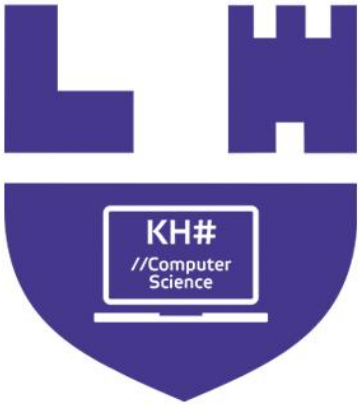
**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**
**Луцький національний технічний університет**
**Факультет комп'ютерних та інформаційних технологій**
**Кафедра комп'ютерних наук**




# ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ

**Конспект лекцій для здобувачів**
**першого (бакалаврського) рівня вищої освіти**
**освітньої програми «Комп'ютерні науки»**
**галузі знань 12 Інформаційні технології**
**спеціальності 122 Комп'ютерні науки**
**денної та заочної форм навчання**

**2025**

УДК 004.41

До друку
Голова вченої ради ФКІТ _____І. С. Кондіус

Електронна копія друкованого видання передана для внесення в репозитарій ЛНТУ
директор бібліотеки _____Н. П. Поліщук

Затверджено вченою радою ФКІТ,
протокол №__ від «__» _____ 2025 року.

Розглянуто та схвалено на засіданні кафедри комп'ютерних наук ЛНТУ,
протокол №__ від «__» _____2025 року.

Завідувач кафедри КН_____В. О. Ліщина

Укладачі: _____ В.О. Ліщина, кандидат технічних наук, доцент кафедри комп'ютерних наук ЛНТУ.
_____ К.В. Вавринюк, асистент кафедри комп'ютерних наук ЛНТУ.

Рецензент: _____ Н. М. Ліщина, кандидат технічних наук, доцент кафедри інженерії програмного забезпечення ЛНТУ

Інтелектуальний аналіз даних: конспект лекцій для здобувачів першого (бакалаврського) рівня вищої освіти освітньої програми «Комп'ютерні науки» галузі знань 12 Інформаційні технології спеціальності 122 Комп'ютерні науки денної та заочної форм навчання / В.О. Ліщина, К.В. Вавринюк. Луцьк: ЛНТУ. 2025. 137 с.

У методичних вказівках наведений конспект лекцій з дисципліни «Інтелектуальний аналіз даних». Призначені для студентів спеціальності 122 «Комп'ютерні науки» денної форми навчання.

# ЗМІСТ

# ВСТУП

Дисципліна «Інтелектуальний аналіз даних» знайомить студентів з основними поняттями, методами та інструментами Data Mining для виявлення закономірностей у великих обсягах даних. Розглядаються ключові техніки, такі як класифікація, кластеризація, прогнозування, аналіз асоціацій, а також технології Web Mining.

Значна увага приділяється практичному застосуванню методів ІАД: дерева рішень, нейронні мережі, генетичні алгоритми, еволюційне програмування та комбіновані підходи. Кожен метод супроводжується прикладами реального використання в бізнесі, соціальній сфері та державному управлінні.

Мета дисципліни – сформувати теоретичну базу та практичні навички застосування інтелектуального аналізу в сучасних інформаційних системах, зокрема для підтримки прийняття рішень.

Завдання дисципліни:

- вивчення методів аналізу даних та їх застосування в управлінських рішеннях;
- засвоєння принципів побудови систем на основі Data Mining;
- набуття навичок розробки та застосування моделей у сферах бізнесу, фінансів, маркетингу та планування;
- опанування практики програмування окремих компонентів систем ІАД.

У результаті вивчення дисципліни студенти повинні:

- знати: принципи роботи основних алгоритмів ІАД, етапи аналізу даних, структуру інформаційно-аналітичних систем;
- вміти: формулювати завдання, будувати математичні моделі, обирати та застосовувати методи аналізу даних, інтерпретувати результати та використовувати їх у практичних ситуаціях.

# ЛЕКЦІЯ 1
# ОСНОВНІ ПОНЯТТЯ ТА ВИЗНАЧЕННЯ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ. ТИПИ ДАНИХ ДЛЯ РОБОТИ В DATA MINING

## 1.1 Історія виникнення та причини розвитку

Поняття Data Mining з'явилося у 1978 році, однак справжньої популярності в сучасному розумінні набуло з першої половини 1990-х років. До цього часу обробку та аналіз даних здебільшого виконували в межах прикладної статистики, орієнтуючись на задачі аналізу невеликих баз даних.

Термін «інтелектуальний аналіз даних» (англ. Data Mining) утворено від слів data (дані) та mining (видобуток), що відображає суть процесу – пошук цінної інформації у великих обсягах даних. Обидва поняття мають спільну ідею: необхідність або просіювати величезні масиви «сировини», або проводити інтелектуальний пошук корисної інформації. В українській мові термін Data Mining трактується по-різному: видобуток даних, витягування інформації, розкопування даних, інтелектуальний аналіз даних, пошук закономірностей, виявлення знань, аналіз шаблонів тощо.

Синонімом інтелектуального аналізу даних часто вважається термін «виявлення знань у базах даних» (KDD – Knowledge Discovery in Databases).

Розвиток технологій баз даних.

- 1960-ті роки. У 1968 році IBM ввела в експлуатацію першу промислову СУБД – систему IMS;
- 1970-ті роки. У 1975 році був створений перший стандарт асоціації по мовах обробки даних – CODASYL, який заклав фундамент мережевої моделі даних. Значний внесок у розвиток теорії баз даних зробив американський математик Едгар Франк Кодд, який запропонував реляційну модель даних;
- 1980-ті роки. Цей період ознаменувався експериментами з новими підходами до структуризації даних та організації доступу до них. У 1985 році була створена мова SQL, яка згодом стала стандартом у більшості СУБД;
- 1990-ті роки. У цей час з'явилися нові типи даних: зображення, документи, звук, карти, а також текстові та часові формати. В мову SQL були додані нові можливості. Саме тоді почали розвиватися технології Data Mining, сховища даних, мультимедійні бази даних та веб-бази даних.

З розвитком технологій зберігання і запису даних у світі почався справжній інформаційний вибух. Діяльність будь-якої організації (комерційної, виробничої, медичної, наукової тощо) супроводжується детальною фіксацією кожної події. У результаті виникла проблема: що робити з усіма цими даними? Стало очевидно, що без ефективної обробки сирі

дані перетворюються на інформаційне «сміття» [1].

Сучасні вимоги до аналізу даних:

- великі обсяги даних;
- різнорідність (кількісні, якісні, текстові тощо);
- зрозумілі та конкретні результати;
- простота використання інструментів.

Традиційна математична статистика, яка довгий час була головним інструментом аналізу, вже не могла ефективно справлятися з новими викликами. Основна причина – орієнтація на усереднення, яке іноді не дає корисної інформації (наприклад, середня температура по лікарні). Методи статистики залишаються корисними для перевірки гіпотез і базового розвідувального аналізу (OLAP – аналітична обробка даних у реальному часі).

Причини популярності Data Mining:

- стрімке зростання обсягів даних;
- комп'ютеризація бізнес-процесів;
- широке поширення Інтернету;
- розвиток інформаційних технологій (покращення СУБД, сховищ даних);
- технологічний прогрес (зростання продуктивності комп'ютерів, обсягів носіїв, поява Grid-систем).

Ілюстрацією популярності Data Mining є зростання кількості результатів у пошуковій системі Google:

- у вересні 2005 року – понад 18 мільйонів сторінок;
- у вересні 2013 – вже 198 мільйонів.

Data Mining – мультидисциплінарна галузь, що виникла і розвивається на базі таких наук як прикладна статистика, розпізнавання образів, штучний інтелект, теорія баз даних та ін.(рис. 1.1).

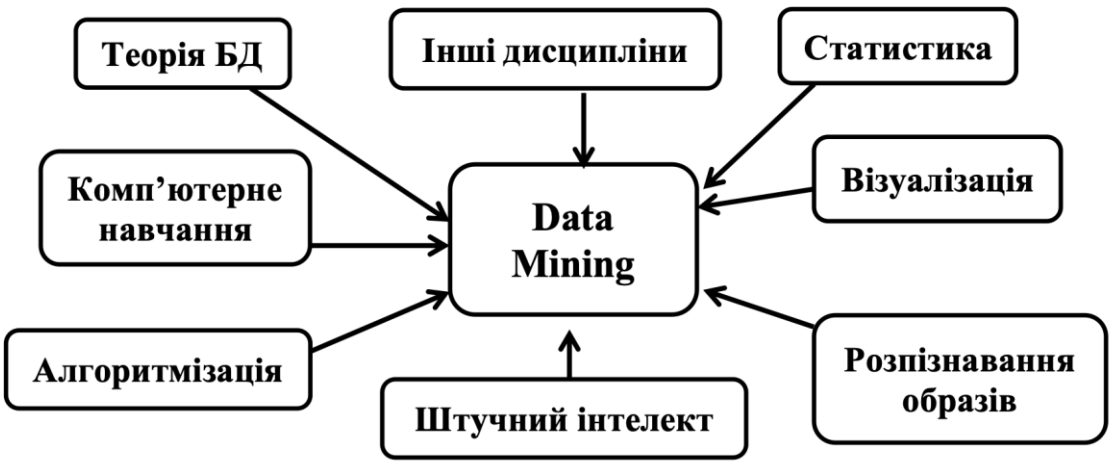



Рисунок 1.1 – Data Mining, як міждисциплінарна галузь

## 1.2 Суть, мета та сфера застосування технології Data Mining

Суть і мета технології Data Mining полягає у виявленні в великих обсягах даних неочевидних, об'єктивних і практично корисних закономірностей.

Неочевидні – це закономірності, які неможливо виявити за допомогою стандартних методів обробки інформації або експертного аналізу.

Об'єктивні – такі, що достовірно відображають дійсність, на відміну від суб'єктивних експертних оцінок.

Практично корисні – тобто такі, що мають прикладне значення і можуть бути використані для прийняття ефективних рішень.

Знання – це сукупність відомостей, що утворюють цілісне уявлення про певний об'єкт, явище або проблему.

Використання знань передбачає їхнє практичне застосування для отримання переваг, наприклад, у конкурентній боротьбі на ринку.

Визначення технології Data Mining.

Data Mining – це процес виявлення прихованої та неструктурованої інформації з даних і подання її у формі, зручній для використання.

За визначенням SAS Institute, Data Mining – це процес виявлення, дослідження та моделювання великих обсягів даних з метою виявлення нових, раніше невідомих структур (моделей), що дають бізнес-переваги.

За визначенням Gartner Group, Data Mining – це процес виявлення нових значущих кореляцій, шаблонів і тенденцій через аналіз великих обсягів збережених даних із використанням методів розпізнавання образів, статистики та математики.

Сфери застосування Data Mining.

Data Mining можна застосовувати в будь-якій сфері, де є дані. Найбільший інтерес до цієї технології проявляють комерційні підприємства, що працюють з інформаційними сховищами даних. Практика показує, що впровадження Data Mining може дати значну економічну віддачу – до 1000% від початкових інвестицій [1].

Наприклад:

- ефект у 10-70 разів перевищував початкові вкладення (350-750 тис. доларів);
- проект вартістю 20 млн доларів окупився за 4 місяці;
- економія 700 тис. доларів на рік завдяки впровадженню Data Mining у британській мережі супермаркетів.

Основні галузі застосування.

Роздрібна торгівля. Аналіз купівельного кошика: виявлення товарів, що часто купуються разом, для оптимізації розкладки товарів, реклами та управління запасами. Аналіз

часових шаблонів: визначення, коли клієнти, наприклад, купують аксесуари після основної покупки (наприклад, батарейки після покупки відеокамери). Прогнозні моделі: аналіз поведінки клієнтів для таргетованого маркетингу.

Банківська справа. Виявлення шахрайства з картками: аналіз транзакцій для виявлення підозрілих шаблонів. Сегментація клієнтів: побудова персоналізованих пропозицій. Прогнозування змін у клієнтській базі: оцінка потенційної цінності клієнтів.

Телекомунікації. Аналіз характеристик викликів: розробка привабливих тарифних планів. Оцінка лояльності клієнтів: фокусування маркетингових зусиль на перспективних користувачах.

Страхування. Виявлення шахрайства: аналіз заяв на відшкодування для виявлення підозрілих зв'язків. Оцінка ризиків: аналіз факторів, що впливають на кількість і суму страхових виплат.

Інші бізнес-галузі. Автомобільна промисловість: прогнозування популярності конфігурацій авто. Гарантійна політика: оцінка майбутніх гарантійних витрат. Авіаперевезення: аналіз часто літаючих пасажирів для оптимізації програм лояльності.

Медицина. Створення експертних систем для діагностики та вибору лікування на основі аналізу симптомів, історій хвороб та результатів обстежень.

Молекулярна генетика і генна інженерія. Виявлення генетичних маркерів, які контролюють фенотипічні ознаки. Ці дослідження мають стратегічне значення для біомедицини та агросектора.

Прикладна хімія. Аналіз структури хімічних сполук з метою передбачення їхніх властивостей. Data Mining допомагає працювати з дуже складними й багатовимірними даними.

Можна навести ще багато прикладів різних областей знання, де методи Data Mining відіграють провідну роль. Особливість цих областей полягає в їх складній системної організації. Вони відносяться головним чином до над кібернетичному рівню організації систем, закономірності якого не можуть бути достатньо точно описані на мові статистичних чи інших аналітичних математичних моделей. Дані в зазначених областях неоднорідні, гетерогенні, нестаціонарні і часто відрізняються високою розмірністю.

### 1.3 Типи закономірностей

Методи Data Mining дозволяють виявляти п'ять основних типів закономірностей: асоціації, послідовності, класифікацію, кластеризацію та прогнозування.

Асоціація має місце тоді, коли кілька подій пов'язані між собою. Наприклад, дослідження в супермаркеті може показати, що 65 % покупців кукурудзяних чіпсів також

купують «Кока-Колу». Якщо ж пропонується знижка на комплект з обох товарів, то напій купують вже в 85 % випадків. Завдяки виявленню таких асоціацій, менеджери можуть оцінити ефективність акцій і вплив комбінацій товарів на продажі.

Послідовність виявляється, коли події пов’язані не лише між собою, а й у часі. Наприклад, після придбання житла у 45 % випадків протягом місяця купується нова кухонна плита, а впродовж двох тижнів 60 % новоселів купують холодильник. Такі закономірності корисні для планування маркетингових кампаній або розробки пропозицій, що враховують часову логіку дій клієнтів [1].

Класифікація полягає у виявленні ознак, що характеризують належність об’єкта до певної заздалегідь визначеної групи. Це досягається шляхом аналізу вже класифікованих даних і побудови правил, які дозволяють віднести нові об’єкти до відповідних категорій.

Кластеризація відрізняється тим, що групи (кластери) заздалегідь не визначені. За її допомогою система самостійно виявляє схожі між собою об’єкти, формуючи однорідні групи без попереднього навчання. Це особливо корисно для виявлення прихованих структур у даних або нових сегментів клієнтів.

Прогнозування базується на аналізі історичних даних, представлених у вигляді часових рядів. Якщо вдається побудувати модель, яка достовірно описує динаміку поведінки певних показників, то з її допомогою можна робити передбачення щодо майбутнього стану системи. Такі моделі широко використовуються в фінансовому аналізі, управлінні запасами, плануванні виробництва тощо.

### 1.4 Класи систем Data Mining

Data Mining – це мультидисциплінарна галузь, що виникла на стику прикладної статистики, розпізнавання образів, штучного інтелекту, теорії баз даних та інших дисциплін (рис. 1.2). Внаслідок цього існує велика кількість методів і алгоритмів, реалізованих у різних сучасних системах Data Mining. Багато з них інтегрують кілька підходів, але кожна система зазвичай має провідну компоненту, яка визначає її основну методологію. Нижче подано класифікацію таких ключових підходів з короткою характеристикою кожного з них.

Предметно-орієнтовані аналітичні системи. До цієї категорії належать системи, призначені для конкретних предметних галузей. Найпоширенішим підкласом є технічний аналіз – набір методів прогнозування динаміки цін і формування інвестиційних портфелів, заснований на емпіричних моделях фінансового ринку. Хоча ці методи використовують прості статистичні засоби, вони глибоко враховують специфіку галузі. Програми цього класу зазвичай доступні за ціною ($300–1000).

Статистичні пакети. Сучасні статистичні пакети (наприклад, SAS, SPSS, STATISTICA)

включають деякі елементи Data Mining. Основна увага в них зосереджена на класичних статистичних методах – кореляційному, регресійному, факторному аналізі тощо. Недоліки:

- потреба у високій кваліфікації користувача;
- велика вартість ($1000–15000);
- фокус на усереднених характеристиках вибірки, які не завжди відображають реальні закономірності.

Нейронні мережі. Ці системи моделюють роботу нейронів у вигляді багаторівневих структур, що навчаються за допомогою механізму зворотного поширення помилки. Їх перевага – здатність виявляти складні нелінійні залежності. Основні недоліки:

- потреба у великих навчальних вибірках;
- відсутність інтерпретованості (так званий "чорний ящик").

Приклади: BrainMaker, NeuroShell, OWL. Вартість – $1500–8000.

Системи на основі аналогічних випадків (CBR). CBR-системи використовують минулі подібні випадки для прийняття рішень. Метод також відомий як «найближчий сусід». Перевага – гнучкість і простота, але є і мінуси:

- відсутність узагальнюючих моделей;
- суб’єктивність у виборі метрики схожості.

Приклади: KATE tools, Pattern Recognition Workbench.

Дерева рішень. Дерева рішень реалізують класифікацію на основі ієрархії правил типу "ЯКЩО – ТО". Їх популярність пояснюється простотою та наочністю, але вони мають обмеження:

- не гарантують пошуку найкращих правил;
- схильні до перенавчання.

Приклади: See5/C5.0, Clementine, SIPINA, IDIS, KnowledgeSeeker. Ціна: $1000–$10 000.

Еволюційне програмування. Цей підхід полягає у поступовому поліпшенні гіпотез у вигляді програм, які виражають залежність цільової змінної. Приклад – система PolyAnalyst, яка застосовує еволюційний підхід до побудови програм. Інший приклад – метод групового урахування аргументів (МГУА), реалізований у системі NeuroShell.

Генетичні алгоритми. Генетичні алгоритми моделюють біологічну еволюцію: популяції хромосом еволюціонують шляхом мутацій, кросинговеру тощо. Вони використовуються для вирішення задач оптимізації і дедалі частіше входять до складу Data Mining-систем.

Приклад: GeneHunter. Ціна – близько $1000.

Алгоритми обмеженого перебору. Метод, заснований на обчисленні частот комбінацій простих логічних подій. Приклад – WizWhy. Незважаючи на заяви розробників про повне покриття правил у даних, система має обмеження:

- максимальна довжина комбінацій (до 6 умов);
- евристичний відбір початкових логічних подій.

Системи Data Mining охоплюють широкий спектр підходів – від класичних статистичних до сучасних нейромережевих і еволюційних. Кожен підхід має свої переваги, недоліки та сферу ефективного застосування. Вибір системи залежить від завдання, обсягів даних, вимог до інтерпретації результатів та доступних ресурсів.

### 1.5 Якісний аналіз даних з використанням DM

Для якісного аналізу будь-яких даних слід дотримуватися загальної схеми використання DM:

- висування гіпотез;
- збір та систематизація даних;
- підбір адекватної моделі;
- тестування та інтерпретація отриманих даних;
- використання у реальних умовах.

Ця схема не залежить від предметної області та сфери діяльності. Вона є універсальною.

Висування гіпотез. Гіпотезою тут будемо вважати припущення про вплив певних факторів на процес, що досліджується.

Автоматизувати процес висування гіпотез є вкрай складно, тому, цю задачу мають вирішувати експерти – фахівці в предметній області.

Слід довіритися їх досвіду та здоровому глузду, максимально використати ці знання про предмет досліджень і зібрати як найбільше гіпотез/припущень.

Зазвичай, добрі результати надають тактики «круглого столу» або «мозкової атаки». На початку слід зібрати та систематизувати всі ідеї, а оцінювати їх пізніше. В результаті повинен бути складений перелік з описів всіх факторів досліджуваного об’єкту[2].

Наприклад, для задачі прогнозування попиту товару потрібно скласти перелік факторів, що впливатимуть на об’єкт і експертно-оцінити суттєвість кожного з них (табл.1.1). Така оцінка не є вирішальною, але від неї починають відштовхуватися.

Таблиця 1.1 Приклад оцінки суттєвості факторів

| Фактор | Суттєвість (0–100) |
|---|---|
| Сезон | 100 |
| День тижня | 80 |
| Об’єм продажів за попередні тижні | 100 |
| Об’єм продажів за аналогічний період минулого року | 95 |

| | |
|---|---|
| Рекламна кампанія | 60 |
| Маркетингові заходи | 40 |
| Якість продукції | 50 |
| Бренд | 25 |
| Коливання ціни від середньоринкової | 60 |
| Наявність подібного товару в конкурентів | 15 |

Згодом, під час аналіз, може з’ясуватися, що фактор, який експерти оцінили як важливий, буде мати незначний вплив на процес і навпаки.

Збір та систематизація даних. Для проведення аналізу необхідно мати якомога більше даних, оскільки це дозволяє оцінити вплив максимальної кількості показників. Надалі легше виключити частину даних, ніж починати новий процес збору.

Існує кілька основних підходів до отримання даних, які можуть бути використані окремо або в комбінації залежно від завдань дослідження та доступних ресурсів.

Отримання даних з внутрішніх джерел. Цей спосіб є відносно простим, оскільки інформація, як правило, зберігається в облікових системах у вигляді таблиць. У таких системах доступні механізми формування звітів та експорту даних.

Отримання відомостей з непрямих даних. Наприклад, при оцінці реального фінансового стану мешканців певного регіону можна використати непрямі показники. У випадку з продажем автомобілів різних цінових категорій аналіз частки дорогих товарів у структурі продажів дозволяє зробити висновки щодо рівня доходів населення.

Використання відкритих джерел. До відкритих джерел належать статистичні збірники, корпоративні звіти, результати маркетингових досліджень та соціологічні опитування, які перебувають у вільному доступі.

Проведення власних маркетингових досліджень. Цей метод є дорогим, однак забезпечує високу ефективність збору релевантних даних. Збір даних на основі експертних оцінок співробітників. Важливо оцінити доцільність витрат на отримання необхідної інформації. Частина даних може бути отримана з відкритих джерел, інші – потребують оплати. Дані щодо діяльності конкурентів часто є дорогими.

Вартість збору інформації суттєво варіюється залежно від методу, тому доцільно зіставляти поточні витрати з очікуваними результатами. Дані, які експерти вважають несуттєвими, можуть бути відхилені. Однак ігнорування значущих даних призведе до того, що аналітична модель ґрунтуватиметься на другорядних факторах і, відповідно, формуватиме нестабільні та недостовірні результати.

### 1.6 Дані, набір даних та їх атрибутів

У широкому розумінні дані – це факти, текст, графіки, картинки, звуки, аналогові або цифрові відео-сегменти. Дані можуть бути отримані в результаті вимірювань, експериментів, арифметичних і логічних операцій. Дані повинні бути представлені у формі, придатній для зберігання, передачі і обробки. Іншими словами, дані – це необроблений матеріал, що надається постачальниками даних і використовується споживачами для формування інформації на основі даних. У таблиці 1.2 представлена двовимірна таблиця, що представляє собою набір даних.

Таблиця 1.2 Двовимірна таблиця «об’єкт-атрибут»

| | Атрибути | | | | |
|---|---|---|---|---|---|
| Об’єкти | Код клієнта | Вік | Сімейний статус | Прибуток | Клас |
| | 1 | 19 | неодр. | 1234 | 1 |
| | 2 | 23 | одр. | 1222 | 1 |
| | 3 | 34 | одр. | 2700 | 1 |
| | 4 | 24 | неодр. | 2343 | 1 |
| | 5 | 26 | одр. | 1765 | 2 |
| | 6 | 32 | розл. | 2652 | 1 |
| | 7 | 19 | неодр. | 1200 | 2 |
| | 8 | 22 | неодр. | 1765 | 2 |
| | 9 | 40 | одр. | 1998 | 1 |
| | 10 | 43 | розл. | 4332 | 1 |

По горизонталі таблиці розташовуються атрибути об’єкта або його ознаки. По вертикалі таблиці – об’єкти. Об’єкт описується як набір атрибутів. Об’єкт також відомий як запис, випадок, приклад, рядок таблиці і т.д.

Атрибут – властивість, що характеризує об’єкт. Наприклад : колір очей людини, температура води і т.д. Атрибут також називають змінною, полем таблиці, виміром, характеристикою.

Змінна (variable) – властивість або характеристика, загальна для всіх досліджуваних об’єктів, прояв якої може змінюватися від об’єкта до об’єкта.

Значення (value) змінної є проявом ознаки.

При аналізі даних, як правило, немає можливості розглянути всю сукупність об’єктів, що нас цікавить. Вивчення дуже великих обсягів даних є дорогим процесом, що вимагає великих затрат часу, а також неминуче призводить до помилок, пов’язаних з людським

фактором.

Цілком достатньо розглянути деяку частину всієї сукупності, тобто вибірку, і отримати цікаву для нас інформацію на її підставі.

Однак розмір вибірки повинен залежати від різноманітності об’єктів, представлених у генеральній сукупності. У вибірці повинні бути представлені різні комбінації і елементи генеральної сукупності.

Генеральна сукупність (population) – вся сукупність досліджуваних об’єктів, що цікавить дослідника.

Вибірка (sample) – частина генеральної сукупності, певним способом відібрана з метою дослідження та отримання висновків про властивості та характеристики генеральної сукупності.

Параметри – числові характеристики генеральної сукупності.

Статистики – числові характеристики вибірки. Часто дослідження ґрунтуються на гіпотезах. Гіпотези перевіряються за допомогою даних.

Гіпотеза – припущення щодо параметрів сукупності об’єктів, яке має бути перевірено на її частині.

Гіпотеза – частково обґрунтована закономірність знань, що служить або для зв’язку між різними емпіричними фактами, або для пояснення факту групи фактів.

Приклад гіпотези: між показниками тривалості життя та якістю харчування є зв’язок. У цьому випадку метою дослідження може бути пояснення змін конкретної змінної, в даному випадку – тривалості життя. Припустимо, існує гіпотеза, що залежна змінна (тривалість життя) змінюється залежно від деяких причин (якість харчування, спосіб життя, місце проживання і т.д.), які і є незалежними змінними [2].

Однак змінна першопочатково не є залежною або незалежною, вона стає такою після формулювання конкретної гіпотези. Залежна змінна в одній гіпотезі може бути незалежною в іншій.

Вимірювання – процес присвоєння чисел характеристикам досліджуваних об’єктів згідно певного правила.

У процесі підготовки даних вимірюється не сам об’єкт, а його характеристики.

Шкала – правило, відповідно до якого об’єктам присвоюються числа.

Багато інструментів Data Mining при імпорті даних з інших джерел пропонують вибрати тип шкали для кожної змінної та/або вибрати тип даних для вхідних і вихідних змінних (символьні, числові, дискретні і безперервні).

Користувачеві такого інструменту необхідно володіти цими поняттями.

Змінні можуть бути числовими даними або символьними.

Числові дані, у свою чергу, можуть бути дискретними і безперервними.

Дискретні дані являються значеннями ознаки, загальне число яких скінченне або нескінченне, але може бути підраховане за допомогою натуральних чисел від одного до нескінченності.

Приклад дискретних даних. Тривалість маршруту тролейбуса (кількість варіантів тривалості скінченне): 10, 15, 25 хв.

Безперервні дані – дані, значення яких можуть набувати якого-завгодно значення в деякому інтервалі. Вимірювання безперервних даних передбачає велику точність.

Приклад безперервних даних: температура, висота, вага, довжина і т.д.

Шкали. Існує п’ять типів шкал вимірювань: номінальна, порядкова, інтервальна, відносна і дихотомічна.

Номінальна шкала (nominal scale) – шкала, яка містить тільки категорії; дані в ній не можуть упорядковуватися, з ними не можуть бути зроблені ніякі арифметичні дії.

Номінальна шкала складається з назв, категорій, імен для класифікації і сортування об’єктів або спостережень за деякою ознакою.

Приклад такої шкали: професії, місто проживання, сімейний стан.

Для цієї шкали застосовні тільки такі операції: дорівнює (=), не дорівнює (≠).

Порядкова шкала (ordinal scale) – шкала, в якій числа присвоюють об’єктам для позначення відносної позиції об’єктів, але не величини відмінностей між ними.

Шкала вимірювань дає можливість ранжувати значення змінних. Вимірювання ж у порядковій шкалі містять інформацію лише про порядок проходження величин, але не дозволяють сказати наскільки одна величина більше іншої, або наскільки вона менше.

Приклад такої шкали: місце (1-ше, 2-ге, 3-є), яке команда отримала на змаганнях, номер студента в рейтингу успішності (1-й, 23-й, і т.д.), при цьому невідомо, наскільки один студент успішніше іншого, відомий лише його номер у рейтингу.

Інтервальна шкала (interval scale) – шкала, різниці між значеннями якої можуть бути обчислені, проте їх відношення не мають сенсу.

Ця шкала дозволяє знаходити різницю між двома величинами, має властивості номінальної та порядкової шкал, а також дозволяє визначити кількісну зміну ознаки.

Приклад такої шкали: температура води в морі вранці – 19 градусів, ввечері – 24, тобто вечірня на 5 градусів вище, але не можна сказати, що вона в 1,26 разів вище.

Номінальна і порядкова шкали є дискретними, а інтервальна шкала – безперервною, вона дозволяє здійснювати точні вимірювання ознаки і виробляти арифметичні операції додавання, віднімання, множення, ділення.

Для цієї шкали застосовні тільки такі операції: одно (=), не дорівнює ( ), більше (>), менше (<), операції додавання (+) і віднімання ( – ).

Відносна шкала (ratio scale) – шкала, в якій є певна точка відліку і можливі відносини

між значеннями шкали.

Приклад такої шкали : вага новонародженої дитини (4 кг і 3 кг). Перша в 1,33 рази важче.

Ціна на картоплю в супермаркеті вище в 1,2 рази, ніж ціна на ринку.

Відносні та інтервальні шкали є числовими .

Для цієї шкали можуть бути застосовані тільки такі операції: рівно (=), не дорівнює (≠), більше (>), менше (<), операції додавання (+) і віднімання (-), множення (*) і ділення (/ ).

Дихотомічна шкала ( dichotomous scale ) – шкала, яка містить тільки дві категорії.

Приклад такої шкали: стать (чоловіча і жіноча).

Приклад використання різних шкал для вимірювань властивостей різних об’єктів, в даному випадку температурних умов, наведено в таблиці даних, зображеної в таблиці 1.3.

Таблиця 1.3 Множина вимірювань властивостей різних об’єктів

| Номер об’єкту | Професія (номінальна шкала) | Середній бал (інтервальна шкала) | Освіта (порядкова шкала) |
|---|---|---|---|
| 1 | Слюсар | 22 | середня |
| 2 | Вчений | 55 | вища |
| 3 | вчитель | 47 | вища |

Приклад використання різних шкал для вимірювань властивостей однієї системи, в даному випадку температурних умов, наведено в таблиці даних, зображеній в таблиці 1.4.

Таблиця 1.4 Множина вимірювань властивостей однієї системи

| Дата змінення | Хмарність (номінальна шкала) | Температура о 7 годині (інтервальна шкала) | Сила вітру (порядкова шкала) |
|---|---|---|---|
| 3 жовтня | Хмарно | 22ºС | Сильний вітер |
| 4 жовтня | напівхмарно | 17ºС | Слабий вітер |
| 5 жовтня | Ясно | 23ºС | Дуже сильний вітер |

Типи наборів даних. Найбільш часто зустрічаються дані, що складаються з записів (record data).

Приклади таких наборів даних: табличні дані, матричні дані, документальні дані, транзакційні або операційні.

Табличні дані – дані, що складаються з записів, кожен з яких складається з фіксованого набору атрибутів.

Транзакційні дані представляють собою особливий тип даних, де кожен запис, що являється транзакцією, включає набір значень.

Приклад транзакційної бази даних, що містить перелік покупок клієнтів магазину,

наведено на рисунку 1.2.

| TID | Items |
|---|---|
| 1 | Bread, Coke, Milk |
| 2 | Beer, Bread |
| 3 | Beer, Coke, Diaper, Milk |
| 4 | Beer, Bread, Diaper, Milk |
| 5 | Coke, Diaper, Milk |

Рисунок 1.2 – Приклад транзакційних даних

Графічні дані. Приклади графічних даних: молекулярні структури; графи (рис. 1.3); карти.

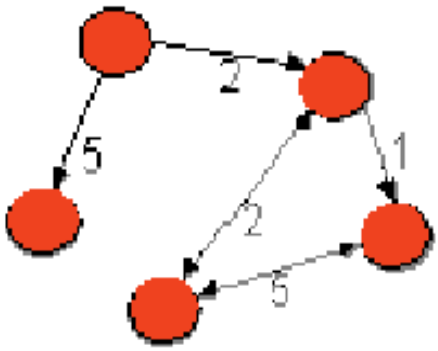



Рисунок 1.3 – Приклад графу

За допомогою карт, наприклад, можна відстежити зміни об’єктів в часі і просторі, визначити характер їх розподілу на площині або в просторі.

Перевагою графічного представлення даних є велика простота їх сприйняття, ніж, наприклад, табличних даних.

Приклад карти, що є картою Кохонена (моделлю нейронних мереж), поданий на рисунку 1.4.

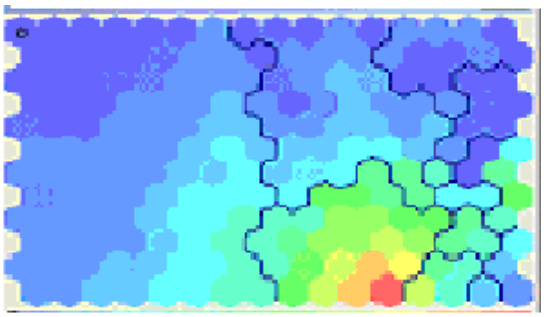


Рисунок 1.4 – Приклад даних типу «Карта Кохонена» хімічні дані

### 1.7 Формати зберігання даних

Одна з основних особливостей даних сучасного світу полягає в тому, що їх стає дуже багато.

Можливі чотири аспекти роботи з даними:

- визначення даних;
- обчислення;
- маніпулювання;
- обробка (збір, передача тощо).

При маніпулюванні даними використовується структура даних типу «файл». Файли можуть мати різні формати.

Більшість інструментів Data Mining дозволяють імпортувати дані з різних джерел, а також експортувати результуючі дані в різні формати.

Дані для експериментів зручно зберігати в якомусь одному форматі.

У деяких інструментах Data Mining ці процедури називаються імпорт / експорт даних, інші дозволяють напряму відкривати різні джерела даних і зберігати результати Data Mining в одному із запропонованих форматів.

Згідно з опитуванням «Формати зберігання даних» найбільше число опитаних (23%) вважають за краще зберігати дані у форматі тієї бази даних, яку вони використовують. У форматі Text, CSV – 18%, по 14% опитаних зберігають дані у форматі Text, space or tab separated і SAS; в форматі Excel – 9%, SPSS – 8%, S-Plus/R – 4%, Weka ARFF – 6%, в інших форматах інструментів Data Mining – 2%.

# ЛЕКЦІЯ 2
# МЕТОДИ І СТАДІЇ DATA MINING

## 2.1 Класифікація стадій Data Mining

Однією з ключових особливостей технології Data Mining є поєднання широкого спектра математичних методів (від класичного статистичного аналізу до сучасних кібернетичних підходів) із новітніми досягненнями в сфері інформаційних технологій. У Data Mining гармонійно поєднуються формалізовані методи та методи неформального аналізу, тобто кількісний і якісний аналіз даних.

До основних методів і алгоритмів Data Mining належать: штучні нейронні мережі, дерева рішень, символьні правила, методи найближчого сусіда і k-найближчих сусідів, метод опорних векторів, байєсовські мережі, лінійна регресія, кореляційно-регресійний аналіз, ієрархічні та неієрархічні методи кластерного аналізу (зокрема алгоритми k-середніх і k-медіан), методи пошуку асоціативних правил (наприклад, алгоритм Apriori), метод обмеженого перебору, еволюційне програмування, генетичні алгоритми, методи візуалізації даних та інші.

Більшість аналітичних методів, які застосовуються в Data Mining, базуються на вже відомих математичних алгоритмах. Новизна полягає у використанні цих методів для вирішення конкретних задач завдяки сучасним технічним і програмним засобам. Варто зазначити, що багато методів Data Mining були розроблені в межах теорії штучного інтелекту.

Метод – це встановлений спосіб або прийом вирішення теоретичних, практичних, пізнавальних чи управлінських задач.

Алгоритм – це чітка послідовність дій, яка перетворює вхідні дані на бажаний результат. Хоч поняття алгоритму виникло задовго до появи комп'ютерів, сьогодні вони є основою багатьох практичних і теоретичних рішень, зокрема в обчислювальних системах.

Data Mining може включати дві або три основні стадії:

Стадія 1. Виявлення закономірностей (вільний пошук / Discovery)

На цьому етапі досліджується набір даних без попередньо сформульованих гіпотез з метою виявлення прихованих закономірностей.

Закономірність – це суттєвий і стійкий взаємозв'язок, що характеризує розвиток певних явищ чи процесів [2].

У системах OLAP аналітик самостійно формулює запити для пошуку закономірностей, тоді як у Data Mining ця функція виконується автоматично, що особливо ефективно при роботі з великими обсягами даних.

Основні дії на цьому етапі:

- виявлення закономірностей умовної логіки (conditional logic);

- виявлення закономірностей асоціативної логіки (associations and affinities);
- виявлення трендів і коливань (trends and variations).

Приклад: У базі даних кадрового агентства система самостійно виявляє закономірності на кшталт:

- «Якщо вік < 20 років і бажаний рівень винагороди > 700 од., то у 75% випадків претендент шукає роботу програміста»;
- «Якщо вік > 35 років і бажаний рівень винагороди > 1200 од., то у 90% випадків шукають керівну посаду».

Механізми:

- індукція правил умовної логіки (класифікація та кластеризація);
- індукція правил асоціативної логіки (асоціативні та послідовні правила);
- виявлення трендів (початковий етап прогнозування).

На цьому етапі також виконується валідація – перевірка достовірності закономірностей на тестовій вибірці, що не брала участі у навчанні моделі (поширено у нейронних мережах та деревах рішень).

Стадія 2. Прогностичне моделювання (Predictive Modeling)

Цей етап базується на результатах вільного пошуку. Виявлені закономірності використовуються для прогнозування невідомих або майбутніх значень.

Основні дії:

- передбачення результатів (outcome prediction);
- прогнозування динаміки (forecasting).

Прогнозування здійснюється у двох основних формах: класифікація – віднесення об’єкта до певного класу на основі наявних даних, та прогнозування – визначення значення змінної на основі відомих трендів або закономірностей.

Індукція vs. Дедукція.

Вільний пошук – індуктивний підхід: від часткового до загального (на основі прикладів формуються загальні правила).

Прогностичне моделювання – дедуктивний підхід: від загального до часткового (на основі загальних правил робляться висновки щодо окремих випадків).

Закономірності можуть бути:

- прозорими (інтерпретованими), наприклад, правила «якщо..., то...»;
- непрозорими («чорними ящиками»), наприклад, штучні нейронні мережі.

Стадія 3. Аналіз винятків (Forensic Analysis)

Цей етап спрямований на виявлення аномалій або відхилень від виявлених закономірностей.

Основна дія: виявлення відхилень (deviation detection), що потребує попереднього

визначення норми.

Приклад: Якщо правило стверджує, що у 90% випадків претенденти з віком > 35 років і бажаною зарплатою > 1200 од. шукають керівну посаду – що з рештою 10%? Це можуть бути або логічно пояснювані винятки, або помилки в даних. У другому випадку стадія виняткового аналізу допомагає очистити дані.

### 2.2 Класифікація технологічних методів Data Mining

Всі методи Data Mining поділяються на дві великі групи за принципом роботи з вихідними навчальними даними. У цій класифікації верхній рівень визначається на підставі того, зберігаються дані після Data Mining чи вони дистилюються для подальшого використання.

Перша група передбачає безпосереднє використання даних, або збереження даних. У цьому випадку вихідні дані зберігаються в явному деталізованому вигляді і безпосередньо використовуються на стадіях прогностичного моделювання та/або аналізу винятків. Проблема цієї групи методів – при їх використанні можуть виникнути складності аналізу надвеликих баз даних.

Методи цієї групи: кластерний аналіз, метод найближчого сусіда, метод k – найближчого сусіда, міркування за аналогією.

У другій групі відбувається виявлення і використання формалізованих закономірностей, або дистиляція шаблонів.

При технології дистиляції шаблонів один зразок (шаблон) інформації витягується з вихідних даних і перетворюється в якісь формальні конструкції, вид яких залежить від використовуваного методу Data Mining. Цей процес виконується на стадії вільного пошуку, у першій же групі методів дана стадія в принципі відсутня. На стадіях прогностичного моделювання та аналізу винятків використовуються результати стадії вільного пошуку, вони значно компактніше самих баз даних. Нагадаємо, що конструкції цих моделей можуть бути трактовані аналітиком або не трактовані («чорні ящики»).

Методи цієї групи : логічні методи, методи візуалізації ; методи крос – табуляції; методи, засновані на рівняннях.

Логічні методи, або методи логічної індукції, включають: нечіткі запити і аналізи, символьні правила, дерева рішень, генетичні алгоритми.

Методи цієї групи є, мабуть, такими, що найкраще інтерпретуються – вони оформляють знайдені закономірності, в більшості випадків, у досить прозорому вигляді з точки зору користувача. Отримані правила можуть включати безперервні і дискретні змінні. Слід зауважити, що дерева рішень можуть бути легко перетворені в набори символьних правил

шляхом генерації одного правила по шляху від кореня дерева до його термінальної вершини. Дерева рішень і правила фактично є різними способами вирішення однієї задачі і відрізняються лише за своїми можливостями. Крім того, реалізація правил здійснюється більш повільними алгоритмами, ніж індукція дерев рішень [2].

Методи крос-табуляції: агенти, баєсовські (довірчі) мережі, крос – таблична візуалізація. Останній метод не зовсім відповідає одній з властивостей Data Mining – самостійного пошуку закономірностей аналітичною системою. Однак, надання інформації у вигляді крос – таблиць забезпечує реалізацію основного завдання Data Mining – пошук шаблонів, тому цей метод можна також вважати одним з методів Data Mining.

Методи на основі рівнянь. Методи цієї групи висловлюють виявлені закономірності у вигляді математичних виразів – рівнянь. Отже, вони можуть працювати лише з чисельними змінними, і змінні інших типів повинні бути закодовані відповідним чином. Це дещо обмежує застосування методів даної групи, проте вони широко використовуються при вирішенні різних завдань, особливо завдань прогнозування.

Основні методи цієї групи: статистичні методи і нейронні мережі.

Статистичні методи найбільш часто застосовуються для вирішення задач прогнозування. Існує безліч методів статистичного аналізу даних, серед них, наприклад, кореляційно – регресійний аналіз, кореляція рядів динаміки, виявлення тенденцій динамічних рядів, гармонійний аналіз.

Інша класифікація поділяє все різноманіття методів Data Mining на дві групи: статистичні та кібернетичні методи. Ця схема поділу заснована на різних підходах до навчання математичних моделей.

Слід зазначити, що існує два підходи віднесення статистичних методів до Data Mining. Перший з них протиставляє статистичні методи і Data Mining, його прихильники вважають класичні статистичні методи окремим напрямом аналізу даних. Відповідно до другого підходу, статистичні методи аналізу є частиною математичного інструментарію Data Mining. Більшість авторитетних джерел дотримується другого підходу.

У цій класифікації розрізняють дві групи методів:

- статистичні методи, засновані на використанні усередненого накопиченого досвіду, який відображений в ретроспективних даних;
- кібернетичні методи, що включають безліч різнорідних математичних підходів.

Недолік такої класифікації: і статистичні, і кібернетичні алгоритми тим чи іншим чином спираються на зіставлення статистичного досвіду з результатами моніторингу поточної ситуації.

Перевагою такої класифікації є її зручність для інтерпретації – вона використовується при описі математичних засобів сучасного підходу до вилучення знань з масивів вихідних

спостережень (оперативних і ретроспективних), тобто в задачах Data Mining.

Статистичні методи Data mining. Ці методи являють собою чотири взаємопов'язаних розділи:

- попередній аналіз природи статистичних даних (перевірка гіпотез стаціонарності, нормальності, незалежності, однорідності, оцінка виду функції розподілу, її параметрів тощо);
- виявлення зв'язків і закономірностей (лінійний і нелінійний регресійний аналіз, кореляційний аналіз та ін);
- багатовимірний статистичний аналіз (лінійний і нелінійний дискримінантний аналіз, кластерний аналіз, компонентний аналіз, факторний аналіз та ін);
- динамічні моделі і прогноз на основі часових рядів.

Кібернетичні методи Data Mining. Інший напрямок Data Mining – це безліч підходів, об'єднаних ідеєю комп'ютерної математики та використання теорії штучного інтелекту.

До цієї групи відносяться такі методи:

- штучні нейронні мережі (розпізнавання, кластеризація, прогнозування);
- еволюційне програмування (в т.ч. алгоритми методу групового обліку аргументів);
- генетичні алгоритми (оптимізація);
- асоціативна пам'ять (пошук аналогів, прототипів);
- нечітка логіка;
- дерева рішень;
- системи обробки експертних знань.

Методи Data Mining також можна класифікувати за задачами Data Mining. Відповідно до такої класифікації виділяємо дві групи. Перша з них – це підрозділ методів Data Mining на вирішальні завдання сегментації (тобто задачі класифікації та кластеризації) і завдання прогнозування.

У відповідності до другої класифікації по задачах методи Data Mining можуть бути спрямовані на отримання описових і прогнозуючих результатів.

Описові методи служать для знаходження шаблонів або зразків, що описують дані, які піддаються інтерпретації з точки зору аналітика. До методів, спрямованих на отримання описових результатів, відносяться ітеративні методи кластерного аналізу, в тому числі: алгоритм k – середніх, k – медіани, ієрархічні методи кластерного аналізу, самоорганізуються карти Кохонена, методи крос – табличної візуалізації, різні методи візуалізації та інші.

Прогнозуючі методи використовують значення одних змінних для передбачення / прогнозування невідомих (пропущених) або майбутніх значень інших (цільових) змінних. До методів, спрямованих на отримання прогнозуючих результатів, відносяться такі методи: нейронні мережі, дерева рішень, лінійна регресія, метод найближчого сусіда, метод опорних

векторів та ін.

### 2.3 Властивості методів Data Mining

Різні методи Data Mining характеризуються певними властивостями, які можуть бути визначальними при виборі методу аналізу даних. Методи можна порівнювати між собою, оцінюючи характеристики їх властивостей.

Серед основних властивостей і характеристик методів Data Mining розглянемо такі: точність, масштабованість, інтерпретованість, здатність до перевірки, трудомісткість, гнучкість, швидкість і популярність.

Масштабованість – властивість обчислювальної системи, яка забезпечує передбачуваний ріст системних характеристик, наприклад, швидкості реакції, загальної продуктивності та ін., при додаванні до неї обчислювальних ресурсів.

Більшість інструментів Data Mining, пропонованих зараз на ринку програмного забезпечення, реалізують відразу кілька методів, наприклад, дерева рішень, індукцію правил і візуалізацію, або ж нейронні мережі, самоорганізовувані карти Кохонена та візуалізацію. В універсальних прикладних статистичних пакетах (наприклад, SPSS, SAS, STATGRAPHICS, Statistica, ін) реалізується широкий спектр найрізноманітніших методів (як статистичних, так і кібернетичних). Слід враховувати, що для можливості їх використання, а також для інтерпретації результатів роботи статистичних методів (кореляційного, регресійного, факторного, дисперсійного аналізу та ін) потрібні спеціальні знання в галузі статистики.

Універсальність того чи іншого інструмента часто накладає певні обмеження на його можливості. Перевагою використання таких універсальних пакетів є можливість відносно легко порівнювати результати побудованих моделей, отримані різними методами. Така можливість реалізована, наприклад, в пакеті Statistica, де порівняння засноване на так званій «конкурентній оцінці моделей». Ця оцінка полягає в застосуванні різних моделей до одного і того ж набору даних і в наступному порівнянні їх характеристик для вибору найкращої з них.

Основні методи. Кілька основних методів, які використовуються для інтелектуального аналізу даних, описують тип аналізу і операцію з відновлення даних. На жаль, різні компанії і рішення не завжди використовують одні й ті ж терміни, що може посилити плутанину і складність, що здається.

Розглянемо деякі ключові методи і приклади того, як використовувати ті чи інші інструменти для інтелектуального аналізу даних.

Асоціація (або відношення), ймовірно, найбільш відомий, знайомий і простий метод інтелектуального аналізу даних. Для виявлення моделей робиться просте зіставлення двох або більше елементів, часто одного і того ж типу. Наприклад, відстежуючи звички покупця, можна помітити, що разом з полуницею зазвичай купують вершки.

Створити інструменти інтелектуального аналізу даних на базі асоціацій або відносин неважко. Наприклад, в InfoSphere Warehouse є майстер, який видає конфігурації інформаційних потоків для створення асоціацій, досліджуючи джерело вхідної інформації, базис прийняття рішень і вихідну інформацію. На рисунку 2.1. наведено відповідний приклад для зразка бази даних.

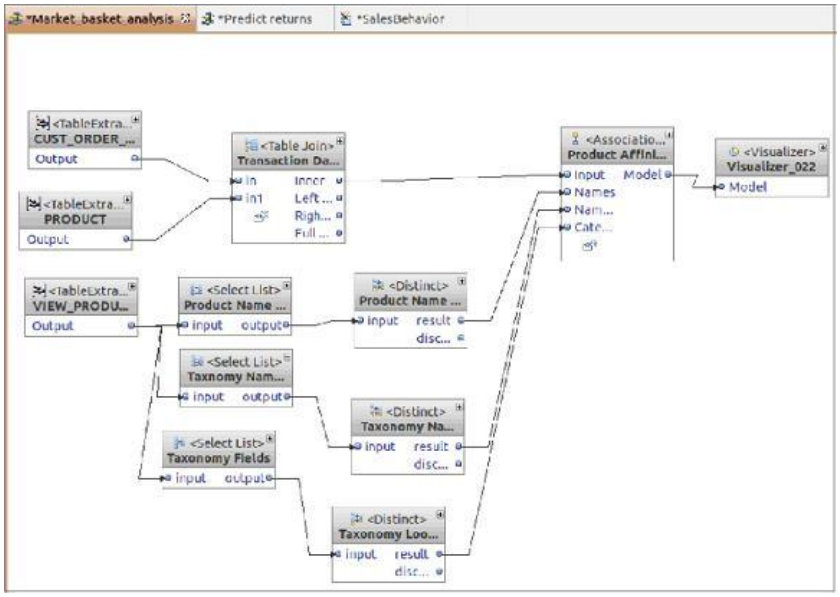



Рисунок 2.1 – Інформаційний потік, який використовується при підході асоціації

Класифікацію можна використовувати для отримання уявлення про тип покупців, товарів або об’єктів, описуючи кілька атрибутів для ідентифікації певного класу. Наприклад, автомобілі легко класифікувати по типу (седан, позашляховик, кабріолет), визначивши різні атрибути (кількість місць, форма кузова, ведучі колеса). Вивчаючи новий автомобіль, можна віднести його до певного класу, порівнюючи атрибути з відомим визначенням. Ті ж принципи можна застосувати і до покупців, наприклад, класифікуючи їх за віком та соціальною групою.

Крім того, класифікацію можна використовувати в якості вхідних даних для інших методів. Наприклад, для визначення класифікації можна застосовувати дерева прийняття рішень. Кластеризація дозволяє використовувати загальні атрибути різних класифікацій з метою виявлення кластерів.

Досліджуючи один або більше атрибутів або класів, можна згрупувати окремі елементи даних разом, отримуючи структурований висновок. На простому рівні при кластеризації використовується один або кілька атрибутів в якості основи для визначення кластера подібних результатів. Кластеризація корисна при визначенні різної інформації, тому що вона корелюється з іншими прикладами, так що можна побачити, де подібності і діапазони узгоджуються між собою.

Метод кластеризації працює в обидві сторони. Можна припустити, що в певній точці

мається кластер, а потім використовувати свої критерії ідентифікації, щоб перевірити це. Графік, зображений на рисунку 2.2., демонструє наочний приклад. Тут вік покупця порівнюється з вартістю покупки. Розумно очікувати, що люди у віці від двадцяти до тридцяти років (до вступу в шлюб і появи дітей), а також в 50-60 років (коли діти покинули будинок) мають більш високий наявний дохід.

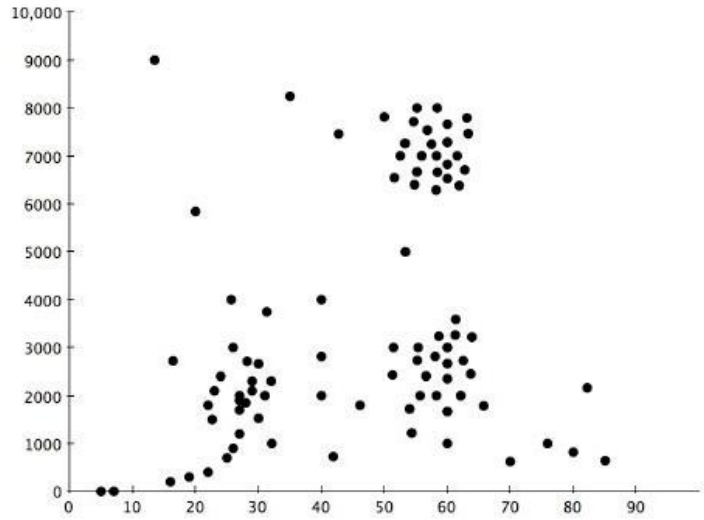



Рисунок 2.2 – Кластеризація

У цьому прикладі видно два кластери, один в районі $ 2000/20-30 років і інший в районі $7000-8000/50-65 років. У даному випадку ми висунули гіпотезу і перевірили її на простому графіку, який можна побудувати за допомогою будь-якого відповідного ПЗ для побудови графіків. Для більш складних комбінацій потрібен повний аналітичний пакет, особливо якщо потрібно автоматично засновувати рішення на інформації про найближчого сусіда.

Така побудова кластерів становить спрощений приклад так званого образу найближчого сусіда. Окремих покупців можна розрізняти за їх буквальною близькістю один до одного на графіку. Досить імовірно, що покупці з одного і того ж кластеру поділяють і інші загальні атрибути, і це припущення можна використовувати для пошуку, класифікації та інших видів аналізу членів набору даних.

Метод кластеризації можна застосувати і у зворотний бік: враховуючи певні вхідні атрибути, виявляти різні артефакти. Наприклад, недавнє дослідження чотиризначних PIN-кодів виявили кластери чисел у діапазонах 1-12 і 1-31 для першої та другої пар. Зобразивши ці пари на графіку, можна побачити кластери, пов’язані з датами (дні народження, ювілеї).

Прогнозування – це широка тема, яка простягається від передбачення відмов компонентів обладнання до виявлення шахрайства і навіть прогнозування прибутку компанії. У поєднанні з іншими методами інтелектуального аналізу даних прогнозування передбачає аналіз тенденцій, класифікацію, зіставлення з моделлю і відносини. Аналізуючи

минулі події або примірники, можна передбачати майбутнє.

Наприклад, використовуючи дані по авторизації кредитних карт, можна об’єднати аналіз дерева рішень минулих транзакцій людини з класифікацією і зіставленням з історичними моделями з метою виявлення шахрайських транзакцій. Якщо, наприклад, купівля авіаквитків збігається з транзакціями, то цілком імовірно, що ці транзакції справжні.

Послідовні моделі, які часто використовуються для аналізу довгострокових даних, – корисний метод виявлення тенденцій, або регулярних повторень подібних подій.

Наприклад, за даними про покупців можна визначити, що в різний час року вони купують певні набори продуктів. За цією інформацією додаток прогнозування купівельної корзини, ґрунтуючись на частоті та історії покупок, може автоматично припустити, що в кошик будуть додані ті чи інші продукти.

Дерево рішень, пов’язане з більшістю інших методів (головним чином, класифікації та прогнозування), можна використовувати або в рамках критеріїв відбору, або для підтримки вибору певних даних в рамках загальної структури. Дерево рішень починають з простого питання, яке має дві відповіді (іноді більше). Кожна відповідь призводить до наступного питання, допомагаючи класифікувати та ідентифікувати дані або робити прогнози.

Дерева рішень часто використовуються з системами класифікації інформації про властивості і з системами прогнозування, де різні прогнози можуть ґрунтуватися на минулому історичному досвіді, який допомагає побудувати структуру дерева рішень і отримати результат.

На практиці дуже рідко використовується тільки один з цих методів. Класифікація і кластеризація – подібні методи. Використовуючи кластеризацію для визначення найближчих сусідів, можна додатково уточнити класифікацію. Дерева рішень часто використовуються для побудови і виявлення класифікацій, які можна простежувати на історичних періодах для визначення послідовностей і моделей.

При всіх основних методах часто має сенс записувати і згодом вивчати отриману інформацію. Для деяких методів це абсолютно очевидно. Наприклад, при побудові послідовних моделей та навчанні з метою прогнозування аналізуються історичні дані з різних джерел і примірників інформації.

В інших випадках цей процес може бути більш яскраво вираженим. Дерева рішень рідко будуються один раз і ніколи не забуваються. При виявленні нової інформації, подій і точок даних може знадобитися побудова додаткових гілок або навіть зовсім нових дерев.

Деякі з цих процесів можна автоматизувати. Наприклад, побудова прогностичної моделі для виявлення шахрайства з кредитними картами зводиться до визначення ймовірностей, які можна використовувати для поточної транзакції, з подальшим оновленням цієї моделі при додаванні нових (підтверджених) транзакцій. Потім ця інформація

реєструється, так що наступного разу рішення можна буде прийняти швидше.

Підготовка даних і очищення даних – часто забувають, але надзвичайно важливий крок у процесі «видобутку даних». У типових проектах «видобутку даних» великі набори даних, зібрані за допомогою деяких автоматичних методів (наприклад, за допомогою Web), служать вхідними даними аналізу. Часто метод, за допомогою якого були зібрані дані, що не був жорстко регульованим, внаслідок чого дані можуть містити значення, що виходять за допустимі межі (наприклад, Дохід: – 100), неможливі комбінації даних (наприклад, Пол: Чоловік, Вагітність: Так) і інші. Аналіз даних, що не були ретельно екрановані на предмет подібних ситуацій, може дати вводять в оману результати, особливо при видобутку даних [2].

# ЛЕКЦІЯ 3
# ЗАДАЧІ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ

## 3.1 Завдання Data Mining

В основу технології Data Mining покладена концепція шаблонів, що представляють собою закономірності. В результаті виявлення цих, прихованих від неозброєного ока закономірностей вирішуються завдання інтелектуального аналізу даних. Різним типам закономірностей, які можуть бути виражені у формі, зрозумілій людині, відповідають певні завдання інтелектуального аналізу даних.

Завдання (tasks) Data Mining іноді називають закономірностями (regularity) або техніками (techniques).

Єдиної думки щодо того, які завдання слід відносити до Data Mining, немає.

Більшість авторитетних джерел перераховують наступні завдання: класифікація, кластеризація, прогнозування, асоціація, візуалізація, аналіз і виявлення відхилень, оцінювання, аналіз зв'язків, підведення підсумків.

Класифікація (Classification). Найбільш проста і поширена задача Data Mining. В результаті рішення задачі класифікації виявляються ознаки, які характеризують групи об'єктів досліджуваного набору даних – класи; за цими ознаками новий об'єкт можна віднести до того чи іншого класу.

Методи рішення. Для вирішення завдання класифікації можуть використовуватися методи: найближчого сусіда (Nearest Neighbor), К-найближчого сусіда (k-Nearest Neighbor); байєсовські мережі (Bayesian Networks); індукція дерев рішень ; нейронні мережі (neural networks).

Кластеризація (Clustering) є логічним продовженням ідеї класифікації. Це завдання більш складне, особливість кластеризації полягає в тому, що класи об'єктів спочатку не визначені. Результатом кластеризації є розбиття об'єктів на групи.

Приклад методу розв'язання задачі кластеризації: навчання «без вчителя» особливого виду нейронних мереж – самоорганізованих карт Кохонена.

Асоціація (Associations). У ході вирішення задачі пошуку асоціативних правил відшукуються закономірності між пов'язаними подіями в наборі даних.

Відмінність асоціації від двох попередніх завдань Data Mining: пошук закономірностей здійснюється не на основі властивостей аналізованого об'єкта, а між кількома подіями, які відбуваються одночасно.

Найбільш відомий алгоритм вирішення задачі пошуку асоціативних правил – алгоритм Apriori.

Послідовність (Sequence), або послідовна асоціація (sequential association).

Послідовність дозволяє знайти тимчасові закономірності між транзакціями. Задача послідовності подібна асоціації, але її метою є встановлення закономірностей не між одночасно наступаючими подіями, а між подіями, пов'язаними в часі (тобто відбуваються з деяким певним інтервалом у часі). Іншими словами, послідовність визначається високою ймовірністю ланцюжка пов'язаних у часі подій.

Фактично, асоціація є окремим випадком послідовності з тимчасовим лагом, рівним нулю. Цю задачу Data Mining також називають задачею знаходження послідовних шаблонів (sequential pattern).

Правило послідовності: після події X через певний час відбудеться подія Y.

Приклад. Після покупки квартири мешканці в 60% випадків протягом двох тижнів купують холодильник, а протягом двох місяців в 50% випадків купується телевізор. Рішення даної задачі широко застосовується в маркетингу і менеджменті, наприклад, при управлінні циклом роботи з клієнтом (управління життєвим циклом клієнта).

Прогнозування (Forecasting). В результаті рішення задачі прогнозування на основі особливостей історичних даних оцінюються пропущені або ж майбутні значення цільових чисельних показників.

Для вирішення таких завдань широко застосовуються методи математичної статистики, нейронні мережі та ін.

Визначення відхилень або викидів (Deviation Detection), аналіз відхилень або викидів. Мета розв'язання даної задачі – виявлення та аналіз даних, що найбільш відрізняються від загальної множини даних, виявлення так званих нехарактерних шаблонів.

Оцінювання (оцінка). Завдання оцінювання зводиться до передбачення безперервних значень ознаки.

Аналіз зв'язків (Link Analysis) – задача знаходження залежностей в наборі даних.

Візуалізація (Visualization, Graph Mining). В результаті візуалізації створюється графічний образ аналізованих даних. Для вирішення задачі візуалізації використовуються графічні методи, що показують наявність закономірностей в даних.

Приклад методів візуалізації – подання даних у 2-D і 3-D вимірах.

Підведення підсумків (Summarization) – завдання, мета якого – опис конкретних груп об'єктів з аналізованого набору даних.

### 3.2 Задачі інтелектуального аналізу даних

Згідно класифікації за стратегіями, задачі Data Mining поділяються на такі групи: навчання з учителем, навчання без вчителя, інші.

Категорія навчання з учителем представлена наступними задачами Data Mining:

класифікація, оцінка, прогнозування.

Категорія навчання без вчителя представлена задачею кластеризації.

У категорію «інші» входять задачі, не включені в попередні дві стратегії.

Задачі інтелектуального аналізу даних, залежно від використовуваних моделей, можуть бути дескриптивними і прогнозуючими.

Відповідно до цієї класифікації, задачі Data Mining представлені групами описових і прогнозуючих завдань.

У результаті рішення описових (descriptive) задач аналітик отримує шаблони, що описують дані, які піддаються інтерпретації.

Ці задачі описують загальну концепцію аналізованих даних, визначають інформативні, підсумкові, відмінні особливості даних. Концепція описових завдань передбачає характеристику і порівняння наборів даних.

Характеристика набору даних забезпечує короткий і стислий опис деякого набору даних.

Порівняння забезпечує порівняльний опис двох або більше наборів даних.

Прогнозуючі (predictive) ґрунтуються на аналізі даних, створенні моделі, передбаченні тенденцій або властивостей нових або невідомих даних.

Досить близьким до вищезгаданої класифікації є розділення задач Data Mining на наступні: дослідження та відкриття, прогнозування та класифікація, пояснення і опис.

Автоматичне дослідження і відкриття (вільний пошук). Приклад задачі: виявлення нових сегментів ринку.

Для вирішення даного класу задач використовуються методи кластерного аналізу прогнозування та класифікація.

Приклад задачі: передбачення зростання обсягів продажів на основі поточних значень.

Методи: регресія, нейронні мережі, генетичні алгоритми, дерева рішень.

Задачі класифікації та прогнозування становлять групу так званого індуктивного моделювання, в результаті якого забезпечується вивчення аналізованого об’єкта або системи. У процесі вирішення цих завдань на основі набору даних розробляється загальна модель або гіпотеза.

Пояснення й опис. Приклад задачі: характеристика клієнтів за демографічними даними і історіями покупок.

Методи: дерева рішень, системи правил, правила асоціації, аналіз зв’язків.

Якщо дохід клієнта більше, ніж 50 умовних одиниць, і його вік – понад 30 років, тоді клас клієнта – перший.

В інтерпретації узагальненої моделі аналітик отримує нове знання. Групування об’єктів відбувається на основі їх подібності.

Отже, у попередній лекції нами були розглянуті методи Data Mining і дії, що виконуються в рамках стадій інтелектуального аналізу даних. Щойно ми розглянули основні завдання інтелектуального аналізу даних.

Нагадаємо, що головна цінність Data Mining – це практична спрямованість даної технології, шлях від сирих даних до конкретного знання, від постановки завдання до готового додатку, за підтримки якого можна приймати рішення.

Велика кількість понять, які об'єдналися в Data Mining, а також різноманітність методів, що підтримують дану технологію, починаючому аналітику можуть нагадати мозаїку, частини якої мало пов'язані між собою [3].

Як же ми можемо зв'язати в одне ціле задачі, методи, дії, закономірності, додатки, дані, інформацію, рішення?

Розглянемо два потоки:

- дані – інформація – знання і рішення;
- завдання – дії і методи рішення – програми.

Ці потоки є «двома сторонами однієї медалі», відображенням одного процесу, результатом якого має бути знання і прийняття рішення.

Від даних до рішень. Для початку розглянемо перший потік. На рисунку 3.1. показаний зв'язок понять «дані», «інформація» і «рішення», яка виникає в процесі прийняття рішень.

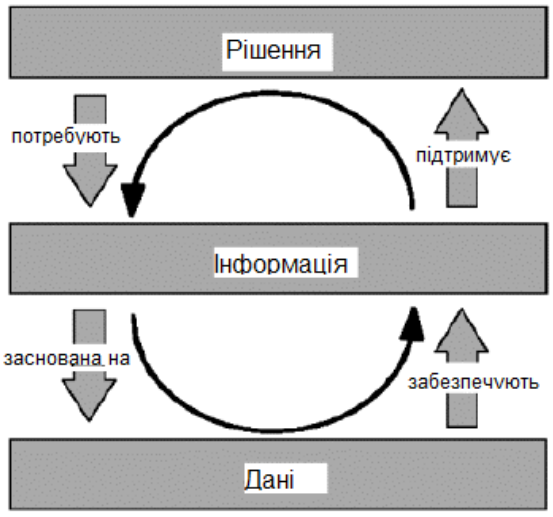



Рисунок 3.1 – Рішення, інформація і дані

Як видно з рисунку, даний процес являється циклічним. Прийняття рішень потребує інформації, яка заснована на даних. Дані забезпечують інформацію, яка підтримує рішення і т.д.

Розглянуті поняття є складовою частиною так званої інформаційної піраміди, в основі якої знаходяться дані, наступний рівень – це інформація, потім йде рішення, завершує піраміду

рівень знання. У міру просування вгору по інформаційній піраміді обсяги даних переходять у цінність рішень, тобто цінність для бізнесу. А, як відомо, метою Business Intelligence є перетворення обсягів даних у цінність бізнесу.

### 3.3 Рівні аналізу

Верхній – рівень додатків – є рівнем бізнесу (якщо ми маємо справу із завданням бізнесу), на ньому менеджери приймають рішення. Наведені приклади додатків: перехресні продажі, контроль якості, утримування клієнтів.

Середній – рівень дій – за своєю суттю є рівнем інформації, саме на ньому виконуються дії Data Mining; на рисунку наведені такі дії: прогностичне моделювання, аналіз зв'язків, сегментація даних та інші.

Нижній – рівень визначення задачі інтелектуального аналізу даних, яку необхідно розв'язати стосовно даних, що є в наявності, на малюнку наведені завдання передбачення числових значень, класифікація, кластеризація, асоціація.

Розглянемо таблицю 3.1, що демонструє зв'язок цих понять.

Таблиця 3.1 Рівні Data Mining

| Рівень 3 | Додатки | Утримання клієнтів | Знання dm | Результат |
|---|---|---|---|---|
| Рівень 2 | Дії | Прогностичне моделювання | Інформація | Метод аналізу |
| Рівень 1 | Задачі | Класифікація | Дані | Запити |

Нагадаємо, що для вирішення задачі класифікації результати роботи першої стадії (індукції правил) використовуються для віднесення нового об'єкта, з певною впевненістю, до одного з відомих, визначених класів на підставі відомих значень.

Розглянемо задачі утримання клієнтів (визначення надійності клієнтів фірми).

Дані – база даних по клієнтах. Є дані про клієнта (вік, стать, професія, дохід). Певна частина клієнтів, скориставшись продуктом фірми, залишилася їй вірна; інші клієнти більше не купували продукти фірми. На цьому рівні ми визначаємо тип завдання – це завдання класифікації.

На другому рівні визначаємо дію – прогностичне моделювання. За допомогою прогностичного моделювання ми з певною частиною впевненості можемо віднести новий об'єкт, в даному випадку, нового клієнта, до одного з відомих класів – постійний клієнт, або це, швидше за все, його разова покупка.

На третьому рівні ми можемо скористатися додатком для прийняття рішення. У результаті придбання знань, фірма може істотно знизити витрати, наприклад, на рекламу,

знаючи заздалегідь, яким із клієнтів слід активно розсилати рекламні матеріали.

Таким чином, протягом кількох лекцій ми визначилися з поняттями «дані», «завдання», «методи», «дії ».

Зараз зупинимося на ще не розглянутому понятті інформації. Незважаючи на поширеність даного поняття, ми не завжди можемо точно його визначити і відрізнити від поняття даних. Інформація, за своєю суттю, має багатогранну природу. З розвитком людства, в тому числі, з розвитком комп'ютерних технологій, інформація набуває все нових і нових властивостей.

### 3.4 Інформація. Властивості інформації

Інформація (від лат. informatio) – це:

- будь-яке повідомлення про щось;
- відомості, що є об'єктом зберігання, обробки та передачі (наприклад, генетична інформація);
- у математиці та кібернетиці – кількісна міра зменшення невизначеності (ентропія), міра організованості системи;
- у теорії інформації – розділ кібернетики, що досліджує закономірності збирання, передачі, обробки та обчислення інформації.

У загальному сенсі інформація – це нові для отримувача відомості про події, об'єкти або процеси, які можуть бути інтерпретовані, передані, збережені, перетворені або використані. Існування інформації можливе лише за наявності джерела, одержувача та каналу зв'язку між ними.

Розглянемо властивості інформації.

Повнота – достатність даних для ухвалення рішень. Наприклад: «продажі товару А скорочуються» через відсутність конкретики дана інформація є неповною, а «з першого кварталу продажі товару А почнуть скорочуватися» є прикладом повної інформації з конкретними часовими рамками.

Достовірність – відповідність дійсності, рівень наявності інформаційного шуму.

Цінність – інформація має бути корисною для конкретного користувача, не може бути абстрактною.

Адекватність – відображення реального стану речей. Адекватна інформація є повною та достовірною.

Актуальність – відповідність поточному моменту часу. Застаріла інформація втрачає значення.

Ясність – зрозумілість для цільової аудиторії.

Доступність – можливість отримання інформації за допомогою доступних джерел і методів.

Суб'єктивність – залежність змісту та інтерпретації інформації від сприйняття одержувача.

Вимоги до інформації.

Динамічність – інформація існує лише в момент інформаційного процесу, в інший час – це лише дані.

Адекватність методів обробки – різна інформація може бути отримана з одних і тих самих даних залежно від методів аналізу.

Дані є об'єктивними, методи – суб'єктивними. Інформація виникає в процесі взаємодії даних і методів.

У контексті бізнесу інформація є основою для прийняття рішень. Її поділяють:

- за джерелом: внутрішня (зсередини організації) та зовнішня (зовнішні фактори, що впливають на бізнес);
- за характером: фактична (відображає доконані події) та прогнозна (передбачення на основі аналізу).

Що таке знання?

Знання – це сукупність фактів, закономірностей, правил та досвіду, які дозволяють вирішувати завдання. Формуються в результаті обробки інформації.

За визначенням Денхема Грея: «Знання – це абсолютне використання інформації та даних разом із практичним досвідом, здібностями, інтуїцією, переконаннями й мотивацією людини».

Властивості знань:

- структурованість – організованість за логічною схемою;
- зручність доступу та засвоєння – легкість розуміння й запам'ятовування;
- лаконічність – стисло викладений зміст без зайвої інформації;
- несуперечливість – відсутність логічних протиріч між знаннями.

Процедури обробки – можливість аналізу, передачі й використання знань (для IT-систем – у вигляді спеціальних форматів).

Розглянемо взаємозв'язок: дані – інформація – знання.

Для розуміння різниці між поняттями розглянемо приклад:

- дані – числа, факти (наприклад, оцінки на іспиті).
- інформація – пояснення, як ці дані стосуються теми (наприклад, конспект лекції).
- знання – розуміння теми, здатність вирішувати завдання (успішне складання іспиту).

Інформація, на відміну від даних, має сенс . Знання в свою чергу є осмисленою та

цінною інформацією.

Піраміда обробки: від даних до знань: дані → обробка → інформація → аналіз і осмислення → знання

Великий обсяг даних не гарантує знань – потрібні якісні алгоритми обробки.

Наприклад, текст іноземною мовою – це інформація. Без перекладача – вона не стає знанням. З перекладачем – процес осмислення можливий.

### 3.5 Задачі класифікації

Класифікація є найбільш простою і водночас найбільш часто розв'язуваною задачею Data Mining. Зважаючи на поширеність задач класифікації необхідно чітке розуміння суті цього поняття.

Наведемо кілька визначень.

Класифікація – системний розподіл досліджуваних предметів, явищ, процесів за родами, видами, типами, з якими-небудь істотними ознаками для зручності їх дослідження; угрупування вихідних понять і розташування їх у певному порядку, що відбиває ступінь цієї схожості.

Класифікація – впорядкована за деяким принципом множина об'єктів, які мають подібні класифікаційні ознаки (одна або декілька властивостей), обраних для визначення схожості або відмінності між цими об'єктами.

Класифікація вимагає дотримання наступних правил:

- в кожному акті ділення необхідно застосовувати тільки одну основу;
- ділення повинне бути пропорційним, тобто загальний обсяг видових понять повинен дорівнювати об'єму діленого родового поняття;
- члени ділення повинні взаємно виключати один одного, їх об'єми не повинні перехрещуватися;
- ділення повинне бути послідовним.

Розрізняють:

- допоміжну (штучну) класифікацію, яка виробляється за зовнішньою ознакою і служить для надання множині предметів (процесів, явищ) потрібного порядку;
- природну класифікацію, яка виробляється за істотними ознаками, що характеризують внутрішню спільність предметів і явищ. Вона є результатом і важливим засобом наукового дослідження, тому що передбачає і закріплює результати вивчення закономірностей об'єктів, що класифікуються [3].

Допоміжна класифікація створюється з метою найбільш швидкого відшукування якогось індивідуального предмету серед предметів, що класифікуються. Мета в цій

класифікації визначає принцип її побудови. В основу допоміжної класифікації лягає яка-небудь зовнішня несуттєва ознака, яка, однак, виявляється корисною у процесі пошуку.

Прикладами допоміжної класифікації можуть бути розподіл студентів курсу в списку в алфавітному порядку або такий же розподіл бібліотечних карток в алфавітному каталозі і т.п. Знаючи порядок букв в алфавіті, ми можемо легко і швидко відшукати потрібне нам прізвище у списку або дані, що цікавлять нас в книзі, в каталозі.

Але знання того, яке місце в допоміжній класифікаційній системі займає той чи інший предмет, не дає можливості щось стверджувати про його властивості. Так, наприклад, те що студент Архипов записаний у списку першим, а студент Яковлєв – останнім, нічого не говорить про їх здібності і риси характеру . Тому допоміжна класифікація не є науковою.

На відміну від допоміжної природна класифікація становить розподіл предметів за класами на підставі їх найбільш суттєвих ознак. Найбільш істотними є такі ознаки предмета, які обумовлюють інші його ознаки. Наприклад, найбільш суттєвою ознакою людини є її здатність до праці. Ця ознака зумовлює наявність у людини таких ознак, як прямоходіння, здатність до спілкування (праця передбачає колектив), здатність до мислення та ін.

Залежно від обраних ознак, їх поєднання і процедури розподілу понять, класифікація може бути:

- простою – розподіл родового поняття тільки за ознакою і тільки один раз до розкриття всіх видів. Прикладом такої класифікації є дихотомія, при якій членами поділу бувають тільки два поняття, кожне з яких суперечить іншому (тобто дотримується принцип: «А і не А»);
- складною – застосовується для поділу одного поняття за різними основами і синтезу таких простих ділень в єдине ціле.

Прикладом такої класифікації є періодична система хімічних елементів.

Під класифікацією будемо розуміти віднесення об’єктів (спостережень, подій) до одного з заздалегідь відомих класів.

Класифікація – це закономірність, що дозволяє робити висновок щодо визначення характеристик конкретної групи. Таким чином, для проведення класифікації повинні бути присутні ознаки, що характеризують групу, до якої належить та чи інша подія або об’єкт (зазвичай при цьому на підставі аналізу вже класифікованих подій формулюються якісь правила).

Класифікація відноситься до стратегії навчання з вчителем (supervised learning), яку також іменують контрольованим або керованим навчанням.

Машинне навчання – узагальнена назва штучної генерації знань з досвіду. Штучна система навчається на прикладах і після закінчення фази навчання може узагальнювати. Тобто система не просто вивчає наведені приклади, а розпізнає певні закономірності в даних для

навчання.

Серед багатьох програмних продуктів варто згадати системи автоматичного діагностування, розпізнавання шахрайства з кредитними картками, аналіз ринку цінних паперів, класифікація ланцюжків ДНК, розпізнавання мовлення та тексту, автономні системи.

Практичне використання відбувається, переважно, за допомогою алгоритмів. Різноманітні алгоритми машинного навчання можна грубо поділити за такою схемою:

Навчання з вчителем (англ. Supervised learning): алгоритм вивчає функцію на основі наданих пар вхідних та вихідних даних. При цьому, в процесі навчання, «вчитель» вказує вірні вихідні дані для кожного значення вхідних даних. Одним з розділів навчання з вчителем є машинна класифікація. Такі алгоритми застосовуються для розпізнавання текстів.

Навчання без вчителя (англ. Unsupervised learning).

Навчання з закріпленням (англ. Reinforcement Learning): алгоритм навчається за допомогою тактики нагороди та покарання для максимізації вигоди для агентів (систем до яких належить компонента, що навчається)

Завданням класифікації часто називають передбачення категоріальної залежної змінної (тобто залежної змінної, що є категорією) на основі вибірки безперервних і/або категоріальних змінних.

Наприклад, можна передбачити, хто з клієнтів фірми є потенційним покупцем певного товару, а хто – ні, хто скористається послугою фірми, а хто – ні, і т.д. Цей тип завдань належить до завдань бінарної класифікації, в них залежна змінна може приймати тільки два значення (наприклад, так чи ні, 0 або 1).

Інший варіант класифікації виникає, якщо залежна змінна може приймати значення з деякої множини визначених класів. Наприклад, коли необхідно передбачити, яку марку автомобіля захоче купити клієнт. У цих випадках розглядається множина класів для залежної змінної.

Класифікація може бути одновимірною (за однією ознакою) і багатовимірною (за двома і більше ознаками).

Багатовимірна класифікація була розроблена біологами при вирішенні проблем дискримінації для класифікування організмів. Однією з перших робіт, присвячених цьому напрямку, вважають роботу Р. Фішера (1930 р.), в якій організми поділялися на підвиди залежно від результатів вимірювань їх фізичних параметрів. Біологія була і залишається найбільш затребуваним і зручним середовищем для розробки багатовимірних методів класифікації.

Розглянемо задачу класифікації на простому прикладі. Припустимо, є база даних про клієнтів туристичного агентства з інформацією про вік і доході за місяць. Є рекламний матеріал двох видів: більш дорогий і комфортний відпочинок і дешевший, молодіжний

відпочинок. Відповідно, визначені два класи клієнтів: клас 1 і клас 2. База даних наведена в таблиці 3.2.

Таблиця 3.2 База даних клієнтів туристичного агентства.

| Код клієнта | Вік | Дохід | Клас |
|---|---|---|---|
| 1 | 18 | 25 | 1 |
| 2 | 22 | 100 | 1 |
| 3 | 30 | 70 | 1 |
| 4 | 32 | 120 | 1 |
| 5 | 24 | 15 | 2 |
| 6 | 25 | 22 | 1 |
| 7 | 32 | 50 | 2 |
| 8 | 19 | 45 | 2 |
| 9 | 22 | 75 | 1 |
| 10 | 40 | 90 | 2 |

Завдання. Визначити, до якого класу належить новий клієнт і який з двох видів рекламних матеріалів йому варто відсилати.

Для наочності представимо нашу базу даних у двомірному просторі (вік і дохід), у вигляді множини об’єктів, що належать класам 1 (помаранчева мітка) і 2 (сіра мітка). На рисунку 3.3 наведені об’єкти з двох класів [3].

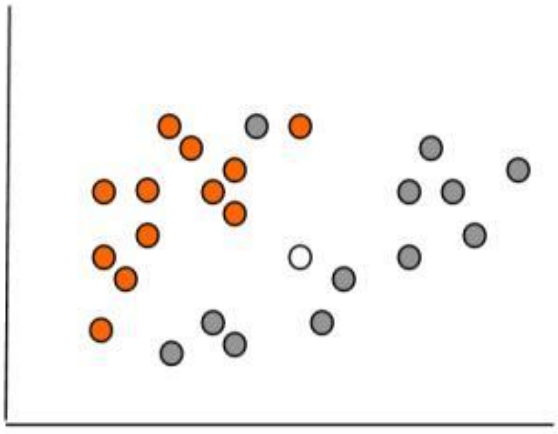


Рисунок 3.2 – Множина об’єктів бази даних у двомірному вимірі

Розв’язок нашої задачі буде полягати в тому, щоб визначити, до якого класу належить новий клієнт, на малюнку позначений білою міткою.

Мета процесу класифікації полягає в тому, щоб побудувати модель, яка використовує прогнозуючі атрибути в якості вхідних параметрів і отримує значення залежного атрибута.

Процес класифікації полягає в розбитті множини об’єктів на класи за певним критерієм.

Класифікатором називається якась сутність, що визначає, якому з визначених класів належить об’єкт за вектором ознак.

Для проведення класифікації за допомогою математичних методів необхідно мати формальний опис об’єкта, яким можна оперувати, використовуючи математичний апарат класифікації. Таким описом у нашому випадку виступає база даних. Кожен об’єкт (запис бази даних) несе інформацію про деякі властивості об’єкта.

Набір вихідних даних (або вибірку даних) розбивають на дві множини: навчальну і тестову.

Навчальна множина (training set) – множина, яка включає дані, що використовуються для навчання (конструювання) моделі.

Така множина містить вхідні та вихідні (цільові) значення прикладів. Вихідні значення призначені для навчання моделі.

Тестова (test set) множина також містить вхідні та вихідні значення прикладів. Тут вихідні значення використовуються для перевірки працездатності моделі.

Процес класифікації складається з двох етапів: конструювання моделі та її використання.

Конструювання моделі: опис множини визначених класів. Кожен приклад набору даних відноситься до одного визначеного класу. На цьому етапі використовується навчальна множина, на ньому відбувається конструювання моделі. Отримана модель представлена класифікаційними правилами, деревом рішень або математичною формулою.

Використання моделі: класифікація нових або невідомих значень. Оцінка правильності (точності) моделі. Відомі значення з тестового прикладу порівнюються з результатами використання отриманої моделі.

Рівень точності – відсоток правильно класифікованих прикладів у тестовій множині. Тестова множина, тобто множина, на якій тестується побудована модель, не повинна залежати від навчальної множини. Якщо точність моделі допустима, можливе використання моделі для класифікації нових прикладів, клас яких невідомий.

### 3.6 Методи, що застосовуються для вирішення задач класифікації

Для класифікації використовуються різні методи. Основні з них:

- класифікація за допомогою дерев рішень;
- байєсівська (наївна) класифікація;
- класифікація за допомогою штучних нейронних мереж;
- класифікація методом опорних векторів;
- статистичні методи, зокрема, лінійна регресія;

- класифікація за допомогою методу найближчого сусіда;
- класифікація cbr – методом;
- класифікація за допомогою генетичних алгоритмів.

Схематичне рішення задачі класифікації деякими методами (за допомогою лінійної регресії, дерев рішень і нейронних мереж) наведені на рисунках 3.3 – 3.5.

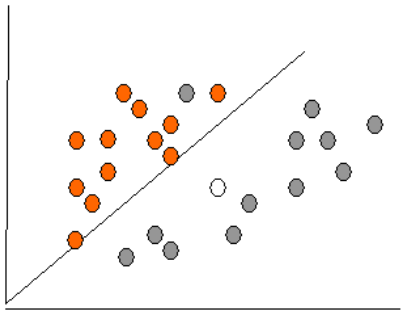


Рисунок 3.3 – Рішення задачі класифікації методом лінійної регресії

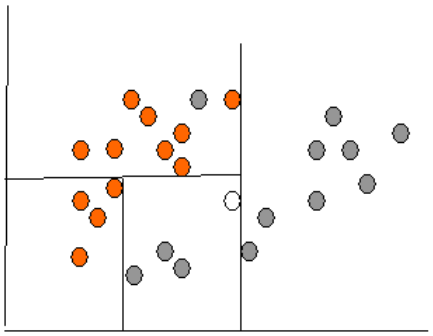


Рисунок 3.4 – Рішення задачі класифікації методом дерев рішень

Рисунок 3.5 – Рішення задачі класифікації методом нейронних мереж

Точність класифікації: оцінка рівня помилок. Оцінка точності класифікації може проводитися за допомогою крос-перевірки. Крос-перевірка (Cross-validation) – це процедура оцінки точності класифікації на даних з тестової множини, яку також називають крос-перевірочною множиною. Точність класифікації тестової множини порівнюється з точністю класифікації навчаючої множини. Якщо класифікація тестової множини дає приблизно такі ж

результати по точності, як і класифікація навчальної множини, вважається, що дана модель пройшла крос-перевірку.

Поділ на навчальну і тестову множину здійснюється шляхом ділення вибірки в певній пропорції, наприклад навчальна множина – дві третини даних і тестова – одна третина даних. Цей спосіб слід використовувати для вибірок з великою кількістю прикладів. Якщо ж вибірка має малі обсяги, рекомендується застосовувати спеціальні методи, при використанні яких навчальна і тестова вибірки можуть частково перетинатися.

Оцінювання класифікаційних методів. Оцінювання методів слід проводити, виходячи з таких характеристик: швидкість, робастність, інтерпретованість, надійність.

Швидкість характеризує час, який потрібен на створення моделі та її використання.

Робастність, тобто стійкість до будь-яких порушень вихідних передумов, означає можливість роботи з зашумленими даними і пропущеними значеннями в даних.

Інтерпретованість забезпечує можливість розуміння моделі аналітиком.

Властивості класифікаційних правил:

- розмір дерева рішень;
- компактність класифікаційних правил.

### 3.7 Завдання кластеризації

Тільки що ми вивчили завдання класифікації, що відноситься до стратегії «навчання з учителем».

У цій частині лекції ми введемо поняття кластеризації, кластера, коротко розглянемо класи методів, за допомогою яких вирішується завдання кластеризації, деякі моменти процесу кластеризації, а також розберемо приклади застосування кластерного аналізу.

Завдання кластеризації схоже з завданням класифікації, є його логічним продовженням, але його відмінність в тому, що класи досліджуваного набору даних заздалегідь не зумовлені.

Синонімами терміну «кластеризація» є «автоматична класифікація», «навчання без вчителя» і «таксономія».

Кластеризація призначена для розбиття сукупності об’єктів на однорідні групи (кластери або класи). Якщо дані вибірки представити як точки в просторі ознак, то завдання кластеризації зводиться до визначення «згущувань точок».

Мета кластеризації – пошук існуючих структур. Кластеризація є описовою процедурою, вона не робить ніяких статистичних висновків, але дає можливість провести розвідувальний аналіз і вивчити «структуру даних».

Саме поняття «кластер» визначене неоднозначно. Перекладається поняття кластер (cluster) як «скупчення», «гроно».

Кластер можна охарактеризувати як групу об’єктів, що мають загальні властивості.

Характеристиками кластера можна назвати дві ознаки:

- внутрішня однорідність;
- зовнішня ізольованість.

Питання, що ставиться аналітиками при вирішенні багатьох завдань, полягає в тому, як організувати дані в наочні структури, тобто розгорнути таксономії [4].

Найбільше застосування кластеризація спочатку отримала в таких науках як біологія, антропологія, психологія. Для вирішення економічних завдань кластеризація тривалий час мало використовувалася через специфіку економічних даних і явищ.

У таблиці 3.3 наведено порівняння деяких параметрів задач класифікації та кластеризації.

Таблиця 3.3 Порівняння класифікації та кластерзації

| Характеристика | Класифікація | Кластеризація |
|---|---|---|
| Контрольованість навчання | Контрольоване навчання | Неконтрольоване навчання |
| Стратегія | Навчання з вчителем | Навчання без вчителя |
| Наявність позначки класу | Навчальна множина супроводжується міткою, що вказує клас, до якого належить спостереження | Мітки класу навчальної множини невідомі |
| Підстава для класифікації | Нові дані класифікуються на підставі навчальної множини | Дано множину даних з метою встановлення існування класів або кластерів даних |

На рисунку 3.6 схематично представлені завдання класифікації і кластеризації.

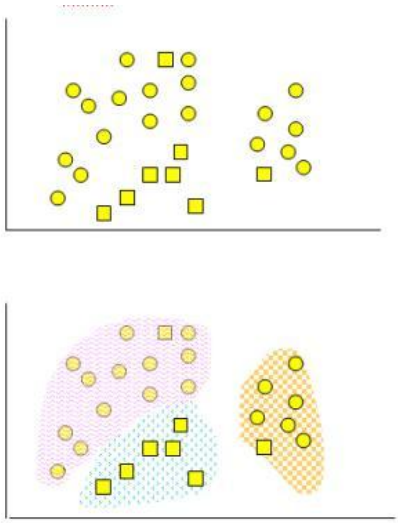


Рисунок 3.6 – Порівняння задач класифікації та кластеризації

Кластери можуть бути такими, що не перетинаються, або ексклюзивними (non-overlapping, exclusive), і такими, що перетинаються (overlapping). Схематичне зображення таких кластерів дано на рисунку 3.7.

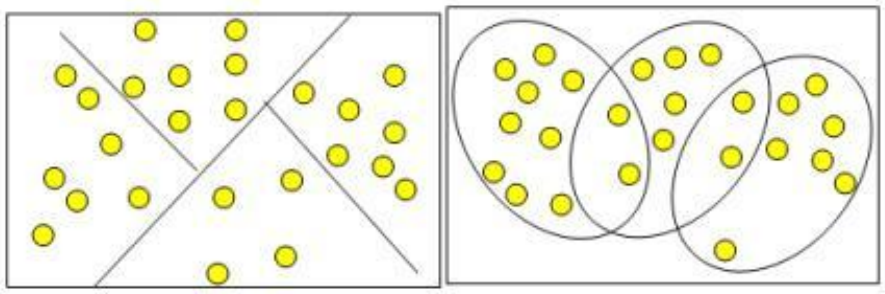


Рисунок 3.7 – Кластери, що не перетинаються і перетинаються

Слід зазначити, що в результаті застосування різних методів кластерного аналізу можуть бути отримані кластери різної форми. Наприклад, можливі кластери «ланцюжкового» типу, коли кластери представлені довгими «ланцюжками», кластери подовженої форми і т.д., а деякі методи можуть створювати кластери довільної форми.

Різні методи можуть прагнути створювати кластери певних розмірів (наприклад, малих або великих) або припускати в наборі даних наявність кластерів різного розміру.

Деякі методи кластерного аналізу особливо чутливі до шумів або викидів, інші – менш.

В результаті застосування різних методів кластеризації можуть бути отримані неоднакові результати, це нормально і є особливістю роботи того чи іншого алгоритму.

Дані особливості слід враховувати при виборі методу кластеризації.

Детальніше про всі властивості кластерного аналізу буде розказано в лекції, присвяченій його методам.

На сьогоднішній день розроблено більше сотні різних алгоритмів кластеризації. Деякі, найбільш часто використовувані, будуть детально описані в наступних лекціях.

Наведемо коротку характеристику підходів до кластеризації.

Алгоритми, засновані на поділі даних (Partitioning algorithms), в тому числі ітеративні: поділ об’єктів на k кластерів, ітеративний перерозподіл об’єктів для поліпшення кластеризації.

Ієрархічні алгоритми (Hierarchy Algorithms): агломерація – кожен об’єкт спочатку є кластером, кластери, з’єднуючись один з одним, формують більший кластер і т.д.

Методи, засновані на концентрації об’єктів (Density-based methods): засновані на можливості з’єднання об’єктів, ігнорують шуми, знаходження кластерів довільної форми.

Грід-методи (Grid-based methods): квантування об’єктів в грід-структури.

Модельні методи (Model-based): використання моделі для знаходження кластерів,

найбільш відповідних даним.

Оцінка якості кластеризації може бути проведена на основі таких процедур:

- ручна перевірка;
- встановлення контрольних точок та перевірка на отриманих кластерах;
- визначення стабільності кластеризації шляхом додавання в модель нових змінних;
- створення і порівняння кластерів з використанням різних методів. Різні методи кластеризації можуть створювати різні кластери, і це є нормальним явищем. Однак створення схожих кластерів різними методами вказує на правильність кластеризації.

### 3.8 Застосування кластерного аналізу

Кластерний аналіз застосовується в різних областях. Він корисний, коли потрібно класифікувати велику кількість інформації. Огляд багатьох опублікованих досліджень, що проводяться за допомогою кластерного аналізу, дав Хартіган (Hartigan, 1975).

Так, в медицині використовується кластеризація захворювань, лікування захворювань або їх симптомів, а також таксономія пацієнтів, препаратів і т.д. В археології встановлюються таксономії кам’яних споруд і древніх об’єктів і т.д. У маркетингу це може бути задача сегментації конкурентів і споживачів. У менеджменті прикладом завдання кластеризації буде розбиття персоналу на різні групи, класифікація споживачів і постачальників, виявлення схожих виробничих ситуацій, при яких виникає шлюб. У медицині – класифікація симптомів. У соціології завдання кластеризації – розбиття респондентів на однорідні групи.

# ЛЕКЦІЯ 4
# ЗАДАЧІ DATA MINING. ПРОГНОЗУВАННЯ Й ВІЗУАЛІЗАЦІЯ. МЕТОДИ ВІЗУАЛІЗАЦІЇ

## 4.1 Задачі прогнозування

Задачі прогнозування вирішуються в найрізноманітніших областях людської діяльності, таких як наука, економіка, виробництво й безліч інших сфер. Прогнозування є важливим елементом організації керування як окремими господарюючими суб'єктами, так і економіки в цілому.

Розвиток методів прогнозування безпосередньо пов'язаний з розвитком інформаційних технологій, зокрема, з ростом обсягів збережених даних і ускладненням методів і алгоритмів прогнозування, реалізованих в інструментах Data Mining.

Завдання прогнозування, мабуть, може вважатися однією з найбільш складних задач Data Mining, воно вимагає ретельного дослідження вихідного набору даних і методів, що підходять для аналізу.

Прогнозування ( від грецького Prognosis), у широкому розумінні цього слова, визначається як випереджаюче відображення майбутнього.

Метою прогнозування є передбачення майбутніх подій.

Прогнозування (forecasting) є однією з задач Data Mining і одночасно одним із ключових моментів при прийнятті рішень.

Прогностика (prognostics) – теорія й практика прогнозування.

Прогнозування спрямоване на визначення тенденцій динаміки конкретного об'єкта або події на основі ретроспективних даних, тобто аналізу його стану колись і тепер. Таким чином, розв'язок задачі прогнозування вимагає деякої навчальної вибірки даних.

Прогнозування – установлення функціональної залежності між залежними й незалежними змінними.

Прогнозування є розповсюдженим і затребуваним завданням у багатьох областях людської діяльності. У результаті прогнозування зменшується ризик прийняття невірних, необґрунтованих або суб'єктивних рішень.

Приклади його задач: прогноз руху грошових коштів, прогнозування врожайності агрокультури, прогнозування фінансової стабільності підприємства.

Крім економічної й фінансової сфери, задачі прогнозування постають в найрізноманітніших областях: медицині, фармакології; популярним зараз стає політичне прогнозування.

Загалом розв'язок задачі прогнозування зводиться до розв'язку таких підзадач:

вибір моделі прогнозування;

аналіз адекватності й точності побудованого прогнозу.

Порівняння задач прогнозування і класифікації.

У попередній темі нами було розглянуто задачу класифікації. Прогнозування подібне із задачею класифікації.

Багато методів Data Mining використовуються для розв’язку задач класифікації і прогнозування. Це, наприклад, лінійна регресія, нейронні мережі, дерева рішень (які іноді так і називають – дерева прогнозування й класифікації) [4].

Завдання класифікації й прогнозування мають подібності й відмінності.

Так у чому ж подібність завдань прогнозування й класифікації? При розв’язку обох завдань використовується двоетапний процес побудови моделі на основі навчального набору і її використання для пророкування невідомих значень залежної змінної.

Відмінність задач класифікації й прогнозування полягає в тому, що в першій задачі передбачається клас залежної змінної, а в другій – числові значення залежної змінної, пропущені або невідомі (які відносяться до майбутнього).

Повертаючись до прикладу про туристичне агентство, розглянутого у попередній лекції, ми можемо сказати, що визначення класу клієнта є розв’язком задачі класифікації, а прогнозування доходу, який принесе цей клієнт наступного року, буде розв’язком задачі прогнозування.

### 4.2 Прогнозування і часові ряди

Прогнозування і часові ряди. Основою для прогнозування служить історична інформація, що зберігається в базі даних у вигляді часових рядів.

Існує поняття Data Mining часових рядів (Time-Series Data Mining).

На основі ретроспективної інформації у вигляді часових рядів можливий розв’язок різних задач Data Mining.

На рисунку 4.1 представлені результати опитування відносно Data Mining часових рядів. Як бачимо, найбільший відсоток (23%) серед розв’язуваних задач займає прогнозування. Далі йдуть класифікація і кластеризація (по 14%), сегментація й виявлення аномалій (по 9%), виявлення правил (8%). На інші задачі доводиться менш, ніж по 6%.

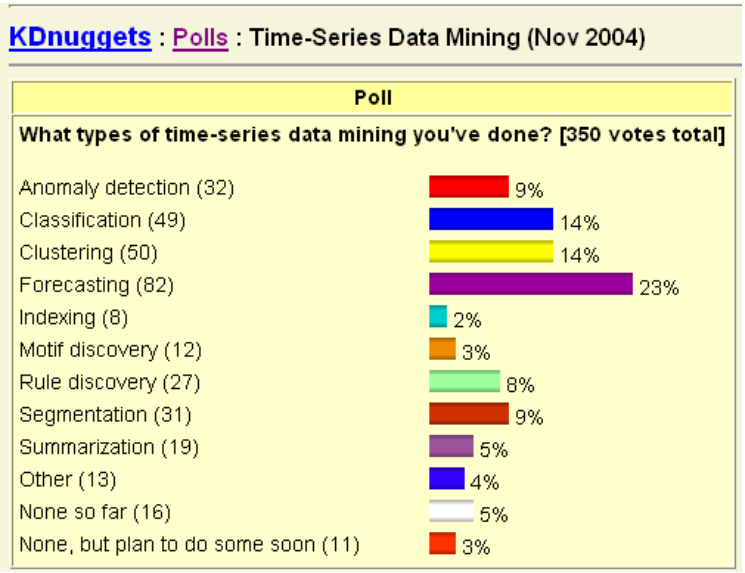



Рисунок 4.1 – Data Mining часових рядів

Однак щоб зосередитися на понятті прогнозування, ми будемо розглядати часові ряди лише в рамках розв'язку задачі прогнозування.

Приведемо дві принципові відмінності часового ряду від простої послідовності спостережень:

Члени часового ряду, на відміну від елементів випадкової вибірки, не є статистично незалежними.

Члени часового ряду не є однаково розподіленими.

Часовий ряд – послідовність спостережуваних значень якої-небудь ознаки, упорядкованих у невипадкові моменти часу.

Відмінністю аналізу часових рядів від аналізу випадкових вибірок є припущення про рівні проміжки часу між спостереженнями і їх хронологічний порядок. Прив'язка спостережень до часу відіграє тут ключову роль, тоді як при аналізі випадкової вибірки вона не має ніякого значення.

Типовий приклад часового ряду – дані біржових торгів.

Інформація, накопичена в різноманітних базах даних підприємства, є часовими рядами, якщо вона розташована в хронологічному порядку й зроблена в послідовні моменти часу.

Аналіз часового ряду здійснюється з метою:

- визначення природи ряду;
- прогнозування майбутніх значень ряду.

У процесі визначення структури й закономірностей часового ряду передбачається виявлення: шумів і викидів, тренду, сезонного компонента, циклічного компонента. Визначення природи часового ряду може бути використане як своєрідна «розвідка» даних. Знання аналітика про наявність сезонного компонента необхідне, наприклад, для визначення

кількості записів вибірки, яка повинна брати участь у побудові прогнозу.

Шуми й викиди будуть докладно обговорюватися в наступних темах курсу. Вони ускладнюють аналіз часового ряду. Існують різні методи визначення й фільтрації викидів, що дають можливість виключити їх з метою більш якісного Data Mining.

### 4.3 Тренд, сезонність і цикл

Основними складовими часового ряду є тренд і сезонний компонент. Складові цих рядів можуть являти собою або тренд, або сезонний компонент.

Тренд є систематичним компонентом часового ряду, який може змінюватися в часі.

Трендом називають невипадкову функцію, яка формується під дією загальних або довгочасних тенденцій, що впливають на часовий ряд.

Прикладом тенденції може виступати, наприклад, фактор росту досліджуваного ринку.

Автоматичного способу виявлення трендів у часових рядах не існує. Але якщо часовий ряд включає монотонний тренд (тобто відзначене його стійке зростання або стійке спадання), аналізувати часовий ряд у більшості випадків неважко.

Існує велика різноманітність постановок задач прогнозування, які можна підрозділити на дві групи: прогнозування односерійних рядів і прогнозування мультисерійних, або взаємовплваючих, рядів.

Група прогнозування односерійних рядів включає задачу побудови прогнозу однієї змінної за ретроспективним даними тільки цієї змінної, без врахування впливу інших змінних і факторів.

Група прогнозування мультисерійних, або взаємовплваючих, рядів включає задачу аналізу, де необхідно враховувати взаємовпливаючі фактори на одну або декілька змінних.

Крім розподілу на класи по односерійності й багатосерійності, ряди також бувають сезонними й несезонними.

Останній розподіл має на увазі наявність або відсутність у часового ряду такої складової як сезонність, тобто включення сезонного компонента.

Сезонна складова часового ряду є періодично повторюваним компонентом часового ряду.

Властивість сезонності означає, що через приблизно рівні проміжки часу форма кривої, яка описує поведінку залежної змінної, повторює свої характерні обриси.

Властивість сезонності важлива при визначенні кількості ретроспективних даних, які будуть використовуватися для прогнозування [4].

Розглянемо простий приклад. На рисунку 4.2. наведений фрагмент ряду, який ілюструє поведінку змінної «обсяги продажу товару X» за період, що становить один місяць. При

вивченні кривої, наведеної на малюнку, аналітик не може зробити припущень щодо повторюваності форми кривої через рівні проміжки часу.

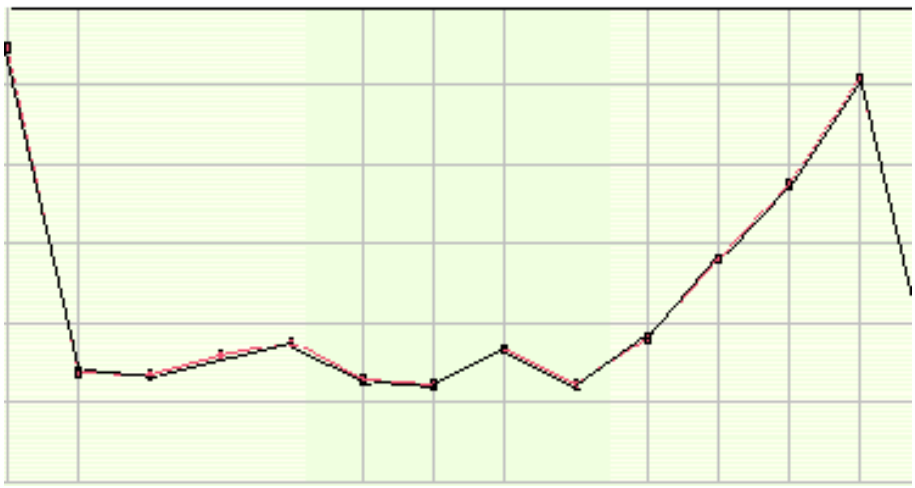


Рисунок 4.2 – Фрагмент часового ряду за сезонний період

Однак при розгляді більш тривалого ряду (за 12 місяців), зображеного на рисунку 4.3, можна побачити явну наявність сезонного компонента. Отже, про сезонність продажів можна говорити тільки, коли розглядаються дані за кілька місяців.

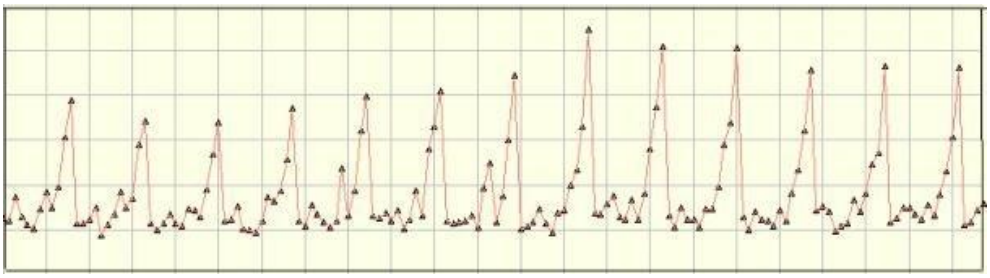


Рисунок 4.3 – Фрагмент часового ряду з 12-ти сезонних періодів

Таким чином, у процесі підготовки даних для прогнозування аналітикові слід визначити, чи володіє ряд, який він аналізує, властивістю сезонності.

Визначення наявності компоненти сезонності необхідно для того, щоб вхідна інформація мала властивість репрезентативності.

Ряд можна вважати несезонним, якщо при розгляді його зовнішнього вигляду не можна зробити припущень про повторюваність форми кривої через рівні проміжки часу.

Іноді по зовнішньому вигляду кривої ряду не можна визначити, є він сезонним чи ні.

Існує поняття сезонного мультиряду. У ньому кожний ряд описує поведінку факторів, які впливають на залежну (цільову) змінну.

Приклад такого ряду – ряди продажів декількох товарів, підданих сезонним коливанням.

При зборі даних і виборі факторів для розв’язку задачі по прогнозуванню в таких випадках слід ураховувати, що вплив обсягів продажів товарів один на одного тут набагато менше, ніж вплив фактору сезонності.

Важливо не плутати поняття сезонного компонента ряду й сезонів природи. Незважаючи на близькість їх звучання, ці поняття відрізняються. Так, наприклад, обсяги продажів морозива влітку набагато більше, ніж в інші сезони, однак це є тенденцією попиту на даний товар.

Дуже часто тренд і сезонність присутні в часовому ряді одночасно.

Приклад. Прибуток фірми росте протягом декількох років (тобто в часовому ряді присутній тренд); ряд також містить сезонний компонент.

Відмінності циклічного компонента від сезонного:

Тривалість циклу, як правило, більше, ніж один сезонний період;

Цикли, на відміну від сезонних періодів, не мають певної тривалості.

При виконанні яких-небудь перетворень зрозуміти природу часового ряду значно простіше, такими перетвореннями можуть бути, наприклад, видалення тренда й згладжування ряду.

При виборі змінних слід ураховувати доступність ретроспективних даних, переваги осіб, що ухвалюють рішення, остаточну вартість Data Mining.

Часто при розв’язку задач прогнозування виникає необхідність пророкування не самої змінної, а зміни її значень.

Друге питання при розв’язку задачі прогнозування – визначення наступних параметрів:

- періоду прогнозування;
- горизонту прогнозування;
- інтервалу прогнозування.

Період прогнозування – основна одиниця часу, на яку робиться прогноз.

Наприклад, ми прагнемо довідатися дохід компанії через місяць. Період прогнозування для цієї задачі – місяць.

Горизонт прогнозування – це число періодів у майбутньому, які покриває прогноз.

Якщо ми прагнемо дізнатися прогноз на 12 місяців уперед, з даними по кожному місяцю, то період прогнозування в цьому завданні – місяць, горизонт прогнозування – 12 місяців.

Інтервал прогнозування – частота, з якою робиться новий прогноз. Інтервал прогнолзування може збігатися з періодом прогнозування.

Рекомендації з вибору параметрів прогнозування. При виборі параметрів необхідно враховувати, що горизонт прогнозування повинен бути не менше, ніж час, який необхідний для реалізації розв’язку, прийнятого на основі цього прогнозу. Тільки в цьому випадку

прогнозування буде мати сенс.

Зі збільшенням горизонту прогнозування точність прогнозу, як правило, знижується, а зі зменшенням горизонту – підвищується.

Ми можемо поліпшити якість прогнозування, зменшуючи час, необхідний на реалізацію розв'язку, для якого реалізується прогноз, і, отже, зменшивши при цьому горизонт і помилку прогнозування.

При виборі інтервалу прогнозування слід вибирати між двома ризиками: вчасно не визначити зміни в аналізованому процесі й високою вартістю прогнозу. При тривалому інтервалі прогнозування виникає ризик не ідентифікувати зміни, що відбувся в процесі, при короткому – зростають витрати на прогнозування [5].

При виборі інтервалу необхідно також ураховувати стабільність аналізованого процесу й вартість проведення прогнозу.

Точність прогнозу, необхідна для розв'язку конкретної задачі, дуже впливає на прогнозуючу систему. Помилка прогнозу залежить від використовуваної системи прогнозу.

Чим більше ресурсів має така система, тим більше шансів одержати більш точний прогноз. Однак прогнозування не може повністю усунути ризики при прийнятті розв'язків. Тому завжди враховується можлива помилка прогнозування.

### 4.4 Види помилок та прогнозів

Точність прогнозу характеризується помилкою прогнозу.

Найпоширеніші види помилок:

Середня помилка (СП). Вона обчислюється простим усередненням помилок на кожному кроці. Недолік цього виду помилки – позитивні й негативні помилки анулюють одна одну.

Середня абсолютна помилка (САП). Вона розраховується як середнє абсолютних помилок. Якщо вона дорівнює нулю, то ми маємо досконалий прогноз. У порівнянні із середньою квадратичною помилкою, цей захід «не надає занадто великого значення» викидам.

Сума квадратів помилок (SSE), середньоквадратична помилка. Вона обчислюється як сума (або середнє) квадратів помилок. Це найбільше часто використовувана оцінка точності прогнозу.

Відносна помилка (ВП). Попередні міри використовували дійсні значення помилок. Відносна помилка виражає якість припасування в термінах відносних помилок.

Види прогнозів. Прогноз може бути короткостроковим, середньостроковим і довгостроковим.

Короткостроковий прогноз становить прогноз на кілька кроків уперед, тобто

здійснюється побудова прогнозу не більше ніж на 3% від обсягу спостережень або на 1-3 кроку вперед.

Середньостроковий прогноз – це прогноз на 3-5% від обсягу спостережень, але не більш 7-12 кроків уперед; також під цим типом прогнозу розуміють прогноз на один або половину сезонного циклу. Для побудови короткострокових і середньострокових прогнозів цілком підходять статистичні методи.

Довгостроковий прогноз – це прогноз більш ніж на 5% від обсягу спостережень.

При побудові даного типу прогнозів статистичні методи практично не використовуються, крім випадків дуже «гарних» рядів, для яких прогноз можна просто «намалювати».

Дотепер ми розглядали аспекти прогнозування, так чи інакше пов'язані із процесом ухвалення рішення. Існують і інші фактори, які необхідно враховувати при прогнозуванні.

Задача 1. Відомо, що аналізований процес відносно стабільний у часі, зміни відбуваються повільно, процес не залежить від зовнішніх факторів.

Задача 2. Аналізований процес нестабільний і дуже сильно залежить від зовнішніх факторів.

Розв'язок першої задачі повинен бути зосереджений на використанні великої кількості ретроспективних даних. При розв'язку другої задачі особливу увагу слід звернути на оцінки фахівця в предметній області, експерта, щоб мати можливість відбити в прогнозуючій моделі всі необхідні зовнішні фактори, а також приділити час для збору даних по цих факторах (збір зовнішніх даних часто набагато складніший збору внутрішніх даних інформаційної системи). Доступність даних, на основі яких буде здійснюватися прогнозування, – важливий фактор побудови прогнозної моделі. Для можливості виконання якісного прогнозу дані повинні бути представницькими, точними й достовірними [5].

Методи прогнозування. Серед розповсюджених методів Data Mining, використовуваних для прогнозування, відзначимо нейронні мережі й лінійну регресію.

Вибір методу прогнозування залежить від багатьох факторів, у тому числі від параметрів прогнозування. Вибір методу слід провадити з обліком усіх специфічних особливостей набору ретроспективних даних і цілей, з якими він будується.

Програмне забезпечення Data Mining, використовуване для прогнозування, повинне забезпечувати користувача точним і достовірним прогнозом. Однак одержання такого прогнозу залежить не тільки від програмного забезпечення й методів, закладених у його основу, але також і від інших факторів, серед яких повнота й вірогідність вихідних даних, своєчасність і оперативність їх поповнення, кваліфікація користувача.

Завдання візуалізації. Візуалізація – це інструментарій, який дозволяє побачити кінцевий результат обчислень, організувати керування обчислювальним процесом і навіть

повернутися назад до вихідних даних, щоб визначити найбільш раціональний напрямок подальшого руху.

У результаті використання візуалізації створюється графічний образ даних. Застосування візуалізації допомагає в процесі аналізу даних побачити аномалії, структури, тренди. При розгляді завдання прогнозування ми використовували графічне представлення часового ряду й побачили, що в ньому є присутнім сезонний компонент. У попередній лекції ми розглядали завдання класифікації й кластеризації, і для ілюстрації розподілу об'єктів у двомірному просторі також використовували візуалізацію.

Можна говорити про те, що застосування візуалізації є більш економічним: лінія тренду або скупчення точок на діаграмі розсіювання дозволяє аналітикові набагато швидше визначити закономірності й прийти до потрібного розв'язку. Таким чином, тут мова йде про використання в Data Mining не символів, а образів.

Головна перевага візуалізації – практично повна відсутність необхідності в спеціальній підготовці користувача. За допомогою візуалізації ознайомитися з інформацією дуже легко, досить усього лише на неї подивитися.

Хоча найпростіші види візуалізації з'явилися досить давно, її використання зараз тільки набирає популярність. Візуалізація не спрямована винятково на вдосконалювання техніки аналізу – за словами Скотта Лейбса, у деяких випадках візуалізація може навіть замінити її.

Візуалізація даних може бути представлена у вигляді: графіків, схем, гістограм, діаграм і т.д.

Коротко роль візуалізації можна описати такими її можливостями:

- підтримка інтерактивного й погодженого дослідження;
- допомога у показі результатів;
- використання очей (зору), щоб створювати зорові образи й осмислювати їх.

Погана візуалізація Результати візуалізації іноді можуть вводити користувача в оману. Приведемо простий приклад поганої візуалізації.

Допустимо, ми маємо базу «Прибуток компанії А» за період з 2000 по 2005 рік. Дані представлені у табличному вигляді (табл. 4.1).

Таблиця 4.1 Прибуток компанії А

| Рік | Прибуток |
|---|---|
| 2000 | 1100 |
| 2001 | 1101 |
| 2002 | 1104 |
| 2003 | 1105 |
| 2004 | 1106 |
| 2005 | 1107 |

Побудуємо гістограму в Excel за цими даними. Гістограма становить візуальне

зображення розподілу даних.

Ця інформація відображається за допомогою серії прямокутників або смуг однакової ширини, висота яких указує кількість даних у кожному класі.

Використовуючи всі значення побудови графіка, прийняті за замовчуванням, одержуємо гістограмму, наведену на рисунку 4.4.

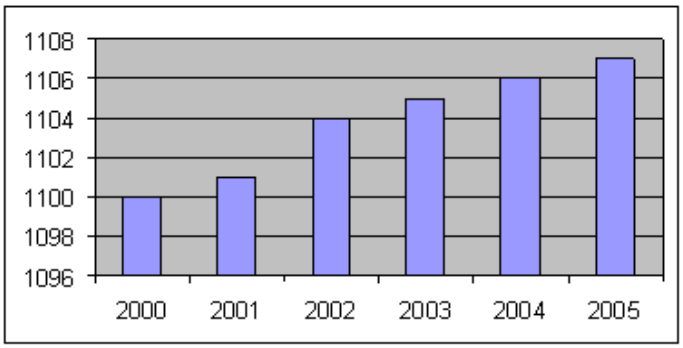



Рисунок 4.4 – Гістограмма, мінімальне значення осі у рівне 1096

Даний малюнок демонструє значне зростання прибутку компанії А за період з 2000 по 2005 року. Однак, якщо ми звернули увагу на вісь у, що показує величину прибутку, те побачимо, що ця вісь перетинає вісь х у значенні, рівному 1096. Фактично, вісь у зі значеннями від 1096 до 1108 вводить користувача в оману. Змінивши значення параметрів, відповідальних за формат осі у, одержуємо графік, наведений на рисунку 4.5.

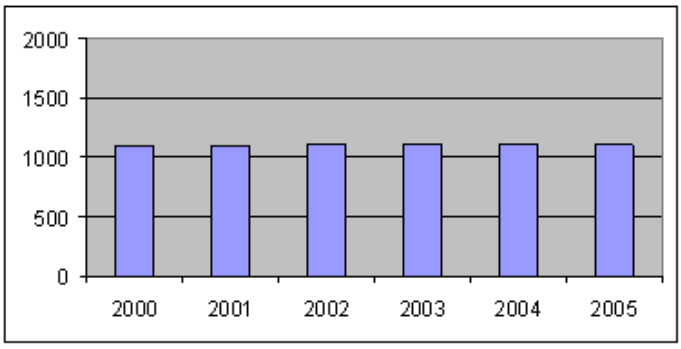



Рисунок 4.5 – Гістограма, мінімальне значення осі у рівне 0

Вісь у зі значеннями від 0 до 2000 дає користувачеві правильну інформацію про незначну зміну прибутків компанії.

Якщо мова йде про велику розмірність і складності вихідних даних, кошти візуалізації забезпечують їхнє різке зменшення, конденсуючи, можливо, мільйони записів даних у прості, легкі для розуміння й маніпулювання показники. Такі показники називають візуальним або графічним способом показу інформації. Візуалізацію можна вважати ключовим фактором у

дослідженні даних, отриманих за допомогою інструментів Data Mining. У таких випадках говорять про візуальний Data Mining.

### 4.5 Візуалізація інструментів Data Mining

Зі зростанням кількості даних, що накопичуються, навіть при використанні як завгодно потужних і різносторонніх алгоритмів Data Mining, стає усе складніше «переварювати» і інтерпретувати отримані результати. А, як відомо, одне з положень Data Mining – пошук практично корисних закономірностей. Закономірність може стати практично корисною, тільки якщо її можна осмислити й зрозуміти.

В 1987 році з ініціативи ACM SIGGRAPH IEEE Computer Society Technical Committee of Computer Graphics, у зв'язку з необхідністю використання нових методів, засобів і технологій даних, були сформульовані відповідні завдання напрямку візуалізації.

До способів візуального або графічного представлення даних відносять графіки, діаграми, таблиці, звіти, списки, структурні схеми, карти і т.д.

Візуалізація традиційно розглядалася як допоміжний засіб при аналізі даних, однак зараз усе більше досліджень говорить про її самостійну роль.

Традиційні методи візуалізації можуть знаходити наступне застосування:

- представляти користувачеві інформацію в наочному вигляді;
- компактно описувати закономірності, властиві вихідному набору даних;
- знижувати розмірність або стискати інформацію;
- відновлювати пробіли в наборі даних;
- знаходити шуми й викиди в наборі даних.

Візуалізація інструментів Data Mining. Кожний з алгоритмів Data Mining використовує певний підхід до візуалізації. У попередніх лекціях ми розглянули ряд методів Data Mining. У ході використання кожного з методів, а точніше, його програмної реалізації, ми одержували якісь візуалізатори, за допомогою яких нам вдавалося інтерпретувати результати, отримані в результаті роботи відповідних методів і алгоритмів [6].

Для дерев рішень це візуалізатор дерева рішень, список правил, таблиця спряженості.

Для нейронних мереж залежно від інструмента це може бути топологія мережі, графік зміни величини помилки, що демонструє процес навчання.

Для карт Кохонена: карти входів, виходів, інші специфічні карти.

Для лінійної регресії в якості візуалізатора виступає лінія регресії.

Для кластеризації: дендрограми, діаграми розсіювання.

Діаграми й графіки розсіювання часто використовуються для оцінки якості роботи того або іншого методу.

Усі ці способи візуального представлення або відображення даних можуть виконувати одну з функцій:

- є ілюстрацією побудови моделі (наприклад, представлення структури (графа) нейронної мережі);
- допомагають інтерпретувати отриманий результат;
- є засобом оцінки якості побудованої моделі;
- поєднують перераховані вище функції (дерево розв'язків, дендрограма).

Існує багато різних способів представлення моделей, але графічне їх представлення дає користувачеві максимальну «цінність».

Користувач, у більшості випадків, не є фахівцем у моделюванні, найчастіше він експерт у своїй предметній області. Тому модель Data Mining повинна бути представлена на найбільш природній для нього мові або, хоча б, містити мінімальну кількість різних математичних і технічних елементів.

Таким чином, доступність є однією з основних характеристик моделі Data Mining. Незважаючи на це, існує й такий розповсюджений і найбільш простий спосіб показу моделі, як «чорний ящик». У цьому випадку користувач не розуміє поведінки тієї моделі, якою користується. Однак, незважаючи на нерозуміння, він одержує результат – виявлені закономірності. Класичним прикладом такої моделі є модель нейронної мережі.

Інший спосіб представлення моделі – представлення її в інтуїтивному, зрозумілому виді. У цьому випадку користувач дійсно може розуміти те, що відбувається «усередині» моделі. Таким чином, можна забезпечити його особисту участь у процесі.

Такі моделі забезпечують користувачеві можливість обговорювати її логіку з колегами, клієнтами й іншими користувачами, або пояснювати її.

Розуміння моделі веде до розуміння її змісту. У результаті розуміння зростає довіра до моделі. Класичним прикладом є дерево рішень. Побудоване дерево рішень дійсно поліпшує розуміння моделі, тобто використовуваного інструмента Data Mining.

Крім розуміння, такі моделі забезпечують користувача можливістю взаємодіяти з моделлю, задавати їй питання й одержувати відповіді. Прикладом такої взаємодії є засіб «що, якщо». За допомогою діалогу «система-користувач» користувач може одержати розуміння моделі.

Тепер перейдемо до функцій, які допомагають інтерпретувати й оцінити результати побудови Data Mining моделей. Це всілякі графіки, діаграми, таблиці, списки і т.д.

Прикладами засобів візуалізації, за допомогою яких можна оцінити якість моделі, є діаграма розсіювання, таблиця спряженості, графік зміни величини помилки.

Діаграма розсіювання становить графік відхилення значень, прогнозованих за допомогою моделі, від реальних. Ці діаграми використовують для безперервних величин.

Візуальна оцінка якості побудованої моделі можлива тільки по закінченню процесу побудови моделі.

Таблиця спряженості використовується для оцінки результатів класифікації. Такі таблиці застосовуються для різних методів класифікації. Оцінка якості побудованої моделі можлива тільки по закінченню процесу побудови моделі.

Графік зміни величини помилки. Графік демонструє зміну величини помилки в процесі роботи моделі. Наприклад, у процесі роботи нейронних мереж користувач може спостерігати за зміною помилки на навчальній й тестовій множинах і зупинити навчання для недопущення «перенавчання» мережі. Тут оцінка якості моделі і його зміни може оцінюватися безпосередньо в процесі побудови моделі.

Прикладами засобів візуалізації, які допомагають інтерпретувати результат, є: лінія тренду в лінійній регресії, карти Кохонена, діаграма розсіювання в кластерному аналізі.

### 4.6 Методи візуалізації

Методи візуалізації, залежно від кількості використовуваних вимірів, прийнято класифікувати на дві групи:

- представлення даних в одному, двох і трьох вимірах;
- представлення даних у чотирьох і більше вимірах.

Представлення даних в одному, двох і трьох вимірах. До цієї групи методів ставляться добре відомі способи відображення інформації, які доступні для сприйняття людською уявою. Практично будь-який сучасний інструмент Data Mining включає способи візуального представлення із цієї групи.

Відповідно до кількості вимірів представлення це можуть бути наступні способи:

- одномірне (univariate) або 1-D;
- двовимірне (bivariate) або 2-D;
- тривимірне, проекційне (projection) або 3-D.

Слід відзначити, що найбільше природно людське око сприймає двомірні представлення інформації.

При використанні двох – і тривимірного представлення інформації користувач має можливість побачити закономірності набору даних:

- його кластерну структуру й розподіл об’єктів на класи (наприклад, на діаграмі розсіювання);
- топологічні особливості;
- наявність трендів;
- інформацію про взаємне розташування даних;

- існування інших залежностей, властивих досліджуваному набору даних.

Якщо набір даних має більше трьох вимірів, то можливі такі варіанти:

використання багатомірних методів представлення інформації (вони розглянуті нижче);

зниження розмірності до одно-, двох – або тривимірного представлення. Існують різні способи зниження розмірності, один з них – факторний аналіз – був розглянутий в одній з попередніх лекцій. Для зниження розмірністі й одночасного візуального представлення інформації на двовимірних картах використовуються карти, що самоорганізуються.

Представлення даних в чотирьох і більше вимірах. Представлення інформації в чотиривимірному й більш вимірах недоступні для людського сприйняття. Однак розроблені спеціальні методи для можливості відображення й сприйняття людиною такої інформації [7].

Найбільш відомі способи багатомірного представлення інформації:

- паралельні координати;
- «особи Чернова»;
- пелюсткові діаграми.

Паралельні координати. У паралельних координатах змінні кодуються по горизонталі, вертикальна лінія визначає значення змінної. Приклад набору даних, представленого в декартових координатах і паралельних координатах, даний на рисунку 4.6. Цей метод представлення багатомірних даних був винайдений Альфредом Інселбергом (Alfred Inselberg ) в 1985 році.

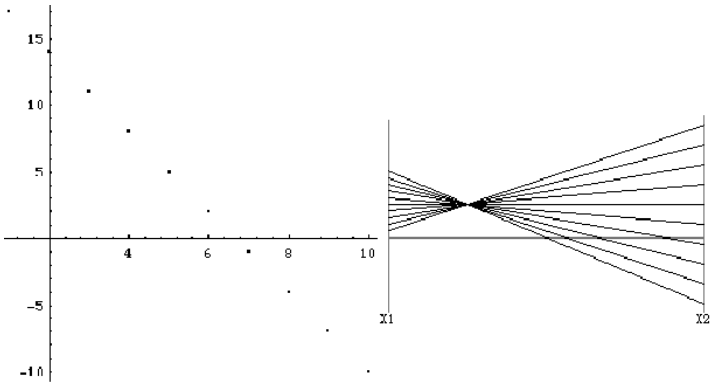



Рисунок 4.6 – Набір даних у декартових координатах і в паралельних координатах

«Особи Чернова». Основна ідея представлення інформації в «особах Чернова» полягає в кодуванні значень різних змінних у характеристиках або рисах людської особи. Приклад такого «особи» наведений на рисунку 4.7.

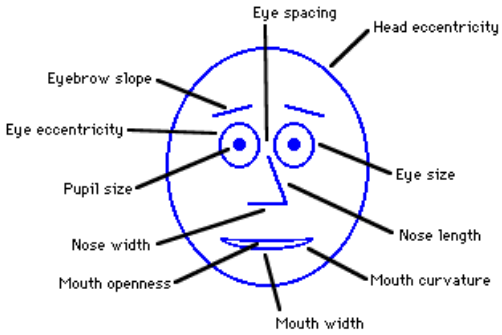



Рисунок 4.7 – «Особа Чернова»

Для кожного спостереження рисується окрема «особа». На кожній «особі» відносні значення змінних представлені як форми й розміри окремих рис особи (наприклад, довжина й ширина носа, розмір очей, розмір зіниці, кут між бровами).

Аналіз інформації за допомогою такого способу відображення заснований на здатності людини інтуїтивно знаходити подібності й відмінності в рисах особи.

На рисунку 4.8 поданий набір даних, кожний запис якого виражений у вигляді «Особи Чернова».

Рисунок 4.8 – Приклад багатомірного зображення даних за допомогою «осіб Чернова»

Перед використанням методів візуалізації необхідно:

- проаналізувати, чи варто зображувати всі дані або ж лише якусь їхню частину;
- вибрати розміри, пропорції й масштаб зображення;
- вибрати метод, який може найбільше яскраво відобразити закономірності, властиві набору даних.

Багато сучасних засобів аналізу даних дозволяють будувати сотні типів різних графіків і діаграм. Тому вибір методу візуалізації, якщо він самостійно здійснюється користувачем, не такий простий і легкий, як може здатися на перший погляд. Наявність великої кількості засобів візуалізації, представлених в інструменті, який застосовує користувач, може навіть викликати розгубленість.

Ту саму інформацію можна представити за допомогою різних засобів. Для того щоб засіб візуалізації міг виконувати своє основне призначення – представляти інформацію в простому й доступному для людського сприйняття вигляді – необхідно дотримуватися законів відповідності обраного розв'язку змісту відображуваної інформації і її функціональному призначенню. Іншими словами, потрібно зробити так, щоб при погляді на візуальне представлення інформації можна було відразу виявити закономірності у вихідних даних і приймати на їхній основі рішення.

Серед двомірних і тривимірних засобів найбільше широко відомі лінійні графіки, лінійні, стовпчикові, кругові секторні й векторні діаграми.

Приведемо рекомендації з використання цих найбільш простих і популярних засобів візуалізації.

За допомогою лінійного графіка можна відобразити тенденцію, передати зміни якої-небудь ознаки в часі. Для порівняння декількох рядів чисел такі графіки наносяться на ті самі осі координат.

Гістограму застосовують для порівняння значень протягом деякого періоду або ж співвідношення величин.

Кругові діаграми використовують, якщо необхідно відобразити співвідношення частин і цілого, тобто для аналізу складу або структури явищ. Складові частини цілого зображуються секторами круга. Сектори рекомендують розміщати по їхній величині: угорі – найбільший, інші – по рухові годинної стрілки в порядку зменшення їх величини. Кругові діаграми також застосовують для відображення результатів факторного аналізу, якщо дії всіх факторів є односпрямованими. При цьому кожний фактор відображається у вигляді одного із секторів кола.

Вибір того або іншого засобу візуалізації залежить від поставленого завдання (наприклад, потрібно визначити структуру даних або ж динаміку процесу) і від характеру набору даних.

Якість візуалізації. Сучасні аналітичні засоби, у тому числі й Data Mining, немислимі без якісної візуалізації. У результаті використання засобів візуалізації повинні бути отримані наочні й виразні, ясні й прості зображення, за рахунок використання різноманітних засобів: кольору, контрасту, границь, пропорцій, масштабу і т.д.

У зв'язку з ростом вимог до засобів візуалізації, а також необхідності порівняння їх між собою, в останні роки був сформований ряд принципів якісної візуальної представлення інформації.

Принципи Тафта (Tufte's Principles) графічне представлення даних високої якості говорить:

надавайте користувачеві найбільшу кількість ідей, у найкоротший час, з найменшою

кількістю чорнила на найменшому просторі;

говоріть правду про дані.

### 4.7 Принципи компонування візуальних засобів

Основні принципи компонування візуальних засобів представлення інформації:

- принцип лаконічності;
- принцип узагальнення й уніфікації;
- принцип акценту на основних значеннєвих елементах;
- принцип автономності;
- принцип структурності;
- принцип стадійності;
- принцип використання звичних асоціацій і стереотипів.

Принцип лаконічності говорить про те, що засіб візуалізації повинен містити лише ті елементи, які необхідні для повідомлення користувачеві істотної інформації, точного розуміння її значення або прийняття (з імовірністю не нижче допустимої величини) відповідного оптимального розв’язку.

Крім позначених вище принципів, засіб візуалізації повинний мати високу надійність і швидкістю, яка влаштує користувача, що приймає на основі цієї інформації рішення.

Представлення просторових характеристик. Окремим напрямком візуалізації є наочне представлення просторових характеристик об’єктів. У більшості випадків такі засоби виділяють на карті окремі регіони й позначають їхніми різними кольорами залежно від значення аналізованого показника.

На рисунку 4.9 наведений приклад такої візуалізації в середовищі Mineset, що є, у цьому випадку, інструментом візуального Data Mining. Карта представлена у вигляді графічного інтерфейсу, що відображає дані у вигляді тривимірного ландшафту довільно визначених і позиціонованих форм (стовпчастих діаграм, кожна з індивідуальними висотою й кольором). Такий спосіб дозволяє наочно показувати кількісні й реляційні характеристики просторово-орієнтованих даних і швидко ідентифікувати в них тренди [7].

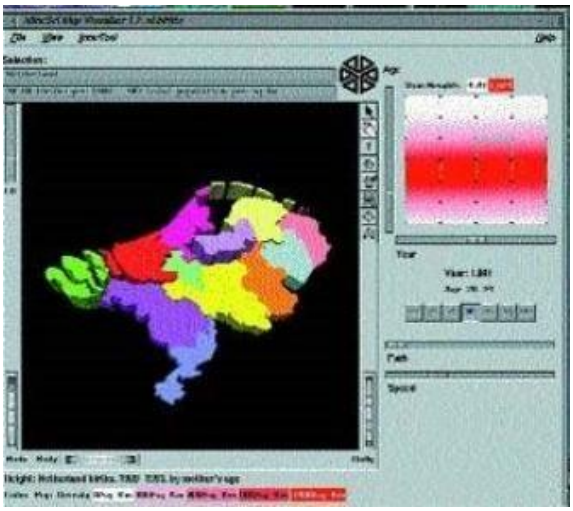


Рисунок 4.9 – Mineset. Ландшафтний візуалізатор

### 4.8 Основні тенденції в області візуалізації

Як ми вже відзначали, за допомогою засобів візуалізації підтримуються важливі завдання бізнесу, серед яких – процес прийняття рішень. У зв'язку із цим виникає необхідність переходу засобів візуалізації на більш якісний рівень, який характеризується появою абсолютно нових засобів візуалізації й поглядів на їх функції, а також розвитком ряду тенденцій у цій області.

Серед основних тенденцій в області візуалізації Філіп Рассом (Philip Russom) виділяє:

- розробка складних видів діаграм;
- підвищення рівня взаємодії з візуалізацією користувача;
- збільшення розмірів і складності структур даних, що представляються візуалізацією.

Розробка складних видів діаграм. Більшість візуалізацій даних побудовані на основі діаграм стандартного типу (секторні діаграми, графіки розсіювання і т.д.). Ці способи є одночасно найстаршими, найбільш елементарними й розповсюдженими. В останні роки перелік видів діаграм, підтримуваних інструментальними засобами візуалізації, суттєво розширився. Оскільки потреби користувачів досить різноманітні, інструменти візуалізації підтримують всілякі типи діаграм. Наприклад, відомо, що бізнес-користувачі віддають перевагу секторним діаграмам і гістограмам, тоді як вчених більше влаштовують візуалізації у вигляді графіків розсіювання й діаграм констеляції. Користувачі, що працюють із геопросторовими даними, сильніше зацікавлені в картах і інших тривимірних представленнях даних.

Електронні інструментальні панелі, у свою чергу, більш популярні серед керівників, що використовують бізнес-аналітичні технології для контролю над показниками роботи компанії. Такі користувачі потребують наочної візуалізації у вигляді «спідометрів», «термометрів» і «світлофорів».

Засоби створення діаграм і презентаційної графіки призначені головним чином для візуалізації даних. Однак можливості такої візуалізації звичайно вбудовані й у безліч різних інших програм і систем – в інструменти звітування й OLAP, кошту для Text Mining і Data Mining, а також в Crm-Додатки й додатка для керування бізнесом. Для створення вбудованої візуалізації багато постачальників реалізують візуалізаційну функціональність у вигляді компонентів, що вбудовуються в різні інструменти, додатки, програми й web-сторінки ( у тому числі інструментальні панелі й персоналізовані сторінки порталів).

Підвищення рівня взаємодії з візуалізацією користувача. Ще зовсім недавно більша частина коштів візуалізації являла собою статичні діаграми, призначені винятково для перегляду. Зараз широко використовуються динамічні діаграми, уже самі по собі, що є користувацьким інтерфейсом, у якому користувач може прямо й інтерактивно маніпулювати візуалізацією, підбираючи нову виставу інформації.

Наприклад, базова взаємодія дозволяє користувачеві обертати діаграму або змінювати її тип у пошуках найбільш повної представлення даних. Крім того, користувач може міняти візуальні властивості – приміром, шрифти, кольори й рамки. У візуалізаціях складного типу (графіках розсіювання або діаграмах констеляції) користувач може вибирати інформаційні крапки за допомогою миші й переміщати їх, полегшуючи тим самим розуміння представлення даних [7].

Більш досконалі методи візуалізації даних часто містять у собі діаграму або будь-яку іншу візуалізацію як складений рівень. Користувач може глибшатися (drill down) у візуалізацію, досліджуючи подробиці узагальнених нею даних, або глибшатися в OLAP, Data Mining або інші складні технології.

Складна взаємодія дозволяє користувачеві змінювати візуалізацію для знаходження альтернативних інтерпретацій даних. Взаємодія з візуалізацією має на увазі мінімальний по своїй складності користувацький інтерфейс, у якому користувач може управляти виставою даних, просто натискаючи на елементи візуалізації, перетаскуючи й поміщаючи представлення об’єктів даних або вибираючи пункти меню. Інструменти OLAP або Data Mining перетворюють безпосередня взаємодія з візуалізацією в один з етапів ітераційного аналізу даних. Кошту Text Mining або керування документами надають такій безпосередній взаємодії характер навігаційного механізму, що допомагає користувачеві досліджувати бібліотеки документів.

Візуальний запит є найбільш сучасною формою складної взаємодії користувача з даними. У ньому користувач може, наприклад, бачити крайні інформаційні крапки графіка розсіювання, вибирати їхньою мишкою й одержувати нові візуалізації, що представляють саме ці крапки. Додаток візуалізації даних генерує відповідна мова запиту, управляє прийняттям запиту базою даних і візуально представляє результуючу безліч. Користувач може

сфокусуватися на аналізі, не відволікаючись на складання запиту.

Збільшення розмірів і складності структур даних, що представляються візуалізацією. Елементарна секторна діаграма або гістограма візуалізує прості послідовності числових інформаційних крапок. Однак нові вдосконалені типи діаграм здатні візуалізувати тисячі таких крапок і навіть складні структури даних – наприклад, нейронні мережі.

Скажемо, кошту OLAP (а також інструменти генерації запитів і випуску звітів) уже давно підтримують діаграми для своїх онлайнових звітів. Нові візуалізаційні програми обновляють контент за рахунок періодично повторюваного зчитування даних. Фактично користувачі візуалізаційних програм лінійні процеси, що відслідковують (коливання фондового ринку, показники роботи комп'ютерних систем, сейсмограми, сітки корисності й ін.), потребують завантаження даних у режимі реального часу або близькому до нього режимі.

Користувачі інструментів Data Mining звичайно аналізують дуже великі набори чисельних даних. Традиційні типи діаграм для бізнесу (секторні діаграми й гістограми) погано справляються з показом тисяч інформаційних точок. Тому інструменти Data Mining майже завжди підтримують якусь форму візуалізації даних, здатну відображати структури й закономірності досліджуваних наборів даних, відповідно до тих аналітичних підходів, які використовується в інструменті.

Крім того, що візуалізація підтримує обробку структурованих даних, вона також є ключовим засобом представлення схем так званих неструктурованих даних, наприклад текстових документів, тобто

Text Mining. Зокрема, засоби Text Mining можуть здійснювати парсинг більших пакетів документів і формувати предметні покажчики понять і тем, освітлених у цих документах. Коли предметні покажчики створені за допомогою нейронної мережевої технології, користувачеві непросто продемонструвати їх без деякої форми візуалізації даних.

# ЛЕКЦІЯ 5
# МЕТОДИ КЛАСИФІКАЦІЇ Й ПРОГНОЗУВАННЯ. ДЕРЕВА РІШЕНЬ. МЕТОД ОПОРНИХ ВЕКТОРІВ. МЕТОД «НАЙБЛИЖЧОГО СУСІДА». БАЙЄСІВСЬКА КЛАСИФІКАЦІЯ

## 5.1 Метод дерев рішень

Метод дерев рішень (decision trees) є одним з найбільш популярних методів розв'язку задач класифікації й прогнозування. Іноді цей метод Data Mining також називають деревами вирішальних правил, деревами класифікації і регресії.

Як видно з останньої назви, за допомогою даного методу вирішуються задачі класифікації й прогнозування. Якщо залежна, тобто цільова змінна приймає дискретні значення, за допомогою методу дерева рішень вирішується задача класифікації. Якщо ж залежна змінна приймає безперервні значення, то дерево рішень установлює залежність цієї змінної від незалежних змінних, тобто вирішує задачу чисельного прогнозування.

У найбільш простому вигляді дерево рішень – це спосіб показу правил в ієрархічній, послідовній структурі. Основа такої структури – відповіді «Так» або «Ні» на низку питань.

На рисунку 5.1 наведений приклад дерева рішень, задача якого – відповістити на запитання: «Чи грати в гольф?» Щоб розв'язати задачу, тобто прийняти рішення, чи грати в гольф, слід віднести поточну ситуацію до одного з відомих класів (у цьому випадку – «грати» або «не грати»). Для цього потрібно відповісти на низку питань, які є у вузлах цього дерева, починаючи з його кореня. Перший вузол нашого дерева «Сонячно?» є вузлом перевірки, тобто умовою. При позитивній відповіді на запитання здійснюється перехід до лівої частини дерева, називаної лівою гілкою, при негативному – до правої частини дерева. Таким чином, внутрішній вузол дерева є вузлом перевірки певної умови. Далі йде наступне питання і т.д., поки не буде досягнутий кінцевий вузол дерева, що є вузлом розв'язку. Для нашого дерева існує два типи кінцевого вузла: «грати» і «не грати» у гольф.

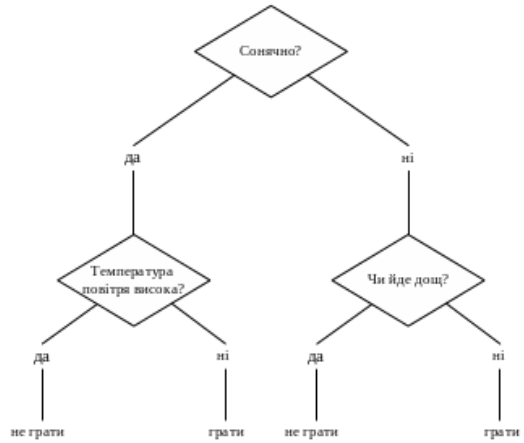



Рисунок 5.1 – Дерево рішень «Чи грати в гольф?»

У розглянутому прикладі вирішується задача бінарної класифікації, тобто створюється дихотомічна класифікаційна модель. У вузлах бінарних дерев розгалуження може відбуватися тільки у двох напрямках, тобто існує можливість тільки двох відповідей на поставлене питання («так» і «ні»). Бінарні дерева є найпростішим, частковим випадком дерев рішень. В інших випадках, відповідей і, відповідно, гілок дерева, що виходять із його внутрішнього вузла, може бути більше двох.

Розглянемо більш складний приклад.

База даних, на основі якої повинне здійснюватися прогнозування, містить наступні ретроспективні дані про клієнтів банку, що є її атрибутами: вік, наявність нерухомості, освіта, середньомісячний дохід, чи повернув клієнт вчасно кредит. Задача полягає в тому, щоб на підставі перерахованих вище даних (крім останнього атрибута) визначити, чи варто видавати кредит новому клієнтові.

На рисунку 5.2. наведений приклад дерева класифікації, за допомогою якого вирішується задача «Чи видавати кредит клієнтові?». Вона є типовою задачею класифікації, і за допомогою дерев рішень одержують досить хороші варіанти її розв’язку.

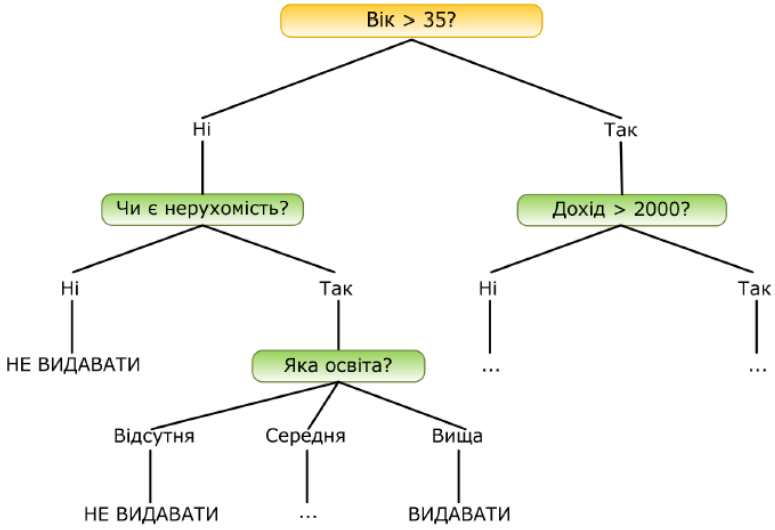



Рисунок 5.2 – Дерево рішень «чи видавати кредит?»

Як ми бачимо, внутрішні вузли дерева (вік, наявність нерухомості, дохід і освіта) є атрибутами описаної вище бази даних.

Ці атрибути називають прогнозуючими, або атрибутами розщеплення (splitting attribute). Кінцеві вузли дерева, або листки, іменуються мітками класу, що є значеннями залежної категоріальної змінної «видавати» або «не видавати» кредит.

Кожна гілка дерева, що йде від внутрішнього вузла, відзначена предикатом розщеплення. Останній може відноситися лише до одного атрибуту розщеплення даного вузла.

Характерна риса предикатів розщеплення: кожний запис використовує унікальний шлях від кореня дерева тільки до одного вузла-розв’язку. Об’єднана інформація про атрибути розщеплення й предикати розщеплення у вузлі називається критерієм розщеплення (splitting criterion).

На рис. 5.2. зображене одне з можливих дерев рішень для розглянутої бази даних. Наприклад, критерій розщеплення «Яка освіта?», міг би мати два предикати розщеплення й виглядати інакше: освіта «вища» і «не вища». Тоді дерево рішень мало б інший вигляд.

Таким чином, для даної задачі (як і для будь-якої іншої) може бути побудовано множина дерев рішень різної якості, з різною прогнозуючою точністю.

Якість побудованого дерева розв’язку досить сильно залежить від правильного вибору критерію розщеплення. Над розробкою й удосконаленням критеріїв працюють багато дослідників. Метод дерев рішень часто називають «наївним» підходом. Але завдяки цілому ряду переваг, даний метод є одним з найбільш популярних для розв’язку задач класифікації.

### 5.2 Переваги дерев рішень

Інтуїтивність і зрозумілість. Дерева рішень є інтуїтивно зрозумілими та легко інтерпретуються користувачем, на відміну від, наприклад, нейронних мереж, які часто вважають «чорними ящиками». Це важливо як для класифікації нових об’єктів, так і для пояснення самої моделі. Дерева рішень дозволяють чітко побачити, чому певний об’єкт віднесено до того чи іншого класу.

Формування правил природною мовою. Дерева рішень дають можливість формулювати правила у зрозумілому для людини вигляді, наприклад: «якщо вік > 35 і дохід > 200, то видати кредит».

Придатність до слабко формалізованих задач. Дерева рішень особливо корисні у тих випадках, коли аналітик не має змоги чітко формалізувати свої знання про предметну область.

Автоматичний вибір значущих змінних. Алгоритми дерев рішень не потребують попереднього відбору змінних — вони самостійно визначають найбільш інформативні атрибути з-поміж усіх доступних.

Конкурентна точність. Моделі, побудовані за допомогою дерев рішень, демонструють точність, яка є порівнянною або вищою за інші методи класифікації, включно зі статистичними методами та нейронними мережами.

Масштабованість. Розроблено алгоритми (наприклад, SLIQ, SPRINT), здатні ефективно працювати з дуже великими базами даних. Їх навчання відбувається за лінійною складністю щодо розміру даних.

Швидкість навчання. Алгоритми дерев рішень значно швидші у навчанні порівняно з

нейронними мережами.

Обробка пропущених значень. Більшість алгоритмів мають вбудовану підтримку обробки відсутніх даних.

Підтримка різних типів змінних. Дерева рішень можуть працювати як з числовими, так і з категоріальними даними, на відміну від багатьох статистичних методів, які потребують числових змінних.

Непараметричність. Дерева рішень не потребують апріорного припущення про розподіл даних чи форму залежності між змінними. Це дозволяє застосовувати їх до задач Data Mining, де відсутня достатня попередня інформація [8].

Процес побудови дерева рішень. Задача класифікації належить до стратегій навчання з учителем, де всі об’єкти у навчальній вибірці вже мають відомі класи.

Побудова дерева включає два основних етапи:

- формування дерева (tree building) — вибір атрибутів і критеріїв розщеплення;
- скорочення дерева (tree pruning) — видалення зайвих гілок з метою запобігання перенавчанню.

Критерії розщеплення. Побудова дерева виконується зверху вниз. На кожному кроці обирається атрибут, який дозволяє найкращим чином розділити множину об’єктів. Основні критерії: міра інформації, індекс Джині (Gini Index).

Проблема надмірної складності дерева. Надто деталізовані дерева, що мають багато гілок, можуть втрачати здатність до узагальнення. Щоб уникнути цього, застосовуються процедури обмеження або скорочення дерева.

Стратегії контролю розміру дерева.

Обмеження при побудові (prepruning) — припинення побудови дерева за заданими умовами (глибина, кількість об’єктів у вузлі тощо). Це знижує час навчання, але може призвести до менш точних моделей.

Скорочення після побудови (postpruning) — видалення гілок, що не покращують точність. Перевага цього підходу — збереження точності класифікації.

Зупинка побудови дерева, основні правила:

- досягнення заданої глибини дерева.
- досягнення мінімальної кількості прикладів у вузлі.
- застосування «ранньої зупинки» для скорочення часу навчання.

Проте ефективнішим вважається підхід, запропонований Бріманом (1984), — відсікання (pruning). Виконується знизу вгору (висхідно) і дозволяє зменшити розгалуження без втрати точності. Зайві гілки видаляються або замінюються піддеревами, які не погіршують якість класифікації. Такі дерева називають усіченими.

Якщо навіть усічене дерево є занадто складним, з нього можна витягти правила. Кожен

шлях від кореня до листка є окремим правилом з умовами, які визначають приналежність до певного класу.

### 5.3 Алгоритми

На сьогоднішній день існує велика кількість алгоритмів, що реалізують дерева рішень: CART, C4.5, CHAID, CN2, Newid, Itrule і інші.

Алгоритм CART (Classification and Regression Tree), як видно з назви, вирішує задачу класифікації й регресії. Він розроблений в 1974-1984 роках чотирма професорами статистики – Leo Breiman (Berkeley), Jerry Friedman (Stanford), Charles Stone (Berkeley) і Richard Olshen (Stanford).

Атрибути набору даних можуть мати як дискретне, так і числове значення.

Алгоритм CART призначений для побудови бінарного дерева рішень. Бінарні дерева також називають двійковими. Приклад такого дерева розглядався на початку лекції.

Інші особливості алгоритму CART:

- функція оцінки якості розбивки;
- механізм відсікання дерева;
- алгоритм обробки пропущених значень;
- побудова дерев регресії.

Кожний вузол бінарного дерева при розбивці має тільки двох нащадків, що називаються дочірніми галузями. Подальший поділ гілок залежить від того, чи багато вихідних даних описує дана гілка. На кожному кроці побудови дерева правило, формоване у вузлі, ділить задану множину прикладів на дві частини. Права його частина (гілка right) – це та частина множини, у якій правило виконується; ліва (гілка left) – та, для якої правило не виконується.

Функція оцінки якості розбивки, яка використовується для вибору оптимального правила, – індекс Gini – був описаний вище. Відзначимо, що дана оціночна функція заснована на ідеї зменшення невизначеності у вузлі. Допустимо, є вузол, і він розбитий на два класи. Максимальна невизначеність у вузлі буде досягнута при розбивці його на дві підмножини по 50 прикладів, а максимальна визначеність – при розбивці на 100 і 0 прикладів.

Правила розбивки. Нагадаємо, що алгоритм CART працює із числовими й категоріальними атрибутами. У кожному вузлі розбивка може йти тільки по одному атрибуту. Якщо атрибут є числовим, то у внутрішньому вузлі формується правило виду xi <= c, Значення c у більшості випадків вибирається як середнє арифметичне двох сусідніх впорядкованих значень змінної xi навчального набору даних. Якщо ж атрибут відноситься до категоріального типу, то у внутрішньому вузлі формується правило xi V(xi), де V(xi) – деяка непорожня підмножина множин значень змінної xi у навчальному наборі даних [8].

Механізм відсікання. Цим механізмом, що має назву minimal cost-complexity tree pruning, алгоритм CART принципово відрізняється від інших алгоритмів конструювання дерев рішень. У розглянутому алгоритмі відсікання – це деякий компроміс між одержанням дерева «підходящого розміру» і одержанням найбільш точної оцінки класифікації. Метод полягає в одержанні послідовності зменшуваних дерев, але дерева розглядаються не всі, а тільки «кращі представники».

Перехресна перевірка (V-fold cross-validation) є найбільш складною й одночасно оригінальною частиною алгоритму CART. Вона становить шлях вибору остаточного дерева, за умови, що набір даних має невеликий обсяг або ж записи набору даних настільки специфічні, що розділити набір на навчальну й тестову вибірку не представляється можливим.

Отже, основні характеристики алгоритму CART: бінарне розщеплення, критерій розщеплення – індекс Gini, алгоритми minimal cost-complexity tree pruning і V-fold cross – validation, принцип «виростити дерево, а потім скоротити», висока швидкість побудови, обробка пропущених значень.

Алгоритм C4.5 будує дерево рішень з необмеженою кількістю гілок у вузла. Даний алгоритм може працювати тільки з дискретним залежним атрибутом і тому може вирішувати тільки задачу класифікації. C4.5 вважається одним з найвідоміших і широко використовуваних алгоритмів побудови дерев класифікації.

Для роботи алгоритму C4.5 необхідне дотримання наступних вимог:

Кожний запис набору даних повинен бути асоційованим з одним з визначених класів, тобто один з атрибутів набору даних повинен бути міткою класу.

Класи повинні бути дискретними. Кожний приклад повинен однозначно відноситися до одного із класів.

Кількість класів повинна бути значно менше кількості записів у досліджуваному наборі даних.

Остання версія алгоритму – алгоритм C4.8 – реалізована в інструменті Weka як J4.8 (Java). Комерційна реалізація методу: C5.0, розроблювач Rulequest, Австралія.

Ми розглянули два відомі алгоритми побудови дерев рішень CART і C4.5. Обидва алгоритми є робастними, тобто стійкими до шумів і викидів даних.

Алгоритми побудови дерев рішень відрізняються наступними характеристиками:

- вид розщеплення – бінарне (binary), множинне (multi-way);
- критерії розщеплення – ентропія, gini, інші;
- можливість обробки пропущених значень;
- процедура скорочення гілок або відсікання;
- можливості витягування правил з дерев.

Жоден алгоритм побудови дерева не можна апріорі вважати найкращим або

досконалим, підтвердження доцільності використання конкретного алгоритму повинне бути перевірене й підтверджене експериментом.

Розробка нових масштабованих алгоритмів. Найбільш серйозна вимога, яка зараз пред'являється до алгоритмів конструювання дерев рішень – це масштабованість, тобто алгоритм повинен мати масштабований метод доступу до даних.

Розроблений ряд нових масштабованих алгоритмів, серед них – алгоритм Sprint, запропонований Джоном Боярином і його колегами. Sprint, що є масштабованим варіантом розглянутого в лекції алгоритму CART, висуває мінімальні вимоги до об'єму оперативної пам'яті.

### 5.4 Метод опорних векторів

У попередніх темі ми розглянули такі методи класифікації й прогнозування як лінійна регресія й дерева рішень; у цій лекції ми продовжимо знайомство з методами цієї групи й розглянемо наступні з них: метод опорних векторів, метод найближчого сусіда (метод міркувань на основі прецедентів) і баєсівську класифікацію.

Метод опорних векторів (Support Vector Machine – SVM) відноситься до групи граничних методів. Він визначає класи за допомогою границь областей.

За допомогою даного методу вирішуються задачі бінарної класифікації. В основі методу лежить поняття площин розв'язків.

Площина (plane) розв'язку розділяє об'єкти з різною класовою приналежністю.

На рисунку 5.3 наведений приклад, у якому беруть участь об'єкти двох типів. Поділяюча лінія задає границю, праворуч від якої – усі об'єкти типу brown (коричневий), а ліворуч – типу yellow (жовтий). Новий об'єкт, що потрапляє праворуч, класифікується як об'єкт класу brown або – як об'єкт класу yellow, якщо він розташувався ліворуч від поділяючої прямої. У цьому випадку кожний об'єкт характеризується двома вимірами.

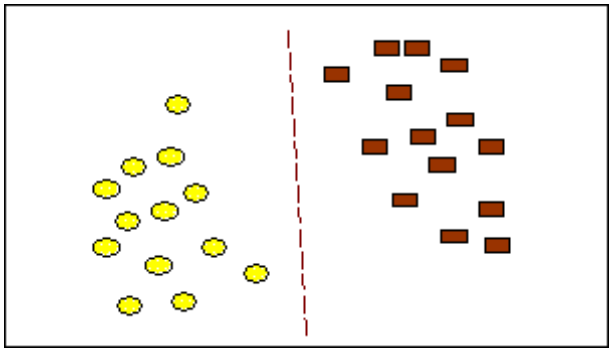


Рисунок 5.3 – Поділ класів прямою лінією

Ціль методу опорних векторів – знайти площину, що розділяє дві множини об'єктів;

така площина показана на рисунку 5.4. На цьому малюнку множина зразків поділена на два класи: жовті об’єкти належать класу A, коричневі – класу B.

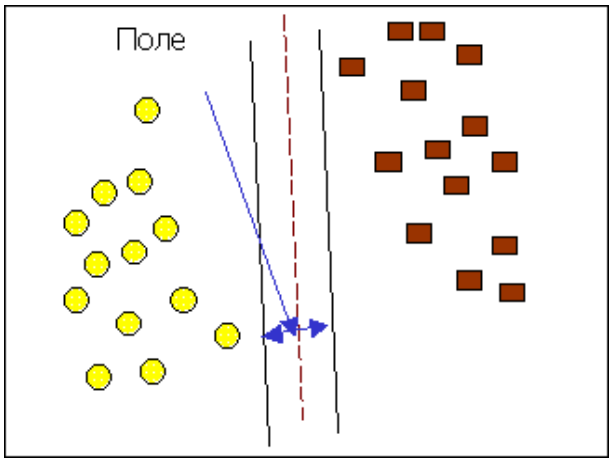



Рисунок 5.4 – До визначення опорних векторів

Метод відшукує зразки, що перебувають на границях між двома класами, тобто опорні вектори; вони зображені на  рисунку 5.5.

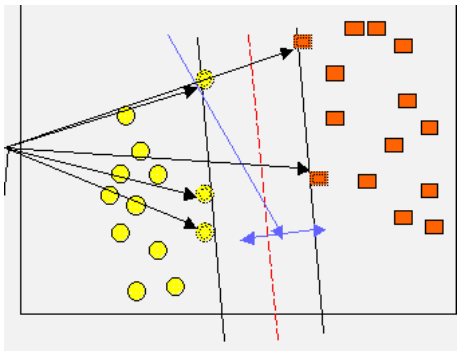


Рисунок 5.5 – Опорні вектори

Опорними векторами називаються об’єкти множини, що лежать на границях областей. Класифікація вважається гарною, якщо область між границями порожня.

На  рисунку 5.5 показано п’ять векторів, які є опорними для даної множини.

### 5.5 Лінійний SVM

Розв’язок задачі бінарної класифікації за допомогою методу опорних векторів полягає в пошуку деякої лінійної функції, яка правильно розділяє набір даних на два класи. Розглянемо задачу класифікації, де число класів рівне двом.

Задачу можна сформулювати як пошук функції f(x), що приймає значення менше нуля для векторів одного класу й більше нуля – для векторів іншого класу. У якості вихідних даних

для розв'язку поставленої задачі, тобто пошуку функції, що класифікує, f(x), дано тренувальний набір векторів простору, для яких відома їхня приналежність до одного із класів. Сімейство функцій, що класифікують, можна описати через функцію f(x). Гіперплощина визначена вектором a і значенням b, тобто f(x)=ax+b. Розв'язок даної задачі проілюстрований на рисунку 5.6.

У результаті розв'язку задачі, тобто побудови Svm-Моделі, знайдена функція, що приймає значення менше нуля для векторів одного класу й більше нуля – для векторів іншого класу. Для кожного нового об'єкта негативне або позитивне значення визначає приналежність об'єкта до одного із класів.

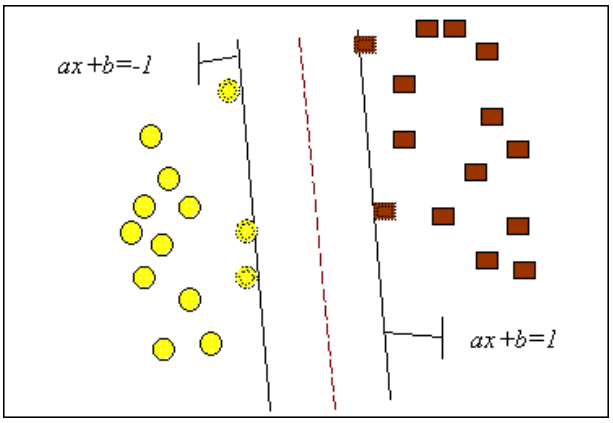



Рисунок 5.6 – Лінійний SVM

Найкращою функцією класифікації є функція, для якої очікуваний ризик мінімальний. Поняття очікуваного ризику в цьому випадку означає очікуваний рівень помилки класифікації.

Прямо оцінити очікуваний рівень помилки побудованої моделі неможливо, це можна зробити за допомогою поняття емпіричного ризику. Однак слід прийняти, що мінімізація останнього не завжди приводить до мінімізації очікуваного ризику. Цю обставину слід пам'ятати при роботі з відносно невеликими наборами тренувальних даних.

Емпіричний ризик – рівень помилки класифікації на тренувальному наборі.

Таким чином, у результаті розв'язку задачі методом опорних векторів для лінійно поділюваних даних ми одержуємо функцію класифікації, яка мінімізує верхню оцінку очікуваного ризику [8].

Однією із проблем, пов'язаних з розв'язком задач класифікації розглянутим методом, є та обставина, що не завжди можна легко знайти лінійну границю між двома класами.

У таких випадках один з варіантів – збільшення розмірності, тобто перенесення даних із площини в тривимірний простір, де можливо побудувати таку площину, яка ідеально розділить множину зразків на два класи. Опорними векторами в цьому випадку будуть служити об'єкти з обох класів, що є екстремальними.

Таким чином, за допомогою додавання так званого оператора ядра й додаткових розмірностей, знаходяться границі між класами у вигляді гіперплощин.

Однак слід пам’ятати: складність побудови Svm-Моделі полягає в тому, що чим вища розмірність простору, тим складніше з ним працювати. Один з варіантів роботи з даними високої розмірності – це попереднє застосування якого-небудь методу зниження розмірності даних для виявлення найбільш істотних компонентів, а потім використання методу опорних векторів.

Як і будь-який інший метод, метод SVM має свої сильні й слабкі сторони, які слід враховувати при виборі даного методу.

Недолік методу полягає в тому, що для класифікації використовується не вся множина зразків, а лише їхня невелика частина, яка перебуває на границях.

Перевага методу полягає в тому, що для класифікації методом опорних векторів, на відміну від більшості інших методів, достатньо невеликого набору даних. При правильній роботі моделі, побудованої на тестовій множині, цілком можливе застосування даного методу на реальних даних.

Метод опорних векторів дозволяє:

одержати функцію класифікації з мінімальною верхньою оцінкою очікуваного ризику (рівня помилки класифікації);

використовувати лінійний класифікатор для роботи з нелінійно поділюваними даними, поєднуючи простоту з ефективністю.

### 5.6 Метод «найближчого сусіда»

Метод «найближчого сусіда» або системи міркувань на основі аналогічних випадків.

Слід відразу зазначити, що метод «найближчого сусіда» («nearest neighbour») відноситься до класу методів, робота яких ґрунтується на зберіганні даних у пам’яті для порівняння з новими елементами. З появою нового запису для прогнозування мають місце відхилення між цим записом і подібними наборами даних, і ідентифікується найбільш подібний (або близький сусід).

Наприклад, при розгляді нового клієнта банку, його атрибути порівнюються з усіма існуючими клієнтами даного банку (дохід, вік і т.д.). Множина «найближчих сусідів» потенційного клієнта банку вибирається на підставі найближчого значення доходу, віку і т.д.

При такому підході використовується термін «k-найближчий сусід» («k-nearest neighbour»). Термін означає, що вибирається k «верхніх» (найближчих) сусідів для їхнього розгляду як множини «найближчих сусідів». Оскільки не завжди зручно зберігати всі дані, іноді зберігається тільки множина «типових» випадків. У такому випадку використовуваний

метод називають міркуванням за аналогією (Case Based Reasoning, CBR), міркуванням на основі аналогічних випадків, міркуванням по прецедентах.

Прецедент – це опис ситуації в комбінації з докладною вказівкою дій, що застосовують у даній ситуації.

Підхід, заснований на прецедентах, умовно можна поділити на наступні етапи:

- збір докладної інформації про поставлене завдання;
- зіставлення цієї інформації з деталями прецедентів, що зберігаються в базі, для виявлення аналогічних випадків;
- вибір прецеденту, найбільш близького до поточної проблеми, з бази прецедентів;
- адаптація обраного розв'язку до поточної проблеми, якщо це необхідно;
- перевірка коректності кожного нового отриманого розв'язку;
- занесення детальної інформації про новий прецедент у базу прецедентів.

Таким чином, висновок, заснований на прецедентах, становить такий метод аналізу даних, який робить висновок щодо даної ситуації за результатами пошуку аналогій, що зберігаються в базі прецедентів.

Даний метод по своїй суті належить до категорії «навчання без вчителя», тобто являється технологією «що навчається самостійно», завдяки чому робочі характеристики кожної бази прецедентів з ходом часу і накопиченням прикладів покращуються. Розробка баз прецедентів по конкретній предметній області відбувається на природній для людини мові, отже, може бути виконана найбільш досвідченими співробітниками компанії – експертами або аналітиками, що працюють у даній предметній області.

Однак це не означає, що CBR – системи самостійно можуть ухвалювати рішення. Останнє завжди залишається за людиною, даний метод лише пропонує можливі варіанти розв'язку й указує на «найрозумніший» з її точки зору.

Переваги методу:

- простота використання отриманих результатів;
- розв'язки не унікальні для конкретної ситуації, можливе їх використання для інших випадків;
- метою пошуку є не гарантовано вірний розв'язок, а кращий з можливих.

Недоліки методу «найближчого сусіда».

Даний метод не створює яких-небудь моделей або правил, що узагальнюють попередній досвід, – у виборі розв'язку вони ґрунтуються на всьому масиві доступних історичних даних, тому неможливо сказати, на якій підставі будуються відповіді.

Існує складність вибору заходу «близькості» (метрики). Від цього заходу головним чином залежить обсяг множини записів, які потрібно зберігати в пам'яті для досягнення задовільної класифікації або прогнозу. Також існує висока залежність результатів класифікації

від обраної метрики.

При використанні методу виникає необхідність повного перебору навчальної вибірки при розпізнаванні, як наслідок цього – обчислювальна трудомісткість.

Типові завдання даного методу – це завдання невеликої розмірності за кількістю класів і змінних.

За допомогою даного методу вирішуються задача класифікації й регресії.

Розглянемо докладно принципи роботи методу k-найближчих сусідів для розв'язку задач класифікації й регресії (прогнозування).

Розв'язок задачі класифікації нових об'єктів. Ця задача схематично зображена на рисунку 5.7. Приклади (відомі екземпляри) відзначені знайомими «+» або «-», що визначають приналежність до відповідного класу («+» або «-»), а новий об'єкт, який потрібно класифікувати, позначений кружечком. Нові об'єкти також називають точками запиту [9].

Наша мета заключається в оцінці (класифікації) відгуку точок запиту з використанням спеціально обраного числа їх найближчих сусідів. Інакше кажучи, ми прагнемо довідатися, до якого класу слід віднести точку запиту: знак «+» або як знак «-».

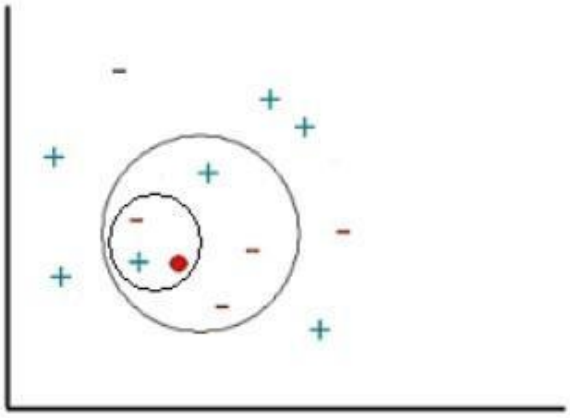


Рисунок 5.7 – Класифікація об'єктів множини при різному значенні параметра k

Для початку розглянемо результат роботи методу k-найближчих сусідів з використанням одного найближчого сусіда. У цьому випадку відгук точки запиту буде класифікований як знак плюс, тому що найближча сусідня точка має знак плюс.

Тепер збільшимо число використовуваних найближчих сусідів до двох. Цього разу метод k – найближчих сусідів не зможе класифікувати відгук точки запиту, оскільки друга найближча точка має знак мінус і обидва знаки рівноцінні (тобто перемога з однаковою кількістю голосів).

Далі збільшимо число використовуваних найближчих сусідів до 5. Таким чином, буде визначена ціла околиця точки запиту (на графіку її границя відзначена червоним (сірим) колом). Тому що в області утримується 2 точки зі знаком «+» і 3 точки зі знаком «-», алгоритм

k-найближчих сусідів привласнить знак «-» відгуку точки запиту.

Розв’язок задачі прогнозування. Далі розглянемо принцип роботи методу k-найближчих сусідів для розв’язку задачі регресії. Регресійні задачі пов’язані із прогнозуванням значення залежної змінної за значеннями незалежних змінних набору даних.

Розглянемо графік, показаний на рисунку 5.8. Зображений на ній набір точок (зелені прямокутники) отриманий по зв’язку між незалежною змінною x і залежною змінною y (крива червоного кольору). Заданий набір зелених об’єктів (тобто набір прикладів); ми використовуємо метод k-найближчих сусідів для прогнозування виходу точки запиту X по даному набору прикладів (зелені прямокутники).

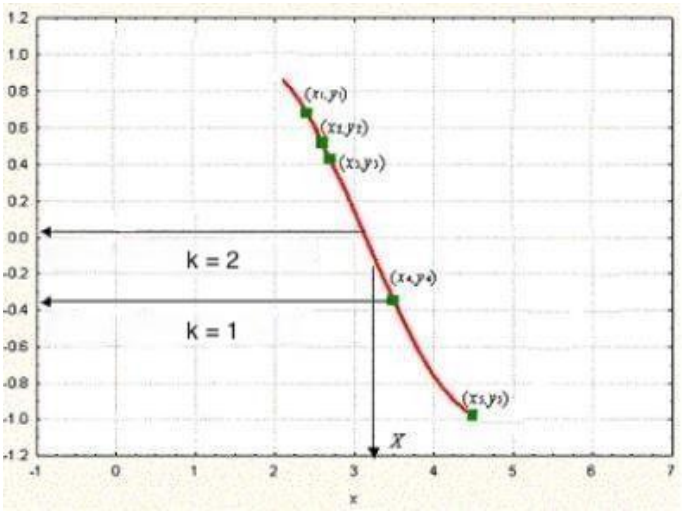



Рисунок 5.8 – Розв’язок задачі прогнозування при різних значеннях параметра k

Спочатку розглянемо як приклад метод k-найближчих сусідів з використанням одного найближчого сусіда, тобто при k, рівному одиниці. Ми шукаємо набір прикладів (зелені прямокутники) і виділяємо з їхнього числа найближчий до точки запиту X. Для нашого випадку найближчий приклад – точка (x4; y4). Вихід x4 (тобто y4), таким чином, приймається в якості результату прогнозування виходу X (тобто Y). Отже, для одного найближчого сусіда можемо записати: вихід Y рівний y4 (Y = y4).

Далі розглянемо ситуацію, коли k рівне двом, тобто розглянемо два найближчі сусіди. У цьому випадку ми виділяємо вже дві найближчі до X точки. На нашому графіку це точки y3 і y4 відповідно. Обчисливши середнє їхніх виходів, записуємо розв’язок для Y у вигляді Y = (y3 + y4)/2.

Розв’язок задачі прогнозування здійснюється шляхом перенесення описаних вище дій на використання довільного числа найближчих сусідів таким чином, що вихід Y точки запиту X обчислюється як середньоарифметичне значення виходів k-найближчих сусідів точки запиту.

Незалежні й залежні змінні набору даних можуть бути як безперервними, так і категоріальними. Для безперервних залежних змінних задача розглядається як задача прогнозування, для дискретних змінних – як задача класифікації.

Прогнозування в задачі прогнозування виходить усередненням виходів k-найближчих сусідів, а розв'язок задачі класифікації заснований на принципі «за більшістю голосів».

Критичним моментом у використанні методу k-найближчих сусідів є вибір параметра k. Він один з найбільш важливих факторів, що визначають якість прогнозної або класифікаційної моделі.

Якщо обране занадто мале значення параметра k, виникає ймовірність великого розкиду значень прогнозу. Якщо обране значення занадто велике, це може привести до сильного зміщення моделі. Таким чином, ми бачимо, що повинне бути обране оптимальне значення параметра k. Тобто це значення повинне бути настільки великим, щоб звести до мінімуму ймовірність неправильної класифікації, і одночасно, досить малим, щоб k сусідів були розташовані досить близько до точки запиту.

Таким чином, ми розглядаємо k параметр, як згладжуючий, для якого повинен бути знайдений компроміс між силою розмаху (розкиду) моделі і її зміщеністю.

Один з варіантів оцінки параметра k – проведення крос-перевірки (Bishop, 1995). Така процедура реалізована, наприклад, у пакеті STATISTICA (Statsoft).

Крос-Перевірка – відомий метод одержання оцінок невідомих параметрів моделі. Основна ідея методу – поділ вибірки даних на v «складок». V «складки» тут є випадковим чином виділені ізольовані підвибірки.

За фіксованим значенням k будується модель k-найближчих сусідів для одержання прогнозів на v-му сегменті (інші сегменти при цьому використовуються як приклади) і оцінюється помилка класифікації. Для регресійних задач найбільш часто в якості оцінки помилки виступає сума квадратів, а для класифікаційних задач зручніше розглядати точність (відсоток коректно класифікованих спостережень).

Далі процес послідовно повторюється для всіх можливих варіантів вибору v. По вичерпанню v «складок» (циклів), обчислені помилки усереднюють й використовуються в якості міри стабільності моделі (тобто міри якості прогнозування в точках запиту). Вищеописані дії повторюються для різних k, і значення, що відповідає найменшій помилці (або найбільшій класифікаційній точності), приймається як оптимальне (оптимальне в сенсі методу крос-перевірки) [9].

Слід враховувати, що крос-перевірка – ємнісна з точки зору обчислень процедура, і необхідно надати час для роботи алгоритму, особливо якщо обсяг вибірки досить великий.

Другий варіант вибору значення параметра k – самостійно задати його значення. Однак

цей спосіб слід використовувати, якщо є обґрунтовані припущення щодо можливого значення параметра, наприклад, про попередні дослідження подібних наборів даних.

Метод k-найближчих сусідів показує досить непогані результати в найрізноманітніших задачах.

Прикладом реального використання описаного вище методу є програмне забезпечення центру технічної підтримки компанії Dell, розроблене компанією Inference. Ця система допомагає співробітникам центру відповідати на велике число запитів, відразу пропонуючи відповіді на розповсюджені питання й дозволяючи звертатися до бази під час розмови по телефону з користувачем. Співробітники центру технічної підтримки, завдяки реалізації цього методу, можуть відповідати одночасно на значне число дзвінків. Програмне забезпечення CBR зараз розгорнуте в мережі Intranet компанії Dell.

Інструментів Data Mining, що реалізують метод k-найближчих сусідів і CBR-Метод, не дуже багато. Серед найбільш відомих: CBR Express і Case Point (Inference Corp.), Apriori (Answer Systems), DP Umbrella (VYCOR Corp.), KATE tools (Acknosoft, Франція), Pattern Recognition Workbench (Unica, США), а також деякі статистичні пакети, наприклад, Statistica.

### 5.7 Байєсівська класифікація

Теорема Баєсса – одна з основних теорем теорії ймовірностей, яка визначає ймовірність настання події, коли відома тільки часткова інформація про подію. Названа на честь Томаса Баєса. У теорії ймовірностей і статистиці, теорема Байеса це теорема з двома різними інтерпретаціями. В байєсівсій інтерпретації, раціонально виразити, суб’єктивну ступінь віри у разі зміни показань для обліку. У «частотній інтерпретації», теорема стосується подання зворотної ймовірності двом подіям. В байєсівській інтерпретації, теорема Байеса має основоположне значення для байєсівської статистики та програми в полях у тому числі науки, техніки, медицини та права. Реалізація теореми Байеса для поновлення висновків про ймовірності називається байєсівським висновком. який запропонував використовувати першу теорему оновити вірування. Тим не менш, історичні праці, опубліковані посмертно WAS.

Байєсовський класифікатор – широкий клас алгоритмів класифікації, заснований на принципі максимуму апостеріорної ймовірності. Для класифікованого об’єкта обчислюються функції правдоподібності кожного з класів, по них обчислюються апостеріорні ймовірності класів. Об’єкт відноситься до того класу, для якого апостеріорна ймовірність максимальна [9].

Апостеріорна ймовірність – умовна ймовірність випадкової події за умови того, що відомі апостеріорні дані, тобто отримані після досліду.

Альтернативні назви: байєсовське моделювання, Байєсівська статистика, метод байєсовських мереж.

Споконвічно байєсовська класифікація використовувалася для формалізації знань експертів в експертних системах, зараз Байєсівська класифікація також застосовується в якості одного з методів Data Mining.

Так звана наївна класифікація або наївно-байєсовський підхід (naive-bayes approach) є найбільш простим варіантом методу, що використовує байєсовські мережі. При цьому підході вирішуються задача класифікації, результатом роботи методу є так звані «прозорі» моделі.

«Наївна» класифікація є простим і зрозумілим методом машинного навчання. Її назва походить від головного припущення, яке лежить в основі алгоритму: усі ознаки (змінні), що використовуються для прийняття рішення, є статистично незалежними одна від одної.

Це припущення робить модель «наївною», оскільки в реальних даних ознаки часто корелюють між собою. Проте, попри свою спрощеність, наївна класифікація часто дає доволі точні результати.

Ключові властивості наївної класифікації.

Урахування всіх змінних. Алгоритм використовує всі доступні ознаки для прийняття рішення, не відкидаючи жодної з них.

Два основні припущення про змінні: усі змінні є однаково важливими та усі змінні є статистично незалежними, тобто значення однієї змінної не впливає на значення іншої.

Більшість інших методів класифікації припускають, що перед початком класифікації ймовірність того, що об’єкт належить тому або іншому класу, однакова; але це не завжди правильно.

Допустимо, відомо, що певний відсоток даних належить конкретному класу. Виникає питання, чи можемо ми використовувати цю інформацію при побудові моделі класифікації? Існує множина реальних прикладів використання цих апріорних знань, що допомагають класифікувати об’єкти. Типовий приклад з медичної практики. Якщо лікар відправляє результати аналізів пацієнта на додаткове дослідження, він відносить пацієнта до якогось певного класу. Яким чином можна застосувати цю інформацію? Ми можемо використовувати її як додаткові дані при побудові класифікаційної моделі.

Відзначають такі переваги байєсовських мереж як методу Data Mining:

- у моделі визначаються залежності між усіма змінними, це дозволяє легко обробляти ситуації, у яких значення деяких змінних невідомі;
- байєсовські мережі досить просто інтерпретуються й дозволяють на етапі прогностичного моделювання легко проводити аналіз за сценарієм «що, якщо»;
- байєсовський метод дозволяє природно сполучати закономірності, виведені з даних, і, наприклад, експертні знання, отримані в явному вигляді;
- використання байєсовських мереж дозволяє уникнути проблеми переучування (overfitting), тобто надлишкового ускладнення моделі, що є слабкою стороною багатьох

методів (наприклад, дерев рішень і нейронних мереж).

Наївно-байєсовський підхід має наступні недоліки:

- перемножувати умовні ймовірності коректно тільки тоді, коли всі вхідні змінні дійсно статистично незалежні; хоча часто даний метод показує досить гарні результати при недотриманні умови статистичної незалежності, але теоретично така ситуація повинна оброблятися більш складними методами, заснованими на навчанні байєсовських мереж;
- неможлива безпосередня обробка безперервних змінних – потрібно їхнє перетворення до інтервальної шкали, щоб атрибути були дискретними; однак такі перетворення іноді можуть приводити до втрати значимих закономірностей.

# ЛЕКЦІЯ 6
# НЕЙРОННІ МЕРЕЖІ. КАРТИ КОХОНЕНА, ЩО САМООРГАНІЗУЮТЬСЯ. МЕТОДИ ПОШУКУ АСОЦІАТИВНИХ ПРАВИЛ

## 6.1 Класифікація нейронних мереж

Одна з можливих класифікацій нейронних мереж – за спрямованістю зв'язків. Нейронні мережі бувають зі зворотними зв'язками й без зворотних зв'язків.

Мережі без зворотних зв'язків:

- мережі зі зворотним поширенням помилки. Мережі цієї групи характеризуються фіксованою структурою, ітераційним навчанням, коректуванням ваг за помилками;
- інші мережі (когнитрон, неокогнитрон, інші складні моделі).

Перевагами мереж без зворотних зв'язків є простота їх реалізації й гарантоване одержання відповіді після проходження даних по шарах. Менший обсяг мережі полегшує процес навчання.

Недоліком цього виду мереж вважається мінімізація розмірів мережі – нейрони багаторазово беруть участь в обробці даних.

Мережі зі зворотними зв'язками

- мережі Хопфілда (задачі асоціативної пам'яті);
- мережі Кохонена (задачі кластерного аналізу).

Перевагами мереж зі зворотними зв'язками є складність навчання, викликана більшим числом нейронів для алгоритмів того самого рівня складності.

Недоліки цього виду мереж – потрібні спеціальні умови, що гарантують збіжність обчислень.

Інша класифікація нейронних мереж: мережі прямого поширення й рекуррентні мережі.

Мережі прямого поширення:

- персептрони;
- мережа Back Propagation;
- мережа зустрічного поширення;
- карта Кохонена.

Рекуррентні мережі. Характерна риса таких мереж – наявність блоків динамічної затримки й зворотних зв'язків, що дозволяє їм обробляти динамічні моделі.

Мережа Хопфілда – це різновид рекурентної нейронної мережі з пов'язаною структурою, в якій кожен нейрон з'єднаний з усіма іншими. Вона працює як асоціативна пам'ять і здатна зберігати та відновлювати патерни (шаблони) за неповними або спотвореними вхідними даними.

Мережа Елмана – це двошарова рекурентна нейронна мережа, в якій схований

(прихований) шар має зворотний зв’язок через спеціальні контекстні вузли. Такий механізм дозволяє враховувати передісторію змін у вхідних даних, накопичувати інформацію та формувати відповідну стратегію реагування.

Нейронні мережі можуть навчатися із учителем або без нього.

При навчанні із учителем для кожного навчального вхідного прикладу потрібне знання правильної відповіді або функції оцінки якості відповіді. Таке навчання називають керованим. Нейронній мережі пред’являються значення вхідних і вихідних сигналів, а вона за певним алгоритмом підбудовує ваги синоптичних зв’язків. У процесі навчання проводиться коректування ваг мережі за результатами порівняння фактичних вихідних значень із вхідними, відомими заздалегідь.

При навчанні без учителя розкривається внутрішня структура даних або кореляції між зразками в наборі даних. Виходи нейронної мережі формуються самостійно, а ваги змінюються за алгоритмом, що враховує тільки вхідні й похідні від них сигнали. Це навчання називають також некерованим. У результаті такого навчання об’єкти або приклади розподіляються по категоріях, самі категорії і їх кількість можуть бути заздалегідь не відомі.

Підготовка даних для навчання. При підготовці даних для навчання нейронної мережі необхідно звертати увагу на наступні істотні моменти.

Кількість спостережень у наборі даних. Слід враховувати той фактор, що чим більше розмірність даних, тим більше часу буде потрібно для навчання мережі.

Робота з викидами. Слід визначити наявність викидів і оцінити необхідність їх присутності у вибірці.

Навчальна вибірка повинна бути представницькою (репрезентативною), вона не повинна містити протиріч, тому що нейронна мережа однозначно зіставляє вихідні значення вхідним.

Нейронна мережа працює тільки із числовими вхідними даними, тому важливим етапом при підготовці даних є перетворення й кодування даних.

При використанні на вхід нейронної мережі слід подавати значення з того діапазону, на якому вона навчалася. Наприклад, якщо при навчанні нейронної мережі на один з її входів подавалися значення від 0 до 10, то при її застосуванні на вхід слід подавати значення із цього ж діапазону або прилеглих.

Нормалізація даних. Метою нормалізації значень є перетворення даних до вигляду, який найбільше підходить для обробки, тобто дані, що надходять на вхід, повинні мати числовий тип, а їх значення повинні бути розподілені в певному діапазоні. Нормалізатор може приводити дискретні дані до набору унікальних індексів або перетворювати значення, що лежать в довільному діапазоні, у конкретний діапазон. Нормалізація виконується шляхом розподілу кожного компонента вхідного вектора на довжину вектора, що перетворює вхідний

вектор в одиничний.

### 6.2 Вибір структури нейронної мережі

Вибір структури нейронної мережі обумовлюється специфікою й складністю розв’язуваної задачі. Для розв’язку деяких типів задач розроблені оптимальні конфігурації.

У більшості випадків вибір структури нейронної мережі визначається на основі об’єднання досвіду й інтуїції розроблювача.

Однак існують основні принципи, якими слід керуватися при розробці нової конфігурації:

- можливості мережі зростають зі збільшенням числа гнізд мережі, щільності зв’язків між ними й числа виділених шарів;
- введення зворотних зв’язків поряд зі збільшенням можливостей мережі піднімає питання про динамічну стабільність мережі;
- складність алгоритмів функціонування мережі (у тому числі, наприклад, введення декількох типів синапсів – збуджуючих, гальмуючих та ін.) також сприяє посиленню потужності нейронної мережі.

Питання про необхідні й достатні властивості мережі для розв’язку того або іншого роду задач становить цілий напрямок нейронної комп’ютерної науки. Тому що проблема синтезу нейронної мережі сильно залежить від розв’язуваної задачі, дати загальні докладні рекомендації важко. Очевидно, що процес функціонування НМ (нейронної мережі), тобто сутність дій, які вона здатна виконувати, залежить від величин синаптичних зв’язків, тому, задавшись певною структурою НМ, що відповідає якому-небудь завданню, розроблювач мережі повинен знайти оптимальні значення всіх змінних вагових коефіцієнтів (деякі синаптичні зв’язки можуть бути постійними) [10].

### 6.3 Карти Кохонена

Карти Кохонена, карти, що самоорганізуються (Self-Organizing Maps). Мережі, що називаються картами Кохонена, – це один з різновидів нейронних мереж, однак вони принципово відрізняються від розглянутих вище, оскільки використовують неконтрольоване навчання. Нагадаємо, що при такому навчанні навчальна множина складається лише зі значень вхідних змінних, у процесі навчання немає порівняння виходів нейронів з еталонними значеннями. Можна сказати, що така мережа вчиться розуміти структуру даних.

В основі ідеї мережі Кохонена лежить аналогія із властивостями людського мозку. Кора головного мозку людину становить плоский аркуш зі згорнутими складками. Таким чином,

можна сказати, що вона має певні топологічні властивості (ділянки, відповідальні за близькі частини тіла, примикають одна до однієї й усе зображення людського тіла відображається на цю двовимірну поверхню).

Найпоширеніше застосування мереж Кохонена – розв'язок задачі класифікації без учителя, тобто кластеризації.

Два з розповсюджених застосувань карт Кохонена: розвідницький аналіз даних і виявлення нових явищ.

Розвідницький аналіз даних. Мережа Кохонена здатна розпізнавати кластери в даних, а також установлювати близькість класів. Таким чином, користувач може поліпшити своє розуміння структури даних, щоб потім уточнити нейромережеву модель. Якщо в даних розпізнані класи, то їх можна позначити, після чого мережа зможе вирішувати задачу класифікації. Мережі Кохонена можна використовувати й у тих задачах класифікації, де класи вже задані – тоді перевага буде в тому, що мережа зможе виявити подібність між різними класами.

Виявлення нових явищ. Мережа Кохонена розпізнає кластери в навчальних даних і відносить усі дані до тем або інших кластерів. Якщо після цього мережа зустрінеться з набором даних, несхожим ні на один з відомих зразків, то вона не зможе класифікувати такий набір і тим самим виявить його новизну.

Мережа Кохонена, на відміну від багатошарової нейронної мережі, дуже проста; вона становить два шари: вхідний і вихідний. Її також називають самоорганізованою картою. Елементи карти розташовуються в деякому просторі, як правило, двовимірному. Мережа Кохонена зображена на рисунку 6.1

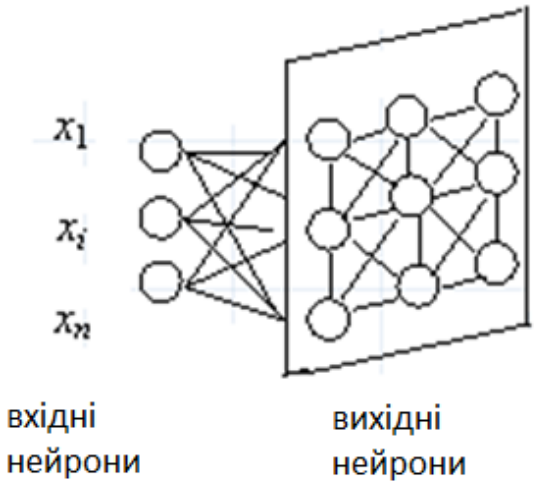



Рисунок 6.1 – Мережа Кохонена

Мережа Кохонена навчається методом послідовних наближень. У процесі навчання таких мереж на входи подаються дані, але мережа при цьому підбудовується не під еталонне

значення виходу, а під закономірності у вхідних даних. Починається навчання з обраного випадковим чином вихідного розташування центрів.

У процесі послідовної подачі на вхід мережі навчальних прикладів визначається найбільш схожий нейрон (той, у якого скалярний добуток ваг і поданого на вхід вектора мінімальні). Цей нейрон оголошується переможцем і є центром при підстроюванні ваг у сусідніх нейронів. Таке правило навчання припускає «змагальне» навчання з урахуванням відстані нейронів від «нейрона-переможця».

Навчання при цьому полягає не в мінімізації помилки, а в підстроюванні ваг (внутрішніх параметрів нейронної мережі) для найбільшого збігу із вхідними даними.

Основний ітераційний алгоритм Кохонена послідовно проходить ряд епох, на кожній з яких обробляється один приклад з навчальної вибірки. Вхідні сигнали послідовно пред'являються мережі, при цьому бажані вихідні сигнали не визначаються. Після пред'явлення достатнього числа вхідних векторів синаптичні ваги мережі стають здатні визначити кластери. Ваги організують так, що топологічно близькі вузли чутливі до схожих вхідних сигналів.

У результаті роботи алгоритму центр кластера встановлюється в певній позиції, задовільним чином кластеризують приклади, для яких даний нейрон є «переможцем». У результаті навчання мережі необхідно визначити міру сусідства нейронів, тобто околицю нейрона-переможця.

Околиця становить кілька нейронів, які оточують нейрона-переможця. Спочатку до околиці належить велика кількість нейронів, далі її розмір поступово зменшується. Мережа формує топологічну структуру, у якій схожі приклади утворюють групи прикладів, що близько перебувають на топологічній карті.

Отриману карту можна використовувати як засіб візуалізації при аналізі даних. У результаті навчання карта Кохонена класифікує вхідні приклади на кластери (групи схожих прикладів) і візуально відображає багатомірні вхідні дані на площині нейронів.

Унікальність методу карт, що самоорганизуються, полягає в перетворенні n-вимірного простору в двовимірний. Застосування двовимірних сіток пов'язане з тим, що існує проблема відображення просторових структур більшої розмірності.

Маючи таке представлення даних, можна візуально визначити наявність або відсутність взаємозв'язку у вхідних даних.

Нейрони карти Кохонена розташовують у вигляді двомірної матриці, розфарбовують цю матрицю залежно від аналізованих параметрів нейронів.

На рисунку 6.2 наведений приклад карти Кохонена.

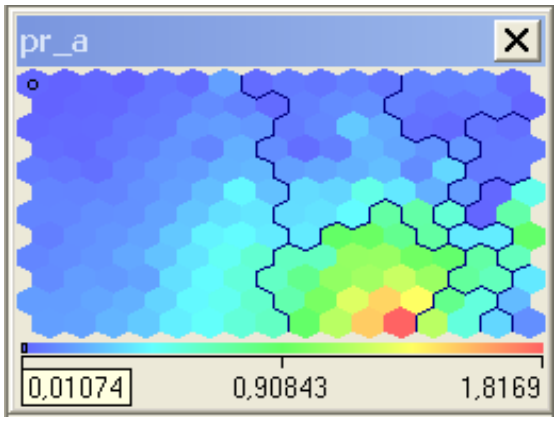



Рисунок 6.2 – Приклад карти Кохонена

Карти Кохонена (як і географічні карти) можна відображати:

- у двомірному вигляді, тоді карта розфарбовується відповідно до рівня виходу нейрона;
- у тривимірному вигляді.

У результаті роботи алгоритму одержуємо такі карти:

- карта входів нейронів;
- карта виходів нейронів;
- спеціальні карти.

Координати кожної карти визначають положення одного нейрона. Так, координати визначають нейрон, який перебуває на перетинанні 15-го стовпця з 30-м поруч у матриці нейронів. Розглянемо, що ж являють собою ці карти.

### 6.4 Карта входів та виходів нейронів

Ваги нейронів підбудовуються під значення вхідних змінних і відображають їхню внутрішню структуру. Для кожного входу малюється своя карта, розфарбована у відповідності зі значенням конкретної ваги нейрона.

При аналізі даних використовують кілька карт входів. На одній з карт виділяють область певного кольору – це означає, що відповідні вхідні приклади мають приблизно однакове значення відповідного входу. Колірний розподіл нейронів із цієї області аналізується на інших картах для визначення схожих або відмітних характеристик.

Карта виходів нейронів. На карту виходів нейронів проектується взаємне розташування досліджуваних вхідних даних. Нейрони з однаковими значеннями виходів утворюють кластери – замкнені області на карті, які включають нейрони з однаковими значеннями виходів.

Спеціальні карти. Це карта кластерів, матриця відстаней, матриця щільності потрапляння й інші карти, які характеризують кластери, отримані в результаті навчання

мережі Кохонена.

Важливо розуміти, що між усіма розглянутими картами існує взаємозв'язок – усі вони є різними розфарбуваннями тих самих нейронів. Кожний приклад з навчальної вибірки має те саме розташування на всіх картах.

Приклад розв'язку задачі. Програмне забезпечення, що дозволяє працювати з картами Кохонена, зараз представлене великою кількістю інструментів. Це можуть бути як інструменти, що включають тільки реалізацію методу карт, що самоорганізуються, так і нейропакети з цілим набором структур нейронних мереж, серед котрих – і карти Кохонена; також даний метод реалізований у деяких універсальних інструментах.

До інструментарію, що включає реалізацію методу карт Кохонена, відносяться Somine, Statistica, Neuroshell, Neuroscalp, Deductor і багато інших. Для розв'язку задачі будемо використовувати аналітичний пакет Deductor.

Нехай є база даних комерційних банків з показниками діяльності за поточний період. Необхідно провести їх кластеризацію, тобто виділити однорідні групи банків на основі показників з бази даних, усього показників – 21.

Вихідна таблиця перебуває у файлі «banks.xls». Вона містить показники діяльності комерційних банків за звітний період.

Спочатку імпортуємо дані з xls-файлу в середовище аналітичного пакета.

На першому кроці майстра запускаємо майстер обробки й вибираємо зі списку метод обробки «Карта Кохонена». Далі слід настроїти призначення стовпців, тобто для кожного стовпця вибрати одне із призначень: вхідне, вихідне, не використовується й інформаційне. Вкажемо всім стовпцям, відповідним до показників діяльності банків, призначення «Вхідний». «Вихідний» не призначаємо.

Наступний крок пропонує розбити вихідну множину на навчальну, тестову й валідаційну. За замовчуванням, програма пропонує розбити множину на навчальну – 95% і тестову – 5%.

На кроці № 5 пропонується настроїти параметри карти: кількість гнізд по X і по Y, їх форму (шестикутну або чотирикутну).

На шостому кроці «Настроювання параметрів зупинки навчання», встановлюємо параметри зупинки навчання й установлюємо епоху, по досягненню якої навчання буде припинено.

На сьомому кроці, налаштовуються інші параметри навчання: спосіб початкової ініціалізації, тип функції сусідства. Можливі два варіанти кластеризації: автоматичне визначення числа кластерів з відповідним рівнем значимості й фіксована кількість кластерів (визначається користувачем). Оскільки нам невідома кількість кластерів, виберемо автоматичне визначення їх кількості.

На восьмому кроці запускаємо процес навчання мережі – необхідно натиснути на кнопку «Пуск» і дочекатися закінчення процесу навчання. Під час навчання можемо спостерігати зміну кількості розпізнаних прикладів і поточні значення помилок.

По закінченню навчання в списку візуалізаторів виберемо «Карту Кохонена» і візуалізатор що-якщо. На останньому кроці будуємо відображення карти Кохонена.

### 6.5 Що таке асоціативні правила?

Асоціація – одна із задач Data Mining. Метою пошуку асоціативних правил (association rule) є знаходження закономірностей між зв'язаними подіями в базах даних.

Дуже часто покупці купують не один товар, а декілька. У більшості випадків між цими товарами існує взаємозв'язок. Так, наприклад, покупець, що купує ноутбук, швидше за все, захоче придбати також сумку. Ця інформація може бути використана для розміщення товару на прилавках.

Асоціативні правила, часто знаходять застосування:

- аналіз Web-блогів;
- у роздрібній торгівлі: визначення товарів, які варто просувати спільно; вибір місця розташування товару в магазині; аналіз споживчого кошика; прогнозування попиту;
- перехресні продажі: якщо є інформація про те, що клієнти придбали продукти А, Б і В, те які з них найімовірніше куплять продукт Г;
- маркетинг: пошук ринкових сегментів, тенденцій купівельної поведінки;
- сегментація клієнтів: виявлення загальних характеристик клієнтів компанії, виявлення груп покупців;
- оформлення каталогів, аналіз збутових кампаній фірми, визначення послідовностей покупок клієнтів (яка покупка піде за покупкою товару А).

Приведемо простий приклад асоціативного правила: покупець, що купує ноутбук, придбає до нього мишку з імовірністю 50%.

Введення в асоціативні правила. Уперше задача пошуку асоціативних правил (association rule mining) була запропонована для знаходження типових шаблонів покупок, здійснених у супермаркетах, тому іноді її ще називають аналізом ринкового кошика (market basket analysis).

Ринковий кошик – це набір товарів, придбаних покупцем у рамках однієї окремо взятої транзакції.

Транзакції є досить характерними операціями, ними, наприклад, можуть описуватися результати відвідувань різних магазинів.

Транзакція – це множина подій, які відбулися одночасно.

Реєструючи всі бізнес-операції протягом усього часу своєї діяльності, торговельні компанії накопичують величезні кількості транзакцій. Кожна така транзакція становить набір товарів, куплених покупцем за один візит.

Отримані в результаті аналізу шаблони включають перелік товарів і число транзакцій, які містять дані набори.

Транзакційна або операційна база даних (Transaction database) становить двовимірну таблицю, яка складається з номера транзакції (TID) і переліку покупок, придбаних під час цієї транзакції.

TID – унікальний ідентифікатор, що визначає кожну угоду або транзакцію.

Приклад транзакційної бази даних, що складається з купівельних транзакцій, наведено в таблиці 6.1 У таблиці перший стовпчик (TID) визначає номер транзакції, у другому стовпчику таблиці наведені товари, придбані під час певної транзакції.

Таблиця 6.1 Транзакційна база даних

| TID | Покупки |
|---|---|
| 100 | Карта пам’яті, DVD-диск, USB-подовжувач |
| 200 | DVD-диск, USB-подовжувач, WEB-камера |
| 300 | Комп’ютерна миша, DVD-диск, WEB-камера |
| 400 | USB-подовжувач, DVD-диск, Карта пам’яті, Комп’ютерна миша |
| 500 | DVD-диск, Карта пам’яті |
| 600 | VGA-кабель |

На основі наявної бази даних нам потрібно знайти закономірності між подіями, тобто покупками.

Шаблони, що часто зустрічаються, або зразки. Допустимо, є транзакційна база даних D. Присвоємо значенням товарів змінні (табл. 6.2).

Карта пам’яті = a

DVD-диск = b

USB-подовжувач = c

Комп’ютерна миша = d

WEB – камера = e

VGA – кабель = f

Таблиця 6.2 Набори товарів, що часто зустрічаються

| TID | Покупки | TID | Покупки |
|---|---|---|---|
| 100 | Флешка, DVD – диск, USB – подовжувач | 100 | a, b, c |

| | | | |
|---|---|---|---|
| 200 | DVD – диск, USB – подовжувач, WEB – камера | 200 | b, c, e |
| 300 | Комп'ютерна миша, DVD-диск, WEB-камера | 300 | d, b, e |
| 400 | USB – подовжувач, DVD – диск, Флешка, Комп'ютерна миша | 400 | c, b, a, d |
| 500 | DVD-диск, Флешка, USB – подовжувач | 500 | b, a, c |
| 600 | VGA-кабель | 600 | f |

Розглянемо набір товарів (Itemset), що включає, наприклад, (флешка, DVD – диск, USB – подовжувач). Виразимо цей набір за допомогою змінних:

abc={a,b,c}

Підтримка. Цей набір товарів зустрічається в нашій базі даних три рази, тобто підтримка цього набору товарів рівна 3:

SUP(abc)=3.

При мінімальному рівні підтримки, рівному трьом, набір товарів abc є шаблоном, що часто зустрічається.

min_sup=3, {Карта пам'яті, DVD – диск, USB – подовжувач} – частий шаблон, що зустрічається.

Підтримкою називають кількість або відсоток транзакцій, що містять певний набір даних.

Для даного набору товарів підтримка, виражена у відсотковому відношенні, рівна 50%.

SUP(abc)=(3/6)*100%=50%

Підтримку іноді також називають забезпеченням набору.

Таким чином, набір становить інтерес, якщо його підтримка вище заданого користувачем мінімального значення (min support). Ці набори називають такими, що часто зустрічаються (frequent).

### 6.6 Алгоритми пошуку асоціативних правил

Характеристики асоціативних правил. Асоціативне правило має вигляд: «З події А випливає подія В». У результаті такого виду аналізу ми встановлюємо закономірність наступного виду: «Якщо в транзакції зустрівся набір товарів (або набір елементів) А, то можна зробити висновок, що в цій же транзакції повинен з'явитися набір елементів В)» Встановлення таких закономірностей дає нам можливість знаходити дуже прості й зрозумілі правила, називані асоціативними.

Основними характеристиками асоціативного правила є підтримка й вірогідність правила.

Розглянемо правило «з покупки флешки випливає покупка USB-подовжувача» для бази даних, яка була наведена вище в таблиці 14.1. Поняття підтримки набору ми вже розглянули. Існує поняття підтримки правила.

Правило має підтримку s, якщо s% транзакцій із усього набору містять одночасно набори елементів A і B або, інакше кажучи, містять обидва товари.

Флешка – це товар A, USB-подовжувач – це товар B. Підтримка правила «з покупки флешки випливає покупка USB-подовжувача» рівна 3, або 50%.

Вірогідність правила показує, яка ймовірність того, що з події A випливає подія B.

Правило «З A випливає B» справедливе з вірогідністю C, якщо c% транзакцій із усієї множини, що містить набір елементів A, також містять набір елементів B.

Якщо число транзакцій, що містять USB-подовжувач, рівне чотирьом, а число транзакцій, що містять також і флешку, рівне трьом, то вірогідність правила рівна (3/4)*100%, тобто 75%.

Вірогідність правила «з покупки USB-подовжувача випливає покупка флешки рівна 75%, тобто 75% транзакцій, що містять товар A, також містять товар B».

Границі підтримки й вірогідності асоціативного правила. За допомогою використання алгоритмів пошуку асоціативних правил аналітик може одержати всі можливі правила виду «З A випливає B», з різними значеннями підтримки й вірогідності. Однак у більшості випадків, кількість правил необхідно обмежувати заздалегідь установленими мінімальними й максимальними значеннями підтримки й вірогідності.

Якщо значення підтримки правила занадто велике, то в результаті роботи алгоритму будуть знайдені правила очевидні й добре відомі. Занадто низьке значення підтримки приведе до знаходження дуже великої кількості правил, які, можливо, будуть у більшій частині необґрунтованими, але не відомими й не очевидними для аналітика. Таким чином, необхідно визначити такий інтервал, «золоту середину», який з однієї сторони забезпечить знаходження неочевидних правил, а з іншого – їх обґрунтованість [10].

Якщо рівень вірогідності занадто малий, то цінність правила викликає серйозні сумніви. Наприклад, правило з вірогідністю в 3% тільки умовно можна назвати правилом.

### 6.7 Методи пошуку асоціативних правил

Алгоритм AIS. Перший алгоритм пошуку асоціативних правил, що називався AIS, (Agrawal, Imielinski and Swami) був розроблений співробітниками дослідного центру IBM Almaden в 1993 році. Із цієї роботи почався інтерес до асоціативних правил; на середину 90-х років минулого століття припадає пік дослідницьких робіт у цій області, і з тих пор щороку з’являється кілька нових алгоритмів.

В алгоритмі AIS кандидати множини наборів генеруються й підраховуються «на льоту», під час сканування бази даних.

Алгоритм SETM. Створення цього алгоритму було мотивовано бажанням використовувати мову SQL для обчислення наборів товарів, що часто зустрічаються. Як і алгоритм AIS, SETM також формує кандидатів «на льоту», ґрунтуючись на перетвореннях бази даних. Щоб використовувати стандартну операцію об'єднання мови SQL для формування кандидата, SETM відокремлює формування кандидата від їхнього підрахунку.

Незручність алгоритмів AIS і SETM – надмірне генерування й підрахунок занадто багатьох кандидатів, які в результаті не є такими, що часто зустрічаються. Для поліпшення їх роботи був запропонований алгоритм Apriori.

Робота даного алгоритму складається з декількох етапів, кожний з етапів складається з наступних кроків:

- формування кандидатів;
- підрахунок кандидатів.

Формування кандидатів (candidate generation) – етап, на якому алгоритм, скануючи базу даних, створює множину і-елементних кандидатів (і – номер етапу). На цьому етапі підтримка кандидатів не розраховується.

Підрахунок кандидатів (candidate counting) – етап, на якому обчислюється підтримка кожного і-елементного кандидата. Тут же здійснюється відсікання кандидатів, підтримка яких менша мінімуму, установленого користувачем (min_sup).

Решту і-елементних наборів називаємо такими, що часто зустрічаються.

Розглянемо роботу алгоритму Apriori на прикладі бази даних D. Ілюстрація роботи алгоритму наведена на рисунку 6.3. Мінімальний рівень підтримки рівний 3.

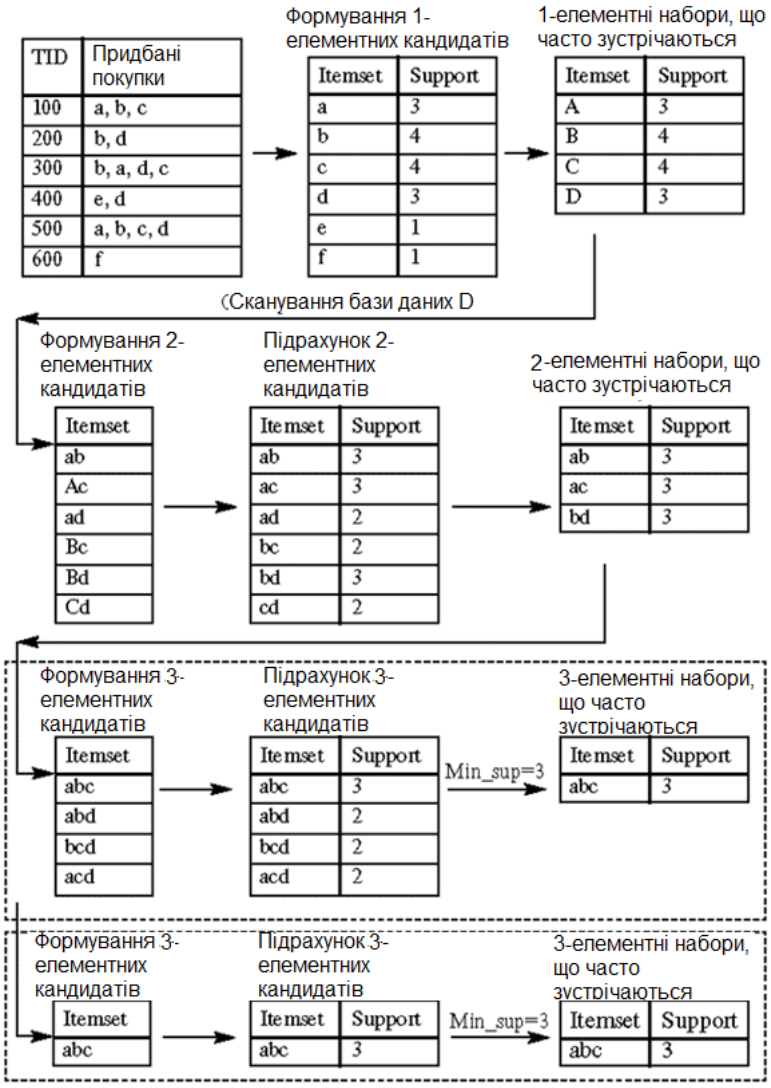



Рисунок 6.3 – Алгоритм Apriori

На першому етапі відбувається формування одноелементних кандидатів. Далі алгоритм підраховує підтримку одноелементних наборів. Набори з рівнем підтримки менше встановленого, тобто 3, відкидаються. У нашому прикладі це набори e і f, які мають підтримку, рівну 1. набори товарів, що залишилися, вважаються одноелементними наборами, що часто зустрічаються, товарів: це набори a, b, c, d.

Далі відбувається формування двоелементних кандидатів, підрахунок їх підтримки й відсікання наборів з рівнем підтримки, меншим 3. двоелементні набори товарів, що залишилися, вважаються двоелементними наборами, що часто зустрічаються, ab, ac, bd, беруть участь у подальшій роботі алгоритму.

Якщо дивитися на роботу алгоритму прямолінійно, на останньому етапі алгоритм формує трьохелементні набори товарів: abc, abd, bcd, acd, підраховує їхню підтримку й відтинає набори з рівнем підтримки, меншим 3. Набір товарів abc може бути названий таким, що часто зустрічається.

Однак алгоритм Apriori зменшує кількість кандидатів, відтинаючи – aprior – тих, які

свідомо не можуть стати такими, що часто зустрічаються, на основі інформації про відсічених кандидатів на попередніх етапах роботи алгоритму.

Відсікання кандидатів відбувається на основі припущення про те, що в набору, що часто зустрічається, товарів усі підмножини повинні бути такими, що часто зустрічаються. Якщо в наборі перебуває підмножина, яку на попередньому етапі було визначено такою, що нечасто зустрічається, цей кандидат уже не включається у формування й підрахунок кандидатів.

Так набори товарів ad, bc, cd були відкинуті як такі, що нечасто зустрічаються, алгоритм не розглядав товарів abd, bcd, acd.

При розгляді цих наборів формування трьохелементних кандидатів відбувалося б за схемою, наведеною у верхньому пунктирному прямокутнику. Оскільки алгоритм апріорі відкинув набори, що свідомо нечасто зустрічаються, останній етап алгоритму відразу визначив набір abc як єдиний трьохелементний набір, що часто зустрічається (етап наведений у нижньому пунктирному прямокутнику).

Алгоритм Apriori розраховує також підтримку наборів, які не можуть бути відсічені апріорі. Це так звана негативна область (negative border), до неї належать набори-кандидати, які зустрічаються рідко, їх самих не можна віднести до таких, що часто зустрічаються, але всі підмножини даних наборів є такими, що часто зустрічаються.

Різновиди алгоритму Apriori. Залежно від розміру найдовшого набору, що часто зустрічається, алгоритм Apriori сканує базу даних певну кількість разів. Різновиди алгоритму Apriori, що є його оптимізацією, запропоновані для скорочення кількості сканувань бази даних, кількості наборів-кандидатів або того й іншого: Aprioritid і Apriorihybrid.

Aprioritid. Цікава особливість цього алгоритму – те, що база даних D не використовується для підрахунку підтримки кандидатів набору товарів після першого проходу.

Із цією метою використовується кодування кандидатів, виконане на попередніх проходах. У наступних проходах розмір закодованих наборів може бути набагато меншим, ніж база даних, і в такий спосіб заощаджуються значні ресурси.

Apriorihybrid. Аналіз часу роботи алгоритмів Apriori і Aprioritid показує, що в більш ранніх проходах Apriori досягає більшого успіху, ніж Aprioritid; однак Aprioritid працює краще Apriori у більш пізніх проходах. Крім того, вони використовують ту саму процедуру формування наборів-кандидатів. Заснований на цьому спостереженні, алгоритм Apriorihybrid запропонований, щоб об’єднати кращі властивості алгоритмів Apriori і Aprioritid. Apriorihybrid використовує алгоритм Apriori у початкових проходах і переходить до алгоритму Aprioritid, коли очікується, що закодований набір первісної множини наприкінці проходу буде відповідати можливостям пам’яті. Однак, перемикання від Apriori до Aprioritid вимагає

залучення додаткових ресурсів.

Partition. Цей алгоритм розбивки (поділу) полягає в скануванні транзакційної бази даних шляхом поділу її на розділи, які не перетинаються, кожний з яких може вміститися в оперативній пам’яті. На першому кроці в кожному з розділів за допомогою алгоритму Apriori визначаються «локальні» набори даних, що часто зустрічаються. На другому підраховується підтримка кожного такого набору щодо всієї бази даних. Таким чином, на другому етапі визначається множина усіх потенційних наборів даних, що зустрічаються.

# ЛЕКЦІЯ 7
# МЕТОДИ КЛАСТЕРНОГО АНАЛІЗУ. ІЄРАРХІЧНІ МЕТОДИ. ІТЕРАТИВНІ МЕТОДИ

## 7.1 Кластерний аналіз

З поняттям кластеризації ми познайомилися в п’ятій темі курсу. У цій лекції ми опишемо поняття «кластер» з математичної точки зору, а також розглянемо методи розв’язку задач кластеризації – методи кластерного аналізу.

На відміну від завдань класифікації, кластерний аналіз не вимагає апріорних припущень про набір даних, не накладає обмеження на показ досліджуваних об’єктів, дозволяє аналізувати показники різних типів даних (інтервальні дані, частоти, бінарні дані). При цьому необхідно пам’ятати, що змінні повинні вимірюватися в порівнюваних шкалах.

Кластерний аналіз дозволяє скорочувати розмірність даних, робити їх наглядними та може застосовуватися до сукупностей тимчасових рядів, тут можуть виділятися періоди схожості деяких показників і визначатися групи тимчасових рядів зі схожою динамікою.

Завдання кластерного аналізу можна об’єднати в наступні групи:

- розробка типології або класифікації;
- дослідження корисних концептуальних схем групування об’єктів;
- представлення гіпотез на основі дослідження даних;
- перевірка гіпотез або досліджень для визначення, чи дійсно типи (групи), виділені тем або іншим способом, присутні в наявних даних.

Як правило, при практичному використанні кластерного аналізу одночасно вирішується декілька із зазначених задач.

Розглянемо приклад процедури кластерного аналізу. Допустимо, ми маємо набір даних A, що полягає з 14-ти прикладів, у яких є по дві ознаки X і Y. Дані в табличній формі не носять інформативний характер. Представимо змінні X і Y у вигляді діаграми розсіювання, зображеної на рисунку 7.1.

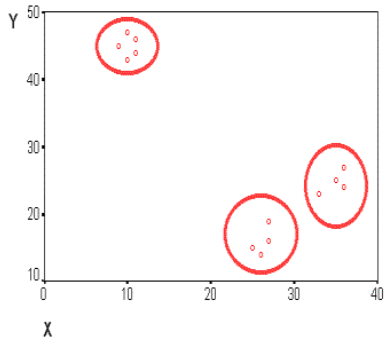



Рисунок 7.1 – Діаграма розсіювання змінних X і Y

На малюнку ми бачимо кілька груп «схожих» прикладів. Приклади (об’єкти), які за значеннями X і Y «схожі» один на одного, належать до одній групі (кластеру); об’єкти з різних кластерів не схожі один на одного.

Критерієм для визначення схожості й відмінності кластерів є відстань між точками на діаграмі розсіювання. Ця подібність можна «виміряти», воно дорівнює відстані між точками на графіку. Найпоширеніший спосіб визначення відстані – обчислення евклідової відстані між двома точками і і j на площині, коли відомі їхні координати X і Y, щоб довідатися відстань між двома точками, треба взяти різницю їх координат по кожній осі, звести її у квадрат, скласти отримані значення для всіх осей і витягти квадратний корінь із суми.

Коли осей більше, чим дві, відстань розраховується в такий спосіб: сума квадратів різниці координат складається зі стількох доданків, скільки осей (вимірів) є присутнім у нашому просторі.

Кластер має наступні математичні характеристики: центр, радіус, середньоквадратичне відхилення, розмір кластера.

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

Радіус кластера – максимальна відстань точок від центру кластера.

Кластери можуть бути, такими що перекриваються, така ситуація виникає, коли виявляється перекриття кластерів. У цьому випадку неможливо за допомогою математичних процедур однозначно віднести об’єкт до одному із двох кластерів. Такі об’єкти називають спірними.

Спірний об’єкт – це об’єкт, який у міру подібності може бути віднесений до декільком кластерам.

Розмір кластера може бути визначений або по радіусу кластера, або по середньоквадратичному відхиленню об’єктів для цього кластера. Об’єкт ставиться до кластера, якщо відстань від об’єкта до центру кластера менше радіуса кластера. Якщо ця умова виконується для двох і більш кластерів, об’єкт є спірним.

Неоднозначність даного завдання може бути усунута експертом або аналітиком.

Робота кластерного аналізу опирається на два припущення. Перше припущення – розглянуті ознаки об’єкта в принципі допускають бажане розбиття сукупності об’єктів на кластери. Друге припущення – правильність вибору масштабу або одиниці вимірювання ознак.

Вибір масштабу в кластерному аналізі має велике значення. Розглянемо приклад. Уявимо собі, що дані ознаки x у наборі даних A на два порядки більші даних ознаки y: значення змінної x перебувають в діапазоні від 100 до 700, а значення змінної y – у діапазоні від 0 до 1.

Тоді, при розрахунках величини відстані між точками, що відображають положення

об’єктів у просторі їх властивостей, змінна, що має більші значення, тобто змінна x, буде практично повністю домінувати над змінною з малими значеннями, тобто змінної y. У такий спосіб через неоднорідність одиниць виміру ознак стає неможливо коректно розрахувати відстані між точками.

Ця проблема вирішується за допомогою попередньої стандартизації змінних. Стандартизація (standardization) або нормування (normalization) приводить значення всіх перетворених змінних до єдиного діапазону значень шляхом вираження через відношення цих значень до якоїсь величини, що відображає певні властивості конкретної ознаки. Існують різні способи нормування вихідних даних.

Два найпоширеніші способи:

- розподіл вихідних даних на середньоквадратичне відхилення відповідних змінних;
- обчислення Z-внеску або стандартизованого внеску.

Поряд зі стандартизацією змінних, існує варіант додавання до кожної з них певного коефіцієнта важливості, або ваги, який би відображав значимість відповідної змінної. У якості ваг можуть виступати експертні оцінки, отримані в ході опитування експертів – фахівців предметної області. Отримані добутки нормованих змінних на відповідні ваги дозволяють одержувати відстані між точками в багатомірному просторі з урахуванням неоднакової ваги змінних.

У ході експериментів можливе порівняння результатів, отриманих з урахуванням експертних оцінок і без них, і вибір кращої з них.

### 7.2 Методи кластерного аналізу

Методи кластерного аналізу можна розділити на дві групи: ієрархічні та неієрархічні. Кожна із груп включає безліч підходів і алгоритмів. Використовуючи різні методи кластерного аналізу, аналітик може одержати різні розв’язки для тих самих даних. Це вважається нормальним явищем.

Ієрархічні методи кластерного аналізу. Суть ієрархічної кластеризації полягає в послідовному об’єднанні менших кластерів у більші або поділі більших кластерів на менші.

Ієрархічні агломеративні методи (Agglomerative Nesting, AGNES). Ця група методів характеризується послідовним об’єднанням вихідних елементів і відповідним зменшенням числа кластерів.

На початку роботи алгоритму всі об’єкти є окремими кластерами. На першому кроці найбільш схожі об’єкти поєднуються в кластер. На наступних кроках об’єднання триває доти, поки всі об’єкти не будуть становити один кластер.

Ієрархічні дивизимні (ділені) методи (Divisive Analysis, DIANA). Ці методи є логічною

протилежністю агломеративним методам. На початку роботи алгоритму всі об’єкти належать одному кластеру, який на наступних кроках ділиться на менші кластери, у результаті утворюється послідовність груп, що розщеплюються.

Програмна реалізація алгоритмів кластерного аналізу широко представлена в різних інструментах Data Mining, які дозволяють вирішувати завдання досить великої розмірності. Наприклад, агломеративні методи реалізовані в пакеті SPSS, дивизимні методи – у пакеті Statgraf.

Ієрархічні методи кластеризації різняться правилами побудови кластерів. У якості правил виступають критерії, які використовуються при розв’язку питання про «схожість» об’єктів при їхнім об’єднанні в групу (агломеративні методи) або поділу на групи (дивизимні методи). Дані методи використовуються при невеликих обсягах наборів даних. Перевагою ієрархічних методів кластеризації є їхню наочність.

Ієрархічні алгоритми пов’язані з побудовою дендрограм (від грецького dendron – «дерево»), які є результатом ієрархічного кластерного аналізу [10].

Дендрограмма (dendrogram) – деревоподібна діаграма, що містить n рівнів, кожний з яких відповідає одному із кроків процесу послідовного укрупнення кластерів та описує близькість окремих точок і кластерів друг до друга, представляє в графічному виді послідовність об’єднання (поділу) кластерів

Міри подібності. Для обчислення відстані між об’єктами використовуються різні міри подібності, їх називають також метриками або функціями відстаней.

Квадрат евклідової відстані. Для надання більшої ваги більш віддаленим один від одного об’єктів можемо скористатися квадратом евклідової відстані шляхом піднесення у квадрат стандартної евклідової відстані.

Манхеттенська відстань (відстань міських кварталів), також називається «хемінговою» або «сіті-блок» відстанню. Ця відстань розраховується як середня різниць по координатах. У більшості випадків ця міра відстані приводить до результатів, подібних розрахунків відстані евкліда. Однак, для цієї міри вплив окремих викидів менший, ніж при використанні евклидової відстані, оскільки тут координати не підносяться до квадру.

Методи об’єднання або зв’язки. Коли кожний об’єкт становить окремий кластер, відстані між цими об’єктами визначаються обраною мірою. Виникає наступне питання – як визначити відстані між кластерами? Існують різні правила, називані методами об’єднання або зв’язками для двох кластерів.

Метод найближчого сусіда або одиночний зв’язок. Тут відстань між двома кластерами визначається відстанню між двома найбільш близькими об’єктами (найближчими сусідами) у різних кластерах. Цей метод дозволяє виділяти кластери як завгодно складної форми за умови, що різні частини таких кластерів з’єднані ланцюжками близьких один до одного елементів. У

результаті роботи цього методу кластери представляються довгими «ланцюжками» або «волокнистими» кластерами, «зчепленими разом» тільки окремими елементами, які випадково виявилися ближче інших один до одного.

Метод найбільш віддалених сусідів або повний зв'язок. Тут відстані між кластерами визначаються найбільшою відстанню між будь-якими двома об'єктами в різних кластерах ( тобто «найбільш віддаленими сусідами»). Метод добре використовувати, коли об'єкти дійсно походять із різних «ділянок». Якщо ж кластери мають до певної міри подовжену форму або їх природній тип є «ланцюговим», то цей метод не слід використовувати.

Метод Варда (Ward's method). У якості відстані між кластерами береться приріст суми квадратів відстаней об'єктів до центрів кластерів, одержуваний у результаті їх об'єднання (Ward, 1963). На відміну від інших методів кластерного аналізу для оцінки відстаней між кластерами, тут використовуються методи дисперсійного аналізу. На кожному кроці алгоритму поєднуються такі два кластери, які приводять до мінімального збільшення цільової функції, тобто внутрішньо групової суми квадратів.

Цей метод спрямований на об'єднання близько розташованих кластерів і «прагне» створювати кластери малого розміру.

Метод незваженого попарного середнього (метод незваженого попарного арифметичного середнього – unweighted pair-group method using arithmetic averages, UPGMA (Sneath, Sokal, 1973)).

У якості відстані між двома кластерами береться середня відстань між усіма парами об'єктів у них. Цей метод слід використовувати, якщо об'єкти дійсно походять із різних «ділянок», у випадках присутності кластерів «ланцюгового» типу, при припущенні нерівних розмірів кластерів.

Метод зваженого попарного середнього (метод зваженого попарного арифметичного середнього – weighted pair-group method using arithmetic averages, WPGM A (Sneath, Sokal, 1973)). Цей метод схожий на метод незваженого попарного середнього, різниця полягає лише в тому, що тут у якості вагового коефіцієнта використовується розмір кластера (число об'єктів, що втримуються в кластері).

Цей метод рекомендується використовувати саме при наявності припущення про кластери різних розмірів.

### 7.3 Алгоритми неієрархічної кластеризації

При великій кількості спостережень ієрархічні методи кластерного аналізу непридатні.

У таких випадках використовують неієрархічні методи, засновані на поділі, які являють собою ітеративні методи дроблення вихідної сукупності. У процесі розподілу нові кластери

формуються доти, поки не буде виконане правило зупинки.

Така неієрархічна кластеризація полягає в поділі набору даних на певну кількість окремих кластерів. Існує два підходи. Перший полягає у визначенні границь кластерів як найбільш щільних ділянок у багатомірному просторі вихідних даних, тобто визначення кластера там, де є велике «згущення точок». Другий підхід полягає в мінімізації міри відмінності об’єктів.

Найпоширеніший серед неієрархічних методів алгоритм k-середніх, також називають швидким кластерним аналізом. На відміну від ієрархічних методів, які не вимагають попередніх припущень щодо числа кластерів, для можливості використання цього методу необхідно мати гіпотезу про найбільш імовірну кількість кластерів.

Алгоритм k-середніх будує k кластерів, розташованих на максимально можливо великих відстанях один від одного. Основний тип задач, які вирішує алгоритм k-середніх, – наявність припущень (гіпотез) щодо числа кластерів, при цьому вони повинні бути різні настільки, наскільки це можливо. Вибір числа k може базуватися на результатах попередніх досліджень, теоретичних міркуваннях або інтуїції.

Опис алгоритму. Вибирається число k, і на першому кроці ці точки вважаються «центрами» кластерів. Кожному кластеру відповідає один центр. Вибір початкових центроїдів може здійснюватися в такий спосіб:

- вибір k-спостережень для максимізації початкової відстані;
- випадковий вибір k-спостережень;
- вибір перших k-спостережень.

У результаті кожний об’єкт призначений певному кластеру. Обчислюються центри кластерів, якими потім і далі вважаються покоординатні середні кластерів. Об’єкти знову перерозподіляються. Процес обчислення центрів і перерозподілу об’єктів триває доти, поки не виконана одна з умов:

- кластерні центри стабілізувалися, тобто всі спостереження належать кластеру, якому належали до поточної ітерації;
- число ітерацій дорівнює максимальному числу ітерацій.

Після одержання результатів кластерного аналізу методом k-середніх слід перевірити правильність кластеризації (тобто оцінити, наскільки кластери відрізняються один від одного). Для цього розраховуються середні значення для кожного кластера. При гарній кластеризації повинні бути отримані дуже відмінні середні для всіх вимірів або хоча б більшої їхньої частини.

Переваги алгоритму k-середніх:

- простота використання;
- швидкість використання;

- зрозумілість і прозорість алгоритму.

Недоліки алгоритму k-середніх:

- алгоритм занадто чутливий до викидів, які можуть спотворювати середнє. Можливим вирішенням цієї проблеми є використання модифікації алгоритму – алгоритм k-медіани;
- алгоритм може повільно працювати на великих базах даних. Можливим вирішенням даної проблеми є використання вибірки даних.

Алгоритм PAM (partitioning around Medoids) є модифікацією алгоритму k-середніх, алгоритмом k-медіани (k-medoids).

Алгоритм менш чутливий до шумів і викидів даних, ніж алгоритм k-means, оскільки медіана менше піддана впливам викидів. PAM ефективний для невеликих баз даних, але його не слід використовувати для великих наборів даних.

Розглянемо приклад. Є база даних клієнтів фірми, яких слід розбити на однорідні групи. Кожний клієнт описується за допомогою 25 змінних. Використання такого великого числа змінних приводить до виділення кластерів нечіткої структури. У результаті аналітикові досить складно інтерпретувати отримані кластери.

Більш зрозумілі й прозорі результати кластеризації можуть бути отримані, якщо замість множини вихідних змінних використовувати якісь узагальнені змінні або критерії, що містять у стислому вигляді інформацію про зв’язки між змінними. Тобто виникає задача зниження розмірності даних. Вона може вирішуватися за допомогою різних методів; один з найпоширеніших – факторний аналіз.

### 7.4 Факторний аналіз

Факторний аналіз – це метод, який застосовується для вивчення взаємозв’язків між значеннями змінних. Його основні цілі – скорочення кількості змінних та їх класифікація. Таким чином, факторний аналіз використовується як для зменшення розмірності даних, так і для вирішення задач класифікації.

У результаті факторного аналізу виділяються головні фактори (або критерії), які у стислій формі містять інформацію про існуючі зв’язки між змінними. Це дозволяє: покращити якість кластеризації, краще інтерпретувати семантику кластерів, надати факторам осмислене значення.

Суть методу полягає в тому, що велика кількість змінних зводиться до меншого числа незалежних впливаючих величин, які називаються факторами. Один фактор може об’єднувати кілька змінних, що мають високу кореляцію між собою. Таким чином, фактори виступають як комплексні характеристики, які максимально пояснюють внутрішні залежності між змінними.

На першому етапі факторного аналізу проводиться стандартизація значень змінних, що

необхідно для усунення впливу різних масштабів вимірювання.

Факторний аналіз базується на гіпотезі, що спостережувані змінні є непрямими проявами меншої кількості схованих (латентних) факторів, які і формують реальну структуру даних.

Одним із популярних методів факторного аналізу є метод головних компонентів (PCA), який передбачає, що всі фактори є незалежними один від одного.

### 7.5 Порівняльний аналіз ієрархічних і неієрархічних методів кластеризації

Перед проведенням кластерного аналізу аналітик стикається з вибором між ієрархічними та неієрархічними методами кластеризації. Вибір методу залежить від особливостей даних, цілей дослідження та наявних обчислювальних ресурсів.

Неієрархічні методи демонструють вищу стійкість до шумів, викидів, некоректного вибору метрики та включення несуттєвих змінних до набору даних. Однак за ці переваги доводиться «платити» потребою у заданих наперед параметрах, таких як кількість кластерів, кількість ітерацій або критерій зупинки. Це може бути складним завданням, особливо для початківців.

У разі відсутності припущень щодо кількості кластерів доцільно використовувати ієрархічні алгоритми, які не вимагають попереднього визначення їхньої кількості, а будують повне дерево вкладених кластерів (дендрограму). Якщо ж обсяг вибірки занадто великий, доцільно провести серію експериментів із поступовим збільшенням кількості кластерів (наприклад, почати з двох і поступово збільшувати), порівнюючи результати. Це забезпечує гнучкість аналізу.

Разом із тим ієрархічні методи мають низку обмежень:

- не придатні для великих обсягів даних;
- залежні від вибору міри близькості;
- менш гнучкі щодо зміни отриманої класифікації.

Проте їх перевагами є наочність результатів і можливість глибшого аналізу структури даних. Також вони дозволяють ефективно ідентифікувати викиди, що може підвищити якість даних і поліпшити результати наступного кластерного аналізу. На цій властивості базується двоетапна кластеризація, де результати ієрархічного аналізу використовуються для подальшої неієрархічної кластеризації.

Ще один важливий аспект – кластеризація всієї сукупності даних або лише вибірки. Це особливо актуально для ієрархічних методів, які не здатні обробляти великі обсяги даних. В такому випадку аналіз вибірки може стати практичним компромісом.

Слід також враховувати, що результати кластеризації не завжди мають чітке

статистичне обґрунтування. Утім, у кластерному аналізі допускається нестатистична інтерпретація результатів, що надає аналітику гнучкість у пошуку задовільного рішення.

Нові алгоритми та модифікації методів кластерного аналізу

Класичні методи кластеризації, розглянуті раніше, довгий час оцінювались за критерієм якості кластеризації за умови, що весь обсяг даних розміщується в оперативній пам’яті. З появою надвеликих баз даних актуальним стало питання масштабованості алгоритмів. Також сучасні вимоги включають:

- незалежність результатів від порядку вхідних даних;
- стабільність до параметрів, що залежать від структури даних.

Сьогодні активно розробляються нові масштабовані алгоритми, здатні обробляти великі обсяги даних.

BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) – алгоритм базується на узагальненому представленні кластерів, що забезпечує високу швидкість і масштабованість. Кластеризація виконується у два етапи: спочатку створюється попередній набір кластерів, а потім до нього застосовуються інші методи кластеризації, зручні для обробки в оперативній пам’яті.

WaveCluster – алгоритм на основі хвильового перетворення. Спочатку простір розбивається на багатовимірну ґратку, потім аналізуються узагальнені характеристики даних у межах ґрат.

CLARA (Clustering Large Applications) – алгоритм, запропонований Kaufmann і Rousseeuw у 1990 році, реалізований у статистичних пакетах (наприклад, S+). Він ґрунтується на вибірці: з бази даних витягуються підмножини, кожна з яких кластеризується, і обирається найкращий варіант. Це ефективніше за PAM для великих наборів, однак якість результату залежить від вибірки.

Clarans, CURE, DBSCAN – формалізують задачу кластеризації як пошук у графі. Кластери формуються на основі вузлів графа, які мінімізують деяку критеріальну функцію.

Основним обмеженням цих алгоритмів є потреба у завданнях щільнісних порогів, що не завжди можливо через складність попереднього аналізу.

На сьогодні масштабовані методи кластеризації активно вдосконалюються. Метою є подолання обмежень поточних алгоритмів, пов’язаних із залежністю від вхідних параметрів і складністю роботи з великими обсягами даних.

# ЛЕКЦІЯ 8
# КОМПЛЕКСНИЙ ПІДХІД ДО ІАД

## 8.1 Традиційний процес Data Mining

Процес Data Mining є свого роду дослідженням. Як будь-яке дослідження, цей процес складається з певних етапів, що включають елементи порівняння, типізації, класифікації, узагальнення, абстрагування, повторення.

Процес Data Mining нерозривно пов'язаний із процесом прийняття розв'язків, він будує модель, а в процесі прийняття розв'язків ця модель експлуатується.

Традиційний процес Data Mining включає наступні етапи: аналіз предметної області, постановка завдання, підготовка даних, побудова моделей, перевірка й оцінка моделей, вибір моделі, застосування моделі, корекція й відновлення моделі.

Етап 1. Аналіз предметної області.

Дослідження – це процес пізнання певної предметної області, об'єкта або явища з певною метою. Процес дослідження укладається в спостереженні властивостей об'єктів з метою виявлення й оцінки важливих, з погляду суб'єкта-дослідника, закономірних відносин між показниками даних властивостей.

Розв'язок будь-якого завдання в сфері розробки програмного забезпечення повинне починатися з вивчення предметної області. Предметна область – це подумки обмежена область реальної дійсності, що підлягає опису або моделюванню й дослідженню. Предметна область складається з об'єктів, що різняться по властивостях, що й перебувають у певних відносинах між собою або взаємодіючих яким-небудь образом.

Дослідникові необхідно вміти виділити істотну їхню частину. Наприклад, при розв'язку завдання «чи видавати кредит?» важливими є всі дані про приватне життя клієнта, аж до того, чи має роботу подружжя, є чи в клієнта неповнолітні діти, який рівень його утвору і т.д. Для розв'язку іншого завдання банківської діяльності ці дані будуть абсолютно неважливі. Істотність даних, таким чином, залежить від вибору предметної області.

У процесі вивчення предметної області повинна бути створена її модель. Знання з різних джерел повинні бути формалізовані за допомогою яких-небудь коштів.

Це можуть бути текстові описи предметної області або спеціалізовані графічні нотації. Існує велика кількість методик опису предметної області, наприклад, методика структурного аналізу SADT і заснована на ньому IDEF0, діаграми потоків даних Гейна-Сарсона, методика об'єктно-орієнтованого аналізу UML і інші. Модель предметної області описує процеси, що відбуваються в предметній області, і дані, які в цих процесах використовуються [7].

Етап 2. Постановка завдання.

Постановка завдання Data Mining включає наступні кроки: формулювання завдання та

формалізація завдання. Постановка завдання включає також опис статичної й динамічної поведінки досліджуваних об’єктів.

Етап 3. Підготовка даних.

Підготовка даних є найважливішим етапом, від якості виконання якого залежить можливість одержання якісних результатів усього процесу Data Mining. Крім того, слід пам’ятати, що на етап підготовки даних, за деякими оцінками, може бути витрачено до 80% усього часу, відведеного на проект.

Визначення й аналіз вимог до даних. На цьому етапі здійснюється так зване моделювання даних, тобто визначення й аналіз вимог до даних, які необхідні для здійснення Data Mining. При цьому вивчаються питання:

- розподілу користувачів;
- доступу до даних;
- аналітичних характеристик системи.

Збір даних. Наявність в організації сховища даних робить аналіз простіше й ефективніше, його використання, з погляду вкладень, обходиться дешевше, чим використання окремих баз даних або вітрин даних. Однак далеко не всі підприємства оснащені сховищами даних. У цьому випадку джерелом для вихідних даних є оперативні, довідкові й архівні БД, тобто дані з існуючих інформаційних систем.

Також для Data Mining може знадобитися інформація з інформаційних систем керівників, зовнішніх джерел, паперових носіїв, а також знання експертів або результати опитувань.

Слід пам’ятати, що в процесі підготовки даних аналітики й розроблювачі не повинні прив’язуватися до показників, які є в наявності, і описати максимальна кількість факторів і ознак, що впливають на аналізований процес.

На цьому етапі здійснюється кодування деяких даних. Допустимо, одним з атрибутів клієнта є рівень доходу, який повинен бути представлено в системі одним зі значень: дуже низьким, низьким, середнім, високим, дуже високим.

Необхідно визначити градації рівня доходу, у цьому процесі буде потрібно співробітництво аналітика з експертом у предметній області. Можливо, для таких перетворень даних буде потрібно написання спеціальних процедур.

Визначення необхідної кількості даних. При визначенні необхідної кількості даних слід ураховувати, чи є дані впорядкованими чи ні.

Якщо дані впорядковані й ми маємо справу з тимчасовими рядами, бажане знати, чи включає такий набір даних сезонну/циклічну компоненту. У випадку присутності у наборі даних сезонної/циклічної компоненти, необхідно мати дані як мінімум за один сезон/цикл.

Якщо дані не впорядковані, тобто події з набору даних не зв’язані за часом, у ході збору

даних слід дотримуватися наступні правил.

Кількість записів у наборі. Недостатня кількість записів у наборі даних може стати причиною побудови некоректної моделі. З погляду статистики, точність моделі збільшується зі збільшенням кількості досліджуваних даних. Можливо, деякі дані є застарілими або описують якусь нетипову ситуацію, і їх потрібно виключити з бази даних. Алгоритми, використовувані для побудови моделей на надвеликих базах даних, повинні бути масштабованими.

Співвідношення кількості записів у наборі й кількості вхідних змінних. При використанні багатьох алгоритмів необхідно певне (бажане) співвідношення вхідних змінних і кількості спостережень. Кількість записів (прикладів) у наборі даних повинне бути значно більше кількості факторів (змінних).

Набір даних повинен бути репрезентативним і представляти якнайбільше можливих ситуацій. Пропорції вистави різних прикладів у наборі даних повинні відповідати реальній ситуації.

Попередня обробка даних. Аналізувати можна як якісні, так і неякісні дані. Результат буде досягнутий і в тому, і в іншому випадку. Для забезпечення якісного аналізу необхідне проведення попередньої обробки даних, яка є необхідним етапом процесу Data Mining.

Оцінювання якості даних. Дані, отримані в результаті збору, повинні відповідати певним критеріям якості. Таким чином, можна виділити важливий підетап процесу Data Mining – оцінювання якості даних.

Якість даних (Data quality) – це критерій, що визначає повноту, точність, своєчасність і можливість інтерпретації даних.

Дані можуть бути високої якості й низької якості, останні – це так звані брудні або «погані» дані. Дані високої якості – це повні, точні, своєчасні дані, які піддаються інтерпретації. Такі дані забезпечують одержання якісного результату: знань, які зможуть підтримувати процес прийняття розв’язків.

Прогноз. Багато компаній стали обертати більше уваги на якість даних, оскільки низька якість веде до зниження продуктивності, прийняттю неправильних бізнес-рішень і неможливості одержати бажаний результат.

Реальність. Дана тенденція зберігається, особливо в індустрії фінансових послуг. У першу чергу це ставиться до фірм, що намагаються виконувати угода Basel II. Неякісні дані не можуть використовуватися в системах оцінки ризиків, які застосовуються для установки цін на кредити й обчислення потреб організації в капіталі. Цікаво відзначити, що суттєво змінилися погляди на способи розв’язку проблеми якості даних. Спочатку менеджери звертали основну увагу на інструменти оцінки якості, уважаючи, що «власник» даних повинен вирішувати проблему на рівні джерела, наприклад, очищаючи дані й перенавчаючи

співробітників. Але зараз їх погляди суттєво змінилися. Поняття якості даних набагато ширше, чим просто їх акуратне введення в систему на першому етапі.

Розглянемо поняття якості даних більш детально. Дані низької якості, або брудні дані – це відсутні, неточні дані з погляду практичного застосування (наприклад, представлені в невірному форматі, не відповідному до стандарту). Брудні дані з'явилися не сьогодні, вони виникли одночасно із системами введення даних.

Брудні дані можуть з'явитися по різних причинах, таким як помилка при введенні даних, використання інших форматів вистави або одиниць виміру, невідповідність стандартам, відсутність своєчасного відновлення, невдале відновлення всіх копій даних, невдале видалення записів-дублікатів і т.д.

Необхідно оцінити вартість наявності брудних даних; інакше кажучи, наявність брудних даних може дійсно привести до фінансових втрат і юридичної відповідальності, якщо їх присутність не запобігає або вони не виявляються й не очищаються.

Розглянемо найпоширеніші види брудних даних:

- пропущені значення;
- дублікати даних;
- шуми й викиди.

Пропущені значення (Missing Values). Деякі значення даних можуть бути пропущені у зв'язку з тим, що:

- дані взагалі не були зібрані (наприклад, при анкетуванні схований вік);
- деякі атрибути можуть бути незастосовні для деяких об'єктів (наприклад, атрибут «річний дохід» не застосуємо до дитини).

Як можна працювати з пропущеними даними?

У процесі обробки даних часто виникає необхідність вирішити, як поводитися з відсутніми (пропущеними) значеннями. Існує кілька основних підходів:

- виключення об'єктів із пропущеними значеннями;
- обчислення нових значень для заповнення пропусків;
- ігнорування пропущених значень під час аналізу;
- заміна пропущених значень на можливі (правдоподібні) значення.

### 8.2 Дублювання даних

Набір даних може включати продубльовані дані, тобто дублікати.

Дублікатами називаються записи з однаковими значеннями всіх атрибутів.

Наявність дублікатів у наборі даних може бути способом підвищення значимості деяких записів. Така необхідність іноді виникає для особливого виділення певних записів з

набору даних. Однак у більшості випадків, продубльовані дані є результатом помилок при підготовці даних.

Існує два варіанти обробки дублікатів. При першому варіанті віддаляється вся група записів, що містить дублікати. Цей варіант використовується в тому випадку, якщо наявність дублікатів викликає недовіра до інформації, повністю її знецінює.

Другий варіант полягає в заміні групи дублікатів на один унікальний запис. Шуми й викиди.

Викиди – різко одмінні об’єкти або спостереження в наборі даних.

Шуми й викиди є досить загальною проблемою в аналізі даних. Викиди можуть як являти собою окремі спостереження, так і бути об’єднаними в якісь групи. Завдання аналітика – не тільки їх виявити, але й оцінити ступінь їх впливу на результати подальшого аналізу.

Досить поширена практика проведення двоетапного аналізу – з викидами й з їхньою відсутністю – і порівняння отриманих результатів.

Різні методи Data Mining мають різну чутливість до викидів, цей факт необхідно враховувати при виборі методу аналізу даних. Також деякі інструменти Data Mining мають вбудовані процедури очищення від шумів і викидів.

Візуалізація даних дозволяє представити дані, у тому числі й викиди, у графічному виді. Приклад наявності викидів зображений на діаграмі розсіювання на рисунку 8.1.

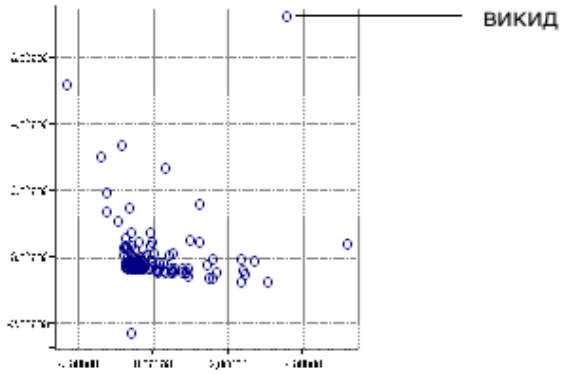



Рисунок 8.1 – Приклад набору даних з викидами

Очевидно, що результати Data Mining на основі брудних даних не можуть вважатися надійними й корисними. Однак наявність таких даних не обов’язково означає необхідність їх очищення або ж запобігання появи. Завжди повинен бути розумний вибір між наявністю брудних даних і вартістю й/або часом, необхідним для їхнього очищення.

### 8.3 Очищення даних

Очищення даних (data cleaning, data cleansing або scrubbing) займається виявленням і видаленням помилок і невідповідностей у даних з метою поліпшення якості даних.

Проблеми з якістю зустрічаються в окремих наборах даних – таких як файли й бази даних. Коли інтеграції підлягає безліч джерел даних, необхідність в очищенні даних суттєво зростає. Це відбувається тому, що джерела часто містять розрізнені дані в різній виставі. Для забезпечення доступу до точних і погоджених даних необхідна консолідація різних вистав даних і виключення інформації, що дублюється. Спеціальні кошти очищення звичайно мають справа з конкретними областями – в основному це імена й адреси – або ж з виключенням дублікатів.

Перетворення забезпечуються або у формі бібліотеки правил, або користувачем в інтерактивному режимі. Перетворення даних можуть бути автоматично отримані за допомогою коштів узгодження схеми.

Метод очищення даних має відповідати ряду важливих вимог:

- виявлення та усунення помилок;
- підтримка інструментами та мінімізація ручної праці;
- інтеграція з перетворенням даних;
- декларативне визначення функцій мапінгу;
- масштабована та надійна інфраструктура.

На сьогоднішній день інтерес до очищення даних зростає. Цілий ряд дослідницьких груп займається загальними проблемами, пов’язаними з очищенням даних, у тому числі, зі специфічними підходами до Data Mining і перетворенню даних на підставі зіставлення схеми.

### 8.4 Етапи очищення даних

Етапи очищення даних. У цілому, очищення даних включає наступні етапи: аналіз даних, визначення порядку й правил перетворення даних, підтвердження, перетворення, зворотний потік очищених даних (завантаження/інтеграція).

Етап № 1. Аналіз даних. Докладний аналіз даних необхідний для виявлення підлягаючих видаленню видів помилок і невідповідностей. Тут можна використовувати як ручну перевірку даних або їх шаблонів, так і спеціальні програми для одержання метаданих про властивості даних і визначення проблем якості.

Етап № 2. Визначення порядку й правил перетворення даних. Залежно від числа джерел даних, ступені їх неоднорідності й забруднення, дані можуть вимагати досить великого перетворення й очищення. Іноді для відображення джерел загальної моделі даних

використовується трансляція схеми; для сховищ даних звичайно використовується реляційна вистава. Перші кроки по очищенню можуть уточнити або змінити опис проблем окремих джерел даних, а також підготувати дані для інтеграції. Подальші кроки повинні бути спрямовані на інтеграцію схеми/даних і усунення проблем множинних елементів, наприклад, дублікатів. Для сховищ у процесі роботи з визначення ETL повинні бути визначені методи контролю й потік даних, що підлягає перетворенню й очищенню.

Перетворення даних, зв’язані зі схемою, так само як і етапи очищення, повинні, наскільки можливо, визначатися за допомогою декларативного запиту й мови маппірования, забезпечуючи, таким чином, автоматичну генерацію коду перетворення. До того ж, у процесі перетворення повинна існувати можливість запуску написаного користувачем коду очищення й спеціальних коштів. Етапи перетворення можуть вимагати зворотного зв’язку з користувачем по тем елементам даних, для яких відсутня вбудована логіка очищення [7].

Етап № 3. Підтвердження. На цьому етапі визначається правильність і ефективність процесу й визначень перетворення. Це здійснюється шляхом тестування й оцінювання, наприклад, на прикладі або на копії даних джерела, – щоб з’ясувати, чи необхідно якось поліпшити ці визначення. При аналізі, проектуванні й підтвердженні може знадобитися безліч ітерацій, наприклад, у зв’язку з тим, що деякі помилки стають помітні тільки після проведення певних перетворень.

Етап № 4. Перетворення. На цьому етапі здійснюється виконання перетворень або в процесі ETL для завантаження й відновлення Сховища даних, або при відповіді на запити по безлічі джерел.

Етап № 5. Зворотний потік очищених даних (завантаження/інтеграція). Після того як помилки окремого джерела вилучені, забруднені дані у вихідних джерелах повинні замінитися на очищені, для того щоб поліпшені дані потрапили також в успадковані додатки й надалі при витягу не вимагали додаткового очищення.

Такий процес перетворення вимагає більших обсягів метаданих (схем, характеристик даних рівня схеми, визначень технологічного процесу й ін.). Для погодженості, гнучкості й спрощення використання в інших випадках, ці метадані повинні зберігатися в депозиторії на основі СУБД. Для підтримки якості даних докладна інформація про процес перетворення повинна записуватися як у депозитарій, так і в трансформовані елементи даних, особливо інформація про повноту й свіжості вихідних даних і походження інформації про першоджерело трансформованих об’єктів і зроблених з ними змінах.

# ЛЕКЦІЯ 9
# СХОВИЩА ДАНИХ ТА OLAP-ТЕХНОЛОГІЇ

## 9.1 Концепція сховищ даних

На початку восьмидесятих років минулого століття, в період бурхливого розвитку реєструючих інформаційних систем, виникло розуміння обмеженості можливостей їх застосування для аналізу даних і побудови на їх основі систем інтелектуального аналізу даних. Реєструючи системи створювалися для автоматизації рутинних операцій по веденню бізнесу – виписки рахунків, оформлення договорів, перевірки стану складу і т. п. Основними вимогами до таких систем були забезпечення транзакційності змін, що вносилися, і максимізація швидкості їх виконання. Саме ці вимоги визначили вибір реляційних СУБД і моделі представлення даних «суть-зв'язок» в якості основного технічного рішення при побудові реєструючих систем.

Для менеджерів і аналітиків у свою чергу були потрібні системи, які б дозволяли:

- аналізувати інформацію в часовому аспекті;
- формувати довільні запити до системи; обробляти великі об'єми даних;
- інтегрувати дані з різних реєструючих систем.

Очевидно, що регіструючи системи не задовольняли жодному з вищезгаданих вимог. У реєструючі системі інформація актуальна тільки на момент звернення до бази даних, в наступний момент часу по тому ж запиту можна отримати абсолютно інший результат. Інтерфейс реєструючі систем розрахований на проведення жорстко певних операцій і можливості отримання результатів на нерегламентований (ad-hoc) запит сильно обмежені. Можливість обробки великих масивів даних також мала через налаштування СУБД на виконання коротких транзакцій і неминучого уповільнення роботи решти користувачів. Відповіддю на виниклу потребу стала поява нової технології організації баз даних – технології сховищ даних (Data Warehouse) [11].

За спостереженнями дослідницької компанії Forrester Research, більшість великих компаній стикаються з наступною проблемою: вони накопичують величезну кількість інформації, яка ніколи не використовується. Практично в будь-якій організації реально функціонує безліч транзакційних систем, орієнтованих на оперативну обробку даних (кожна для конкретного класу задач) і безперервно поповнюють численні бази даних. Окрім цього, часто підприємства володіють величезними об'ємами інформації, що зберігається в так званих успадкованих системах. Всі ці дані розподілені по мережах персональних комп'ютерів, зберігаються на мейнфреймах, робочих станціях і серверах. Таким чином, інформація є, але вона розосереджена, неузгоджена, неструктурована, часто надмірна і не завжди достовірна. Тому в більшості організацій ці дані до цих пір не можуть бути використані для ухвалення

критичних бізнес-рішень. На вирішення цього протиріччя і направлена концепція сховищ даних (Data Warehouse) (рис. 9.1.).

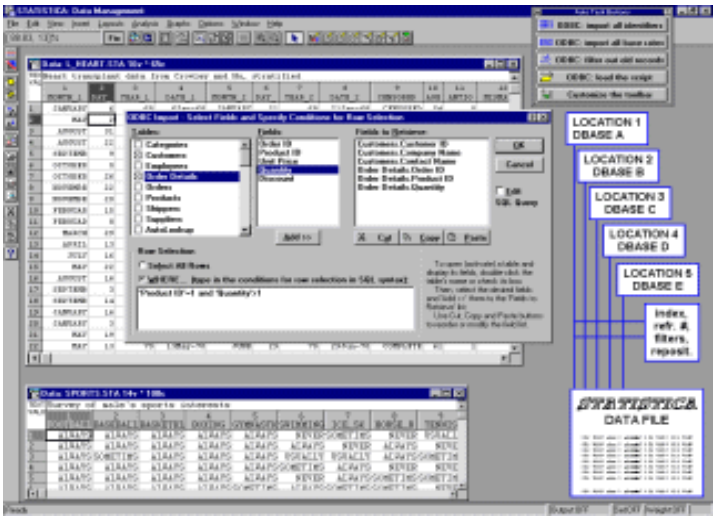



Рисунок 9.1 – Сховище даних SENS компанії StatSoft Enterprise Systems.

У основі концепції сховищ даних лежить ідея об'єднання корпоративних даних, розсіяних по системах оперативної обробки даних, історичних архівах і інших зовнішніх джерелах. Ці джерела можуть містити дані, які не використовуються безпосередньо в системах обробки інформації, але що є життєво необхідними для СППР: законодавча база (включаючи податкові прогнози), плани розвитку галузей, статистичні дані, електронні довідники. Як показує практика, рішення, прийняте на основі лише внутрішніх даних, найчастіше виявляється некоректним.

Мета концепції сховищ даних – прояснити відмінності в характеристиках даних в операційних і аналітичних системах, визначити вимоги даним, що поміщаються в сховище, визначити загальні принципи і етапи його побудови, основні джерела даних, дати рекомендації по рішенню потенційних проблем, що виникають при їх вивантаженні, очищенні, узгодженні, транспортуванні і завантаженні до цільової базу даних сховища.

Предметом концепції сховищ даних є не аналіз даних, а власне дані, тобто концепція їх підготовки для подальшого аналізу.

### 9.2 Технології побудови сховищ даних

Ідея, покладена в основу технологій інформаційних сховищ, полягає в тому, що проводити оперативний аналіз безпосередньо на базі інформаційних систем неефективно. Натомість, всі необхідні для аналізу дані витягуються з декількох традиційних баз даних (в основному, реляційних), перетворюються і потім поміщаються в одне джерело даних –

сховище даних.

В процесі занурення дані:

- очищаються – усунення непотрібної інформації;
- агрегуються – обчислення сум, середніх;
- трансформуються – перетворення типів даних, реорганізація структур зберігання;
- об’єднуються із зовнішніх і внутрішніх джерел – приведення до єдиних форматів;
- синхронізуються – відповідність одному моменту часу.

Сьогодні, технології побудови сховищ даних є основою для створення повноцінних інтелектуальних система налізу даних, орієнтованих на рішення слабко структурованих задач прийняття рішень, оскільки вони містить дані, що володіють наступними властивостями:

Цілісністю і внутрішнім взаємозв’язком. Хоча дані занурюються з різних джерел, але вони об’єднані єдиними законами іменування, способами вимірювання атрибутів і т.д. Це має велике значення для корпоративних організацій, в яких одночасно можуть експлуатуватися різні по своїй архітектурі обчислювальні системи, що представляють однакові дані по – різному. Наприклад, можуть використовуватися декілька різних форматів представлення дат, або один і той же показник може називатися різним чином, наприклад, «вірогідність доведення інформації» і «вірогідність отримання інформації». В процесі занурення подібні невідповідності усуваються автоматично.

Предметною орієнтованістю. Локальні бази даних містять мегабайти інформації, абсолютно не потрібної для аналізу (адреси, поштові індекси, ідентифікатори записів і т.п.). Подібна інформація не заноситься в сховище, що обмежує спектр даних, що розглядаються при ухваленні рішення до мінімуму.

Відсутністю часової прив’язки. Оперативні системи охоплюють невеликий інтервал часу, що досягається за рахунок періодичної архівації даних. Сховища даних, навпаки, містять історичні дані, накопичені за великий інтервал часу (роки, десятиліття).

Доступністю виключно для читання. Модифікація даних не проводиться, оскільки вона може привести до порушення цілісності сховища даних. Оскільки не потрібно мінімізувати час занурення, то структура сховища може бути оптимізована для обробки певних запитів, що досягається за рахунок денормалізації реляційної схеми, попередньої агрегації і побудови найбільш доречних індексів.

Інтегрованість означає, що дані задовольняють вимогам всього підприємства, а не одній функції бізнесу. Цим сховище даних гарантує, що однакові звіти, що згенерували для різних аналітиків, міститимуть однакові результати.

Незмінність означає, що, потрапивши один раз в сховищі, дані там зберігаються і не змінюються. Дані в сховищі можуть лише додаватися.

Всі дані, які містяться в сховищі, можна розділити на наступні категорії:

- метадані (дані про даних);
- агреговані дані;
- детальні дані.

Важливою особливістю систем інтелектуального аналізу даних на основі сховищ даних є метадані. Це різного роду системні словники, що дозволяють контролювати склад і структуру інформації в сховищі, управляти процесами завантаження і розрахунків і т.п. Без прошарку управлінських даних сховище з часом загрожує перетворитися на велике електронне звалище.

Ключовою відмінністю інформації сховища є не тільки спосіб його наповнення (з різних зовнішніх і внутрішніх джерел), але і модель зберігання даних, особливо це стосується агрегованої інформації. У сховищі інформація розміщується в денормалізованому вигляді у формі класичної «сніжинки» або «зірки». Такий підхід дозволяє істотно понизити час відгуку бази даних при виконанні запитів. Не вдаючись до технологічних особливостей проектування, відзначимо, що відбувається це за рахунок деякої надмірності зберігання даних. Така форма зберігання інформації в корпоративних сховищах є загальносвітовою практикою. Разом з тим частина детальних даних цілком може зберігатися в нормалізованих таблицях. Вимога до спеціальних структур зберігання обумовлена деяким спеціальним способом використання даних, про що доцільно сказати докладніше. Використовувати інформацію, накопичену в сховищі, можна як за допомогою традиційних звітів, так і з використанням динамічних запитів до бази даних. Існують також абсолютно специфічні способи використання інформації, призначені спеціально для аналітичних завдань. До них відносяться так звані OLAP-технології (On-line Analytical Processing) і технології інтелектуального аналізу даних. З практичної точки зору рішучий крок, який робить OLAP – технологія, полягає в тому, щоб, відмовившись від зайвої спільності, зробити процес аналізу максимально швидким. В рамках цієї технології передбачається, що склад і структура показників для аналізу відомий наперед і міняється дуже рідко (для деяких видів систем – практично не міняється). Користувач може виконувати над даними в такому багатовимірному уявленні набір OLAP-операцій підйому (консолідації по деяких напрямах), спуску (деталізації по деякому напряму), повороту (зміни напряму сортування). Детальні дані в системах сховищ даних найчастіше є джерелом для інтелектуального аналізу [11].

Таким чином, дані, занурені в інформаційне сховище даних, організовуючись в інтегровану цілісну структуру, володіють природними внутрішніми зв’язками, набувають нових властивостей, що дає їм можливість набути статус інформації.

Розглянемо характеристики інтеграції даних в сховищі даних. Як вже відомо, метою інтеграції даних є отримання єдиної та цільної картини бізнес – даних. Для її досягнення застосуємо модель, яка включає додатки, продукти, технології та методи:

- додатки – це рішення, створені постачальниками відповідно до вимог клієнтів, які використовують один або більше продуктів інтеграції даних;
- продукти – це готові комерційні рішення, що підтримують одну або більше технології інтеграції даних;
- технології реалізують один або більше методів інтеграції даних;
- методи – це підходи до інтеграції даних, незалежні від технологій.

Існує три основні методи інтеграції даних: консолідація, федералізація і розповсюдження.

Консолідація даних – це основний підхід, який використовується програмними додатками сховищ даних для побудови і підтримки оперативних складів даних і корпоративних сховищ. Консолідація даних також може знайти застосування для створення залежної вітрини даних, але в цьому випадку в процесі консолідації використовується тільки одне джерело даних (наприклад, корпоративне сховище). У середовищі сховищ даних однієї з найпоширеніших технологій підтримки консолідації є технологія ETL (витягання, перетворення і завантаження – extract, transform, and load). Ще одна поширена технологія консолідації даних – управління змістом корпорації (Enterprise Content Management – ECM). Більшість рішень ECM направлені на консолідацію і управління неструктурованими даними, такими як документи, звіти і web – сторінки.

Перевагою консолідації даних є те, що цей підхід дозволяє здійснювати трансформацію значних об’ємів даних (реструктуризацію, узгодження, очищення і/або агрегацію) в процесі їх передачі від первинних систем до кінцевих місць зберігання. Деякі складнощі, пов’язані з даним підходом, – це значні обчислювальні ресурси, які потрібні для підтримки процесу консолідації даних, а також істотні ресурси пам’яті, необхідні для підтримки кінцевого місця зберігання. Але з урахуванням постійного вдосконалення апаратних засобів це не проблема.

Федералізація даних. Процес забезпечує єдину віртуальну картину одного або декількох первинних файлів даних. Якщо бізнес-додаток генерує запит до цієї віртуальної картини, то процесор федералізації даних витягує дані з відповідних первинних складів даних, інтегрує їх так, щоб вони відповідали віртуальній картині і вимогам запиту, і відправляє результати бізнес-додатку від якого прийшов запит. За визначенням, процес федералізації даних завжди полягає у витяганні даних з первинних систем на підставі зовнішніх вимог. Всі необхідні перетворення даних здійснюються при їх витяганні з первинних файлів. Інтеграція корпоративної інформації (Enterprise Information Integration – EII) – це приклад технології, яка підтримує федеральний підхід до інтеграції даних. Один з ключових елементів федеральної системи – це метадані, які використовуються процесором федералізації даних для доступу до первинних даних. В деяких випадках ці метадані можуть складатися виключно з визначень віртуальної картини, які ставляться у відповідність первинним файлам. У більш передових

рішеннях метадані також можуть містити детальну інформацію про кількість даних, що знаходяться в первинних системах, а також про шляхи доступу до них. Така розширена інформація може допомогти федеральному рішенню оптимізувати доступ до первинних систем.

Вважається, що основна перевага федерального підходу – той факт, що він забезпечує доступ до поточних даних і позбавляє від необхідності консолідувати первинні дані в новому сховищі даних. Але слід пам'ятати, що федералізація даних не дуже добре підходить для витягання і узгодження великих масивів даних або для тих додатків, де існують серйозні проблеми з якістю даних в первинних системах. Ще один істотний чинник – потенційний вплив на продуктивність і додаткові витрати на доступ до численних джерел даних під час виконання програми.

Розповсюдження даних. Додатки розповсюдження даних здійснюють копіювання даних з одного місця в інше. Ці додатки зазвичай працюють в оперативному режимі і проводять переміщення даних до місць призначення, тобто залежать від певних подій. Оновлення в первинній системі можуть передаватися в кінцеву систему синхронно або асинхронно. Синхронна передача вимагає, щоб оновлення в обох системах відбувалися під час однієї і тієї ж фізичної транзакції. Незалежно від використовуваного типу синхронізації, метод розповсюдження гарантує доставку даних в систему призначення. Така гарантія – це ключова відмітна ознака розповсюдження даних. Більшість технологій синхронного розповсюдження даних підтримують двосторонній обмін даними між первинними і кінцевими системами. Прикладами технологій, що підтримують розповсюдження даних, є інтеграція корпоративних додатків (Enterprise Application Ntegration – EAI) і тиражування корпоративних даних (Enterprise Data Replication – EDR).

Великою перевагою методу розповсюдження даних є те, що він може бути використаний для переміщення даних в режимі реального часу або близькому до нього. Інші переваги включають гарантовану доставку даних і двостороннє розповсюдження даних. Метод розповсюдження даних може також використовуватися для урівноваження робочого навантаження, створення резервних копій і відновлення даних, зокрема у разі надзвичайних ситуацій. Практичне застосування цього методу відрізняється чималою різноманітністю як в плані продуктивності, так і відносно можливостей реструктуризації і очищення даних. Деякі корпоративні продукти розповсюдження даних можуть підтримувати переміщення і реструктуризацію крупних масивів даних, тоді як продукти EAI часто мають обмежені можливості пересування великої кількості даних і їх реструктуризації. Одна з причин подібної відмінності – той факт, що в центрі архітектури тиражування корпоративних даних лежать дані, а в центрі технології EAI – повідомлення або транзакції [11].

Гібридний підхід. Методи, які використовуються додатками інтеграції даних, залежать

як від потреб бізнесу, так і від технологічних вимог. Достатньо часто додаток інтеграції даних використовує так званий гібридний підхід, який включає декілька методів інтеграції. Хороший приклад такого підходу – інтеграція даних про клієнтів (Customer Data Ntegration – CDI), метою якої є забезпечення узгодженої картини інформації про клієнтів. Найпростіший підхід до CDI – це створення консолідованого сховища даних про клієнтів, який містить дані, отримані з первинних систем. Відставання інформації в консолідованому сховищі залежатиме від режиму консолідації даних (оперативний або пакетний) і від частоти оновлення цієї інформації. Інший підхід до CDI – це федералізація даних, коли визначаються віртуальні бізнес – представлення даних про клієнтів в первинних системах. Ці уявлення використовуються бізнес-додатками для доступу до поточної інформації про клієнтів в первинних системах. При федеральному підході також може використовуватися довідковий файл метаданих для зв'язку інформації про клієнтів на основі загальних ключових елементів.

### 9.3 Вітрини та кіоски даних

У найбільш загальному виді сховища даних можуть бути розбиті на два типи: корпоративні сховища даних (Enterprise Data Warehouses) і кіоски або вітрини даних (Data Marts).

Корпоративні сховища даних містять інформацію, що відноситься до всієї корпорації і зібрану з безлічі оперативних джерел для консолідованого аналізу. Зазвичай такі сховища охоплюють цілий ряд аспектів діяльності корпорації і використовуються для ухвалення як тактичних, так і стратегічних рішень. Корпоративне сховище містить детальну і узагальнену інформацію, його об'єм може досягати від 50 Гбайт до одного або декількох терабайт. Вартість створення і підтримки корпоративних сховищ може бути дуже високою. Зазвичай їх створенням займаються централізовані відділи інформаційних технологій, причому створюються вони зверху вниз, тобто спочатку проектується загальна схема, і тільки тоді починається заповнення даними. Такий процес може займати декілька років.

Кіоски або вітрини даних містять підмножину корпоративних даних і будуються для відділів або підрозділів усередині організації. Кіоски даних часто будуються силами самого відділу і охоплюють конкретний аспект, що цікавить співробітників даного відділу. Кіоск даних може отримувати дані з корпоративного сховища (залежний кіоск) або, що поширеніше, дані можуть поступати безпосередньо з оперативних джерел (незалежний кіоск). Кіоски і сховища даних будуються за схожими принципами і використовують практично одні і ті ж технології.

Багато компаній, що усвідомлюють необхідність розробки корпоративного сховища даних, все ж таки не в силах справитися зі всіма завданнями виділення, стандартизації і

об’єднання терабайт даних. Натомість вони вважають за краще будувати кіоски (або вітрини) даних (Data Marts) – спеціалізовані сховища даних, присвячені тільки одному напряму діяльності організації. Кіоск (вітрина) даних – це, найчастіше, найбільш керований різновид сховища даних. Його безперечний недолік полягає в тому, що без сховища даних, яке охоплювало б інформацію всього підприємства, неможливо порівнювати і аналізувати дані по всіх відділах і процесах. У багатьох компаніях вже зрозуміли, що кіоски (вітрини) даних можуть послужити хорошу службу і навіть стати єдино можливим рішенням для виконання термінових аналітичних завдань, але створення спеціалізованих кіосків без попередньої розробки корпоративної інфраструктури сховища даних може згодом привести до великих труднощів.

За класичним визначенням, вітрина (кіоск) даних (Data Mart) є підмножиною сховища даних, що відображає специфіку підрозділу (бізнес – об’єкт) і що забезпечує підвищену продуктивність. Таким чином, вітрина є ланкою, на якій базується конкретна аналітична система для вирішення свого кола завдань. Проте можлива ситуація, коли деяка область діяльності підприємства практично не корелює з іншими, і можливо побудувати відповідну вітрину даних автономно, без прив’язки до корпоративного сховища. Тоді вітрина поповнюватиметься даними безпосередньо з оперативних систем обробки транзакцій. Такі вітрини даних отримали назву незалежних, на відміну від класичних залежних від сховища даних і поповнюваних з нього вітрин.

Схема «зірка» – популярний тип моделі даних для вітрин даних. Дана модель характеризується наявністю таблиці фактів, оточеної пов’язаними з нею таблицями розмірностей. Запити до такої структури включають прості об’єднання таблиці фактів з кожною з таблиць розмірностей. Характеризується високою продуктивністю запитів. Проектується для виконання аналітичних запитів. Характеризується невеликою надмірністю даних і високою в порівнянні з нормалізованими структурами продуктивністю. Деякі промислові СУБД і інструменти класу OLAP/Reporting уміють використовувати переваги схеми «зірка» для скорочення часу виконання запитів.

Факти. Факти – це зазвичай числові величини, що зберігаються в таблиці фактів і є предметом аналізу. Приклади фактів: об’єм операцій, кількість проданих одиниць товару і так далі. Факти мають ряд властивостей, на яких ми коротко зупинимося.

Адитивні факти. Адитивність визначає можливість підсумування факту уздовж певної розмірності. Такі факти можна підсумовувати і групувати уздовж всієї розмірності на будь-яких рівнях ієрархії.

Напівадитивні факти. Напівадитивний факт – це факт, який можна підсумовувати уздовж певної розмірності, і не можна – уздовж інших. Прикладом може служити залишок на рахунку (або залишок товару на складі). Дану величину не можна підсумовувати уздовж

розмірності ЧАС. Проте сума залишків по рахунках уздовж розмірності є предметом для аналізу.

Фахівці рекомендують моделювати напівадитивні факти так, щоб зробити їх більш адитивними. Наприклад, представити відсоток складовими його величинами.

Неадитивні факти. Неадитивні факти взагалі не можна підсумовувати. Приклад неадитивного факту – відношення (наприклад, виражене у відсотках).

Таблиці-покриття. Таблиці-покриття використовуються з метою моделювання поєднання розмірностей, для яких відсутні факти. Наприклад, потрібно знайти кількість категорій продуктів, які сьогодні жодного разу не продавалися. Таблиця фактів продажів не може відповісти на дане питання, оскільки вона реєструє лише факти продажів. Для того, щоб модель дозволяла відповідати на подібні питання, потрібна додаткова таблиця фактів (яка, по суті справи, не містить фактів).

Схема «сніжинка» використовується для нормалізації схеми «зірка». Вона декілька скорочує надмірність в таблицях розмірності. Одною з переваг є швидше виконання запитів про структуру розмірності (запити вигляду «Вибрати всі рядки з таблиці розмірності на певному рівні»), які дуже часто виконуються при аналізі даних, і можуть затримувати хід аналізу. Проте основною відмінністю схеми «сніжинка» є не економія дискового простору, а можливість мати таблиці фактів з різним рівнем деталізації. Наприклад, фактичні дані на рівні дня, а планові – на рівні місяця.

Методика побудови вітрин (кіосків) даних з простої теоретичної дисципліни поступово перетворюється на складну науку, повну варіацій і напрямів. Якщо раніше було відомо лише про EDW (Enterprise Data Warehouse), то тепер з’явилися поступово розвиваємі вітрини даних (Incremental Architected Data Mart, ADM), розподілені вітрини (кіоски) даних (Distributed Data Mart, DDM), об’єднані вітрини даних (Federated Data Mart, FDM). Розглянемо деякі з цих нових напрямків.

Системи об’єднаних вітрин даних. У багатьох організаціях склалася практика реалізації багаточисельних сховищ даних. Хоча, за визначенням, існує лише одне сховище даних, а всі останні об’єкти є його підмножиною або вітринами (кіосками) даних, що поступово розвиваються, але не всі організації дотримуються цього правила. Таким чином, в багатьох компаніях існує два, три, десяток і навіть більше систем сховищ даних. Поширення сховищ даних привело до розвитку архітектури сховища даних підприємства, а саме: до появи об’єднаних систем сховищ даних або вітрин (кіосків) даних.

Система об’єднаних вітрин даних характеризується спільним використанням загальних інформаційних ресурсів, усуваючи, таким чином, надмірність і гарантуючи достовірність інформації по всій організації [11].

Позитивними рисами об’єднаних вітрин даних є: загальна семантика бізнес-правила;

один набір процесів витягання і бізнес-правил; децентралізовані ресурси і управління; паралельна розробка.

Недоліками такого архітектурного рішення є: необхідність в координуванні робіт; складнощі в подоланні «політичних» моментів і вирішенні питань авторських прав; потрібна узгодженість серед різних відділів по питаннях архітектури, бізнес-правил і семантики; складне технічне середовище; наявність багаточисельних репозиторіїв метаданих.

Непроектуємі вітрини даних. Поява непроектуємих вітрин даних (Non – Architected Data Marts) пояснюється, перш за все, складнощами, пов'язаними з реалізацією систем EDW і FDW. Брудні і швидко отримувані набори даних не піддаються очищенню і, отже, не можуть використовуватися для подальшої інтеграції з будь-якими іншими джерелами даних систем сховищ даних. Дуже швидко вони перетворюються на застарілі системи, які лише додають проблем, а не вирішують їх. Для цих систем характерні багаточисельні процеси витягання, безліч бізнес-правил, невірогідність інформації.

Позитивними рисами непроектуємих вітрин даних є: висока продуктивність; низька вартість. Недоліками таких систем є: недостовірна інформація; багаточисельні процеси витягання; багаточисельні бізнес-правила; підвищена складність при інтеграції.

Система вітрин (кіосків) даних, що поступово розвиваються. Ця архітектура є альтернативою сховища даних підприємства. Для наповнення таких вітрин зазвичай використовується інструментальний засіб класу підприємства, що реалізовує стратегію «витягаєш один раз, наповнюєш багато».

Перевагами таких вітрин даних є: єдиний набір процесів витягання; здійснимий масштаб. Недоліки: найбільш ефективні при використанні інструментального засобу класу підприємства; необхідність в архітектурі вітрин даних підприємства (Enterprise Data Mart Architecture, EDMA).

Методика побудови вітрин (кіосків) даних виявилася напрямом ринку проектів інтелектуального аналізу даних, що нестримно розвивається, швидко змінюється. Якщо раніше не було механізмів їх ефективного проектування, і був лише один способів їх створення, в даний час можна знайти незчисленне число таких інструментів і ряд технологій життєздатної архітектури таких систем. За умови вибору відповідної архітектури і належного підходу до проекту можна побудувати систему сховища та вітрин даних, яка забезпечить не лише високе повернення інвестицій, але і значно підвищить ефективність функціонування всього підприємства.

### 9.4 OLAP-технологія

Вміння швидко і головне правильно приймати рішення має в сучасному бізнесі

принципове значення для досягнення успіху. Проте кількість інформації, що впливає на предмет рішення, інколи може бути просто-таки величезною. Що робити в такій ситуації? Покластися на випадок або все ж таки взятися за повномасштабний аналіз? Досвідчений керівник незмінно вибере другий спосіб, тим більше що сьогодні існує ряд технологій, здатних спростити процес прийняття і моделювання рішень при великій кількості «вхідної» інформації.

Власне, аби спростити роботу з багатоцільовими даними і не загрузнути в їх океані, а також уміло перетворити набір кількісних показників на якісні, і застосовується метод OLAP – On-Line Analytical Processing (оперативна аналітична обробка). Останній, на відміну від інших способів автоматизації бізнес-діяльності, дає можливість отримати користувачеві «на виході» не готове чітко структуроване рішення, що видається після включення раніше налагодженого майстра обробки форм, а своєрідний матеріал для наочної і, якщо можна так виразитися, творчої оцінки існуючої ситуації. Тому сфера вживання OLAP-аналізу зазвичай обмежується менеджерським складом підприємств різних розмірів, якому доводиться часто займатися тактичними і стратегічними завданнями на зразок аналізу ключових показників діяльності і сценаріїв розвитку, маркетинговим і фінансово-економічним аналізом груп товарів або послуг, а також довгостроковим прогнозуванням роботи підприємства або його підрозділів. Для цього користувач OLAP-систем отримує в руки потужний і головне дуже гнучкий інструмент створення різних звітів по вибраних їм же розрізах і напрямах. При цьому методики OLAP більш досконаліша за звичні електронні таблиці, адже окрім простих функцій створення таблиць, графіків і діаграм, OLAP-системи дають можливість отримати узагальнені дані по самостійно вибраних критеріях, вмить поглибиться в деталі вибраних напрямів, відфільтрувати, сортувати або відкинути непотрібні цифри або показники. Наприклад, якщо менеджерові продажів компанії потрібно отримати сезонні зведення динаміки продажів вибраній категорії товарів, система запропонує йому всілякі дані про продажі за місяць, квартал, рік, а також знайде і проаналізує їх залежність від зазначених чинників, скажімо, часу проведення маркетингових акцій. Крім того, базуючись на одній лише статистиці продажів, OLAP-система може виявити ефективність роботи різних підрозділів компанії, у тому числі і в розрізі географічної ієрархії їх взаємодії. При цьому параметри, що характеризують успішність підрозділів, вибираються менеджером самостійно і у ряді випадків можуть стати інструментом мотивації успішного персоналу.

Щоб зрозуміти, як працюють OLAP-системи, досить звернутися до її механізмів. Найбільш показове поняття OLAP-технології – гіперкуб (метакуб), що є умоглядною фігурою в багатовимірному просторі, утвореному площинами даних, які важливі для діяльності підприємства. При цьому сама OLAP-система виступає саме в ролі гіперкуба, здатного накопичувати в собі всю інформацію, що цікавить керівника. Як ребра куба виступають різні

категорії товарів або послуг, що надаються компанією. Наприклад, ціна виробленого або конкурентного товару, компанії-учасники виробничого циклу, підрядчики при організації послуг, об'єми продажів, географія самої компанії. Важливо відзначити, що градація різних осей квадрантів куба може мати різну структуру, а крім того, самі осі можуть бути взаємозалежними. Так, вісь часу може бути розбита по роках, кварталах, тижнях, а вісь доходів або шкідливих викидів при виробництві – прологарифмована. Інформація, необхідна для аналізу в даний момент, вирізається з гіперкуба перетином площини даних, що використовуються при аналізі, немов його шар або частина. При цьому розрізи можуть проходити як через весь куб, так і обмежуватися певними рамками і межами осей.

Технічно системи оперативного аналізу даних зазвичай функціонують у зв'язці з сховищами та вітринами даних, а клієнтські OLAP-системи встановлюються на будь-яких призначених для користувача комп'ютерах корпоративної інформаційної системи. Рідше OLAP-модулі взаємодіють з іншими системами автоматизації, адже бази даних останніх досить часто мають вельми своєрідний вигляд і набір спеціальних показників. Втім, для сучасного українського підприємства характерна нетипова ситуація, коли є декілька систем автоматизації (для вирішення різних завдань) і, як наслідок, дані зберігаються розрізнено, в результаті відсутній єдиний погляд на управлінську інформацію. Тому в процесі складання звіту беруть участь два фахівці – програміст, що забезпечує запити до баз даних, і економіст, що намагається за допомогою електронних таблиць звести ці дані в звіт, необхідний керівництву. Як показує практика, подібна модель взаємодії користувача звіту (керівника) і самих даних незмінно приводить до ефекту «зіпсованого телефону», не говорячи вже про істотні витрати часу. І з даної точки зору використання OLAP-систем також представляється вельми раціональним, адже використання декількох інформаційних систем незмінно приводить до «надлишку» даних, які можуть бути впорядковані OLAP-системою.

У чому ж відмінність OLAP-системи від сховища даних? З точки зору користувача, відповідь на це питання досить проста: у мірі предметної структурованості інформації. Працюючи з OLAP-додатком, користувач застосовує звичні економічні категорії і показники – види матеріалів і готової продукції, регіони продажів, об'єм реалізації, собівартість, прибуток і тому подібне. А для того, щоб сформувати будь-який, навіть досить складний запит, користувачеві не доведеться вивчати SQL. При цьому відповідь на запит буде отримана протягом всього декількох секунд. Крім того, працюючи з OLAP – системою, економіст може користуватися такими звичними для себе інструментами, як електронні таблиці або спеціальні засоби побудови звітів. Таким чином, якщо сховище даних – в основному об'єкт уваги спеціаліста по інформаційним технологіям, то OLAP без перебільшення можна назвати програмним засобом з арсеналу економіста. Адже саме економіст має справу з самими різними аналітичними задачами: маркетинговим аналізом, аналізом продажів, аналізом бюджетних

показників, аналізом фінансової звітності і так далі.

### 9.5 Основні архітектури OLAP-систем

Системи інтелектуального аналізу даних зазвичай володіють засобами надання користувачеві агрегованих даних для різних вибірок з початкового набору у зручному для сприйняття й аналізу вигляді. Як правило, такі агреговані функції утворюють багатовимірний (і, отже, не реляційний) набір даних, який нерідко називають гіперкубом або метакубом. Осі цього куба містять параметри (вимірювання), а клітинки – залежні від них агреговані значення.

Хоча фізичне зберігання таких даних може реалізовуватися в реляційних таблицях, у даному контексті нас цікавить логічна структура даних. Уздовж кожної осі гіперкуба дані можуть бути організовані у вигляді ієрархії, що відображає різні рівні деталізації. Завдяки цій моделі користувачі можуть формулювати складні запити, генерувати звіти й отримувати підмножини даних відповідно до своїх потреб.

Технологія комплексного багатовимірного аналізу даних отримала назву OLAP (On-Line Analytical Processing). OLAP є ключовим компонентом сучасних сховищ даних. Концепцію OLAP у 1993 році запропонував Едгар Кодд – відомий дослідник баз даних і автор реляційної моделі. У 1995 році, на основі його підходів, було сформульовано тест FASMI (Fast Analysis of Shared Multidimensional Information), який включає п’ять основних вимог до OLAP-систем:

- Fast (Швидкість): надання результатів аналізу за прийнятний час (зазвичай не більше 5 секунд), навіть за рахунок меншої деталізації;
- Analysis (Аналіз): підтримка логічного та статистичного аналізу із збереженням результатів у зрозумілому для користувача вигляді;
- Shared (Спільний доступ): підтримка багатокористувацького режиму з відповідними механізмами безпеки;
- Multidimensional (Багатовимірність): концептуальна багатовимірна модель даних з підтримкою ієрархій;
- Information (Інформаційність): доступ до будь-якої необхідної інформації незалежно від її розміру та місця зберігання.

OLAP-функціональність може реалізовуватися по-різному: від простих інструментів в офісних програмах до складних серверних аналітичних систем. Проте перед тим як розглядати реалізації, варто зрозуміти, що собою являють OLAP-куби з логічної точки зору.

Розібравшись з концепцією OLAP-куба, варто ознайомитися з деякими базовими термінами, які використовуються у багатовимірному аналізі даних:

- Summary – агреговані значення в клітинках куба;

- Measure – вихідні числові дані, на основі яких обчислюються агрегати;
- Dimension (вимірювання) – параметри, за якими здійснюється групування та аналіз;
- Member – конкретне значення вимірювання (наприклад, рік, регіон, клієнт).

Кожне вимірювання може мати кілька рівнів деталізації. Наприклад, аналіз може проводитися на рівні країни, міста чи окремого клієнта. Чим нижчий рівень деталізації, тим точніший і глибший аналіз. Можливість отримання зрізів даних з різним рівнем деталізації відповідає одній з ключових вимог до сховищ даних – забезпечення гнучкості аналітики.

Також у клітинках OLAP-куба можуть зберігатися результати різних агрегатних функцій SQL, таких як MIN, MAX, AVG, COUNT, а також статистичних метрик – наприклад, дисперсії, середньоквадратичного відхилення тощо.

Вимірювання в OLAP-моделі можуть бути організовані у вигляді ієрархій, що відображають різні рівні деталізації даних. Наприклад, у випадку аналізу замовлень клієнтів, значення вимірювання географія можуть мати трирівневу ієрархію: на першому рівні – країни, на другому рівні – міста, на третьому рівні – клієнти.

Це означає, що в кожній країні може бути декілька міст, а в кожному місті – декілька клієнтів. Така структура дозволяє здійснювати як узагальнений аналіз (на рівні країни), так і деталізований (до окремого клієнта).

Відзначимо, що ієрархії можуть бути збалансованими, як ієрархії, засновані на даних типу «дата-час», і незбалансованими. Типовий приклад незбалансованої ієрархії – ієрархія типу «керівник-підлеглий». Іноді для таких ієрархій використовується термін Parent-child hierarchy.

Існують також ієрархії, що займають проміжне положення між збалансованими і незбалансованими, вони позначаються терміном ragged – «нерівний». Зазвичай вони містять такі елементи, логічні «батьки» яких розміщені не на безпосередньо вищому рівні.

Багатовимірний аналіз даних може бути проведений за допомогою різних засобів, які умовно можна розділити на клієнтські і серверні OLAP-засоби. Клієнтські OLAP-засоби є додатки, що здійснюють обчислення агрегатних даних (сум, середніх величин, максимальних або мінімальних значень) і їх відображення, при цьому самі агрегатні дані містяться в кеші усередині адресного простору такого OLAP-засобу. Якщо початкові дані містяться в настільній СУБД, обчислення агрегатних даних проводиться самим OLAP – засобом. Якщо ж джерело початкових даних – серверна СУБД, багато хто з клієнтських OLAP-засобів посилає на сервер SQL-запити і в результаті отримують агрегатні дані, обчислені на сервері.

Відзначимо, що клієнтські OLAP-засоби застосовуються, як правило, при малому числі вимірювань (зазвичай рекомендується не більше шести) і невеликій різноманітності значень цих параметрів, – адже отримані агрегатні дані повинні уміщатися в адресному просторі подібного засобу, а їх кількість росте експоненціально при збільшенні числа вимірювань.

Тому навіть найпримітивніші клієнтські OLAP-засоби, як правило, дозволяють зробити попередній підрахунок об'єму необхідної оперативної пам'яті для створення в ній багатовимірного куба. Багато клієнтських OLAP-засобів дозволяють зберегти вміст кеша з агрегатними даними у вигляді файлу, що, у свою чергу, дозволяє не проводити їх повторне обчислення. Відзначимо, що нерідко така можливість використовується для відчуження агрегатних даних з метою передачі їх іншим організаціям або для публікації. Типовим прикладом таких відчужуваних агрегатних даних є статистика захворюваності в різних регіонах і в різних вікових групах, яка є відкритою інформацією, що публікується міністерствами охорони здоров'я різних країн і Всесвітньою організацією охорони здоров'я.

Ідея збереження кеша з агрегатними даними у файлі отримала свій подальший розвиток в серверних OLAP-засобах, в яких збереження і зміна агрегатних даних, а також підтримка сховища, що містить їх, здійснюються окремим додатком або процесом, званим OLAP-сервером. Клієнтські додатки можуть запрошувати подібне багатовимірне сховище і у відповідь отримувати ті або інші дані. Деякі клієнтські додатки можуть також створювати такі сховища або оновлювати їх відповідно до початкових даних, що змінилися [11].

Переваги застосування серверних OLAP-засобів в порівнянні з клієнтськими OLAP – засобами: у разі застосування серверних засобів обчислення і зберігання агрегатних даних відбуваються на сервері, а клієнтський додаток отримує лише результати запитів до них, що дозволяє в загальному випадку понизити мережевий трафік, час виконання запитів і вимоги до ресурсів, споживаним клієнтським додатком. Відзначимо, що засоби аналізу і обробки даних масштабу підприємства, як правило, базуються саме на серверних OLAP-засобах, наприклад, таких як Oracle Express Server, Microsoft SQL Server 2000 Analysis Services, Hyperion Essbase, продуктах компаній Crystal Decisions, BusinessObjects, Cognos, SAS Institute. Оскільки всі провідні виробники серверних СУБД створюють ті або інші серверні OLAP-засоби, вибір їх достатньо широкий і майже у всіх випадках можна придбати OLAP – сервер того ж виробника, що і у самого сервера баз даних.

OLAP-система складається з множини компонент. На найвищому рівні уявлення система включає джерело даних, OLAP-сервер і клієнта. Джерело даних є засіб, з якого беруться дані для аналізу. Дані з джерела переносяться або копіюються на OLAP-сервер, де вони систематизуються і готуються для швидшого згодом формування відповідей на запити. Клієнт – це призначений для користувача інтерфейс до OLAP-сервера.

Джерела даних. Джерелом в OLAP-системах є сервер, що поставляє дані для аналізу. Залежно від області використання OLAP-продукту джерелом може служити сховище даних, успадкована база даних, що містить загальні дані, набір таблиць, об'єднуючих фінансові дані або будь-яка комбінація перерахованого. Здатність OLAP – продукту працювати з даними з різних джерел дуже важлива. Вимога єдиного формату або єдиної бази, в яких би зберігалися

всі початкові дані, не підходить адміністраторам баз даних. Крім того, такий підхід зменшує гнучкість і потужність OLAP-продукту. Адміністратори і користувачі вважають, що OLAP-продукти, що забезпечують витягання даних не тільки з різних, але і з безлічі джерел, виявляються гнучкішими і кориснішими, ніж ті, що мають жорсткіші вимоги.

Сервер. Прикладною частиною OLAP-системи є OLAP-сервер. Ця складова виконує всю роботу і зберігає в собі всю інформацію, до якої забезпечується активний доступ. Архітектурою сервера управляють різні концепції. Зокрема, основною функціональною характеристикою OLAP – продукту є використання для зберігання даних багатовимірної (ММБД, MDDB) або реляційної (РДБ, RDB) бази даних.

MOLAP. MOLAP – це Multidimensional On-Line Analytical Processing, тобто Багатовимірний OLAP. Це означає, що сервер для зберігання даних використовує ММБД. Оскільки більшість OLAP-продуктів засновані на ММБД, під OLAP часто розуміють також і MOLAP. Сенс використання ММБД очевидний. Вона може ефективно зберігати багатовимірні за своєю природою дані, забезпечуючи засоби швидкого обслуговування запитів до бази даних. Дані передаються від джерела даних в багатовимірну базу даних, а потім база даних піддається агрегації. Попередній розрахунок – це те, що прискорює OLAP – запити, оскільки розрахунок зведених даних вже проведений. Час запиту стає функцією виключно часу, необхідного для доступу до окремого фрагмента даних і виконання розрахунку. Цей метод підтримує концепцію, згідно якої робота проводиться одного разу, а результати потім використовуються знову і знову. Багатовимірні бази даних є відносно новою технологією. Використання ММБД має ті ж недоліки, що і більшість нових технологій. А саме – вони не так стійкі, як РБД, і в тій же мірі не оптимізовані. Інше слабке місце ММБД полягає в неможливості використовувати більшість багатовимірних баз в процесі агрегації даних, тому потрібний час для того, щоб нова інформація стала доступна для аналізу.

«Вибух» бази даних є феномен багатовимірних баз. Не дивлячись на те, що ця проблема досліджувалася фахівцями, проте, важко пояснити, чому і як це відбувається. Представляється, що це пов’язано з розрідженістю бази даних і попередньою агрегацією даних. Якщо багатовимірна база даних містить невелике число елементів даних, порівнянне з кількістю забезпечуваних нею рівнів агрегації, кожен фрагмент даних вноситиме більший внесок до всіх отримуваних з нього даних. Коли база даних «вибухає», розмір її стає істотно більше, ніж він повинен бути. Складно визначити умови «вибуху» бази даних або передбачити, чи «вибухне» якась конкретна структура. Одним з підходів, який, схоже, може допомогти вирішити проблему «вибуху», є динамічне управління розрідженими даними. Ця методика дозволяє аналізувати свої власні моделі зберігання і оптимізувати їх з метою запобігання «вибуху» бази даних.

ROLAP. ROLAP – це Relational On-Line Analytical Processing, тобто Реляційний OLAP.

Термін ROLAP означає, що OLAP-сервер базується на реляційній базі даних. Початкові дані вводяться в реляційну базу даних, зазвичай по схемі «зірка» або схемі «сніжинка», що сприяє скороченню часу витягання. Сервер забезпечує багатовимірну модель даних за допомогою оптимізованих SQL-запитів. Існує ряд причин для вибору саме реляційною, а не багатовимірної бази даних. РБД – це добре відпрацьована технологія, що має безліч можливостей для оптимізації. До того ж, РБД підтримують крупніші об'єми даних, чим ММБД. Вони якраз і спроектовані для таких об'ємів. Основним аргументом проти РБД є складність запитів, необхідних для отримання інформації з великої бази даних за допомогою SQL. Недосвідчений SQL-програміст міг би з легкістю обтяжити цінні системні ресурси спробами виконати який-небудь подібний запит, який в ММБД виконується набагато простіше.

Прикладний OLAP (HOLAP). Безумовно, це найбільша область, і це, загалом, те, з чим зазвичай зв'язують і що зазвичай розуміють під терміном «OLAP». Прикладний OLAP, як правило, складається з багатовимірних баз даних, доступ до яких відбувається через конкретний додаток, або, можливо, через безліч додатків. Постачальники в даній області ринку в основному пропонують клієнти для бази даних. Клієнт може бути як простим засобом перегляду, так і могутнішим додатком.

Настільний OLAP (DOLAP). Представниками настільного OLAP є продукти, що необов'язково з'єднуються з сервером. Вони можуть запускатися в основному на клієнтській частині, хоча дані у формі куба даних можуть завантажувати і з сервера. Той факт, що куб даних будується і зберігається на машині користувача, дозволяє рекомендувати їх тим, хто часто використовує портативні комп'ютери або хто нечасто запускає настільки складні звіти, що для їх формування необхідна вища швидкість клієнта, а, отже, і могутніший сервер для її забезпечення.

Швидка реалізація запитів є імперативом для OLAP. Це один з базових принципів OLAP – здатність інтуїтивно маніпулювати даними вимагає швидкого витягання інформації.

Різні постачальники дотримуються різних методів відбору параметрів, що вимагають попередньої агрегації і числа заздалегідь обчислюваних величин. Підхід до агрегації впливає одночасно і на базу даних і на час реалізації запитів. Якщо обчислюється більше величин, вірогідність того, що користувач запитає вже обчислену величину, зростає, і тому час відгуку скорочується, оскільки не доведеться запрошувати початкову величину для обчислення. Проте, якщо обчислити всі можливі величини – це не краще рішення – у такому разі істотно зростає розмір бази даних, що зробить її некерованою, та і час агрегації буде дуже великим. До того ж, коли в базу даних додаються числові значення, або якщо вони змінюються, інформація ця повинна відбиватися на заздалегідь обчислених величинах, залежних від нових даних.

Клієнт. Клієнт – це якраз те, що використовується для уявлення і маніпуляцій з даними в базі даних. Клієнт може бути і достатньо нескладним – у вигляді таблиці, що включає такі можливості OLAP, як, наприклад, обертання даних і поглиблення в дані. Воно повинно бути спеціалізованим, але мати такий же простий засіб поглядання звітів або бути таким же могутнім інструментом, як створений на замовлення додаток, спроектований для складних маніпуляцій з даними. Клієнт є настільки важливий, що безліч постачальників зосереджують свої зусилля виключно на розробці клієнта. Все, що включається до складу цих додатків, представляє собою стандартний погляд на інтерфейс, наперед задані функції і структуру, а також швидкі рішення для більшості стандартних ситуацій, наприклад, популярні фінансові пакети. Наперед створені фінансові додатки дозволять фахівцям використовувати звичні фінансові інструменти без необхідності проектувати структуру бази даних або загальноприйняті форми і звіти.

Інструмент запитів/генератор звітів. Інструмент запитів або генератор звітів пропонує простій доступ до OLAP-даних. Вони мають простий у використанні графічний інтерфейс і дозволяють користувачам створювати звіти переміщенням об’єктів методом «drag and drop». Тоді як традиційний генератор звітів надає користувачеві можливість швидко випускати форматовані звіти, генератори звітів, підтримуючі OLAP, формують актуальні звіти. Кінцевий продукт є звіт, що має можливості поглиблення в дані до рівня подробиць, обертання (півотінг) звітів, підтримки ієрархій і ін.

Додатки. Додатки – це тип клієнта, що використовує бази даних OLAP. Вони ідентичні інструментам запитів і генераторам звітів, описаним вище, але, крім того, вони вносять до продукту ширші функціональні можливості. Додаток, як правило, володіє більшою потужністю, чим інструмент запиту.

Середовище. Зазвичай постачальники OLAP забезпечують середовище розробки для створення користувачами власних настроєних додатків. Середовище розробки в цілому є графічним інтерфейсом, що підтримує об’єктно-орієнтовану розробку додатків. До того ж,

більшість постачальників забезпечують API, який може використовуватися для інтеграції баз даних OLAP з іншими додатками.

Розглянемо деякі напрями діяльності основних виробників програмних засобів підтримки OLAP – технологій з урахуванням вищенаведеної класифікації архітектур.

Різні постачальники реалізують OLAP на основі власних корпоративних уявлень про те, що повинно входити в ідеальний OLAP-продукт. Постачальники, розглянуті нижче, дотримувалися різних підходів, включаючи і ті, що засновані на багатовимірній базі даних, реляційній базі даних і додатках, що реалізують можливості OLAP на різних рівнях. Процес їх реального функціонування може служити кількісним параметром при розгляді різних підходів до OLAP. З цією метою OLAP Council розробив атестаційне завдання APB-1 для кількісного порівняння роботи різних OLAP-продуктів. OLAP Council є консорціумом, утвореним декількома постачальниками OLAP для підтримки ключових принципів OLAP. Вони визнають, що OLAP є найважливішою технологією для забезпечення корпоративних аналітиків інструментами, потрібними для виконання необхідного їм аналізу. У квітні 1996 р. було випущено перше атестаційне завдання в області OLAP – APB-1. Контрольне завдання визначає параметри бази даних і встановлює набір з 10 запитів, що відображають нормальне використання. Постачальник OLAP створює відповідну базу даних і потім запускає запити. Окреме вимірювання – AQT (середній час запиту, Average Query Time), генерується на основі часу, який витрачається на завантаження бази даних, агрегацію даних і подальший запуск запитів. Правила визначають, що легально і що нелегально – наприклад, чи потрібно розраховувати наперед значення даних, які з розрахованих величин можуть зберігатися і ін.

В області MOLAP архітектури лідером є компанія Oracle [43]. Лінійка продуктів Express та Oracle OLAP Services є корпоративним підходом Oracle до OLAP, включаючи сервер, клієнтську частину, можливості ROLAP і Інтернет – рішення. Express Server був головною OLAP-машиною для Oracle. Він надавав багатовимірну базу даних і був машиною для інших OLAP-продуктів фірми. До того ж, Oracle пропонує Personal Express – локально працюючий сервер Express. Він надає користувачам доступ і можливість працювати з базою даних в автономному режимі. Все це ідеально підходить для мобільних комп’ютерів. Головною особливістю Express Server є здатність використовувати різноманітні джерела даних. Дані для багатовимірної бази даних можуть збиратися з реляційної бази (за допомогою Express Relational Access Manager), багатовимірної бази, табличного або плоского файлу. Користувачі можуть розробляти власні OLAP-додатки для Express за допомогою Express Objects, який забезпечує розробника графічним інтерфейсом і об’єктно-орієнтованим підходом для створення додатків. Oracle також забезпечує безліч шляхів доступу до бази даних. Express Analyzer містить графічний інтерфейс до бази даних і дозволяє користувачеві легко формувати звіти. Крім того, Analyzer може мати загальні об’єкти з Express Objects, а

також випускати додатки, розроблені з їх допомогою. Discoverer найточніше можна описати як інструмент запитів до даних. Він простіший, ніж Analyzer, проте найбільш популярний серед засобів запитів до даним. Крім цього, Express містить додаткові таблиці, які можна використовувати спільно з Excel.

У Oracle існує два готові додатки. Це Financial Analyzer і Sales Analyzer. Вони використовують машину і кеш даних Express Server, і містять конфігурації звітів, розроблені для потреб фінансових аналітиків і аналітиків продажів. Вони корисні і користувачам, оскільки визначають важливі функції, що зазвичай беруть участь в обох видах аналізу, який реалізовано в Express Server.

Web Agent і Web Publisher – це засоби створення Інтернет-ресурсів, що надаються Express. Використовуючи будь-який з цих засобів, користувач може створити динамічний і інтерактивний сайт, що забезпечується застосуванням різних можливостей OLAP. Web Agent в більшій мірі інструмент розробника, він є набором заздалегідь певних процедур, включених в Express Server SPL. Сайти можуть будуватися так само, як будуються призначені для користувача інтерфейси до бази даних. Для кінцевого користувача існує Web Publisher. Web Publisher зв'язаний з Express Analyzer і має можливість для створення власних інтерактивних сайтів тими, хто не має серйозного досвіду програміста. Web Publisher в основному є «майстром», який веде користувача через всі етапи побудови сайту і забезпечує графічний інтерфейс для підтримки його створення.

Arbor Software Corporation – це головний суперник Oracle. Її продуктом є Essbase, а найостаннішим його релізом був Essbase Server 5. Дуже популярний сервер Arbor для різних OLAP-продуктів, що говорить про те, що багато постачальників OLAP не обов'язково випускають повні додатки, а можуть використовувати як базу даних Arbor, і потім створювати інтерфейси до бази даних. Як приклад можна привести Comshare і Web – компоненту для Arbor – Crystalinfo, Seagate Software, що випускається. Останнім часом Arbor уклав партнерську угоду з IBM. Багатовимірне зберігання даних в Arbor Essbase Server буде замінено на DB2 від IBM. Передбачається, що це буде ROLAP – система, але фактично це не так. Це просто OLAP-система без однієї з кращих властивостей багатовимірних баз даних, але з перевагами системи РБД.

В доповнення до таблиць Essbase Spreadsheet Add-in, що забезпечують користувачів можливостями OLAP, Arbor пропонує WIRED for OLAP (засіб аналізу і презентацій), Crystal Info for Essbase (генератор звітів і розкладів) і SQL Drill-Through, що дозволяє користувачам проглядати подробиці даних в початкових реляційних базах. Arbor також випустив Arbor Essbase Adjustment Module. Цей додаток допомагає користувачам в підготовці звітів, що регулярно випускаються. Він сприяє автоматизації форматування звітів і процесів розрахунку. Крім того, існує ще Arbor Currency Conversion Module, здатний конвертувати різні валюти в

національну на основі моделі для відстежування обмінних курсів.

В області ROLAP визнаним лідером є компанія MicroStrategy. Їх філософією є відсутність обмежень на розмір сховища даних, так що немає жодних проблем з його збільшенням. Оскільки вони є виробниками реляційного OLAP, рівень їх аналітичних систем достатньо високий. У MicroStrategy немає OLAP-машини, яка могла б працювати локально, що незручно для користувачів, які часто працюють з ноутбуками або для тих, хто просто вважає за краще працювати автономно. Проте, у них існує продукт, DSS Broadcaster, що дозволяє посилати дані на різні вихідні пристрої. DSS Broadcaster посилає дані за запитом або коли відбувається певна подія. Наприклад, менеджерові може відсилатися щоденне оновлення з сумами прибутку за попередній день. Ця інформація може поступати по електронній пошті, на пейджер або мобільний телефон, а також факсом.

DSS Server є центральним продуктом в лінійці продуктів MicroStrategy. Це могутня машина, що дозволяє іншим агентам діставати доступ до реляційної бази даних в багатовимірному режимі. DSS Server містить різноманітні драйвери баз даних для оптимізації їх під необхідну реляційну базу даних (вони підтримують Oracle, DB2, Sybase, Red Brick, Informix, і інші реляційні бази). До того ж, акцент робиться на здатність їх зростання і включає драйвер для адаптації до дуже великих баз даних (Very Large Databases, VLDBs), розмір яких перевищує терабайти. Природа реляційного OLAP-продукту обмежує MicroStrategy в можливості надання дійсно індивідуального сервера для автономної роботи, проте за допомогою DSS Agent набір даних може завантажуватися і аналіз може виконуватися і в автономному режимі. DSS Agent є клієнт або клієнтський інструмент до DSS Server. Однією з переваг DSS Agent є використання інтелектуальних агентів для автоматизації бізнес – процесів. Наприклад, за допомогою DSS Agent можна створити агента, що знає, коли і де необхідно шукати дані і потім що з ними робити потім (тобто, як проводити їх очищення і куди їх помістити). Використовуючи агентів, можна автоматизувати безліч звичайних, але часто повторюваних завдань. DSS Executive використовує можливості DSS Agent для реалізації високоякісного генератора звітів і засобу аналізу. Він використовує об’єктно-орієнтований підхід і інтерфейс, що працює за технологією «drag-and-drop» для швидкого створення додатків управлінських інформаційних систем силами самих користувачів. І, нарешті, MicroStrategy пропонує доповнення Excel Add-in, яке може використовуватися для додання таблицям функціональних можливостей OLAP.

Нове покоління Інтернет-орієнтованого OLAP від MicroStrategy представлене DSS Web 5.0. Однією з примітних властивостей DSS Web 5.0 є підтримка Microsoft webcasting standart. Це дозволяє автоматично передавати web-сторінки на комп’ютер користувача. У числі найважливіших можливостей DSS Web можна назвати здатність зберігати карти або діаграми, отримані з Інтернет, майстри звітів і пакети звітів, що настроюються.

Як приклад прикладного OLAP-продукту можна узяти Comshare. Не дивлячись на те, що це додаток, воно доповнює продукт функціональними можливостями OLAP. Comshare Decision проявляє гнучкість щодо використовуваного спільно з ним сервера. Arbor Essbase і Oracle Express – всі ці багатовимірні сервери баз даних можуть використовуватися спільно з Decision [11].

Hyperion Software – це також виробник прикладного OLAP-продукту, що випускає виключно OLAP-клієнти. Останнім продуктом був Hyperion MBA, або Multidimensional Business Analyst, що замінив HyperionOLAP. Згідно з останніми даними, Hyperion займав другий за величиною сегмент ринку. Природа продукту гарантує, що велика частина функціональних можливостей заснована на серверній базі даних. Hyperion популярний завдяки своїм додатковим аналітичним можливостям, що реалізовуються у формі складних вимірювань, заздалегідь певних функцій і звітності. Метою його є формування могутнього фінансового пакету, що включає OLAP.

Hyperion пропонує два клієнтські OLAP-рішення. Перше, HyperionMBA, використовує OLAP для бізнес-аналізу. Як це зазвичай буває в OLAP-додатках, Hyperion застосовує в своїх рішеннях складні вимірювання і заздалегідь певні функції для розрахунків і маніпуляцій з валютами. Програма Hyperion Analytic Accounting включає властивості OLAP в розрахункові пакет.

Cognos служить непоганим прикладом настільного OLAP-продукту. Це означає, що велика частина обробки проводиться не на сервері, а локально. Impromptu є інструментом запитів, використовуваним для витягання даних з багатовимірної бази даних. Дані потім поміщаються в Powerplay, яка зберігає куб даних на робочому столі комп’ютера користувача.

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Viktoriia H., Bohdan D. Інтелектуальний аналіз даних з використанням weka explorer. International scientific and technical conference Information technologies in metallurgy and machine building. 2024. С. 287-289. URL: https://doi.org/10.34185/1991-7848.itmm.2023.01.077 (дата звернення: 08.05.2025).

2. Unsupervised Learning: Association Rules / K. J. Cios et al. Data Mining. Boston, MA. P. 289-306. URL: https://doi.org/10.1007/978-0-387-36795-8_10 (date of access: 09.04.2025).

3. Supervised Learning: Statistical Methods / K. J. Cios et al. Data Mining. Boston, MA. P. 307-379. URL: https://doi.org/10.1007/978-0-387-36795-8_11 (date of access: 08.04.2025).

4. Supervised Learning: Decision Trees, Rule Algorithms, and Their Hybrids / K. J. Cios et al. Data Mining. Boston, MA. P. 381-417. URL: https://doi.org/10.1007/978-0-387-36795-8_12 (date of access: 08.04.2025).

5. Data Science / Mr. Vasudev Shahapur et al. International Journal of Advanced Research in Science, Communication and Technology. 2022. P. 487-495. URL: https://doi.org/10.48175/ijarsct-2904 (date of access: 27.04.2025).

6. Classification of Landsat 8 Images Using Convolutional Neural Network Based on Minimum Noise Fraction Transform / V.O. Lishchyna et al. 2024 35th Conference of Open Innovations Association (FRUCT), Tampere, Finland, 24-26 April 2024. 2024. URL: https://doi.org/10.23919/fruct61870.2024.10516385 (date of access: 07.05.2025).

7. Guillod J. Data Science. Python Programming for Mathematics. Boca Raton, 2024. P. 189-219. URL: https://doi.org/10.1201/9781003565451-12 (date of access: 04.05.2025).

8. Jeyaraj R., Pugalendhi G., Paul A. Data Science. Big Data with Hadoop MapReduce. Includes bibliographical references and index., 2020. P. 357-369. URL: https://doi.org/10.1201/9780429321733-7 (date of access: 05.05.2025).

9. Ліщина Н., Ліщина В. Однопрохідний алгоритм аналітичного опису контурів об’єктів. Die wichtigsten vektoren für die entwicklung der wissenschaft im jahr 2020. 2020. URL: https://doi.org/10.36074/24.01.2020.v1.20 (дата звернення: 07.05.2025).

10. Krishna G. G. Multilingual NLP. International Journal of Advanced Engineering and Nano Technology. 2023. Vol. 10, no. 6. P. 9-12. URL: https://doi.org/10.35940/ijaent.e4119.0610623 (date of access: 09.04.2025).

11. Knowledge Graph OLAP / C. G. Schuetz et al. Semantic Web. 2020. P. 1-35. URL: https://doi.org/10.3233/sw-200419 (date of access: 27.04.2025).

Інтелектуальний аналіз даних : конспект лекцій з навчальної дисципліни для здобувачів першого (бакалаврського) рівня вищої освіти освітньої програми «Комп'ютерні науки» галузі знань 12 Інформаційні технології спеціальності 122 Комп'ютерні науки денної та заочної форм навчання / уклад. В.О. Ліщина, К.В. Вавринюк. Луцьк: ЛНТУ. 2025. 137 с.

Комп'ютерний набір і верстка: К. Вавринюк

Підписано до друку 2025 р.
Формат 60х 84/16. Гарнітура Times New Roman.
Папір офсетний 80 г/м2. Друк офсетний.
Ум. друк. арк. 7,5. Обл.-вид. арк. 75.
Наклад 50 прим. Зам. №

Інформаційно-видавничий відділ
Луцького національного технічного університету
43018, Луцьк-18, вул. Львівська,75.
Друк – ІВВ ЛНТУ