МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
імені В. Н. КАРАЗІНА

МЕТОДИ АНАЛІЗУ «ВЕЛИКИХ ДАНИХ»

Методичні рекомендації з курсу
«Прикладні задачі аналізу великих даних» для здобувачів вищої освіти
спеціальності «Прикладна математика»

Харків – 2023

УДК: 519.2
М 54

Рецензенти:

Ю. В. Ромашов – д. техн. н., професор кафедри комп'ютерно-інтегрованих технологій, автоматизації та мехатроніки Харківського національного технічного університету радіоелектроніки;

О. М. Дацок – канд. техн. н., доцент кафедри біомедичної інженерії Харківського національного технічного університету радіоелектроніки.

Затверджено до друку рішенням Науково-методичної ради Харківського національного університету імені В. Н. Каразіна (протокол № 5 від 10.06.2021 р.)

М 54 **Методи** аналізу «великих даних» : методичні рекомендації з курсу «Прикладні задачі аналізу великих даних» для здобувачів вищої освіти спеціальності «Прикладна математика» / уклад. Н. М. Кізілова. – Харків : ХНУ імені В. Н. Каразіна, 2023. – 92 с.

У виданні систематизовані сучасні методи аналізу великих масивів даних, які належать до «великих даних» (Big Data), будування математичних моделей, систем для накопичення, зберігання і використання даних. Показано, як на основі результатів статистичної обробки великих даних будуються математичні моделі, які дозволяють пояснювати поведінку досліджуваної динамічної системи і прогнозувати її еволюцію залежно від різних можливих сценаріїв і параметрів. Розглянуті приклади обробки й аналізу медичних даних (показники серцево-судинної системи людини і динаміка розповсюдження епідемій), астрофізичних та ряду інших з відкритих джерел інформації.

Для здобувачів вищої освіти математичних, медичних, біологічних, екологічних спеціальностей, які займаються обробкою, аналізом «великих даних» і використанням результатів математичного моделювання в різних галузях науки.

УДК 519.2

ЗМІСТ

ВСТУП

«Великі дані» (Big Data) – це сукупність технологій, які дозволяють здійснювати таки операції:

1) обробляти значно більші, порівняно зі «стандартними» технологіями, об'єми даних;

2) вміти працювати з даними, які з великою швидкістю додаються у значних об'ємах. Тобто даних не просто багато, а їх постійно стає все більше й більше;

3) вміти працювати зі структурованими і мало стуктурованими даними паралельно і у різних аспектах.

Такі вміння дозволяють виявляти закономірності, які не можуть бути виявлені звичайними методами, що дає можливості оптимізації багатьох сфер нашого життя: державного управління, медицини, телекомунікацій, фінансів, транспорту, виробництва та ін.

Визначальними характеристиками для великих даних є, окрім їх великого об'єму (volume), ще й інші, які підкреслюють складність задачі обробки й аналізу цих даних, а саме:

1) підхід VVV (volume, velocity, variety), тобто стрімко зростають як фізичний об'єм, так і швидкість приросту даних, необхідність їх швидкої обробки і здатність обробляти дані різних типів. Був розроблений компанією Meta Group у 2001 р. з метою вказати на рівну значимість управління даними всіма трьома аспектами;

2) підхід 7V (volume, velocity, variety, veracity, viability, value, visualization) – аналогічний підхід, але розширений на більшу кількість ознак великих даних.

Підходи і методи Big Data корисні під час вирішення наступних задач:

- прогнозування ринкової ситуації;
- маркетинг і оптимізація продажів;
- вдосконалення продукції;
- ухвалення управлінських рішень;
- підвищення продуктивності праці;
- ефективна логістика;
- моніторинг стану основних фондів, ресурсів.

3) Запропоновані також і більш детальні підходи з більшою кількістю ознак.

«Світовий обсяг даних» аналітики оцінюють та прогнозують у розмірі:

2003 р. – $\sim 5 \cdot 10^9$ ГБ (1 ГБ = 10^9 байтів);

2020 р. – $\sim 4 \cdot 10^{12}$ ГБ;

2025 р. – $\sim 10^{14}$ ГБ,

причому більшу частину даних генерувати будуть не звичайні споживачі, а підприємства (промисловий інтернет речей та ін.).

Зростаюча важливість «великих даних» пов'язана з такими факторами, як ***цифрова трансформація суспільства***, яка включає наступні джерела даних:

1) ***Wearables (wearable technology***) – різні гаджети або одяг, який може інтерактивно взаємодіяти з навколишнім середовищем, сприймати та відсилати сигнали, обробляти інформацію і запускати у відповідь якісь дії;
2) ***цифровий двійник (Digital Twin)*** – цифрова копія фізичного об'єкта або процесу, яка допомагає моделювати та оптимізувати ефективність його роботи. Концепція «цифрового двійника» є частиною 4-ї промислової революції і покликана допомогти підприємствам швидше виявляти фізичні проблеми, точніше прогнозувати їх результати і виготовляти більш якісні продукти);
3) ***штучна нейронна мережа (ШНМ)*** – це математична модель, а також її програмне або апаратне втілення, яка побудована за принципом організації та функціонування біологічних нейронних мереж (систем нервових клітин мозку). ШНМ є системою з'єднаних і взаємодіючих між собою простих процесорів (штучних нейронів). З точки зору машинного навчання, нейронна мережа являє собою окремий випадок методів розпізнавання образів, дискримінантного аналізу, методів кластеризації і т. п. З математичної точки зору, навчання нейронних мереж – це багатопараметрична задача нелінійної оптимізації;
4) ***машинне навчання (Machine Learning)*** – це клас методів штучного інтелекту, характерною рисою яких є не пряме вирішення задачі, а навчання в процесі застосування готових рішень для великої кількості подібних завдань. Для побудови таких методів використовуються засоби математичної статистики, чисельних методів, методів оптимізації, теорії ймовірностей, теорії графів, а також різні техніки роботи з даними в цифровій формі. Розрізняють два типи навчання:
а) навчання за прецедентами (індуктивне навчання), засноване на виявленні емпіричних закономірностей у даних;
б) дедуктивне навчання (експертні системи) передбачає формалізацію знань експертів і їх перенесення в комп'ютер у вигляді бази знань;
5) ***коботи (коллаборативні роботи)*** – це автоматичні пристрої, які можуть працювати спільно з людиною для створення або виробництва різних продуктів. Як і звичайні промислові роботи, коботи складаються з маніпулятора і програми управління, яка генерує керування діями робота, визначає необхідні рухи виконавчих органів за допомогою маніпулятора. Колаборативні роботи застосовуються на виробництві в вирішенні промислових задач, які не можна повністю автоматизувати;
6) ***штучний інтелект (Artificial Intelligence)*** – це властивість інтелектуальних систем виконувати творчі функції, які властиві людині; наука і технології створення інтелектуальних машин, інтелектуальних комп'ютерних програм;
7) ***індустрія 4.0 (Industry 4.0)*** – це концепція 4-ї промислової революції, що охоплює сфери, які зазвичай не класифікуються як галузь, наприклад, «розумний дім», «розумне місто». На заводах «промисловості 4.0» є машини, які доповнені бездротовим підключенням і сенсорами, що підключені до системи, яка може візуалізувати всю виробничу лінію та приймати рішення. По суті, промисловість 4.0 – це тенденція до автоматизації та обміну даними

у виробничих технологіях та процесах, які включають кіберфізичні системи (CPS), інтернет речей (IoT), індустріальний інтернет речей (IIOT), хмарні обчислення, когнітивні обчислення та штучний інтелект. Концепція включає розумне виготовлення (Smart Industry), розумні фабрики (Smart Factory), розумні міста (Smart City) і т.п.;

8) ***інтелектуальне виробництво (Smart Factory)*** – це концепція виробництва з інтегрованими комп'ютерними технологіями, яка включає високий рівень адаптованості та швидкі зміни дизайну, цифрові інформаційні технології та більш гнучку підготовку технічної робочої сили. Інші цілі іноді включають швидкі зміни рівня виробництва на основі попиту, оптимізацію ланцюга поставок, ефективне виробництво та утилізацію. У цій концепції розумна фабрика має сумісні системи, багатомасштабне динамічне моделювання, інтелектуальну автоматику, сильну кібербезпеку та мережеві датчики;

9) ***інтелектуальне підприємство (Intelligent Enterprise)*** – це підхід до управління, який застосовує технології та нові парадигми обслуговування для вирішення проблеми підвищення ефективності бізнесу. Концепція була сформульована в книзі «Розумне підприємство» Джеймса Брайана Куінна («Вільна преса», 1992). Вона свідчить про те, що інтелект є основним ресурсом у виробництві та наданні послуг. Книга описує підхід до управління та обчислювальну інфраструктуру, необхідну для ефективного використання цього інтелекту. Цей підхід називають управлінням знаннями;

10) ***інтернет речей (Internet of Things, IoT)*** – це концепція обчислювальної мережі фізичних предметів (речей), що оснащені вбудованими технологіями для взаємодії один з одним або з зовнішнім середовищем, яка розглядає організацію таких мереж як явище, здатне перебудувати економічні та суспільні процеси, що виключає з частини дій і операцій необхідність участі людини. Концепція була сформульована в 1999 р. як осмислення перспектив широкого застосування засобів радіочастотної ідентифікації для взаємодії фізичних предметів між собою і з зовнішнім оточенням. Приклад: побутові прилади (будильник, кондиціонер), домашні системи (садового поливу, охоронна, освітлення), датчики (теплові, руху і освітленості) і «речі» (лікарські препарати з ідентифікаційною міткою) взаємодіють один з одним за допомогою комунікаційних мереж (інфрачервоних, бездротових, силових, слабкострумових) і забезпечують повністю автоматичне виконання процесів (вмикають кавоварку, змінюють освітленість, нагадують про прийом ліків, підтримують температуру, забезпечують полив саду, дозволяють зберігати енергію і керувати її споживанням;

11) ***промисловий інтернет речей*** – це система об'єднаних комп'ютерних мереж і підключених до них промислових (виробничих) об'єктів із вбудованими датчиками і програмним забезпеченням для збору та обміну даними, з можливістю віддаленого контролю й управління в автоматизованому режимі, без участі людини;

12) ***інтернет дронів (Internet of Drones, IoD)*** – це компонента інтернету речей, яка пов'язана з використанням дронів та інформацією, що отримана за їх допомогою. Це промисловий інструмент, який працює в галузі повітряних, водних, космічних, підземних інспекцій і підтримується в режимі реального часу з будь-якої точки світу в любому часі;
13) ***віртуальна реальність (Virtual reality)*** – це різновид реальності в формі тотожності матеріального й ідеального, що створюється та існує завдяки іншій реальності. У вужчому розумінні – ілюзія дійсності, створювана за допомогою комп'ютерних систем, які забезпечують зорові, звукові та інші відчуття. Технології створення віртуальної реальності: окуляри, шоломи, рукавиці віртуальної реальності, кімнати віртуальної реальності, контролери зі зворотним зв'язком;
14) ***доповнена реальність (Augmented reality, AR)*** – це термін, що позначає всі проекти, спрямовані на доповнення реальності будь-якими віртуальними елементами. AR – складова частина змішаної реальності (mixed reality), в яку також входить «доповнена віртуальність» (коли реальні об'єкти інтегруються у віртуальне середовище);
15) ***хмарні обчислення (Cloud Computing)*** – це модель забезпечення зручного мережевого доступу на вимогу до деякого загального фонду обчислювальних ресурсів (наприклад, мереж передачі даних, серверів, пристроїв зберігання даних, додатків і сервісів – як разом, так і окремо), які можуть бути оперативно надані та звільнені з мінімальними експлуатаційними витратами або зверненнями до провайдера. Споживачі хмарних обчислень можуть значно зменшити витрати на інфраструктуру інформаційних технологій і гнучко реагувати на зміни обчислювальних потреб, використовуючи властивості обчислювальної еластичності (Elastic computing) хмарних послуг;
16) ***інтернет загроз (Internet of Threats, IoTh)*** – це швидкий розвиток інтернету речей, який вимагає нового погляду на безпеку. Завдяки вразливості мережі та потенціалу перебоїв у виробничих процесах компаніям потрібно розробити нові стратегії пом'якшення та управління кібер-ризиками. IoTh містить: контентні ризики (неналежний контент, незаконний контент, електронна безпека, шкідливе ПЗ), спам, кібершахрайство) та комунікаційні ризики (незаконний контакт, кібер-переслідування, шкідливе ПЗ, рекламне ПО, шпигунське ПЗ, браузерні експлойт, спам, кібершахрайство – нігерійські листи, фішинг, вішинг, фармінг).

1. ВИДИ І СТРУКТУРА РІЗНИХ ТИПІВ «ВЕЛИКИХ ДАНИХ»

Приклад 10V підходу до «великих даних» стисло наведений на рис.1.

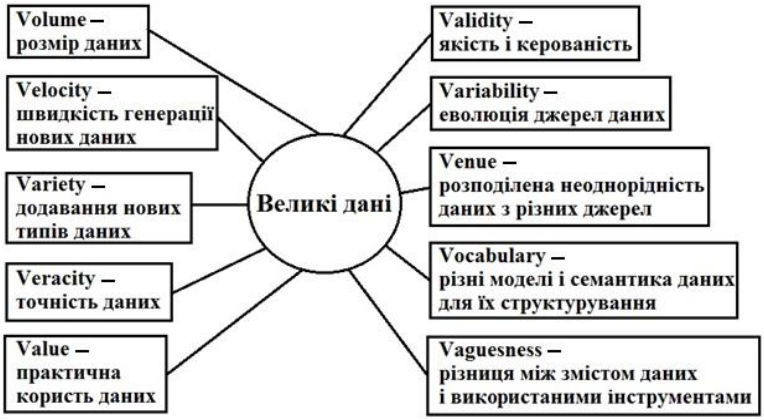


Рис. 1. Приклад 10V підходу до Big Data

Великі масиви даних можна отримувати шляхом довгочасних спостережень за динамічною системою (volume). При цьому результат спостережень являє собою часовий ряд або ряди, доповнений описом системи у вигляді набору інших характеристик, у вигляді

$$X = \{X(t_0), X(t_1), X(t_2), ...X(t_n)\} \equiv \{\mathrm{X}_j\}_{j=0}^{n}, \qquad (1.1)$$

де t_j, $j = 0,1,...,n$ – дискретні інтервали часу реєстрації вектору параметрів $\overline{X} = (x_1, x_2, ..., x_k)$ системи.

Історично найдовші часи спостережень відповідають даним спостережень зоряного неба і вимірювання координат об'єктів на небі. Із розвитком техніки спостережень і вимірювань з'являється можливість проводити вимірювання з меншим кроком Δt, що дає можливість вивчати більш швидкі процеси (динаміку подвійних зірок, пульсарів і т. д.). Обробка таких даних на різних масштабах часу дозволяє виявляти нові явища, які неможливо виявити на даних короткочасних спостережень. Крім того, з'являється нова високоточна техніка, яка дозволяє знаходити нові зоряні об'єкти, які раніше не вдавалося спостерігати, а також реєструвати нові параметри (ультрафіолетове (УФ), інфрачервоне (ІЧ) випромінювання, потоки γ-частинок і т.д.), тобто інша ознака «великих даних» – variety. В результаті масиви даних наповнюються зі все зростаючою швидкістю (volume).

1.1. Приклади «великих даних»

1.1.1. Динаміка сонячної активності

В якості прикладу розглянемо дані спостережень за активністю Сонця. Зміна сонячної активності починається з появи на поверхні зірки візуально більш світлої ділянки, яка згодом розростається, а її яскравість убуває. Після цього на її поверхні утворюються невеликі темні плями – пори, розмірами всього в декілька тисяч кілометрів (рис.1.1). Поступово розростаючись і зливаючись, вони утворюють плями. Вони холодніше навколишньої поверхні на 2000-3000 ºС (середня температура поверхні Сонця – 6000 ºС).

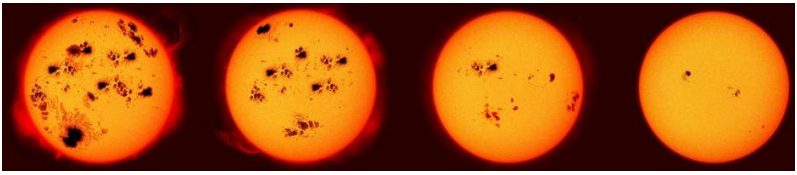


Рис. 1.1. Періодична динамка плям на Сонці

Перші повідомлення про плями на Сонці належать до спостережень 800 р. до н. е. в Китаї. У той же час спостерігачі висунули гіпотезу про вплив плям на людину. Майже одночасно про існування плям на Сонці в 1610 і 1611 рр. заявили Давид Фабриціус, патер Шейнер, Галілео Галілей і Томас Герріот. Пізніше був виявлений вплив сонячних плям на пацієнтів з інфарктом міокарда, у яких спостерігалися різкі ускладнення захворювань серцево-судинної системи в періоди активного плямоутворення.

Одним із доказів впливу сонячних плям на організм людини є роботи французьких лікарів Сарду і Фора (1922 р.). В результаті спостережень за 298 хворими протягом 9 міс. вони показали, що проходження плям через центральний меридіан Сонця збігається з 84 % випадків загострення різних симптомів хронічних захворювань і з появою важких ускладнень хвороби. Були також встановлені достовірні кореляції між числом плям та епілептичими нападами.

Іншою ознакою активності є сонячні спалахи – вибухи на поверхні Сонця, які відбуваються через раптове стиснення плазми під дією сильних магнітних полів. В результаті такого стиснення утворюється стрічка з плазми довжиною в десятки-сотні тисяч км. Розвиток спалахів за часом не однаковий і може варіювати від декількох хвилин до декількох годин. Спалахи теж безпосередньо впливають на стан людини. Медична статистика показує, що в день появи сонячного спалаху в 1,5–2 рази збільшується число людей зі скаргами на захворювання різних систем організму, а також число летальних випадків.

Для чисельної характеристики сонячної активності введено міжнародне відносне число сонячних плям (число Вольфа, Wolfer number W), назване на

честь швейцарського астронома Рудольфа Вольфа, яке обчислюється за формулою

$$W = k (f + 10g), \quad (1.2)$$

де f – кількість спостережуваних плям; g – кількість спостережуваних груп плям; k – коефіцієнт, визначений для кожного спостерігача і телескопа.

За міжнародну систему прийняті числа Вольфа, які в 1849 р. почала публікувати Цюрихская обсерваторія (k=1 в (1.2)). На сьогодні час зведення всіх спостережень і аналіз даних проводяться в Центрі аналізу даних щодо впливу Сонця SIDC[1] (Бельгія). Існують також ряди чисел Вольфа, відновлені за непрямими даними для епохи, що передує 1849 року.

Швейцарським астрономом М. Вальдмайером отримана емпірична залежність між середньорічними значеннями числа Вольфа і сумарною площею (F, у мільйонних частках півсфери) сонячних плям:

$$F = 16{,}7\ W, \quad (1.3)$$

але є низка вказівок на зміну характеру зв'язку (1.3) з часом.

Результати реєстрації різних показників Сонця, таких як f, g, F, параметри сонячної корони, інтенсивність і спектр ІЧ, УФ, оптична яскравість, спектр оптичного випромінювання (ОВ), зображення поверхні Сонця і сонячної корони тощо використовуються у вигляді загальної бази даних астрофізичних досліджень. Таким чином, набір виміряних даних складається як з часових рядів вигляду (1.1) зі змінним інтервалом Δt, так і з 2D–зображень, результатів перетворення електромагнітних сигналів у графіки спектр-потужність (рис. 1.2), описових даних (дата і час спостереження, відстань до Сонця, і т.д.). Наприклад, сервіс SIDC пропонує необмежений доступ до даних у визначених форматах для дослідницьких потреб (рис. 1.2). В двох білих вікнах виведені значення відповідних потоків енергії зареєстрованого електромагнітного випромінювання в різних частотних діапазонах, які доступні в табличному вигляді[2]. Таким чином, результати спостережень є ***структурованими даними***, які можуть бути записані в рядках і стовбцях відповідних таблиць.

Ці дані дають можливість обчислювати значення W (1.2), уточнювати кореляції вигляду (1.3), шукати кореляції інших часових рядів з параметрами сонячної активності. Деякі параметри ще не мають свого фізичного пояснення. Так, супутники реєструють сплески – це експериментальний продукт, який виявляє сплески, але поки не може їх класифікувати. Це показник локальної яскравості, який дає деяке уявлення про їх природу та/або інтенсивність: 0 – слабкий сплеск або несонячне випромінювання, 1–3 – сонячний сплеск зростаючої інтенсивності (1 – малий, 2 – середній, 3 – сильний).

[1] http://sidc.oma.be/
[2] http://sidc.oma.be/humain/flux_realtime.php

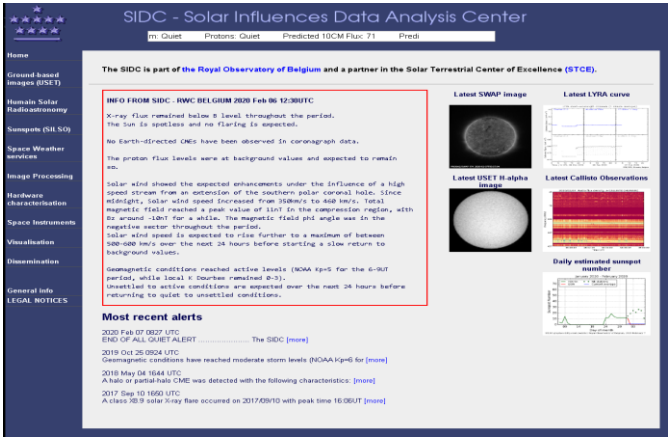


Рис. 1.2. Інтерфейс сервісу SIDC

На рис.1.3 наведені результати щоденних вимірювань іншого міжнародного показника активності Сонця S_n (International sunspot number) протягом 2008–2020 років. Отримана крива (жовта лінія) має значні осциляції і після осереднення за 30 діб дає більш згладжену криву (синя лінія), яка, в свою чергу, після згладжування дає так звану ***лінію тренду*** (червона лінія). Остання крива вказує на ~11–річний період коливань активності. Після обробки даних протягом століть і тисячоліть можна виявити закономірності, які дозволять будувати математичні моделі вигляду

$$\frac{dS_n(t)}{dt} = Q(S_n(t-\tau)), \quad S_n(t_0) = S_n^0 \tag{1.4}$$

і за їх допомогою передбачувати динаміку активності $S_n(t)$, а тому попереджати загрозу посилення захворювань, які пов'язані з максимумами активності.

Функція Q і час τ є невизначеними параметрами моделі. Статистичний аналіз дозволяє виявити відповідні апроксимації і знайти розв'язок задачі (1.4) у вигляді кривих $S_n(t)$, порівняння яких з даними спостережень в наступні роки дозволяє виявити найбільш точні апроксимації (червоні штрихові криві на рис.1.3 – результати різних прогнозів для Q і τ).

На рис.1.4 наведені результати спостережень за кількістю плям і аналіз нерегулярних довго- та короткоперіодичних осциляцій, які не можна помітити на даних спостережень принаймні <10 років так був виявлений ~11–річний цикл коливань протягом останніх 3 століть. Осереднені за роком криві (рис. 1.4а) дозволяють виявити і менш виражений віковий цикл коливань (рис. 1.4б).

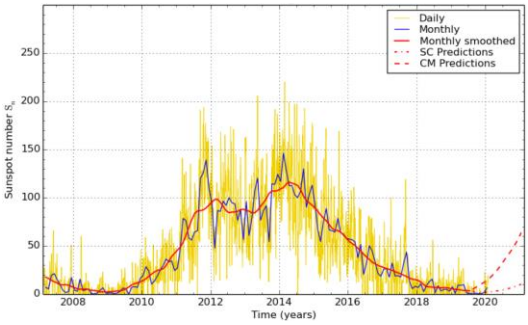


Рис. 1.3. Щоденні значення S_n за період 2008–2020 років (пояснення в тексті)

На приблизно десятирічну періодичність у збільшенні і зменшенні кількості сонячних плям на Сонці вперше звернув увагу в першій половині XIX сторіччя німецький астроном Г. Швабе, а потім – Р. Вольф. «11-річним» цикл називають умовно: його довжина за XVIII–XX ст. змінювалася від 7 до 17 років, а в XX ст. в середньому була ~10,5 років.

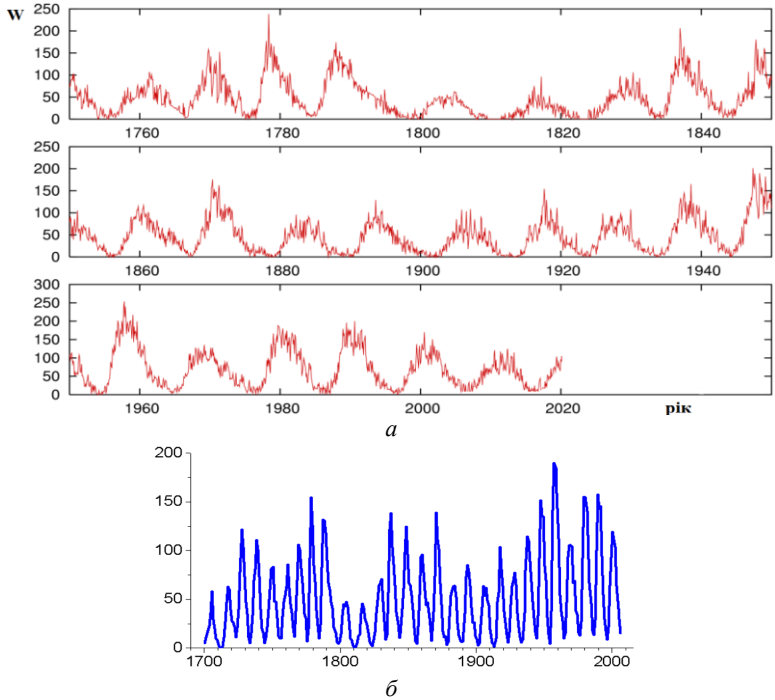


Рис. 1.4. Середньомісячні (а) і річні (б) кількості плям (число Вольфа) з 1750 року

«Цикл Швабе» (або «цикл Швабе–Вольфа») є найбільш помітно вираженим циклом сонячної активності. Відповідно, твердження про наявність 11-річної циклічності в сонячної активності іноді називають «законом Швабе–Вольфа». Цей цикл характеризується досить швидким (в середньому ~4 р.) збільшенням числа сонячних плям, а також іншими проявами сонячної активності, а потім

наступним більш повільним (~7 р.), його зменшенням. Під час циклу спостерігаються й інші періодичні зміни, наприклад – поступове зрушення зони освіти сонячних плям до екватора («закон Шпьорера»). Для пояснення подібної періодичності у виникненні плям зазвичай використовується теорія сонячного динамо.

Різноманітні «великі дані» сонячної активності доступні з відкритих джерел у форматах info (описовий), plot (графічний), txt, cvs, xls із запізненням всього ~15 хв (рис.1.5).

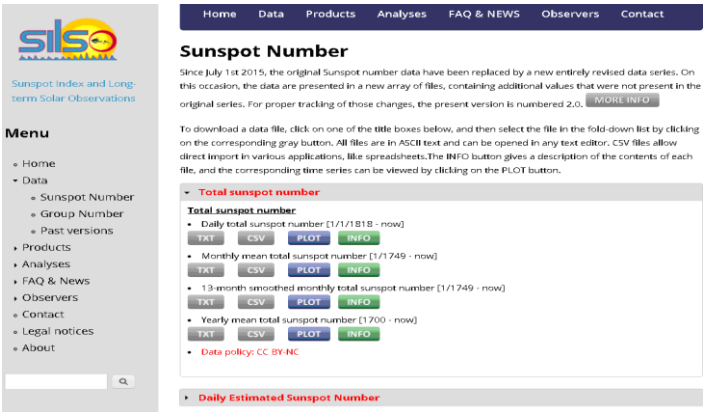


Рис. 1.5. Відкриті дані в різних форматах

Готові сервіси обробки цих даних для подальшого аналізу і використання розроблені різними обсерваторіями, інститутами і групами[3], наприклад, Velociraptor – це алгоритм, який аналізує сонячні УФ зображення, записані Екстремальним ультрафіолетовим візуальним телескопом (EIT) на борту Сонячної та Геліосферної обсерваторії (SoHO), а також дослідником сонячної корони (TRACE). Він одночасно оцінює видимий вектор руху та зміну яскравості від двох послідовних зображень (рис.1.6). Алгоритм заснований на багатомасштабному алгоритмі оптичного потоку, отриманому з локальної методики на основі матриці градієнта яскравості.

Велоцираптор може бути використаний для вивчення коронального диференціального обертання та для ідентифікації корональних подій як областей, що демонструють значну зміну яскравості або незвичне поле швидкості. Коло потенційних інтересів включає спалахи і викиди корональної маси (CME), корональну сейсмологію, динаміку протуберанців, магнітогідродинамічні хвилі дослідження MHD – попередників космічних погодних подій (сонячних бур).

[3] CACTUS; http://solardemon.oma.be/ , https://umbra.nascom.nasa.gov/eit/ , http://www.lmsal.com/TRACE/ , http://sidc.oma.be/velociraptor/ , http://sdoatsidc.oma.be/web/sdoatsidc/SoftwareSPoCA/ , SoFAST service STAFF viewer http://www.staff.oma.be/default.jsp

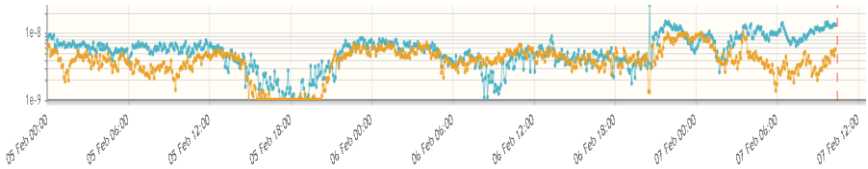


Рис. 1.6. Приклад запису двох корональних параметрів 5 лютого 2020 року

Таким чином, на прикладі даних реєстрації активності Сонця зрозуміло, що первинна ***обробка даних вимірювань*** не обмежується тільки ***статистичним аналізом***, але включає також ***складні математичні підходи й алгоритми***, які потребують глибоких знань математики.

1.1.2. Динаміка пандемії Covid–19

Показовим прикладом важливості підходів до обробки й аналізу «великих даних» є пандемія covid–19, яка офіційно розпочалася з лютого 2019 р. і протягом року охопила >200 країн і територій, вразила >175 млн людей, у тому числі >3.5 млн померло. Відповідні статистичні дані доступні у численних локальних та всесвітніх відкритих джерелах[4]. Спочатку вони включали число нових щоденних значень числа інфікованих і померлих I_{day} і D_{day} і їх загальних (кумулятивних) значень на поточний день I_{tot} і D_{tot}. Потім до цих даних були долучені щоденні і кумулятивні значення числа діагнозованих (викритих) $M_{day,tot}$, активних $A_{day,tot}$, критичних (під киснем) $C_{day,tot}$, які одужали $R_{day,tot}$, протестованих $T_{day,tot}$, які захворіли повторно $W_{day,tot}$ і вакцинованих $V_{day,tot}$ пацієнтів. Таким чином, за всіма загальними ознаками (4V) дані динаміки covid–19 можна зарахувати до «великих даних».

Статистичний аналіз розподілів всіх параметрів у різних країнах дозволив виявити сходні і відмінні риси (рис.1.7). Метою аналізу є отримання на базі знайдених статистичних залежностей найбільш адекватні математичні моделі, які відповідають знайденій динаміці, і на їх основі прогнозування можливих сценаріїв розвитку епідемії в конкретній країні або регіоні. Це дозволяє знайти «вузькі» місця та запланувати найбільш ефективні міри боротьби, такі як необхідного ступеня жорсткості карантину, підвищення чисельності койко-місць, кисневих апаратів і т. д.

Багаторічні попередні спостереження за динамікою сезонних вірусних і бактеріальних інфекційних захворювань привели до розвитку дисципліни «математична епідеміологія», який базується виключно на математичному моделюванні динаміки епідемій. Ще в 1760–1766 рр. Д. Бернуллі вивчав можливості збільшення середньої тривалості життя за рахунок усунення віспи,

[4] https://ourworldindata.org/coronavirus-source-data https://www.worldometers.info/coronavirus/

яка в ті часи була однією з головних причин смерті[5]. В 1870-х рр. Р. Кох та Л. Пастер показали, що інфекційні хвороби передаються через мікроорганізми, і була доведена важливість вакцінування. На початку XIX ст. Дж. Сноу[6] припустив, що епідемії спадають, коли зменшується доступність сприйнятливих осіб (susceptible, S(t)) – «епідемічного палива». Пізніше В. Хамер[7] проаналізував дані про захворюваність на кір у Великій Британії і виявив 2-річний період спалахів. Лікар Р. Росс застосував теорію ймовірностей для зрозуміння передачі інфекції[8], а епідеміолог А. Маккендрік[9] використав підходи статистичної фізики і кінетики хімічних реакцій для опису передачі інфекції як результату соціальних контактів між інфікованими I(t) і сприйнятливими S(t) людьми, що привело до перетворення математичної епідеміології в наукову дисципліну, в якій популяція розглядається як сукупність «частинок» різних типів, які випадково рухаються і зустрічаються, тому кожне «зіткнення» між частинками з груп S і I приводить до переходу частинки S-типу у I-тип з деякою ймовірністю.

Історично перша логістична математична модель записувалась як звичайне диференціальне рівняння (ЗДР)

$$I' = kI\left(1-I\right), \quad \mathrm{I}(0) = \mathrm{I}_0, \tag{1.5}$$

розв'язок якого $I(t) = I_0 e^{kt} / (1 + I_0(e^{kt} - 1))$ описує сигмоподібну криву з асимптотою $\lim_{t\to\infty} I(t) = 1$, що відповідає поширенню інфекції на всю популяцію у разі відсутності імунітету (перший принцип теоретичної епідеміології). Потім М. Кермаком[10] була доведена порогова теорема для інфекцій, які надають постійний імунітет, були досліджені умови ендемічності, вивчені різні епідеміологічні моделі, що вплинуло на експоненціальний розвиток математичної епідеміології.

Аналіз даних і математичне моделювання дозволило зрозуміти, що люди не є «частинками» в моделі, що привело до сучасних теорій «соціальних потоків». Було показано, що базова репродуктивна швидкість (basic reproduction number, BRN) інфекції $\Re_0$, яка вираховується як очікувана кількість випадків вторинного зараження від одного захворілого, є важливим індексом локальної

---

[5] Bernoulli D. Essai d'une nouvelle analyse de la mortalité cause par la petite vérole et des avantages de l'inoculation pour la prévenir. Histoire de l'académie royale des sciences avec les mémoires de mathématique et de physique tirés des registres de cette académie. Paris, 1766.

[6] Bacaër N. *A Short History of Mathematical Population Dynamics*. Springer-Verlag, London, 2011. 162 p.

[7] Hamer W. H. The Milroy Lectures on Epidemic Disease in England: the Evidence of Variability and of Persistency of Type, Bedford Press, 1906.

[8] Ross R. An application of the theory of probabilities to the study of a priori pathometry. Part I, Proc. R. Soc. A, Contain. Pap. Math. Phys. Charact. 1916. V. 92(638). P. 204-230.

[9] McKendrick A. The rise and fall of epidemics, Paludism. 1912. V. 1. P. 54-66.

[10] Kermark M., Mckendrick A. Contributions to the mathematical theory of epidemics. Part I, Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci. 1927. V. 115(5). P. 700-721.

епідемічної загрози. В рамках математичного моделювання $\Re_0$ – це фактор зростання числа інфікованих в групі

$$I(S) \sim \Re_0^{-1} \log(S / S_0), \quad S_0 = S(0). \tag{1.6}$$

Зараз найбільш популярна SIR-модель має вигляд системи (ЗДР)

$$\begin{aligned} S' &= \lambda - \beta SI - \mu S + \delta R, \\ I' &= \beta SI - \gamma I - (\mu + \phi) I, \\ R' &= \gamma I - (\mu + \delta) R, \end{aligned} \tag{1.7}$$

де $\bullet'$ – похідна за часом; λ – швидкість зростання популяції; β – швидкість інфікування; ϕ – смертність від хвороби; μ – смертність від інших причин; γ і δ – швидкості одужання та ослаблення імунітету.

Більш складні моделі враховують можливість повторного інфікування, доступність і рівень якості лікування та ін. Наприклад, SEIRS-модель записується як

$$\begin{aligned} \mathrm{S}' &= \lambda - \beta \mathrm{SI} - \mu \mathrm{S} + \delta \mathrm{R}, & \mathrm{I}' &= \alpha \mathrm{E} - \left(\gamma + \frac{\varepsilon}{1 + \mathrm{bI}} \right) \mathrm{I} - (\mu + \phi) \mathrm{I}, \\ \mathrm{E}' &= \beta \mathrm{SI} - (\alpha + \mu) \mathrm{E}, & \mathrm{R}' &= \left(\phi + \frac{\varepsilon}{1 + \mathrm{bI}} \right) \mathrm{I} - (\mu + \delta) \mathrm{R}, \end{aligned} \tag{1.8}$$

де E – контактні (exposed) особи, ε – ефективність лікування в лікарні, b – коефіцієнт насичування лікарень хворими, α залежить від інкубаційного періоду вірусу.

Далі розшукуються розв'язки (1.7), (1.8): стаціонарний f_0, $f = \{S, E, I, \mathrm{R}\}$ та у вигляді малих збурень $f(t) = f_0 + \tilde{f} \cdot e^{\lambda t}$, $\tilde{f} << f_0$ – амплітуда. Якщо $\mathrm{Re}(\lambda) \equiv \Re_0 < 0$, стаціонарний розв'язок системи стійкий і епідемія спадає з часом. Критерій стійкості системи $\Re_0 < 0$ дозволяє проаналізувати[11] динаміку за різних значень параметрів, які визначаються для моделі із результатів статистичної обробки «великих даних» (рис. 1.7). Методами якісної теорії диференціальних рівнянь можна побудувати фазові портрети (1.7), (1.8) за різних значень параметрів і запропонувати найбільш ефективні заходи для боротьби з епідемією.

---

[11] Захарова А. А., Кізілова Н. М. Дослідження кореляцій динаміки захворювання на COVID-19 з деякими соціально-економічними факторами // Вісник Харківського національного університету імені В. Н. Каразіна, Сер. Мат. модел. Інф. технол. Авт. сист. управл. 2020. № 47. С. 49-56. Костецька В. В., Кізілова Н. М. Математичне моделювання динаміки пандемії COVID-19 // там же. 2020. № 48. С. 65-71.

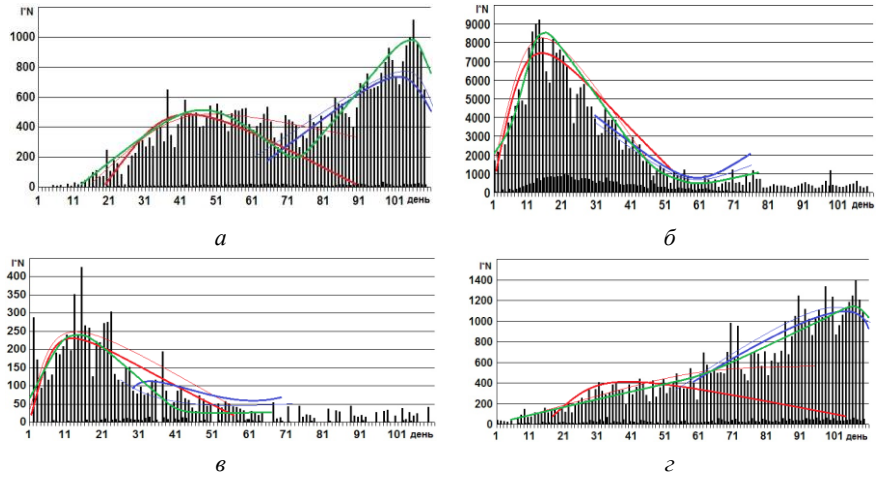


Рис. 1.7. Часові ряди I_{day} (світлий кольор) і D_{day} (темний кольор), і їх динаміка за різними моделями: Україна (а), Іспанія (б), Норвегія (в) і Індонезія (г)

1.2. Основні принципи роботи з великими даними

1. ***Горизонтальна масштабованість***. Об'єм даних із кожним днем стає все більше, і тому треба постійно збільшувати кількість обчислювальних вузлів, за якими розподіляються ці дані, причому обробка має відбуватись без погіршення продуктивності;
2. ***Відмовостійкість.*** Оскільки обчислювальних вузлів у кластері стає все більше (десятки тисяч), зростає ймовірність виходу машин із ладу. Методи роботи з великими даними мають враховувати ймовірність таких ситуацій і передбачати превентивні заходи;
3. ***Локальність даних***. Оскільки дані розподілені по великій кількості обчислювальних вузлів, то, якщо вони фізично знаходяться на одному сервері, а обробляються на іншому, витрати на передачу даних можуть бути невиправдано великими. Тому обробку даних бажано проводити на тій же машині, на якій вони зберігаються.

Означені принципи відрізняються від загальноприйнятих для традиційних, централізованих, вертикальних моделей зберігання добре структурованих даних.

Технології роботи

1. ***MapReduce*** – модель розподілених обчислювань у комп'ютерних кластерах, представлена компанією Google. Згідно з моделлю задача розділяється на значну кількість еквівалентних простіших завдань, які виконуються на вузлах кластера, а по закінченню виконання зводяться у кінцевий результат.

2. ***NoSQL*** (від англ. Not Only SQL) – загальний термін для різних нереляційних баз даних і сховищ, який не означає якусь конкретну технологію чи продукт. Звичайні реляційні бази даних добре підходять для обробки досить швидких і однотипних запитів, а на складних і гнучко побудованих задачах, які характерні для великих даних, навантаження перевищує розумні межі і використання СУБД стає неефективним.
3. ***Hadoop*** – freeware-утиліти, бібліотеки і фреймворки для розробки і виконання розподілених програм, які працюють на кластерах із 102–104 вузлів.
4.***R*** – мова програмування для статистичної обробки даних і роботи з графікою; стала стандартом для статистичних програм.
5. ***Апаратні рішення***. Великі корпорації (Teradata, McKinsey, EMC та ін.) розробляють апаратно-програмні комплекси, які призначені для обробки великих даних. Такі комплекси являють собою телекомунікаційні шафи, які містять кластер серверів і програмне забезпечення для паралельної обробки запитів, а також (можливо) апаратні рішення для аналітичної обробки в оперативній пам'яті (наприклад, апаратно-програмні комплекси Hana компанії SAP і комплекс Exalytics компанії Oracle, Business Intelligence компанії McKinsey), незважаючи на те, що така обробка початково не є масово-паралельною, а об'єми оперативної пам'яті одного вузла обмежуються кількома терабайтами.

Методи аналізу «великих даних»

1. Data Mining (видобуток даних, інтелектуальний аналіз даних, глибинний аналіз данних) – це сукупність методів виявлення у даних раніше невідомих, нетривіальних, практично корисних знань, які необхідні для прийняття рішень, наприклад: навчання асоціативних правил (association rule learning), класифікація (подів на категорії), кластерний аналіз, регресійний аналіз, виявлення й аналіз відхилень та ін.).
2. ***Статистичний аналіз*** – аналіз часових рядів, A/B-тестування (A/B testing, split testing) – метод маркетингового дослідження; під час його використання контрольна група елементів порівнюється із набором тестових груп, у яких один чи кілька показників були змінені, щоб з'ясувати, які зі змін покращують цільовий показник.
3. ***Візуалізація аналітичних даних*** – подання інформації у вигляді малюнків, діаграм, з використанням інтерактивних можливостей і анімації як для отримання результатів, так і для використання у якості вихідних даних для подальшого аналізу. Дуже важливий етап аналізу великих даних, що дозволяє показати найважливіші результати аналізу у найбільш зручному для сприйняття вигляді.
4. ***Просторовий аналіз (spatial analysis)*** – клас методів, що використовують топологічну, геометричну і географічну інформацію, яка вилучається із даних.

5. ***Когнітивний аналіз даних*** – аналіз з використанням когнітивістики (когнітивна наука, від лат. cognitio – пізнання) – міждисциплінарний науковий напрям, який поєднує теорію пізнання, когнітивну психологію, нейрофізіологію, когнітивну лінгвістику і теорію штучного інтелекту. У когнітивістиці використовуються комп'ютерні моделі, взяті з теорії штучного інтелекту, а також експериментальні методи, взяті з психології і фізіології вищої нервової діяльності. Ключовими технічними досягненнями, які зробили когнітивістику можливою, стали нові методи сканування мозку (томографія, електроенцефалографія та ін.) та розробка все більш потужних комп'ютерів.

6. ***Змішання та інтеграція даних (data fusion and integration)*** – набір технік, які дозволяють інтегрувати різнорідні дані з різних джерел з метою проведення глибинного аналізу (наприклад, цифрова обробка сигналів, обробка природньої мови, включно з тональним аналізом).

7. ***Машинне навчання*** (з учителем або без учителя) – використання моделей, що побудовані на базі статистичного аналізу чи методів розпізнавання для побудови комплексних прогнозів на основі моделей процесів/явищ.

8. ***Нейронні мережі (штучний інтелект)*** - евристичні алгоритми пошуку, які використовуються для розв'язання задач оптимізації і моделювання шляхом випадкового підбору, комбінування і варіації потрібних параметрів із використанням механізмів, аналогічних до натурального відбору у природі (генетичні алгоритми, swarm intelligence, ants' logic).

9. ***Розпізнавання образів*** – розділ кібернетики, що розробляє теоретичні основи та методи класифікації й ідентифікації предметів, явищ, процесів, сигналів, ситуацій та ін. об'єктів, які характеризуються скінченим набором властивостей і ознак. Протягом біологічної еволюції утворилися різні механізми розпізнавання образів за допомогою зорової, слухової, тактильної, електричної та ін. систем. Створення штучних систем розпізнавання образів залишається складною теоретичною й технічною проблемою. Необхідність у такому розпізнаванні виникає в різних галузях – від військової справи й систем безпеки до оцифровування різних аналогових сигналів. Традиційно задачі розпізнавання образів включають у коло задач штучного інтелекту.

10. ***Прогнозна аналітика (predictive analytics)*** – клас методів аналізу даних для прогнозування майбутньої поведінки об'єктів і суб'єктів із метою прийняття оптимальних рішень на основі статистичних методів, інтелектуального аналізу даних, теорії ігор та ін. Використовується в актуарних розрахунках, фінансових послугах, страхуванні, телекомунікації, торгівлі, туризмі, охороні здоров'я, фармацевтиці та ін.

11. ***Імітаційне моделювання (simulation)*** – метод, що дозволяє будувати моделі, які описують процеси так, як вони би проходили у дійсності. Імітаційне моделювання можна розглядати як різновид експериментальних випробувань.

12. ***Краудсорсинг*** – класифікація і збагачення даних силами широкого, неозначеного кола особистостей, які виконують цю роботу без вступу у трудові стосунки.

Моделі даних. У класичній теорії баз даних, модель даних є формальна теорія подання та обробки даних у системі управління базами даних (СУБД), яка включає як мінімум три аспекти:
– аспект ***структури***: методи опису типів і логічних структур даних у БД;
– аспект ***маніпуляції***: методи маніпулювання даними;
– аспект ***цілісності***: методи опису і підтримки цілісності БД.

1.3. Види моделей даних

1. ***Логічні моделі*** (ієрархічна, мережева, реляційна, об'єктно-орієнтована, документна, зоряна, моделі «сутність – зв'язок», «сутність – атрибут – значення», сніжинки).
2. ***Фізичні моделі*** (плоска, таблична, інвертована).
3. ***Інші моделі*** (асоціативна, кореляційна, семантична, модель XML, MultiValue, павутинна, іменовані графи, склад трійок).

1.3.1. Ієрархічна модель

Ієрархічна структура (модель БД) будується у вигляді ієрархічної деревоподібної структури, у якій для кожного головного об'єкта існує кілька підлеглих, а для кожного підлеглого об'єкта може бути тільки один головний. На найвищому рівні ієрархії перебуває кореневий об'єкт. Прикладом ієрархічної структури даних є структура державної влади, керівництва підприємства, школи. Ієрархічною БД є файлова система, що складається з кореневого каталогу, в якому є ієрархія підкаталогів і файлів.

Концептуальна схема ієрархічної моделі являє собою сукупність типів записів, пов'язаних типами зв'язків одним чи кількома деревами. Усі типи зв'язків цієї моделі належать до виду «один до декількох» і зображуються у вигляді стрілок.

Таким чином, взаємозв'язки між об'єктами нагадують взаємозв'язки в генеалогічному дереві за єдиним винятком: для кожного породженого (підлеглого) типу об'єкта може бути тільки один вхідний (головний) тип об'єкта. Тобто ієрархічна модель даних допускає тільки два типи зв'язків між об'єктами: «один до одного» і «один до декількох». Ієрархічні бази даних є навігаційними, тобто доступ можливий тільки за допомогою заздалегідь визначених зв'язків.

Під час моделювання подій, як правило, необхідні зв'язки типу «багато до декількох». Як одне з можливих рішень зняття цього обмеження можна запропонувати дублювання об'єктів. Однак дублювання об'єктів створює можливості неузгодженості даних.

Наприклад, якщо ієрархічна БД містить інформацію про покупців і їх замовлення, то буде існувати об'єкт «покупець» і об'єкт «замовлення». Об'єкт «покупець» матиме покажчики від кожного замовника до фізичного

розташування замовлень покупця в об'єкт «замовлення». У цій моделі запит, спрямований вниз по ієрархії (наприклад, які замовлення належать цьому покупцеві), має просту відповідь, але запит, спрямований вгору по ієрархії, більш складний (наприклад, який покупець помістив це замовлення).

Достоїнство ієрархічної бази даних полягає в тому, що її навігаційна природа забезпечує швидкий доступ під час проходження вздовж заздалегідь визначених зв'язків. Однак негнучкість моделі даних і, зокрема, неможливість наявності в об'єкта декількох батьків, а також відсутність прямого доступу до даних роблять її непридатною в умовах частого виконання запитів, не запланованих заздалегідь. Ще одним недоліком ієрархічної моделі даних є те, що інформаційний пошук із нижніх рівнів ієрархії не можна спрямувати по розміщених вище вузлах.

БД з ієрархічною моделлю є одними з найстаріших і стали першими СУБД для мейнфреймів. Розроблялися в 1950–1960, наприклад, Information Management System (IMS) фірми IBM.

1.3.2. Мережна модель

У мережевій моделі кожний об'єкт може одночасно виступати як у ролі головного, так і підлеглого елемента. Це означає, що кожний об'єкт може брати участь у довільній кількості зв'язків. Зв'язок у цьому випадку може встановлюватися явно, коли значення деяких полів є посилання на дані, що містяться в іншому файлі. Приклад: структура автобусних маршрутів, де із будь-якого населеного пункту існують маршрути до інших. Перша мережева модель CODASYL (1960-ті рр.) та одночасно незалежне розроблення ієрархічної БД фірмою North American Rockwell були пізніше взяті за основу IMS – власної розробки IBM.

Подібно до ієрархічної, мережну модель також можна подати у вигляді орієнтованого графа, але такий граф може містити цикли, а будь-яка його вершина може мати кілька батьківських вершин. Мережева структура набагато гнучкіша і виразніша від ієрархічної і придатна для моделювання більш широкого класу задач.

Ієрархічні і мережеві БД називають ***БД з навігацією***. Ця назва відповідає технології доступу до даних, яка була використана під час написання програми обробки даних. При цьому доступ до даних по шляхах, не передбачених під час створення БД, може потребувати нерозумно тривалого часу.

Підвищуючи ефективність доступу до даних і скорочуючи таким чином час відповіді на запит, ***принцип навігації*** разом з цим підвищує і ступінь залежності програм і даних. Програми обробки даних виявляються жорстко прив'язаними до поточного стану структури БД і повинні бути переписані у разі її змін. Операції модифікації і видалення даних вимагають переустановлення покажчиків, а маніпулювання даними залишається записоорієнтованим. Крім того, принцип навігації не дозволяє істотно підвищувати рівень мови

маніпулювання даними, щоб зробити його доступним користувачу-непрограмісту чи навіть програмісту-непрофесіоналу. Для пошуку запису-мети в ієрархічній або мережевій структурі програміст повинен спочатку визначити шлях доступу, а потім – крок за кроком переглянути всі записи, що трапляються на цьому шляху.

Наскільки складними є схеми представлення ієрархічних і мережевих БД, настільки і трудомістким є проектування конкретних прикладних систем на їхній основі. Як показує досвід, тривалі терміни розроблення прикладних систем нерідко призводять до того, що вони постійно перебувають на стадії розроблення і доопрацювання. Складність практичної реалізації БД на основі ієрархічної і мережевої моделей визначила створення реляційної моделі даних.

1.3.3. Реляційна модель

У реляційній (relational) моделі дані й взаємозв'язки між ними подаються за допомогою прямокутних таблиць. Рядки в реляційній БД називають записами, а стовпці – полями (наприклад, бібліографічний опис). Наукове обґрунтування основ реляційної моделі було здійснено Едгаром Ф. Коддом в 70-х рр. XX ст. як більш зручний засіб збереження, вибірки й маніпулювання даними, ніж ієрархічні й мережеві БД. Спочатку ця модель зацікавила лише наукові кола. Уперше її було використано у БД Ingres (Берклі) та System R (IBM), що були лише дослідними прототипами, анонсованими протягом 1976 року. Модель двовимірної таблиці дозволяє звертатися до даних як по рядках, так і по стовпцях, що є значною перевагою.

Назва «реляційна» пов'язана з тим, що кожен запис у таблиці даних містить інформацію, яка стосується (related) якогось конкретного об'єкта. Крім того, зв'язані між собою (тобто такі, що знаходяться в певних відношеннях – relations) дані навіть різних типів у моделі можуть розглядатися як одне ціле.

Таблиця має наступні властивості:
– кожний елемент таблиці являє собою один елемент даних;
– повторювані групи відсутні;
– усі стовпці в таблиці однорідні; це означає, що елементи стовпця мають однакову природу;
– стовпцям присвоєні унікальні імена;
– у таблиці немає двох однакових рядків.

Порядок розміщення рядків і стовпців у таблиці довільний; таблиця такого типу називається відношенням. У сучасній практиці для рядка використовується термін «запис», а для стовпця термін «поле».

Основною відмінністю пошуку даних в ієрархічних, мережевих і реляційних БД є те, що ієрархічні і мережеві моделі даних здійснюють зв'язок і пошук між різними об'єктами за структурою, а реляційні – за значенням ключових атрибутів (наприклад, можна знайти всі записи, значення яких у полі «рік видання» дорівнює 2000, але не можна знайти 2000-й рядок).

Оскільки реляційна структура концептуально проста, вона дозволяє реалізовувати невеликі і прості (і тому легкі для створення) БД, навіть персональні, сама можливість реалізації яких ніколи навіть і не розглядалася в системах з ієрархічною чи мережевою моделлю. Недоліком реляційної моделі даних є надмірність по полях (для створення зв'язків між різними об'єктами бази даних). Практично всі існуючі на сьогодні комерційні БД і програмні продукти для їх створення використовують реляційну модель даних (MS Access).

1.4. Життєвий цикл даних – це послідовність етапів, яку конкретна порція даних проходить від початкового етапу створення або отримання до моменту архівації або видалення (рис. 1.8).

Створення даних (Data Generation / Data Capture).

На цьому етапі дані генеруються або захоплюються. Цей етап зазвичай ще ділять на три типи отримання даних:

1. Придбання даних (Data Acquisition) – отримання організацією даних, вже згенерованих поза підприємства.
2. Запис даних (Data Entry) – створення нових даних оператором або комп'ютером. Дані мають цінність для підприємства.

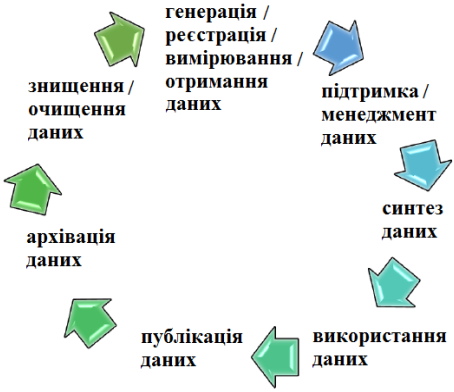


Рис. 1.8. Схема життєвого циклу даних

3. Реєстрація сигналів (Signal Reception) – це захоплення даних пристроями. Особливо важливо в системах управління, але останнім часом є особливо цінним під час використання такого підходу, як інтернет речей.

Обслуговування даних (Data Maintenance)

Після того як дані були створені, необхідно їх зберігати і обслуговувати. Необхідно здійснювати доставку даних у точку, де їх будуть використовувати або виробляти над ними маніпуляції (наприклад, операції синтезу). Можна говорити про те, що обслуговування даних – це обробка даних без отримання з них корисної інформації для замовника. Найчастіше обслуговування даних включає в себе такі дії з даними, як переміщення, інтеграція, очищення, збагачення та ETL-процеси (Extract, Transform, Load). Обслуговування даних зазвичай має на увазі застосування широкого спектра методів зі сфери управління даними (Data Management).

Синтез даних (Data Synthesis) – це процес отримання додаткової цінності з даних за допомогою використання індуктивної логіки і сторонніх інформаційних джерел – порівняно нова стадія в життєвому циклі даних. Він

використовується не у всіх моделях життєвих циклів даних. Це стадія, на якій із даними працюють аналітики, причому вони можуть використовувати в своїй роботі методи моделювання ризиків, актуарного моделювання, моделювання для прийняття інвестиційних рішень, штучного інтелекту, swarm intelligent, ants' logic, і т. д. На цій стадії використовується індуктивна логіка, яка вимагає використання експертної думки, тому що саме компетенції експертів необхідні для побудови різних математичних моделей (скорингу та ін.).

Використання даних (Data Usage)

На цій стадії дані застосовуються в якості корисної інформації для задач, які повинні виконуватися й управлятися на основі даних. Ці завдання можуть бути поза життєвого циклу даних. Проте дані стають все більш значущою частиною бізнес-процесів підприємств. Дані самі можуть бути продуктом або послугою (або бути частиною продукту або послуги), яка запропонована підприємством. Використання даних має спеціальні завдання в рамках управління даними (Data Governance). Одне із завдань полягає в законному використанні даних ц необхідному вигляді. Це називається «дозволене використання даних» (Permitted use of data). Можуть існувати регуляційні або договірні обмеження на те, як фактично можна використовувати дані, а частина ролі управління даними (Data Governance) полягає в забезпеченні дотримання цих обмежень.

Публікація даних (Data Publication)

Під час використання даних можлива ситуація, коли дані відправляються за межі підприємства. У цьому випадку говорять про публікацію даних. Публікація даних – це винесення даних за межі підприємства. Прикладом цього процесу може бути маклер, що розсилає щомісячні звіти клієнтам. Всі дані, які були розіслані, вже не можуть бути відкликані. Якщо були розіслані дані з неправильними значеннями, то такі дані не можуть бути виправлені, оскільки вони вже стають недоступні для підприємства.

Управління даними (Data Governance) може знадобитися, щоб допомогти прийняти рішення про те, як будуть оброблятися неправильні дані, які були відправлені з підприємства.

Архівація даних (Data Archivation)

Дані можуть бути використані як одноразово, так і кілька разів, але потім рано чи пізно життєвий цикл даних починає підходити до кінця.

Перша стадія цього стану полягає в архівації даних, тобто в копіюванні даних у пасивне середовище, в якосу вони зберігається, для тих випадків, коли вони знадобляться знову в активній (виробниче середовище), і видалення цих даних з усіх активних виробничих середовищ. Архів даних – це місце, де зберігаються дані без їх обслуговування, використання або публікації. У разі необхідності дані можуть бути відновлені з архіву.

Знищення даних (Data Purging) – це послідовність операцій для виконання незворотного видалення даних, що робить неможливим як відновлення даних, так і отримання залишкової інформації (Data Remanence) про них. Це одна з найбільш складно реалізованих процедур управління даними. Навіть з теоретичної точки зору існує команда запису значення в комірку пам'яті, але команди стирання значення як такої немає. Для знищення даних необхідно виготовити високопродуктивне джерело випадкових чисел і перезаписати ними весь носій інформації (перезапису області зберігання недостатньо, оскільки зберігається інформація про вихідний кількості даних). Інакше кажучи, під час знищення даних необхідно не тільки зробити недоступними від відновлення на фізичному рівні самі дані, але і пов'язану з ними інформацію в інших наборах даних. Знищення даних регламентується в Державному Стандарті країни.

2. ОСНОВНІ СТАТИСТИЧНІ МЕТОДИ АНАЛІЗУ «ВЕЛИКИХ ДАНИХ»

Методи статистичного аналізу даних включають дескриптивну статистику, параметричні, непараметричні і номінальні методи, а також відомі види аналізу:

– спектральний аналіз
– кореляційний аналіз;
– регресійний аналіз;
– дисперсійний аналіз;
– кластерний аналіз;
– дискримінантний аналіз;
– факторний аналіз;
– багатовимірний аналіз;
– критерії згоди;
– вейвлет–аналіз;
– фрактальний аналіз.

Процедура обробки й аналізу великих даних включає в себе

1) ***подання вихідних даних*** у програмах Excel і RStudio (вектори, масиви, матриці, списки, таблиці);
2) ***статистична обробка*** даних у програмах Excel і RStudio: підрахунок описових статистик, графічне представлення даних;
3) ***угруповання даних***, виявлення значущих кореляцій, залежностей і тенденцій внаслідок аналізу наявної інформації, виявлення відносин між даними різного типу;
4) застосування різних методів виділення, вилучення й угруповання даних, які дозволяють виявити ***систематизовані структури даних*** і вивести з них ***правила для прийняття рішень*** і ***прогнозування*** їх наслідків;
5) ***візуалізація*** вихідної інформації й аналітичних даних;
6) можливості ***графічного представлення*** інформації: графічні функції відображення одновимірних і багатовимірних даних, графічний висновок з використанням графічних параметрів, 3D– і 4D–графіка.

2.1. Описова статистика / дескриптивна статистика (англ. Descriptive statistics) займається обробкою емпіричних даних, їх систематизацією, наочним поданням у формі графіків і таблиць, а також їх кількісним описом за допомогою основних статистичних показників. Цей підхід протиставляється статистичному висновку в тому сенсі, що не формулює висновків про генеральну сукупність на підставі результатів дослідження окремих випадків. Статистичний висновок же передбачає, що властивості і закономірності, виявлені під час дослідження об’єктів вибірки, також притаманні генеральної сукупності.

Описова статистика використовує три основні ***методи агрегування даних***:
– ***табличне представлення***. Статистична таблиця – система рядків і стовпців, в якій у певній послідовності викладається статистична інформація про різні явища / процеси;
– ***графічне зображення***;
– розрахунок ***статистичних показників***

Методи середнього рівня дають усереднену характеристику сукупності об'єктів за певною ознакою:

– середнє значення;
– математичне очікування;
– дисперсія;
– стандартна помилка;
– стандартне відхилення;
– ексцес;
– асиметрія.
– мінімум;
– максимум
– медіана;
– мода;
– квантиль;
– довірчий інтервал;

Середнє значення функції визначається різними способами і найбільш поширені з них:

– середні Колмогорова $\mathrm{K}^{-1}\left(\frac{1}{n}\sum_{j=1}^{n}K(a_j)\right)$, де $K(a)$ – функція, у тому числі

– середнє арифметичне (x): $\frac{1}{n}\sum_{j=1}^{n}a_j$; зважене (kx): $\frac{1}{n}\sum_{j=1}^{n}k_j a_j$;

– середньоквадратичне (x^2) : $\sqrt{\frac{1}{n}\sum_{j=1}^{n}a_j^2}$;

– середнє гармонійне (x^{-1}) : $\left(\frac{1}{n}\sum_{j=1}^{n}a_j^{-1}\right)^{-1}$;

– середнє геометричне (log) : $\sqrt[n]{\prod_{j=1}^{n}a_j}$;

– середнє степеневе (x^p) : $\left(\frac{1}{n}\sum_{j=1}^{n}a_j^{p}\right)^{1/p}$;

– середнє зважене – $\mathrm{K}^{-1}\left(\frac{1}{n}\sum_{j=1}^{n}k_j K(a_j)\right)$, де $\sum_{j=1}^{n}k_j=1$, у тому числі зважені середнє арифметичне, геометричне, гармонійне, степеневе.

– середнє хронологічне – узагальнює значення ознаки для однієї і тієї ж одиниці або сукупності в цілому, що змінюються в часі.

Математичне очікування (сподівання) M(X) (Expected value, E(X)) – одне з найважливіших понять у теорії ймовірностей, що означає середнє (зважене) значення випадкової величини X.

Для дискретної випадкової величини X з розподіленням $P(X = x_j) = p_j$, де $\sum_{j=1}^{\infty} p_j = 1$ математичне очікування $M(X) = \sum_{j=1}^{\infty} p_j x_j$.

Якщо $\chi_X(x)$ – функція розподілення випадкової величини X, то її математичне очікування обчислюється як інтеграл Лебега-Стілтьєса $M(X) = \int_{-\infty}^{\infty} x d\chi_X(x)$. Якщо X – абсолютно безперервна випадкова величина, то замість попереднього інтегралу маємо $M(X) = \int_{-\infty}^{\infty} x\chi_X(x)dx$, тобто маємо зважування за щільністю розподілу.

Математичне очікування випадкового вектора дорівнює вектору, компоненти якого дорівнюють математичним очікуванням компонент випадкового вектора.

Дисперсія (variance) D(X) – це міра розсіяння значень величини відносно середнього значення M(X): $D(x) = M\left[(X - M(X))^2\right]$. Для дискретної та безперервної випадкових величин відповідно маємо визначення:

$$D(\mathrm{X}) = \sum_{j=1}^{n} p_j \left(x_j - M(X)\right)^2 \text{ і } \mathrm{D}(X) = \int_{-\infty}^{\infty} \left(x - M(X)\right)^2 \chi_X(x)dx. \qquad (2.1)$$

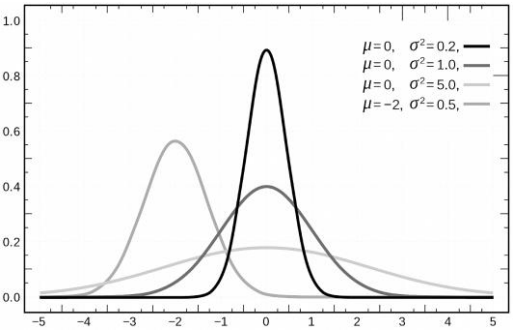


Рис. 2.1. Приклади нормальних розподілень з різними μ, σ

Більші значення дисперсії свідчать про більші відхилення значень випадкової величини від центру розподілу. Серед різних видів розподілень найбільш цікавим є випадок т. зв. «нормального» розподілення, або розподілення Гауса, яке має вигляд

$$\chi(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right), \qquad (2.2)$$

де $\mu = M(x)$, $\sigma = \sqrt{D(x)}$ – середньоквадратичне відхилення. Приклади розподілень (1.10) для різних значень μ, σ наведені на (рис.2.1).

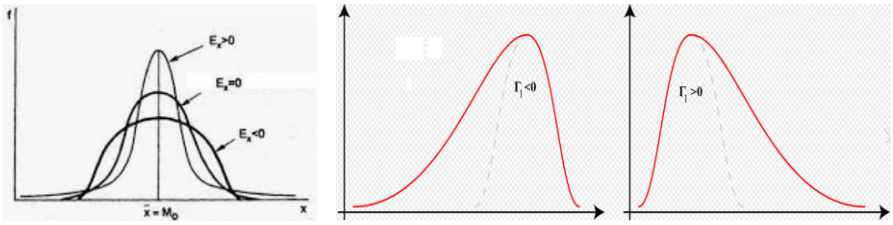
Стандартна похибка середнього в математичній статистиці – це статистичний параметр, величина, яка характеризує вибіркове розподілення, зокрема стандартне відхилення вибіркового середнього, розраховане за вибіркою розміру {n} із генеральної сукупності. Термін був вперше введений Удні Юлом у 1897 року. Величина стандартної помилки залежить від обсягу

вибірки n і дисперсії генеральної сукупності $D(s) = \sigma / \sqrt{n}$, яка може бути невідомою, і обчислюється за формулою $SE_x = s / \sqrt{n}$, де s – стандартне відхилення випадкової величини на основі незміщеної оцінки її вибіркової дисперсії.

Коефіцієнт ексцесу (Kurtosis) – це числова характеристика розподілу ймовірностей дійсної випадкової величини. Коефіцієнт ексцесу характеризує «крутість», тобто стрімкість підвищення кривої розподілу у порівнянні з нормальним розподіленням.

Ексцесом (за Фішером) теоретичного розподілу називають характеристику, яка обчислюється за формулою $\Gamma_2 = \mu_4 / \sigma^4 - 3$, де $\mu_4 = M\left[(X - M(X))^4\right]$ – 4-й центральний момент. За такого визначення (–3 в формулі) коефіцієнт ексцесу стандартного нормального розподілу дорівнює нулю; $\mu_4 > 0$, якщо пік розподілу близько до M(X) гострий, і $\mu_4 < 0$, якщо пік дуже гладкий (рис. 2.2а).

Коефіцієнт асиметрії Фішера (skewness) – це числова характеристика розподілу ймовірностей дійсної випадкової величини, яка обчислюється як $\Gamma_1 = \mu_3 / \sigma^3$ і вказує на міру нерівності крутизни двох «фронтів» кривої розподілення (рис. 2.2б.)



а *б*

Рис. 2.2. Розподілення з різними значеннями μ_4 (а) і Γ_1 (б)

Довірчий інтервал (confidence interval, CI) – це тип інтервальної оцінки, яка обчислюється за даними спостереження і покриває невідомий статистичний параметр із заданою надійністю. Це інтервал, у межах якого з заданою довірчою імовірністю можна чекати значення оцінюваної (шуканої) випадкової величини. Застосовується для повнішої оцінки точності порівняно з точковою оцінкою. Метод довірчих інтервалів розробив американський статистик Єжи Нейман, виходячи з ідей англійського статистика Рональда Фішера.

Наприклад, можна сказати: результати опитування показали, що кандидат набере на виборах 32 % голосів. Проте математично правильніше сказати: з імовірністю 87 % кількість голосів набраних кандидатом згідно з опитуваннями лежить в інтервалі 32 ± 4%, де ± 4% – це довірчий інтервал.

Байєсова статистика – це статистична теорія, яка ґрунтується на байєсівській інтерпретації ймовірності, а саме що ймовірність відображає ступінь

довіри події, яка може змінитися, якщо буде зібрана певна нова інформація. Цей підхід відрізняється від частотного підходу на ймовірність як фіксоване значення. Баєсівська ймовірність, або ступінь довіри, може базуватися на апріорних знаннях про подію, таких як результати попередніх спостережень.

Байєсівський дизайн експериментів включає в себе концепцію, яка називається «вплив апріорної впевненості». Цей підхід використовує техніки статистичного аналізу для включення результатів попередніх експериментів в планування наступного експерименту. Це досягається шляхом оновлення «довіри» через використання апріорного і апостеріорного розподілів. Це дозволяє під час планування експериментів використовувати ресурси всіх видів.

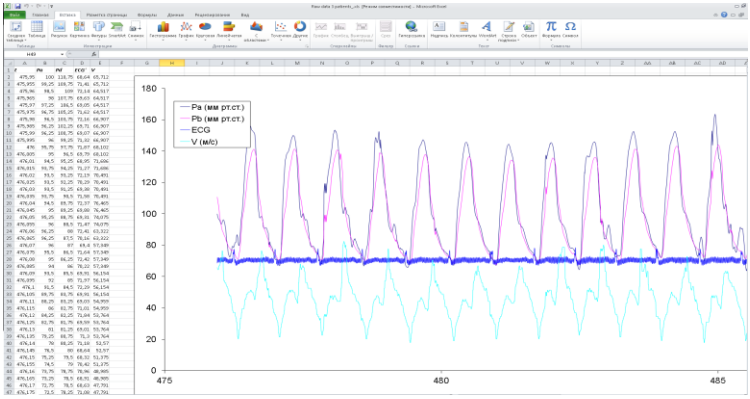
Перелічені методи статистичної обробки даних включені до всіх ***пакетів прикладних програм***, таких як:

– електронні таблиці Excel, Lotus 1-2-3, QuattroPro;

– математичні пакети загального призначення Mathcad, MatLab, SkyLab.

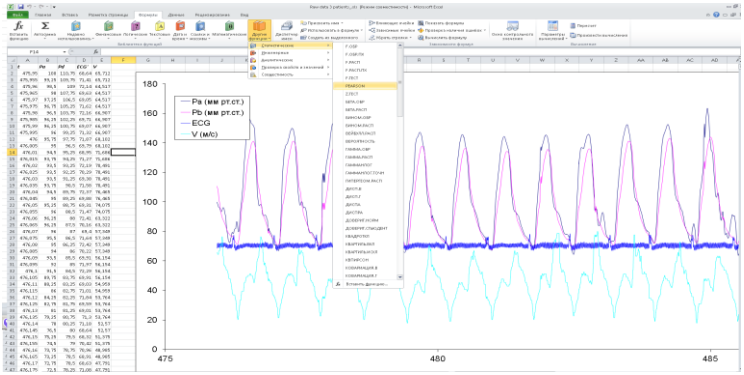
Набагато більші можливості мають професійні статистичні пакети (STATISTICA, SPSS, STADIA, STATGRAPHICS, RStudio), які використовуюсь найсучасніші методи математичної статистики для обробки даних і призначені для користувачів, добре знайомих із методами математичної статистики.

RStudio[12] (не R-Studio !) – вільне середовище розробки програмного забезпечення з відкритим вихідним кодом для мови програмування R, який розроблений для статистичної обробки даних, у тому числі для обробки графічної інформації (кривих, зображень). RStudio написана мовою програмування C ++ і використовує фреймворк Qt для графічного інтерфейсу користувача. Доступна в версіях RStudio Desktop, в якій програма виконується на локальній машині як звичайна програма; і RStudio Server, в якій надається доступ через браузер до RStudio, встановленої на віддаленому Linux-сервері. Дистрибутиви RStudio Desktop доступні для Linux, OS X і Windows.

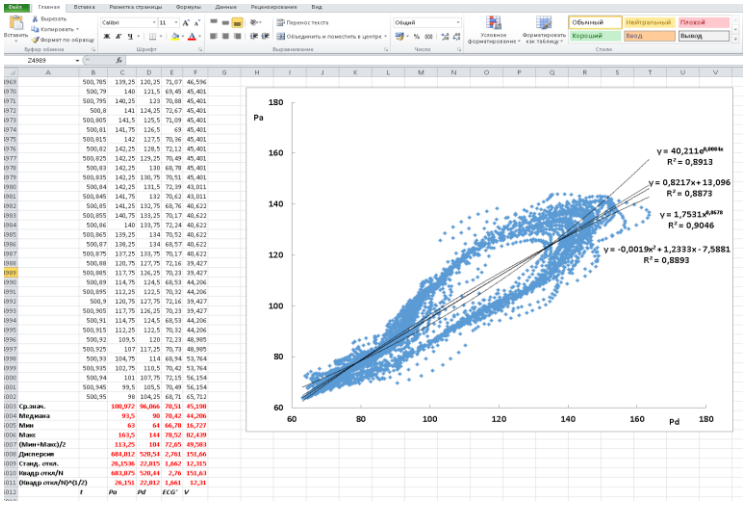


а

[12] https://rstudio.com/



б



в

Рис.2.3. Результати первинної статистичної обробки пульсових хвиль: вихідні дані (а), доступні математичні і статистичні функції (б), обчислення середніх, дисперсій, похибок і трендів (в)

Існують пакети програм, з якими можуть працювати користувачі без глибокої математичної підготовки, а також нестатистичні пакети, які вирішують завдання класифікації, такі як PolyAnalyst, DA-система, ARGONAVT, LOREG, OTEX і різноманітні нейромережеві пакети.

Приклад застосування табличного процесора MS Office Excel для найпростішої статистичної обробки результатів цілодобового вимірювання електрокардіограми (ЕКГ), артеріального тиску и швидкості руху крові крізь ділянку артерії (добовий моніторинг) наведений на рис. 2.3а,б,в.

2.2. Часові ряди та методи їх обробки

Часовий ряд – це послідовність спостережень, які впорядковані за часом у вигляді масиву точок або безперервної функції

$$\{X(t_j), j = 1, 2, ..., n\} \rightarrow X(t). \quad (2.3)$$

Мета аналізу часових рядів:

1) визначення природи ряду, дослідження структури, закономірностей;

2) прогнозування (передбачення майбутніх значень часового ряду за теперішнім і минулим значеннями).

Найпростіша модель часового ряду має вигляд

$$x(t) = f(t) + s(t) + c(t) + \varepsilon(t), \quad (2.4)$$

де $f(t)$ – це ***лінія тренду*** (систематична складова), яка будується шляхом математичної обробки статистичних даних з наближенням методом найменших квадратів;

$s(t)$ – це ***сезонна компонента***, яка відображає повторюваність зміни деяких процесів протягом певних періодів;

$c(t)$ – ***циклічні коливання*** щодо середнього значення, виявляються після віднімання тренду від сигналу;

$\varepsilon(t)$ – це ***випадкова компонента*** (шум, помилка), для усунення якої використовують різні способи фільтрації шуму, що дозволяють побачити регулярну складову більш чітко.

Таким чином, регулярні часові ряди мають тренд і сезонну компоненту. ***Тренд*** – це загальна систематична лінійна або нелінійна компонента, яка може змінюватися за часом. ***Сезонна складова*** – це компонента, яка періодично повторюється. Наприклад, продажі компанії можуть зростати з року в рік, але вони також містять сезонну складову (як правило, 25 % річних продажів припадає на грудень і тільки 4 % – на серпень).

Цю загальну модель можна зрозуміти на «класичному» ряді[13] – що представляє місячні міжнародні авіаперевезення (в тисячах) протягом 12 років з 1949 по 1960 рр. (рис. 2.4а). Графік місячних перевезень ясно показує майже лінійний тренд, тобто є стійке зростання перевезень з року в рік (приблизно в 4 рази більше пасажирів перевезено в 1960 р., ніж в 1949 р.). У той же час характер місячних перевезень повторюється, вони мають майже один і той же характер в кожному річному періоді (наприклад, перевезень більше в відпускні періоди, ніж в інші місяці). Цей приклад показує досить певний тип моделі часового ряду, в якій амплітуда сезонних змін збільшується разом з трендом. Такого роду моделі називаються моделями з мультипликативною сезонністю. Інший приклад сезонних коливань захворюваності і смертності від сезонного грипу[14] наведений на рис. 2.4б.

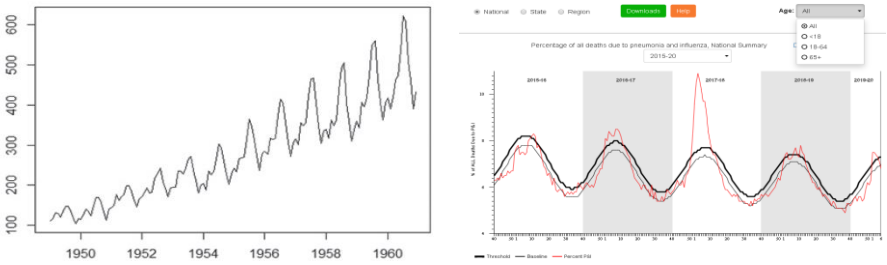
---

[13] Box and Jenkins, 1976.

[14] Відкрите джерело https://gis.cdc.gov/grasp/fluview/mortality.html

2.2.1. Аналіз тренду

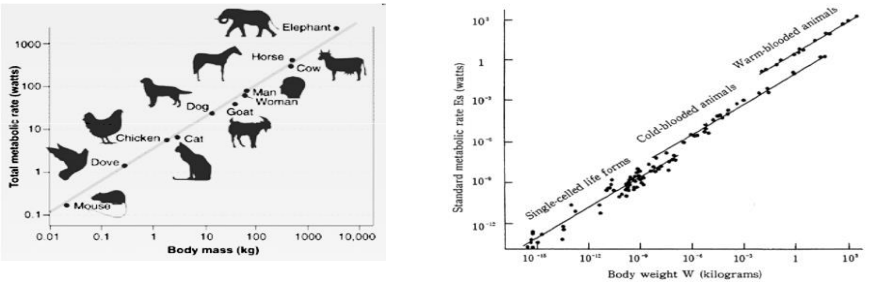
Не існує «автоматичного» способу виявлення тренду в часовому ряді. Якщо тренд є монотонним (стійко зростає або стійко убуває), то аналізувати такий ряд зазвичай неважко. Якщо часові ряди містять значну помилку, то першим кроком виділення тренду є згладжування. Багато монотонних часових рядів можна добре наблизити лінійною функцією. Якщо ж є явна монотонна нелінійна компонента, то дані спочатку слід перетворити, щоб усунути нелінійність. Зазвичай для цього використовують логарифмічне, експоненціальне або (рідше) поліноміальне перетворення даних.



а *б*

Рис. 2.4. Приклади великих даних:
міжнародні авіаперевезення (а); смертність від сезонного грипу (б)

Наочним прикладом аналізу ліній тренду є залежність між середньою масою тварини і метаболічною швидкістю (metabolic rate), – тобто кількістю енергії, яка виробляється в одиниці об'єму тіла в одиницю часу P(Дж/кг/с). Вперше ця залежність буда отримана Максом Кляйбером у 1932 р.[15] (рис.2.5а), а потім узагальнена на випадок всіх живих організмів, що існують (рис.2.5б).



а *б*

Рис. 2.5. Залежність P(M) для деяких теплокровних (а)
порівняно з одноклітинними організмами і холоднокровними (б)

[15] Kleiber M. Body size and metabilozm. Hilgardia. 1932;6:315-351.

Коефіцієнт детермінації ($R^2 \in [0;1]$) – це частка дисперсії залежної змінної, яка пояснюється розглянутою моделлю залежності. Це універсальна міра залежності однієї випадкової величини від інших, яка обчислюється за формулою

$$R^2 = 1 - \frac{D(y \mid x)}{D(y)} = 1 - \frac{\sigma^2}{\sigma_y^2}, \qquad (2.5)$$

де $D(y)$ – дисперсія випадкової величини, $D(y \mid x)$ – умовна (за факторами x) дисперсія залежної змінної (дисперсія помилки моделі).

У випадку лінійної залежності R^2 є квадратом так званого множинного коефіцієнта кореляції між залежною змінною і незалежними. Зокрема, для моделі парної лінійної регресії коефіцієнт детермінації дорівнює квадрату звичайного коефіцієнта кореляції (між *y* і *x*). Чим ближче значення R^2 до 1, тим сильніше залежність. Під час оцінки регресійних моделей це інтерпретується як відповідність моделі даних. Для прийнятних моделей передбачається, що коефіцієнт детермінації повинен бути хоча б не менше 50 %. Моделі з $R^2 > 80\%$ можна визнати досить хорошими (коефіцієнт кореляції перевищує 90%). Значення $R^2 = 1$ означає достовірну функціональну залежність між змінними.

2.2.2. Згладжування – це спосіб локального усереднення даних, за якого несистематичні компоненти взаємно погашають одна одну. Загальний метод згладжування використовує так зване «ковзаюче середнє», в якому кожен член ряду замінюється простим або зваженим середнім n сусідніх членів, де n – ширина «вікна» (рис. 2.6). Замість середнього можна використовувати медіану

A1A1	A1	A1
A2(A1+A2+A3)/3	(A1+A2+A3)/3	(A1+A2+A3)/3
A3(A2+A3+A4)/3	(A1+A2+A3+A4+A5)/5	(A1+A2+A3+A4+A5)/5
A4(A3+A4+A5)/3	(A2+A3+A4+A5+A6)/5	(A1+A2+A3+A4+A5+A6+A7)/7
…	…	…
(…)/3	(…)/5	(…)/7
(…)/3	(…)/5	(…)/5
(…)/3	(…)/3	(…)/3
AnAn	An	An

Рис. 2.6. Приклад формул методу «ковзаючого середнього»

значень, що потрапили у вікно. Основна перевага медіанного згладжування, порівняно зі згладжуванням ковзаючим середнім, полягає в тому, що результати стають більш стійкими до викидів всередині вікна. Таким чином, якщо в даних є викиди (пов'язані, наприклад, із помилками вимірювань), то згладжування медіаною зазвичай призводить до більш гладких або, принаймні,

більш «надійних» кривих, порівняно зі змінним середнім з тим же самим вікном. Основний недолік медіанного згладжування в тому, що за відсутності явних викидів він призводить до більш «зубчастих» кривих (чим згладжування ковзаючим середнім) і не дозволяє використовувати ваги.

Якщо помилка вимірювання велика, використовується метод найменших квадратів, зважених щодо відстані, метод негативного експоненціально зваженого згладжування, або метод бікубічних сплайнів.

Аналіз сезонності

Періодична сезонна залежність (сезонність) являє собою інший загальний тип компонент часового ряду. Це поняття було проілюстровано раніше на прикладі авіаперевезень пасажирів. В такому разі кожне спостереження дуже схоже на сусіднє і є повторювана сезонна складова (в тому самому місяці минулого року). Ця залежність може бути формально визначена як кореляція порядку *k* між кожним *i*-м елементом ряду *i* (*i*-*k*)-м елементом, яку можна виміряти за допомогою автокореляції (тобто кореляції між самими членами ряду); *k* називають лагом (запізнюванням). Якщо помилка вимірювання не дуже велика, то сезонність можна визначити візуально, розглядаючи поведінку членів ряду через кожні *k* тимчасових одиниць.

Сезонні складові часового ряду можуть бути знайдені за допомогою коррелограм (автокоррелограм), які показують чисельно і графічно коефіцієнти автокореляції і їх стандартні помилки для послідовності лагів із певного діапазону (наприклад, від 1 до 30). На корелограмі зазвичай відзначається діапазон в розмірі двох стандартних помилок на кожному лазі, проте зазвичай величина автокореляції цікавіша, ніж її надійність, тому що інтерес в основному представляють дуже сильні (а отже, високо значущі) автокореляції.

2.2.3. Кореляційний / регресійний аналіз

Регресійний аналіз – це набір статистичних методів дослідження впливу однієї або декількох незалежних змінних $(X_1, X_2, ..., X_k)$ на залежну змінну Y у вигляді

$$\mathrm{Y} = \mathrm{Y}(X_1, X_2, ..., X_k). \qquad (2.6)$$

Цей аналіз визначає ступінь детермінованості варіації залежної змінної предикторами (незалежними змінними), прогнозує значення залежної змінної за допомогою незалежної(-их), визначає внесок окремих незалежних змінних у варіацію залежної, і дозволяє знайти найбільш істотний впливовий фактор(и).

Найпростішим видом залежності (2.2) є лінійна регресія:

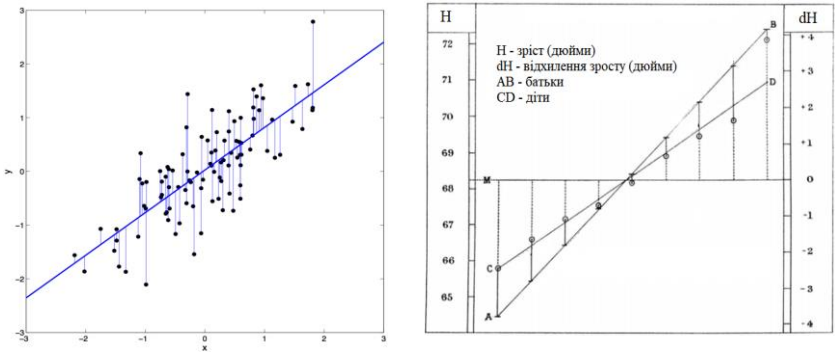
$$Y = a_0 + a_1 X_1 + a_2 X_2 + ... + a_n X_n, \qquad (2.7)$$

де коефіцієнти a_j знаходять методом найменших квадратів з умови мінімуму сумарної відстані точок $(X_1, X_2, ..., X_k)$ від кривої (рис. 2.7а) $Y=Y(X_1, X_2, ..., X_k)$:

$$\chi = \sum_{j=1}^{n} (Y_j - \tilde{Y}(j)) \to \min, \tag{2.8}$$

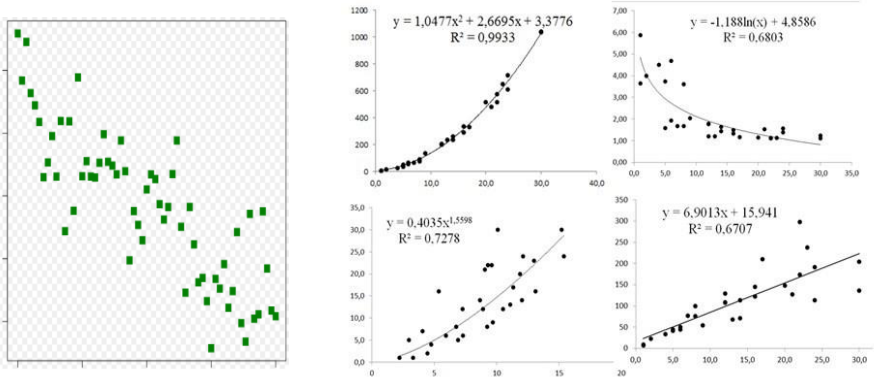
що вимагає розв'язання системи алгебраїчних рівнянь $\frac{\partial \chi(\bar{a})}{\partial a_j} = 0$, з яких знаходять коефіцієнти регресії.

Цей аналіз вперше ввів британський антрополог і статистик Ф. Гальтон[16]. Шляхом аналізу даних вимірювань він знайшов достовірні залежності між ростом батьків (X) та їх дітей (Y) у вигляді $Y \approx M(Y) + 2(X - M(X))/3$ (рис. 2.7б).



а *б*

Рис. 2.7. Пояснення методу найменших квадратів (а) і лінії тренду з роботи Ф. Гальтона (б)



а *б*

Рис. 2.8. Діаграма розсіювання (а) і приклади обчислення ліній тренду (б)

[16] Гальтон Ф. Регресія до середнього в спадковості росту. Лист до Британської асоціації сприяння розвитку науки в Абердіні. 1885

Для графічного представлення кореляційного зв'язку можна використовувати прямокутну систему координат з осями, які відповідають обом змінним. Кожна пара значень маркується за допомогою певного символу. Такий графік називається ***діаграмою розсіювання***. Пакети статистичних програм дозволяють автоматично обчислювати рівняння ліній тренду різних виглядів і відповідні значення коефіцієнту достовірності наближення R^2 (рис. 2.8).

Параметричні показники кореляції

Важливою характеристикою спільного розподілу двох випадкових величин є ***коваріація*** (або кореляційний момент). Коваріація є спільним центральним моментом другого порядку і визначається як математичне очікування добутку відхилень випадкових величин:

$$\begin{aligned}\operatorname{cov}(X,Y) &= M\left((X-M(X))\cdot(Y-M(Y))\right)= \\ &= M(XY)-M(X)\cdot M(Y)\le\sqrt{D(X)\cdot D(Y)}.\end{aligned} \tag{2.9}$$

Властивості коваріації:

1. Коваріація двох незалежних випадкових величин дорівнює нулю.
2. Абсолютна величина коваріації двох випадкових величин не перевищує середнього геометричного їх дисперсій:

$$\operatorname{cov}(X,Y)\le\sqrt{D(X)\cdot D(Y)}\,.$$

3. Коваріація має розмірність добутку розмірності двох випадкових величин, тобто залежить від одиниць їхнього вимірювання, що значно ускладнює її використання в цілях кореляційного аналізу.

Для усунення недоліку коваріації, який пов'язаний з її розмірністю, був введений лінійний коефіцієнт кореляції (або коефіцієнт кореляції Пірсона), який розробили Карл Пірсон, Френсіс Еджуорт і Рафаель Уелдон в 90-х роках XIX ст. ***Коефіцієнт кореляції Пірсона*** розраховується за формулою

$$K_P=\frac{\operatorname{cov}(X,Y)}{\sigma(X)\sigma(Y)}=\frac{\sum(X-M(X)(Y-M(Y))}{\sqrt{\sum(X-M(X))^2(\mathrm{Y}-M(\mathrm{Y}))^2}} \tag{2.10}$$

і змінюється в діапазоні від -1 до +1 .

Кореляційний аналіз – це метод обробки статистичних даних, за допомогою якого вимірюється ступінь зв'язку між двома або більше змінними. Він тісно пов'язаний із регресійним аналізом (також часто зустрічається термін «кореляційно-регресійний аналіз», який є більш загальним статистичним поняттям), з його допомогою визначають необхідність включення тих чи інших факторів у рівняння множинної регресії, а також оцінюють отримане рівняння регресії на відповідність виявленим зв'язкам використовуючи коефіцієнт детермінації. На рис. 2.9 наведені різноманітні приклади типів діаграм розсіювання з різними значеннями R^2.

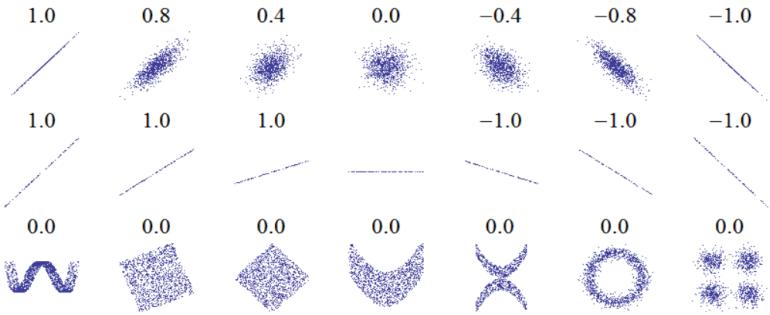


Рис.2.9. Види кореляційних полів і їх коефіцієнти R^2

Автокореляційна функція (АКФ) – це залежність взаємозв'язку між функцією (сигналом) і її зрушеною копією від величини тимчасового зсуву. Для детермінованих сигналів $f(t)$ АКФ визначається інтегралом:

$$\Psi(\tau) = \int_{-\infty}^{\infty} f(t) f^*(t-\tau) dt \qquad (2.11)$$

і показує зв'язок сигналу (функції) зі своєю копією, яка зміщена на величину τ , * означає комплексне сполучення.

Для випадкових процесів АКФ випадкової функції X (t) має вигляд

$$K(\tau) = E\left(X(t)X^*(t-\tau)\right). \qquad (2.12)$$

Якщо початкова функція строго періодична, то на графіку авто кореляційної функції теж буде строго періодична функція. Таким чином, з цього графіка можна судити про періодичність вхідної функції і про її частотні характеристики.

Взаємнокореляційна функція (cross-correlation) – це стандартний метод оцінки ступеня кореляції двох послідовностей. Вона часто використовується для пошуку в довгій послідовності коротшою заздалегідь відомою. Для двох рядів даних $f(t)$ і $g(t)$ взаємна кореляція визначається як

$$(f \times g)_n \equiv \sum_j f_j^* g_{j+n}, \qquad (2.13)$$

де *n* - зсув ряду даних *g(t)* відносно *f(t)*, еквівалентний τ в (2.12).

Для безперервних функцій *f (t)* і *g (t)* взаємна кореляція визначається як

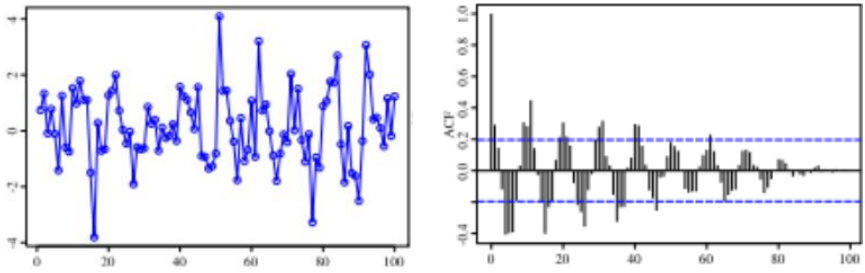
$$(f \times g)(t) \equiv \int_{-\infty}^{\infty} f^*(\tau) g(t+\tau) d\tau . \qquad (2.14)$$

Метод використовується підчас обробки сигналів, наприклад для розпізнавання відбитого від об'єкта локаційного сигналу (радарів, сонарів) в умовах перешкод. Також він використовується для аналізу випадкових процесів, наприклад у вимірюваннях і математичній статистиці. Автокореляційна функція широко застосовується для аналізу складних коливань, наприклад ЕКГ, ЕЕГ, МКГ, ФКГ людини та ін.

Дослідження корелограм

Корелограма (або графік автокореляції) – це графік залежності автокореляції вибірки від затримки (лагу): *(f×g)(n)* або *(f×g)(τ)* відповідно. Під час дослідженні корелограм слід пам'ятати, що автокореляції послідовних лагів формально залежні між собою.

Розглянемо наступний приклад (рис. 2.10а). Якщо перший член ряду тісно пов'язаний з другим, а другий з третім, то перший елемент повинен також якимось чином залежати від третього і т.д. (рис. 2.10б). Це приводить до того, що періодична залежність може істотно змінитися після видалення автокореляцій першого порядку, тобто після взяття різниці з лагом 1.



а *б*

Рис. 2.10. Ряд даних (а) і корелограма даних (б)

Корелограми – це ефективний спосіб перевірки хаотичності або навпаки – взаємозв'язку даних вимірювань. Наявність хаотичності перевіряється обчисленням автокореляцій значень даних зі змінними лагами. Якщо дані дійсно випадкові, то коефіцієнти автокореляцій близькі до нуля для будь-якого значення лагу. Якщо дані невипадкові (є прихована осцилююча залежність), то один або кілька коефіцієнтів автокореляції будуть значно відрізнятися від нуля. Якщо використовується крос-кореляція, то корелограми називають крос-корелограмами. Крім того, корелограми часто використовуються на стадії ідентифікації моделей (моделей авторегресії, ковзаючого середнього, в моделі Бокса–Дженкінса та ін.). Якщо аналітик не перевіряє вибірку на хаотичність, то законність багатьох його статистичних висновків ставиться під сумнів.

Вперше термін «кореляція» ввів у науковий обіг французький палеонтолог Жорж Кюв'є в XVIII столітті. Він розробив «закон кореляції» частин і органів живих істот, за допомогою якого можна відновити вигляд викопної

тварини, маючи в розпорядженні лише частину її останків. У статистиці слово «кореляція» першим став використовувати англійський біолог і статистик Френсіс Гальтон в кінці XIX століття.

Існують і інші коефіцієнти кореляції, подані нижче

Коефіцієнт рангової кореляції Кендалла

Нехай $(x_1, y_1),(x_2, y_2),...(x_n, y_n)$ – набір спостережень спільних випадкових величин *X* і *Y* відповідно тому всі значення x_k і y_k не є однаковими для будь-якого *k=1..n*. Будь-яка пара спостережень (x_i, y_i) і (x_j, y_j) називається ***узгодженою***, якщо узгоджені ряди для обох елементів: тобто, якщо $x_i > x_j$, $y_i > y_j$, або якщо $x_i < x_j$, $y_i < y_j$. Пара називається ***неузгодженою (або дисонуючою***), якщо $x_i > x_j$, $y_i < y_j$ та $x_i < x_j$, $y_i > y_j$. Якщо $x_i = x_j$ та $y_i = y_j$, то пара не є ні узгодженою, ні неузгодженою.

Тоді коефіцієнт кореляції Кендалла обчислюється як

$$K_K = \frac{s_1 - s_2}{n(n-1)/2}, \tag{2.15}$$

де s_1 – кількість узгоджених пар, s_2 – кількість неузгоджених пар.

Коефіцієнт кореляції Кендала часто використовується для статистичної оцінки в перевірці статистичних гіпотез для визначення чи можуть дві змінні розглядатись як статистично залежні. Цей тест є непараметричний, оскільки він не залежить від будь-яких припущень про розподіл *X* або *Y* або розподіл *(x, y)*.

Коефіцієнт рангової кореляції Спірмена

Нехай $(x_1, y_1),(x_2, y_2),...(x_n, y_n)$ – набір спостережень спільних випадкових величин *X* і *Y* відповідно. Для кожного елемента (x_i, y_i) обчислюється його ранг R_i. Тоді коефіцієнт кореляції Спірмена обчислюється як

$$K_S = 1 - 6\left(n(n^2-1)\right)^{-1} \sum_j d_j^2, \tag{2.16}$$

де d_j – різниця рангів x_j і y_j.

Цей коефіцієнт був введений англійським психологом Ч. Е. Спірменом, який розробив оптимізовані методики математичної статистики, створив двофакторну теорію інтелекту і факторний аналіз. Він відкрив, що результати навіть непорівнянних когнітивних тестів відображають єдиний фактор, який він назвав g-фактором. Коефіцієнт кореляції Спірмена використовується для статистичної оцінки в перевірці статистичних гіпотез.

Коефіцієнт кореляції знаків Фехнера обчислюється як

$$K_F = \frac{N^+ - N^-}{N^+ + N^-}, \qquad (2.17)$$

де N^+/N^- – число пар (x_j, y_j), у яких знаки відхилень значень від їх середніх збігаються / не збігаються відповідно.

Ч. Е. Спірмен ввів термін «факторний аналіз» і широко використовував його для аналізу показників когнітивної діяльності, що привело його до формулювання факторних моделей здібностей. Він також застосував математичні оцінки до психологічних явищ і перетворив результати свого аналізу в теорію, яка істотно вплинула на сучасну психологію.

Обмеження кореляційного аналізу

Застосування аналізу можливе за наявності достатньої кількості спостережень для вивчення. На практиці вважається, що число спостережень має не менше ніж в 5–6 разів перевищувати число факторів (також зустрічається рекомендація використовувати пропорцію, не менше ніж в 10 разів більш за кількість факторів). У разі якщо число спостережень перевищує кількість факторів в десятки разів, в дію вступає закон великих чисел, який забезпечує взаємопогашення випадкових коливань.

Також необхідно, щоб сукупність значень усіх факторних і результативних ознак підпорядковувалася багатовимірному нормальному розподіленню. Якщо обсяг сукупності недостатній для проведення формального тестування на нормальність розподілення, то відмінний від нормального закон визначається візуально на основі кореляційного поля. Якщо в розташуванні точок на цьому полі спостерігається лінійна тенденція, можна припустити, що сукупність вихідних даних підпорядковується нормальному закону. Крім того, вихідна сукупність значень повинна бути якісно однорідною. Сам по собі факт кореляційної залежності не дає підстави стверджувати, що одна із змінних передує або є причиною змін або те, що змінні взагалі причинно пов'язані між собою, а не спостерігаються дія третього фактора.

Часткові автокореляційні функції (ЧАКФ) – це інший корисний метод дослідження періодичності сигналів, який полягає в дослідженні часткової АКФ сигналу з віддаленим впливом автокореляцій із меншими лагами всередині обраного лагу, що дає більш чітку картину періодичних залежностей.

Періодична складова для конкретного лагу *k* може бути видалена узяттям різниці відповідного порядку. Це означає, що з кожного *i*-го елемента ряду віднімається *(i-k)*-й елемент. Є два аргументи на користь таких перетворень:

1) таким чином можна визначити приховані періодичні складові ряду. Пам'ятаємо, що автокореляції на послідовних лагах залежні і видалення деяких автокореляцій змінить інші автокореляції, які, можливо, зменшували їх, і, таким чином, зробити деякі інші сезонні складові більш помітними;

2) видалення сезонних складових робить ряд стаціонарним, що необхідно для застосування АРПСС (авторегресія проінтегрованого сковзаючого середнього) та інших методів, наприклад спектрального аналізу.

Процедури оцінки параметрів і прогнозування припускають, що математична модель процесу відома. В реальних даних часто немає чітко виражених регулярних складових. Окремі спостереження містять значну помилку, тоді як потрібно не тільки виділити регулярні компоненти, але також зробити прогноз. Методологія АРПСС, яка була розроблена Боксом і Дженкінсом (1976), дозволяє це зробити. Метод надзвичайно популярний у багатьох прикладних моделях, і практика підтвердила його потужність і гнучкість. АРПСС – достатньо складний метод, його не так просто використовувати, і потрібна тривала практика, щоб ним оволодіти.

Два основні процеси моделі АРПСС:

1. ***Авторегресія (АР)***. Більшість часових рядів містять елементи, які послідовно залежать один від одного. Таку залежність можна виразити рівнянням:

$$X(t) = X_0 + \sum_{j=1}^{n-1} C_j X(t - \tau_j) + \varepsilon(t), \qquad (2.18)$$

де $X_0 = \text{const}$, ε – випадковий вплив, τ_j – лаг, який для часових рядів можна задати як $\tau_j = j \cdot \Delta t$, Δt = постійний (або змінний) крок за часом, C_j – параметри АР.

Процес АР буде стаціонарним тільки тоді, коли його параметри лежать у певному діапазоні. Наприклад, якщо є тільки один параметр, то він повинен знаходитися в інтервалі $-1 < C_j < 1$. В протилежному випадку попередні значення накопичуватимуться і значення наступних можуть бути необмеженими, отже, ряд не буде стаціонарним. За наявності кількох параметрів АР можна визначити аналогічні умови, що забезпечують стаціонарність.

2. ***Процес прослизаючого середнього***. На відміну від процесу АР, в цьому процесі кожен елемент ряду схильний до сумарному впливу попередніх помилок. У загальному вигляді це можна у вигляді (2.17), де $C_j(..., \tau_{j-1}, \tau_j, \tau_{j+1}, ...)$ – параметри прослизаючого середнього.

Оцінювання моделі:

– оцінки параметрів. Якщо значення обчисленої статистики не значимі, відповідні параметри в більшості випадків видаляються з моделі;

– інші критерії якості, наприклад, порівняння прогнозу, побудованого за урізаним рядом з відомими (вихідними) даними.

3. ***Просте експоненціальне згладжування (ПЕЗ)*** – це такий спосіб згладжування часових рядів, коли на наступне значення впливають усі попередні спостереження (тобто попередня історія динамічного процесу) і при цьому враховується старіння інформації у міру віддалення. Таким чином, чим

раніше було зроблене спостереження порівняно з поточним моментом часу, тим менше воно повинно впливати на величину прогнозної оцінки, що досягається шляхом надання меншого вагового коефіцієнту в міру «старіння» інформації. ПЕЗ реалізується за допомогою рекурентної формули:

$$\hat{X}(t) = aX(t) + (1-a)\hat{X}(t-1), \tag{2.19}$$

де $a \in]0;1[$ – константа, що згладжує.

Застосування (2.19) до часових рядів з трендом веде до систематичної помилки за рахунок відставання згладжених значень $\hat{X}(t)$ від фактичних, тому для підтримки рівня тренду застосовується спеціальне двопараметричне лінійне експоненційне згладжування з двома параметрами $\{a,b\} \in]0;1[$, що згладжують (метод Хольта):

$$\begin{aligned} \hat{X}(t) &= aX(t) + (1-a)(\hat{X}(t-1) + b(t-1)), \\ b(t) &= b(X(t) - X(t-1)) = (1-b)\,b(t-1). \end{aligned} \tag{2.20}$$

Ітеративна процедура (2.20) згладжує одночасно і тренд, і випадкові збурення.

4. ***Індекси якості підгонки***

Найпряміший спосіб оцінки прогнозу, отриманого на основі певного значення – побудувати графік спостережуваних значень і прогнозів на один крок вперед (рис. 2.11). З наведеного графіка видно, на яких ділянках прогноз краще або гірше.

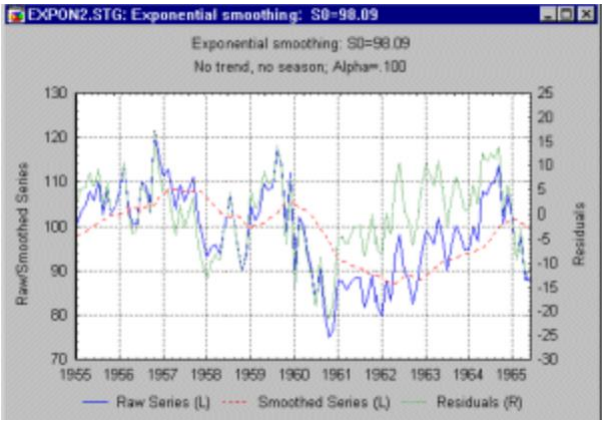


Рис. 2.11. Порівняння точності прогнозу на базі різних моделей

Візуальна перевірка точності прогнозу часто дає найкращі результати, але також є інші можливості оцінки помилки для визначення моделі з найкращими параметрами:

1. ***Середня помилка (СО)*** обчислюється простим усередненням помилок на кожному кроці. Очевидним недоліком цього заходу є те, що позитивні і негативні помилки анулюють одна одну, тому вона не є хорошим індикатором якості прогнозу.

2. ***Середня абсолютна помилка (САО)*** обчислюється як середнє абсолютних помилок. Якщо вона =0, то маємо досконалу підгонку (прогноз). У порів-

нянні із середньою квадратичною помилкою, цей захід не надає надто великого значення викидам.

3. ***Сума квадратів помилок (SSE),*** среднеквадратична помилка. Ці величини обчислюються як сума (або середнє) квадратів помилок. Це найбільш часто використовувані індекси якості підгонки.

4. ***Відносна помилка (ВО).*** У всіх попередніх використовувалися справжні значення помилок. Звісно ж, природно висловити індекси якості підгонки в термінах відносних помилок. Наприклад, під час прогнозу місячних продажів, які можуть сильно флуктувати (наприклад, сезонні) з місяця в місяць, ви можете бути цілком задоволені прогнозом, якщо він має точність 10 %. Іншими словами, під час прогнозування абсолютна помилка може бути не такою цікавою, як відносна. Щоб врахувати відносну помилку, було запропоновано кілька різних додаткових індексів.

2.2.4. Перевірка статистичних гіпотез – це клас задач математичної статистики.

Статистична гіпотеза – це припущення про вид розподілу і властивості випадкової величини, що спостерігається, яке можна підтвердити або спростувати із застосуванням статистичних методів.

2.2.4.1. Критерій Стьюдента

t-критерій Стьюдента – це метод статистичної перевірки гіпотез (статистичних критеріїв). Найбільш часті випадки застосування t-критерію пов'язані з перевіркою рівності середніх значень у двох вибірках. Критерій був розроблений У. Госсетом для оцінки якості пива в компанії «Гіннес». У зв'язку із зобов'язаннями нерозголошення комерційної таємниці (керівництво Гіннесса вважало такою таємницею використання в своїй діяльності статистичного апарату), стаття Госсета вийшла в 1908 р. в журналі «Біометрика» під псевдонімом «Student»[17] (Студент). Застосування критерію пов'язано з т. н. «нульовою гіпотезою».

Нульова гіпотеза (Null Hypothesis H_0) – це прийняте за замовчуванням припущення про те, що не існує зв'язку між двома подіями, що спостерігаються, поки не можна довести зворотне (аналогія «презумпції невинності», коли підозрюваного вважають невинним, поки не буде доведено протилежне). Ціллю статистичної обробки даних спостережень є спростування нульової гіпотези, тобто підтвердження наявності зв'язку між двома подіями, феноменами, і т. п. Математична статистика дає кількісні умови, за виконання яких нульова гіпотеза може бути відкинута. Часто в якості нульової гіпотези виступають припущення про відсутність взаємозв'язку або кореляції між

[17] Student. The probable error of a mean // Biometrika. 1908. № 6 (1). P. 1-25.

досліджуваними змінними, про відсутність відмінностей (однорідності) в розподілах в 2-х або/та більш вибірках.

Під час статистичної обробки даних треба спробувати показати неспроможність H_0, неузгодженість її з існуючими даними. При цьому мається на увазі, що повинна бути прийнята інша, альтернативна (конкуруюча) гіпотеза. Якщо статистичний аналіз таки підтверджує H_0, то вона не відкидається.

1) ***Одновибірковий t-критерій***. Застосовується для перевірки нульової гіпотези H_0 про рівність математичного очікування E(X) серії спостережень $\{X_1, X_2, ..., X_n\}$ деякому значенню m (тобто E(X)=m).

З урахуванням передбачуваної незалежності спостережень і оцінки дисперсії t-статистика має вигляд

$$t = \sqrt{n}\,\frac{E(X) - m}{\sigma_X}. \qquad (2.21)$$

Отже, у разі перевищення значення статистики за абсолютною величиною критичного значення конкретного розподілу (за заданого рівня значущості) нульова гіпотеза відкидається.

2) ***Двовибірковий t-критерій для незалежних вибірок.*** Нехай є дві незалежні вибірки обсягами n_1 і n_2 нормально розподілених випадкових величин X_1 і X_2. Необхідно перевірити нульову гіпотезу H_0 рівності математичних очікувань цих випадкових величин (тобто $E(X_1) = E(X_2)$). Тоді критерій має вигляд

$$t = \sqrt{n_1 n_2}\,\frac{|E(X_1) - E(X_2)|}{\sqrt{\sigma^2_{X_1} n_2 + \sigma^2_{X_2} n_1}}. \qquad (2.22)$$

Якщо розміри рядів значно відрізняються (наприклад, $n_1 >> n_2$), застосовується точніша формула

$$t = \frac{\sqrt{n_1 n_2 (n_1 + n_2 - 2)}\,|E(X_1) - E(X_2)|}{\sqrt{((n_1 - 1)\sigma^2_{X_1} + (n_2 - 1)\sigma^2_{X_2})(n_1 + n_2)}}. \qquad (2.23)$$

Кількість ступенів свободи такої динамічної системи $n = n_1 + n_2 - 2$.

3) ***Інші критерії.*** Існують також двовибірковий *t*-критерій для залежних вибірок; перевірка лінійного обмеження на параметри лінійної регресії; перевірка гіпотез про коефіцієнт лінійної регресії.

2.2.4.2. Критерій Фішера (F-критерій) – це статистичний критерій, тестова статистика якого за виконання нульової гіпотези має розподіл Фішера (F-розподіл). Приклади F-тестів:

– ***F-тест на рівність дисперсій***. Нехай є дві вибірки об'ємом *m* і *n* відповідно випадкових величин *X* і *Y*, що мають нормальний розподіл. В цьому

випадку H_0 стверджує, що D(X)=D(Y), тобто необхідно перевірити рівність дисперсій вибірок, і статистика тесту є

$$F(m,n) = \sigma_X^2 / \sigma_Y^2 . \quad (2.24)$$

Якщо статистика (2.23) більше критичного значення, що відповідає обраному рівню значущості, то дисперсії випадкових величин визнаються неоднаковими;

– ***Перевірка значущості лінійної регресії.*** Цей тест дуже важливий у регресійному аналізі і є окремим випадком перевірки обмежень (*k*). В цьому випадку H_0 стверджує, що всі коефіцієнти при чинниках регресійної моделі (2.7) є нулі (тобто всього маємо *k-1* обмеження). В цьому випадку коротка модель – це просто константа в якості фактора, тобто коефіцієнт детермінації короткої моделі дорівнює нулю.

Статистика тесту:

$$F(n,k) = \frac{R^2(n-k)}{(k-1)(1-R^2)} . \quad (2.25)$$

Відповідно, якщо значення цієї статистики більше критичного значення за даного рівня значущості, то H_0 відкидається, що означає статистичну значущість регресії. В іншому випадку залежність визнається незначною.

2.2.5. Кластерний аналіз (англ. cluster – гроно, скупчення) був вперше запропонований в 1939 р. американським зоопсихологом Робертом Тріоном. Головне призначення кластерного аналізу – розбивка безлічі досліджуваних об'єктів і ознак на однорідні у відповідному розумінні групи, або кластер, що є задачею класифікації даних і виявлення в них структури в ній.

Задача кластерного аналізу полягає в тому, щоби різнорідні дані, які містяться в множині *Q*, розбити на ціле число *m* підмножин (кластерів) Q_1, Q_2, ..., Q_m так, щоб кожен об'єкт G_j належав до однієї і тільки однієї підмножині розбиття і щоб об'єкти, які належать одному і тому ж кластеру, були подібними, в той час як об'єкти, які належать різним кластерам, були різнорідними.

Рішенням задачі кластерного аналізу є розбиття, що задовольняють певним критеріям оптимальності. Цей критерій може являти собою деякий функціонал, що виражає рівні бажаності різних розбивок угруповань, який називають цільовою функцією. Наприклад, в якості цільової функції може бути взята внутрішньогрупова сума квадратів відхилення:

$$W = \sigma_n = \sum_{j=1}^{n} \left(x_j - \mu \right)^2 = \sum_{j=1}^{n} x_j^2 - \frac{1}{n} \left(\sum_{j=1}^{n} x_j \right)^2 , \quad (2.26)$$

де x_j – чисельна міра (міри) об'єктів множини *Q*.

Значна перевага кластерного аналізу в тому, що він дозволяє виробляти розбиття об'єктів не за одним параметром, а за цілим набором ознак. Крім того, кластерний аналіз на відміну від більшості математико-статистичних методів, не накладає ніяких обмежень на вид розглянутих об'єктів і дозволяє розглядати безліч вихідних даних практично довільної природи. Це має велике значення, наприклад, для прогнозування кон'юнктури, коли показники мають різноманітний вигляд, що утруднює застосування традиційних економетричних підходів.

Кластерний аналіз має важливе значення для класифікації сукупностей часових рядів, які характеризують економічний розвиток (наприклад, загальногосподарської та товарної кон'юнктури). Тут можна виділяти періоди, коли значення відповідних показників були досить близькими, а також визначати групи часових рядів, динаміка яких найбільш схожа.

Кластерний аналіз застосовується:

– у маркетингу для сегментації конкурентів і споживачів;
– у менеджменті для розбиття персоналу на різні за рівнем мотивації групи, класифікації постачальників, виявлення схожих виробничих ситуацій;
– у медицині для класифікації симптомів, пацієнтів, препаратів;
– у соціології і психології для розбиття респондентів на однорідні групи.

Кластерний аналіз працює навіть тоді, коли даних мало і не виконуються вимоги нормальності розподілів випадкових величин та інші вимоги класичних методів статистичного аналізу.

Кластерний аналіз використовується для:

– розробки типології або класифікації;
– дослідження корисних концептуальних схем групування об'єктів;
– генерації гіпотез на основі дослідження даних;
– перевірки гіпотез або дослідження, чи дійсно типи (групи), які виділені тим чи іншим способом, присутні в наявних даних.

Кластер має наступні ***математичні характеристики***: центр, радіус, середньоквадратичне відхилення, розмір кластера (рис. 2.12а,б)

Центр кластера – це середнє геометричне місце точок у просторі змінних.

Радіус кластера – максимальна відстань точок від центру кластера.

Кластери можуть перекриватися. У цьому випадку неможливо за допомогою математичних процедур однозначно віднести об'єкт до одного з двох кластерів. Такі об'єкти називають спірними.

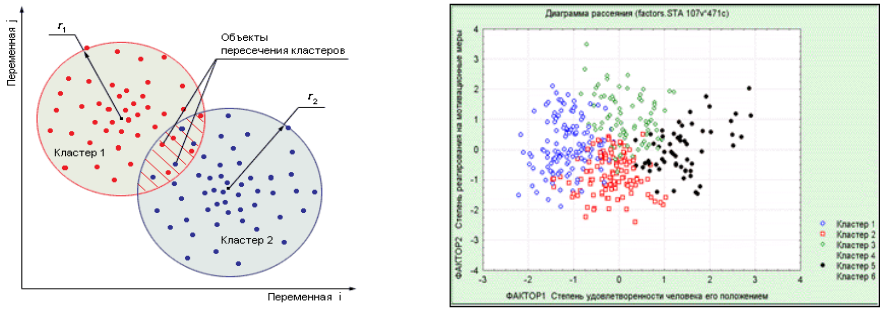
Спірний об'єкт – це об'єкт, який у міру подібності може бути віднесений до декількох кластерів.

Розмір кластера може бути визначений або за радіусом кластера, або за середньоквадратичним відхиленням об'єктів від центру кластера.

Об'єкт належить до кластеру, якщо відстань від об'єкта до центру кластеру менше за радіус кластера. Якщо ця умова виконується для двох

і більше кластерів, об'єкт є спірним. Неоднозначність цього рішення може бути усунена експертом або аналітиком.

Вибір масштабу в кластерному аналізі має велике значення. Наприклад, якщо ознаки x в наборі даних A на два порядки більше даних ознаки y: значення змінної x знаходяться в діапазоні від 100 до 1000, а значення змінної y – в діапазоні від 0 до 1. Тоді під час розрахунків відстані між точками, яка відповідає локації об'єктів в просторі їх властивостей, змінна x буде практично повністю домінувати над змінною y. Таким чином, через неоднорідність одиниць вимірювання ознак стає неможливо коректно розрахувати відстані між точками.



а *б*

Рис. 2.12. Кластери даних у просторі двох ознак: 2 (а) і 6 (б) кластерів, які перекриваються

Існує >100 різних алгоритмів кластеризації, однак найбільш часто використовуються ієрархічний кластерний аналіз і метод *k*-середніх. Проблема вибору вирішується за допомогою попередньої нормалізації (normalization), яка приводить всі змінні до єдиного діапазону значень у вигляді одного з перетворень

$$z = \frac{x-\mu}{\sigma}\ ,\ z = \frac{x}{\mu}\ ,\ z = \frac{x}{x_{\max}}\ ,\ z = \frac{x-\mu}{x_{\max}-x_{\min}}\ . \qquad (2.27)$$

де $x_{\max}, x_{\min}$ – найбільше та найменше значення x на заданому інтервалі.

Крім того, використовують коефіцієнти важливості (ваги), які відповідають значимості відповідної змінної, згідно з експертною оцінкою.

У кластерному аналізі для кількісної оцінки подібності вводиться поняття метрики. Подібність або відмінність між класифікованими об'єктами встановлюється залежно від метричної відстані між ними. Якщо кожен об'єкт описується k ознаками, то він може бути представлений як точка в k-вимірному просторі, і схожість з іншими об'єктами буде визначатися як відповідна відстань.

Відстанню (метрикою) між об'єктами в просторі параметрів називається така величина d_{ab}, яка задовольняє умовам:

1. $d_{ab} > 0,\ d_{aa} = 0;$
2. $d_{ab} = d_{ba};$
3. $d_{ab} + d_{bc} \geq d_{ac}.$

Мірою близькості (подібності) називається обмежена величина m_{ab}, яка зростає зі зростанням близькості об'єктів, а також:

1. m_{ab} неперервна;
2. $m_{ab} = m_{ba};$
3. $0 \leq m_{ab} \leq 1.$

Існує можливість простого переходу від відстаней до міри близькості за формулою $m=(1+d)^{-1}$.

Типи метрик:

1) евклідова відстань $d_{E,ij} = \left(\sum_{l=1}^{m} \left(x_i^l - x_j^l \right)^2 \right)^{1/2}$;

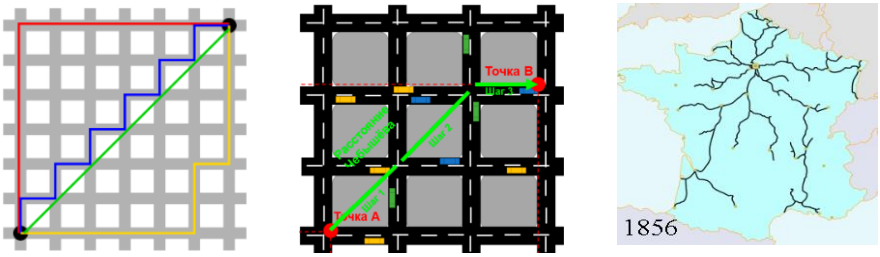
2) квадрат евклідової відстані $d_{E,ij}^2 = \sum_{l=1}^{m} \left(x_i^l - x_j^l \right)^2$;

3) узагальнена ступенева відстань (Міньковського) (h > 1) $d_{\kappa,ij} = \left(\sum_{l=1}^{m} \left(x_i^l - x_j^l \right)^{\kappa} \right)^{1/h}$;

4) відстань міських кварталів (Манхеттенська відстань, рис. 2.13а) $d_{\text{H},ij} = \sum_{l=1}^{m} \left| x_i^l - x_j^l \right|$;

5) відстань Чебишева (рис. 2.13б) $d_{\text{Ch},ij} = \max_{1 \leq i,\, j \leq l} \left| x_i - x_j \right|$;

6) французька залізнична метрика (рис.2.13в) – найкоротша відстань між двома точками на поверхні вздовж залізничних колій.



а *б* *в*

Рис .2.13. Деякі типи метрик: Манхеттенська відстань (а); відстань Чебишева (б); французька залізнична метрика (в)

Евклідова відстань є найпопулярнішою метрикою в кластерному аналізі. Це просто геометрична відстань в багатовимірному просторі. Геометрично вона найкраще об'єднує об'єкти в кулястих скупченнях.

Квадрат евклідової відстані використовується для додавання великих ваг більш віддаленим один від одного об'єктів.

Узагальнена ступенева відстань представляє математичний інтерес як універсальна метрика.

Відстань Чебишова. Цю відстань варто використовувати, коли необхідно визначити два об'єкти як «різні», якщо вони відрізняються за якоюсь однією ознакою.

Манхеттенська відстань (відстань міських кварталів), так звана «хеммінгова» або «сіті-блок»–відстань. У більшості випадків ця метрика приводить до результатів, подібних до отриманих з евклідовою відстаню, однак для МВ вплив окремих похибок менший, ніж під час використання ЕВ, оскільки тут координати НЕ зводяться в квадрат.

Відсоток незгоди – обчислюється, якщо дані є категоріальними (не числа), за формулою:

$$d = (\text{кількість таких пар } (x_j, y_j), \text{ що } x_j \neq y_j)/n.$$

Для ***виділення кластерів*** використовуються ієрархічні ***агломеративні*** (АМ) і ***дивізимні*** (подільні) (ДМ) методи.

1) ієрархічні агломеративні методи (Agglomerative Nesting (AGNES)) – це група методів послідовного об'єднання елементів з відповідним зменшенням числа кластерів.

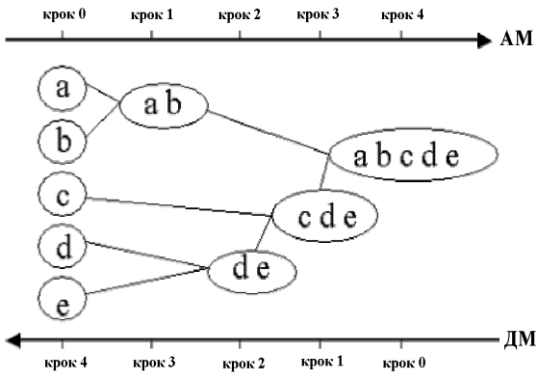


Рис. 2.14. Схема агломеративних (АМ) і дивізимних (ДМ) методів кластеризації

На початку роботи алгоритму кожний об'єкт є окремим кластером. На першому кроці найбільш схожі об'єкти об'єднуються в новий кластер. На наступних кроках об'єднання триває до тих пір, поки всі об'єкти не будуть складати один кластер.

2) Дівизимні методи (DIvisive ANAlysis (DIANA)) – це логічна протилежність АМ. На початку роботи алгоритму всі об'єкти належать одному кластеру, який на наступних кроках ділиться на менші кластери, в результаті чого утворюється послідовність розділених кластерів (рис. 2.14).

Правила об'єднання або зв'язку

1) Одиничний зв'язок (***метод найближчого сусіда***). Відстань між двома кластерами визначається відстанню між двома найбільш близькими об'єктами (найближчими сусідами) в різних кластерах. В результаті ітерацій кластери мають тенденцію бути представленими довгими «ланцюжками».

2) Повний зв'язок (***метод найбільш віддалених сусідів***). Відстані між кластерами визначаються найбільшою відстанню між будь-якими двома об'єктами в різних кластерах (найбільш віддаленими сусідами). Метод добре працює, коли об'єкти насправді з різних за всіма параметрами «гаїв». Якщо ж кластери мають в деякому роді подовжену форму (ланцюгову), метод непридатний.

3) ***Незважене попарне середнє*** (unweighted pair-group method using arithmetic averages, UPGMA). Відстань між двома різними кластерами обчислюється як середня відстань між усіма парами об'єктів в них. Метод ефективний, коли об'єкти в дійсності формують різні «гаї», однак він працює однаково добре і випадках протяжних (ланцюгових) кластерів.

4) ***Зважене попарне середнє*** (weighted-PGMA= WPGMA). Ідентично до UPGMA, але число об'єктів кластеру є ваговим коефіцієнтом і краще працює у випадку нерівних за розміром кластерів.

5) ***Незважений центроїдний метод*** (unweighted pair-group method using the centroid average). Відстань між двома кластерами визначається як відстань між їх центрами тяжкості.

6) ***Зважений центроїдний метод*** (weighted pair-group method using the centroid average). Ідентичний попередньому, але розміри кластерів використовуються як вагові коефіцієнти.

7) ***Метод Варда*** (Ward, 1963). Використовує методи дисперсійного аналізу для оцінки відстаней між кластерами. Метод мінімізує внутрішньогрупову суму квадратів

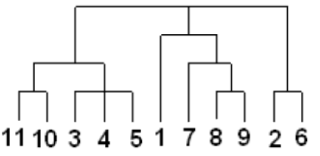
$$ESS = \sum_i \sum_j \sum_k \left| X_{ijk} - \bar{X}_{jk} \right|^2 \to \min \qquad (2.28)$$

для будь-яких двох (гіпотетичних) кластерів, які можуть бути сформовані на кожному кроці. Таким чином, цільовою функцією задачі оптимізації є сума квадратів відстаней між кожною точкою (об'єктом) і середньої по кластеру, який містить цей об'єкт. На кожному кроці об'єднуються такі два кластери, які приводять до мінімального збільшення цільової функції. Цей метод направлений на об'єднання близько розташованих кластерів, є досить ефективним, однак виникають проблеми за наявності кластерів з малими та великими розмірами.

Об'єднання, або метод деревоподібної кластеризації використовується під час формування кластерів несхожості або відстані між об'єктами. Ці відстані можуть визначатися в одновимірному або багатовимірному просторі. Наприклад, якщо треба кластеризувати типи їжі в кафе, то можна взяти до уваги кількість калорій, які в ній містяться, ціну, суб'єктивну оцінку смаку, вегетаріанська чи ні та ін.

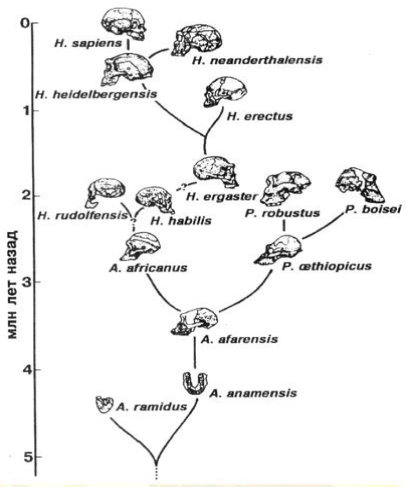
Узагальнена алгомеративна процедура. На першому кроці кожен об'єкт вважається окремим кластером, обчислюється матриця відстаней D (nxn). На наступному кроці об'єднуються два найближчі об'єкти, які утворюють новий клас, визначаються відстані від цього класу до всіх інших об'єктів, і розмірність матриці є *n-1*. Процедура повторюється поки все об'єкти не об'єднаються в один клас. Якщо відразу кілька об'єктів (класів) мають мінімальну відстань, то можливі дві стратегії: вибрати одну випадкову пару (метод ієрархічної класифікації, частіше використовується), або об'єднати відразу всі пари (метод найближчих сусідів, рідше використовується) (не плутати з алгоритмом «найближчого сусіда»!).

Ієрархічні алгоритми пов'язані з побудовою ***дендрограм*** (від грецького dendron – «дерево»), які є результатом ієрархічного кластерного аналізу. Приклад кластеризації 11 об'єктів за методом найближчого сусіда наведений на рис. 2.15а, а дендрограма зв'язку приметів з деякими сучасними мавпами і людиною за сукупністю ознак – на рис. 2.15б. На рис. 2.15в,г представлені еволюційні дерева гомінід і деяких тварин, побудовані за відстанями між парами.

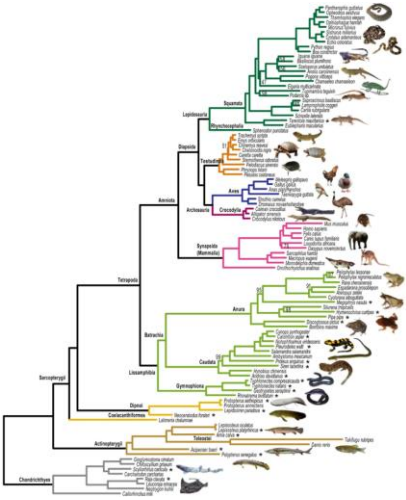


а

б



в



г

Рис. 2.15. Приклад дендрограми (а); дерево зв'язку приматів, мавп і людини (б); еволюційне древо гомінід (в) і тварин (г)

Метод k–середніх (Дж. Мак-Куїн, 1967)

Процес класифікації починається з завдання початкових умов (кількість утворених кластерів, поріг завершення процесу класифікації і т. д.). На відміну від ієрархічних процедур метод *k*-середніх не вимагає обчислення і зберігання матриці відстаней між об'єктами. Для початку класифікації повинні бути задані *k* обраних об'єктів, які будуть служити еталонами, тобто центрами кластерів. Алгоритми еталонного типу зручні і швидкодіючі, зручні для обробки великих статистичних сукупностей. Важливу роль грає вибір початкових умов, які впливають на тривалість процесу класифікації і на його результати.

Алгоритм методу k–середніх має наступні кроки:

1. Розглядаємо *n* спостережень $\overline{X} = (x_1, x_2, ..., x_n)$, кожне з яких характеризується *m* ознаками. Ці спостереження треба розбити на *k* кластерів. Для початку з *n* спостережень відбираються випадковим чином або задаються експертом, виходячи з будь-яких апріорних міркувань, *k* точок (об'єктів), які приймаються за еталони. Кожному еталону присвоюється порядковий номер 1,2,…,*k*, який одночасно є і номером кластеру.

2. На першому кроці з решти (*N*-*k*) об'єктів витягується точка X_j з координатами $X_j = (x_{j1}, x_{j2}, ..., x_{jm})$ і перевіряється, до якого з еталонів (центрів кластерів) вона знаходиться найближче. Для цього використовується одна з вищевказаних метрик. В результаті об'єкт X_j приєднується до найближчого центру кластеру.

Після цього еталон замінюється новим, переліченим з урахуванням приєднаної точки, і його вага (кількість об'єктів, що входять до цього кластеру) збільшується на одиницю. Якщо зустрічаються дві або більше мінімальних відстані, то *i*-й об'єкт приєднують до центру з найменшим порядковим номером.

3. На наступному кроці вибираємо точку X_{j+1}, і для неї повторюються всі процедури. Таким чином, через (*n*-*k*) кроків всі об'єкти сукупності виявляться віднесеними до одного з *k* кластерів, але на цьому процес розбиття не закінчується.

4) Для того щоб домогтися стійкості розбиття, всі точки $(x_1, x_2, ..., x_n)$ знову перераховують і приєднують до раніше отриманих кластерів, при цьому вагові коефіцієнти продовжують накопичуватися. Нове розбиття порівнюється з попереднім. Якщо обидва збігаються, робота алгоритму завершується. В іншому випадку цикл повторюється з іншими *k*-еталонами, метрикою, і т.д.

5. Остаточне розбиття має центри тяжкості, які не збігаються з еталонами, їх можна позначити $(C_1, C_2, ..., C_k)$. При цьому кожна точка X_j *(j= 1,2,…,n)* буде відноситися до найближчого кластеру у розумінні вибраної метрики.

6. Є дві модифікації методу k–середніх: 1) перерахунок центру тяжкості кластеру робиться після кожної зміни його складу; 2) або лише після того, як буде завершено перегляд всіх даних. В обох випадках ітеративний алгоритм цього методу мінімізує дисперсію всередині кожного кластеру, хоча в явному вигляді такий критерій оптимізації не використовується.

2.2.6. Дискримінантний аналіз – це розділ багатовимірного статистичного аналізу, який дозволяє вивчати відмінності між двома і більше групами об'єктів за кількома змінними одночасно. Дискримінантний аналіз – це загальний термін, що належить до кількох тісно пов'язаних зі статистичними процедурами, які можна розділити на методи інтерпретації міжгрупових відмінностей і методи класифікації спостережень за групами. Під час інтерпретації потрібно відповісти на питання: чи можливо, використовуючи певний набір змінних, відрізнити одну групу від іншої, наскільки добре ці змінні допомагають провести дискримінацію і які з них найбільш інформативні?

Методи класифікації, які пов'язані з отриманням однієї або декількох функцій, забезпечують можливість зарахування конкретного об'єкта до однієї з груп. Ці функції називаються класифікуючими і залежать від значень змінних таким чином, що з'являється можливість зарахувати кожен об'єкт до однієї з груп.

Завдання дискримінантного аналізу можна розділити на ***три типи***.

1. Завдання першого типу часто зустрічаються в медичній практиці. Припустимо, що ми маємо в своєму розпорядженні інформацію про деяке число індивідуумів, хвороба кожного з яких належить до одного з двох або більше діагнозів. На основі цієї інформації потрібно знайти функцію, що дозволяє поставити у відповідність новим індивідуумам характерні для них діагнози. Побудова такої функції і становить завдання дискримінації.
2. Другий тип завдань належить до ситуації, коли ознаки приналежності об'єкта до тієї чи іншої групи втрачені і їх потрібно відновити. Прикладом може служити визначення статі давно померлої людини за останками, знайденими під час археологічних розкопок.
3. Завдання третього типу пов'язані з передбаченням майбутніх подій на підставі наявних даних. Такі завдання виникають під час прогнозу віддалених результатів лікування, наприклад прогноз виживання оперованих хворих.

2.2.7. Факторний аналіз

Фактори – це гіпотетичні безпосередньо не вимірювані приховані (патентні) змінні, які в тією чи іншою мірою пов'язані з вимірюваними характеристиками – проявами цих факторів. Ідея факторного аналізу заснована на припущенні, що є низка величин, які не відомі досліднику і які викликають різні співвідношення між змінними (впливають на них). Тобто структура зв'язків між *n* аналізованими ознаками $(x_1, x_2, ..., x_n)$ може бути пояснена тим, що всі ці змінні залежать (лінійно або нелінійно) від меншого числа інших, безпосередньо не вимірюваних факторів $\{f_1, f_2, ..., f_m\}$, $m<n$, які прийнято називати загальними.

Таким чином, факторний аналіз (в широкому розумінні) – сукупність моделей і методів, що орієнтовані на виявлення, конструювання та аналіз внутрішніх факторів за інформацією про їх «зовнішні» прояви. У вузькому

розумінні під факторним аналізом розуміють методи виявлення гіпотетичних (неспостережуваних) чинників, які можуть пояснити кореляційну матрицю кількісних спостережуваних змінних.

Існують 2 рівні факторного аналізу: 1) розвідка (exploration), коли невідомі ні кількість факторів, ні структура зв'язку; 2) перевірка (confirmation), коли здійснюється перевірка гіпотез про структуру факторів та характер впливу.

Більшість моделей конструюється так, щоб загальні фактори виявилися некорельованими. При цьому в загальному випадку не постулюється можливість однозначного відновлення значень кожної із спостережуваних ознак X(j) по відповідних значеннях загальних факторів $\{f_1, f_2, ..., f_m\}$, а саме допускається, що будь-яка з вихідних ознак *X(j)* залежить також і від деякої своєї «специфічної» випадкової компоненти *e(j)* характерного фактора, який і обумовлює статистичний характер зв'язку між X(j) і $\{f_1, f_2, ..., f_m\}$.

Кінцева мета статистичного дослідження на основі факторного аналізу, як правило, полягає у виявленні та інтерпретації латентних загальних факторів з одночасним прагненням мінімізувати їх число і ступінь залежності *X(j)* від своїх характерних факторів *e(j).*

Як і в будь-якій моделі, ця мета може бути досягнута лише наближено. Прийнято вважати статистичний аналіз такого роду успішним, якщо велике число змінних вдалося пояснити малим числом факторів.

Чи є фактори причинами зв'язку між спостереженнями $(x_1, x_2, ..., x_n)$ або вони є деякими агрегованими теоретичними конструкціями, залежить від інтерпретації моделі.

Основні етапи методу факторного аналізу

1. Формування мети;
 1a. Дослідницька (виявлення факторів і їх аналіз)
 1b. Прикладна (побудова агрегованих характеристик для прогнозування і управління);
2. Вибір сукупності ознак і об'єктів;
3. Отримання вихідної структури фактора;
4. Коригування структури фактора виходячи з цілей дослідження.

Під час перевірочного факторного аналізу критерій якості – відповідність структури заданої дослідником, під час розвідувального досягнення «простої структури», коли зв'язок змінних з будь-яким фактором виражений максимально чітко.

Іноді корекція призводить до корельованих факторів. Це дозволяє повторно застосувати факторний аналіз, використовуючи в якості вихідної інформації значення факторів. У зв'язку з цим можливий такий проміжний етап, як

5. Виявлення факторів другого порядку – більш загальних і глибоких категорій досліджуваного явища/процесу.
6. Інтерпретація і використання результатів.

***Основні ідеї факторного аналізу*:**
– сутність явищ/процесів, що спостерігаються, укладена в їх простих і різноманітних проявах, які можуть бути пояснені за допомогою комбінації декількох основних факторів;
– загальну сутність можна осягти через нескінченне наближення.
Реалізуючи ці принципи, факторний аналіз дозволяє отримати просту модель взаємозв'язку явищ на більш високому, причиному рівні, що дуже цінно як у теоретичних, так і в практичних дослідженнях.

Для ефективного застосування методу:
1) необхідно встановити межі області, наділеної структурою, провести аналіз обсягу даних, встановити рівень розрізнення змінних;
2) при виборі змінних важливо зберегти можливість їх класифікації;
3) число змінних *p* повинно відповідати числу спостережень *n*: $n >> p$.

Факторний аналіз визначає кількісні співвідношення між змінними. Ці співвідношення можуть бути виражені в коефіцієнтах або процентних відношеннях, які вказують, до якої міри розглянуті змінні схильні до впливу деяких загальних факторів. Для зручності будемо вважати досліджувані спостереження $(x_1, x_2, ..., x_n)$ нормованими.

Традиційна модель заснована на використанні матриці спостережень X_{ik}, елемент x_{ik} якої – значення *k*-ї ознаки для *i* -го об'єкта у вигляді лінійних комбінацій значень факторів $\{f_1, f_2, ..., f_m\}$ з невязками e_{ik}:

$$X_{ik} = a_{1k} f_{i1} + a_{2k} f_{i2} + ... + a_{pk} f_{ip} + e_{ik}, \qquad (2.29)$$

де a_{jk} – коефіцієнти впливу факторів на ознаку *k*.

Вибір коефіцієнтів a_{jk} здійснюється за критерієм мінімізації кореляцій між векторами нев'язок e_{ik}.

Співвідношення факторного аналізу формально відтворюють запис моделі множинних регресій, в якій під $\{f_1, f_2, ..., f_m\}$ розуміються так звані пояснюючі змінні (фактори-аргументи). Однак принципова відмінність моделі факторного аналізу від регресійних схем полягає в тому, що змінні f_j, що виступають у ролі аргументів у моделях регресії, безпосередньо не спостерігаються в моделях факторного аналізу, в той час як у регресійному аналізі значення f_j вимірюються на статистично обстежених об'єктах.
Лінійна факторна модель в матричній формі виглядає як $X = AF + E$, де A_{ij} – прямокутна *nxm* матриця навантажень загальних факторів на досліджувані ознаки $\{X_1, X_2, ..., X_n\}$, що пов'язують їх з неспостережуваними загальними факторами $\{f_1, f_2, ..., f_m\}$. Якщо матриця A_{ij} велика, то фактори добре описують поведінку спостережуваної ознаки. Вектор-стовпець *E* визначає ту частину кожної із спостережуваних величин, яка не може бути пояснена загальними

факторами. Нев'язка E включає характерну частину досліджуваних змінних – E_s і помилку вимірювань E_n. Передбачається, що компоненти F і E некорельовані; без обмеження спільності можна розглядати $D(e_i)=1, i=1-n$ і $D(f^{(j)})=1, j=1-m$. Тоді математичне очікування спостережень $E(X)=0$ і матриця коваріацій є

$$D(X)=D(AF+E)=AA^T+L^2, \tag{2.30}$$

де AA^T називається матрицею спільності і відображає зміну величин об'єктів під впливом загальних факторів; діагональні елементи цієї матриці – це спільності h_i^2, а L^2 – характерна матриця, яка вказує на специфічний зв'язок змінних, при чому її діагональні елементи $L^2=1-h_i^2$ визначають характерність.

Таким чином, фактори, які впливають на змінні, можна розділити на три групи:

1) загальні фактори $f(j)$ ($j=1$-m): фактори, які впливають на кілька змінних $X(i)$, ($i=1$-n) одночасно (фактор, що входить в усі спостереження називається головним);

2) характерні фактори e_{si} ($i=1$-p): фактори, які одночасно впливають тільки на одну змінну;

3) фактори похибки e_{ni} ($i=1$-p): фактори, до яких належить похибка в спостереженнях, а n_i, n_j можуть бути випадковими компонентами.

Основні відмінності між загальними і характерними факторами: загальний фактор впливає на кілька змінних $X(i)$ відразу, визначаючи одну модель поведінки змінних, а специфічні – лише на одну змінну; причому змінна $X(i)$ може одночасно залежати від декількох загальних факторів, але тільки від одного характерного і одного фактора похибки.

Аналогічно дисперсію спостережуваних змінних можна розбити наступним чином: $\sigma_i^2=h_i^2+I_{si}^2+I_{ni}^2$, де I_{si}^2 – надійність, а I_{ni}^2 – характерність або ненадійність.

Отже, модельна оцінка змінних AF відтворює вихідні дані з точністю до залишків E, що представляють нев'язку (характерні і помилкові фактори) і являє собою загальну (скорочену) частину, а E – специфічну частину.

Таким чином, якщо дана вибірка спостережуваного вектора X з n елементами, то головна чисельна задача лінійної факторної моделі в тому, щоб оцінити матрицю навантажень A. Часто необхідно додатково оцінити і характерну матрицю $L2$. Таким чином, сукупність $\{A, L^2\}$ представляє структуру факторного аналізу.

Зазвичай, в факторному аналізі мало уваги приділяється характерному фактору і фактору похибки, щоб зв'язати застосовуваний факторний аналіз виключно з загальними факторами. Проте нехтування специфічними факторами не завжди виправдано. Присутність змінної з високою характерною дисперсією або високою компонентою дисперсії помилки може бути сигналом, що дана

змінна, ймовірно, випадає із загального ряду і пов'язана зі змінними, ще не включеними в розгляд. Якщо ця змінна важливіша, ніж інші, то повинні бути введені нові змінні.

Проте, традиційний факторний аналіз націлений, перш за все, на аналіз загальних факторів і відповідних факторних навантажень, тому звикла модель факторного аналізу на нормованому спостереженні записується у вигляді (2.2). Геометрично сукупність факторів *f(j)* задає базис простору факторів, а *A* – перетворення.

Таким чином, фактично маємо 2-етапну процедуру:

1. Шукаємо факторні рішення за критерієм, що враховує тільки нев'язки.
2. За допомогою обертання приводимо до виду, який найбільш відповідає цілям дослідження.
3. Ідеальною матрицею навантажень вважаємо ту, яка дозволяє максимально чітко розділити змінні за тим, який чинник проявляється в них найбільш сильно.

Отже, сформульована загальна модель факторного аналізу дозволяє ефективно досягати цілей дослідження і вже виходячи з побудови, задає певний напрямок інтерпретації результатів.

Один з необхідних моментів дослідження становить перевірка гіпотез, пов'язаних із природою і параметрами використовуваної моделі факторного аналізу. Теорія статистичних критеріїв стосовно моделей факторного аналізу розроблена слабо. Поки існують лише так звані критерії адекватності моделі, тобто критерії, призначені для перевірки гіпотези, яка полягає в тому, що досліджуваний вектор спостережень *X* допускає подання за допомогою моделі факторного аналізу з даними (заздалегідь обраним) з числом загальних факторів *m*.

2.2.8. Спектральний аналіз – це аналіз у частотній області або оцінка спектральної щільності. Він являє собою розкладання складного сигналу на більш прості частини як суму багатьох індивідуальних частотних компонентів. Будь-який процес, який вимірює різні величини (наприклад, амплітуди, потужності, інтенсивності) залежно від частоти або фази, можна назвати спектральним аналізом. Особливо добре підходять для цього періодичні функції sin(ωt), cos(ωt) або $e^{i\omega t}$, тому загальні математичні методи спектрального аналізу належать до ***аналізу Фур'є***.

Для визначення амплітуд спектра функції $X(t)$ маємо загальну формулу перетворення Фур'є

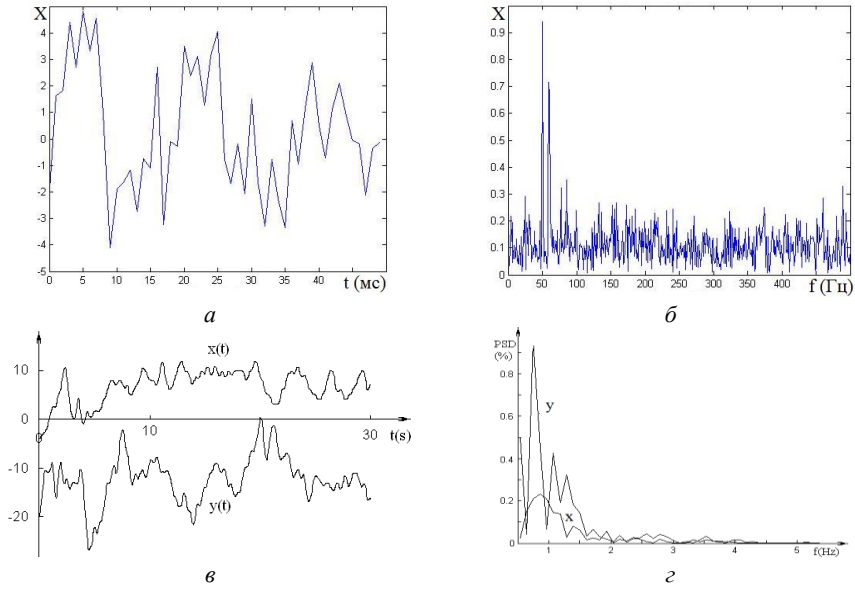
$$\tilde{X}(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} X(t)\, \mathrm{e}^{-\mathrm{i}\omega \mathrm{t}}\, dt\,, \qquad (2.31)$$

де ω – частота.

Оскільки в прикладних дослідження реєстрація сигналу проводиться з певною частотою дискретизації, функція $X(t)$ є часовим рядом і замість (2.4) для неї використовуються дискретне перетворення Фур'є, алгоритми і код якого

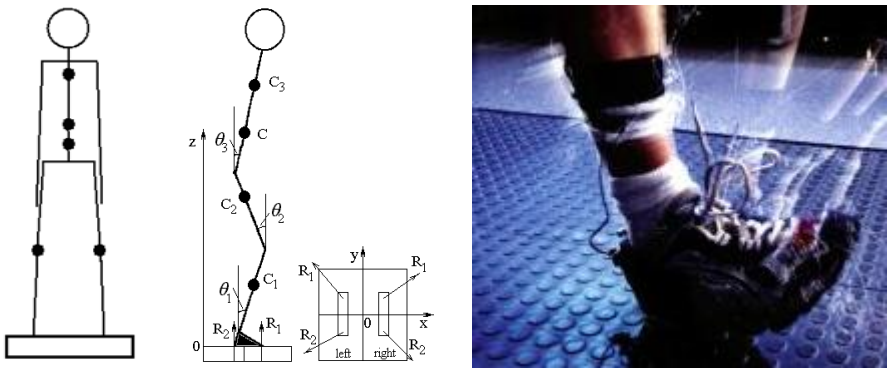
для C^{++}, C#, Python, Delphi, R можна знайти у відкритих джерелах. Найбільш популярним є швидке перетворення Фур'є (БПФ, Fast Fourier Transformation FFT), яке має складність (число арифметичних операцій) $O(N\log(N))$ на відміну від звичайного перетворення (2.4), яке має складність $O(N^2)$, де N – число точок часового ряду.

В пакеті програм MatLab ця опція виконується функцією *fft(x)*, приклад роботи якої наведений на рис. 2.16. Вихідний дискретний сигнал (рис. 2.16а) має два максимуми з частотами 50 і 60 Гц (рис. 2.16б), а решта гармонік відповідають випадковій компоненті (шум). До використання функції *fft(x)* можна спочатку відфільтрувати шум будь-яким придатним фільтром з бібліотеки MatLab і отримати чистіший результат. На рис. 2.16в наведений приклад медичних даних – ***часові ряди***, які відповідають координатам {x(t), y(t)} проекції центра мас тіла людини на горизонтальну поверхню, які можна зареєструвати стаціонарним обладнанням (стабілограф, рис. 2.17а,б) або будь-яким з wearables (вкладники до взуття, «розумне взуття» та ін. доступні на ринку (рис. 2.17в). Аналіз спектра коливань центру мас тіла показує, що майже на всіх частотах амплітуди коливань y(t) у сагітальній (боковій) площині значно більші, ніж амплітуди коливань x(t) у фронтальній площині, що є нормою для здорової людини. За наявності патологічних або вікових змін в організмі, може спостерігатися інша залежність.



а *б* *в* *г*

Рис. 2.16. Приклади незгладженого (а) і згладженого (в) сигналів та їх спектрів без (б) і з (г) попередньою фільтрацією



а *б* *в*

Рис. 2.17. Вимірювання координат центру мас людини за допомогою стабілографа (фронтальний (а), боковий (б) і сагітальній (в) вигляди)

Слід зауважити, що результати спектрального аналізу можна отримати як у вигляді залежностей амплітуд спектру $Amp(\omega) \equiv (\mathrm{Re}(\tilde{X})^2 + \mathrm{Im}(\tilde{X})^2)^{1/2}$ (рис. 2.16б), так і для спектральної щільності міцності сигналу PSD (Power Spectral Density, рис. 2.16г)

$$PSD(\omega) = \lim_{T\to\infty} \frac{\left|\tilde{E}_T(\omega)\right|}{T}, \tag{2.32}$$

де $\tilde{E}_T(\omega)$ – результат перетворення Фур'є (2.4) від міцності сигналу $E_T(\mathrm{T}) = \int_{-T/2}^{T/2} X^2(t)dt$, де T – заданий малий інтервал. При цьому для медичних даних з рис. 2.16в (2.32) використалась окремо для кривих *x(t)* і *y(t).*

2.2.9. Фрактальний аналіз – це обчислення фрактальної розмірності кривої $X(t)$ або зображення і виявлення розбіжностей між геометричною та однією з фрактальних розмірностей, найчастіше ***розмірністю Хаусдорфа***

$$D_\chi = \lim_{\varepsilon\to 0+} M_\alpha^\varepsilon(\Omega), \tag{2.33}$$

де $M_\alpha(\Omega)$ – α-міра Хаусдорфа множини Ω, $M_\alpha^\varepsilon(\Omega) = \inf\left(\Theta_\alpha(\Xi)\right)$, де *inf* береться по всім ε-покриттям Ω.

Для обмеженої множини в метричному просторі використовують ***розмірність Мінковського***

$$D_m = -\lim_{\varepsilon\to 0} \frac{\ln(\mathrm{N}(\varepsilon))}{\ln(\varepsilon)}, \tag{2.34}$$

де N – мінімальне число елементів, яке потрібне для повного покриття відповідної фрактальної структури на площині або у просторі стандартними

елементами з характерним розміром ε. Якщо ліміт не існує, можна розглядати верхню і нижню межу і говорити, відповідно, про верхнью та нижнью розмірності Міньковського. Таким чином, фрактальна розмірність – це мінімальне число ε-мірних «квадратів» для покриття заданої множини.

Фрактальні (самоподібні) структури утворюються шляхом виключення з геометричної кривої, перетину або з об'єму матеріалу деяких складових частин, які поступово самоподібно зменшуються з масштабом зменшення ε. Типовими прикладами фрактальних пористих структур є множина Кантора (Канторовий пил, рис. 2.18а) з $D_m = \ln 2 / \ln 3 \approx 0.6309$, крива фон Коха (рис. 2.18б) з $D_m = \ln 4 / \ln 3 \approx 1.2619$, серветка Серпіньського (рис. 2.18в) з $D_m = \ln 3 / \ln 2 \approx 1.585$, губка Менгера (рис. 2.18г) з $D_m = \ln 20 / \ln 3 \approx 2.7268$, та багато іншого. У випадку коли D_m дорівнює геометричній розмірності об'єкта (n=0,1,2,3), структура не є фрактальною.

Відповідно до визначення (2.7), чисельні розрахунки фрактальної розмірності проводяться шляхом покриття кривої або зображення квадратами (метод ***box-counting***) з довжиною сторони $a = \varepsilon^{-n}, n = 0,1,2,...$ послідовно. При цьому обличується число квадратів, які містять хоча би одну точку кривої (зображення). На рис. 2.19а наведений приклад розрахунків для кривої фон Коха. Обчислення проводять до тих пір, поки залежність $\ln N_n(\ln \varepsilon_n)$ не наблизиться з достатньою точністю до прямої (рис. 2.19б).

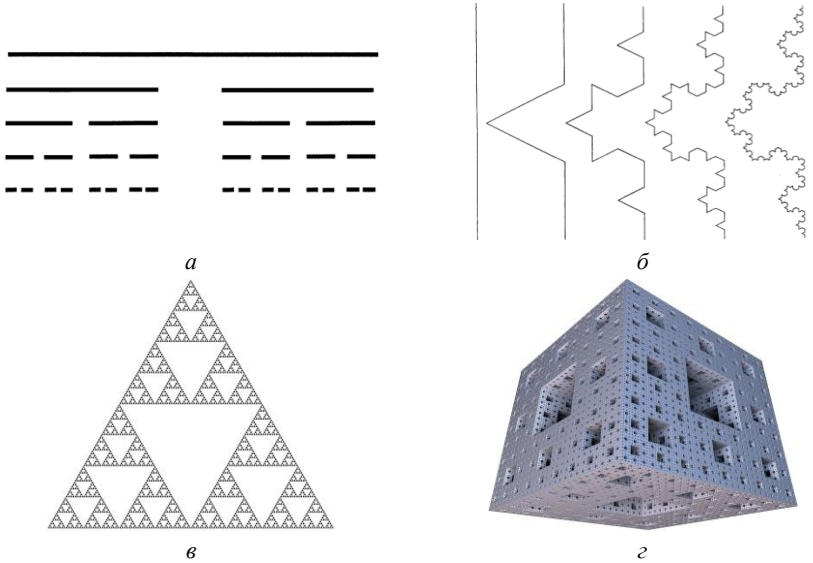
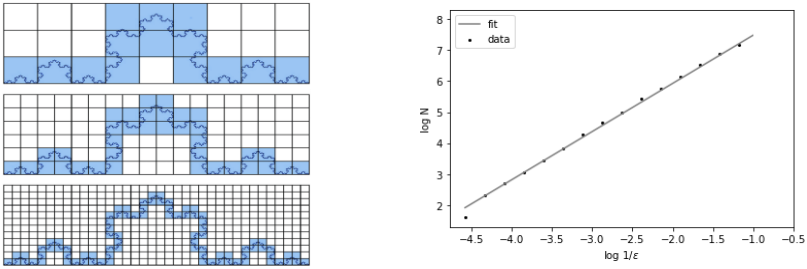


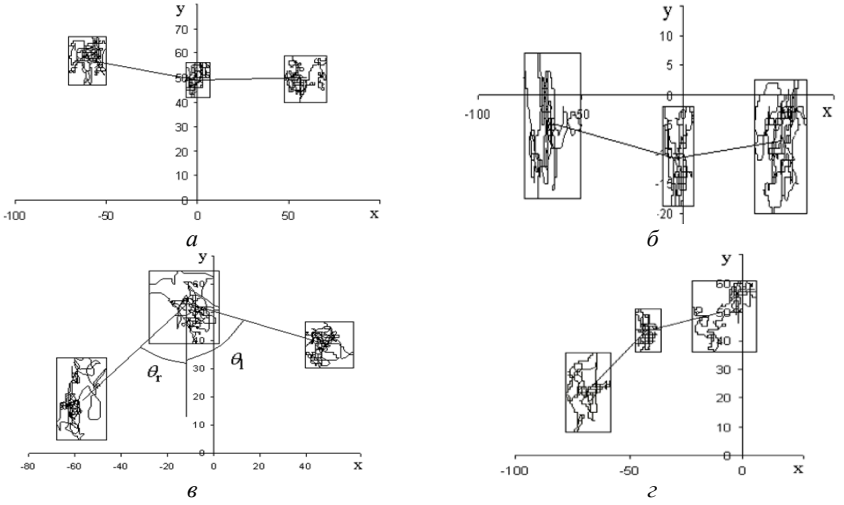
Рис. 2.18. Приклади фрактальних структур: пил Кантора (а); крива Коха (б); серветка Серпінського (г); губка Менгера (д)



а *б*

Рис. 2.19. Процедура методу box-counting (а) і її результат (б)

Таким чином, фрактальний аналіз дозволяє ввести додаткові компоненти до аналізу інформації великих даних. Так, під час вимірювання координат $(x(t), y(t))$ центра мас людини за допомогою стабілографа (Рис. 2.17), шляхом виключення часу із залежностей $x(t)$ і $y(t)$ можна отримати траєкторії $y(x)$ руху проекції центру мас тіла людини на горизонтальну поверхню. Стандартні тести в позиціях «смірно» і «вольно» з переносом вали тіла на ліву і праву ноги, показують, що вигляд траєкторій, їх амплітуди и локація різні у різних випробуваних (рис. 2.20), від симетричних низько– (рис. 2.20а) і високо– амплітудних (рис. 2.20б) коливань до слабко– (рис. 2.20в) і сильно– (рис. 2.20г) асиметричних. Використання фрактальної розмірності дозволяє поповнити набір діагностичних параметрів ще значеннями $D_{\chi 1,2,3}$ для кожного з тестів і, таким чином, забезпечити точнішу диференціальну діагностику захворювань.



а *б* *в* *г*

Рис. 2.20. Результати вимірювань траєкторій $y(x)$ у різних випробуваних: симетричні низько-амплітудні (а); симетричні високо-амплітудні (б); слабко-асиметричні (г); високо-асиметричні (д) динамічні поведінки

2.2.10. Вейвлет-аналіз призначений для виявлення самоподібності кривих, сигналів, структур, зображень і т. п. Формально це є представленням функції $X(t)$ відносно або повної ортонормованої множини деяких базових функцій $\{\phi_j(t)\}_j$, або надповної множини або фрейму векторного простору для простору Гілберта, інтегрованих з квадратом. На відміну від методу Фур'є, $\{\phi_j(t)\}_j$ не є гармонічними функціями, і тому вайвлет-аналіз дозволяє наближати самоподібні стохастичні функції. Наприклад, в якості базової функції можна використати $\phi(t) = (\sin(k\pi t) - \sin(\pi t))/(\pi t)), k > 1$. Симетричні базові функції $\phi(0) > 0$ немонотонно спадають до $\phi(\infty) = 0$, наприклад як найпростіші з вейвлетів – «мексиканську шляпу» (Mexican hat, рис. 2.21а). Тоді можна ввести підпростір функцій

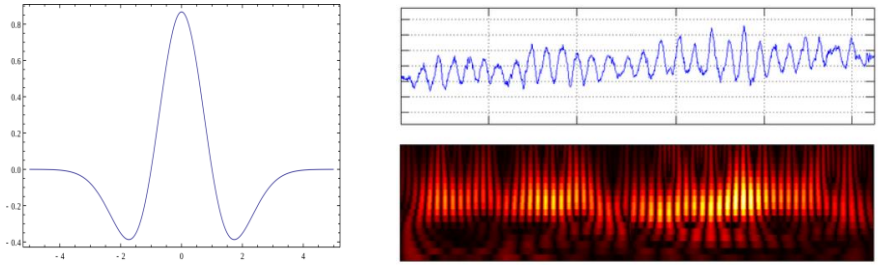
$$\phi_{a,b}(t) = \frac{1}{\sqrt{a}}\phi\left(\frac{t-b}{a}\right), \tag{2.35}$$

де a>0 і b визначають масштаб і зсув вихідного вейвлету $\phi(t)$, тому проекція функції $X(t)$ на підпростір

$$X_a(t) = \int_{\mathbb{R}} X_\phi\{X\}(a,b)\cdot\phi_{a,b}(t)db, \tag{2.36}$$

де $X_\phi\{X\}(a,b) = \int_{\mathbb{R}} X(t)\phi_{a,b}(t)dt$ – вейвлет-коефіцієнти.

Фактично, $[a^{-1}, 2a^{-1}]$ – це є смуга частот, в якій проводиться пошук самоподібності. Результати розрахунків за (2.36) для різних наборів (a,b) у (1.43), формують спектрограми – частотні залежності коефіцієнтів (позначаються кольором) на різних масштабах (рис. 2.21б).



а *б*

Рис. 2.21. Вейвлет Mexican hat (а) і приклад спектрограми сигналу ЕЕГ[18] (б)

[18] Deng S., Srinivasan R., Lappas T. EEG classification of imagined syllable rhythm using Hilbert spectrum methods // J. Neural Eng. 2010. №7(4). 046006.

У випадку дискретних сигналів замість (2.35), (2.36) використовуються їх дискретні аналоги, де замість функцій і інтегрування – масиви точок і скінчені суми. Результати вейвлет-аналізу тих самих координат $(x(t), y(t))$ центра мас людини (Рис. 2.17, 2.20) дозволяють виявити додаткові параметри, а саме міру самоподібності сигналів і її зміни протягом часу експерименту, наприклад, за рахунок підвищеної втомленості, захворювань опорно-рухової, скелетно-м'язової або нервової систем (рис. 2.22).

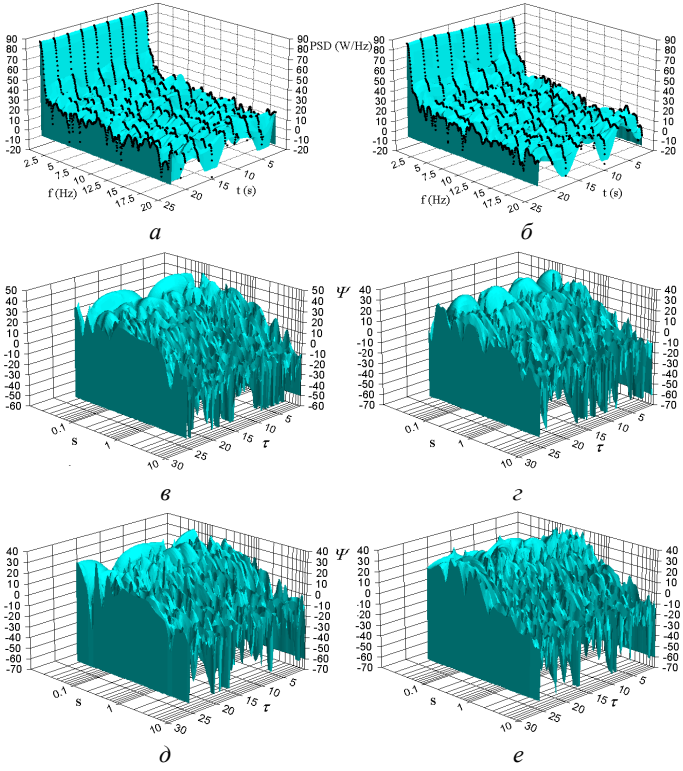


Рис. 2.22. Щільність потужності спектра дискретного вейвлет-перетворення коливань здорових (а, б) та хворих (в, г) випробуваних у фронтальній (а, в) та боковій (д, е) площинах

3. ГОЛОВНІ АНАЛІТИЧНІ МОДЕЛІ НАУКИ О ДАНИХ

Наука про дані (Data Science) – це область, яка пов'язана із збиранням, очищенням (фільтрацією), підготовкою, вирівнюванням та аналізом для отримання інформації з баз даних для вирішення будь-якої проблеми, що фактично є поєднання статистики, математики та програмування.

Один із способів оцінити, як організація в певний час взаємодіє з даними і визначає, де вони придатні для досягнення більш точного огляду, прозріння і передбачення, та його різних підсистем за чотирма основними шляхами (рис. 3.1):

1) описова аналітика;
2) діагностична аналітика;
3) прогнозна аналітика;
4) рекомендаційна аналітика.

Рис. 3.1. Основні типи аналізу даних

Чотири типи аналітики та їх комбінації використовуються разом із вимірюваннями, які засновані на ймовірнісних правилах та за часом (в минулому, сьогоденні та майбутньому). Фокус верхнього лівого та нижнього правого квадрантів, тобто діагностичної та приписної аналітики, є минулим та майбутнім, тоді як фокус нижнього лівого та верхнього правого квадрантів, тобто описової та прогнозної аналітики, – минуле та майбутнє.

Кінцевою метою науки про дані є перетворення необроблених даних у продукти даних, які мають практичне застосування.

Аналіз даних – це наука вивчення вихідних даних з метою прийняття правильних рішень шляхом вироблення значущих висновків. Основні відмінності традиційної аналітики від аналізу великих даних наведені в Табл. 3.1.

Таблиця 3.1. Основні відмінності традиційної аналітики від аналізу великих даних

Концепція	Традиційна аналітика	Аналіз великих даних
Зосереджується на	Описова аналітика Діагностична аналітика	Прогностична аналітика. Наука про дані. Інновації в машинному навчанні.
Набори даних	Обмежені набори даних Менше типів даних Очищені дані Структуровані дані	Необмежені набори даних. Більше типів даних. Необроблені дані. Напівструктуровані та неструктуровані дані.
Моделі даних	Прості	Складні
Архітектура даних	Централізовані бази даних, де складні і великі проблеми вирішуються в єдиній системі	Розподілені бази даних, де складні і великі проблеми вирішуються діленням на багато шматків
Схема зберігання даних	Фіксована/статична	Динамічна

Модель даних – це процес побудови діаграми для глибокого розуміння, організації та зберігання даних для їх обслуговування, доступу та використання. Процес подання даних у графічному форматі допомагає бізнесу та технологічним експертам зрозуміти дані та дізнатися, як ними користуватися.

Продукти даних визначають, який тип профілів був би бажаним для отримання результатів продуктів даних, які збираються в процесі збору даних. Збір даних зазвичай включає в себе збір неструктурованих даних з різних джерел. Як приклад, цей процес може включати отримання необроблених даних або оглядів з веб-сайтів.

Три типи продуктів даних:

• дані, що використовуються для прогнозування;
• дані, що використовуються для рекомендації;
• дані, що використовуються для порівняльної оцінки.

Змінення даних (data munging, data wrangling) – це процес перетворення та відображення даних з одного формату (необроблені дані) в інший формат (бажаний формат) з метою зробити їх більш придатними, простішими та цінними для аналітики. Як тільки повторна перевірка даних буде виконана з будь-якого джерела, наприклад з інтернету, її потрібно зберегти у простому у використанні форматі.

Припустимо, джерело даних надає огляди з точки зору рейтингу в зірках (1–5 зірок); це можна зіставити зі змінною відповіді у вигляді $x \in \{1, 2, 3, 4, 5\}$. Інше джерело даних надає огляди за допомогою системи оцінки великого пальця (вгору або додолу); про це можна зробити висновок із змінною відповіді виду $x \in \{\text{позитивний, негативний}\}$. Для прийняття комбінованого рішення подання першої відповіді джерела даних (п'ятибальна оцінка) має бути перетворено у другу форму (логічна оцінка в два бали), розглядаючи, наприклад, одну та дві зірки як негативні, а три, чотири та п'ять зірки як позитивні. Цей процес часто вимагає більшого розподілу часу, щоб бути забезпеченим якісно.

3.1. Описова аналітика (Descriptive Analytics) – це процес узагальнення історичних даних та виявлення закономірностей та тенденцій. Це дозволяє детально досліджувати відповіді на такі запитання, як «що сталося?», «що зараз відбувається?» Цей тип аналітики використовує історичні дані та дані в режимі реального часу для розуміння того, як наблизитися до майбутнього. Ціль описової аналітики полягає у спостереженні за ознаками/причинами минулого успіху чи невдачі.

Описовий аналіз даних забезпечує наступне:

• інформація про достовірність / невизначеність даних;
• ознаки несподіваних закономірностей;
• оцінки та зведення та організуйте їх у графіки, діаграми та таблиці та
• значні спостереження за формальним аналізом.

Після того, як дані групуються, різні статистичні показники використовуються для аналізу даних та висновків з точки зору

1. міри ймовірності;
2. міри центральної тенденції;
3. міри варіабельності;
4. міри відхилення від нормальності;
5. графічне зображення.

Міра центральної тенденції вказує на центральне значення розподілу, яке включає середнє значення, медіану і моду, однак одного лише центрального значення недостатньо для повного опису розподілу. Нам потрібен показник поширення/розсіювання фактичних даних, крім мір центральності. Стандартне відхилення є більш точним, щоб знайти міру дисперсії. Ступінь дисперсності вимірюється мірами варіабельності, що може змінюватися від одного розподілу до іншого.

Описовий аналіз має важливе значення, оскільки він допомагає визначити нормальність розподілу. Міра відхилення від нормальності включає в себе середньоквадратичне відхилення, перекіс та ексцес.

Міра ймовірності включає в себе стандартну похибку середнього та довірчого обмежень, що використовуються для встановлення обмежень для конкретного ступеня достовірності. ***Статистичні методи*** можна застосовувати для аналізу виведення, щоб зробити висновки і прогнози на основі даних. ***Графічні методи*** використовуються для перекладу числових даних у більш реалістичну та зрозумілу форму.

3.2. Прогностична аналітика (Predictive Analytics) займається моделюванням (екстраполяцією) даних для створення впевнених прогнозів на майбутнє або будь-які невідомі події за допомогою бізнес-прогнозування та моделювання, яке залежить від спостережуваних минулих подій.

Вони стосуються питань «що буде?», «чому це відбудеться?» Прогнозна модель використовує статистичні методи для аналізу поточних та історичних фактів для прогнозування. Прогностична аналітика використовується в актуарній науці, плануванні потенціалу, електронній комерції, фінансових послугах, страхуванні, безпеці інтернету, маркетингу, медичному та медичному обслуговуванні, фармацевтиці, роздрібних продажах, транспорті, електрокомунікаціях, ланцюгах поставок та інших галузях. Бізнес-аналітика включає в себе сегментацію клієнтів, оцінку ризиків, запобігання відтоку, прогнозування продажів, аналіз ринку, фінансове моделювання тощо.

Процес прогнозного моделювання поділяється на три фази: планування, побудова та реалізація.

Планування включає в себе масштаб та підготовку. Для побудови прогнозної моделі потрібно встановити чіткі цілі, очистити та впорядкувати дані, виконати обробку даних, включаючи відсутні значення та виправлення викидів, зробити описовий аналіз даних із статистичним розподілом та створити набори даних, що використовуються для побудови моделі. Це може зайняти близько 40 % загального часу.

Підфазами побудови моделі є моделювання та перевірка. Тут можна написати код моделі, побудувати модель, розрахувати оцінки та перевірити дані. Це та частина, яку можуть залишити науковці з питань даних або технічні аналітики. Це може зайняти близько 20 % загального часу.

Підфазами дії є розгортання, оцінка та моніторинг. Він включає розгортання та застосування моделі, формування рейтингових оцінок для споживачів або продуктів (виявити найпопулярніший продукт), використання результатів моделі для комерційних цілей, оцінку ефективності моделі та моніторинг моделі. Це може зайняти близько 40 % загального часу.

Аналіз настрою (Sentiment analysis) – це найпоширеніша модель прогнозної аналітики. Ця модель приймає простий текст як вхідні дані і надає оцінку настрою як вихідну, що, у свою чергу, визначає, чи є сентимент нейтральним, позитивним чи негативним. Найкращим прикладом прогнозної аналітики є обчислення кредитного балу, який допомагає фінансовим установам, таким як банківський сектор, визначити ймовірність сплати клієнтом рахунків за час.

Іншими моделями проведення прогнозної аналітики є

- аналіз часових рядів;
- економетричний аналіз;
- дерева рішень;
- наївний Байєсівський класифікатор;
- ансамблі;
- підсилення;
- підтримка векторних машин;
- лінійна та логістична регресія;
- штучна нейронна мережа;
- обробка природних мов;
- машинне навчання.

Цікавий приклад аналізу даних наведений у відомому творі «Книга джунглів» Р. Кіплінга[19]. Умовно кажучи, ведмідь Балу найняв аналітика даних (Мауглі), щоб той допоміг знайти їжу. Мауглі мав доступ до великої бази даних, яка складалася з даних про джунглі, дорожню карту, її істот, гори, дерева, кущі, місця меду та події, що відбувалися в джунглях. Мауглі представив Балу детальний звіт, в якому було підсумовано, де він знайшов мед за останні півроку, що допомогло Балу вирішити, куди піти на полювання далі; цей процес називається ***описовою аналітикою***.

Згодом Мауглі підрахував ймовірність знайти мед у певних місцях за певний час, використовуючи передові технології машинного навчання. Це ***прогнозна аналітика***. Далі Мауглі визначив і представив райони, які не придатні для полювання, виявивши першопричину, чому ці місця не підходять. Це ***діагностична аналітика***. Далі, Мауглі дослідив набір можливих дій та запропонував/рекомендував дії для отримання більше меду. Це ***рекомендацій-***

[19] Prabhu C.S.R., et al. Big Data Analytics: Systems, Algorithms, Applications. Springer. 2019.

на аналітика. Крім того, він визначив найкоротші шляхи до джунглів для Балу, щоб мінімізувати його зусилля у пошуку їжі. Це називається ***оптимізацією.***

3.3. Діагностична аналітика (Diagnostic Analytics) – це форма аналітики, яка досліджує дані, щоб відповісти на запитання «чому це сталося?» Це свого роду аналіз корінних причин, який фокусується на процесах та причинах, ключових факторах та невидимих закономірностях.

Етапи діагностичної аналітики:

1. Визначити проблему для аналізу.
2. Провести аналіз, тобто знайти статистично обґрунтовану залежність між двома наборами даних. Це можна зробити за допомогою наступних методів:
• мультирегресія;
• карти, що самоорганізовуються;
• кластерний та факторний аналіз;
• баєсова кластеризація;
• k-найближчі сусіди;
• аналіз основних компонентів;
• графік та аналіз спорідненості.
3. Фільтрування діагнозів: аналітик повинен ідентифікувати 1–2 головні впливові фактори із набору можливих причин.

3.4. Рекомендаційна аналітика (Prescriptive analytics) визначає, які дії вживати, щоб змінити небажані тенденції. Ця аналітика визначається як виведення оптимальних рішень з планування з урахуванням передбачуваного майбутнього та вирішення таких питань, як «що нам робити?» та «чому ми це робитимемо?»

Рекомендаційна аналітика базується на:
• оптимізації, яка допомагає досягти найкращих результатів;
• стохастичній оптимізації, яка допомагає зрозуміти, як визначити невизначеність даних для прийняття кращих рішень та досягнення найкращих результатів.

Рецептивна аналітика – це поєднання даних, бізнес-правил та математичних моделей. Вона використовує оптимізаційні та імітаційні моделі, такі як аналіз чутливості та сценаріїв, лінійне та нелінійне програмування та моделювання Монте-Карло.

3.5. Мережеві науки (Network Science)

У мережевих системах для аналізу даних використовують стандартизовані теоретико-графічні методи. Ці методи мають припущення, що ми точно знаємо різні деталі, такі як взаємозв'язок вузлів між собою. Картографування мережевих топологій може бути дорогим, трудомістким, неточним, а ресурси, які вони вимагають, часто недоступні, оскільки набори даних складаються з мільярдів вузлів і сотень мільярдів посилань у великих децентралізованих

системах, таких як інтернет. Цю проблему можна вирішити, поєднавши методи статистичної фізики та теорії випадкових графів.

Для аналізу великомасштабних мережевих систем використовуються наступні кроки:

1. ***Розрахувати сукупну статистику***, що представляє інтерес: кількість вузлів, щільність зв'язків, розподіл ступенів вузлів, діаметр мережі, найкоротшу довжину шляху між парою вузлів, кореляцію між сусідніми вузлами, коефіцієнт кластеризації, зв'язок, центральність вузла та вплив вузла. У розподілених системах це можна ефективно обчислити та оцінити за допомогою різних методів вибірки.
2. ***Визначити ансамблі статистичної ентропії***. Цього можна досягти за допомогою просторів ймовірностей шляхом присвоєння ймовірностей всім можливим реалізаціям мережі, які узгоджуються із заданою сукупною статистикою, використовуючи аналітичні інструменти, такі як обчислювальні статистичні методи.
3. ***Оцінити передбачувані властивості системи*** на основі сукупної статистики її топології мережі.

3.6. Обчислювальні моделі

Великі дані надзвичайно швидко зростають як мінімум у 4-х вимірах: обсяг, різноманітність, швидкість та достовірність (4V: volume, variety, velocity and veracity) і потребують вдосконалених структур даних, ***нових обчислювальних моделей*** для вилучення деталей та нового алгоритмічного підходу для обчислень.

Обчислювальні моделі залежать від:

1) структури великих даних;
2) інженерії характерних рис (feature engineering);
3) обчислювального алгоритму.

3.7. Структури великих даних

У великих даних потрібні спеціальні структури для обробки величезних наборів даних: хеш-таблиці, поїзди, структури на основі дерев, такі як B-дерева та K-D-дерева, які найкраще підходять для обробки великих даних.

1. ***Хеш-таблиці*** використовують хеш-функцію для обчислення індексу та відображення ключів до значень. ***Імовірнісні структури даних*** відіграють важливу роль у наближеному впровадженні алгоритму у великі дані. Ці структури даних використовують хеш-функції для рандомізації елементів та підтримки набору операцій, таких як об'єднання та перетин, і тому їх можна легко розпаралелювати.

Приклади імовірнісних структур даних: (1) запит на членство – фільтр Query–Bloom, (2) HyperLogLog, (3) count-min sketch, (4) MinHash.

Фільтр Блума, який був запропонований Б. Г. Блумом у 1970 р., є просторово ефективною імовірнісною структурою даних, що дозволяє зменшити кількість точних перевірок того, чи є елемент «членом» чи «не членом» групи. Тут запит повертає ймовірність із результатом «можливо, встановлено» або «точно, не встановлено». На відміну від хеш-таблиць і інших структур даних, фільтр Блума представляє універсальний набір всіх можливих елементів. В цьому випадку всі біти в його бітовому масиві рівні одиниці. Фільтр може давати помилково позитивні значення, але не може давати помилково негативні. Приклад схеми роботи фільтру Блума з 5 хеш-функціями наведений на Рис. 3.2.

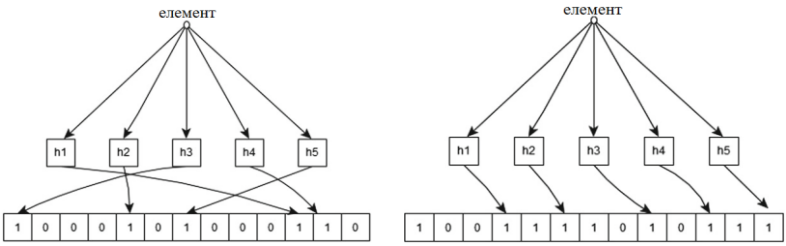


Рис. 3.2. Схема роботи фільтру Блума

Бітовий вектор – це базова структура даних для фільтра Блума. Кожен порожній фільтр Bloom являє собою бітовий масив із «m» бітів і спочатку є пустим. Коли елемент додається до фільтра, він хешується *k* функціями h_1, h_2 ... h_k за модулем m, в результаті чого в бітовий масив входять індекси *k*, яки приймають значення 0 або 1, відповідно до значень хеш-функцій h_1, h_2 h_3, h_4, h_5. Щоб перевірити приналежність елемента, ми знову хешуємо елемент з тими ж функціями хешування і перевіряємо, чи встановлений кожен відповідний біт. Імовірність помилково позитивного визначення можна обчислити як

$$P = \left(1 - \left(1 - m^{-1}\right)^{kn}\right)^k, \tag{3.1}$$

де m – розмір бітового масиву, k – кількість хеш-функцій, n – кількість очікуваних елементів. Розмір бітового масиву m можна обчислити як $m = n \ln P / (\ln 2)^2$, а оптимальну кількість хеш-функцій – як $k = m \ln 2 / n$.

Алгоритм HyperLogLog (HLL) – є розширенням алгоритму LogLog, похідного від алгоритму Флайоле–Мартіна (1984). Це імовірнісна структура даних, яка використовується для оцінки потужності набору даних для розв'язання задач проблеми шляхом апроксимації кількості унікальних елементів у мультимножині. Для обчислення точної потужності мультимножини потрібен обсяг пам'яті, пропорційний до потужності, що не є практичним для величезних наборів даних. Алгоритм HLL використовує значно менше пам'яті за рахунок отримання лише апроксимації потужності. Як випливає з назви, HLL вимагає пам'яті ~*O(log2log2n)*, де *n* – потужність набору даних.

Алгоритм HyperLogLog використовується для оцінки кількості унікальних елементів у списку. Припустимо, що веб-сторінка має мільярди користувачів і ми хочемо обчислити кількість унікальних відвідувань нашої веб-сторінки. Наївним підходом було б зберігати кожен відмінний ідентифікатор користувача в наборі, і тоді розмір набору буде враховуватися за величиною. Коли ми маємо справу з величезними обсягами наборів даних, підрахунок потужності цим способом буде неефективним, оскільки набір даних займе багато пам'яті. Але якщо нам не потрібна точна кількість окремих відвідувань, тоді ми можемо використовувати HLL, оскільки він був розроблений для оцінки кількості мільярдів унікальних значень.

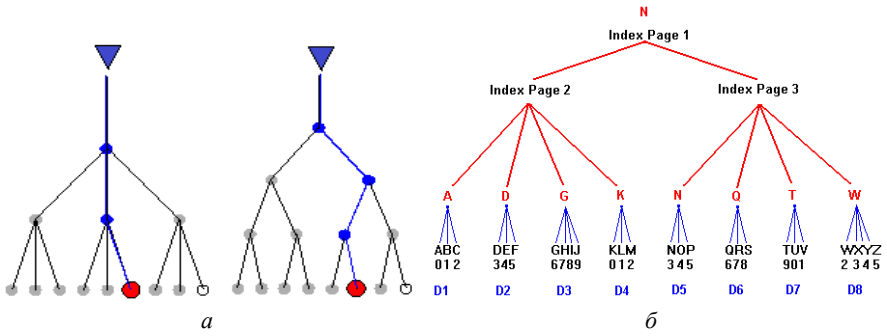
```
HyperLogLog
  def add(cookie_id: String): Unit
  def cardinality( ):
    //|A|
  def merge(other: HyperLogLog):
    //|A ∪ B|
  def intersect(other: HyperLogLog):
    //|A ∩ B| = |A| + |B| − |A ∪ B|
```

Рис.3.3. Приклад коду для 4-х основних операцій HLL

Чотири основні операції HLL (Рис. 3.3):

1. Додати новий елемент до множини.
2. Розрахувати міцність (cardinality) множини.
3. Об'єднати дві множини.
4. Розрахувати міцність перетину множин.

B-дерева – це ефективна структура даних для зберігання великих даних та швидкого пошуку, яка була запропонована Р. Баєром і Є. М. МакКрейтом у 1970 році. У бінарному дереві кожен вузол має щонайбільше двох дітей, і складність часу для виконання будь-якої операції пошуку становить ~O(log2N). B-дерево – це узагальнення бінарного дерева на випадок m дочерних вузлів у кожного батьківського вузла (*m* – коефіцієнт розгалуження). Через достатньо великий коефіцієнт розгалуження B-дерево є однією з найшвидших структур даних (Рис. 3.4). Під час пошуку даних в БД зі структурою



а *б*

Рис. 3.4. Приклади з постійними (а) і змінними вздовж дерева (б) коефіцієнтами розгалужень

B-дерева ключі пошуку зберігаються у відсортованому порядку для послідовного обходу, використовується ієрархічний індекс для мінімізації кількості операцій читання з диска, для прискорення вставки і видалення використову-

ються частково повні блоки, за допомогою рекурсивного алгоритму індекс пошуку зберігається збалансованим.

Для прискорення пошуку в багатовимірному просторі А. Гутманом в 1982 р. була запропонована динамічна структура даних у вигляді R-дерева.

Feature Engineering – це компонента обробки даних, де аналізуються та вибираються необхідні властивості, риси (features). Витягти та вибрати потрібні дані для аналізу з величезного набору даних життєво важливо і складно.

Побудова об'єкта – це процес, який виявляє відсутні дані про зв'язки між об'єктами та розширює простір об'єктів шляхом генерації додаткових функцій, корисних для прогнозування та кластеризації. Він передбачає автоматичне перетворення заданого набору вихідних функцій для створення нового набору потужних функцій за допомогою виявлення прихованих закономірностей, що сприяє кращому досягненню поліпшень точності та зрозумілості.

Вилучення об'єктів використовує функціональне відображення для вилучення набору нових функцій із існуючих об'єктів. Це процес перетворення оригінальних об'єктів у простір нижчого виміру. Кінцевою метою процесу вилучення ознак є пошук найменшого набору нових функцій шляхом певної трансформації на основі показників ефективності. Існує кілька алгоритмів для вилучення функцій. Підхід нейромережевої мережі прямого зв'язку та аналіз основних компонентів (PCA) відіграють життєво важливу роль у вилученні функцій, замінюючи вихідні атрибути «n» іншими наборами «m» нових функцій.

Вибір ознак – це етап попередньої обробки даних, який вибирає підмножину об'єктів із існуючих оригінальних об'єктів без перетворення для завдань класифікації та аналізу даних. Це процес вибору підмножини ознак «m» із вихідного набору ознак «n», де $m \leq n$. Роль вибору ознак полягає в оптимізації точності прогнозування та прискоренні процесу вивчення результатів алгоритму за рахунок зменшення простору ознак.

Feature Learning – репрезентативне навчання або вивчення ознак – це набір прийомів, які автоматично трансформують вхідні дані до формулювань, необхідних для виявлення або класифікації ознак. Це усуває громіздку ручну інженерію функцій, дозволяючи машині одночасно навчатися та використовувати функції для виконання конкретних завдань машинного навчання. Таке навчання може бути як контрольоване, так і неконтрольоване.

Контрольовані навчання – з позначеними вхідними даними, і являють собою процес прогнозування вихідної змінної (Y) із вхідних змінних (X) за допомогою відповідного алгоритму для вивчення функції відображення від вхідного до вихідного; приклади включають контрольовані нейронні мережі, багатошаровий персептрон та контрольоване вивчення словників.

Під час ***неконтрольованого навчання*** ознаки вивчаються з неміченими вхідними даними і використовуються для пошуку прихованої структури в даних; приклади включають у себе вивчення словників, незалежний аналіз компонентів, автокодери, факторизацію матриць та інші форми кластеризації.

Ансамблеве навчання – це модель машинного навчання, де одна і та ж проблема може бути вирішена шляхом навчання кількох мереж. Ансамблева модель навчання намагається побудувати набір гіпотез та об'єднати їх для використання, що значно підвищує точність прогнозних і класифікаційних моделей. Найчастіші техніки ансамблевого навчання – Bagging, Boosting, Stacking.

3.8. Обчислювальні алгоритми

Підприємства дедалі більше покладаються на аналіз своєї великої кількості даних, щоб прогнозувати реакцію споживачів та рекомендувати їм свої товари. Для аналізу таких даних було розроблено ряд алгоритмів. Класифікація, регресія та низка подібності є основними принципами, на яких багато алгоритмів використовуються в прикладній науці даних. Найчастіше використовуваними ***алгоритмами*** є:

- кластеризація K-середніх значень;
- майнінг правил асоціації;
- лінійна регресія;
- логістична регресія;
- C4,5 – алгоритм для побудови дерев рішень, розроблений Д. Квінланом;
- підтримка векторної машини (SVM) ;
- апріорний;
- Максимізація очікування (EM) ;
- AdaBoost;
- наївний баєсівський.

3.9. Моделі програмування *(Programming Models)* – це абстракція над існуючим обладнанням або інфраструктурою. Це модель обчислень для ефективного написання комп'ютерних програм на розподілених файлових системах, що використовують великі дані, і легко впорається з усіма потенційними проблемами за допомогою набору абстрактних бібліотек виконання та мов програмування.

Необхідні вимоги до моделей програмування великих даних включають у себе:

1. Підтримку операцій з великими даними.
 1.1. Розділення обсягів даних.
 1.2. Швидкий доступ до даних.
 1.3. Розподілення обчислень по вузлах.
 1.4. Комбінування після закінчення.

2. Обробку стійкості до несправностей.
2.1. Реплікацію розділів даних.
2.2. Відновлення файлів за потреби.
3. Увімкнення масштабування шляхом додавання полиць (racks).
4. Оптимізацію для певних типів даних, таких як документ, графік, таблиця, значення ключів, потоки та мультимедіа.

Приклади: MapReduce, передача повідомлень, спрямований ациклічний графік, робочий процес, об'ємна синхронна паралель та подібність до SQL – це стандартні моделі програмування для аналізу великих наборів даних.

4. МЕТОДИ Й АЛГОРИТМИ ШТУЧНОГО ІНТЕЛЕКТУ В АНАЛІЗІ «ВЕЛИКИХ ДАНИХ»

4.1. Складові компоненти, методи та алгоритми штучного інтелекту

Штучний інтелект (ШІ) – це розділ математичних наук, який займається моделюванням інтелекту людей і тварин у машинах, які запрограмовані думати, як живі істоти, і імітувати їх дії. Цей термін також може застосовуватися до будь-якої машини, яка проявляє риси людського розуму, такі як навчання і рішення проблем. Ідеальною характеристикою ШІ є його здатність націоналізувати і робити дії, які мають найбільші шанси на досягнення конкретної мети.

Відповідно до ***теореми Теслера,*** ШІ – це те, що ще не було зроблено.

У розмовній мові термін ШІ часто використовується для опису машин (або комп'ютерів), які імітують «когнітивні» функції, які люди пов'язують з людським розумом, такі як розпізнавання, навчання, прийняття рішень і розв'язання задач. В підручниках з ШІ часто визначають цю область як дослідження «***інтелектуальних агентів***»: будь-якого пристрою, який сприймає навколишнє середовище і вживає заходів, які збільшують його шанси на успішне досягнення своїх цілей.

Компонентами ШІ є машинне навчання (Machine learning), яке належить до концепції, згідно з якою комп'ютерні програми можуть автоматично вчитися й адаптуватися до нових даних без допомоги людини. Методи глибокого навчання (Deep learning) забезпечують автоматичне навчання за рахунок поглинання величезних обсягів неструктурованих даних, таких як текст, зображення або відео.

Сучасні методи ШІ використовують міждисциплінарний підхід, засований на математиці, інформатиці, лінгвістиці, біології, філософії, психології та інших науках. Застосовують також деякі технології прийняття рішень, відомі з живої природи, наприклад розум бджіл (swarm intelligence), логіка мурахів (ant's logic), розум слизовиків (slime mould).

ШІ заснований на принципі, що людський інтелект можна визначити таким чином, щоб машина могла імітувати його і виконувати завдання, від найпростіших до більш складних. Цілі ШІ включають у себе імітацію когнітивної діяльності людини.

У міру розвитку технологій попередні тести, які визначали ШІ, застарівають. Наприклад, машини, які обчислюють базові функції або розпізнають текст за допомогою оптичного розпізнавання символів, більше не вважаються втіленням ШІ, оскільки ця функція тепер сприймається як невід'ємна функція комп'ютера.

Сучасні можливості машин, які класифікуються як ШІ, включають в себе: розуміння людської мови, конкурування на найвищому рівні в стратегічних ігрових системах (таких як шахи, Го), недосконало-інформаційні ігри (покер),

інтелектуальна маршрутизація в тематичній мережі доставки, самостійне водіння автомобіля, військові моделювання, медичні рішення.

Існують наступні ***різновиди ШІ***:

1) ***«слабкий» ШІ*** зазвичай простий і орієнтований на вирішення однієї задачі;

2) ***«сильний» ШІ*** (Artificial General Intelligence AGI, загальний штучний інтелект) виконує більш складні завдання, подібно до інтелекту людини;

3) ***штучний біологічний інтелект*** (Artificial Bioligical Intelligence, ABI) – це спроби імітувати «природний» інтелект.

Основні підходи ШІ:

1) ***символізм*** (формальна логіка), наприклад: «Якщо у здорової дорослої людини жар, то вона, можливо, хвора на грип»;

2) ***баєсівська логіка*** (див. с. 30): «Якщо у пацієнта жар, скоректуйте вірогідність того, що у нього грип»;

3) ***аналогізатори*** (популярні у рутинних бізнес-додатках): «Після вивчення записів хворих з жаром виявилося, що a % цих пацієнтів мають грип»;

4) ***принципи роботи нейронів мозку***, які можуть навчатися, наприклад, порівнюючи отриманий результат з бажаним, змінюючи силу зв'язків між сусідніми нейронами, щоб укріпити ті зв'язки між ними, які виявилися корисними.

Ці основні підходи можуть пересікатися з іншими. Наприклад, нейронні мережі можуть навчитися робити висновки, узагальнювати та проводити аналогії. Алгоритми навчання працюють, тому що стратегії, алгоритми та висновки, які добре працювали в минулому, ймовірно, продовжували добре працювати і в майбутньому (сонце сходить кожний ранок протягом останніх 10 000 днів, тому, ймовірно, воно зійде і завтра вранці). Вони можуть мати нюанси, наприклад, «a % сімей лебедів мають географічно окремі види з різними кольоровими варіантами, тому є b % ймовірності, що існують невідкриті чорні лебеді».

Відповідно до ***принципу лези бритви Оккама***, «найпростіша теорія інтерпретації даних є найбільш вірогідною» і, таким чином, ШІ повинен бути спроєктований таким чином, щоб він віддав перевагу більш простим теоріям і уникав випадків, коли складна теорія краще відповідає результатам обробки даних. Використання занадто складної теорії, придатної для відповідності всім минулим навчальним даним, призводить до ***перенавчання ШІ***. Багато систем намагаються зменшити перенавчання за рахунок визнання теорії відповідно до тих тем, де вона добре відповідає даним, але не обирає теорію відповідно до її складності. Крім класичного перенавчання, яке також може розчарувати, є «засвоєння неправильного уроку», наприклад: класифікатор зображень, навчений лише зображень коричневих коней і чорних кішок, може зробити висновок, що всі коричневі плями на зображенні, скоріше за все, належать коням, а чорні – кішкам.

Порівняно з людиною ШІ має недоліки. Зокрема, у людей є потужні механізми розпізнавання інформації про «наївну фізику», що дозволяє навіть маленьким дітям легко робити висновки типу «якщо я штовхну цій предмет зі столу, то він впаде на підлогу».

4.2. Історія розвитку концепції, методів і алгоритмів ШІ

Розуміння того, що цифрові комп'ютери можуть моделювати будь-який процес формального міркування, відоме як теза Черча–Тьюринга. Поряд з одночасними відкриттями в нейробіології, теорії інформації і кібернетики, це спонукало дослідників розглянути можливість створення електронного мозку. Тьюринг запропонував змінити питання з того, «чи є машина розумною», на «чи може машина демонструвати розумну поведінку».

У ***1943 р.*** американський нейрофізіолог і кібернетик У. МакКуллох у співпраці з В. Піттсом запропонували концепцію повних за Тьюрингом «штучних нейронів», яка вважається першою опублікованою працею[20] в галузі ШІ. В роботі були створені обчислювальні моделі на основі математичних алгоритмів, званих ***пороговою логікою*** (threshold logic), що по суті стало першою математичною моделлю нейронної мережі.

У ***1954 р.*** комп'ютери вивчають стратегії гри в шашки і до ***1959 р***., як повідомлялося, грали краще, ніж середня людина. Комп'ютери розв'язували задачі з алгебри, доводили теореми з логіки («Logic Theorist», перший запуск у ***1956 р***.) і розмовляли англійською мовою.

У ***1955 р***. ШІ був введений як академічна дисципліна в вузах.

До ***середини 1960-х*** дослідження в США значною мірою фінансувалися Міністерством оборони США, і лабораторії були створені по всьому світу. Засновники ШІ з оптимізмом дивилися у майбутнє: Г. Саймон передбачав, що «протягом двадцяти років машини будуть здатні виконувати будь-яку роботу, яку може виконувати людина». М. Мінський погодився з тим, написавши: «Протягом одного покоління ... проблема створення штучного інтелекту буде істотно вирішена». Але прогрес сповільнився, і в ***1974*** р. у відповідь на критику видатного математика Дж. Лайтхілла і триваючий тиск Конгресу США з метою фінансування більш продуктивних проектів уряди США і Великої Британії припинили дослідні дослідження в галузі ШІ. Почалось розчарування і втрата фінансування («зима штучного інтелекту»).

На початку ***1980***-х почався комерційний успіх експертних систем, заснованих на ШІ. У ***1985*** р. ринок ШІ вже перевищив 10^6. У той же час вдалий комп'ютерний проект 5-го покоління в Японії надихнув уряди США і Велико британії на відновлення фінансування наукових розробок. Однак, починаючи

[20] McCulloch W., Pitts W. A Logical Calculus of Ideas Immanent in Nervous Activity. Bull. Math. Biophys. 1943. № 5(4). P. 115-133.

з краху ринку машин Lisp в ***1987*** р., ШІ знову втратив репутацію, і почався другий, більш тривалий період «зими».

Розвиток великомасштабної інтеграції метал-оксид-напівпровідників в формі технології транзисторів дозволив розробити практичну технологію штучних нейронних мереж (***1980***-ті рр.). Віхою в цій області була публікація в ***1989*** р. книги «Analog VLSI – Реалізація нейронних систем Міда і Ісмаіла.

Наприкінці ***1990***-х рр. і на початку XXI ст. ШІ почав використовуватися для логістики, інтелектуального аналізу даних, медичної діагностики та в інших галузях. Успіх був обумовлений збільшенням обчислювальної потужності, підвищеною увагою до вирішення конкретних проблем, новими зв'язками між ШІ та іншими сферами, такими як статистика, економіка та математика, а також прагненням дослідників до математичних методів і наукових стандартів.

У ***1997 р.*** Deep Blue стала першою комп'ютерною шаховою системою, яка перемогла чинного чемпіона світу з шахів Гаррі Каспарова.

У ***2011 р.*** у вікторині «Jeopardy! quiz show» (питання – відповідь) система ШІ відповідей на питання «Watson» фірми IBM обіграла двох найбільших Jeopardy! –чемпіонів зі значним відривом.

У ***2012*** р. швидші комп'ютери, алгоритмічні поліпшення і доступ до великих обсягів даних дозволили домогтися прогресу в галузі машинного навчання і сприйняття.

У ***2014*** р. фірма «Kinect», яка забезпечує 3D–інтерфейс руху тіла для «Xbox 360» і «Xbox One», використала алгоритми, що з'явилися в результаті тривалих досліджень ШІ, як і інтелектуальні персональні помічники в смартфонах.

У ***2015*** р. заснована на ШІ AlphaGo успішно перемогла професійного гравця в Го, що привело до відновлення фінансування і новим успіхам ШІ. У ***2016*** р. AlphaGo виграла 4 з 5 ігор в Го у поєдинку з чемпіоном з Го Лі Сідолом, ставши першою комп'ютерною системою «Go-play», яка обіграла професійного гравця Go. На саміті «Future of Go» у ***2017*** р. «AlphaGo» виграла матч із 3 іграми з Ке Цзе, який на той час постійно утримував світовий рейтинг № 1 протягом двох років. Мюррей Кемпбелл із «Deep Blue» назвав перемогу AlphaGo «кінцем ери ... настільних ігор, і настав час рухатись далі». Відомо, що Го – це відносно більш складна гра, ніж шахи. Пізніше «AlphaGo» був вдосконалений, узагальнений для інших ігор, таких як шахи, за допомогою «AlphaZero» та «MuZero», щоб грати в багато різних відеоігор, які раніше оброблялись окремо, на додаток до настільних ігор. Інші програми обробляють ігри з недосконалою інформацією; наприклад, для покеру на рівні людини це «Плурібус» та «Цефей» (покерні боти).

Таким чином, ***2015*** р. став знаковим для ШІ, бо кількість програмних проектів, які використовують ШІ в Google, зросла з «епізодичного використання» у 2012 р. до понад 2700 проектів. Алгоритми ШІ були значно вдосконалені, що підтверджується нижчим рівнем помилок у завданнях обробки зображень. Через зростання інфраструктури хмарних обчислень і збільшення дослідницьких інструментів та наборів даних кількість доступних нейронних

мереж збільшувалась. Також корпорація «Майкрософт» розробляла системи Skype, яка може автоматично перекладати з однієї мови іншою, та системи Facebook, яка може описувати зображення сліпим людям.

У ***2016*** р. Китай значно прискорив своє державне фінансування; з огляду на великий обсяг даних та швидко зростаючий обсяг наукових досліджень, деякі спостерігачі вважають, що КНР може бути на шляху до того, щоб стати «наддержавою ШІ».

У ***2017*** р. під час опитування кожна п'ята компанія повідомила, що «включила ШІ в деякі пропозиції або процеси».

У ***2020*** р. «Natural Language Processing systems» (системи обробки природних мов), такі як GPT-3 (тоді це була найбільша штучна нейронна мережа), відповідали людським показникам за вже існуючими тестами, хоча і без того, щоб система досягла здорового розуміння змісту тестів.

«AlphaFold 2 DeepMind» продемонстрував здатність визначати за допомогою ШІ 3D-структуру білка за години, а не за місяці.

Досягнутий рівень розпізнавання обличчя з точністю 99 %.

Протягом більшої частини своєї історії дослідження ШІ були розділені на підгалузі, які часто не взаємодіють між собою. Ці підгалузі засновані на технічних поняттях, таких як конкретні цілі (наприклад, «***робототехніка***» або «***машинне навчання***»), використання певних інструментів (логіка або штучні нейронні мережі), філософські відмінності і принципи, а також соціальні фактори (певні установи або конкретні дослідники).

Традиційні ***цілі ШІ*** – міркування, представлення знань, планування, навчання, обробка природної мови, сприйняття і здатність переміщати об'єкти і маніпулювати ними. Загальна розвідка – одна з довгострокових цілей ШІ.

Підходи ШІ – статистичні методи, обчислювальний інтелект, традиційний символічний ШІ.

Інструменті ШІ – математична логіка, алгоритми пошуку, штучні нейронні мережі, математична оптимізація, теорія ймовірності й ін.

4.3. Машинне навчання

Машинне навчання (Machine Learning) – це вивчення комп'ютерних алгоритмів, які автоматично вдосконалюються завдяки досвіду (фундаментальна концепція досліджень ШІ).

Збільшення потоку даних, що надходять з різних джерел, створило величезну кількість ресурсів, однак використання цих ресурсів для будь-якого корисного завдання вимагає глибокого розуміння характеристик даних. Мета алгоритмів ***машинного навчання*** – вивчити ці характеристики та використовувати їх для прогнозування в майбутньому. Однак у контексті великих даних застосування алгоритмів машинного навчання покладається на ефективні методи обробки даних, такі як використання паралельності даних під час роботи з величезними базами даних. Отже, методології машинного навчання

все частіше стають статистичними та менш заснованими на правилах для обробки такого масштабу даних.

Статистичні підходи до машинного навчання залежать від трьох важливих послідовностей завдань:

– наявність великої колекції даних, що показують, як люди виконують те чи інше завдання;

– розробка моделі для вивчення зібраних даних;

– визначення параметрів моделі на базі вивчених даних.

Машинне навчання не є єдиним типом алгоритмів. Це спостерігається з більшої точки зору, де важливим є розуміння даних для забезпечення аналізу даних, підготовки та прийняття рішень. Крім того, ідеї з базової математики, статистики та експертизи використовуються в машинному навчанні для розв'язання задач великих даних. Машинне навчання, як правило, поєднує або використовує інформацію з різних галузей для розпізнавання зразків, аналізу даних та знань. Основною метою машинного навчання є прогнозування, виходячи за рамки єдиного відкриття закономірностей у даних. Під час розробки алгоритмів машинного навчання дається припущення про базовий розподіл даних.

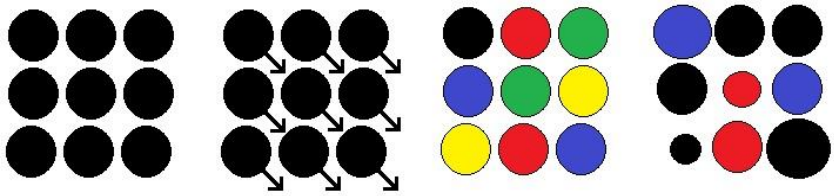
Машинне навчання також стикається з проблемами використання великих даних (рис. 4.1):

• навчання великим масштабам даних (volume);

• навчання потокового передавання даних (velocity);

• навчання для різних типів даних (variety);

• навчання неповним даним (versatility).



а б в г

Рис. 4.1. Ілюстрація проблем ШІ у «великих даних»: розмір (а), швидкість (б), різноманітність (в), неповнота, невпевненість (г)

У машинному навчанні використовуються генеративні або дискримінаційні алгоритми. Наприклад, є задача класифікації об'єктів, яку можна розв'язати як із генеративним, так і з дискримінаційним підходом. Припустимо, існує два класи фруктів – яблука ($y = 1$) і апельсини ($y = 0$), а x – особливість фруктів. З огляду на дані навчання, ***дискримінаційний підхід*** знайде межу прийняття рішення, яке розділяє яблука та апельсини. Крім того, щоб класифікувати новий плід як яблуко або апельсин, він перевіряє, на яку сторону межі рішення він потрапляє, і робить прогноз. Іншого підходу дотримується ***генеративний підхід***. Спочатку, розглядаючи яблука, ми будуємо модель того, як виглядають

яблука. Потім, розглядаючи апельсини, ми можемо побудувати окрему модель того, як виглядають апельсини. Нарешті, щоб класифікувати новий фрукт, ми поєднуємо новий фрукт із моделлю яблук та апельсинів, щоб з'ясувати, чи є цей фрукт більше схожим на яблука чи апельсини, що раніше спостерігалося в даних тренувань.

На практиці генеративна модель має вищу ***асимптотичну похибку***, ніж дискримінаційна модель. Однак спостерігається, що генеративна модель наближається до своєї асимптотичної похибки швидше, ніж дискримінаційна модель, можливо, саме з навчальними зразками, які мають лише логарифмічне, а не лінійне число параметрів. Зазвичай можна припустити, що дискримінаційна модель може бути правильною, навіть коли генеративна модель є неправильною, але не навпаки.

Існують різні ***генеративні моделі***:
– наївний Баєс;
– суміші експертів;
– баєсівські мережі;
– приховані моделі Маркова (HMM);
– випадкові поля Маркова (MRF).

Так само існує багато ***дискримінаційних моделей***:
– логістична регресія SVM;
– умовно випадкове поле (CRF);
– нейронні мережі.

Як правило, дискримінаційний та генеративний алгоритми утворюють пари для вирішення певного завдання. Наприклад, «Naive-Bayes» та логістична регресія є відповідною парою для класифікації, тоді як HMM та CRF є відповідною парою для вивчення послідовних даних. З точки зору пояснювання, генеративні моделі мають перевагу над дискримінаційними моделями.

Алгоритми машинного навчання для великих даних поділяються на декілька галузей. Однією з тих галузей, яка широко застосовується, є контрольоване машинне навчання. Загалом контрольоване навчання працює з двома різними змінними, в основному вхідними навчальними даними (x) та вихідними мітками (y), де алгоритм навчання вивчає функцію відображення від входу (x) до результату (y), тобто $y = f(x)$. Мета цієї функції відображення – передбачити вихідну мітку y^* для нових вхідних даних.

Цей підхід називають ***контрольованим навчанням***, оскільки процес вивчення алгоритму на основі навчальних даних можна розглядати як нагляд за навчальним процесом (рис. 4.2). Оскільки справжні мітки відомі раніше, прогнози алгоритмів коригуються шляхом порівняння з основною істиною. Навчання припиняється, коли алгоритм досягає задовільних показників. Керовані алгоритми навчання з ширшої точки зору можна класифікувати як класифікаційні, так і регресійні методи. Метою класифікаційних алгоритмів є класифікація вхідних даних (x) на дискретні класи (y), тоді як методи регресії моделюють незалежні змінні (x) для прогнозування реальної залежної змінної (y).

Приклад представлення ***дерева рішень*** для прийняття рішення: чи клієнт придбає ноутбук? За наявності персональних даних додаємо вік та інформацію про кредит. Дерево рішень базується на індуктивному висновку, де проводиться багато спостережень, щоб зрозуміти закономірність, вивести пояснення та алгоритм прийняття рішень. Навчання досягається апроксимацією дискретних, реальних чи відсутніх атрибутів для побудови дерева. Приклад дерева рішень наведений на рис. 4.3. В основі може використатися ***логістична регресія*** – це прогнозуючий алгоритм, метою якого є побудова моделі, яка знаходить зв'язок між залежними та незалежними змінними. Взагалі, залежна змінна є двійковою (0 або 1), а незалежні змінні можуть бути номінальними, порядковими тощо. Хоча коефіцієнти, що генеруються в логістичній регресії, оцінюються за рівнянням, яке включає трансформацію ймовірностей щодо ознак, отриманих із навчальних даних.

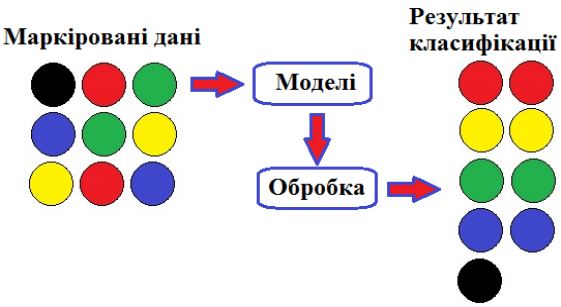


Рис. 4.2. Графічна модель контрольованого машинного навчання

Таким чином, алгоритми машинного навчання класифікуються як

1. ***Навчання без вчителя*** – це здатність виявляти закономірності у потоках вхідних даних, яка не вимагає попереднього маркування даних.
2. ***Контрольоване навчання*** включає в себе як класифікацію, так і числову регресію.

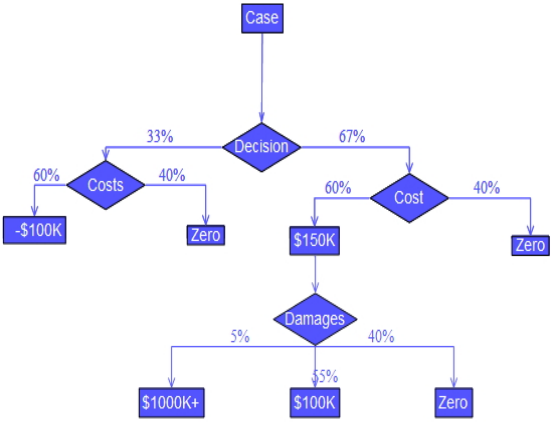


Рис. 4.3. Дерево рішень для оцінки ймовірності купівлі товару із заданою ціною.

Така ***класифікація*** використовується, щоб визначити, до якої категорії входить набір даних і що відбувається після того, як програма бачить низку прикладів з кількох категорій. Машинне навчання використовує регресійні залежності, тобто математичні функції, які описують взаємозв'язок між входами і виходами та передбачають, як виходи повинні змінюватися залежно від змін вхідних даних. ***Класифікатори*** в регресії можуть розглядатися як «апроксиматорні функції», які намагаються визначити невідому (можливо,

неявну) функцію залежності. Наприклад, класифікатор спаму можна розглянути як навчальну функцію, яка співставляє текст електронного листа з однією з двох категорій: «спам» або «не спам». Навчання може оцінювати учасників за обчислювальною складністю, за складністю вибору або за іншими критеріями оптимізації. У «***навчанні з підкріпленням***» агент нагороджується за хороші відповіді та карається за помилкові. Далі агент використовує цю послідовність знань, щоб сформувати свою стратегію подальших дій.

Використання в робототехніці. Удосконалені роботи-маніпулятори та інші промислові роботи, широко використовувані в сучасних закладах, можуть на власному досвіді навчитися ефективно рухатись, не маючи наявності тренінгу та прослизання шестерень. Сучасний мобільний робот у невеликому, статичному та видимому середовищі може легко визначити своє місце розташування та навести карту свого оточення; однак динамічне середовище, таке як (в ндоскопії) внутрішня частина тіла пацієнта, яке постійно рухається за рахунок скорочень серця і м'язів, дихання тощо, представляє велику проблему для робота-хірурга або діагноста.

Існує відомий ***Парадокс Моравека*** (1988 р.), що «порівняно легко змусити комп'ютери шукати результати на рівні дорослих в інтелектуальних тестах або іграх в шашки, і важко або неможливо давати їм навички однорічної дитини». Це твердження означає, що всупереч поширеній думці висококогнітівні процеси вимагають відносно невеликих обчислень, в той час як низькорівневі сенсомоторні операції вимагають великих обчислювальних ресурсів і їх нелегко запрограмувати в роботі ШІ. Це пов'язано, наприклад, з тим, що, на відміну від шах або гри в ГО, на фізичні здібності людини безпосередньо впливав еволюційний відбір протягом мільйонів років.

4.4. Розум бджолиного рою

Swarm intelligence (SI, розум рою) – це концепція колективної поведінки в природних або штучних децентралізованих системах, що самоорганізуються, яка використовується в ШІ. Термін SI був введений в 1989 р. Х. Бені і Ц. Ваном в контексті клітинних роботизованих систем.

Системи SI складаються з сукупності ***простих агентів,*** або ***боїдів***, які локально взаємодіють один з одним і зі своїм середовищем. Агенти виконують дуже прості правила, і хоча немає централізованої структури управління, яка визначає, як окремі агенти повинні вести себе, локальні і певною мірою випадкові взаємодії між такими агентами приводять до появи «розумної» глобальної поведінки, невластивої для будь-якого самостійного агента.

Приклади поведінки і взаємодії боїдів у природних системах включають колонії мурах, бджолині сім'ї, пташині зграї, полювання на яструбів, випас тварин, зграї риб, зростання бактерій, інтелект мікробів.

Застосування принципів SI до роботів називається ройовою робототехнікою, в той час як SI належать до більш загального набору алгоритмів.

Передбачення SI використовувалося в контексті задач прогнозування. Крім того, підходи, аналогічні до підходів, що були запропоновані для ройової робототехніки, розглядаються для генетично модифікованих організмів в синтетичному колективному розумі.

Концепція боїдів була розроблена в 1986 р. К. Рейнольдсом як імітація зграйної поведінки птахів (рис. 4.4а) у комп'ютерній програмі «штучного життя», а назва «боїд» первісно стосувалося птахоподібного об'єкту.

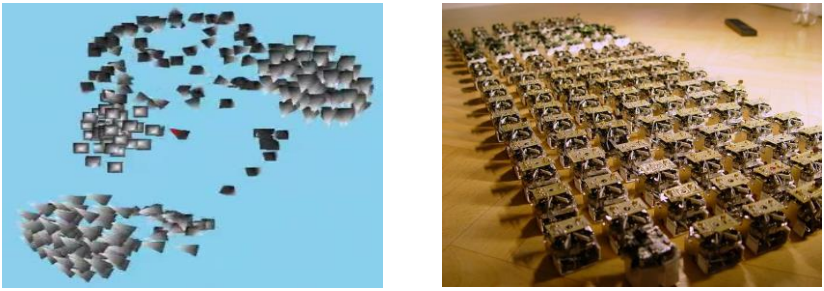
Як і більшість штучних симуляторів життя, Boids є прикладом колективної поведінки; тобто складність Boids виникає через взаємодію окремих агентів, які дотримуються набору простих правил, таких як:

1) ***розподіл***: рухайтеся, щоб уникнути локальних скупченостей;
2) ***вирівнювання***: тримайтеся середнього курсу місцевих побратимів;
3) ***згуртованість***: тримайтеся, щоб рухатися до середнього положення (центру мас) місцевих родичів;
4) можуть бути додані більш складні правила, такі як ***уникнення перешкод*** і ***пошук мети***.

В останні роки ця концепція була реалізована в колективах («зграях») мініатюрних (і навіть мікроскопічних) роботів, які можуть ходити, плавати, літати і виконувати певні функції (рис.4.4б).

Самохідні частинки (self-propelling particles, SPP) – це окремий випадок боїдів, який був запропонований в 1995 р.[21]. Сукупність SPP моделюється набором частинок, які рухаються з постійною швидкістю, але реагують на випадкові обурення, приймаючи на кожному часі середній напрямок руху інших частинок в їх локальній околиці.

Моделі SPP прогнозують, що колективи тварин поділяють певні властивості на рівні групи, незалежно від типу тварин у зграї. Системи SPP породжують емерджентну поведінку (emergency behavior), яка відбувається в багатьох різних масштабах, деякі з яких виявляються універсальними і надійними.

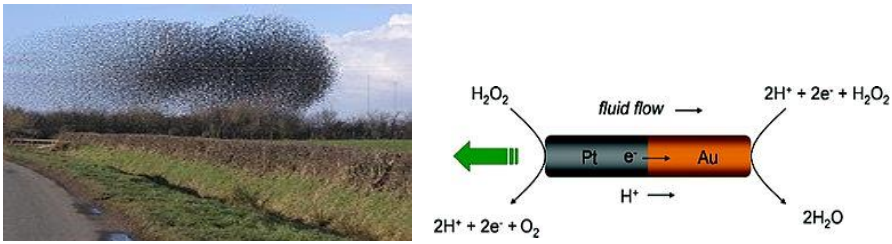


а *б*

Рис. 4.4. Приклад неоднорідної сукупності різних боїдів (а) і реалізація алгоритму boids у сукупності міні-роботів (б)

[21] Vicsek T., Czirok A., Ben-Jacob E. et al. Novel type of phase transition in a system of self-driven particles // Phys. Rev. Lett. 1995. Vol. 75, № 6. P. 1226-1229.

У теоретичній фізиці концепція SPP породила проблему знайти мінімальні статистичні моделі, які відображають таку поведінку. У 2020 р. дослідники з ун-ту Лестера повідомили про нерозпізнані зграї самохідних частинок, які вони назвали «малі вихори» (swirlons). Вихровий стан свірлонів складається з локальних завихрень, утворених групами самохідних частинок, що обертаються навколо загального центру мас. У відповідь на зовнішнє навантаження свірлони рухаються з постійною швидкістю, пропорційною до доданої силі, як об'єкти в в'язкому середовищі. Вихори притягуються один до одного і зливаються, утворюючи більший спільний вихор. Приклади такої складної поведінки легко знайти в природі (рис. 4.5а), але існують і прості фізичні системи (рис. 4.5б), в яких складна динаміка свірлонів може бути реалізована.



а *б*

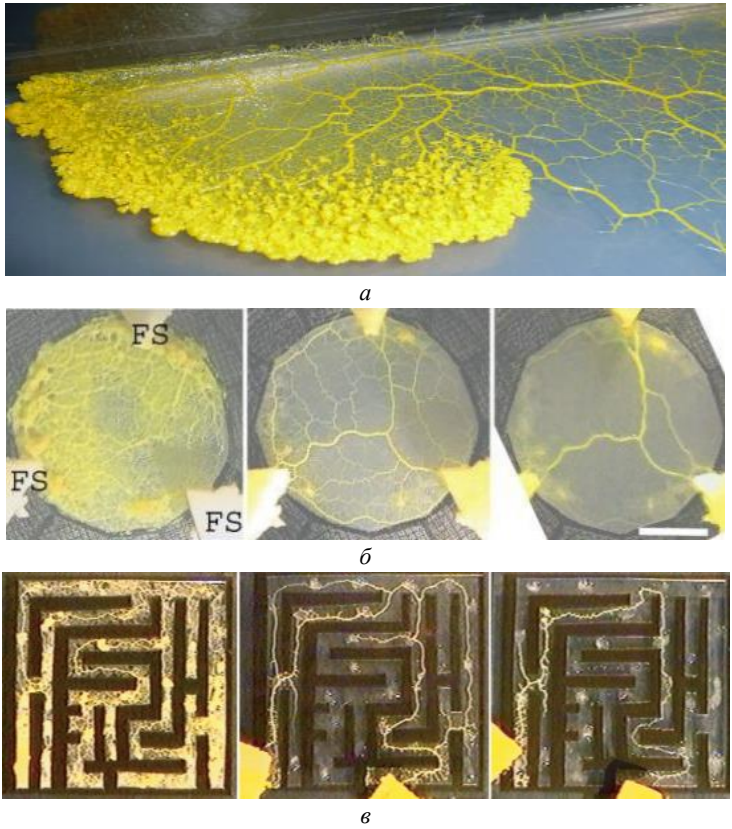
Рис. 4.5. Складна вихорова динаміка в зграї птахів (а) і фізичній системі (б)

Алгоритми SI:

1. ***Еволюційні алгоритми*** (EA).
2. ***Диференціальна еволюція*** (DE).
3. ***Оптимізація колоній мурах*** (ACO) – це імовірнісний метод, корисний в задачах, які пов'язані з пошуком кращих шляхів через графи. Штучні «мурахи» – агенти моделювання – знаходять оптимальні рішення шляхом пересування по ***простору параметрів***, який представляє всі можливі рішення. Природні мурахи відкладають феромони, що направляють їх до ресурсів під час дослідження навколишнього середовища. Змодельовані «мурахи» аналогічним чином записують своє становище і якість своїх рішень, щоб на більш пізніх ітераціях моделювання знаходили більше мурах для кращих рішень;
4. ***Стохастичний дифузний пошук*** (SDS) – агентно-імовірнісний метод глобального пошуку й оптимізації, який найкраще підходить для задач, в яких цільова функція може бути розкладена на кілька незалежних часткових функцій. Кожен агент підтримує гіпотезу, яка ітеративно перевіряється шляхом оцінки випадково обрано часткової цільової функції, параметризованої поточною гіпотезою агента.
5. ***Штучний інтелект рою*** (ICP) – це метод посилення колективного розуму мережевих груп людей з використанням алгоритмів управління, змодельованих за зразком природних систем (Human Swarming або Swarm AI). Метод з'єднує в реальному часі групи учасників, які обмірковують і сходяться в рішеннях як

динамічні рої, яким одночасно задають питання. Метод дозволяє надати бізнес-командам можливості генерувати високоточні фінансові прогнози, любителям спорту – можливості перевершити ринки ставок у Вегасі, а групам лікарів – можливості ставити діагнози зі значно більшою точністю, ніж традиційні методи.

4.5. Розум слизовиків – це концепція, яка почалася з вивчення незвичайних властивостей слизовиків *physarum polycephalum*. Це організм, який належить до групи слизовиків і є родичем амеб, хоча зовні більше схожий на гриб. Більшу частину часу він є однією велетенською клітиною (рис. 4.6а)



а

б

в

Рис. 4.6. Слизовик physarum polycephalum (а) формує мережі (б) і знаходить оптимальний шлях крізь лабіринт (в)[22]

розміром до кількох десятків сантиметрів із численними ядрами та переплетеною мережею трубочок, яка здатна до активного руху в пошуках джерела

[22] Nakagaki T., Yamada H., Toth A. Intelligence: Maze-solving by an amoeboid organism // Nature. 2000. Vol. 407. P. 470-475.

їжі. Функцію нервової системи у нього виконує складна внутрішня мережа трубок, які змінюють діаметр та перекачують рідину й поживні речовини через організм. Що частіше слизовик рухається в конкретному напрямку, то більш чіткою залишається архітектура трубочок, що до нього веде. Це чимось схоже на процес зберігання спогадів нейронами: якщо ми часто згадуємо щось, то зв'язки між клітинами й самі спогади посилюються, але якщо не користуватися певними спогадами, то відповідні нейронні зв'язки з часом слабшають і зникають. Завдяки цій властивості слизовик здатний швидко розв'язувати задачу Штейнера пошуку найкоротшої мережі, що з'єднує заданий скінченний набір точок на площині (задача про мінімальне древо) (рис. 4.6б), а також знаходити і запам'ятовувати шлях до їжі крізь лабіринт (рис. 4.6в). Принципи «розуму слизовиків» вважають дуже перспективними для розв'язання задач оптимізації на мережах і використання в ШІ.

4.6. Застосування і ризики ШІ

– в галузі охорони здоров'я для дозування ліків і різних видів лікування пацієнтів, а також для хірургічних процедур в операційній;
– безпілотні автомобілі, човни, дрони. Кожна з цих машин повинна зважити наслідки будь-якої дії, до якої вони вдаються, оскільки кожна дія вплине на кінцевий результат (наприклад, запобігти зіткненню);
– у фінансовій індустрії для виявлення і маркування діяльності в банківській справі та фінансах, таких як незвичайне використання дебетових карт і великі депозити на рахунках – все це допомагає відділу з боротьби з шахрайством в банку;
- спрощення і полегшення торгівлі за рахунок спрощення оцінки пропозиції, попиту і цін на цінні папери.

Можливі ризики ШІ:

– ШІ може стати настільки високорозвиненим, що люди не зможуть за ними встигати, і машини будуть самі модернізувати і копіювати себе з експоненційною швидкістю;
– ШІ може втручатися у приватне життя людей і навіть використовуватися в якості зброї;
– є проблема етичності ШІ і того, чи слід ставитися до інтелектуальних систем, таких як роботи зі ШІ, як до людей;
– безпілотні автомобілі викликають досить багато суперечок, оскільки їх ШІ, як правило, розрахований на найменший можливий ризик і найменші втрати. Якби їм представився сценарій одночасного зіткнення з цією людиною або іншою, ці автомобілі розрахували б варіант, який завдасть найменшої шкоди;
– ШІ може вплинути на зайнятість людей. Оскільки багато галузей прагнуть автоматизувати певні робочі місця за допомогою інтелектуального обладнання, є побоювання, що люди будуть витіснені з ринку робочої сили. Безпілотні автомобілі можуть усунути необхідність у таксі і програмах обміну автомобілями, в той час як виробники можуть легко замінити людську працю машинами.

ПИТАННЯ ДО САМОКОНТРОЛЮ

1. Які три головні характеристики «Big Data» і які основні підходи до обробки «Big Data»?
2. Опишіть проблеми сучасної аналітичної архітектури для дослідників «Big Data».
3. Як у дереві рішень алгоритм вибирає атрибути для розділення?
4. Джон пішов до лікаря з приводу сильного головного болю. Лікар попрохав Джона про всяк випадок зробити аналіз крові на незвичний грип, який, як підозрюється, вражає 1 з 5000 людей у цій країні. Тест на 99 % точний, в тому сенсі, що ймовірність помилково позитивного тесту становить 1 %, а ймовірність а помилково негативного – нуль. Тест Джона повернувся позитивним. Яка ймовірність того, що Джон захворів на грип?
5. Який класифікатор вважається обчислювально ефективним для задач із великими розмірами? Чому?
6. Команда дослідників даних працює над проблемою класифікації, в якій набір даних містить багатокорельовані змінні, і більшість з них є категоріальними змінними. Можливість використання якого класифікатора повинна розглянути команда? Чому?
7. Команда дослідників даних працює над проблемою класифікації, в якій набір даних містить багатокорельовані змінні і більшість з них є безперервними. Команда хоче, щоб модель виводила ймовірності на додаток до міток класів. Який класифікатор слід розглянути команді? Чому?
8. Які ви знаєте джерела «Big Data»? Яки типи даних?
9. Назвіть головні способи обробки «Big Data» і їх особливості.
10. Назвіть аналітичні моделі «Big Data» і їх особливості.

ЛІТЕРАТУРА

1. Data Science and Big Data Analytics. EMC Educational Series. Wiley. 2015. URL: http://index-of.co.uk/Big-Data-Technologies/Data%20Science%20and%20Big%20Data%20Analytics.pdf
2. Bahga A., Madisetti V. Big Data Analytics: A Hands-On Approach. 2019. URL:https://www.pdfdrive.com/big-data-analytics-a-hands-on-approach-e158549112.html
3. Marr B. Big Data in Practice: How 45 Successful Companies Used Big Data Analytics to Deliver Extraordinary Results. Wiley. 2016. URL: https://www.pdfdrive.com/big-data-in-practice-how-45-successful-companies-used-big-data-analytics-to-deliver-extraordinary-results-d158103513.html
4. Mailund Th. Beginning Data Science in R: Data Analysis, Visualization, and Modelling for the Data Scientist. 2017. URL: https://www.pdfdrive.com/beginning-data-science-in-r-data-analysis-visualization-and-modelling-for-the-data-scientist-d181093942.html
5. Теорія ймовірностей та математична статистика : навчальний посібник / Кушлик-Дивульська О. І., Поліщук Н. В., Орел Б. П., Штабалюк П. І. Київ: НТУУ «КПІ», 2014. URL: https://ela.kpi.ua/bitstream/123456789/18378/1/5%20%D0%9A%D1%83%D1%88%D0%BB%D0%B8%D0%BA-%D0%94%D0%B8%D0%B2%D1%83%D0%BB%D1%8C%D1%81%D1%8C% D0%BA%D0%B0.pdf
6. Василенко О. А., Сенча І. А. Математично-статистичні методи аналізу в прикладних дослідженнях: навчальний посібник. Одеса, 2012. URL: http://www.immsp.kiev.ua/postgraduate/Biblioteka_trudy/Vasylenko_Sencha_Matem._statyst.metody_2011.pdf
7. Математична статистика: начальний посібник / Лебєдєв Є. О., Лівінська Г. В., Розора І. В., Шарапов М. М. Київ: КНУ ім. Т. Г. Шевченка, 2016. URL: http://applstat.univ.kiev.ua/ukr/docs/materials/matstat.pdf
8. Медведєв М. Г., Пащенко І. О. Теорія ймовірностей та математична статистика: підручник. Київ, 2008. URL: http://pdf.lib.vntu.edu.ua/books/2015/Medvedev_2008_536.pdf
9. Руденко В. М. Математична статистика: навчальний посібник. Київ, 2012. URL: https://shron1.chtyvo.org.ua/Rudenko_Volodymyr/Matematychna_statystyka.pdf
10. Олещенко Л. М. Технології оброблення великих даних: навчальний посібник. КПІ імені Ігоря Сікорського, Київ, 2021. URL: https://ela.kpi.ua/bitstream/123456789/42206/1/%D0%9AonspLekts_Tekhnolohii-obroblennia-velykykh-danykh_%D0%9Eleshchenko.pdf

ДЛЯ НОТАТОК

Навчальне видання

Кізілова Наталія Миколаївна

МЕТОДИ АНАЛІЗУ
«ВЕЛИКИХ ДАНИХ»

Методичні рекомендації з курсу
«Прикладні задачі аналізу великих даних» для здобувачів вищої освіти
спеціальності «Прикладна математика»

Коректор *О. В. Анцибора*
Комп'ютерне верстання *В. В. Савінкова*
Макет обкладинки *І. М. Дончик*

Формат 60х84/16. Ум. друк. арк. 5,68. Наклад 50 пр. Зам. № 231/21.

Видавець і виготовлювач
Харківський національний університет імені В. Н. Каразіна,
61022, м. Харків, майдан Свободи, 4.
Свідоцтво суб'єкта видавничої справи ДК № 3367 від 13.01.2009
Видавництво ХНУ імені В. Н. Каразіна