:* норма l_1, мангеттенська норма

- Медіанне абсолютне відхилення від медіани
- Розмах
    - Різниця між найбільшим та найменшим значеннями в наборі даних
- Порядкові статистики
    - Метрики на основі значень даних, відсортованих від самих малих до самих крупних.
    - *Синонім:* ранги.
- Перцентиль
    - Таке значення, що P відсоток значень приймає дане значення або менше та $(100 - P)$ процент значень приймає дане значення або більше.
    - *Синонім*: квантиль.
- Міжквартильний розмах
    - Різниця між 75-м та 25-м перцентилями.
    - *Синоніми:* MKP, IQR.

1.5.1 Стандартне відхилення та пов'язані з ним оцінки

Найбільш широко використовувані оцінки варіабельності засновані на різницях або відхиленнях, між оцінкою центрального положення і даними, що спостерігаються. Для набору даних $\{1, 4, 4\}$ середнє дорівнює 3, а медіана – 4. Відхилення від середнього являють собою різниці: $1 - 3 = -2$, $4 - 3 = 1$, $4 - 3 = 1$. Ці відхилення говорять про те наскільки дані розкидані навколо центрального значення.

Один із способів виміряти варіабельність полягає в тому, щоб оцінити типове значення цих відхилень. Усереднення самих відхилень мало, тому негативні відхилення нейтралізують позитивні. Фактично сума відхилень від середнього дорівнює нулю. Натомість простий підхід полягає в тому, щоб взяти середнє абсолютних значень відхилень від середнього значення. У попередньому прикладі абсолютне значення відхилень дорівнює $\{2, 1, 1\}$, а середнє — $(2 + 1 + 1) / 3 = 1,33$. Це і є середнє абсолютне відхилення, яке обчислюється за такою нижче формулою:

$$\text{Середнє абсолютне відхилення} = \frac{\sum_{i=1}^{n} |x_i - \overline{x}|}{n}, \tag{1.4}$$

де $\overline{x}$ – середнє значення у вибірці, або вибіркове середнє.

Найвідомішими оцінками варіабельності є дисперсія та стандартне відхилення, що ґрунтуються на квадратичних відхиленнях. Дисперсія – це середнє квадратичних відхилень, стандартне відхилення – квадратний корінь з дисперсії.

$$\text{Дисперсія} = s^2 = \frac{\sum(x-\overline{x})^2}{n-1} \quad (1.5)$$

$$\text{Стандартне відхилення} = s = \sqrt{\text{дисперсія}} \quad (1.6)$$

Стандартне відхилення інтерпретується набагато простіше ніж дисперсія, оскільки воно знаходиться на тій же шкалі вимірювання, що і вихідні дані. Однак, враховуючи його складнішу та інтуїтивно менш зрозумілу формулу, може здатися дивним, що у статистиці стандартному відхиленню віддається перевага порівняно із середнім абсолютним відхиленням. Таке переважання має корені в статистичній теорії: математично робота з квадратичними значеннями набагато зручніше, ніж із абсолютними, особливо у разі статистичних моделей.

Ступені свободи та n *або* $n-1$*?*

У книгах зі статистики завжди так чи інакше обговорюється питання: чому у формулі дисперсії у нас у знаменнику $n-1$ замість n, який призводить до поняття ступенів свободи? На практиці ця відмінність не є важливою, оскільки n зазвичай є великим настільки, що вже не має великого значення, чи поділ виконуватиметься на n або $n-1$.

Відмінність базується на передумові, що ви хочете отримати оцінки популяції, виходячи з вибірки з неї. Якщо у формулі дисперсії застосувати інтуїтивно зрозумілий знаменник n, то справжні значення дисперсії та стандартного відхилення у популяції будуть недооцінені. Така оцінка називається зміщеною. Однак якщо поділити на $n-1$ замість n, то стандартне відхилення стає незміщеною оцінкою.

Ні дисперсія та стандартне відхилення, ні середнє абсолютне відхилення не стійкі до викидів та граничних значень. Дисперсія та стандартне відхилення чутливі до викидів найбільше, оскільки вони ґрунтуються на квадратичних відхиленнях.

Робастною оцінкою варіабельності є абсолютне медіанне відхилення від медіани (median absolute deviation, MAD):

$$\text{Медіанне абсолютне відхилення} = \text{медіана}(|x_1 - m|,\ |x_2 - m|, \ldots |x_N - m|), \quad (1.7)$$

де m– це медіана.

Як і у випадку з медіаною, медіан абсолютне відхилення від медіани не під впливом граничних значень. Можна також обчислити усічене стандартне відхилення за аналогією із середнім усіченим

Зауваження. Дисперсія, стандартне відхилення, середнє абсолютне відхилення та медіанне абсолютне відхилення від медіани не є еквівалентними оцінками, навіть у разі, коли дані надходять із нормального розподілу. Насправді стандартне відхилення завжди більше середнього абсолютного відхилення, яке, своєю чергою, більше медіанного абсолютного відхилення. Іноді абсолютне

відхилення (MAD) множать на постійний поправочний коефіцієнт, щоб поставити цю метрику в ту ж шкалу, що і стандартне відхилення у разі нормального розподілу. Коефіцієнт 1,4826, що часто використовується, означає, що 50% нормального розподілу потрапляє в діапазон ±MAD

Стандартне відхилення та пов'язані з ним оцінки

- Середнє абсолютне відхилення $= \frac{\sum_{i=1}^{n}|x_i-\overline{x}|}{n}$, де $\overline{x}$ – середнє значення у вибірці, або вибіркове середнє.
- Дисперсія $= s^2 = \frac{\sum(x-\overline{x})^2}{n-1}$
- Стандартне відхилення $= s = \sqrt{\text{дисперсія}}$
- Медіанне абсолютне відхилення $=$ медіана$(|x_1 - m|,\ |x_2 - m|, \ldots |x_N - m|)$, де m – це медіана.

1.5.2 Оцінки на основі перцентилів

Інший підхід до оцінювання дисперсії заснований на розгляді розкиду сортованих даних. Статистичні показники з урахуванням сортованих (ранжованих) даних називаються порядковими статистиками. Елементарна міра – це розмах: різниця між найбільшим і найменшим числом. Мінімальні та максимальні значення як такі корисно знати, оскільки вони допомагають виявляти викиди, але розмах надзвичайно чутливий до викидів і не дуже корисний як загальна міра дисперсності в даних.

$$\text{Розмах} = \max_i\{x_i\} - \min_i\{x_i\} \tag{1.8}$$

Щоб уникнути чутливості до викидів, ви можете звернутися до розмаху даних після відкидання значень з кожного кінця. Ці типи оцінок формально ґрунтуються на різницях між перцентилями. У наборі даних P-й процентиль є таким значенням, що принаймні P відсотків значень приймає це значення або менше, і принаймні $(100 - P)$ відсотків значень приймає це значення або більше. Наприклад, для знаходження 80-го перцентиля треба відсортувати дані. Потім, починаючи з найменшого значення, пройти 80% вгору до найбільшого значення. Зазначимо, що медіана – це те саме, що й 50-й процентиль. Перцентиль, сутнісно, аналогічний квантилю, але квантилі індексуються частками (так, квантиль 0,8 — те саме, як і 80-й процентиль).

Загальноприйнятим способом оцінки варіабельності є різниця між 25-м та 75-м процентилями, яка називається міжквартильним розмахом (interquartile range, IQR).

Приклад. Нехай маємо дані: $3, 1, 5, 3, 6, 7, 2, 9$. Ці числа ми сортуємо, отримавши $1, 2, 3, 3, 5, 6, 7, 9$. 25-й процентиль рівний 2,5, а 75-й процентиль дорівнює 6,5, тому міжквартильний розмах буде $6{,}5 - 2{,}5 = 4$.

Для великих наборів даних розрахунок точних процентилей буває обчислювально дуже витратним, оскільки він вимагає сортування всіх значень даних. В статистичних обчислювальних системах використовуються спеціальні алгоритми, які отримують наближений процентиль, обчислюючи його дуже швидко та гарантовано забезпечуючи певну точність.

Якщо є парне число даних (n – парне), то процентиль є неоднозначним поняттям. Насправді можна взяти будь-яке значення між порядковими статистиками $x_{(j)}$ та $x_{(j+1)}$, де j задовольняє:

$$100 \cdot \frac{j}{n} \leq P \leq 100 \cdot \frac{j+1}{n} \tag{1.9}$$

Оцінки на основі перцентилів

- Розмах = $\max_i\{x_i\} - \min_i\{x_i\}$.
- Перцентиль – таке значення, що P відсоток значень приймає дане значення або менше та $(100 - P)$ процент значень приймає дане значення або більше.
- Міжквартильний розмах – різниця між 75-м та 25-м перцентилями.

1.5.3 Приклад розрахунку оцінок на основі перцентилів в Python

Повернемось до таблиці 1.2 із даними про населення та рівень вбивств в США. Кадр даних пакету pandas надає методи обчислення стандартного відхилення і квантилей. Використовуючи квантилі, ми можемо легко визначити міжквартильний розмах (IQR). Для робастної оцінки медіанного абсолютного відхилення (MAD) ми використовуємо функцію robust.scale.mad з statsmodels:

Код

```
import pandas as pd
from statsmodels import robust

state = pd.read_csv('state.csv')

print('Стандартне відхилення')
print(state['Населення'].std())

print('Дисперсія')
print(state['Населення'].var())

print('Квантиль (75%)')
print(state['Населення'].quantile(0.75))
```

```
print('Квантиль (25%)')
print(state['Населення'].quantile(0.25))

print('Міжквартильний розмах')
print(state['Населення'].quantile(0.75) - state['Населення'].quantile(0.25))

print('Медіанні абсолютні відхилення')
print(robust.scale.mad(state['Населення']))
```

Вивід

```
Стандартне відхилення
12958845.349848783
Дисперсія
167931672801297.44
Квантиль (75%)
16071301.0
Квантиль (25%)
3080462.75
Міжквартильний розмах
12990838.25
Медіанні абсолютні відхилення
4444159.454059282
```

Стандартне відхилення майже втричі більше від медіанного абсолютного відхилення MAD. І це не дивно, оскільки стандартне відхилення є чутливим до викидів.

1.6 Контрольні питання

1. Якою є мета первинного аналізу даних?
2. Що таке прямокутні дані?
3. Які пакети Python передбачені для роботи з електронними таблицями, для оцінки центрального положення та варіабельності?
4. Які оцінки центрального положення ви знаєте?
5. У яких випадках медіана є кращим показником центральної тенденції, ніж середнє?
6. Що таке викид? Яку проблему він складає під час первинного аналізу даних?
7. Що таке оцінка варіабельності?
8. У яких випадках використовується зважене середнє, а не просте арифметичне середнє? Наведіть приклади.
9. Якими є основні оцінки варіабельності? Які формули для розрахунків?
10. Чи є оцінки на основі перцентилів стійкими до викидів? Чому?

Лекція 2 – Візуалізація даних

2.1 Читання даних. Формати даних CSV та Excel

CSV-файл (Comma-Separated Values) – це простий текстовий файл у вигляді прямокутної таблиці. Кожен рядок у файлі відповідає одному рядку таблиці, а поля в рядку розділені роздільником – зазвичай це кома, хоча також можуть використовуватися інші символи, такі як табуляція або крапка з комою. Оскільки файл є простим текстом, його можна відкривати, переглядати та редагувати за допомогою будь-якого редактора, передавати між операційними системами без конвертації та зберігати в системах контролю версій без ризику пошкодження бінарних даних.

Формат має свої корені в ранніх днях обміну даними, коли єдиним надійним способом переміщення табличних даних між різними програмами було записування їх у вигляді простого рядка символів. Простота специфікації – один рядок на запис, поля, розділені роздільником – зробила CSV універсальною мовою для обміну даними, яка зберігається навіть після появи більш складних форматів.

Разом із знайомим читачу форматом даних таблиць Excel (.xls або .xlsx), CSV файли є одним з найбільш розповсюджених форматів даних з яким працюють аналітики даних.

2.1.1 Вигляд CSV файлу

Розглянемо невеликий приклад:

data.csv

```
name,age,city
Alice,31,New York
Bob,27,Seattle
```

- Перший рядок часто розглядається як заголовок, що позначає назву кожного стовпця.
- Другий і третій рядки містять значення даних.
- Коми розділяють три колонки, а символи нового рядка завершують кожен запис.

2.1.2 CSV проти формату електронних таблиць Excel

Робочі книги Excel (.xls та .xlsx) та файли CSV зберігають табличні дані, проте належать до принципово різних категорій файлів. Порівняння цих форматів даних показано в таблиці 2.1.

Табл. 2.1. Порівняння форматів даних CSV та Excel.

Аспект	CSV	Excel (xlsx)
Структура	Плоский, односторінковий, текстовий рядок	Ієрархічний ZIP-пакет, що містить XML для декількох аркушів, стилів, діаграм, макросів
Типи даних	Всі значення є рядками; визначення типу залишається за програмою, що їх використовує	Багаті вбудовані типи (числа, дати, логічні значення, багатий текст, формули)
Метадані	Не підтримується форматування, коментарі або приховані рядки/стовпці	Повна підтримка форматування комірок, умовного форматування, перевірки даних, коментарів тощо
Формули	Не зберігаються; відображаються тільки отримані значення	Формули зберігаються і можуть бути перераховані при відкритті
Розмір файлу	Зазвичай менший, оскільки не містить стилів та бінарних блоків	Більший розмір через вбудований XML, стилі та додаткові медіафайли
Портативність	Може бути прочитаний будь-яким текстовим редактором, будь-якою мовою програмування, будь-якою ОС	Потрібне програмне забезпечення, яке розуміє формат Open XML (Excel, LibreOffice або спеціалізовані бібліотеки)

На практиці CSV є вибором, коли вам потрібен легкий, портативний і зручний для читання знімок даних, який буде використовуватися програмно. Excel показує себе з найкращого боку, коли дані потрібно презентувати, анотувати або обробляти інтерактивно – наприклад, коли ви хочете вбудувати формули, умовне форматування або кілька пов'язаних таблиць в один файл.

Оскільки CSV не має поняття типів даних, програми, що читають CSV, повинні вирішувати, як інтерпретувати кожну колонку. Excel, навпаки, записує передбачуваний тип (наприклад, «Дата») і може зберігати його протягом сеансів. Ця різниця є двосічним мечем: простота CSV надає йому універсального застосування, але також покладає тягар визначення типу на читача.

2.1.3 Імпорт даних CSV та Excel у pandas

Функція в pandas, яка виконує основну роботу, – це pandas.read_csv. Припустимо, що файл people.csv містить трирядковий приклад із розділу 2.1.1. У результаті буде створено DataFrame із стовпцями name, age та city і двома рядками даних.

Код

```
import pandas as pd

df = pd.read_csv('people.csv')
print(df.head())
```

Вивід

```
    name  age      city
```

```
0  Alice   31  New York
1    Bob   27   Seattle
```

Читання файлу Excel відбувається аналогічно, але використовується функція read_excel (підтримає як файли .xls так і .xlsx).

Код

```
import pandas as pd

df = pd.read_excel('people.xlsx')
print(df.head())
```

Вивід

```
    name  age      city
0  Alice   31  New York
1    Bob   27   Seattle
```

2.2 Візуалізація даних, що змінюються в часі

Лінійний графік — це один із найпростіших і найефективніших способів простежити поглядом послідовність точок даних. Коли змінна змінюється за впорядкованим виміром (найчастіше за часом), проведення лінії між послідовними спостереженнями перетворює таблицю з сухими цифрами на історію, яку можна прочитати одним поглядом. У цьому розділі ми розглянемо невеликий набір даних, імпортуємо його в Python за допомогою pandas і перетворимо його на лінійний графік. По ходу ми обговоримо ситуації, в яких лінійний графік є найефективнішим, а також сценарії, в яких він може ввести в оману.

2.2.1 Чому лінійний графік?

Уявіть, що у вас є датчик, який щогодини реєструє температуру; акції на біржі про які щодня реєструється ціна закриття, або веб-сайт, який реєструє кількість відвідувачів за хвилину. У кожному випадку точки даних упорядковані природним чином, і ви хочете побачити тенденцію – напрямок, темп, піки, плато. Лінійний графік саме це і робить:

- Безперервність – з'єднуючи точки лінією, ви неявним чином припускаєте, що змінна плавно змінюється між спостереженнями.
- Порівняння – можна накласти кілька ліній, щоб порівняти серії, які мають однаковий індекс (наприклад, продажі двох лінійок продуктів за місяцями).
- Читабельність – навіть швидкий погляд дозволяє побачити рух вгору або вниз, перетворюючи список чисел на інтуїтивно зрозумілу візуалізацію.

Через це лінійні графіки є стандартом для аналізу часових рядів, моніторингу інформаційних панелей та будь-яких ситуацій, де незалежна змінна має природний порядок.

2.2.2 Коли слід уникати лінійних графіків

Лінійний графік може бути оманливим, якщо базові припущення не справджуються. Звертайте увагу на такі тривожні ознаки:

- Категоріальна вісь x – якщо горизонтальна вісь представляє окремі категорії без логічного порядку (наприклад, «червоний», «синій», «зелений»), малювання ліній натякає на прогресію, якої насправді немає. Стовпчикові діаграми або точкові графіки є більш надійними.
- Рідкісне або нерегулярне вибіркове спостереження – коли спостереження є нечисленними та рідкісними, лінія заповнює прогалини, які можуть не існувати. Діаграма розсіювання може передавати дані більш достовірно.
- Велика кількість рядів – накладання десятків ліній швидко перетворюється на візуальну плутанину. Спершу розгляньте можливість агрегації даних.
- Небезперервні змінні – якщо вимірювана величина є бінарною або дуже дискретною, лінія може створювати ілюзію безперервності. Більш доречним може бути використання гістограми.

2.2.3 Приклад

Розглянемо наступний приклад. Нехай ми маємо дані про температуру як показано нижче.

data.csv

```
date,temp_c
2025-01-01,7.4
2025-01-02,8.3
2025-01-03,5.2
2025-01-04,5.8
2025-01-05,5.0
2025-01-06,6.5
2025-01-07,7.0
2025-01-08,6.0
2025-01-09,7.5
2025-01-10,6.2
2025-01-11,5.8
2025-01-12,6.7
2025-01-13,6.0
```

Побудуємо графік залежності температури від дати. Для цього виконаємо наступний код:

Код

```
import pandas as pd
import matplotlib.pyplot as plt

df = pd.read_csv('data.csv')

df["temp_c"].plot(
    title="Графік щоденної температури",
    ylabel="Температура",
    xlabel="Дата"
)

plt.show()
```

Результуючий графік показано на рис. 2.1.

Рис. 2.1 – Побудований лінійний графік.

2.3 Візуалізація розділу даних

Для табличних даних, що містять описи різних об’єктів використання лінійного графіку буде недоречним. В даному випадку, ми не маємо інформації про зміну стану об’єкту із плином часу, а проміжного значення між різними об’єктами немає.

Всі охоплені нами в попередній лекції оцінки узагальнюють дані у одному-одному числі з метою опису центрального положення чи варіабельності даних. Також корисно розвідати те, як дані розподілені в сукупності.

2.3.1 Перцентилі і коробчасті діаграми

У попередній лекції ми дізналися, як перцентилі можуть використовуватися для вимірювання розкиду даних. Перцентилі також важливі для узагальнення всього розподілу загалом. Загально прийнято повідомляти про квартилі (25-й, 50-й та 75-й перцентилі) та децилі (10-й, 20-й, ..., 90-й перцентилі). Перцентиль особливо важливі узагальнення хвостів (зовнішнього розмаху) розподілу. Масова культура узвичаїла термін «однопроцентовики», який відноситься до людей у верхньому 99-му відсотку багатства.

Коробкова діаграма (іноді її називають діаграмою «коробка з вусами») – це один із найкомпактніших способів передати розподіл набору даних. За допомогою декількох штрихів він показує, де знаходиться основна частина спостережень, як вони розподілені, чи є розподіл симетричним і чи є якісь значення, що надто віддалені від решти.

У табл. 2.2 показані деякі відсотки рівня вбивств по штатах. Медіана дорівнює 4 вбивствам на 100 тис. людей, незважаючи на те, що є досить велика варіабельність: 5-й перцентиль становить всього 1,6, тоді як 95-й перцентиль – 6,51.

Табл. 2.2. Перцентилі рівня вбивств по штатам.

5%	25%	50%	75%	95%
1,60	2,42	4,00	5,55	6,51

Коробчасті діаграми, введені у вживання Тьюкі, засновані на відсотках і забезпечують швидкий спосіб візуалізації розподілу даних. На рис. 2.2 показана коробчаста діаграма населення за штатами.

Код

```
import pandas as pd
import matplotlib.pyplot as plt

state = pd.read_csv('state.csv')

ax = (state['Population']/1000000).plot.box()
ax.set_ylabel('Population (millions)')

plt.tight_layout()
plt.show()
```

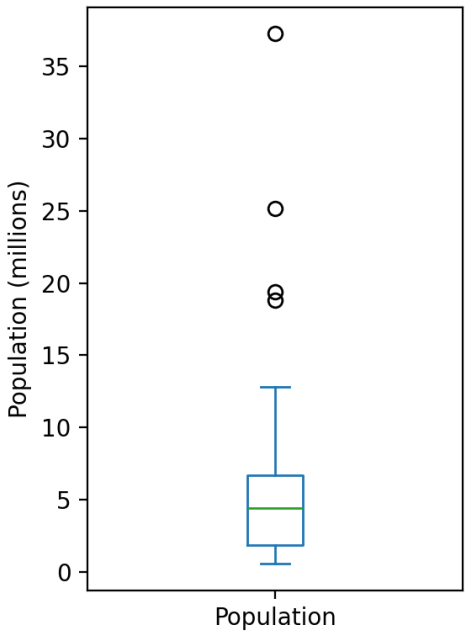


Рис. 2.2 – Коробкова діаграма, що візуалізує розподіл населення.

Пакет pandas надає низку базових розвідувальних графіків для кадру даних. Одним із них є коробчасті діаграми.

З цієї коробчатої діаграми ми одразу бачимо, що медіанна чисельність населення штатів становить близько 5 млн. чоловік, половина штатів припадає на чисельність від 2 до 7 млн., і існують деякі викиди з високим населенням. Верхня і нижня частини коробки – це відповідно 75-й і 25-й перцентилі. Медіана показана горизонтальною лінією у прямокутнику. Пунктирні лінії, які називаються вусами, простягаються від верхньої та нижньої частини коробки, і говорять про розмах для основної частини даних. Існує багато варіантів коробчастої діаграми , наприклад, у мові R і в Python функція побудови розширює вуса до найдальшої точки за межами коробки, за винятком того, що вона не виходитиме за межі 1,5-кратного міжквартильного розмаху (IQR). Інші обчислювальні системи можуть використовувати інше правило. Будь-які дані за межами вусів зображуються у вигляді окремих точок або кіл.

2.3.2 Частотна таблиця

Частотна таблиця змінної ділить діапазон змінної на рівновіддалені сегменти і повідомляє, скільки значень потрапляє у кожен сегмент. Приклад показано в таблиці 2.3.

Код

```
import pandas as pd

state = pd.read_csv('state.csv')
```

```
binnedPopulation = pd.cut(state['Population'], 10)
print(binnedPopulation.value_counts())
```

Табл. 2.3. Частотна таблиця населення штатів.

№	Розмах кошику	Кількість	Штати
1	563 626 – 4 232 658	24	WY, VT, ND, AK, SD, DE, MT, RI, NH, ME, HI, ID, NE, WV, NM, NV, UT, KS, AR, MS, IA, CT, OK, OR
2	4 232 659 – 7 901 691	14	KY, LA, SC, AL, CO, MN, WI, MD, MO, TN, AZ, JN, MA, WA
3	7 901 692 – 11 570 724	6	VA, NJ, NC, GA, MI, OH
4	11 570 725 – 15 239 757	2	PA, IL
5	15 239 758 – 18 908 790	1	FL
6	18 908 791 – 22 577 823	1	NY
7	22 577 824 – 26 246 856	1	TX
8	26 246 857 – 29 915 889	0	
9	29 915 890 – 33 584 922	0	
10	33 584 923 – 37 253 956	1	CA

Найменш густонаселеним штатом є Вайомінг з населенням 563 626 осіб (згідно з переписом 2010 р.), а найгустонаселенішим — Каліфорнія з населенням 37 253 956 осіб. Це дає нам розмах 37253956-563626 = 36690330, який ми повинні розділити на рівні кошики, скажімо, 10 кошиків. За наявності 10 рівнорозмірних кошиків кожен з них матиме ширину 3669033; таким чином, перший кошик буде простягатися від 563 626 до 4 232 658. Верхній кошик, 33 584 923-37 253 956, має всього один штат – Каліфорнію. Два кошики, які йдуть безпосередньо нижче штату Каліфорнія, залишаються порожніми, поки ми не досягнемо штату Техас. Важливо враховувати порожні кошики; той факт, що в цих кошиках немає значення, є корисною інформацією. Можливо, також буде корисно поекспериментувати з різними розмірами кошиків. Якщо вони дуже великі, то важливі ознаки розподілу можуть бути затушовані. Якщо вони занадто малі, то результат буде занадто гранулярним, і можливість спостерігати загальну картину буде втрачено.

І таблиці частот, і перцентилі узагальнюють дані з допомогою створення кошиків, тобто частотних інтервалів. У загальному випадку квартили та децилі матимуть однакову кількість у кожному кошику (рівнокількісні кошики), але розміри кошиків відрізнятимуться. Частотна таблиця, на відміну від них, матиме різні кількості в кошиках (рівнорозмірні кошики) і розміри кошиків будуть однаковими.

2.3.3 Гістограми

Гістограма це спосіб візуалізації частотної таблиці, де кошики (частотні інтервали) відкладаються на осі x, а кількість даних – на осі y.

Пакет pandas підтримує гістограми для кадрів даних завдяки методу DataFrame.plot.hist . Використовуйте іменований аргумент bins для визначення кількості кошиків. Різні методи побудови графіків повертають координатний об’єкт, який дозволяє точніше відрегулювати візуалізацію за допомогою пакета matplotlib .

Код

```
import pandas as pd
import matplotlib.pyplot as plt

state = pd.read_csv('state.csv')

ax = (state['Population'] / 1000000).plot.hist()
ax.set_xlabel('Population (millions)')

plt.tight_layout()
plt.show()
```

В результаті вийде гістограма, яка показана на рис. 2.3. У загальному випадку гістограми будуються таким чином, що:

- число кошиків (або, що те саме, розмір кошика) задається користувачем;
- розміщення стовпців гістограми безперервно – прогалини між ними відсутні, якщо немає порожнього кошика.

Статистичні моменти. У статистичній теорії центральне положення та варіабельність згадуються як моменти розподілу першого та другого порядку. Моменти третього та четвертого порядку – це асиметрія та ексцес. Під асиметрією розуміється зміщення даних до більших чи менших значень, а під ексцесом – схильність даних до граничних значень. Для вимірювання асиметрії та ексцесу, як правило, метрики не використовуються; натомість вони виявляються при візуальному огляді зображень.

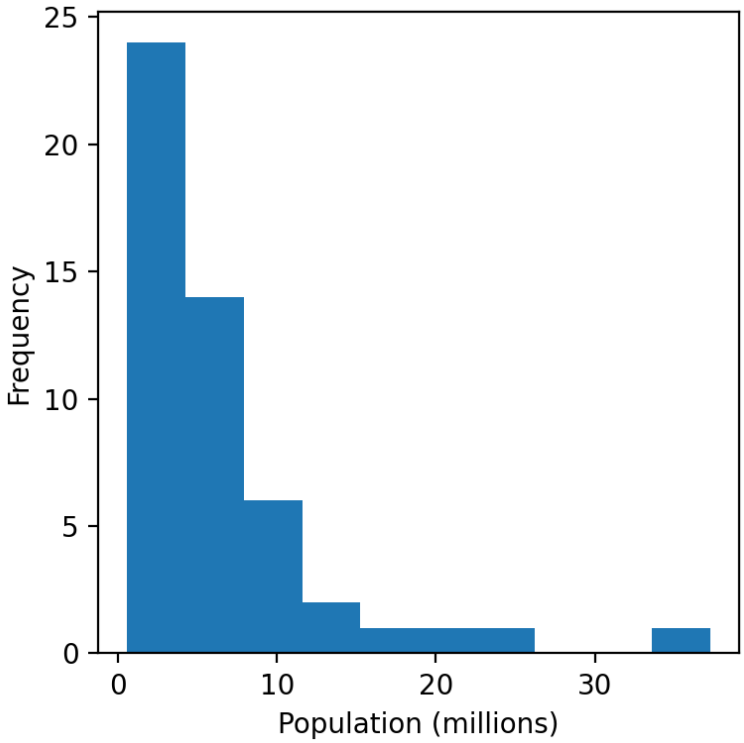


Рис. 2.3 – Гістограма розподілу населення.

2.3.4 Графік щільності

Ключова відмінність від гістограми, показаної на попередньому рисунку, полягає в шкалі осі): графік щільності (рис. 2.4) відповідає відображенню гістограми як частки, а не кількостей. Зверніть увагу, що загальна площа під кривою щільності дорівнює 1, і замість кількостей у кошиках ви обчислюєте площі під кривою між будь-якими двома точками на осі x, які відповідають частки розподілу між цими двома точками.

Код

```
import pandas as pd
import pandas as pd
import matplotlib.pyplot as plt

state = pd.read_csv('state.csv')

ax = state['Murder.Rate'].plot.hist(density=True, xlim=[0, 12],
                                    bins=range(1,12))
state['Murder.Rate'].plot.density(ax=ax)
ax.set_xlabel('Murder Rate (per 100,000)')

plt.tight_layout()
plt.show()
```

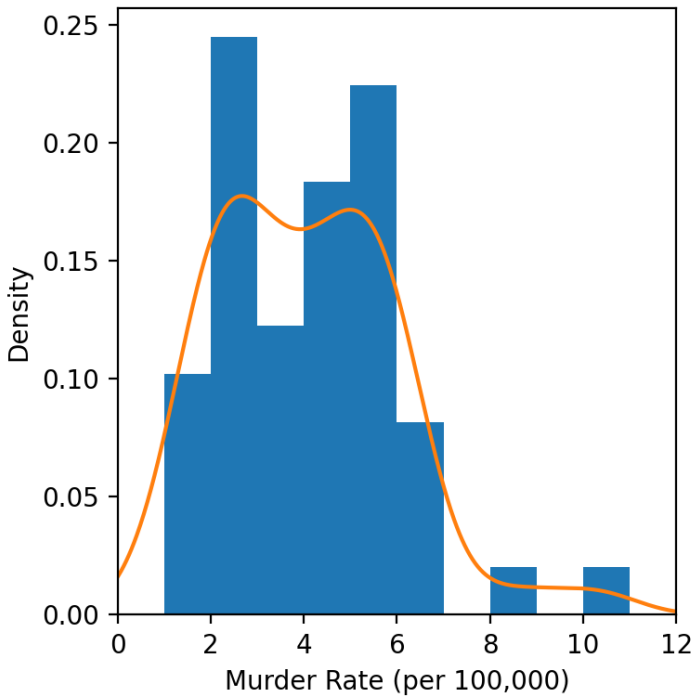


Рис. 2.4 – Графік щільності кількості вбивств.

2.4 Розвідування двійкових та категоріальних даних

Мода. Мода – це значення (або декілька значень у разі, якщо вони є однаковими), яке з'являється в даних частіше за інших. Наприклад, модою затримок через проблеми в аеропорту Даллас/Форт-Уерт є фактор «запізнення прибуття повітряного судна» (Inbound) . Ось ще один приклад: у більшості куточків США модою релігійних уподобань буде християнство. Мода є простою зведеною статистикою для категоріальних даних і зазвичай для числових даних не використовується.

Очікуване значення. Особливим типом категоріальних даних є дані, в яких категорії представлені або можуть бути пов'язані з дискретними значеннями на однаковій шкалі вимірювання. Маркетолог нової «хмарної» технології, наприклад, продає два рівні веб-служб, один за ціною 300 $ на місяць, а інший за ціною 50 $ на місяць. При цьому він пропонує безкоштовні вебінари для формування списку потенційних клієнтів, і фірма вважає, що 5% відвідувачів підпишуться на веб-служби за 300$, 15% на веб-служби за 50$, і 80% не підпишуться зовсім. Ці дані можна узагальнити для проведення фінансових розрахунків в одному «очікуваному значенні», що є формою середнього зваженого, в якому вагами виступають ймовірності.

Очікуване значення, чи математичне очікування, обчислюється так: 1. Помножити кожен результат ймовірність його наступу. 2. Підсумувати ці значення.

У прикладі «хмарної» служби очікуване значення відвідувача вебінару, таким чином, складе 22,50 $ на місяць, яке розраховується таким чином:

$$EV = 0{,}05 * 300 + 0{,}15 * 50 + 0{,}80 * 0 = 22{,}5$$

Очікуване значення, по суті, є формою середнього зваженого: воно привносить поняття майбутніх очікувань і ймовірнісних ваг, часто ґрунтуючись на суб'єктивному судженні. Очікуване значення (математичне очікування) є фундаментальним поняттям в оцінюванні ринкової вартості бізнесу та складанні бюджету довгострокових витрат – наприклад, очікуване значення для п'ятирічних прибутків від нового придбання або очікуване скорочення витрат від нової обчислювальної системи обліку пацієнтів у клініці.

Ймовірність. Ми вже згадували вище про можливість того, що певне значення відбудеться. Більшість людей мають інтуїтивне розуміння ймовірності, часто зіштовхуючись із цим поняттям у прогнозах погоди (імовірність дощу) чи спортивному аналізі (ймовірність перемоги). Спорт та ігри найчастіше виражаються як шанси, або переваги, які легко конвертуються у ймовірності (якщо шанси на перемогу команди переважують і дорівнюють 2 до 1, то її ймовірність виграшу дорівнює 2/(2 + 1) = 2/3).

2.4.1 Стовпчикові діаграми

У разі категоріальних даних уявлення про них дають прості частки чи відсоткові співвідношення.

Отримання зведеної інформації про двійковій чи категоріальної змінної з кількома категоріями є досить простою задачею: з'ясовуємо частку одиниць чи частки важливих категорій. Наприклад, зібрано дані про відсоток затриманих рейсів через проблеми в аеропорту Даллас/Форт-Уерт, починаючи з 2010 року. Затримки розподілені за категорією в силу факторів, пов'язаних з перевізником (Carrier), системними затримками при управлінні повітряним рухом (ATC), погодних умов (Weather), міркувань безпеки (Security) або запізнення повітряного судна (Inbound), що прибуває.

Стовпчикові діаграми – це загальноприйнятий візуальний інструмент для відображення однієї-єдиної категоріальної змінної, який можна часто зустріти в популярній пресі. Категорії перераховані на осі x, а частоти чи частки – на осі y. На рис. 2.5 показані річні затримки авіарейсів через проблеми в аеропорту Даллас/Форт-Уерт .

Код

```
import pandas as pd
import pandas as pd
import matplotlib.pyplot as plt

dfw = pd.read_csv('dfw_airline.csv')
print(100 * dfw / dfw.values.sum())
```

```
ax = dfw.transpose().plot.bar(legend=False)
ax.set_xlabel('Cause of delay')
ax.set_ylabel('Count')

plt.tight_layout()
plt.show()
```

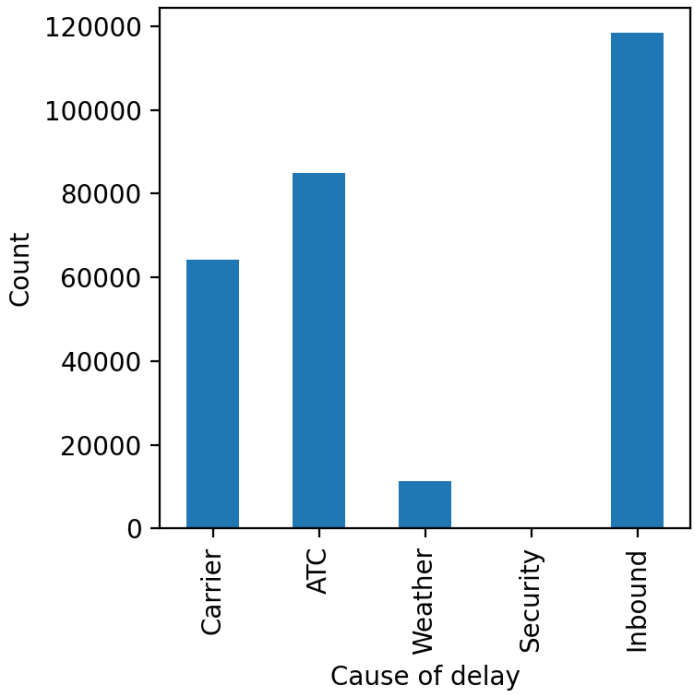


Рис. 2.5 – Стовпчикова діаграма.

Зазначимо, що стовпчикова діаграма нагадує гістограму. На стовпчикової діаграмі вісь x представляє різні категорії факторної змінної, тоді як на гістограмі вісь x представляє значення однієї-єдиної змінної на числовій шкалі. На гістограмі стовпчики зазвичай зображуються впритул один до одного, а розриви вказують на те, що значення даних відсутні. На стовпчиковій діаграмі стовпчики відображаються окремо один від одного.

Кругові діаграми є альтернативою стовпчиковим діаграмам, хоча статистики та експерти з візуалізації даних зазвичай уникають кругових діаграм, як менш візуально інформативних.

Числові дані як категоріальні дані. Раніше ми розглянули частотні таблиці на основі розбивки даних на кошики, у результаті числові дані неявно конвертувалися в упорядкований фактор. У цьому сенсі гістограми та стовпчикові діаграми подібні, за одним винятком — категорії на осі x у стовпчиковій діаграмі не впорядковані. Конвертування числових даних у категоріальні є важливим і широко використовується етапом в аналізі даних, оскільки ця процедура зменшує

складність (і розмір) даних. Вона допомагає виявляти зв'язок між ознаками, особливо у початкових стадіях аналізу.

2.4.2 Кругові діаграми

Кругова діаграма – це візуальне скорочення для «наскільки велика кожна частина цілого?». Вона найкраще працює, коли у вас є одна категоріальна змінна, яка підсумовується або підраховується, і ви хочете показати відносний внесок кожної категорії в загальну суму. Типовими прикладами використання є звіти про частку ринку, розподіл бюджету або відповіді на опитування, де питання має обмежену кількість варіантів відповідей.

Оскільки діаграма кодує розмір як кутовий розмах, людське око не так точно оцінює площу, як порівнює довжини (як у гістограмі). Отже, кругову діаграму слід використовувати в ситуаціях, коли:

- Кількість секторів невелика — зазвичай менше шести або семи.
- Різниця між сегментами є значною; сегмент, який становить лише кілька відсотків від загальної кількості, важко помітити.
- Ви передаєте одноразовий знімок, а не тенденцію в часі.

Приклад кругової діаграми для набору даних продажів іграшок за категоріями показано на рис. 2.6.

Код

```
import pandas as pd
import matplotlib.pyplot as plt

toys = pd.read_csv('toys.csv', index_col=0)
ax = toys['sales_usd'].plot.pie()

plt.tight_layout()
plt.show()
```

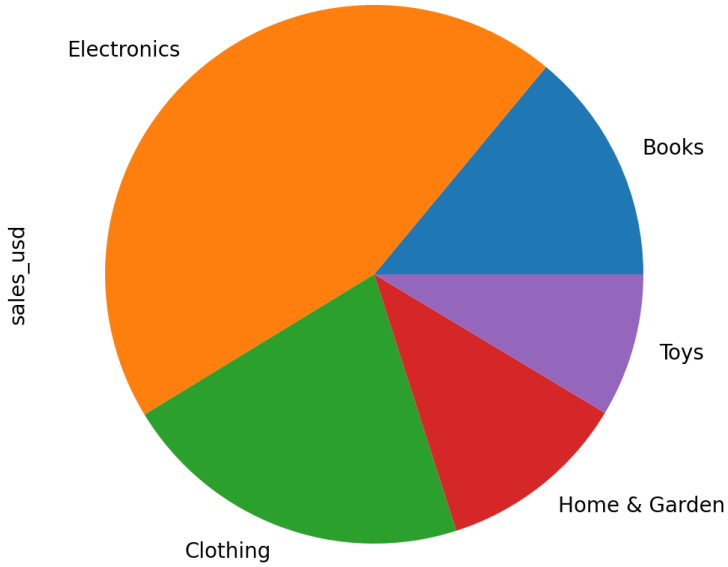


Рис. 2.6 – Кругова діаграма.

2.5 Кореляція

Розвідувальний аналіз даних у багатьох проектах моделювання (чи то в науці про дані або в статистичному дослідженні) передбачає обстеження кореляції серед провісників і між провісниками та цільовою змінною. Прийнято говорити, що змінні X та Y (кожна з вимірюваними даними) корелюють позитивно, якщо високі значення X супроводжуються високими значеннями Y та низькі значення X супроводжуються низькими значеннями Y. Якщо високі значення X супроводжуються низькими значеннями Y, і навпаки, то змінні корелюють негативно .

Розглянемо ці дві змінні, ідеально кореловані у тому сенсі, що кожна рухається паралельно від низького значення до високого: $v_1 = \{1,2,3\}, v_2 = \{4,5,6\}$.

Векторна сума добутків дорівнює $1 \cdot 4 + 2 \cdot 5 + 3 \cdot 6 = 32$. Тепер спробуємо перетасувати одну з них і обчислити повторно - векторна сума добутків ніколи не буде більшою за 32. Тому ця сума добутків може використовуватися як метрика; тобто спостережувану суму, рівну 32, можна порівнювати з численними випадковими перетасовуваннями. Значення, що виробляються цією метрикою, проте не особливо змістовні, крім як з посиланням на розподіл повторних вибірок.

Найкорисніший стандартизований варіант – це коефіцієнт кореляції, що дає оцінку кореляції між двома змінними, які завжди знаходяться на однаковій шкалі вимірювання. Для того, щоб обчислити коефіцієнт кореляції Пірсона, ми множимо відхилення від середнього для змінної 1 на відхилення для змінної 2, а потім ділимо результат на добуток стандартних відхилень:

$$r = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{(n-1)s_x s_y} \quad (2.1)$$

Зверніть увагу, що ми ділимо на $n-1$, а не на n . Коефіцієнт кореляції завжди знаходиться між $+1$ (ідеальна позитивна кореляція) та -1 (ідеальна негативна кореляція); 0 свідчить про відсутність кореляції.

Змінні можуть мати нелінійний зв'язок, і в цьому випадку коефіцієнт кореляції може не бути корисною метрикою. Зв'язок між податковими ставками та надходженнями, отриманими за рахунок податків, є прикладом: коли податкові ставки збільшуються від 0, податкові надходження теж збільшуються. Однак, як тільки податкові ставки досягають високого рівня та наближаються до 100%, ухилення від сплати податків збільшується, і податкові надходження зменшуються.

Інші оцінки кореляції. У статистиці давно були запропоновані інші типи коефіцієнтів кореляції, такі як коефіцієнт рангової кореляції Спірмена ρ (ро) або коефіцієнт рангової кореляції Кенделла τ (тау). Ці коефіцієнти кореляції ґрунтуються на ранзі даних, тобто номерах спостережень у наборі. Оскільки вони працюють з рангами, а не значеннями, ці оцінки робастні до викидів і можуть впоратися з деякими типами нелінійності. Однак дослідники даних для розвідувального аналізу зазвичай можуть дотримуватися коефіцієнта кореляції Пірсона та його робастних альтернатив. Рангові оцінки привабливі головним чином у разі малих наборів даних та специфічних перевірок гіпотез.

2.5.1 Кореляційна матриця

У табл. 2.4, яка називається кореляційною матрицею, показано кореляцію між щоденним доходом від акцій телекомунікаційних компаній з липня 2012 року по червень 2015 року. З таблиці видно, що Verizon (VZ) та ATT (T) мають найвищу кореляцію. Інфраструктурна компанія Level Three (LVLT) має найнижчу кореляцію. Зверніть увагу на діагональ з одиниць (кореляція акції із самою собою дорівнює 1) та надмірність інформації вище та нижче діагоналі.

Код

```
import pandas as pd
import matplotlib.pyplot as plt

telecom = pd.read_csv('telecom.csv', index_col=0)
print(telecom.head())
print(telecom.corr())
```

Табл. 2.4. Матриця кореляції.

	T	CTL	FTR	VZ	LVLT
T	1,000	0,475	0,328	0,678	0,279
CTL	0,475	1,000	0,420	0,417	0,287

FTR	0,328	0,420	1,000	0,287	0,260
VZ	0,678	0,417	0,287	1,000	0,242
LVLT	0,279	0,287	0,260	0,242	1,000

2.5.2 Діаграма розсіювання

Стандартним засобом візуалізації зв'язку між двома змінними з вимірюваними даними є діаграма розсіювання, чия вісь x представляє одну змінну, вісь y іншу, а кожна точка на графіці є записом. Подивимося на рис 2.7 з графіком, що містить щоденні фінансові повернення акцій ATT та Verizon.

Фінансові повернення мають сильний позитивний зв'язок: у більшості торгових днів обидві акції рухаються вгору або вниз у тандемі (правий верхній та лівий нижній квадранти). Існує менше днів, коли ціна на одну акцію значно падає, в той час як ціна на іншу акцію зростає, і навпаки (правий нижній та лівий верхній квадранти).

Тоді як на рис. 2.7 показано лише 754 точки даних, очевидно, наскільки складно ідентифікувати деталі у центрі графіка.

Код

```
import pandas as pd
import matplotlib.pyplot as plt

telecom = pd.read_csv('telecom.csv', index_col=0)

ax = telecom.plot.scatter(x='T', y='VZ')
ax.set_xlabel('ATT (T)')
ax.set_ylabel('Verizon (VZ)')

plt.tight_layout()
plt.show()
```

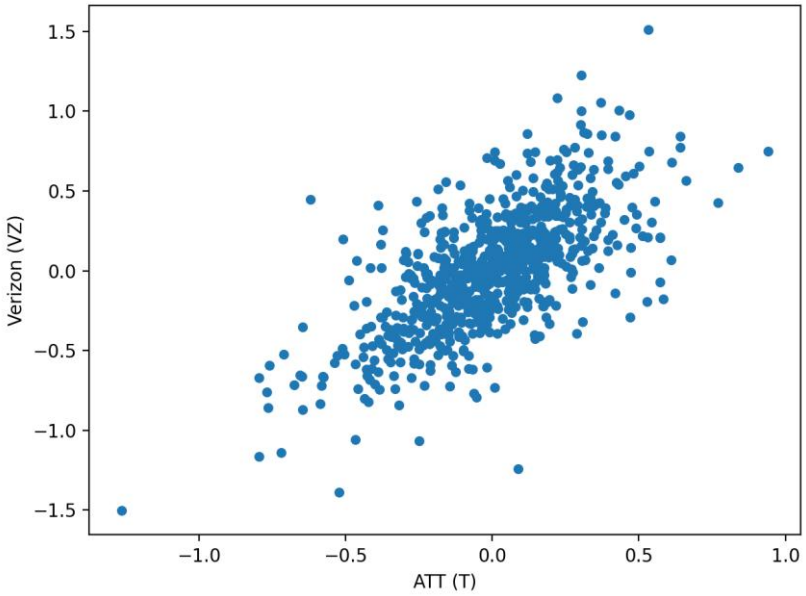


Рис. 2.7 – Діаграма розсіювання.

2.6 Контрольні питання

1. У чому полягає основна відмінність між лінійним графіком та гістограмою?
2. Чому для відображення категоріальних даних краще використовувати гістограму, а не кругову діаграму, як зазначено в тексті?
3. Що представляє мода в наборі даних і коли вона особливо корисна для категоріальних даних?
4. Як обчислюється очікуване значення для категоріальної змінної з різними ймовірностями, як показано в прикладі хмарної служби?
5. Який діапазон значень може приймати коефіцієнт кореляції Пірсона?
6. Чому коефіцієнт кореляції Пірсона може вводити в оману, коли між змінними існує нелінійна залежність?
7. Яка основна мета графіка щільності порівняно з гістограмою, і чому загальна площа під графіком щільності дорівнює 1?
8. Що представляє діагональ кореляційної матриці, і чому вона містить усі значення 1,0?
9. Як діаграма розсіювання допомагає візуалізувати взаємозв'язок між двома числовими змінними, і який закономірність ви очікуєте побачити при сильній позитивній кореляції?
10. Що означає коефіцієнт кореляції 0 для взаємозв'язку між двома змінними?
11. Чому дослідник може вибрати коефіцієнт кореляції Спірмена замість коефіцієнта кореляції Пірсона?
12. Яке основне обмеження використання кругових діаграм для візуалізації даних?
13. Як перетворення числових даних у категоріальні дані (за допомогою бінінгу) допомагає в аналізі даних, як обговорюється в тексті?
14. Як би ви інтерпретували коефіцієнт кореляції +0,95 між двома змінними?

Лекція 3 – Побудова та підгонка лінійної регресії

3.1 Проста лінійна регресія

Важливим питанням аналізу даних є відповідь на запитання: чи пов'язана змінна X (або, що більш ймовірно, $X_1, \ldots, X_p$) зі змінною Y, і якщо так, то в чому цей зв'язок полягає, і чи можемо ми його використати для того, щоб передбачити Y?

Проста лінійна регресія, або парна лінійна регресія, моделює зв'язок між величиною однієї змінної та величиною другої, наприклад у міру збільшення X збільшується і Y. Або ж у міру збільшення X зменшується і Y. Кореляція – ще один спосіб виміряти те, яким чином дві змінні зв'язані між собою. Різниця між ними полягає в тому, що кореляція вимірює силу зв'язку між двома змінними, тоді як регресія оцінює природу зв'язку кількісно.

3.2 Рівняння регресії

Проста лінійна регресія оцінює, наскільки саме зміниться Y коли X змінюється на деяку величину. Для коефіцієнта кореляції змінні X та Y взаємозамінні. У разі регресії ми намагаємося передбачити змінну Y за змінною X, використовуючи лінійний зв'язок (тобто прямий):

$$Y = b_0 + b_1 X. \tag{3.1}$$

Ця формула читається, як «Y дорівнює b_1, помножене на X плюс константа b_0». Компонент рівняння b_0 називається перетином (або константою), а b_1 – нахилом по відношенню до осі x. Змінна Y називається *відгуком* або *залежною змінною*, оскільки вона залежить від X. Змінна X називається *провісником* (предиктором, від англ. predictor), або *незалежною змінною*. Спільнота машинного навчання тяжіє до використання інших термінів, називаючи Y *ціллю* та X – *вектором ознак*.

3.3 Приклад: визначення впливу бавовняного пилу на легені

Розглянемо діаграму розсіювання на рис. 3.1, що показує число років, протягом яких робітник зазнав впливу бавовняного пилу (Exposure) проти показника об'єму легень (PEFR – «пікова об'ємна швидкість видиху»). Яким чином змінна PEFR пов'язана з Exposure? Важко сказати щось конкретне, просто дивлячись на зображення.

Код

```
import pandas as pd
import matplotlib.pyplot as plt

lung = pd.read_csv('LungDisease.csv')

lung.plot.scatter(x='Exposure', y='PEFR')

plt.tight_layout()
plt.show()
```

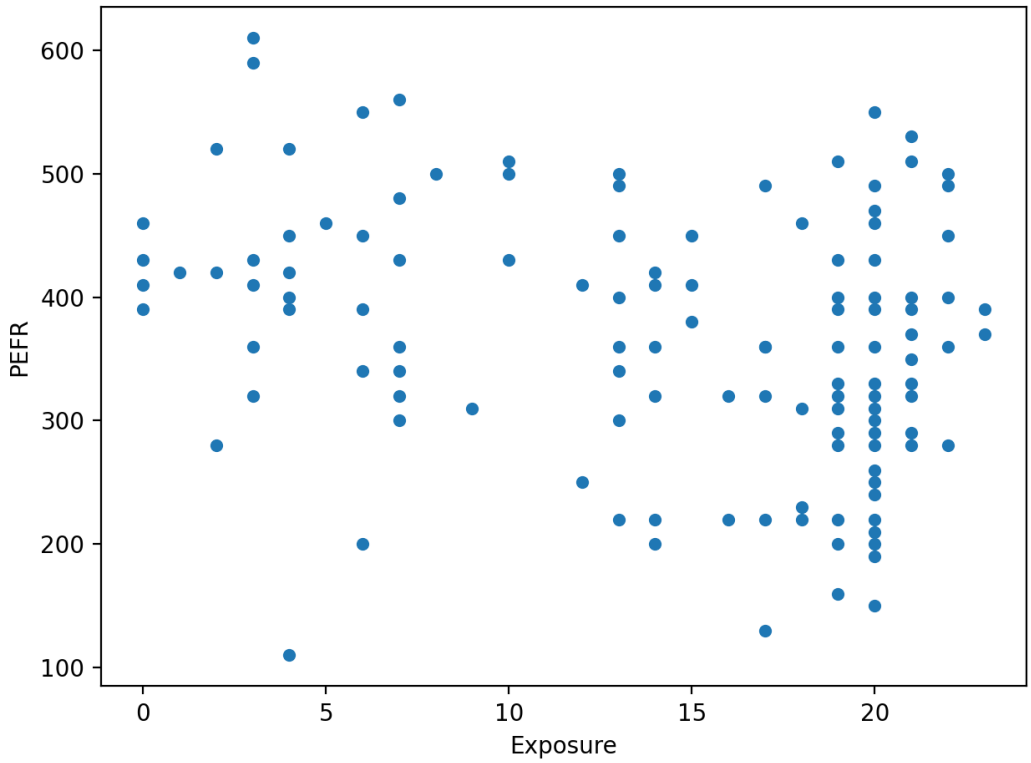


Рис. 3.1 – Діаграма розсіювання пікової об'ємної швидкість видиху.

Проста лінійна регресія намагається відшукати «оптимальну» пряму, щоб передбачити відгук PEFR, як функцію від передбачувальної змінної Exposure:

$$\text{PEFR} = b_0 + b_1 Exposure, \tag{3.2}$$

де b_0 – перетин, b_1 – коефіцієнт регресії.

Для навчання лінійної регресії в Python використовується клас LinearRegression з бібліотеки scikit-learn. Для навчання моделі використовується метод fit, в який передаються незалежні змінні (вектор ознак) та залежна змінна навчальної вибірки. Для передбачення використовується метод predict.

Код

```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression


lung = pd.read_csv('LungDisease.csv')

lung.plot.scatter(x='Exposure', y='PEFR')

plt.show()

predictors = ['Exposure']
outcome = 'PEFR'

model = LinearRegression()
model.fit(lung[predictors], lung[outcome])

print('Intercept')
print(model.intercept_)
print('Coefficient Exposure')
print(model.coef_[0])

lung.plot.scatter(x='Exposure', y='PEFR')
plt.plot(lung['Exposure'], model.predict(lung[predictors]))

plt.show()
```

Результат навчання моделі показано на рис. 3.2. Тепер стає очевидно, що пікова об’ємна швидкість видиху залежить від кількості років вдихання бавовняного пилу на фабриці.

Отримані коефіцієнти становлять: константа: 424.583, нахил: -4.185.

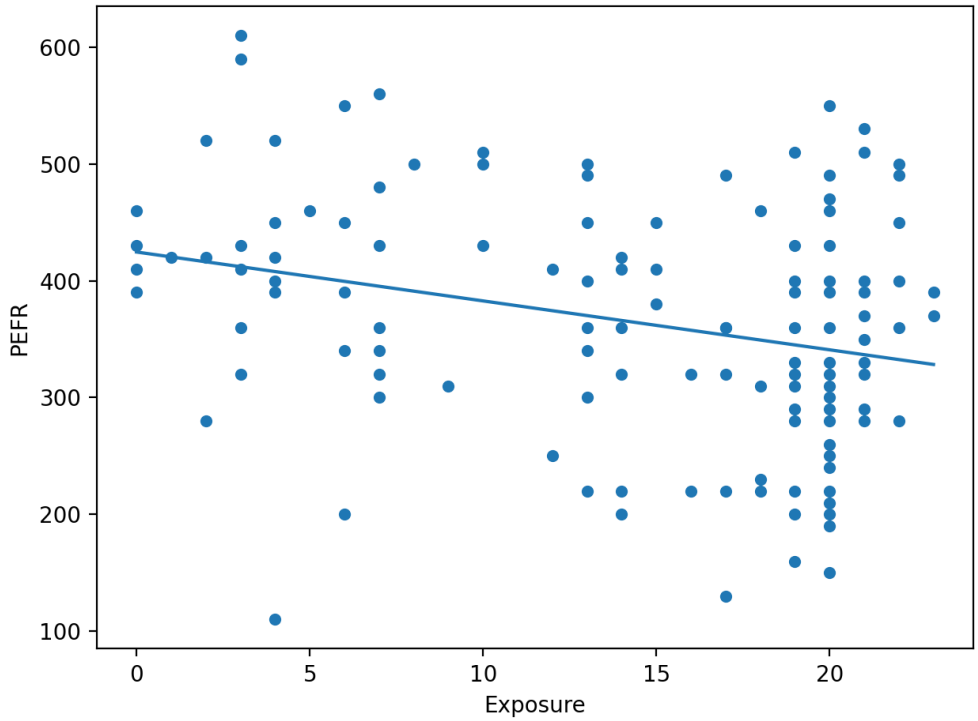


Рис. 3.2 – Підігнана пряма регресії.

3.3 Підігнані значення та залишки

У регресійному аналізі важливими поняттями є *підігнані значення* та *залишки*. Як правило, дані не лягають рівно на пряму, тому рівняння регресії має включати явний залишковий член e_i:

$$Y_i = b_0 + b_1 X_i + e_i. \tag{3.3}$$

Підігнані значення, також звані передбаченими значеннями, у типовій ситуації позначаються як $\hat{Y}_i$ (Y з капелюхом). Вони задаються такою формулою:

$$\hat{Y}_i = \hat{b}_0 + \hat{b}_1 X_i. \tag{3.4}$$

Позначення $\hat{b}_0$ і $\hat{b}_1$ свідчить, що ці коефіцієнти оцінюються відносно відомим, тобто є оціночними.

- Позначення з «капелюхом» використовується для диференціації між оцінками та відомими значеннями.
    - Так, символ $\hat{b}$ (b з капелюхом) є оцінкою невідомого параметра b.
        - Чому в статистиці проводиться відмінність між оцінкою та істинним значенням?
        - Оцінка має невизначеність, тоді як істинне значення фіксоване.
- Ми обчислюємо залишки $\hat{e}_i$ шляхом віднімання передбачених значень від вихідних даних:
    - $\hat{e}_i = Y_i - \hat{Y}_i$.

На рис. 3.3 показано залишки підігнаних значень.

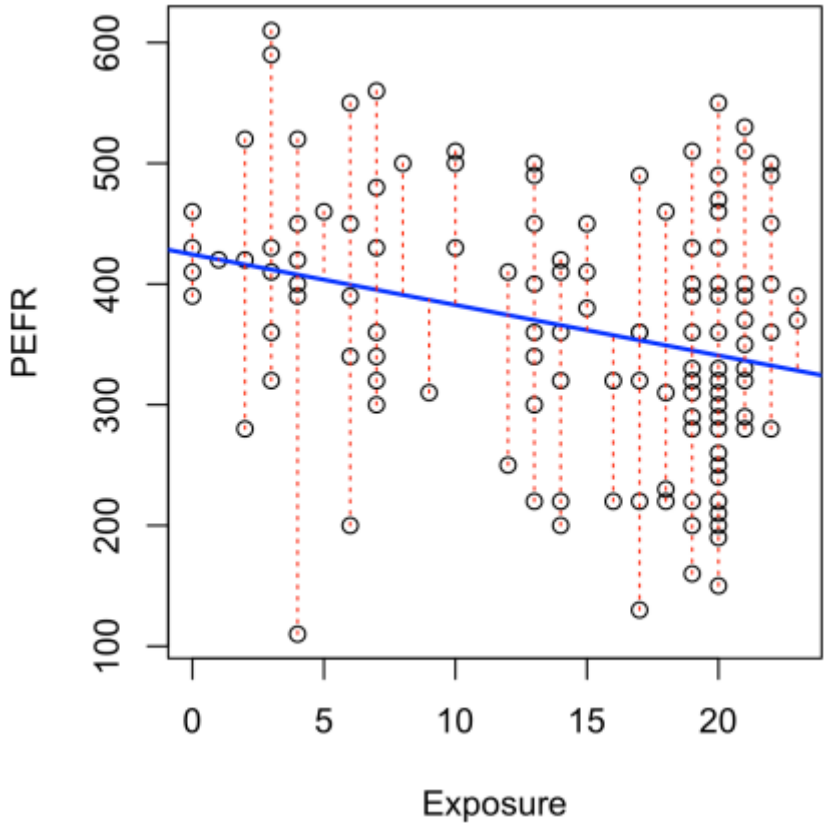


Рис. 3.3 – Залишки підігнаних значень.

3.4 Найменші квадрати

Яким чином виконується підгонка моделі до даних? Коли існує чіткий зв'язок, ви можете подумки уявити підгонку прямої вручну. Насправді пряма регресії є оцінкою, яка мінімізує значення суми квадратичних залишків, також іменованих *сумою квадратів залишків* чи *залишковою сумою квадратів* (residual sum of squares, RSS):

$$RSS = \sum_{i=1}^{n} \left(Y_i - \hat{Y}_i\right)^2 = \sum_{i=1}^{n} \left(Y_i - \hat{b}_0 - \hat{b}_1 X_i\right)^2 \qquad (3.5)$$

Оцінки $\hat{b}_0$ та $\hat{b}_1$ – це значення, які мінімізують суму квадратів залишків (RSS).

Метод мінімізації суми квадратів залишків називається *регресією на основі найменших квадратів*, чи регресією *на основі звичайних найменших квадратів* (ЗНК). Цей метод часто приписується Карлу Фрідріху Гауссу, німецькому математику, але він був вперше опублікований в 1805 французьким математиком Андре-Марі Лежандром (Adrien-Marie Legendre). Регресію на основі найменших квадратів можна легко та швидко обчислити за допомогою стандартної статистичної обчислювальної системи.

Історично, обчислювальна зручність найменших квадратів є однією з причин широкого застосування цього методу в регресії. З появою великих даних обчислювальна швидкість, як і раніше, залишається важливим фактором. Найменші квадрати, як і середнє значення, як метод чутливий до викидів, хоча цей факт має тенденцію бути значною проблемою тільки в малих або помірних за розміром наборах даних.

З появою великих даних регресія широко використовується для формування моделі з метою передбачення індивідуальних наслідків для нових даних замість статистичного пояснення даних, наявних під рукою (тобто передбачувальної моделі). У цьому випадку головними елементами, що цікавлять, є підігнані значення $\hat{Y}$.

У маркетингу регресія може використовуватися для передбачення зміни в доході у відповідь на розмір рекламної кампанії. Університети використовують регресію для передбачення середнього академічного бала GPA студентів на основі їхніх балів за іспит на визначення академічних здібностей SAT.

Регресійна модель, яка підігнана до даних добре, налаштована так, що зміни в X призводять до змін в Y. Однак саме по собі рівняння регресії не доводить напрямок причинно-наслідкової обумовленості. Висновки про причинно-наслідкову обумовленість слід робити, виходячи з ширшого контексту розуміння зв'язку. Наприклад, рівняння регресії могло б показати певний зв'язок між числом натискань на веб-рекламі та числом конверсій. Саме наше знання маркетингового процесу, а не рівняння регресії приводить нас до висновку про те, що натискання на рекламі генерують продажі, і не навпаки.

Термінологія регресії. Коли аналітики та дослідники використовують термін «регресія» сам по собі, вони зазвичай мають на увазі лінійну регресію; у центрі уваги зазвичай перебуває розробка лінійної моделі для пояснення зв'язку між передбачувальними змінними та числовою змінною результату. У своєму формальному статистичному сенсі регресія також охоплює нелінійні моделі, які виробляють функціональний зв'язок між передбачувальними змінними і змінною результату. У спільноті машинного навчання цей термін також часом використовують у широкому сенсі для посилання на використання будь-якої прогностичної моделі, що виробляє передбачуваний числовий результат (на відміну від класифікаційних методів, які передбачають двійковий або категоріальний результат).

Ключові терміни для простої лінійної регресії

- Відгук
    - Змінна, яку ми намагаємось передбачити.
    - *Синоніми:* залежна змінна, Y-змінна, ціль, результат.
- Незалежна змінна

    - Змінна, яка використовується для передбачення відгуку.
    - *Синоніми:* X-змінна, провісник, предиктор, передбачувальна змінна, ознака, атрибут.
- Запис
    - Вектор, який складається з значень провісників і значення результату для окремого елемента даних чи випадку.
    - *Синоніми:* рядок, випадок, прецедент, зразок, екземпляр, приклад.
- Перетин
    - Перетин регресійної прямої з віссю y, тобто передбачене значення, коли $X = 0$.
    - *Синоніми:* b_0, β_0, точка перетину, коефіцієнт зсуву на осі y.
- Коефіцієнт регресії
    - Нахил регресійної прямої по відношенню до осі x.
    - *Синоніми:* нахил, b_1, β_1, оцінки параметрів, ваги, кутовий коефіцієнт.
- Підігнані значення
    - Оцінки $\hat{Y}_i$, отримані з регресійної прямої.
    - *Синонім:* передбачені значення.
- Залишки
    - Різниця між значеннями, що спостерігаються, і підігнаними значеннями.
    - *Синонім:* помилки.
- Найменші квадрати
    - Метод підгонки регресії шляхом мінімізації суми квадратів залишків.
    - *Синоніми:* стандартний метод найменших квадратів, стандартний МНК.

3.5 Множинна лінійна регресія

Коли провісників кілька, рівняння регресії просто розширюється їх розміщення:

$$Y = b_0 + b_1X_1 + b_2X_2 + \cdots + b_pX_p + e. \quad (3.6)$$

Замість прямої тепер у нас лінійна модель – зв'язок між кожним коефіцієнтом та його змінною (ознакою) є лінійним.

Всі інші поняття з простої лінійної регресії, такі як підгонка найменшими квадратами, підігнані значення та залишки, відносяться і до умов множинної лінійної регресії. Наприклад, підігнані значення задаються наступною формулою:

$$\hat{Y}_i = \hat{b}_0 + \hat{b}_1X_{1,i} + \hat{b}_2X_{2,i} + \cdots + \hat{b}_pX_{p,i}. \quad (3.7)$$

Ключові терміни для множинної лінійної регресії

- Формула множинної лінійної регресії:

$$Y = b_0 + b_1X_1 + b_2X_2 + \cdots + b_pX_p + e$$

- Підігнані значення:

$$\hat{Y}_i = \hat{b}_0 + \hat{b}_1X_{1,i} + \hat{b}_2X_{2,i} + \cdots + \hat{b}_pX_{p,i}$$

- Корінь із середньоквадратичної помилки:
  - Квадратний корінь із середньоквадратичної помилки регресії (це найбільш широко використовується метрика порівняння регресійних моделей).
  - *Синонім:* RMSE.
- Стандартна помилка залишків:
  - Те саме, що і середньоквадратична помилка, але скоригована для степенів свободи.
  - *Синонім:* RSE.
- R-квадрат
  - Частка дисперсії, яка пояснюється моделлю, зі значеннями в інтервалі від 0 до 1.
  - *Синоніми:* коефіцієнт детермінації, R^2.
- t-Статистика
  - Коефіцієнт для провісника, поділений на стандартну помилку коефіцієнта, що дає метрику для порівняння важливості змінних моделі
- Зважена регресія
  - Регресія, в якій записам поставлені у відповідність різні ваги

3.6 Оцінювання результативності моделі

Найважливішою метрикою результативності з погляду науки даних є *корінь із середньоквадратичної помилки* (RMSE). RMSE – це квадратний корінь із середньоквадратичної помилки в передбачених $\hat{y}_i$ значеннях:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}}. \tag{3.8}$$

Вона вимірює сукупну точність моделі і є основою її порівняння з іншими моделями (включаючи моделі, підігнані з допомогою спеціальних технічних прийомів машинного навчання). Схожою на RMSE є *стандартна помилка*

залишків (RSE). У цьому випадку ми маємо p провісників, RSE задається такою формулою:

$$\text{RSE} = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n-p-1}}. \tag{3.9}$$

Єдина різниця полягає в тому, що знаменник є ступенями свободи, на противагу числу записів (p – кількість змінних). Насправді різниця між RMSE і RSE для лінійної регресії є дуже малою, особливо для застосунків великих даних.

Ще одна корисна метрика, яку ви побачите на виході з обчислювальних систем, називається *коефіцієнтом детермінації* або *статистикою R-квадрат* або R^2. R-квадрат варіює в інтервалі від 0 до 1 і вимірює частку варіації даних, пояснювану в моделі. Він корисний головним чином в пояснювальних застосуваннях регресії, де ви хочете визначити, наскільки добре модель підігнана до даних. Формула для R^2 така:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2}. \tag{3.10}$$

Знаменник пропорційний дисперсії відгуку Y.

Оцінювання результативності моделі

- Корінь із середньоквадратичної помилки (RMSE):

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}}.$$

- Стандартна помилка залишків (RSE):

$$\text{RSE} = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n-p-1}}.$$

- Коефіцієнт детермінації (статистика R^2):

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2}.$$

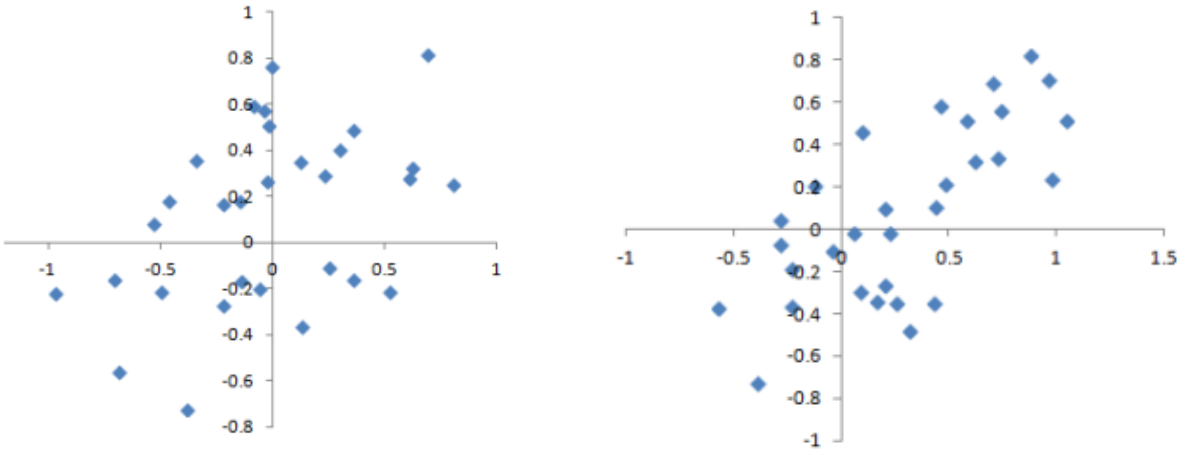
3.7 Контрольні питання

1. Що таке «відгук» у лінійній регресії?
2. Як називається змінна, що використовується для передбачення відгуку?
3. Яке рівняння описує просту лінійну регресію?
4. Що означає коефіцієнт β_1 у рівнянні регресії?
5. Що таке «залишки» у регресійному аналізі?
6. Що мінімізує метод найменших квадратів?
7. Яку інтерпретацію має від'ємний коефіцієнт регресії (наприклад, -4.185 у прикладі з PEFR)?
8. Чому регресія не доводить причинно-наслідкового зв'язку?
9. Що означає R^2 (коефіцієнт детермінації)?
10. Що таке «підігнані значення» $\hat{Y}$?
11. У якому випадку використовується множинна лінійна регресія? Яке рівняння множинної регресії?
12. Чому метод найменших квадратів чутливий до викидів?

Лекція 4 – Бутстрап. Довірчі інтервали

4.1 Зміщені дані та випадковий відбір

Статистичне зміщення відноситься до помилок виміру або відбору, які систематичні і породжуються процесом виміру або відбору даних. Важливо проводити різницю між помилками внаслідок випадковості та помилками внаслідок зміщення. Розглянемо фізичний процес стрілянини з рушниці по мішені. Не слід очікувати потрапляння в абсолютний центр мішені щоразу і або навіть що таке відбудеться взагалі. Незміщений процес призведе до помилки, але вона буде випадковою і не виявить тенденцію до будь-якого з напрямків (рис. 4.1а). Наведені на рис. 4.1б результати показують зміщений процес – як і раніше є випадкова помилка як у напрямку x, так і в напрямку y, але крім цього є зміщення. Постріли демонструють тенденцію потрапляти у правий верхній квадрант .



(а) (б)

Рис. 4.1 – Діаграма розсіювання стрілянини з рушниці.

Зміщення виникає у різних формах і може бути помітним чи невидимим. Коли результат дійсно наводить на думку про зміщення (наприклад, спираючись на еталон або фактичні значення), це сигналізує про те, що статистична або модель, що автоматично навчається, була задана неправильно або важлива змінна була втрачена з поля зору.

4.1.1 Випадковий відбір

Щоб уникнути проблеми зміщеної вибірки сьогодні існують різноманітні методи, які дозволяють досягати репрезентативності, але в їх основі лежить *випадковий відбір* .

Організувати випадковий відбір не завжди просто, і належне визначення досяжної популяції є ключем. Припустимо, що ми хочемо згенерувати

репрезентативний профіль покупців, і нам потрібно провести їхній пілотний статистичний огляд. Огляд має бути репрезентативним, але він трудомісткий.

Спочатку ми маємо визначити, хто є покупцем. Ми могли б відібрати всі записи покупців із сумою покупки, що перевищує 0. Чи варто включати всіх попередніх покупців? Чи варто включати компенсації? Внутрішні тестові покупки? Перекупників? Як білінгового агента, і покупця?

Далі ми маємо визначити процедуру відбору. Вона може полягати в тому, щоб «відібрати 100 покупців навмання». Там, де задіяний відбір із потоку (наприклад, реально-часові транзакції покупців або відвідувачі веб-сайту), особливу важливість приймають міркування щодо часу (наприклад, відвідувач веб-сайту о 10:00 будня може відрізнятися від відвідувача веб-сайту в 22:00 у вихідні).

У *стратифікованому відборі* населення поділяється на страти, і випадкові вибірки беруться з кожної *страти* . Політичні соціологи могли б спробувати з'ясувати електоральні переваги білих, чорношкірих та латиноамериканців. Проста випадкова вибірка, взята з населення США, призвела б до дуже малої кількості афроамериканців і латиноамериканців, і тому в стратифікованому відборі цим стратам могла б бути надана більша вага для отримання еквівалентних розмірів вибірок.

4.1.2 Розмір проти якості: коли розмір має значення?

В епоху великих даних іноді викликає подив твердження «Чим менше, тим краще». Час і зусилля, що витрачаються на випадковий відбір зменшують зміщення, але й дозволяють приділяти більше уваги розвідування даних і якість даних. Наприклад, пропущені дані та викиди можуть містити корисну інформацію. Розшук відсутніх значень або обчислення викидів у мільйонах записів можуть мати недозволено дорогу вартість, але ця робота у вибірці, що складається з кількох тисяч записів, цілком здійсненна. Виведення даних на графіки та ручне обстеження захлинаються, якщо даних надто багато.

Коли ж *потрібні* потужні обсяги даних?

Класичний сценарій важливості великих даних передбачає, що не лише великі, а й розріджені. Візьмемо, наприклад, пошукові запити, які отримують Google, де стовпці – це умови, рядки – окремі пошукові запити, а значення осередків дорівнюють 0 або 1 залежно від того, чи містить запит термін чи ні. Завдання полягає в тому, щоб найкраще визначити передбачене призначення пошуку для заданого запиту. В англійській мові понад 150 тис. слів, і Google опрацьовує понад 1 трлн запитів на рік. В результаті вийде величезна матриця, переважна більшість записів в якій дорівнюватимуть 0.

За своїм розмахом це завдання справді відноситься до завдань обробки великих даних – на більшість запитів можуть бути повернені ефективні пошукові результати лише тоді, коли накопичені такі величезні обсяги даних. І чим більше даних накопичується, тим кращі результати. Для популярних критеріїв пошуку це

не така проблема – ефективні дані можна знайти досить швидко для невеликої кількості надзвичайно популярних тем, що знаходяться в тренді в той чи інший час. Дійсна важливість сучасної пошукової технології полягає у здатності повертати докладні та корисні результати для величезної різноманітності пошукових запитів, включаючи ті, які виникають із частотою, скажімо, всього один на мільйон.

4.1.3 Систематична помилка відбору

Систематична помилка відбору (або зміщення, упередженість при відборі – selection bias) відноситься до відбору даних усвідомлено або неусвідомлено таким чином, що це призводить до оманливого або ефемерного висновку.

Якщо ви визначаєте гіпотезу і проводите добре пророблений експеримент з метою її перевірки, то можете бути впевненим у висновку. Однак найчастіше це не так. Натомість часто дивляться на наявні дані у спробі розглянути регулярності. Але чи є регулярність реальною чи вона лише продукт *прочісування даних* , тобто докладної ревізії даних, доки з'явиться щось цікаве? Серед статистиків популярна приказка: «Якщо мучити дані надто довго, то рано чи пізно вони дадуть свідчення».

Різницю між явищем, у якому ви засвідчуєтесь, коли перевіряєте гіпотезу за допомогою експерименту, і явищем, яке ви виявляєте, переслідуючи наявні дані, можна роз'яснити наступним уявним експериментом.

Припустимо, що хтось говорить вам, що він може змусити приземлитися монету орлом, яку він підкидає, протягом наступних 10 кидків. Ви приймаєте виклик (еквівалент експерименту), і він приступає до 10-кратного підкидання монети, і щоразу монета приземляється орлом. Цілком очевидно, що ви припишіть цій людині якийсь особливий талант – ймовірність, що в результаті 10 кидків монети вона просто по чистій випадковості повернеться орлом, становить 1 з 1000.

Тепер припустимо, що диктор на стадіоні просить, щоб усі присутні 20 тис. людей підкинули монету 10 разів і повідомили працівника стадіону, якщо вони отримають 10 орлів поспіль. Шанс, що хтось на стадіоні дістанеться до 10 орлів, надзвичайно високий (понад 99% – це 1 мінус ймовірність того, що ніхто не отримає 10 орлів). Безумовно, відбір постфактум людини (або людей), яка отримала 10 орлів на стадіоні, не говорить про те, що вона має якийсь особливий талант — швидше за все, це просто удача.

Оскільки неодноразова ревізія великих наборів даних є в науці про дані ключовою ціннісною пропозицією, від якої важко відмовитися, систематичній помилці відбору слід приділяти особливу увагу. Форму систематичної помилки відбору, що має особливе значення для дослідників даних, Джон Елдер, засновник компанії Elder Research, шановної консалтингової компанії в галузі видобутку регулярностей даних, називає *ефектом безкрайнього пошуку* . Якщо ви неодноразово будуєте різні моделі і задаєте різні питання в умовах великих

наборів даних, то ви знайдете щось цікаве. Чи є знайдений результат по-справжньому чимось, що заслуговує на увагу чи це випадковий викид?

Проти цього можна вжити захисних заходів, задіявши контрольний набір з відкладеними даними, а іноді більше одного контрольного набору, на основі яких можна підтвердити результативність. Крім цього, Елдер також виступає за використання того, що він називає *перетасовуванням цілей* (по суті, це перестановний тест) для перевірки достовірності передбачуваних асоціацій, які пропонує модель видобутку регулярностей даних.

Типові форми систематичної помилки відбору в статистиці, на додаток до ефекту безкрайнього пошуку, включають невипадковий відбір, дані, одержувані в результаті відбору за принципом зняття вершків, відбір тимчасових інтервалів, які підкреслюють той чи інший статистичний ефект, і зупинку експерименту, коли результати виглядають «цікавими».

4.1.4 Явище регресії до середнього

Регресія до середнього значення відноситься до явища, пов'язаного з послідовними вимірами заданої змінної: граничні спостереження мають тенденцію супроводжуватися більш центральними значеннями. Надання особливої уваги та сенсу граничного значення може призвести до однієї з форм систематичної помилки відбору.

Регресія до середнього значення є наслідком окремої форми систематичної помилки відбору. Коли ми відбираємо новобранця з найкращою результативністю, навичка та удача, ймовірно, цьому сприяють. У свій наступний сезон навичка, як і раніше, буде на місці, але в більшості випадків успіх буде відсутній, і тому його результативність впаде – вона регресуватиме. Це явище було вперше ідентифіковано Френсісом Гальтоном в 1886, який описав його у зв'язку з генетичними тенденціями; наприклад, діти надзвичайно високих чоловіків схильні не бути такими ж високими, що й їхні батьки (рис. 4.2).

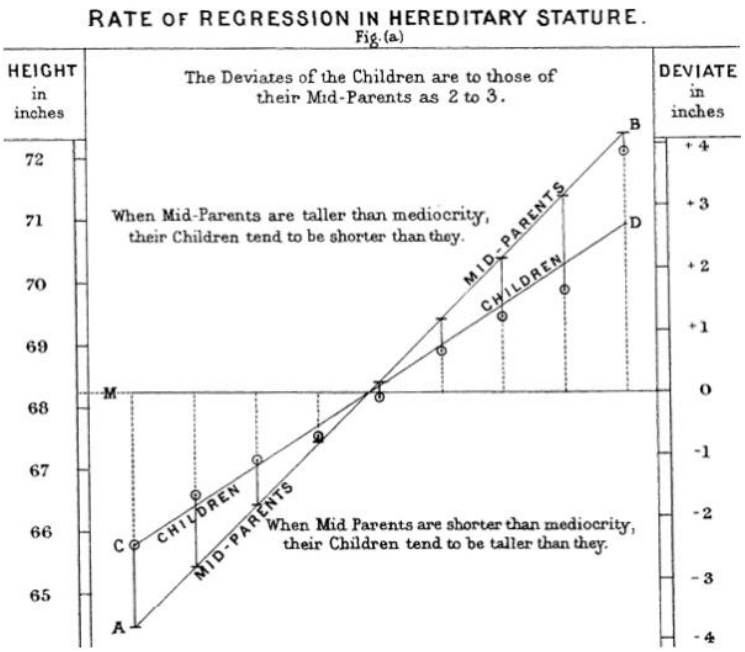


Рис. 4.2 – Регресія до середнього – дослідження Ф. Гальтона.

Важливо. Регресія, тобто «повернення назад», до середнього відрізняється від методу статистичного моделювання, такого як лінійна регресія, в якому лінійний зв'язок оцінюється між передбачуваними змінними та змінними результатами.

4.2 Вибірковий розподіл статистичної величини

Термін «*вибіркове розподіл»* статистичної величини позначає розподіл деякої вибіркової статистики над численними вибірками, які витягуються з однієї й тієї ж популяції. Значна частина класичної статистики займається отриманням статистичних висновків (малих) вибірок до (дуже великих) популяцій.

У типовій ситуації вибірка витягується з метою виміру чого-небудь (за допомогою *вибіркової статистики*) або моделювання чого-небудь (за допомогою статистичної або автоматичної моделі). Оскільки наша оцінка чи модель ґрунтується на вибірці, вона може бути помилковою; вона може бути інакшою, якщо ми вирішимо отримати іншу вибірку. Ми, отже, зацікавлені знати, наскільки вона може відрізнятися, ключовою проблемою є *вибіркова варіабельність*. Якби ми мали багато даних, ми могли б витягувати додаткові вибірки і спостерігати розподіл вибіркової статистики безпосередньо. Як правило, ми обчислюватимемо нашу оцінку або модель, використовуючи стільки даних, скільки їх є в наявності, так що можливість отримання додаткових вибірок з популяції є далеко не завжди.

Важливо. Важливо проводити різницю між розподілом індивідуальних точок даних, іменованим розподілом даних, і розподілом вибіркової статистики, іменованим вибірковим розподілом.

Розподіл вибіркової статистики, такий як середнє, ймовірно, буде більш регулярним і мати форму дзвону, ніж розподіл самих даних. Чим більша вибірка, на якій ґрунтується статистика, тим більше вона є істинною. Крім того, чим більша вибірка, тим вужчим є розподіл вибіркової статистики .

Це твердження ілюструється прикладом із використанням річного доходу для позикозаявників кредитного клубу Lending Club. Візьмемо з цих даних три вибірки: вибірку 1000 значень, вибірку 1000 середніх по 5 значень та вибірку 1000 середніх по 20 значень. Потім збудуємо гістограму кожної вибірки.

Гістограма індивідуальних значень даних широко розкидана та скошена до вищих значень, як і слід очікувати з даними про доходи. Обидві гістограми середніх з 5 і 20 значень мають більш компактний і більш дзвоновий вигляд.

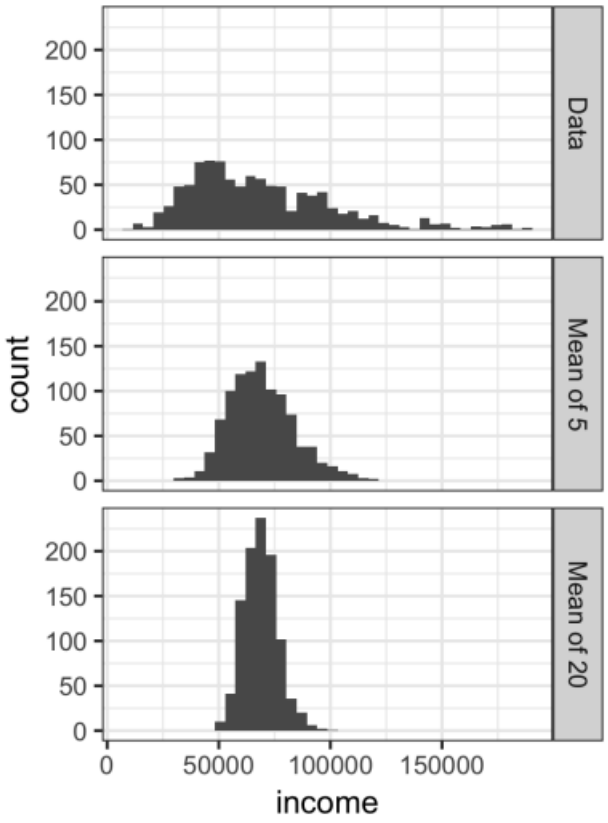


Рис. 4.3 – Розподіл середнього позикозаявників.

4.2.1 Центральна гранична теорема

Явище, яке ми щойно описали, називається *центральною граничною теоремою*. Воно каже, що середні значення, витягнуті з численних вибірок, будуть нагадувати знайому дзвоноподібну нормальну криву, навіть якщо вихідна популяція не є нормально розподіленою, за умови, що розмір вибірок досить великий і відступ даних від нормальності не надто великий. Центральна гранична теорема дозволяє використовувати нормально-апроксимаційні формули, такі як t-розподіл, що застосовуються у обчисленні розподілів вибірок для статистичного висновку, а саме довірчі інтервали та перевірки гіпотез.

У традиційних статистичних перевірках центральній граничній теоремі приділяється велика увага, тому що вона лежить в основі механізму довірчих інтервалів та перевірок гіпотез, які займають половину вмісту таких текстів. Дослідники даних повинні знати про цю її роль, але оскільки формальні перевірки гіпотез та довірчі інтервали відіграють у науці про дані незначну роль, і так чи інакше завжди є *бутстрап*, центральна гранична теорема не займає якесь особливе місце у практиці науки про дані.

4.2.2 Стандартна помилка

Стандартна помилка – це одиночна метрика, яка узагальнює варіабельність статистичної величини у вибірковому розподілі. Стандартну помилку можна оцінити з використанням статистичної величини, спираючись на стандартне відхилення s значень вибірки та розмір вибірки n.

У міру збільшення розміру вибірки стандартна помилка зменшується відповідно до того, що спостерігалося на попередньому рисунку. Зв'язок між стандартною помилкою та розміром вибірки іноді носить назву *правила квадратного кореня* з n: для скорочення стандартної помилки у 2 рази розмір вибірки має бути збільшений у 4 рази:

$$\text{Стандартна помилка} = \frac{s}{\sqrt{n}}, \tag{4.1}$$

де s – стандартне відхилення, n – розмір вибірки.

Достовірність формули стандартної помилки випливає із центральної граничної теореми. Насправді, вам не потрібно спиратися на центральну граничну теорему, щоб зрозуміти стандартну помилку. Розглянемо наступний підхід до вимірювання стандартної помилки:

1. Зібрати низку абсолютно нових вибірок із популяції.
2. По кожній новій вибірці вирахувати статистику (наприклад, середнє).
3. Розрахувати стандартне відхилення статистики, обчисленої на кроці 2; використовувати її як оцінку стандартної помилки.

На практиці цей підхід отримання нових вибірок для оцінювання стандартної помилки в типовій ситуації не здійснимий (і статистично дуже марнотратний). На щастя, як виявилося, немає необхідності отримувати нові вибірки; замість цього можна використовувати *бутстраповські* повторні вибірки. У сучасній статистиці бутстрап став типовим способом оцінювання стандартної помилки. Цей спосіб можна використовувати практично для будь-якої статистики. Він не спирається на центральну граничну теорему чи інші припущення про природу розподілу.

Важливо. Стандартне відхилення проти стандартної помилки . Не плутайте стандартне відхилення (яке показує варіабельність окремих точок даних) із стандартною помилкою (яка показує варіабельність вибіркової метрики).

4.3 Бутстрап та довірчі інтервали

Один з простих та ефективних способів оцінювання вибіркового розподілу статистичної величини або модельних параметрів полягає в тому, щоб витягувати додаткові вибірки з поверненням із самої вибірки та перераховувати статистику або модель кожної повторної вибірки. Дана процедура називається бутстрапом (від англ, bootstrap – розкрутка, самоналаштування), і вона не пов'язана з будь-якими припущеннями про те, що дані або вибіркова статистика нормально розподілені.

Концептуально ви можете представити бутстрап як реплікацію вихідної вибірки тисячі або мільйони разів з тим, щоб отримати гіпотетичну популяцію, яка втілює всі знання, виходячи з оригінальної вибірки (вона просто більша). Потім із цієї гіпотетичної популяції можна вибирати вибірки з метою оцінювання вибіркового розподілу (рис. 4.4).

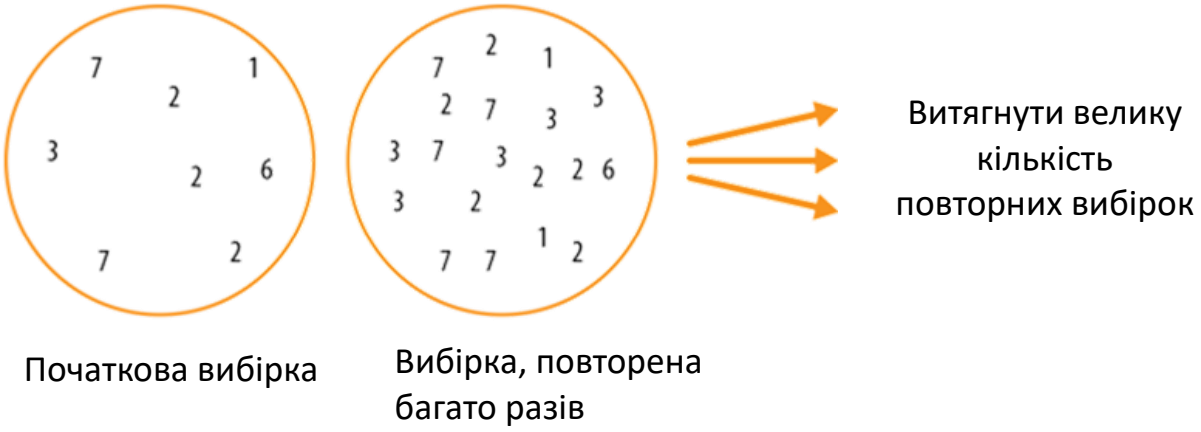


Рис. 4.4 – Створення повторних вибірок.

Концептуально повторний відбір бутстрапа є простим, і економіст і демограф Джуліан Саймон у своїй праці 1969 року «Методи фундаментального дослідження в соціології» опублікував резюме прикладів повторного відбору, включно з бутстрапом. Однак цей метод також є обчислювально ємним, і до початку широкого поширення обчислювальних потужностей він залишався фізично нездійсненною можливістю.

Метод отримав свою назву і набув популярності з опублікуванням книжки стенфордського статистика Бредлі Ефрона і завдяки кільком статтям у журналах наприкінці 1970-х і на початку 1980-х років. Цей прийом особливо був популярний серед дослідників, які застосовували статистику, але не були фахівцями-статистиками, і призначався для використання з метриками або моделями, де математичні апроксимації не були легкодоступними.

Вибірковий розподіл середнього було добре опрацьовано, починаючи з 1908 року, чого не можна було сказати щодо вибіркового розподілу багатьох інших метрик. Бутстрап можна використовувати для визначення розміру вибірок,

експериментів з різними значеннями і, щоб зрозуміти, як вони впливають на вибірковий розподіл.

Коли бутстрап був представлений уперше, його зустріли зі значним скептицизмом. Для багатьох він був пов'язаний із трюком перетворення соломи на золото. Цей скептицизм випливав із нерозуміння мети бутстрапа.

Важливо. Бутстрап не компенсує малий розмір вибірки. Він не створює нові дані і при цьому не заповнює дірки в наявному наборі даних. Він просто повідомляє про те, як поводитимуться численні додаткові вибірки, коли їх будуть витягувати з популяції, такої як наша вихідна вибірка.

Алгоритм бутстрапівського повторного відбору показано нижче:

Алгоритм бутстрапівського повторного відбору

1. Витягти вибіркове значення, записати його і повернути назад.
2. Повторити перший крок n разів.
3. Записати середнє для повторно відібраних значень
4. Повторити R разів кроки 1-3.
5. Використовувати R результатів, щоб:
    - обчислити їх стандартне відхилення (воно оцінює стандартну помилку вибіркового середнього);
    - побудувати гістограму або коробчасту діаграму;
    - знайти довірчий інтервал.

Алгоритм бутстрапівського повторного відбору є методом оцінки розподілу статистики (наприклад, середнього) за допомогою повторного відбору з поверненням. Як видно з алгоритму, він полягає у генерації багатьох бутстрап-вибірок шляхом випадкового вибору спостережень із вихідної вибірки з поверненням, обчисленні статистики (наприклад, середнього) для кожної такої вибірки та аналізі розподілу цих статистик. Після достатньої кількості ітерацій (наприклад, 1000) отримані дані використовуються для обчислення стандартної помилки (як стандартного відхилення бутстрап-статистик), побудови гістограм або коробчастих діаграм для візуалізації розподілу, а також визначення довірчого інтервалу за допомогою перцентилів бутстрап-розподілу. Цей підхід надає гнучкий та обґрунтований спосіб статистичного висновування, не вимагаючи припущення про тип розподілу вихідних даних.

Ключові ідеї для бутстрапа

- Бутстрап (відбір зразків із набору даних із поверненням) є потужним інструментом визначення варіабельності вибіркової статистики.
- Бутстрап може застосовуватися однаковим чином у різних обставинах без обширного аналізу математичних апроксимацій вибіркових розподілів.
- Цей метод також дозволяє оцінювати вибіркові розподіли для статистик, де математична апроксимація не розроблена.

Повторний відбір проти бутстрапування. Іноді термін «повторний відбір» використовують як синонім для терміна «бутстрапування», який було щойно представлено в загальному вигляді. Найчастіше термін «повторний відбір» також охоплює процедури перестановки, де численні вибірки об'єднуються і відбір може здійснюватися без повернення. У будь-якому разі термін «бутстрап» завжди має на увазі вибірку зі спостережуваного набору даних із поверненням.

4.3.1 Довірчі інтервали

Частотні таблиці, гістограми, коробчасті діаграми і стандартні помилки – всі вони є способами зрозуміти потенційну помилку в оцінці вибірки. Довірчі інтервали – це ще один такий спосіб. Людині природним чином властиво уникати невизначеності; люди (особливо експерти) говорять «Я не знаю» вкрай рідко. Аналітики і менеджери, визнаючи наявність невизначеності, проте невиправдано довіряють оцінці, коли вона представлена єдиним числом (точковою оцінкою). Представляючи оцінку не єдиним числом, а діапазоном, ви протидієте цій тенденції. Довірчі інтервали роблять це способом, який бере свій початок у статистичних принципах відбору.

Довірчі інтервали завжди супроводжуються рівнем покриття, що виражається (високим) відсотком, скажімо, 90 або 95%. 90%-вий довірчий інтервал можна уявити так: це інтервал, який оточує центральні 90% бутстрапівського вибіркового розподілу вибіркової статистики.У більш загальному випадку, x%-ний довірчий інтервал навколо вибіркової оцінки повинен у середньому містити схожі вибіркові оцінки в x% випадків (коли виконано схожу процедуру відбору).

Ключові терміни для довірчих інтервалів

- Рівень довіри (confidence level)
    - Відсоток довірчих інтервалів, сконструйованих однаково з однієї і тієї ж популяції, які очікувано міститиму цільову статистику.
- Кінцеві точки інтервалу (interval endpoints)
    - Верх та низ довірчого інтервалу.

Загальний алгоритм для бутстрапівського довірчого інтервалу

1. Витягти з даних випадкову вибірку розміру n із поверненням (повторна вибірка).
2. Записати цільову статистику для повторної вибірки.
3. Повторити кроки 1–2 багато (R) разів.
4. Для x%-ного довірчого інтервалу відсікти $[(1 - [x/100])/2]$% від R результатів повторного відбору з обох кінців розподілу.
5. Як точки відсікання прийняти кінцеві точки x%-ного бутстрапівського довірчого інтервалу.

Бутстрап – інструмент загального характеру. Бутстрап є інструментом загального характеру, який використовується з метою генерування довірчих інтервалів для більшості статистик або модельних параметрів. Статистичні підручники та обчислювальні системи з корінням у більш ніж півстолітньому безкомп'ютерному статистичному аналізі також посилатимуться на довірчі інтервали, генеровані формулами, особливо на t-розподіл.

4.3.2 Приклад оцінки довірчого інтервалу

На рис. 4.5 показано 90%-вий довірчий інтервал для середньорічного доходу позикозаявників на основі вибірки з 20 значень, для якої середнє значення склало 57 573 $. Відсоток, пов'язаний із довірчим інтервалом, називається рівнем довіри. Що вищий рівень довіри, то ширший інтервал. Крім того, що менша вибірка, то ширший інтервал (тобто тим більша невизначеність). Обидві властивості досить логічні: що більше ви хочете бути впевненим і що менше даних у вас є, то ширшим слід зробити довірчий інтервал, щоб бути достатньо впевненим в отриманні істинного значення.

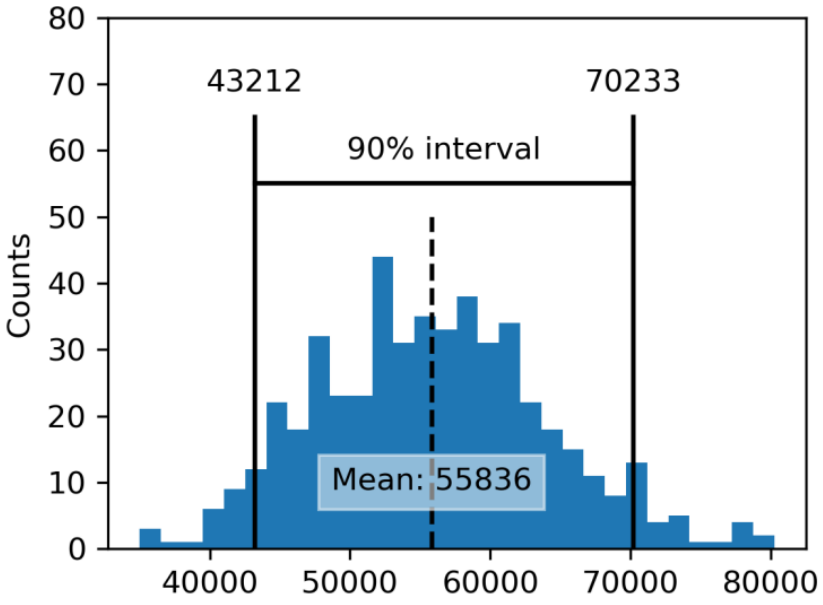


Рис 4.5 – Середнє значення із 90% довірчим інтервалом.

Довірчий інтервал – інструмент для отримання уявлення про те, наскільки варіабельним може бути результат вибірки.

Важливо. Для дослідника даних довірчий інтервал є інструментом для отримання уявлення про те, наскільки варіабельним може бути результат вибірки.

Дослідники даних використовують цю інформацію не для публікації академічної роботи або представлення результату контролюючому органу (що зазвичай і робить науковий дослідник), а, найімовірніше, щоб повідомити про потенційну помилку в оцінці і, можливо, дізнатися, чи необхідна більша вибірка. Зрозуміло, коли ми маємо результат у вигляді вибірки, нас найбільше цікавить, якою є ймовірність, що істинне значення лежить усередині деякого інтервалу. Насправді це не те питання, на яке відповідає довірчий інтервал, але в підсумку воно зводиться до того, як більшість людей інтерпретує відповідь. Запитання про ймовірність, пов'язану з довірчим інтервалом, починається з фрази «яка ймовірність, що з урахуванням процедури добору і популяції...». Відповідь на протилежне запитання – «якою є ймовірність, що (щось є істинним щодо популяції) з урахуванням результату вибірки», пов'язана зі складнішими розрахунками і глибшими факторами, які не піддаються точному визначенню.

4.3.3 Оцінка довірчих інтервалів в регресії

Алгоритм бутстрапівської оцінки довірчих інтервалів коефіцієнтів регресії

1. Розглядати кожен рядок (включаючи змінну результату) як окремий «пакет» і помістити всі пакети в коробку.
2. Вийняти пакет навмання, записати його значення та повернути його в коробку.

3. Повторити n разів крок 2; тепер у вас є одна бутстраповська повторно відібрана вибірка.
4. Виконати підгонку регресії до бутстрапівської вибірки і записати оціночні коефіцієнти.
5. Повторити кроки 2-4, скажімо, 1000 разів.
6. Тепер у вас є 1000 бутстраповських значень для кожного коефіцієнта; знайти відповідні відсотки для кожного з них (наприклад, 5-й і 95-й для 90% довірчого інтервалу).

Помилка точки даних. Великий інтерес для дослідників даних становлять інтервали навколо передбачених значень $y\left(\hat{Y}_i\right)$. Невизначеність навколо $\hat{Y}_i$ випливає з двох джерел:

- невизначеність у тому, якими є релевантні передбачувальні змінні та їхні коефіцієнти (див. наведений вище алгоритм бутстрапа);
- додаткова помилка, притаманна окремим точкам даних.

Ви можете думати про помилку окремої точки даних у такий спосіб: навіть якби ми знали точно, яким було рівняння регресії (наприклад, якби ми мали величезне число записів для підгонки), фактичні значення результату для заданої безлічі значень провісників варіюватимуться.

Наприклад, кілька будинків – кожен із 8 кімнатами, загальною площею 6500 кв. футів, 3 ванними кімнатами і цоколем – можуть мати різні вартості. Ми можемо змоделювати цю окрему помилку залишками від підігнаних значень.

Моделювання помилки регресійної моделі або помилки окремої точки даних

1. Взяти бутстрапівську вибірку з даних (у деталях роз'яснену раніше).
2. Виконати підгонку регресії і передбачити нове значення.
3. Взяти навмання один-єдиний залишок з початкового підгонки регресії, додати його до передбаченого значення і записати результат.
4. Повторити кроки 1-3, скажімо, 1000 разів.
5. Знайти 2,5-й та 97,5-й відсотки результатів.

4.4 Контрольні питання

1. Що таке статистична похибка і чим вона відрізняється від випадкової похибки в аналізі даних?

2. Чому випадкова вибірка вважається ключовим принципом статистичного висновку і що вона допомагає запобігти?
3. Що таке стратифікована вибірка і коли вона є кращою за просту випадкову вибірку?
4. Чому менший набір даних високої якості може бути ціннішим за більший набір даних нижчої якості в аналізі даних?
5. Що таке вибіркова похибка і як вона може призвести до оманливих висновків навіть при великих наборах даних?
6. Що таке регресія до середнього значення і хто першим виявив це статистичне явище?
7. Що стверджує центральна гранична теорема про вибірковий розподіл середнього значення для великих вибірок?
8. Як змінюється стандартна похибка зі збільшенням розміру вибірки і що таке «правило квадратного кореня» у цьому контексті?
9. Що таке метод бутстрап і чому він особливо корисний у науці про дані та статистиці, де математичні наближення є складними?
10. Чому є помилковим уявлення, що метод бутстрап може компенсувати невеликий розмір вибірки?
11. Яка основна мета довірчого інтервалу і чим він відрізняється від точкової оцінки?
12. Як рівень довіри (наприклад, 90% проти 95%) пов'язаний з шириною довірчого інтервалу?
13. Як регресія до середнього значення може призвести до помилкових висновків в аналізі даних, особливо в оцінці ефективності?

Лекція 5 – Передбачення за допомогою регресії. Факторні змінні. Перехресний контроль

5.1 Як працювати із категорійними даними (факторними змінними)?

Факторні змінні, іменовані також категоріальними змінними, приймають граничне число дискретних значень. Наприклад, метою позики може бути «консолідація заборгованості», «весілля», «автомобіль» тощо. Двійкова (так/ні) змінна, іменована також індикаторною змінною, є особливим випадком факторної змінної. Регресія вимагає на вході числові дані, тому факторні змінні потрібно перекодувати, щоб їх можна було використовувати в моделі. Підхід, який найчастіше зустрічається, полягає в конвертуванні змінної в множину двійкових фіктивних змінних

Фіктивні змінні – це двійкові змінні, що приймають значення 0 і 1 і виводяться шляхом перекодування факторних даних для використання в регресії та інших моделях.

Опорне кодування – тип кодування в якому один рівень фактора вибирається як опорний, а інші фактори зіставляються із цим рівнем.

Кодувальник з одним активним станом – тип кодування, загальноприйнятий у співтоваристві машинного навчання, у якому зберігаються всі рівні чинників. Широко використовується в деяких алгоритмах самонавчання; водночас цей прийом не підходить для множинної лінійної регресії.

Девіаційне кодування – тип кодування, при якому кожен рівень порівнюється не з опорним рівнем, а з сукупним середнім.

5.1.1 Приклад: дані житлового фонду округу Кінг

У даних житлового фонду округу Кінг є факторна змінна, що відповідає типу власності; нижче показано малу підмножину із шести записів. Є три можливих значення: Multiplex, Single Family і Townhouse. Для того щоб скористатися зазначеною факторною змінною, ми маємо конвертувати її в множину двійкових змінних. Це робиться шляхом створення двійкової змінної для кожного можливого значення факторної змінної.

Табл. 5.1. Дані про тип власності.

№	PropertyType
1	Multiplex
2	Single Family
3	Single Family
4	Single Family
5	Single Family
6	Townhouse

Кодувальник з одним активним станом. У Python ми можемо конвертувати категоріальні змінні у фіктивні за допомогою методу get_dummies пакета pandas. За замовчуванням метод повертає кодування категоріальної змінної з одним активним станом:

Код

```
import pandas as pd

pd.get_dummies(house['PropertyType']).head()
```

Табл. 5.2. Перетворені дані про тип власності за допомогою кодувальника з єдиним активним станом.

№	PropertyTypeMultiplex	PropertyTypeSingleFamily	PropertyTypeTownhouse
1	1	0	0
2	0	1	0
3	0	1	0
4	0	1	0
5	0	1	0
6	0	0	1

Опорне кодування. Іменований аргумент drop_first повертатиме $P-1$ стовпців. Використовуйте його, щоб уникнути проблеми мультиколінеарності:

Код

```
import pandas as pd

pd.get_dummies(house['PropertyType'], drop_first=True).head()
```

У деяких алгоритмах, що автоматично навчаються, як-от найближчі сусіди і моделі на основі дерев рішень, кодування з одним активним станом є стандартним способом представлення факторних змінних. У регресійному формулюванні факторна змінна з P чітко помітними рівнями зазвичай подається матрицею тільки з $P-1$ стовпчиками. Це зумовлено тим, що регресійна модель у типовій ситуації включає член перетину. Говорячи про перетин, після того як ви визначили значення для $P-1$ двійкових стовпчиків, значення P-го стає відомим і може вважатися надлишковим. Додавання P-го стовпця викличе помилку мультиколінеарності.

Табл. 5.3. Перетворені дані про тип власності за допомогою опорного кодування.

№	PropertyTypeSingleFamily	PropertyTypeTownhouse
1	0	0
2	1	0
3	1	0
4	1	0
5	1	0
6	0	1

5.1.2 Лінійна регресія для факторних змінних

Код

```
import pandas as pd
from sklearn.linear_model import LinearRegression

house = pd.read_csv(house_sales.csv')
predictors = ['SqFtTotLiving', 'SqFtLot', 'Bathrooms', 'Bedrooms', 'BldgGrade',
'PropertyType']
outcome = 'AdjSalePrice'
X = pd.get_dummies(house[predictors], drop_first=True)
lm = LinearRegression()
lm.fit(X, house[outcome])
print('Перетин')
print(lm.intercept_)
print('Коефіцієнти:')
for name, coef in zip(X.columns, lm.coef_):
    print(name, coef)
```

Вивід

```
Перетин
-446841.3663116747
Коефіцієнти:
SqFtTotLiving 223.3736289250377
SqFtLot -0.07036798136813434
Bathrooms -15979.013473415189
Bedrooms -50889.732184830194
BldgGrade 109416.30516146196
PropertyType_Single Family -84678.21629549236
PropertyType_Townhouse -115121.9792160916
```

Результат роботи програми. Результат регресії показує два коефіцієнти, що відповідають типу власності Property Type: PropertyTypeSingle Family і PropertyTypeTownhouse. Коефіцієнт Multiplex відсутній, оскільки він неявно визначається, коли PropertyTypeSingle Family == 0 і PropertyTypeTownhouse == 0. Ці коефіцієнти інтерпретуються як відносні для Multiplex, і тому вартість будинку з типом власності single Family є меншою майже на 85 000 $, і вартість будинку з типом власності Townhouse є меншою більш ніж на 150 000 $.

5.1.3 Інші способи кодування факторів

Існує кілька інших способів кодування факторних змінних, іменованих системами контрастного кодування. Наприклад, девіаційне кодування, іменоване також сумовими контрастами, порівнює кожен рівень із сукупним середнім. Ще одним контрастом є поліноміальне кодування, яке підходить для впорядкованих факторів. За винятком упорядкованих чинників, дослідники

даних зазвичай не стикаються з іншими типами кодування крім опорного кодування або кодувальника з одним активним станом.

5.1.4 Упорядковані факторні змінні

Деякі факторні змінні відображають рівні фактора. Вони називаються впорядкованими факторними змінними або впорядкованими категоріальними змінними. Наприклад, категорія якості позики може бути A, B, C тощо. – кожна категорія несе в собі більший ризик, ніж попередня категорія. Нерідко впорядковані факторні змінні можуть бути конвертовані в числові значення і використовуватися як є. Наприклад, змінна «клас будинка» – це впорядкована факторна змінна. Кілька типів категорій якості наведено в табл. 5.4.

Табл. 5.4. Приклад упорядкованих факторних змінних.

Значення	**Опис**
1	Низькобюджетне
2	Нижче середнього
5	Задовільне
10	Дуже гарне
12	Розкішне
13	Особняк

Хоча категорії якості мають конкретний сенс, числове значення впорядковано від низу до верху, відповідаючи будинкам вищої якості. Розгляд упорядкованих чинників як числової змінної зберігає інформацію, що міститься в упорядкуванні, яка буде втрачена, якщо його конвертувати у фактор.

5.2 Інтерпретування рівняння регресії

У науці про дані найважливіше застосування регресії полягає в передбаченні залежної змінної (результату). У деяких випадках, однак, отримання глибшого уявлення безпосередньо із самого рівняння, щоб зрозуміти природу зв'язку між провісниками та результатом, є цінністю. У цьому розділі наведено рекомендації щодо обстеження рівняння регресії та його інтерпретації.

Ключові терміни для інтерпретування рівняння регресії

- Корельовані змінні
    - Коли передбачувальні змінні високо корельовані, складно інтерпретувати окремі коефіцієнти.
- Мультиколінеарність

    - Коли передбачувальні змінні мають ідеальну або майже ідеальну кореляцію, регресія може бути нестабільною або її неможливо обчислити.
    - Синоніми: колінеарність, солінійність
- Спотворюючі змінні
    - Важливий провісник, який, якщо його опустити, призводить до уявних зв'язків у рівнянні регресії.
- Головні ефекти
    - Зв'язок між передбачуваною змінною та змінною результату, яка не залежить від інших змінних.
- Взаємодії
    - Взаємозалежний зв'язок між двома або кількома провісниками та результатом (відгуком).

5.2.1 Корельовані провісники

У множинній регресії передбачувальні змінні часто корелюють одна з одною. Як приклад давайте проекзаменуємо коефіцієнти регресії для моделі, підігнаної раніше.

Вивід

```
Перетин
-446841.3663116747
Коефіцієнти:
SqFtTotLiving 223.3736289250377
SqFtLot -0.07036798136813434
Bathrooms -15979.013473415189
Bedrooms -50889.732184830194
BldgGrade 109416.30516146196
PropertyType_Single Family -84678.21629549236
PropertyType_Townhouse -115121.9792160916
```

Коефіцієнт для спалень Bedrooms є від'ємним! З цього випливає, що додавання спальні в будинок зменшить його вартість. Як це може бути? Це викликано тим, що передбачувані змінні корельовані: у більших будинках спостерігається тенденція до більшої кількості спалень, і саме розмір будинку керує його вартістю, а не кількість спалень. Розглянемо два будинки одного й того самого розміру: розумно очікувати, що будинок із більшим числом спалень, але менших за площею вважатиметься менш бажаним.

Наявність корельованих провісників ускладнює інтерпретацію знака і значення регресійних коефіцієнтів (і може роздути стандартну похибку

оціночних значень). Змінні для спалень, розміру будинку і кількості ванних кімнат – усі вони є корельованими. Це ілюструється наведеним прикладом, у якому виконується підгонка ще однієї регресії після видалення змінних SqFtTotLiving (житлова площа), SqFtFinBasement (цокольна площа) і Bathrooms (ванні кімнати) з рівняння. Тепер коефіцієнт для спалень є позитивним – відповідно до того, що ми очікували б (хоча тепер, коли ці змінні були виключені, він насправді діє як замінник для розміру будинку). Корельовані змінні є лише однією складністю, пов'язаною з інтерпретуванням регресійних коефіцієнтів. У моделі немає змінної, яка відповідає за місце розташування будинку, і модель змішує дуже різні типи районів. Місцезнаходження може бути спотворювальною змінною.

5.2.2 Мультиколінеарність

Граничний випадок корельованих змінних продукує мультиколінеарність – умову, за якої існує надлишок серед передбачувальних змінних. Ідеальна мультиколінеарність трапляється, коли одна передбачувальна змінна може бути виражена як лінійна комбінація інших Мультиколінеарність відбувається, коли:

- змінна включається до складу моделі багаторазово помилково;
- з факторної змінної створюються P фіктивних змінних замість $P - 1$;
- дві змінні майже ідеально корельовані одна з одною.

Питання мультиколінеарності в регресії має бути вирішене – змінні необхідно виключати доти, доки мультиколінеарність не зникне. Регресія не має добре визначеного розв'язку за присутності ідеальної мультиколінеарності. Багато пакетів, зокрема на мовах R і Python, автоматично обробляють деякі типи мультиколінеарності. Наприклад, якщо двічі включити змінну SqFtTotLiving у регресію даних, то результати будуть такими самими, що й для початкової моделі.

У разі неідеальної мультиколінеарності обчислювальна система може отримати рішення, але результати можуть бути нестабільними.

Примітка. Мультиколінеарність не становить якоїсь особливої проблеми для нерегресійних методів, таких як дерева, кластеризація і найближчі сусіди, і в таких методах рекомендується залишати P фіктивних змінних (замість $P - 1$). Проте навіть у цих методах ненадмірність у передбачувальних змінних, як і раніше, є перевагою.

5.2.3 Спотворюючі змінні

З корельованими змінними проблема полягає у включенні змінних до складу: включення різних змінних, що мають схожий передбачувальний зв'язок із відгуком. Зі змінними, що спотворюють, проблемою є виключення змінних зі складу: важлива змінна не включена в рівняння регресії. Наївна інтерпретація коефіцієнтів рівняння може призвести до неспроможних висновків.

Приклад: спотворюючі змінні.

Візьмемо, наприклад, рівняння регресії для округу Кінг. Регресійні коефіцієнти SqFtLot (площа земельної ділянки), Bathrooms (ванні кімнати) і Bedrooms (спальні) – усі є негативними. Початкова регресійна модель не містить змінну, яка представляла б місце розташування – дуже важливий провісник ціни на нерухомість.

Вивід

```
Перетин
-446841.3663116747
Коефіцієнти:
SqFtTotLiving 223.3736289250377
SqFtLot -0.07036798136813434
Bathrooms -15979.013473415189
Bedrooms -50889.732184830194
BldgGrade 109416.30516146196
PropertyType_Single Family -84678.21629549236
PropertyType_Townhouse -115121.9792160916
```

Для того щоб змоделювати місцерозташування, включимо змінну ZipGroup (група поштових індексів), яка віднесе поштовий індекс до однієї з п'яти груп від найменш дорогого (1) до найдорожчого (5). Змінна ZipGroup, безсумнівно, є дуже важливою: будинок у найдорожчій групі поштових індексів оцінюється як такий, що має вищу продажну ціну майже на 340 000 $. Коефіцієнти SqFtLot і Bathrooms тепер є позитивними, і додавання ванної збільшує продажну ціну на 7500 $ Коефіцієнт для Bedrooms, як і раніше, є негативним. Хоча цей феномен суперечить логіці, він добре відомий у торгівлі нерухомістю. Наявність у будинків однакової загальної житлової площі та більшої кількості і, отже, менших за розміром спалень асоціюється з менш цінними будинками.

Вивід

```
Перетин: -6.709e+05
Коефіцієнти:
SqFtLot 4.692e-01
Bedrooms -4.139e+04
PropertyTypeSingle Family 2.113e+04
SqFtTotLiving 2.112e+02
Bathrooms 5.537e+03
BldgGrade 9.893e+04
PropertyTypeTownhouse -7.741e+04
ZipGroup2 5.169e+04
ZipGroup3 1.142e+05
ZipGroup4 1.783e+05
ZipGroup5 3.391e+05
```

5.3 Перехресний контроль

Усі класичні статистичні регресійні метрики є «внутрішньовибірковими» метриками – вони застосовуються до тих самих даних, які використовувалися для підгонки моделі. Інтуїтивно ви розумієте, що буде цілком логічним відкладати трохи вихідних даних, не використовуючи їх для підгонки моделі, а потім застосовувати модель до відкладених даних, щоб побачити, як добре вона справляється зі своєю роботою. Зазвичай ви будете використовувати більшу частину даних для підгонки моделі, а решту – для її тестування.

Ідея перевірки поза вибіркою не є новою, але вона не утвердилася до тих пір, поки великі набори даних не стали більш переважаючими; маючи в розпорядженні малий набір даних, аналітики, як правило, хочуть використовувати всі наявні дані і на їх основі виконувати підгонку кращої моделі.

Використання відкладеної вибірки ставить вас у залежність від деякої невизначеності, що виникає просто через варіабельність у малій відкладеній вибірці. Наскільки відрізнятимуться результати аналізу моделі, якби ви отримували іншу відкладену вибірку?

Перехресний контроль розширює ідею відкладеної вибірки до послідовних відкладених вибірок.

Псевдокод базового k-блочного перехресного контролю

1. Відкласти $1/k$ даних як відкладену вибірку
2. Натренувати модель на даних, що залишилися.
3. Застосувати модель до відкладеної вибірки $1/k$ (виставити їй бал) та записати необхідні метрики оцінювання результативності моделі.
4. Відновити перші $1/k$ даних і відкласти наступне $1/k$ (за винятком будь-яких записів, які були обрані вперше).
5. Повторити кроки 2-4.
6. Повторювати доти, доки кожен запис не буде використаний у відкладеній частці.
7. Усереднити чи іншим чином скомбінувати метрики аналізу моделі.

Розподіл даних на тренувальну та відкладену вибірки також називається розподілом на блоки.

5.4. Контрольні питання

1. Що таке факторні змінні (категоріальні змінні)?
2. Як у Python перекодувати факторну змінну в фіктивні змінні?

3. Чому використовується параметр drop_first=True у get_dummies() для регресії?
4. Який рівень вважається опорним у кодуванні з видаленням першого стовпця?
5. Чому коефіцієнт Bedrooms спочатку був від’ємним, а після видалення SqFtTotLiving став позитивним?
6. Що таке корельовані провісники?
7. Що таке спотворююча змінна?
8. Навіщо використовується перехресний контроль?
9. Чому не можна використовувати одні й ті самі дані для навчання та тестування?
10. Яка роль перетину у регресії з фіктивними змінними?
11. Як кодувати впорядковані факторні змінні (наприклад, якість будинку)?
12. Чому не слід кодувати впорядковані факторні змінні як категоріальні?
13. Як інтерпретувати від’ємний коефіцієнт для фіктивної змінної?
14. Що таке головний ефект у регресії?
15. Що таке взаємодія у регресії?

Лекція 6 – A/B тестування

Планування експериментів є наріжним каменем практичної статистики з додатками фактично в усіх галузях дослідження. Мета полягає в тому, щоб спланувати експеримент, який підтвердить або відхилить гіпотезу. Дослідники даних стикаються з потребою проводити безперервні експерименти, особливо щодо користувацького інтерфейсу та товарного маркетингу. У цій лекції подано огляд традиційного планування експериментів та обговорено кілька поширених завдань у науці про дані. У ній також буде розглянуто кілька часто цитованих у статистичному виведенні понять і дано пояснення їхнього сенсу й актуальності (або відсутності такої) для науки про дані.

Щоразу згадка статистичної значущості, p-значень або перевірки на основі t-статистики відбувається, як правило, в контексті класичного «конвеєра» статистичного висновку. Цей процес починається з гіпотези («препарат А кращий за наявний стандартний препарат», «ціна А прибутковіша за наявну ціну В»).

Експеримент (це може бути A/B-тест) призначений для перевірки гіпотези, побудованої таким чином, щоб забезпечувати незаперечні результати. Дані збирають і аналізують, і далі роблять висновок. Термін «висновок» відображає намір застосувати експериментальні результати, які передбачають лімітований набір даних, до більшого процесу або популяції. Типовий конвеєр експерименту показано на рис. 6.1.

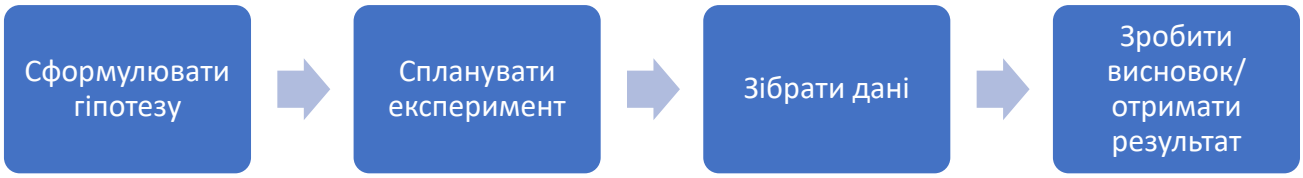


Рис. 6.1 – Класичний конвеєр експерименту.

6.1 Що таке A/B тестування?

A/B-тест – це експеримент із двома групами для визначення найкращого з двох варіантів, двох продуктів, двох процедур, двох лікарських засобів тощо. Нерідко один варіант із двох є стандартним існуючим варіантом або відсутній взагалі. Якщо використовується стандартний варіант (або ж він відсутній), то він називається контрольним. Типова гіпотеза полягає в тому, що пропонований варіант кращий за контрольний.

Приклади A/B тестування:

- тестування двох методів обробітку ґрунту, щоб визначити, яка з них призводить до найкращого проростання саджанців;
- тестування двох методів лікування, щоб визначити, який з них ефективніше пригнічує рак;

- тестування двох цін, щоб визначити, яка з них приносить більше чистого прибутку;
- тестування двох заголовків веб-сторінки, щоб визначити, який із них породжує більше натискань;
- тестування двох веб-оголошень, щоб визначити, яке з них генерує більше конверсій.

Під час A/B тесту необхідно чітко визначати, чи є отриманий ефект наслідком різних варіантів або випадковістю? Належний A/B-тест має випробовуваних, які можуть бути віднесені до того чи іншого варіанта експерименту. Піддослідним може бути людина, саджанець, відвідувач веб-сайту; головне, що піддослідному пропонується варіант експерименту. В ідеальному випадку випробовуваних рандомізують (призначають у випадковому порядку) за двома запропонованими варіантами. Завдяки цьому ви знаєте, що будь-яка різниця між тестовими групами відбувається внаслідок одного з двох:

- ефекту різних варіантів експерименту;
- чистої випадковості (тобто випадок, можливо, призвел до того, що результативніші випробовувані були природним чином сконцентровані в A або B).

6.2 Приклад експерименту

Необхідно звернути увагу на перевірочну статистику, або метрику, яку ви використовуєте для порівняння групи A з групою B. Можливо, найпоширенішою метрикою в науці про дані є двійкова змінна: наявність або відсутність натискання, наявність або відсутність купівлі, наявність або відсутність шахрайства тощо. Ці результати узагальнюються в таблиці розміру 2x2. У табл. 6.1 наведено таблицю 2x2 з результатами тестування фактичної ціни. Якщо метрика представлена безперервною змінною (сумою покупки, прибутком тощо), або кількістю (наприклад, днями в стаціонарі, відвіданими сторінками), то результат може бути показаний по-різному.

Табл. 6.1. Таблиця з результатами тестування фактичної ціни.

Результат	**Ціна А**	**Ціна Б**
Куплено	200	182
Не куплено	23 539	22 406

Якщо цікавить не конверсія, а виручка з розрахунку на один перегляд сторінки, то результати цінового тесту в таблиці можуть мати такий вигляд:

- Прибуток/сторінка з ціною A: середнє = 3,87, стандартне відхилення = 51,10
- Прибуток/сторінка з ціною B: середнє = 4,11, стандартне відхилення = 62,98

Статистичні обчислювальні системи, включно з R і Python, генерують в такому форматі результат за замовчуванням, але це не означає, що вся ця інформація є корисною або релевантною. Ви можете бачити, що наведені стандартні відхилення не надто корисні. Судячи з усього, вони говорять про те, що численні значення могли б бути від'ємними, коли від'ємний виторг не є можливим. Ці дані складаються з малого набору відносно високих значень (перегляди з конверсіями) і величезної кількості нульових значень (перегляди без конверсій). Дуже важко узагальнити варіабельність таких даних в одному-єдиному числі, хоча середнє абсолютне відхилення від середнього (7,68 для A і 8,15 для B) видається більш розумним, ніж стандартне відхилення.

6.3 Навіщо потрібна контрольна група?

Чому в A/B тесті не можна проігнорувати контрольну групу і просто виконати експеримент, застосувавши варіант експерименту, який цікавить, тільки до однієї групи і порівнявши результат із попереднім досвідом? Без контрольної групи немає жодної гарантії, що «інші умови будуть рівними», а будь-яка різниця дійсно зумовлена варіантом експерименту (або випадковим чином). Коли є контрольна група, вона підпорядковується тим самим умовам (за винятком варіанта, що цікавить), що й тестова група. Якщо просто порівнювати з «базовою лінією» або попереднім досвідом, то, крім варіанта, можуть різнитися й інші чинники.

Використання A/B-тестування в науці про дані, як правило, знаходиться у веб-контексті. Як варіанти експерименту можуть виступати дизайн веб-сторінки, ціна товару, формулювання заголовка оголошення або будь-який інший елемент. При цьому необхідно серйозно задуматися про те, як забезпечити збереження принципів рандомізації.

У типовій ситуації випробовуваним в експерименті є відвідувач вебсайту, а вимірюваними нами наслідками, у яких ми зацікавлені, – натискання, купівлі, тривалість відвідування, кількість відвідуваних вебсторінок, чи переглянута певна сторінка тощо. У стандартному A/B-експерименті потрібно вибрати одну метрику заздалегідь. Можуть бути зібрані і представляти інтерес численні поведінкові метрики, але якщо експеримент очікувано веде до вибору між варіантом A і варіантом B, то одну метрику, або перевірну статистику, має бути встановлено заздалегідь. Вибір перевірочної статистики після того, як експеримент проведено, відкриває двері для зміщення внаслідок упередженості дослідника.

Засліплення у статистичному дослідженні. Сліпе дослідження – це дослідження, у якому випробовувані не обізнані про те, що їм пропонують варіант А або варіант В. Поінформованість про той чи інший варіант може вплинути на відгук. У подвійному сліпому дослідженні дослідники і помічники (наприклад, лікарі та медсестри в медичному дослідженні) не обізнані про те, які випробовувані беруть участь і який варіант їм пропонують. Сліпе дослідження неможливе, коли природа варіанта прозора, наприклад когнітивна психотерапія за допомогою комп'ютера на відміну від психолога.

6.4 Чому тільки A/B? Чому не C, D...?

A/B-тести популярні у світі маркетингу та електронної комерції, але це далеко не єдиний тип статистичного експерименту. Додаткові варіанти цілком можливі. Випробовувані можуть бути піддані повторним вимірювальним дослідженням. Фармацевтичні випробування, де суб'єкти дефіцитні, дорогі й беруть участь протягом довгого часу, іноді плануються з численними можливостями зупинити експеримент і досягти остаточного висновку. У традиційному плануванні статистичного експерименту центральна увага приділяється відповіді на статистичне запитання про ефективність зазначених варіантів. Дослідники даних менше цікавляться питанням: «Чи є різниця між ціною А і ціною В статистично значущою?», ніж питанням: «Яка з численних можливих цін є найкращою?». Для цього використовується відносно новий тип планування експерименту: багаторукий бандит.

Отримання дозволу. У науковому та медичному дослідженнях, у яких беруть участь люди, зазвичай потрібно отримувати їхню згоду на проведення експерименту, а також схвалення інституційної ревізійної ради з питань етики. Експерименти в бізнесі, які виконуються в рамках безперервних операцій, майже ніколи не піддаються збору попередніх згод. У більшості випадків (наприклад, цінові експерименти або експерименти, пов'язані з тим, який заголовок показати або яку пропозицію слід зробити) така практика є загальноприйнятою.

Компанія Facebook, однак, зіткнулася з цими загальноприйнятими правилами у 2014 році, коли проводила експерименти з емоційним тоном у користувацьких стрічках новин. Компанія використовувала сентиментний аналіз для класифікації постів стрічки новин на позитивні або негативні, потім поміняла позитивно-негативний баланс у матеріалі, який вона показувала користувачам. Кілька випадково відібраних користувачів відчували на собі більш позитивні пости, тоді як інші – більш негативні. Було виявлено, що користувачі, які читали більш позитивну стрічку новин, з більшою ймовірністю самі відправляли позитивні пости, і навпаки.

Однак величина ефекту була малою, при цьому компанія Facebook зіткнулася з великою критикою в тому плані, що експеримент проводився без відома користувачів. Деякі користувачі вважали, що компанія Facebook цілком

могла підштовхнути надзвичайно пригнічених користувачів до краю, коли ті отримували негативну версію свого каналу.

6.5 Для чого необхідна перевірка значущості?

Перевірки гіпотез, так звані перевірки значущості, набули значного поширення в традиційному статистичному аналізі, що зустрічається в опублікованих дослідженнях. Такі перевірки призначені для того, щоб допомогти дізнатися, чи може випадковість бути відповідальною за спостережуваний ефект. У типовій ситуації A/B-тест конструюється з урахуванням гіпотези. Наприклад, гіпотеза може полягати в тому, що ціна B приносить вищий прибуток.

Навіщо нам потрібна гіпотеза? Чому не можна просто поглянути на результат експерименту і зупинитися на будь-якому варіанті, який працює краще? Відповідь криється у схильності людського розуму недооцінювати розмах природної випадкової поведінки. Один із проявів цієї схильності полягає в невмінні передбачати граничні події, або так званих «чорних лебедів» (концепція, згідно з якою важкопрогнозовані та рідкісні події, які мають значні наслідки, мають особливі характеристики). Ще одним її проявом є тенденція неправильно тлумачити випадкові події як такі, що мають ознаки певної значущості. Статистична перевірка гіпотез була винайдена як спосіб захистити дослідників від того, щоб бути обдуреним випадковістю.

Ключові терміни для перевірки гіпотез

- Нульова гіпотеза
    - Гіпотеза у тому, що виною всьому є випадковість.
- Альтернативна гіпотеза
    - Гіпотеза, що компенсує нульову (те, що ви сподіваєтесь довести).
- Одностороння перевірка
    - Перевірка гіпотези, коли кількість випадкових результатів підраховується лише в одному напрямі.
- Двостороння перевірка
    - Перевірка гіпотези, коли кількість випадкових результатів підраховується у двох напрямах.

6.6 Значущість A/B експерименту

У належно спланованому A/B-тесті ви збираєте дані про варіанти A і B таким чином, що будь-яка різниця між A і B повинна відбутися внаслідок одного з двох:

- випадковості у віднесенні досліджуваних у групи;
- справжньої різниці між A та B.

Статистична перевірка гіпотез є подальшим аналізом A/B-тесту або будь-якого рандомізованого експерименту з метою визначити, чи є випадковість розумним поясненням спостережуваної різниці між групами A і B.

6.6.1 Нульова гіпотеза

Використовується наступна логіка: «З урахуванням схильності людини реагувати на незвичайну, але випадкову поведінку та тлумачити її як щось змістовне і реальне, в наших експериментах нам знадобиться доказ того, що різниця між групами є більш граничною, ніж та, яка обґрунтовано могла б бути породжена випадковістю».

Ця логіка пов'язана з базовим припущенням про те, що варіанти експерименту еквівалентні і будь-яка різниця між групами зумовлена випадковістю. **Це базисне припущення називається нульовою гіпотезою.**

І наша надія тоді полягає у тому, що ми зможемо на ділі довести неправильність нульової гіпотези та показати, що результати для груп A і B різняться більше, ніж те, що може породити випадковість. Один із шляхів зробити це лежить через процедуру повторного відбору з перестановкою, в якій ми перетасовуємо результати груп A і B і надалі неодноразово роздаємо дані у групи аналогічних розмірів, а потім спостерігаємо за тим, як часто ми отримуємо таку ж граничну різницю, що й різниця, що спостерігається. Об'єднані перетасовані результати з груп A та B та процедура повторного їх відбору втілюють нульову гіпотезу у тому, що групи A і B є еквівалентними і взаємозамінними, і називається *нульової моделлю*.

6.6.2 Альтернативна гіпотеза

Перевірки гіпотез передбачає як нульову гіпотезу, а й компенсуючу її альтернативну гіпотезу. Ось кілька прикладів:

- нульова гіпотеза – «різниці між середніми в групі A і групі B немає», альтернативна гіпотеза – «A відрізняється від B» (може бути більше або менше);
- нульова гіпотеза – «$A \leq B$», альтернативна – «$A > B$»;

- нульова гіпотеза – «B не більша за A на X%», альтернативна – «B більша за A на X%».

Взяті разом нульова та альтернативна гіпотези охоплюють абсолютно всі наявні можливості. Природа нульової гіпотези визначає структуру перевірки гіпотези.

6.6.3 Одностороння перевірка гіпотези проти двосторонньої

Нерідко в A/B-тесті ви перевіряєте нову можливість (скажімо, B) щодо взятої за замовчуванням можливості (A) і від самого початку виходите з того, що дотримуватиметеся початкової можливості, якщо тільки нова можливість не виявиться виразно кращою. У такому разі при перевірці гіпотез вам буде потрібно захиститися від того, щоб не бути обдуреним випадковістю у напрямі на користь B. Вас не хвилює, що ви можете бути обдуреними випадковістю в іншому напрямі, бо залишатиметеся на боці A, якщо тільки B не виявиться виразно кращою. Тому ви хочете мати спрямовану альтернативну гіпотезу (B краща за A). У такому випадку ви використовуєте перевірку односторонньої гіпотези (або гіпотезу з одним хвостом). Це означає, що граничний шанс призводить тільки до односпрямованого підрахунку в бік p-значення.

Якщо ви хочете, щоб перевірка гіпотези захистила вас від того, щоб бути обдуреним випадковістю в будь-якому напрямі, то альтернативна гіпотеза має бути двосторонньою (A відрізняється від B і може бути більшою або меншою). У такому разі ви використовуєте двосторонню гіпотезу (або гіпотезу з двома хвостами). Це означає, що граничний шанс призводить до двоспрямованого підрахунку в бік p-значення.

Перевірка гіпотези з одним хвостом часто відповідає природі ухвалення рішення в A/B-тестуванні, в якому ухвалення рішення є обов'язковим, і одній можливості зазвичай присвоюють статус «за замовчуванням», якщо тільки інша не виявляється кращою. Однак обчислювальні системи, включно з R, як правило, за замовчуванням надають на виході двосторонню перевірку, і багато спеціалістів-статистиків віддають перевагу більш консервативній двосторонній перевірці, тільки щоб запобігти суперечкам.

Тема відмінностей між перевірками з одним хвостом або з двома хвостами є доволі заплутаною і не має прямого стосунку до науки про дані, де прецизійність (висока точність) розрахунків p-значення не особливо важлива.

- Альтернативна гіпотеза «B краще за A» – одностороння альтернативна гіпотеза (гіпотеза з одним хвостом). Якщо це не так вас не хвилює, чи ви можете бути введені в оману випадковістю в інший бік.
- Альтернативна гіпотеза «A відрізняється від B, та може бути або гіршою або кращою» – двостороння гіпотеза (гіпотези з двома хвостами).

6.6.4 Перестановний тест

У процедурі перестановки задіюються дві або більше вибірки, як правило, групи в A/B-тесті або іншій перевірці гіпотез. Перестановка означає зміну порядку проходження значень, або їхню пермутацію. Перший крок у перестановочній перевірці гіпотези полягає в об'єднанні результатів із груп A і B (і, груп C, D. ..., якщо вони використовуються). У цьому полягає логічне втілення нульової гіпотези – варіанти експерименту, які були запропоновані групам, не відрізняються. Потім ми перевіряємо цю гіпотезу шляхом випадкового вилучення груп із цієї об'єднаної множини і дивимося, наскільки вони відрізняються одна від одної.

Алгоритм перестановочного тесту

1. Об'єднати результати з різних груп у єдиний набір даних.
2. Перетасувати об'єднані дані, потім у випадковому порядку витягти (без повернення) повторну вибірку того самого розміру, що й група A (очевидно, що вона міститиме дані з різних груп).
3. З даних, що залишилися, у випадковому порядку витягти (без повернення) повторну вибірку того ж розміру, що і група B.
4. Зробити те ж саме для груп C, D і т. д. Тепер ви зібрали один набір повторних вибірок, які відображають розміри вихідних вибірок.
5. Залежно від статистики або оцінки, яка була обчислена для вихідних вибірок (наприклад, різниця у групових частках), тепер розрахувати її для повторних вибірок та записати; це буде однією ітерацією перестановки.
6. Повторити попередні кроки R разів для отримання перестановного розподілу перевірочної статистики.

Тепер повернемося до різниці між групами і порівняємо її з набором перестановлених різниць:

- Якщо різниця, що спостерігається, переконливо лежить в межах набору перестановлених різниць, то ми нічого не довели – різниця знаходиться всередині діапазону того, що може спричинити випадковість.
- Однак, якщо спостерігається різниця лежить поза більшою частиною перестановочного розподілу, ми приходимо до висновку, що випадковість не несе відповідальності. Говорячи технічною мовою, різниця є *статистично значущою.*

6.7 Контрольні питання

1. Яка основна мета дизайну експериментів в статистиці?
2. Що таке A/B-тестування?
3. Чому важлива контрольна група в A/B-тестуванні?
4. Що таке нульова гіпотеза в A/B-тестуванні?
5. Що таке альтернативна гіпотеза?
6. Коли використовують односторонню перевірку гіпотези, а коли двосторонню?
7. Що таке перестановочний тест?
8. Чому важлива рандомізація в A/B-тестуванні?
9. Чому не можна вибирати метрику після проведення експерименту?
10. Що є метою перевірки гіпотез у A/B-тестуванні?
11. В чому проблема використання історичних даних замість контрольної групи?
12. Як перестановочний тест допомагає визначити статистичну значущість?
13. Яка мета засліплення у експериментах?

Лекція 7 – Дисперсійний аналіз та багаторукий бандит

7.1 Приклад перестановочного тесту для електронної комерції

Розглянувши на попередній лекції A/B тестування та перестановочний тест, давайте розберемо на прикладі, як розібратись чи є отриманий результат статистично значущим, коли експеримент проводиться для двох випадків. А далі розглянемо, як правильно провести експеримент для багатьох варіантів. Початкові дані представлено на в табл. 7.1.

Табл. 7.1. Зібрані дані про покупки.

Результат	**Ціна A**	**Ціна B**
Куплено	200	182
Не куплено	23 539	22 406

Розрахуємо конверсію для варіанта A:

$$\frac{200}{23539 + 200} \cdot 100\% = 0{,}8425\% \tag{7.1}$$

та конверсія варіанта B:

$$\frac{182}{22406 + 182} \cdot 100\% = 0{,}8057\% \tag{7.2}$$

Отже маємо, що:

1. Ціна A забезпечує майже на 5% кращу конверсію!
2. Але різниця конверсій складає лише 0,0368%

Чи може краща конверсія бути зумовленою випадковістю? Для відповіді на це запитання застосуємо перестановочний тест:

1. Помістити картки, позначені 1 і 0, у коробку: вона представлятиме передбачувану спільну інтенсивність конверсії 382 одиниць та 45 945 нулів: 0,008246 = 0,8246%.
2. Перетасувати та витягти повторну вибірку розміру 23 739 (число n таке саме, що й у ціни A), записати число одиниць.
3. Записати число одиниць в 22 588, що залишилися (число n таке ж, що і у ціни B).
4. Записати різницю у частині одиниць.
5. Повторити кроки 2-4.

6. Відповісти на запитання: як часто спостерігалася різниця ≥ 0,0368?

Ми можемо оцінити p-значення з нашого перестановочного тесту шляхом взяття частки числа разів, коли перестановочний тест породжує різницю, рівну чи більшу, ніж різниця, що спостерігається. p-значення становить 0,308, а значить, ми очікувано будемо досягати такого ж граничного результату, як і цей, або більш граничного через випадковість, що перевищує 30% часу. **Результат не є статистично значущим!**

Ключові терміни для статистичної значущості та p-значення

- p-значення
    - З урахуванням випадкової моделі, яка втілює нульову гіпотезу, p-значення є ймовірністю випадкового отримання результатів настільки ж незвичайних або граничних, як і результати спостереження.
- Альфа
    - Імовірнісний поріг «незвичайності», який випадкові результати повинні перевершити, щоб фактичні наслідки вважалися статистично значимими. Традиційно або 1%, або 5%.
    - *Синонім:* рівень значущості.
- Помилка 1-го роду
    - Помилковий висновок у тому, що ефект є реальним (тоді, як і зумовлений випадковістю).
- Помилка 2-го роду
    - Помилковий висновок у тому, що ефект зумовлений випадковістю (тоді, як і він є дійсним).
- Якщо результат лежить поза випадкової варіації, то прийнято казати, що він є статистично значущим.

7.2 Множинне тестування

Чи можна застосувати такий же самий підхід для множинного тестування?

Наприклад: є 20 передбачувальних змінних і одна змінна результату, і всі вони згенеровані випадковим чином. В такому випадку є досить хороші шанси на те, що принаймні один провісник (хибним чином) виявиться статистично значущим, якщо виконати серію з 20 перевірок значимості з альфа лише на рівні 0,05 – це помилка 1-го роду.

Імовірність, що одна з змінних пройде перевірку, правильно показавши незначущість, дорівнює 0,95, тому ймовірність того, що кожен з 20 провісників пройде перевірку, правильно показавши незначущість, дорівнюватиме $0{,}95^{20} =$

0,36. Імовірність того, що принаймні один провісник (хибним чином) покаже значимість, є зворотною цієї ймовірності та дорівнює 0,64. Зазначене явище відоме як **інфляція альфи**.

«Якщо мучити дані занадто довго, то рано чи пізно вони дадуть свідчення». Чим більше змінних ви додаєте або більше моделей виконуєте, тим більша ймовірність того, що щось проявиться як значуще просто по чистій випадковості. У завданнях контрольованого самонавчання контрольний набір із відкладеними даними, де моделі визначаються на даних, які модель не бачила раніше, знижує цей ризик. У завданнях статистичного та автоматичного навчання, не пов'язаних із позначеним відкладеним набором, зберігається ризик приходу до висновків, що ґрунтуються на статистичному шумі.

Способи вирішення:

- Тестування не окремих підгруп даних, а всіх даних в сукупності
- Процедура коригування Бонферроні: ділення альфа на число порівнянь.
- «Чесна значуща різниця» Тьюкі, або HSD Тьюкі (honest significant difference): використовується при порівнянні декількох групових середніх/

7.2.1 Приклади множинного тестування

Для варіантів A-C ви могли б запитати:

- Чи відрізняється B від A?
- Чи відрізняється від C від B?
- Чи відрізняється A від C?

У клінічному випробуванні ви, можливо, захочете подивитися на результати терапії на кількох етапах. Інші приклади множинного тестування:

- порівняння численних попарних різниць за всіма групами;
- розгляд результатів численних підгруп («ми не знайшли значущого ефекту варіанта за сукупністю, але ми знайшли ефект для незаміжніх жінок молодше 30»);
- випробування великої кількості статистичних моделей;
- включення великої кількості змінних у моделі;
- постановка великої кількості різних питань (тобто різних можливих результатів).

Сухий залишок для дослідників даних щодо множинного тестування такий:

- для передбачуваного моделювання ризик отримання ілюзорної моделі, очевидна ефективність якої є продуктом випадковості,

зменшується за рахунок перехресного контролю та використання відкладеної вибірки;

- для інших процедур без зазначеного відкладеного набору, призначеного для перевірки роботи моделі, ви повинні покладатися на:
    - усвідомлення, що чим більше ви опитуєте дані та ними маніпулюєте, тим більше шансів те що, що у гру вступить випадок;
    - евристики на основі повторного відбору та симуляції з метою забезпечення випадкових зразків, з якими можуть зіставлятися результати, що спостерігаються;
    - для ситуацій, пов'язаних з множинними статистичними порівняннями (тобто множинними перевірками значущості), існують статистичні коригувальні процедури.

7.3 Дисперсійний аналіз

Припустимо, замість A/B-тесту ми порівнювали численні групи, скажімо A, B, C і D, кожна з яких містить числові дані. Статистична процедура, яка перевіряє статистично значущу різницю серед груп, називається дисперсійним аналізом (ANalysis Of VAriance, ANOVA).

Розглянемо набір даних про «прилипність» користувачів до сторінок на нашому сайті. Очевидно, що ми зацікавлені в тому, щоб користувач якомога довше був на нашому сайті. Різним користувачам випадково запропоновано кілька варіантів сторінки. Дані про час відвідування в секундах показано в табл. 7.2.

Табл. 7.2. Дані про «прилипність» до сторінок.

	Сторінка 1	**Сторінка 2**	**Сторінка 3**	**Сторінка 4**
	164	178	175	155
	172	191	193	166
	177	182	171	164
	156	185	163	170
	195	177	176	168
Середнє	172	185	176	162
Загальне середнє	173,75			

Тепер з'являється головоломка (рис. 7.1). Коли ми порівнювали лише дві групи, все було просто: ми дивилися лише на різницю між середніми кожної групи. В умовах чотирьох середніх існує шість можливих порівнянь між групами:

- сторінка 1 порівняно зі сторінкою 2;
- сторінка 1 порівняно зі сторінкою 3;
- сторінка 1 порівняно зі сторінкою 4;
- сторінка 2 порівняно зі сторінкою 3;
- сторінка 2 порівняно зі сторінкою 4;
- сторінка 3 порівняно із сторінкою 4.

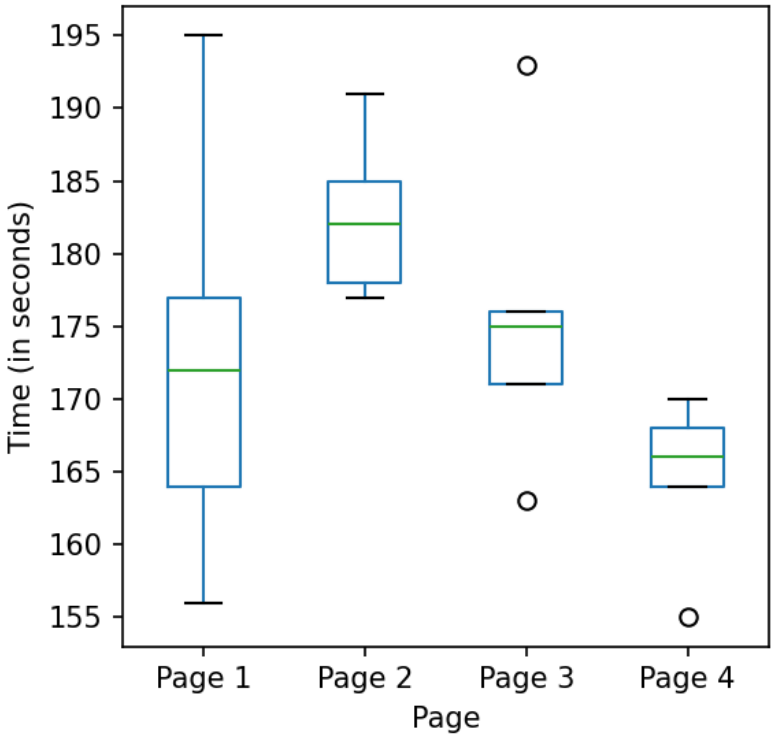


Рис. 7.1 – Коробкові діаграми для часу відвідування веб-сторінок.

Чим більше ми робимо таких попарних порівнянь, тим більшим є потенціал для того, щоб виявитися ошуканим випадковістю. Замість того, щоб турбуватися про всі можливі порівняння між окремими сторінками, які ми могли б провести, можна виконати лише один сукупний тест, який дає відповідь на запитання: чи можуть усі сторінки мати в своїй основі однакову прилипність, і різниці між цими сторінками бути обумовлені випадковістю. Як загальний набір часів сеансів розподілявся між чотирма сторінками?

Процедура дисперсійного аналізу

1. Об'єднати усі дані в одній коробці.
2. Перетасувати та витягти чотири повторні вибірки з п'ятьма значеннями в кожній.
3. Записати середнє значення кожної із чотирьох груп.
4. Записати дисперсію серед середніх значень чотирьох груп.
5. Повторити кроки 2-4 багато разів (скажімо, 1000).

p-значення, розраховане даних 0,09278. Іншими словами, маючи ту ж саму опорну прилипливість, у 9,3% випадків інтенсивність відгуків серед чотирьох сторінок могла відрізнятися тією ж мірою, якою фактично спостерігалася, по чистій випадковості. Цей ступінь неймовірності недотягує до традиційного статистичного порогу в 5%, тому ми робимо висновок, що різниця між чотирма сторінками могла виникнути випадково.

7.4 F-статистика

Аналогічно перевірці на основі t-статистики, яка може використовуватись замість перестановного тесту для порівняння середніх значень двох груп, для дисперсійного аналізу існує статистична перевірка на основі F-статистики.

Зазначена F-статистика спирається на відношення дисперсії по всіх групових середніх (тобто варіантного ефекту) до дисперсії внаслідок залишкової помилки. Чим вище це відношення, тим більше статистично значущим є результат. Якщо дані підпорядковуються нормальному розподілу, то статистична теорія зобов'язує статистику мати певний розподіл. На цій підставі є можливість обчислити p-значення.

ANOVA розділяє загальну варіацію даних на дві частини:

- міжгрупова дисперсія: варіація між середніми значеннями груп (наскільки середні значення груп відрізняються від загального середнього значення).
- внутрішньогрупова дисперсія: варіація всередині кожної групи (наскільки точки даних варіюються навколо середнього значення групи).

F-статистика – це відношення цих двох дисперсій:

$$F = \frac{\text{середні квадрати (міжгрупова дисперсія)}}{\text{середні квадрати (внутрішньогрупова дисперсія)}} \qquad (7.3)$$

Яка інтерпретація цього співвідношення? Якщо середні значення груп ідентичні, дисперсія між групами повинна бути невеликою (подібною до дисперсії всередині групи). Якщо середні значення груп відрізняються, дисперсія

між групами стає великою порівняно з дисперсією всередині групи. Високе значення F – сильний доказ того, що середні значення груп не є однаковими.

Двосторонній дисперсійний аналіз. Щойно описаний A-B-C-D-тест є одностороннім дисперсійним аналізом, в якому ми маємо один фактор, що варіюється (групу). Але ми могли б залучити другий фактор, скажімо «вихідний день порівняно з буднім днем», де дані збираються за кожною комбінацією (вихідний день групи А, будній день групи В, вихідний день групи В тощо). Це був би двосторонній дисперсійний аналіз, і ми будемо працювати з ним так само, як і з одностороннім дисперсійним аналізом шляхом виявлення «ефекту взаємодії» групи і відшукуємо різницю між середніми для цих підмножин та варіантним середнім. Можна бачити, що регулярний дисперсійний аналіз, а потім двосторонній дисперсійний аналіз є першими кроками по дорозі до повної статистичної моделі, такої як регресія та логістична регресія, в якій можуть бути змодельовані численні фактори та їх ефекти.

Ключові ідеї для дисперсійного аналізу

- Дисперсійний аналіз (ANOVA) – це статистична процедура для аналізу результатів експерименту з численними групами.
- Зазначена процедура є розширенням аналогічних процедур для A/B тесту, що використовується для визначення, чи сукупна варіація серед груп всередині діапазону випадкової варіації.
- Корисним результатом дисперсійного аналізу є виявлення дисперсійних компонентів, асоційованих з груповими варіантами, ефектами взаємодії та помилками.

7.5 Хто такі багаторукі бандити?

Багаторукі бандити пропонують підхід до тестування, особливо веб-тестування, який дає змогу виконувати явну оптимізацію й ухвалювати швидші рішення, ніж традиційний статистичний підхід до планування експериментів. Традиційний A/B-тест передбачає збір даних в експерименті відповідно до визначеного дизайну для відповіді на конкретне питання, наприклад: «Що краще, лікування А чи лікування Б?». Передбачається, що після отримання відповіді на це питання експеримент закінчується і ми переходимо до дій на основі отриманих результатів. Ви, мабуть, бачите кілька труднощів у такому підході. По-перше, наша відповідь може бути непереконливою: «ефект не доведений». Іншими словами, результати експерименту можуть вказувати на ефект, але якщо ефект є, ми не маємо достатньо великої вибірки, щоб довести його (відповідно до традиційних статистичних стандартів). Яке рішення ми приймаємо? По-друге, ми можемо захотіти почати використовувати результати, отримані до завершення експерименту. По-третє, ми можемо захотіти мати право змінити свою думку або

спробувати щось інше на основі додаткових даних, отриманих після завершення експерименту. Традиційний підхід до експериментів і перевірки гіпотез датується 1920-ми роками і є досить негнучким. Поява потужних комп'ютерів і програмного забезпечення дозволила застосовувати більш гнучкі підходи. Більше того, наука про дані (і бізнес загалом) не так переймається статистичною значущістю, як оптимізацією загальних зусиль і результатів. Алгоритми бандитів, які дуже популярні в веб-тестуванні, дозволяють тестувати кілька варіантів одночасно і доходити висновків швидше, ніж традиційні статистичні моделі. Вони отримали свою назву від ігрових автоматів (рис. 7.2), які використовуються в азартних іграх, також званих однорукими бандитами (оскільки вони налаштовані таким чином, що витягують гроші з гравця в постійному потоці).

Рис. 7.2 – «Однорукий бандит».

Ключові терміни для багаторуких бандитів

- Багаторукий бандит – уявний ігровий автомат з кількома важелями, або руками, на які гравець може натискати на вибір, при цьому кожна рука має різний виграш; тут узятий як аналогія багатоваріантного експерименту.
- Важіль (рука) – варіант в експерименті (наприклад, «заголовок А у веб-тесті»).
- Виграш – експериментальний аналог виграшу в автоматі (наприклад, «клієнт клацає на посилання»).

Якщо уявити ігровий автомат з більш ніж одним важелем, кожен з яких виплачує гроші з різною швидкістю, ви отримаєте багаторукого бандита, що є повною назвою цього алгоритму. Ваша мета — виграти якомога більше грошей і, більш конкретно, якомога швидше визначити і зупинитися на виграшній руці. Складність полягає в тому, що ви не знаєте, з якою загальною швидкістю виплачують руки — ви знаєте тільки результати окремих потягувань за руки.

Припустимо, що кожен «виграш» становить однакову суму, незалежно від того, яка «рука». Відмінність полягає в ймовірності виграшу. Нехай спочатку ви пробуєте кожну ручку 50 разів і отримуєте такі результати:

- ручка A: 10 виграшів із 50;
- ручка B: 2 виграші із 50;
- ручка C: 4 виграші із 50.

Один з крайніх підходів полягає в тому, щоб сказати: «Схоже, ручка A є переможцем – давайте припинимо пробувати інші ручки і залишимося при A». Це дозволяє повною мірою скористатися інформацією, отриманою під час початкового випробування. Якщо A дійсно кращий, ми отримуємо перевагу на ранньому етапі. З іншого боку, якщо B або C дійсно кращі, ми втрачаємо будь-яку можливість це виявити.

Інший крайній підхід полягає в тому, щоб сказати: «Все це виглядає як випадковість – давайте продовжувати вибирати їх порівну». Це дає максимальну можливість альтернативам A проявити себе. Однак у процесі цього ми застосовуємо те, що здається гіршими методами лікування. Як довго ми можемо це дозволяти?

Алгоритми бандитів використовують гібридний підхід: ми починаємо частіше вибирати A, щоб скористатися його очевидною перевагою, але не відмовляємося від B і C. Ми просто вибираємо їх рідше. Якщо варіант A продовжує перевершувати інші, ми продовжуємо перерозподіляти ресурси (використання) від варіантів B і C і частіше використовуємо варіант A. Якщо, з іншого боку, варіант C починає показувати кращі результати, а варіант A – гірші, ми можемо перерозподілити використання з варіанту A назад на варіант C. Якщо один з них виявляється кращим за варіант A, а це було приховано в початковому випробуванні через випадковість, тепер він має можливість проявитися в подальших випробуваннях.

Тепер подумайте про застосування цього до веб-тестування. Замість декількох ручок ігрових автоматів, ви можете мати декілька пропозицій, заголовків, кольорів тощо, які тестуються на веб-сайті. Клієнти або клацають (це «виграш» для продавця), або не клацають. Спочатку пропозиції показуються випадково і рівномірно. Однак, якщо одна пропозиція починає перевершувати інші, її можна показувати («витягувати») частіше.

Але якими мають бути параметри алгоритму, що змінює частоту витягування? На які «частоти витягування» ми повинні змінити і коли ми повинні

змінити? Ось один простий алгоритм, алгоритм епсилон-жадібності для A/B-тестування:

Епсілон-жадібний алгоритм багаторукого бандиту

1. Згенерувати рівномірно розподілене випадкове число в інтервалі між 0 та 1.
2. Якщо число знаходиться між 0 та епсілон (де епсілон — це число між 0 та 1 у типовій ситуації досить мале), випадковим чином обрати один з варіантів A, B, C,…
3. Якщо число більше або дорівнює ϵ, то показати будь-яку пропозицію, яка дотепер мала найвищу інтенсивність відгуків.

Епсилон – це єдиний параметр, який керує цим алгоритмом. Якщо епсилон дорівнює 1, ми отримуємо стандартний простий експеримент A/B (випадковий розподіл між A і B для кожного суб'єкта). Якщо епсилон дорівнює 0, ми отримуємо суто жадібний алгоритм — такий, що вибирає найкращий доступний варіант (локальний оптимум). Він не шукає подальших експериментів, а просто розподіляє суб'єктів (відвідувачів веб-сайту) за найкращим варіантом.

7.6 Модифікації епсілон-жадібного алгоритму багаторуких бандитів

Більш складний алгоритм використовує «вибірку Томпсона». Ця процедура «вибирає» (тягне за ручку бандита) на кожному етапі, щоб максимізувати ймовірність вибору найкращої ручки. Звичайно, ви не знаєте, яка ручка найкраща – в цьому і полягає вся проблема! – але, спостерігаючи за виграшем при кожному наступному розіграші, ви отримуєте більше інформації. Вибірка Томпсона використовує байєсівський підхід: спочатку припускається певний попередній розподіл винагород, використовуючи так званий бета-розподіл (це звичайний механізм для визначення попередньої інформації в байєсівській задачі). У міру накопичення інформації від кожного розіграшу ця інформація може оновлюватися, що дозволяє краще оптимізувати вибір наступного розіграшу, щоб вибрати правильний важіль.

Алгоритми бандита можуть ефективно обробляти 3+ методи лікування і рухатися до оптимального вибору «найкращого». Для традиційних процедур статистичного тестування складність прийняття рішень для 3+ методів лікування значно перевищує складність традиційного A/B-тестування, і перевага алгоритмів бандита набагато більша.

7.7 Контрольні питання

1. Що таке p-значення у контексті перестановочного тесту? Як його інтерпретувати?
2. Які саме кроки виконуються в перестановочному тесті при порівнянні двох варіантів A і B? Опишіть їх у ваших власних словах.

3. Як розраховується різниця у конверсії між двома варіантами та як вона впливає на вирішення задачі значимості?
4. Що таке інфляція альфа? Які фактори збільшують ймовірність виявлення хибної значущості у множинних порівняннях?
5. Відповідно до якої логіки будуються множинні попарні порівняння в прикладі A-B-C? Які можливі помилки тут збільшуються?
6. Що таке дисперсійний аналіз (ANOVA)? Чому він корисний замість окремих парних тестів?
7. Як інтерпретувати результат: «F-значення = 3,5, p = 0,04» у контексті порівняння кількох груп? Чи є це підставою відхилити нульову гіпотезу при α=0,05?
8. Що таке «багаторукий бандит»? Поясніть метафору в термінах веб-тестування.
9. Як визначаються «важелі» і «виграш» у контексті алгоритмів бандитів? Приведіть приклад, який відповідає веб-проєкту.
10. Наведіть практичний сценарій (можете придумати), коли використання бандитського алгоритму суттєво покращить результат порівняно з традиційним A/B тестом, і поясніть ключові параметри, що забезпечили таку ефективність.

Лекція 8 – Лінійні, метричні та ймовірнісні методи класифікації

8.1 Що таке класифікація?

Класифікація – це керований метод машинного навчання, коли модель намагається передбачити правильну мітку для заданих вхідних даних. Наприклад, алгоритм може навчитися передбачати, чи є даний електронний лист спамом або ні.

У класифікації існує два типи учнів: ліниві та активні учні.

Активні учні – це алгоритми машинного навчання, які спочатку будують модель на основі навчального набору даних, перш ніж робити будь-які прогнози щодо майбутніх наборів даних. Вони витрачають більше часу на процес навчання через своє бажання отримати краще узагальнення під час навчання від вивчення вагових коефіцієнтів, але їм потрібно менше часу, щоб робити прогнози. Більшість алгоритмів машинного навчання – це активні учні – алгоритми, що швидко навчаються, і нижче наведено кілька прикладів:

- логістична регресія;
- метод опорних векторів;
- дерева рішень;
- штучні нейронні мережі.

З іншого боку, ліниві учні або учні на основі прикладів не створюють жодної моделі одразу на основі навчальних даних, і саме звідси походить їхня лінивість. Вони просто запам'ятовують навчальні дані, і кожного разу, коли виникає необхідність зробити прогноз, вони шукають найближчого сусіда з усіх навчальних даних, що робить їх дуже повільними під час прогнозування. Деякі приклади такого роду:

- метод K-найближчих сусідів;
- міркування на основі конкретних випадків.

8.2 Логістична регресія

У разі побудови регресійної моделі ендогенна змінна кількісна. У моделях класифікації пояснюється якісна змінна. Розглянемо, як ідея регресії може бути застосована в цьому разі.

Логістична регресія або логіт-регресія – це статистична модель, що використовується для передбачення ймовірності виникнення деякої події p за значеннями множини ознак X. Імовірність p обчислюється за наступною формулою:

$$p = \frac{1}{1 + e^{-y}} \tag{8.1}$$

де y – значення регресії: $y = a + b_1 x_1 + b_2 x_2 + \cdots + b_n x_n$

На рисунку 8.1 показано кроки перетворення лінійної регресії в модель класифікації – логістичну регресію. Розраховані імовірності для кожної точки даних із використанням побудованої логістичної кривої показано на рис. 8.2.

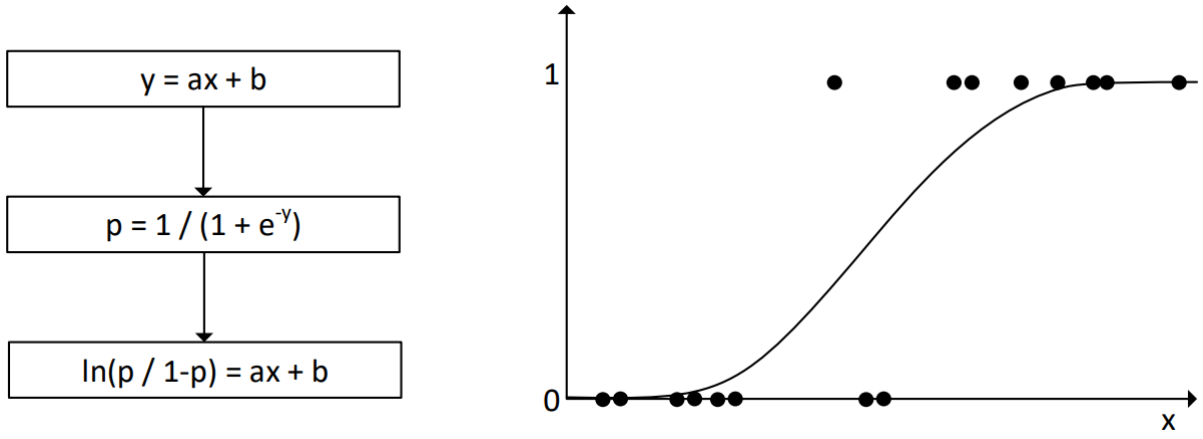


Рис. 8.1 – Кроки перетворення лінійної регресії в логістичну регресію (модель класифікації).

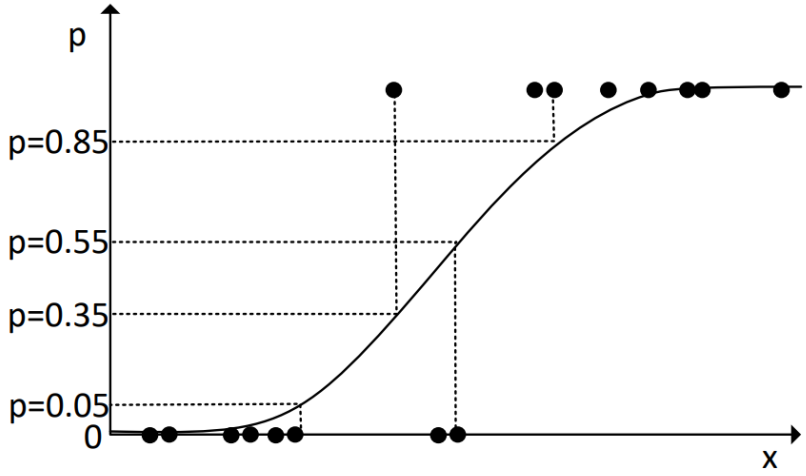


Рис. 8.2 – Розраховані імовірності для точок даних.

Фактично, під час побудови логістичної регресії оцінюється співвідношення шансів OR (odds ratio), тобто співвідношення ймовірності того, що подія відбудеться і ймовірності того, що подія не відбудеться:

$$OR = \frac{p}{1-p} \tag{8.2}$$

де p – ймовірність успіху, $\log(OR) = y$

В межах моделі об’єкт належить до одного з класів $\{0;\ 1\}$ з огляду на те, чи перевищує його оцінка ймовірності поріг відсікання k

$$\hat{y} = \{0, p < k; 1, p \geq k\} \quad (8.3)$$

Розраховані належності до класів 1 та 0 показано на рис. 8.3.

Для оцінки коефіцієнтів логістичної регресії використовують метод максимальної правдоподібності. Основою методу є функція правдоподібності (likehood function), що виражає щільність ймовірності спільної появи результатів вибірки.

Для побудови логістичної регресії навчальна вибірка готується стандартно з тією відмінністю, що вихідне поле може бути тільки дискретного типу. На етапі визначення входів моделі необхідно пам’ятати, що для успішного навчання кількість прикладів має в кілька разів перевищувати кількість вхідних ознак. За малої кількості даних доводиться штучно спрощувати структуру регресійній моделі, залишаючи найбільш істотні ознаки. Для вихідного поля (залежної змінної) необхідно визначити, що є негативною, а що позитивною подією. Це залежить від конкретного завдання. Наприклад, якщо прогнозується ймовірність наявності захворювання, то позитивним результатом буде клас «Хворий пацієнт», негативним – «Здоровий пацієнт».

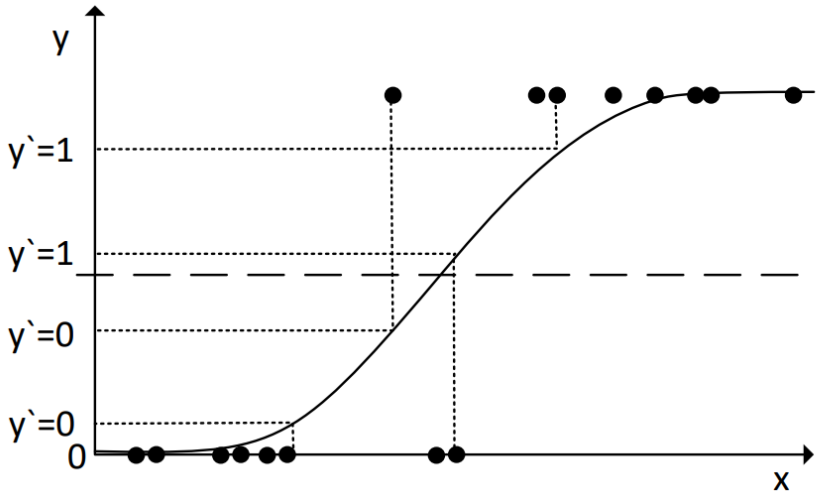


Рис. 8.3 – Належність точок даних до класів 0 та 1.

8.3 Оцінка результатів класифікації

В результаті побудови моделі виникають чотири варіанти класифікації:

- **TP (True Positives)** – правильно класифіковані позитивні приклади (істинно позитивні випадки);
- **TN (True Negatives)** – правильно класифіковані негативні приклади (істинно негативні випадки);

- **FN (False Negatives)** – позитивні приклади, класифіковані як негативні (помилка II типу). Це так званий «помилковий пропуск», коли подія, що цікавить нас помилково не виявляється (хибнонегативні приклади);
- **FP (False Positives)** – негативні приклади, класифіковані як позитивні (помилка I типу). Це помилкове виявлення, тому що за браком події помилково ухвалюється рішення про її наявність (хибнопозитивні випадки).

На основі результатів класифікації розраховуються такі метрики класифікації:

- точність моделі:

$$\text{точність} = \frac{TP + TN}{Total}: \tag{8.4}$$

- частка помилок:

$$\text{частка помилок} = \frac{FP + FN}{Total} \tag{8.5}$$

- чутливість (Sensitivity) – частка істинно позитивних випадків, що були правильно ідентифіковані моделлю:

$$Se = \frac{TP}{TP + FN} \tag{8.6}$$

- специфічність (Specificity) – частка істинно негативних випадків, які були правильно ідентифіковані моделлю

$$\text{Sp} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{8.7}$$

8.4 Метод опорних векторів

Основна ідея класифікатора на опорних векторах полягає в тому, щоб будувати розподіляючу поверхню з використанням невеликої підмножини точок, що перебуватимуть у зоні, критичної для поділу, тоді як інші спостереження навчальної вибірки ігноруються (є «резервуаром» для оптимізаційного алгоритму).

Метод опорних векторів будує функцію класифікації F у вигляді

$$F(x) = sign(\langle w, x\rangle + b) \tag{8.8}$$

де w – нормальний вектор, що розподіляє гіперплощини, b – допоміжний параметр. Ті об’єкти, для яких $F(x) = 1$ належать до одного класу, об’єкти, для яких $F(x) = -1$ – до іншого.

Необхідно знайти такі w і b, що максимізують відстань до кожного класу $-\frac{1}{\|w\|}$. Запишемо це у вигляді завдання оптимізації, що є стандартною задачею квадратичного програмування і вирішується за допомогою множників Лагранжа (пошук мінімума $-\frac{1}{\|w\|}$ еквівалентний пошуку максимума $\|w\|$)

$$\begin{cases} \arg\max\|w\|^2, \\ y_i(< w, x_i > +b) \geq 1 \end{cases} \tag{8.9}$$

Опорними векторами називаються спостереження, що перебувають безпосередньо на кордоні розподіляючої смуги або на неправильному для свого класу боці щодо кордонів зазору (рис. 8.4).

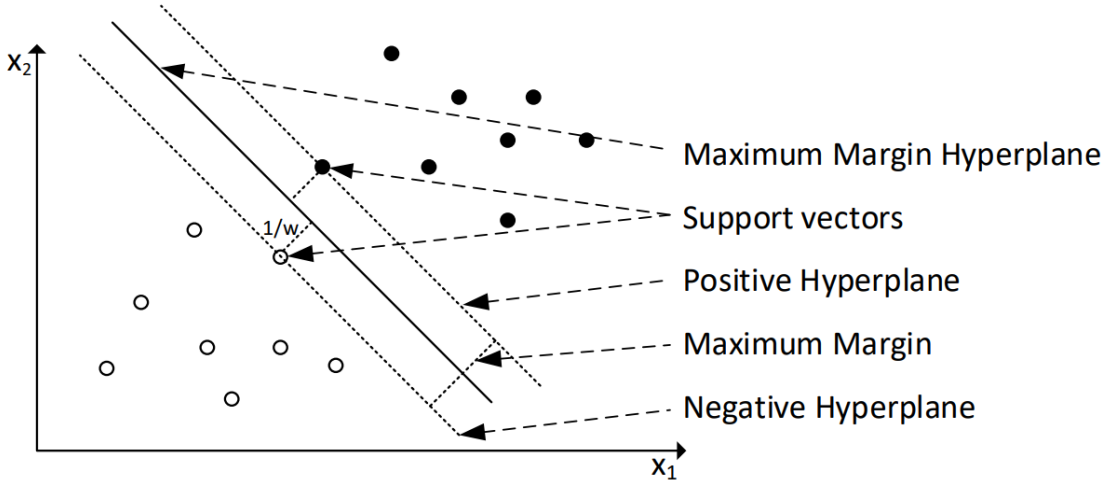


Рис. 8.4 – Розподіляюча пряма методу опорних векторів та опорні вектори.

Якщо є два класи спостережень і передбачається лінійна форма кордону між класами, то можливі два випадки:

- у разі лінійної розподільності спостережень можлива побудова гіперплощини $F(x) = \sum w_i x_i + b$. Оскільки таких гіперплощин може бути безліч, то оптимальною є така поверхня, яка максимально віддалена від навчальних точок, тобто має максимальний проміжок M(margin);
- в іншому випадку, хмари точок перекриваються і обидва класи лінійно нерозподільні.

Оптимальну розподіляючу гіперплощину такого класифікатора $F(x) = \sum w_i x_i + b$ також знаходять з умови максимізації ширини проміжку M, але при цьому дозволяється невірно класифікувати деяку невелику групу спостережень, що належать до опорних векторів. Для цього задається додаткова умова оптимізації, допустима кількість порушень кордону проміжку та їх вираженість, що зазвичай вибирається з використанням перехресної перевірки. Математично

пошук рішення зводиться до задачі квадратичної оптимізації з лінійними обмеженнями, що гарантовано сходиться до одного глобального мінімуму.

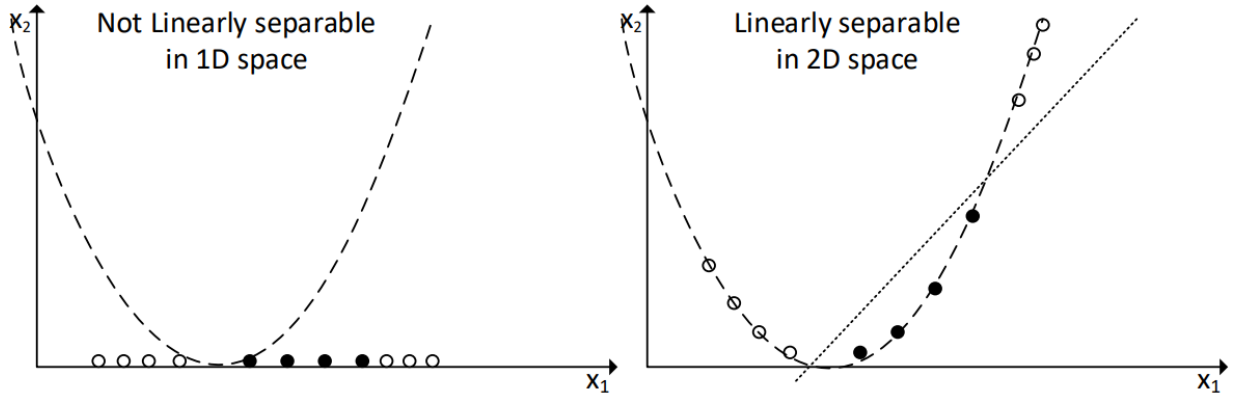
Оскільки на розташування гіперплощини впливають тільки ті спостереження, що перебувають на кордонах проміжку або порушують його, то вирішальне правило такого класифікатора є досить стійким до викидів більшості точок, розташованих поза «критичної зони» поділу. Ця властивість відрізняє його від інших класифікаторів.

Що робити із нелінійним зв'язком? За наявності нелінійного зв'язку між ознаками і відгуком якість лінійних класифікаторів може виявитися незадовільною. Для подолання проблеми нелінійності елементи навчальної вибірки вкладаються в простір X більш високої розмірності з допомогою відображення $\phi(X)$. Водночас відображення ϕ вибирається так, щоб в новому просторі X вибірка була лінійно розподільна. Класифікатор набуває вигляду:

$$F(x) = sign(\langle w, \phi(x)\rangle + b) \qquad (8.10)$$

Вираз $k(x, x') = \langle \phi(x), \phi(x')\rangle$ називається ядром класифікатора. З математичної точки зору ядром може бути будь-яка позитивно визначена симетрична функція двох змінних. Позитивна визначеність необхідна для того, щоб відповідна функція Лагранжа в задачі оптимізації була обмежена знизу, тобто задача оптимізації була б коректно визначена. Точність класифікатора залежить, зокрема, від вибору ядра.

Розглянемо приклад. Нехай на вході маємо одновимірні дані x_1, що не є лінійно розподільними (рис. 8.5а). Використовуючи квадратичне ядро відобразимо дані в двовимірний простір $\{x_1, x_2\}$, тепер дані стають лінійно розподільними (рис. 8.5б).



(а) (б)

Рис. 8.5 – Робота із лінійно нерозподільними даними в методі опорних векторів.

8.5 Метод k-найближчих сусідів

Основою kNN-класифікатора є гіпотеза компактності, що передбачає: тестований об'єкт d матиме таку ж мітку класу, як і навчальні об'єкти в локальній області його найближчого оточення. У варіанті 1NN аналізований об'єкт належить до певного класу в залежності від інформації про його найближчого сусіда. У варіанті kNN кожен об'єкт належить до переважного класу найближчих сусідів, де k – параметр алгоритму.

Вирішальні правила в методі kNN визначаються межами суміжних сегментів діаграми Вороного, що розподіляє площину на n опуклих багатокутників, кожен з яких містить один і тільки один об'єкт навчальної вибірки. В n-мірних просторах кордони рішень складаються з сегментів $(n-1)$-мірних напівплощин, утворених опуклими багатогранниками Вороного.

Приклад роботи. Алгоритм будується за принципом «більшості голосів», тобто як результат оголошується мітка класу-переможця. На рис. 8.6 тестований об'єкт при k = 5 буде віднесений до класу «чорних».

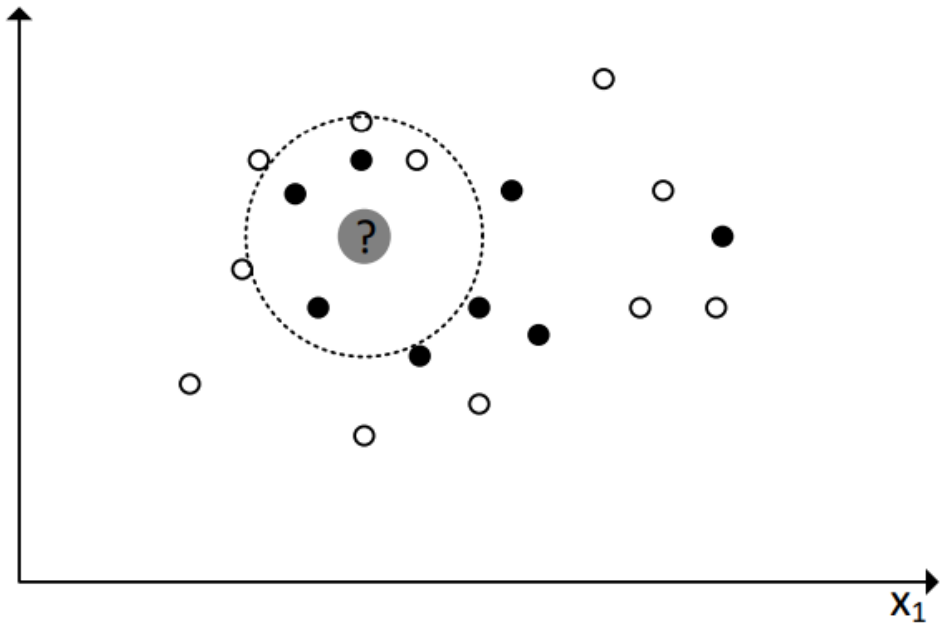


Рис. 8.6 – Визначення класу нового прикладу в методі k-найближчих сусідів.

8.6 Метод наївного Байєса

Наївний байєсівський класифікатор – ймовірнісний класифікатор, заснований на теоремі Байєса з нежорсткими припущеннями про незалежність подій, та задає формальний метод, що дозволяє в процесі ухвалення рішень врахувати нову інформацію. Основою байєсівської класифікації є гіпотеза

максимальної ймовірності, тобто вважається, що об'єкт А належить класу H_k, якщо досягається найбільша апостеріорна ймовірність:

$$\max\{P(H_i|A)\}. \tag{8.11}$$

За формулою Байєса:

$$P(H_i|A) = \frac{P(A|H_i)P(H_i)}{P(A)}, \tag{8.12}$$

де $P(A|H_i)$ - ймовірність зустріти об'єкт A серед об'єктів класу H_i; $P(H_i)$ та $P(A)$ – апріорні ймовірності класу H_i та об'єкта A (остання, не впливає на вибір класу, нею можна знехтувати).

Коротке нагадування теорії ймовірностей

- Подія А називається незалежною від події В, якщо ймовірність події А не залежить від того, чи відбулася подія В чи ні.
- Сумою кількох подій називається подія, що містить появу хоча б однієї із цих подій.
- Імовірність суми двох незалежних подій дорівнює сумі ймовірностей цих подій $P(A + B) = P(A) + P(B)$.
- Якщо якісь події утворюють повну групу подій, то сума їх ймовірностей дорівнює 1.
- Добутком кількох подій називається подія, що містить спільну появу всіх цих подій.
- Імовірність добутку двох незалежних подій дорівнює добутку їх ймовірностей.
- Імовірність добутку двох залежних подій дорівнює добутку ймовірності однієї з них та умовної ймовірності іншої за наявністю першої

$$P(AB) = P(A)P(B|A),$$
$$P(B|A) = P(B)P(A|B).$$

Якщо зробити «наївне» припущення, що всі ознаки, які описують об'єкти, що класифікуються, абсолютно рівноправні між собою та не пов'язані один з одним, то $P(A|H_i)$ можна обчислити як добуток ймовірностей зустріти ознаку A_j серед об'єктів класу H_i

$$P(A|H_i) = \prod_j P\left(A_j\middle|H_i\right), \tag{8.13}$$

де $P\left(A_j\middle|H_i\right)$ – імовірнісна оцінка вкладу ознаки A_j в те, що $A \in H_i$.

Розглянемо приклад. Нехай є два класи людей, одні схильні до кредитування (YES), інші – ні (NO). Необхідно долучити нову людину, дані про вік і доходи якої відомі, до одного з класів.

Скористаємося формулою Байєса:

$$P(NO|X) = \frac{P(X|NO)P(NO)}{P(X)}, \tag{8.14}$$

де $P(NO)$ – апріорна ймовірність; $P(X|NO)$ – умовна ймовірність; $P(X)$ – гранична ймовірність; $P(NO|X)$ – апостеріорна ймовірність.

Далі розраховуємо:

1. Апріорну ймовірність:

$$P(NO) = \frac{\text{к-сть об'єктів NO}}{\text{Загальну к-сть}}. \tag{8.15}$$

2. Граничну ймовірність:

$$P(X) = \frac{\text{к-сть подібних}}{\text{Загальну к} - \text{сть}}. \tag{8.16}$$

3. Умовну ймовірність:

$$P(X|NO) = \frac{\text{к-сть подібних серед NO}}{\text{Загальна к} - \text{сть } NO}. \tag{8.17}$$

Підставляючи результати розрахунків за формулами (8.15)–(8.17) в (8.14) отримаємо апостеріорну імовірність належності конкретного прикладу до класу NO.

8.7 Дерева рішень та похідні алгоритми

Дерево рішень (decision tree) як алгоритм машинного навчання – об'єднання логічних правил типу «ЯКЩО ... ТО ...» (if-then) в структуру «дерева», створюючи ієрархічну структуру правил. Під час побудови дерева рішень обчислюється приріст інформації (на основі оцінки ентропії). Ентропія відповідає ступеню хаосу в системі. Чим вище ентропія, тим менше впорядкована система і навпаки. Інформація протилежна ентропії.

Ентропія Шеннона визначається для системи з N можливими станами так:

$$H(Y) = -\sum p_i \log_2 p_i\,, \tag{8.18}$$

де p_i – ймовірності знаходження системи в i-му стані.

Ідея алгоритму побудови дерева рішень. Основою алгоритмів побудови дерева рішень є принцип жадібної максимізації приросту інформації – на кожному кроці вибирається та ознака, за якою під час розподілу приріст інформації виявляється найбільшим. Далі процедура повторюється рекурсивно, поки ентропія не буде дорівнювати нулю або якійсь малій величині (якщо дерево не підлаштовується ідеально під навчальну вибірку, щоб уникнути перенавчання). У різних алгоритмах застосовуються різні евристики для «ранньої зупинки» або «відсікання», щоб уникнути побудови перенавченого дерева.

Розглянемо приклад. Нехай є 10 куль, п'ять з яких білі, п'ять – чорні (рис. 8.7). Вони розміщені послідовно, і потрібно побудувати класифікатор для

передбачення кольору кулі. На початковому етапі ентропія системи максимальна і дорівнює 1.

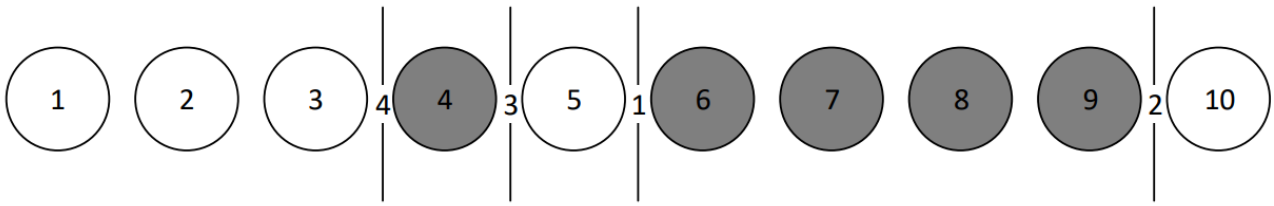


Рис. 8.7 – Дані для класифікації.

1. Проведемо перший поділ після номера 5

$$E_{11} = -\left[\frac{1}{5}\log\left(\frac{1}{5}\right) + \frac{4}{5}\log\left(\frac{4}{5}\right)\right] = 0.72,\ E_{12} = E_{11}.$$

2. Другий поділ проведемо після номера 9

$$E_{21} = 0, E_{22} = E_{21}.$$

3. Третій поділ проведемо після номера 4

$$E_{31} = -\left[\frac{3}{4}\log\left(\frac{3}{4}\right) + \frac{1}{4}\log\left(\frac{1}{4}\right)\right] = 0.81,\ E_{32} = 0.$$

4. Четвертий поділ проведемо після номера 3

$$E_{41} = E_{42} = 0.$$

Сфера застосування дерев рішень досить широка, в цьому розділі будемо використовувати цей апарат для вирішення завдань класифікації. Величезна перевага дерев рішень в тому, що вони легко інтерпретуються і зрозумілі для людини. Ще однією перевагою дерев рішень є їхня здатність виявляти нетипові випадки (на відміну, наприклад, від логістичної регресії).Для побудови дерева рішень навчальна вибірка готується стандартно, вихідне поле для дерева рішень єдине та дискретне.

Випадковий ліс. Для підвищення точності моделі з використанням дерев рішень також застосовуються ансамблеві алгоритми машинного навчання, зокрема випадковий ліс. Створюється багато випадкових підвибірок простим вибором з заміщенням. Модель навчається на кожній підвибірці, підсумки роботи усіх моделей усереднюються. Ефективність досягається завдяки тому, що базові алгоритми, що пройшли навчання на різних підвибірках, виходять досить різними, їхні помилки взаємно компенсуються, а також за рахунок того, що об’єкти-викиди можуть не потрапляти до деяких навчальних підвибірок. Випадковий ліс ефективний на малих вибірках, коли видалення навіть малої частини навчальних об’єктів призводить до побудови істотно різних моделей.

Градієнтний бустінг. Бустінг – це процедура послідовної композиції алгоритмів машинного навчання, коли кожен наступний алгоритм прагне компенсувати недоліки композиції попередніх алгоритмів. Тут також робляться вибірки даних, проте вже не за випадковою ознакою. Тепер кожна наступна вибірка складається з тих даних, на яких помилився попередній алгоритм. Бустінг над вирішальними деревами (градієнтний бустінг) вважається одним з найбільш ефективних методів з точки зору якості класифікації. У багатьох експериментах спостерігалося практично необмежене зменшення частоти помилок на незалежній тестовій вибірці в міру нарощування композиції. Більш того, якість на тестовій вибірці часто продовжує поліпшуватися навіть після досягнення безпомилкового розпізнавання всієї навчальної вибірки.

8.8 Контрольні питання

1. Що є основною відмінністю між активними і лінивими учнями у машинному навчанні?
2. Чому логістична регресія, метод опорних векторів, дерева рішень і штучні нейронні мережі відносяться до активних учнів?
3. Що таке "співвідношення шансів" (odds ratio) у контексті логістичної регресії?
4. Які чотири результати класифікації виникають після побудови моделі? Наведіть відповідні скорочення та їхній зміст.
5. Що таке опорні вектори у методі опорних векторів?
6. Чому у методі опорних векторів використовуються ядра? Наведіть приклад ядра.
7. Як визначається клас нового об'єкта у методі kNN?
8. Яке «наївне» припущення робить наївний байєсівський класифікатор? Яка формула використовується для обчислення апостеріорної ймовірності у наївному байєсівському класифікаторі?
9. Що таке ентропія у контексті дерев рішень? Як алгоритм дерев рішень вибирає ознаку для розподілу?
10. Яка основна перевага дерев рішень порівняно з логістичною регресією?
11. Які переваги має метод опорних векторів у порівнянні з іншими класифікаторами щодо стійкості до викидів?

Лекція 9 – Методи зменшення розмірності даних

Нерідко змінні варіюватимуться разом (совар'юватимуться), і частина варіації в одній змінній фактично дублюється варіацією в іншій (наприклад, рахунок і чайові в ресторані). Метод головних компонент – це технічний прийом виявлення шляху, яким числові змінні соваріюються.

Ключові терміни для аналізу головних компонент

- Головна компонента
  - Лінійна комбінація передбачувальних змінних.
- Навантаження
  - Ваги, які дають змогу перетворювати провісники на компоненти. Синонім: ваги.

9.1 Ідея методу головних компонент

Ідея методу головних компонент полягає в тому, щоб поєднати численні числові провісні змінні в менший набір змінних, які являють собою зважені лінійні комбінації вихідного набору.

- Менший набір змінних, головні компоненти, «пояснює» більшу частину варіабельності повного набору змінних, зменшуючи розмірність даних.
- Ваги, використовувані для формування головних компонент, розкривають відносні внески вихідних змінних у нові головні компоненти.

Метод головних компонент був уперше запропонований Карлом Пірсоном. У своїй статті, яка, можливо, стала першою роботою, присвяченою неконтрольованому самонавчанню, Пірсон визнавав, що в багатьох завданнях у передбачувальних змінних є варіабельність, і тому він розробив метод головних компонент як метод моделювання цієї варіабельності.

9.2 Простий приклад

Для двох змінних X_1 і X_2 є дві головні компоненти Z_i ($i = 1$ або $i = 2$) :

$$Z_i = w_{(i,1)}X_1 + w_{i,2}X_2. \quad (9.1)$$

Ваги $(w_{i,1}, w_{i,2})$ називаються навантаженнями компонент. Вони перетворюють вихідні змінні на головні компоненти. Перша головна компонента, Z_1 – це лінійна комбінація, яка найкраще пояснює сумарну варіацію. Друга головна компонента – Z_2 – ортогональна першій і пояснює більшість решти варіації, як може (якби існували додаткові компоненти, то кожна додаткова була б ортогональною до інших). Загальноприйнято також обчислювати головні компоненти на відхиленнях від середнього змінних, а не на самих значеннях.

Код

```
import pandas as pd
```

```
import matplotlib.pyplot as plt

from sklearn.decomposition import PCA

sp500 = pd.read_csv('sp500_data.csv', index_col=0)
print(sp500.head())

pcs = PCA(n_components=10)
pcs.fit(sp500)

sp500.plot.scatter(x='XOM', y='CVX')

plt.tight_layout()
plt.show()

explained_variance = pd.DataFrame(pcs.explained_variance_)
ax = explained_variance.plot.bar(legend=False)

plt.tight_layout()
plt.show()
```

Вивід

```
              ADS        CA      MSFT  RHT  CTSH       CSC       EMC  ...
1993-01-29  0.0  0.060124 -0.022100  0.0   0.0  0.018897  0.007368  ...
1993-02-01  0.0 -0.180389  0.027621  0.0   0.0  0.018889  0.018425  ...
1993-02-02  0.0 -0.120257  0.035900  0.0   0.0 -0.075573  0.029482  ...
1993-02-03  0.0  0.060124 -0.024857  0.0   0.0 -0.151128  0.003689  ...
1993-02-04  0.0 -0.360770 -0.060757  0.0   0.0  0.113350 -0.022114  ...
```

На рис. 9.1 показано перетворений набір даних за допомогою методу головних компонент. Пунктирні лінії показують напрямок двох головних компонент: перша проходить уздовж довгої осі еліпса і друга – уздовж короткої осі. Ви бачите, що більшість варіабельності у двох поверненнях акцій пояснюється першою головною компонентою. Це має сенс, оскільки курси енергетичних акцій мають тенденцію переміщатися всією групою. Метод головних компонент центрує дані, тобто середнє дорівнює нулю.

Обидві ваги першої головної компоненти від'ємні, але інвертування знака всіх ваг не змінює головної компоненти. Наприклад, використання ваг 0,747 і 0,665 для першої головної компоненти еквівалентно від'ємним вагам, так само як нескінченна пряма, яка визначається початком координат і точкою (1,1), є однаковою з прямою, яка визначається початком координат і точкою (-1,-1).

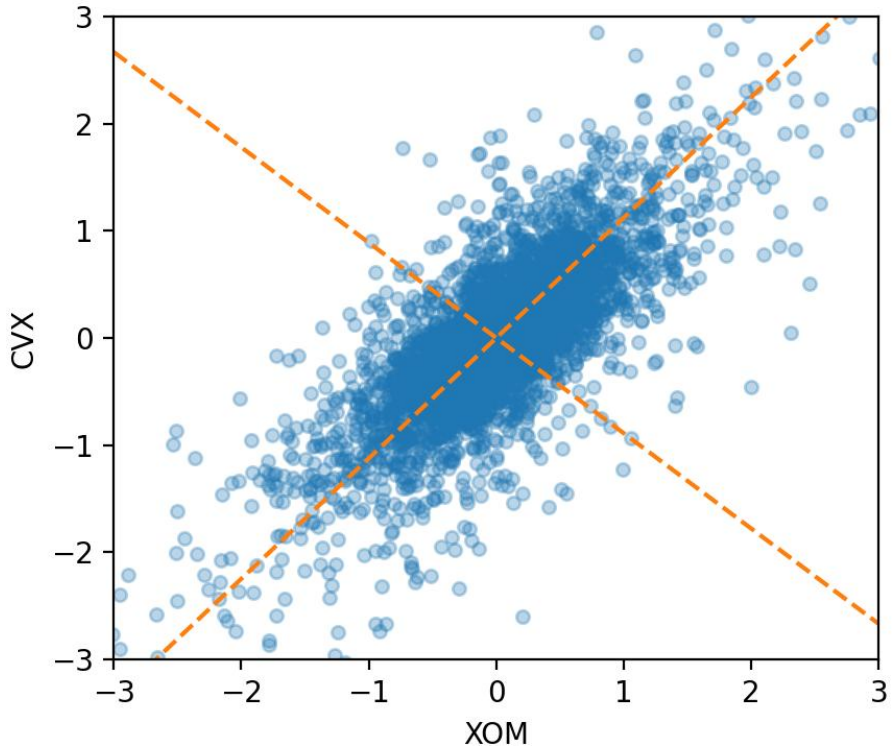


Рис. 9.1 – Головні компоненти для повернення акцій Chevron та ExxonMobil.

9.3 Обчислення головних компонент

Перехід від двох змінних до більшої кількості змінних є досить прямолінійним. Для першої компоненти потрібно просто внести додаткові передбачувані змінні до лінійної комбінації, призначаючи ваги, які оптимізують колекцію коваріації з усіх передбачуваних змінних у цю першу головну компоненту.

Обчислення головних компонент – це класичний статистичний метод, що спирається на кореляційну матрицю даних або на матрицю коваріацій, і виконується дуже швидко, не залежачи від ітерації. Як зазначено раніше, аналіз головних компонент працює тільки з числовими змінними, не категоріальними.

Алгоритм обчислення головних компонент

1. При створенні першої головної компоненти аналіз головних компонент приходить до лінійної комбінації передбачувальних змінних, яка максимізує відсоток поясненої сумарної дисперсії.
2. Ця лінійна комбінація далі стає першим «новим» провісником Z_1.
3. Аналіз головних компонент повторює цей процес, використовуючи ті самі змінні, але з різними вагами, щоб створити другий новий провісник, Z_2 . Зважування виконується так, щоб Z_1 і Z_2 не корелювалися.

4. Процедура триває доти, доки не буде стільки нових змінних, або компонент Z_i ,скільки вихідних змінних X_i.
5. Залишити стільки компонент, скільки потрібно для того, щоб було охоплено більшу частину дисперсії.
6. У цьому місці вийде набір ваг для кожної компоненти.
    - Останній крок полягає в конвертуванні вихідних даних у нові оцінки головних компонент шляхом застосування ваг до вихідних значень.
    - Ці нові оцінки можуть далі використовуватися як зменшений набір передбачувальних змінних

9.4 Інтерпретування головних компонент

Природа головних компонент часто розкриває інформацію про структуру даних. Покажемо візуалізацію поясненої дисперсії кожної з компонент на рис. 9.2. Дисперсія першої головної компоненти є доволі великою (як це часто і буває), але інші верхні головні компоненти також важливі.

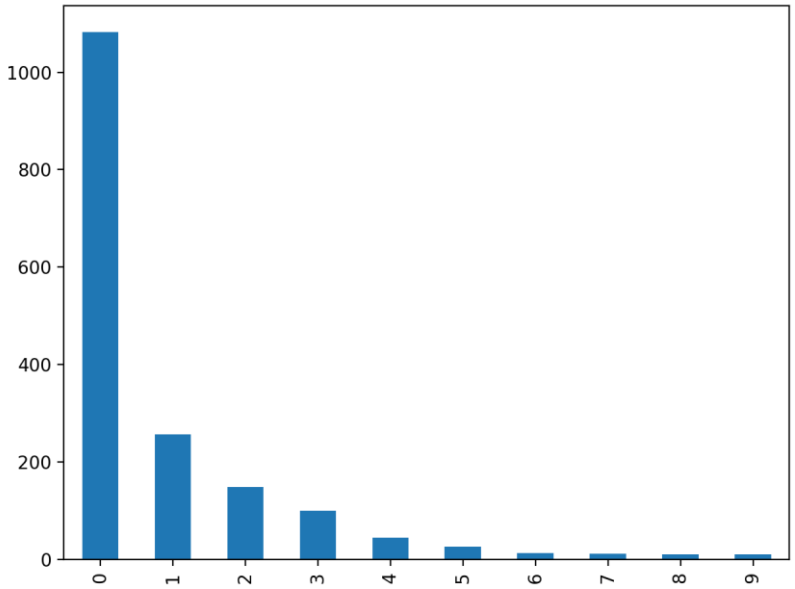


Рис. 9.2 – Пояснена дисперсія кожною з головних компонент для лідуючих акцій.

Аналіз головних компонентів (PCA) перетворює складні, корельовані змінні в новий набір некорельованих компонентів, які фіксують максимальну дисперсію в даних, пропонуючи спрощений підхід до інтерпретації.

Перший крок полягає у вивченні частки загальної варіації, що пояснюється кожною складовою зазвичай візуалізується за допомогою діаграми – для визначення кількості значущих складових; наприклад, перших двох або трьох часто достатньо, якщо вони разом пояснюють 70–90 % варіативності.

Важливо, що інтерпретація залежить від ретельного аналізу навантажень – ваг, що пов'язують вихідні змінні з кожною складовою, – де високі абсолютні значення вказують на сильний вплив; складова з великими позитивними навантаженнями для «доходу» та «рівня освіти» може бути інтерпретована як така, що представляє «соціально-економічний статус», тоді як складова, в якій переважають «температура» та «вологість», може сигналізувати про «кліматичні умови». Цей процес дозволяє уникнути надмірного пристосування, зосереджуючись на найважливіших закономірностях, а не лише на найбільших числах, і в кінцевому підсумку виявляє приховані структури, такі як ідентифікація ключових чинників поведінки клієнтів або спрощення високорозмірних наборів даних для більш чіткої візуалізації та подальшого аналізу, при цьому визнаючи, що названі компоненти є конструктами, отриманими з властивих даним кореляцій.

9.5 Скільки компонент вибрати?

Якщо ваша мета полягає в тому, щоб зменшити розмірність даних, то ви маєте ухвалити рішення про те, скільки головних компонент вибрати. Найпоширеніший підхід полягає у використанні нерегламентованого правила відбирати компоненти, які пояснюють «більшу частину» дисперсії. Це можна зробити візуально, наприклад, на рис. 9.2 було б цілком природно обмежити аналіз верхніми п'ятьма компонентами. Як альтернативу ви можете відібрати верхні компоненти таким чином, щоб кумулятивна дисперсія перевищувала поріг, приміром 80%. Крім того, ви можете проінспектувати навантаження, щоб визначити, чи має компонента інтуїтивно зрозумілу інтерпретацію.

Перехресний контроль надає більш формальний метод відбору числа значущих компонент. Метод головних компонент дозволяє позбутися мультиколінеарності.

Аналіз головних компонентів (PCA) ефективно усуває кореляцію між змінними шляхом математичної переорієнтації даних у нову систему координат, де осі вирівнюються з напрямками максимальної дисперсії в наборі даних. Ця трансформація, досягнута за допомогою ортогонального обертання вихідних змінних, створює нові компоненти, які статистично не корелюються між собою, тобто коваріація між будь-якими двома головними компонентами дорівнює точно нулю. Цей процес працює, оскільки PCA розкладає матрицю коваріації вихідних даних на власні вектори (напрямки найбільшої дисперсії) і власні значення (величина поясненої дисперсії), а потім проектує дані на ці ортогональні осі. В результаті перша головна компонента фіксує домінуючу модель варіації, не піддаючись впливу другої, яка фіксує наступну за величиною ортогональну модель, і так далі. Ця декореляція є фундаментальною для корисності PCA, оскільки вона дозволяє аналітикам працювати зі спрощеним набором незалежних змінних, вільних від надмірності корельованих ознак, зберігаючи при цьому основну структуру розподілу даних.

9.6 Контрольні питання

1. Яка основна мета методу головних компонент (PCA)?
2. Як називаються ваги, які перетворюють передбачувальні змінні на головні компоненти?
3. Чому перед застосуванням PCA необхідно центрувати дані (відняти середнє)?
4. Що максимізує перша головна компонента?
5. Чому наступні головні компоненти ортогональні до попередніх?
6. Як PCA вирішує проблему мультиколінеарності?
7. Чим відрізняються навантаження від оцінок головних компонент?
8. Якщо в даних 12 числових змінних, скільки головних компонент можна отримати?
9. Що означає «відсоток поясненої дисперсії» для компоненти?
10. Як визначити, скільки головних компонент залишити?
11. Що означає високе абсолютне значення навантаження для змінної в компоненті?
12. Чи можна застосовувати PCA до категоріальних змінних?
13. У прикладі з акціями Chevron та ExxonMobil, проаналізуйте чому перша головна компонента пояснює більшу частину варіації?
14. Якщо навантаження першої компоненти від’ємні, чи змінюється компонента при інвертуванні знаків?

Рекомендовані джерела інформації

1. Хабарлак К.С. Аналіз даних та знань [Електронний ресурс] : методичні рекомендації до виконання практичних робіт для здобувачів ступеня бакалавра спеціальності 124 Системний аналіз (F4 Системний аналіз та наука про дані) / К.С. Хабарлак, Л.С. Коряшкіна, Т.А. Желдак, Т.В. Хом'як ; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – Дніпро : НТУ «ДП», 2025. – 62 с.
2. Математичні методи інтелектуального аналізу даних: [навчальний посібник для здобувачів першого рівня вищої освіти спеціальності 124 Системний аналіз] / Т. Шабельник, О. Дяченко. – Маріуполь: МДУ, 2021. – 163 с
3. Кононова К. Ю. Машинне навчання: методи та моделі / К. Ю. Кононова. – Харків: ХНУ імені В. Н. Каразіна, 2020. – 301 с.
4. Хабарлак К.С. Самонавчання складних систем [Електронний ресурс] : конспект лекцій для здобувачів ступеня магістра освітньо-професійної програми «Системний аналіз» зі спеціальності 124 Системний аналіз / К.С. Хабарлак, Т.А. Желдак ; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – Дніпро : НТУ «ДП», 2024. – 112 с.
5. Документація бібліотеки машинного навчання scikit-learn. URL: https://scikit-learn.org .
6. Документація бібліотеки аналізу даних в Python: pandas. URL: https://pandas.pydata.org/ .
7. Practical Statistics for Data Scientists / P. Bruce, A. Bruce, P. Gedeck. – O'Reilly Media, 2020.
8. Хабарлак К.С. Аналіз та обробка великих даних [Електронний ресурс] : конспект лекцій для здобувачів ступеня магістра освітньо-професійної програми «Системний аналіз» зі спеціальності 124 Системний аналіз / К.С. Хабарлак, Т.В. Хом'як ; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – Дніпро : НТУ «ДП», 2024. – 111 с.

Навчальне видання

Хабарлак Костянтин Сергійович
Хом'як Тетяна Валеріївна

АНАЛІЗ ДАНИХ ТА ЗНАНЬ

Конспект лекцій
для здобувачів ступеня бакалавра
зі спеціальності 124 Системний аналіз
(F4 Системний аналіз та наука про дані)

Видано в авторській редакції.

Електронний ресурс.
Підписано до видання 01.08.2025. Авт. арк. 8,0.

Національний технічний університет «Дніпровська політехніка».
49005, м. Дніпро, просп. Дмитра Яворницького, 19.