МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ

ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ОПОРНИЙ КОНСПЕКТ ЛЕКЦІЙ

з дисципліни

"ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ"

Тернопіль-2007

Л.І.Гончар. // Опорний конспект лекцій з дисципліни ”Інтелектуальний аналіз даних” для студентів напрямку „Комп’ютерні науки”.-Тернопіль, 2007.

Укладачі: **Гончар Людмила Іванівна**, доцент кафедри КН, ТНЕУ

Відповідальний за випуск: **Дивак Микола Петрович,** д.т.н., професор,

завідувач кафедри КН, ТНЕУ

Шпак Володимир Богданович,

інженер кафедри КН, ТНЕУ

Рецензенти: завідувач кафедри Безпеки інформаційних технологій ТНЕУ, д.т.н., професор **Карпінський М.П.**

доцент кафедри Біотехнічних систем Тернопільського державного технічного університету імені І.Пулюя, к.т.н., доцент **Шадріна Г.М.**

Затверджено на засіданні кафедри комп'ютерних наук ТНЕУ,

Протокол № 9 від 7 лютого 2007 р.

ЗМІСТ

ВСТУП .....

I. Дейтамайнінг — засоби інтелектуального аналізу даних у СППР.....

1.1. Представлення нової технології інтелектуального аналізу даних.....

1.2. Суть і складові інтелектуальної фази при прийнятті рішень .....

1.3. Комп'ютеризовані засоби підтримки інтелектуальної фази .....

1.4. Методологічні засоби підтримки інтелектуальної фази .....

II. Розвиток і призначення дейтамайнінгу (DataMining) .....

2.1. Поняття DataMining .....

2.2. Інтеграція OLAP-технологій та ІАД .....

2.3. DataMiningі сховища даних .....

III. Характеристика процесів і активностей дейтамайнінгу .....

3.1. Процеси дейтамайнінгу .....

3.2. Користувачі та дії дейтамайнінгу.....

3.3. Дерево методів дейтамайнінгу .....

1.1 3.3.1. Збереження даних (Data Retention).....

3.3.2. Дистиляціяшаблонів (DataDistilled) .....

IV. Генетичні алгоритми.....

4.1. Генетичні успадкування —концептуальна засада генетичних алгоритмів.....

4.2. Загальна схема генетичних алгоритмів.....

4.3. Доступне програмне забезпечення генетичних алгоритмів .....

V. Програмні агенти в СППР .....

5.1. Призначення і основні характеристики програмних агентів.....

5.2. Програмні агенти у СППР та BIC.....

VI. Доступне програмне забезпечення дейтамайнінгу.....

VII. Засоби Data Mining в Microsoft SQL Server 2000.....

VIII. Сфера застосування технологій інтелектуальних обчислень.....

8.1. Бізнес-застосування Data Mining.....

8.2. Технології ІАД та український ринок .....

ЛІТЕРАТУРА .....

ВСТУП

Засоби сучасної інформаційної технології в останній час уможливили накопичення і зберігання великих обсягів даних про бізнесові процеси. Ці дані можуть знаходитися в корпоративних базах або сховищах даних. Вони містять важливі закономірності і зв’язки між системними характеристиками, які можуть бути використані для прийняття обгрунтованих управлінських рішень. Наразі виникла проблема розробки методів відкриття таких закономірностей, про існування яких користувачі можуть і не знати. Проте традиційний аналіз даних передбачує введення даних в стандартні або настроєні користувачем моделі, тобто в будь-якому випадку допускається, що зв'язки між різними показниками добре відомі і можуть бути виражені математично. Однак, в багатьох випадкахзв'язки не можуть бути апріорі відомі. У таких ситуаціях моделювання стає неможливим і тут можна застосовувати дейтамайнінг (*DataMining) – інтелектуальний аналіз даних (ІАД).* Тому, особливо важливим аспектом підготовки спеціалістів напрямку "Комп'ютерні науки” є успішне засоєння ними дисципліни "Інтелектуальний аналіз даних”.

У результаті вивчення дисципліни “Інтелектуальний аналіз даних” студент повинен :

знати - сутність та призначення DataMining; характеристики процесів та активностей дейтамайнінгу;дерево методів дейтамайнінгу;доступне програмне забезпечення ІАД; призначення та основні характеристики генетичних алгоритмів і програмних агентів;

вміти - будувати дерево методів дейтамайнінгу; проводити кластерний аналіз засобами дейтамайнінгу;здійснювати вибір відповідних логічних методів із побудовою таблиці трансакцій;будувати крос-таблицю;вміло застосовувати доступне програмне забезпечення дейтамайнінгу.

Опорний конспект лекцій з дисципліни “Інтелектуальний аналіз даних” включає 8 розділів, кожний із яких містить необхідний методичний матеріал для вивчення даного предмету.

I. Дейтамайнінг — засоби інтелектуального аналізу даних у СППР

1.1. Представлення нової технології інтелектуального аналізу даних (ІАД)

Комп'ютерні технології із застосуванням інтелектуальних обчислень переживають свій розквіт. Це пов'язано, головним чином, з потоком нових ідей, що виходять з галузі комп'ютерних наук, яка утворилась на перетині штучного інтелекту, статистики та теорії баз даних. Зараз відбувається стрімкий зріст числа програмних продуктів, що використовують нові технології, а також типів задач, де їх застосування надає значного економічного ефекту. Елементи автоматичної обробки і аналізу даних, що називають DataMining (знаходження знань) стають невід'ємною частиною концепції електронних сховищ даних та організації інтелектуальних обчислень. Простий доступ користувача до сховища даних забезпечує тільки отримання відповідей на питання, що були задані, в той час як технологія datamining дозволяє побачити ("знайти") приховані правила і закономірності у наборах даних, які користувач не може передбачити, і застосування яких може сприяти збільшенню прибутків підприємства.

DataMining переводиться як "видобуток" чи "добування даних". Нерідко поруч з DataMining зустрічаються слова "інтелектуальний аналіз даних". Справа в тому, що людський розум сам по собі не пристосований для сприйняття великих масивів різнорідної інформації. Але і традиційна математична статистика, яка довгий час претендувала на роль основного інструмента аналізу даних, також нерідко відстає при вирішенні складних життєвих задач. Вона оперує усередненими характеристиками вибірки, що часто є фіктивними величинами (типу середньої температури пацієнтів в лікарні, середньої висоти будинку на вулиці тощо). Тому методи математичної статистики виявляються корисними, головним чином, для перевірки заздалегідь сформульованих гіпотез.

Можливості інтелектуального аналізу

Більшість підприємств накопичують під час своєї діяльності величезні обсяги даних, але єдине, що вони хочуть від них одержати - це корисну інформацію. Яким чином можна довідатися з даних про те, що є найбільш потрібним для їхніх клієнтів, як найефективніше використати наявні ресурси або як мінімізувати втрати? Для вирішення цих проблем призначені новітні технології інтелектуального аналізу. Вони використовують складний статистичний аналіз і моделювання для знаходження моделей і відношень, прихованих у базі даних - таких моделей, що не можуть бути знайдені звичайними методами. Доти поки модель не відповідає існуючим реально відношенням, неможливо отримати успішні результати. Технології інтелектуального аналізу можуть не тільки підтвердити емпіричні спостереження, але і знайти нові, невідомі раніше моделі. За допомогою методів datamining можна знайти таку модель, що приведе до радикального поліпшення у фінансовому і ринковому становищі компанії.Хоча інструментарій інтелектуального аналізу і звільняє користувача від можливих складностей у застосуванні статистичних методів, він все-таки потребує від нього розуміння роботи цього інструментарію й алгоритмів, на яких він базується. Крім цього, технологія знаходження нового знання в базі даних не може дати відповіді на ті питання, що не були задані. Вона не заміняє аналітиків чи менеджерів, а дає їм сучасний, могутній інструмент для поліпшення роботи, яку вони виконують.

1.2. Суть і складові інтелектуальної фази при прийнятті рішень

Процес створення рішення розпочинається з фази обдумування**,** протягомякоїдосліджується реальність (ситуація прийняття рішення), ідентифікується проблема та визначається особа або група осіб, відповідальних за її розв'язок (тримачі проблеми).

Інтелектуальна фаза розпочинається з ідентифікації організаційної мети або цілей, зв'язаних з поточними бізнесовими питаннями (наприклад, обчислення оптимального рівня запасу, формування замовлень на виготовлення продукції, проведення диверсифікаційних заходів тощо), визначення, чи ці питання взагалі існують і наскільки вони важливі. Проблеми, котрі потребують вирішення, зазвичай виникають при незадовільному стані бізнесової діяльності, коли фактичні здобутки не виправдовують витрачених на їх отримання зусиль або суттєво відрізняються стандартів продуктивності (наприклад, фінансових індикаторів).

Ідентифікація проблеми включає, крім визначення реальності її існування, виділення її симптомів, окреслення її масштабів та формулювання проблеми в явному виді. Часто те, що описується як проблема (наприклад, надмірні витрати) може бути тільки симптомом або мірою іншої проблеми (наприклад, як невідповідний рівень запасу). Оскільки реальні, світового рівня проблеми зазвичай надзвичайно ускладнені в силу дії багатьох взаємозв'язаних чинників (що, в загальному випадку, не підвладні творцям рішення), тому в практичному менеджменті інколи важко відрізнити симптоми від конкретних проблем.

Існування проблеми можна визначити за допомогою моніторингу бізнесових подій і транзацій та шляхом аналізування рівня організаційної продуктивності. Вимірювання продуктивності і створення відповідної моделі має базуватися на реальних даних. Збирання даних і оцінювання майбутніх значень параметрів системи є найбільш важким кроком аналізу. Виділимо головні випадки, що можуть мати місце протягом етапів збирання та оцінювання даних і які створюють труднощі при ідентифікації проблеми та її вирішення:

- Дані є недоступними. В даному разі заміна їх неточними оцінками, які можуть бути покладені при створенні моделі, приводить до помилкових рішень.
- Одержування даних може бути дорогим, тобто витрати матеріальних і трудових ресурсів або непосильні творцю рішень, або значно перевищують потенційні вигоди від рішення.
- Оцінювання даних часто буває суб'єктивним, тобто не відображати реальний стан речей.
- Важливі дані, що впливають на результат, є якісними, тому виникає проблема їх квантифікації.
- Великий обсяг надходжуваних даних, які не може збагнути творець рішення (інформаційне перевантаження). В даному разі рішення або не створюється вчасно, або вони грунтується на фрагментах загальної картини реальної ситуації, що в любому випадку є небажаним.
- Наслідки чи результати рішення можуть відбуватися за межами визначеного періоду часу. В такому випадку вартісні елементи рішень, такі як значення доходів, витрат і прибутків, мають бути придатними для запису в різні моменти часу. З цією метою, зокрема, використовується підхід поточної вартості майбутнього (present-value), пов'язаний з дисконтуванням витрат і надходжень.
- Часто робляться припущення, що майбутні дані будуть подібні до історичних. В такому разі має бути упевненість в тому, що минулі ситуації повторюються. Якщо так не можна діяти, то потрібно передбачити природу можливих змін даних і включити це в аналіз.

Як тільки буде завершене попереднє дослідження, то це дає можливість визначити, чи дійсно проблема існує, де вона розміщена і настільки вона суттєва. Інтелектуальна фаза, як правило, завершується формальним формулюванням проблеми.

1.3. Комп'ютеризовані засоби підтримки інтелектуальної фази

Перша вимога щодо підтримки рішення для інтелектуальної фази є отримання здатності переглядати зовнішні і внутрішні інформаційні джерела для окреслення можливостей бізнесової системи та ідентифікації проблем, а також щоб надійно інтерпретувати виявлені скануванням (переглядом) ситуаційної обстановки факти і закономірності. Підтримуючі рішення засоби інформаційної технології тут виявляються надзвичайно доречними.

Найбільш пристосованими (можливо і спеціально розробленими) для першої фази створення рішень є так звані *орієнтовані на дані* системи підтримки прийняття рішень (*Data-driven DSS)* . Ця категорія включає системи управління створенням звітів, сховище даних і системи аналізу, виконавчі інформаційні системи (ВІС), географічні інформаційні системи (ГІС), системи бізнесової інформації (BusinessIntelligenceSystems), системи оперативного аналітичного оброблення OLAP(on-lineanalyticprocessing). В цих типах СППР робиться наголос на доступі і маніпулювання з великими БД структурованих даних, часовими рядами внутрішніх данихкомпанії і деякими зовнішніми даними.

Наприклад, головне призначення ВІС є підтримка інтелектуальної фази за допомогою безперервного моніторингу зовнішньої і внутрішньої інформації, перегляду ранніх ознак проблем і можливостей. В даний час на ринку програмних продуктів пропонуються десятки комерційних продуктів виконавчих інформаційних систем, створених різними компаніями світу, лідерами серед яких є:

корпорації *PilotSoftware, Inc*., що володіє 25 % ринку ВІС за доходом, найбільш відомими ВІС цієї корпорації -- *Commander Center, Lightship* і *Lightship Lens*. У центрі уваги цих програмних продукту — ідентифікація і стеження за ключовими індикаторами (показчиками) діяльності фірми;

ComshareInc. , що володіє 60 % ринку ВІС за доходом. Найбільш відомою ВІС цієї корпорації є *Commander EIS* (дозволяє розпізнавати ключові індикатори або "важливі коефіцієнти успіху", а далі відстежувати їх).

Набули розповсюдження також інші ВІС, зокрема розроблена фірмою Execucomз використанням відомого продукту IFPS/Plus*Executive Edge*; інститут SAS розробив SAS/EIS як середовище для розробки ВІС, що включає об'єкти для побудови ВІС.

Системи сховищ даних, які дозволяють маніпулювання даними за допомогою комп'ютеризованих інструментальних засобів, пристосовані до специфічних задач, є більш загальнимиінструментальними засобами і операціями, що забезпечують додаткові функціональні можливості. На даний час пропонуються сотні різних засобів для створення сховищ даних. У створенні великих сховищ даних лідирують корпорації IBM, Informix, NCR, Oracle, Red Brick, SAS, Sybase, Microsoft. Крім того, на ринку продуктів для побудови і використання сховищ даних значне місце займають Brann Software. Business Objects,

Cayenne Software, Computer Associates, MicroStrategy, Prism Solutions, Brio Technology, Cognos, Platinum Technology .

СППР з оперативною аналітичною обробкою (OLAP) забезпечують найвищий рівень функціональних можливостей і підтримки рішення, яка поєднана з аналізом великих сукупностей історичних даних. Однією з найбільш відомих реалізацій ідеї оперативної аналітичної обробки, що інтенсивно впроваджується в Україні, є сімейство програмних продуктів Oracle Express OLAP, котре являє собою інструментально-технологічне програмне забезпечення, призначене для створення прикладних аналітичних систем підтримки прийняттярішень на основі багатовимірного аналізу даних. Є низка інструментальних засобів для кінцевого користувача, доступні для підтримки OLAP. Вони включають Business Object Inc.Business Objects, програмне забезпечення AG Esperant, Andyne PaBLO, Visualizer IBM і Platinum Forest & Trees. Ці, а також десятки інших інструметальних засобів OLAP продуктивно можуть використовуватися для підтримки інтелектуальної фази створення рішень.

Системи бізнесової інформації (бізнес-інтелектуальні системи) призначені для аналізу великих за обсягом масивів даних, поданих у вигляді *гіперкубів даних* .

Географічна інформаційна система (ГІС) – програмно-апаратний комплекс, призначений для збору, керування, аналізу і відображення просторово-розподіленої інформації. ГІС є підтримуюча система, яка представляє дані з використання карт (мап). Вона допомагає менеджерам мати доступ, показувати і аналізувати дані, які мають географічний зміст і значення. Окремі типи ГІС доречні в аналізі маршрутизації і розміщення, маркетингу і в інших традиційних областях бізнесу. Також програмне забезпечення ГІС забезпечує зв'язок між інтерфейсом користувача і базою даних, тому користувач може запитувати і аналізувати просторові дані. Прикладом цього типу програмного забезпечення є програмне забезпечення ГІС ArcInfo8 підприємства ESRI. ArcInfo призначено, щоб допомогти користувачам запитувати і бачити просторові дані. Інший, широко використовуваний продукт настільного відображення, є MapInfo.

Важливим джерелом інформації для підтримки інтелектуальної фази створення рішень є традиційні інформаційні систем менеджменту (МІС), наприклад 1С, а також сучасні широкомасштабні корпоративні системи (R/3, Scala 5, OracleApplication, Baan-IV, ГАЛАКТИКА), в яких забезпечується інтегроване оброблення інформації всіх бізнесових областей (маркетинг, виробництво, фінанси).

Засоби і технології дейтамайнінгу (Data Mining) також виявитися надзвичайно корисними для фази обдумування проблеми . До числа найбільш відомих програмних продуктів дейтамайнінгу слід віднести PolyAnalyst, MineSet, KnowlengeSTUDIO**.** Для прогнозування окремих показників і параметрів бізнесової діяльності на інтелектуальній фазі створення рішення в останній час широко застосовуються програмні засоби нейромереж. На ринку програмних продуктів пропонується десятки придатних для використання нейропакетів (наприклад, NeuroShelL).

Орієнтовані на знання СППР, зокрема експерні системи (ЕС) і правило-орієнтовані СППР також підтримують інтелектуальну фазу. ЕС можуть надати поради про природу проблеми, її класифікацію, її серйозність і тому подібне. ЕС можуть створювати рекомендації щодо придатності вибраного для розв’язування проблеми підходу та ймовірності успіху розв'язування. Однією із перших областей успішного застосування ЕС є

проблеми інтерпретації інформації і діагностика. Ці можливості можуть використовуватися в інтелектуальній фазі.

1.4. Методологічні засоби підтримки інтелектуальної фази

На стадії обдумування і формулювання проблеми, котра вимагає подальшого вирішення, зазвичай основне "інтелектуальне" навантаження лягає на творця рішення. Тому методологічна підтримка має бути зорієнтовна головно на *суб'єктивне оцінювання* інформації і обставин. З цією метою використовується низка методів і підходів, більшість яких вмонтовані в методологічну базу різних СППР. До числа таких методів можна віднести: *дерева цілей, оцінювання імовірностей, матриця аномальних подій, мозкова атака, метод Дельфі, метод історичних аналогій, порівняльний аналіз, вивчення прикладів, журі або симульоване опитування думок.* З метою упорядкування вхідної інформації і на цій основі отримання якісно нової може застосовуватися *морфологічний аналіз* як упорядкований спосіб розгляду предметів і отримання систематизованої інформації стосовно всіх можливих розв'язків досліджуваної проблеми . Для задач таксономії окремих елементів рішення може застосовуватися *кластерний аналіз.* Виокремлення проблеми і її симптомів, виділення підпроблем та установлення їх ієрархії зручно проводити за допомогою *методів дерев рішень* та *діаграм впливу.*

II. Розвиток і призначення дейтамайнінгу (DataMining)

2.1. Поняття *DataMining*

У 70-х роках минулого століття широко застосовувалася практика, коли компанії наймали аналітиків з бізнесу, котрі, використовуючи статистичні пакети подібні SAS і SPSS, виконували аналіз трендів даних і проводили їх кластерний аналіз. Як тільки стало технологічно можливим і доцільним зберігати великі обсяги даних, менеджери виявили бажання самим мати доступ до даних, подібних тим, що генеруються в пам'яті касового апарата роздрібної торгівлі й аналізувати їх. Запровадження штрихових кодів і глобальна гіпертекстова система Інтернету також зробили реальною можливість для компаній збирати великі обсяги нових даних. Однак у зв'язку з цим виникло питання про інструментальні засоби добування корисної інформації з нагромаджених обсягів «сирих» даних. Ці засоби опісля отримали назву «DataMining» (дейтамайнінг).

Слід зауважити, що протягом багатьох років компанії проводили статистичні дослідження своїх даних. Коли статистик аналізує дані, то він спочатку висуває гіпотезу про можливий зв'язок між певними даними, а потім посилає запит до бази даних і використовує відповідні статистичні методи, щоб довести або спростувати сформульовану гіпотезу. Це підхід називається *«режимом верифікації» («verificationmode»).* На противагу йому програмне забезпечення дейтамайнінгу функціонує в *«режимі відкриття» (discoverymode),* тобто виявляє приховані, часто невідомі для користувачів *шаблони (patterns)* зв'язків між даними, а не аналізує наперед створену гіпотезу щодо них.

За останні роки надзвичайно зріс інтерес до дейтамайнінгу з боку ділових користувачів, котрі вирішили скористатися перевагами даної технології для отримання конкурентної переваги в бізнесі (див. http://www.datamining.com/). Зростаюча зацікавленість

щодо впровадження дейтамайнінгу (ДМ) у результаті закінчилася появою низки комерційних продуктів, кожен з яких має таку саму назву, описаний низкою подібних елементів, але фактично має неоднакові функціональні можливості й ґрунтується на різних особливих технічних підходах.

Менеджери з інформаційних технологій, що мають завдання підібрати відповідну СППР, часто безпосередньо зустрічаються зі складними питаннями стосовно реагування на потреби бізнес-користувачів через те, що засадні принципи створення дейтамайнінгу набагато складніші, ніж традиційні запити і формування звітів, крім того, вони відчувають підсилений тиск щодо часу реалізації потреб користувачів, тобто користувачі вимагають розробити дейтамайнінг якомога швидше. Проте очевидною перешкодою для розроблення і впровадження в корпораціях рішень з дейтамайнінгу є наявність багатьох різних підходів до нього, що мають свої певні властивості й переваги, у той час як фактично тільки кількома основними методами формуються основи більшості систем ДМ. У цьому контексті важливою є однозначна інтерпретація самого поняття дейтамайнінгу.

Дейтамайнінг (Datamining) — це тип аналітичних додатків які підтримують рішення, розшукуючи за прихованими шаблонами (patterns) інформацію в базі даних. Цей пошук може бути зроблений або користувачем (тобто тільки за допомогою виконання запитів) або інтелектуальною програмою, яка автоматично розшукує в базах даних і знаходить важливі для користувача зразки інформації. Відповіді на інформаційні запити подаються в бажаній для користувача формі (наприклад, у вигляді діаграм, звітів тощо).

Англомовний термін «Datamining» часто перекладається як *«добування даних»; «добування знань»; «добування інформації»; «аналіз, інтерпретація і подання інформації зі сховища даних»; «вибирання інформації із масиву даних».* У даній книзі буде використовуватися як основний термін *«дейтамайнінг»* — україномовна транскрипція початково запровадженого і однозначно вживаного в англомовній літературі терміна «Datamining».

Добування даних — це процес фільтрування великих обсягів даних для того, щоб підбирати відповідну до контексту задачі інформацію. Вживається також термін *«Datasurfing»* (дослідження даних в Інтернеті). Корпорація IBM визначає ДМ, як «процес екстракції з великих баз даних заздалегідь невідомої, важливої інформації, що дає підстави для дій та використання її для розроблення критичних бізнесових рішень». Інші визначення не пов'язують ні з обсягом бази даних, ні з тим, чи використовується підготовлена інформація в бізнесі, але переважно ці умови загальні.

Інструментальні засоби добування даних використовують різноманітні методи, включаючи доказову аргументацію (case-basedreasoning), візуалізацію даних, нечіткі запити й аналіз, нейромережі та інші. Доказову аргументацію (міркування за прецедентами) застосовують для пошуку записів, подібних до якогось певного запису чи низки записів. Ці інструментальні засоби дають змогу користувачеві конкретизувати ознаки подібності підібраних записів. За допомогою візуалізації даних можна легко і швидко оглядати графічні відображення інформації в різних аспектах (ракурсах). Ці та інші методи частково були розглянуті раніше, а детальніше будуть розглянуті далі.

Дейтамайнінг як процес виявлення в загальних масивах даних раніше невідомих, нетривіальних, практично корисних і доступних для інтерпретації знань, необхідних для прийняття рішень у різних галузях людської діяльності, практично має нічим не обмежені сфери застосування. Але, насамперед, методи ДМ нині більше всього заінтригували

комерційні підприємства, що створюють проекти на основі сховищ даних (DataWarehousing), хоча наявність сховища даних не є обов'язковою умовою здійснення дейтамайнінгу. Досвід багатьох таких підприємств свідчить, що рівень рентабельності від застосування дейтамайнінгу може досягати 1000 %. Наприклад, відомі повідомлення про економічний ефект, за якого прибутки у 10—70 раз перевищували первинні витрати, що становили від 350 до 750 тис. дол. Є відомості про проект у 20 млн дол., який окупився всього за 4 місяці. Інший приклад — річна економія 700 тис. дол. за рахунок упровадження дейтамайнінгу в мережі універсамів у Великобританії.

Дейтамайнінг являє собою велику цінність для керівників і аналітиків у їх повсякденній діяльності. Ділові люди усвідомили, що за допомогою методів ДМ вони можуть отримати відчутні переваги в конкурентній боротьбі.

2.2. Інтеграція OLAP-технологій та ІАД

Оперативна аналітична обробка та інтелектуальний аналіз даних - дві складові частини процесу підтримки прийняття рішень. Але сьогодні більшість систем OLAP загострює увагу тільки на забезпеченні доступу до багатовимірних даних, а більшість засобів ІАД, що працюють у сфері закономірностей, мають справу з одновимірними перспективами даних. Ці два види аналізу повинні бути тісно об'єднані, тобто системи OLAP повинні фокусуватися не тільки на доступі, але і на пошуку закономірностей. Як відмітив N. Raden, "багато компаній створили ... прекрасні сховища даних, ідеально розклавши по поличках гори невживаної інформації, яка сама по собі не забезпечує ні швидкою, ні достатньо грамотної реакції на ринкові події".

Вчений K. Parsaye вводить складений термін "OLAP Data Mining" (багатовимірний інтелектуальний аналіз) для позначення такого об'єднання інший науковець J. Han пропонує ще простішу назву - "OLAP Mining", і пропонує декілька варіантів інтеграції двох технологій.

"Cubing then mining". Можливість виконання інтелектуального аналізу повинна забезпечуватися над будь-яким результатом запиту до багатовимірного концептуального уявлення, тобто над будь-яким фрагментом будь-якої проекції гіперкуба показників.

"Mining then cubing". Подібно даним, витягнутим з сховища, результати інтелектуального аналізу повинні представлятися в гіперкубічній формі для подальшого багатовимірного аналізу.

"Cubing while mining". Цей гнучкий спосіб інтеграції дозволяє автоматично активізувати однотипні механізми інтелектуальної обробки над результатом кожного кроку багатовимірного аналізу (переходу між рівнями узагальнення, витягання нового фрагмента гіперкуба і т. д.).

На жаль, дуже небагато виробників надають сьогодні достатньо могутні засоби інтелектуального аналізу багатовимірних даних в рамках систем OLAP. Проблема також полягає в тому, що деякі методи ІАД (байєсівські мережі, метод найближчого сусіда) непридатні для завдань багатовимірного інтелектуального аналізу, оскільки засновані на визначенні схожості деталізованих прикладів і не здатні працювати з агрегованими даними .

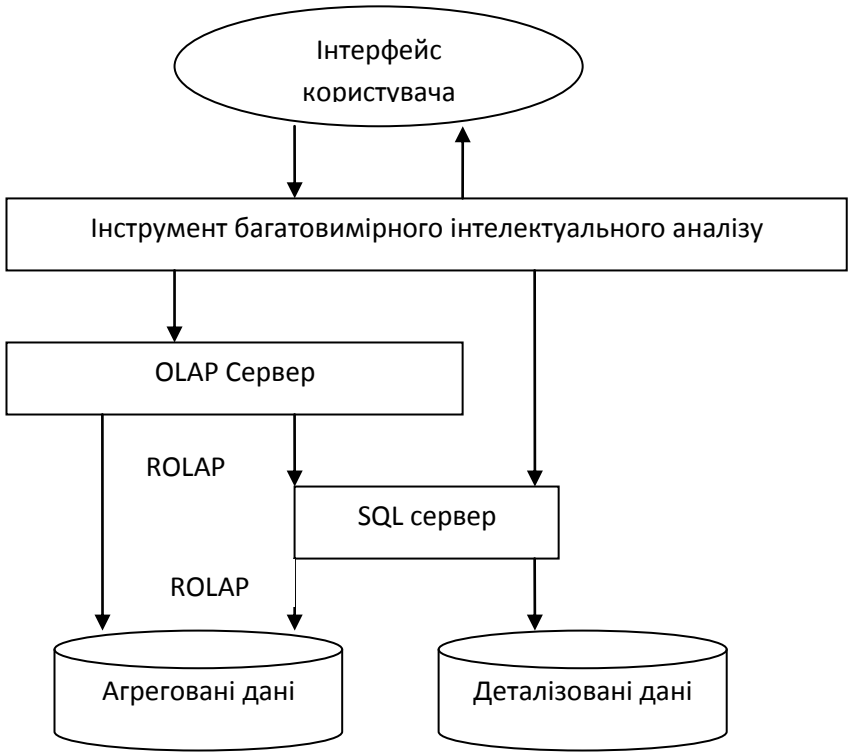


Рисунок 1. 8. Архітектура системи багатовимірного інтелектуального аналізу даних

Дуже часто виникає питання про різницю між засобами інтелектуального аналізу і OLAP-системами (On-LineAnalyticalProcessing) - засобами оперативної аналітичної обробки.

OLAP - це частина технологій, скерованих на підтримку прийняття рішення. Звичайні засоби формування запитів і звітів описують саму базу даних. Технологія OLAP використовується для відповіді на задані питання. При цьому користувач сам формує гіпотезу про дані чи відношення між даними і після цього використовує серію запитів до бази даних для підтвердження чи відхилення цих гіпотез. Засоби DataMining відрізняються від засобів OLAP тим, що замість перевірки передбачуваних взаємозалежностей, вони на основі наявних даних можуть будувати моделі, що дозволяють кількісно оцінити ступінь впливу досліджуваних факторів. Крім того, засоби інтелектуального аналізу дозволяють робити нові гіпотези про характер невідомих, але реально існуючих відношень у даних.

Сучасні технології інтелектуального аналізу опрацьовують інформацію з метою автоматичного пошуку шаблонів, характерних для яких-небудь фрагментів неоднорідних багатомірних даних. На відміну від оперативної аналітичної обробки даних у DataMining тягар формулювання гіпотез і виявлення незвичайних шаблонів перекладено з людини на комп'ютер.

Приклади формулювань задач при використанні методів OLAP і DataMining

OLAP	DataMining
Які середні показники травматизму для людей, що палять і не палять?	Які фактори найкраще передбачають нещасні випадки?
Які середні розміри телефонних рахунків існуючих клієнтів у порівнянні з рахунками колишніх клієнтів (що відмовилися відпослуг телефонної компанії)?	Які характеристики відрізняють клієнтів, що, цілком ймовірно, збираються відмовитися від послуг телефонної компанії?
Яка середня величина щоденної купівлі по вкраденій та невкраденій кредитній картці?	Які схеми купівлі характерні для шахрайства з кредитними картками?

2.3. DataMining і сховища даних

Для успішного проведення всього процесу знаходження нових знань необхідною умовою є наявність сховища даних.

Отже, **сховище даних** - це предметно-орієнтований, інтегрований, прив'язаний до часу, незмінний збір даних для підтримки процесу прийняття управлінських рішень. Предметна орієнтація означає, що дані об'єднані в категорії і зберігаються відповідно до тих областей, що вони описують, а не до їх застосувань. Інтегрованість означає, що дані задовольняють вимогам усього підприємства (у його розвитку), а не єдиної функції бізнесу. Тим самим сховище даних гарантує, що однакові звіти,згенеровані для різних аналітиків, будуть містити однакові результати.

Прив'язка до часу означає, що сховище можна розглядати як сукупність "історичних" даних: можна відновити картину на будь-який момент часу.

Атрибут часу завжди є явно присутнім у структурах сховища даних.

Незмінність означає, що, потрапивши один раз у сховище, дані вже не змінюються на відміну від оперативних систем, де дані зобов'язані бути присутніми тільки в останній версії, оскільки постійно змінюються. У сховище дані лише долучаються.

Для рішення переліченого ряду задач, що неминуче виникають при організації й експлуатації інформаційного сховища, повинно існувати спеціалізоване програмне забезпечення. Сучасні засоби адміністрування сховища даних мають забезпечити ефективну взаємодію з інструментарієм знаходження нового знання.

III. Характеристика процесів і активностей дейтамайнінгу

3.1. Процеси дейтамайнінгу

Засоби сучасної інформаційної технології в останній час уможливили накопичення і зберігання великих обсягів даних про бізнесові процеси. Ці дані можуть знаходитися в корпоративних базах або сховищах даних. Вони містять важливі закономірності і зв’язки між системними характеристиками, які можуть бути використані для прийняття обгрунтованих ділових рішень. Наразі виникла проблема розробки методів відкриття таких закономірностей, про існування яких користувачі можуть і не знати. Проте традиційний аналіз даних передбачує введення даних в стандартні або настроєні користувачем моделі, тобто в будь-якому випадку допускається, що зв'язки між різними показниками добре відомі і можуть бути виражені математично. Однак, в багатьох випадкахзв'язки не можуть бути апріорі відомі. У таких ситуаціях моделювання стає неможливим і тут можна застосовувати дейтамайнінг (*DataMining)* .

Традиційно мали місце два типи статистичних аналізів: *підтверджуючий (confirmatoryanalysis)* і *дослідницький аналіз (exploratoryanalysis).* У підтверджуючому аналізі будь-хто має конкретну гіпотезу і в результаті аналізу або підтверджує, або спростовує її. Однак недоліком підтверджуючого аналізу є недостатня кількість гіпотез у аналітика. За дослідницького аналізу виявляють, підтверджуються чи спростовуються підхожі гіпотези. Тут система, а не користувач, бере ініціативу за аналізу даних.

Здебільшого термін «дейтамайнінг» використовується для описання автоматизованого процесу аналізу даних, в якому система сама бере ініціативу щодо генерування взірців, тобто дейтамайнінг належить до інструментальних засобів дослідницького аналізу.

В загальному вигляді можна виділити три класи процесів дейтамайнінгу: *відкриття, пророче моделювання* і *аналіз аномалій* (див. рис.2). Процеси, що входять в ці класи, досить різноманітні, але в своїй основі мають низку загальних ознак, зокрема: дані, що несуть цінну інформацію, часто глибоко приховані в середині по справжньому великих баз даних, які інколи містять дані за багато років. У деяких випадках ці дані консолідуються в сховища даних; обчислювальне середовище дейтамайнінгу звичайно орієнтовано на архітектуру клієнт/сервер; найдосконаліші нові інструментальні засоби, включаючи продвинуті інструментальні засоби візуалізації, допомагають переміщувати інформаційну "руду", зариту в корпоративних файлах або архівних експортованих даних, щоб отримати корисний результати. Найновіші засоби добування інформації - текстовий дейтамайнінг - також дозволяють досліджувати корисні "непрограмовані" дані (неструктурний текст, який зберігається в різних позиціях, як наприклад, базі даних LotusNotes, текстові файли на Internet або корпоративному Інтранет); реальним добувальником інформації часто є кінцевий користувач, котрий займається практичними обробками даних (DrillDown/Up) та іншими інструментальними засобами запиту, щоб створювати епізотичні запити і одержувати швидкі відповіді, маючи при цьому незначну комп’ютерну підготовку або не володіючи ніякою майстерністю програмування; попадання на інформаційну ”жилу" часто включає виявлення непередбаченого результату і вимагає, щоб кінцеві користувачі думали творчо;інструментальні засоби дейтамайнінгу легко комбінуються з електронними таблицями та іншими інструментальними засобами розробки програмного забезпечення. Тому здобуті в результаті дейтамайнінгу дані можуть бути швидко і легко аналізуватися та оброблюватися; через великі обсяги даних інколи необхідно використовувати паралельне виконання дейтамайнінгу.

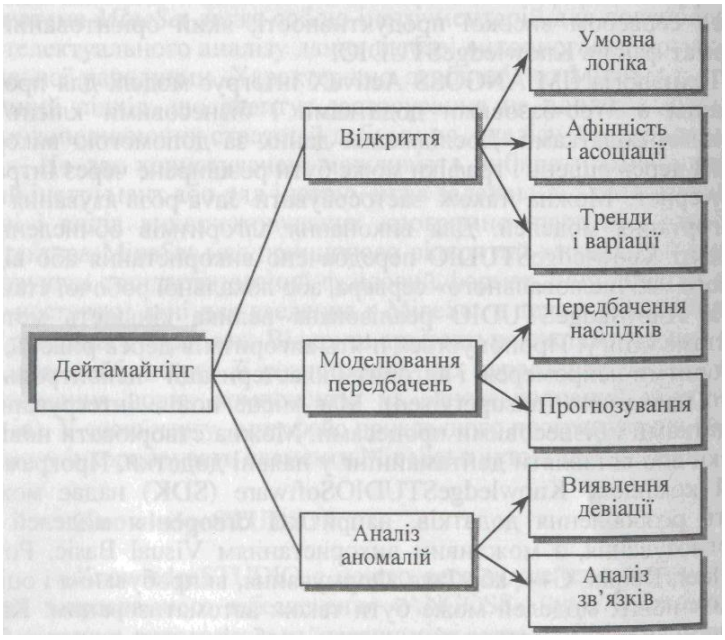


Рисунок 2. Типи процесів дейтамайнінгу

Відомі п'ять загальних типів інформації, що можуть бути одержані засобами дейтамайнінгу:

- *класифікація:* дозволяє робити висновок щодо визначення характеристик конкретної групи (наприклад, споживачі, які були втрачені через дії конкурентів);
- *кластерізація:* ототожнює групи елементів, які використовують спільно зображуючий параметр сигналу даних (кластерізація відрізняється від класифікації, бо не вимагається наперед визначена характеристика);
- *асоціація:* ідентифікуєзв'язки або відношення між подіями, які відбувалися колись (наприклад, зміст кошика відвідань магазину за покупками)
- *упорядковування:* подібно асоціації, крім того, установлюється зв'язок в часовому вимірі (наприклад, повторний візит до супермаркету або фінансове планування виготовлення продукту);
- *прогнозування:* оцінює майбутні значення, засновані на взірцях, здобутих з великого набору даних (наприклад, прогнозування попиту).

3.2. Користувачі та дії дейтамайнінгу

Необхідно відрізняти описані щойно *процеси* від дій дейтамайнінгу, за допомогою яких процеси дейтамайнінгу можуть бути виконані, і *користувачів,* які виконують ці дії. Спершу про користувачів. Дії дейтамайнінгу, зазвичай, виконуються трьома різними типами користувачів: виконавцями (executives), кінцевими користувачами (endusers) і аналітиками (analysts). Усі користувачі, як правило, виконують три види дії дейтамайнінгу всередині корпоративного середовища: епізодичні; стратегічні; безперервні (постійні).

Безперервні і стратегічні дії дейтамайнінгу часто стосуються безпосередньо виконавців і менеджерів, хоч аналітики також можуть у цьому їм допомагати.

3.3. Дерево методів дейтамайнінгу

Технології дейтамайнігу використовують велике число методів, частина з яких запозичена з інструментарію штучного інтелекту, іншу частину складають або класичні статистичні методи, або іноваційні методи, породжені останніми досягненнями інформаційної технології. Верхній рівень дихотономії технологій дейтамайнінгу може бути оснований на тому, чи зберігаються дані після дейтамайнінгу, чи вони дистилюються для подальшого використання.

На рис. 3. показано класифікаційне дерево методів дейтамайнінгу, де відображені основні класи і підкласи методів, причому гілкування можна продовжити, через те, що низка методів, наприклад, кластерний аналіз, нейромережі, дерева рішень включають багато різновидів. Зупинимося на короткому аналізі складових дерева методів дейтамайнінгу, приділяючи більше уваги тим з них, які мало висвітлені в україномовній літературі.

1.2 *3.3.1. Збереження даних (Data Retention).*

1.3 ***В той час, як при дистиляції шаблонів ми аналізуємо дані, виділяємо взірець і потім залишаємо (або забуваємо) дані, то при підході збереження дані зберігаються для зіставляння з взірцем (шаблоном). Коли надходять нові елементи даних, то вони порівнюються з попереднім набором даних.***

Кластерний аналіз – це спосіб групування багатовимірних об’єктів, що базується на зображенні результатів окремих спостережень точками геометричного простору з наступним виділенням груп як “грон” цих точок. Термін “кластерний аналіз” запропонований К. Тріоном в 1939 р. (cluster -грона, скупчення, пучок *англ.).*

Синонімами (хоч з обмовками і не завжди) виступають вирази: *автоматична класифікація, таксономія, розпізнавання без навчання, розпізнавання образів без вчителя, самонавчання* та інш. В дейтамайнінгу кластерний аналізвикористовується в основному для задач таксономії.

Основна мета цього виду аналізу - виділити в початкових багатовимірних даних такі однорідні підмножини, щоб об'єкти всередині груп були схожі у відомому значенні один на одного, а об'єкти з різних груп не схожі. Під “схожими” розуміється близькість об'єктів в багатовимірному просторі ознак, і тоді задача зводиться до виділення в цьому просторі природних скупчень об'єктів, які і вважаються однорідними групами.

В кластерному аналізі використовуються десятки різних алгоритмів і методів (один з таких методів - K-Means реалізований в системі дейтамайнінгу KnowledgeSTUDIO).

Метод “найближчого сусіда” (“nearest neighbor”) - добре відомий приклад підходу, який основується на збереження даних. При цьому набір даних тримається в пам'яті для порівняння з новими елементами даних. Коли презентується новий записдля передбачення, знаходяться “відхилення" між ним і подібними наборами даних, і найбільш подібний (або найближче сусідній) ідентифікується.

Наприклад, якщо розглядається новий споживач банківських послуг, то атрибути пропонованого клієнта порівнюються з всіма існуючими банківськими клієнтами (наприклад, вік і прибуток перспективного порівняно з віком і прибутком існуючих

клієнтів). Потім множина найближчих “сусідів” для перспективного клієнта вибирається на підставі найближчого значення прибутку, віку тощо. При такому підході використовується термін “*K-найближчий сусід" (K-nearest neighbor”)*. Термін означає, що вибираються K верхніх (самих найближчих) сусідів (наприклад, десять верхніх) для розгляду розгляду в перспективі. Наступне найближче порівняння виконується, щоб вибрати серед нових продуктів (наприклад, послуг банку), що найбільш відповідає перспективі на основі продуктів, які використовуються верхніми K сусідами. Добре відомим прикладом програмного продукту з компонентами найближчим сусідом є система *Darwin*™ корпорації TMC.

Звичайно, дуже дорого тримати всі дані, і тому інколи зберігається тільки множина “типових випадків", наприклад, набір із ста “типових клієнтів", як основадля порівняння. Цей підхід часто називається міркування за аналогією (на основі аналогічних випадків).

Міркування за аналогією (case-based reasoning - CBR) або *міркування за прецендентами (аналогічними випадками).* Даний метод має дуже просту ідею – щоб зробити прогноз на майбутнє або вибрати правильне рішення, система CBR находить близькі аналогії в минулому при різних ситуаціях і відбирає ту відповідь, яка за схожими ознаками була правильною. Інструментальні засоби міркування за прецендентами знаходять записи в базі даних, які подібні до описаних записів. Користувачописує, як сильний зв'язок має бути перед тим, щоб пропонувати увазі новий випадок. Ця категорія інструментальних засобів також зветься *міркування на основі пам’яті (memory-based reasoning).*

Програмне забезпечення CBR пробує виміряти “відхилення (дистанцію)", що основується на вимірювання одного запису по відношенню до інших записів і згруповує записи за подібністю. Ця методика мала успіх при аналізуванні зв'язків в текстах вільного формату. Web-сайт www.ai-cbr.org є ресурс штучного інтелекту і області технології міркування за прецендентами. На сайті є великий список посилань на продавців інструментальних засобів міркування за прецендентами і консультантів. Приклади систем, які використовують CBR, включають Katetools (Acknosoft, Франція), PatternRecognitionWorkbench (Unica, США).

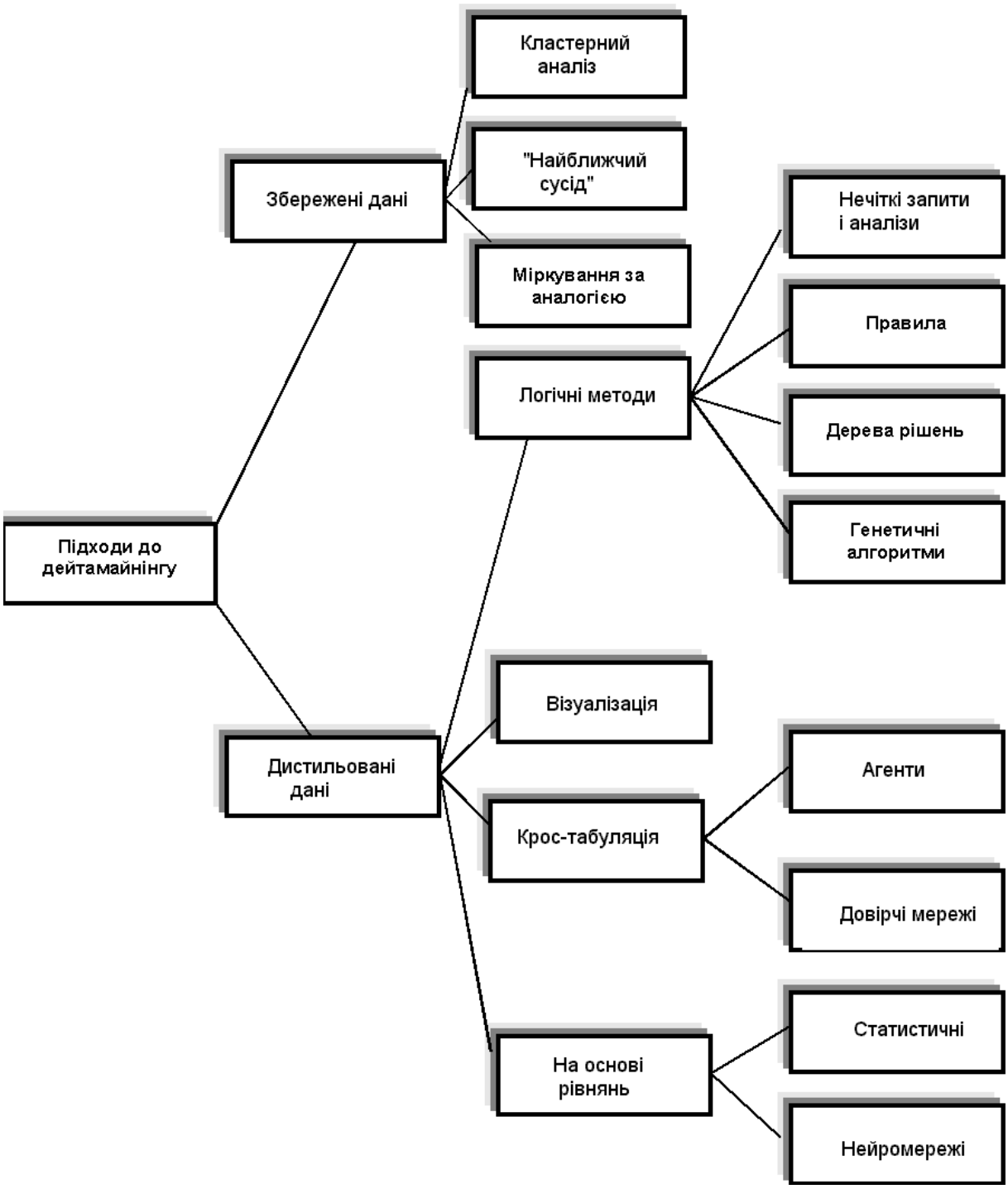


Рисунок 3. Дерево методів дейтамайнінгу

Очевидна ключова проблема цього методу полягає в виборі “типового" клієнта як випадку для порівняння. Інша вразлива проблема полягає в тому, що необхідно обробити бази даних з великим числом нецифровихзначень (наприклад, багато продуктів супермаркету або автомобільні запасні частини).

3.3.2.Дистиляція шаблонів (DataDistilled)

При цій технологіївибираютьвзірець або шаблон з набору даних, потім використовують його з різними намірами. Природно, тут виникають перші два запитання:

Які типи шаблонів можуть бути вибрані і як вони будуть подаватися?Очевидно, шаблон потрібно виразити формально. Ця альтернатива приводить до чотирьох виокремлених підходів: *логічні методи, візуалізація, крос-табуляційні (Cross- tabulational) методи*і *на основі рівнянь (equational).*

1.4 Логічні методи (підходи).***Методи логічного підходу в системах дейтамайнінгу можуть бути розділені на чотири групи: нечіткі запити і аналізи, правила, дерева рішень, генетичні алгоритми.***

Нечіткі запити і аналізи (Fuzzy Query and Analysis). Ця категорія інструментальних засобів дейтамайнінгу основується на відгалуженні математики, що називається нечіткою логікою(fuzzy logic), або логікою невпевненості і розмитості (fuzziness). Вона надає рамкудлявиявленнярозмитості і рангування результатів запитів. Компанія Fuzzy Tech, яка розробляє програмне забезпечення нечітких запитів, має Web-сайт з цікавою і досить повною інформацією про цей інструментальний засіб (http://www.fuzzytech.com/index.htm).

Правила. Правила продукції достатньо відомі, зокрема вони досить часто застосовуються в правило-орієнтованих СППР. Розглянемо основні інші різновиди правил та особливості їх застосування в дейтамайнінгу.

Логічні зв'язки між елементами ділових процесів звичайно частіше за все подаються як правила. Найпростіші типи правил виражаються *умовними* або *афінними (асоціативними)*зв'язками (відношеннями).

Умовне правило є твердження типу: *Якщо умова 1 -- Тоді умова 2.*

Наприклад, в демографічній базі даних може мати місце правило: Якщо “професія=Атлет - Тоді вік< 30” . Тут порівнюється значення полів даної таблиці тобто, використовується представлення виразом "атрибут-значення". В даному прикладі Професія є атрибут, а Атлет - значення.

Афінінна логіка (Affinity logic) є чітка як в термінах мови вираження, так і в термінах структури даних, які використовуються. *Афінний аналіз (або асоціативний аналіз*) є пошук взірців і умов, які описують як різні елементи “групуються разом " або “ставляться разом" в серії подій або транзакцій. Афінне правило має форму: *Коли елемент (позиція) 1- Також елемент (позиція) 2.*

Приклад цього є “Коли фарба, Також пензель фарби”. Проста система афінного аналізу аналізу використовує таблицютранзакцій (наприклад, табл.1), щоб ідентифікувати елементи, що становлять групу елементів транзакцій.

Тут, поле “номер транзакції” використовується, щоб створити групу елементів, в той час як відповідне поле включає об’єкти, які групуються. У цьому прикладі, схожість (affinity) тразакцій 123 і 124 є пара (фарба, пензель фарби). Логічні умови і асоціації часто комбінуються, створюючи гібридну структуру - *прозору(transparent) логіку.*

Правила можуть також працювати добре на багатовимірних даних і OLAPданих, тому що вони можуть мати справу з діапазонами числових даних і їхніх логічних форматів, що дозволяє розглядати шаблони вздовж багатократноїрозмірності.

Правила індукції. Правила індукції -- це процес перегляду набору даних і створення взірців. За допомогою автоматичного дослідження набору даних, як показано на рис. 4,

система індукції формує гіпотези, які приводять до взірців (шаблонів). Процес по суті подібній до того, як людина-аналітик проводить дослідницький аналіз.

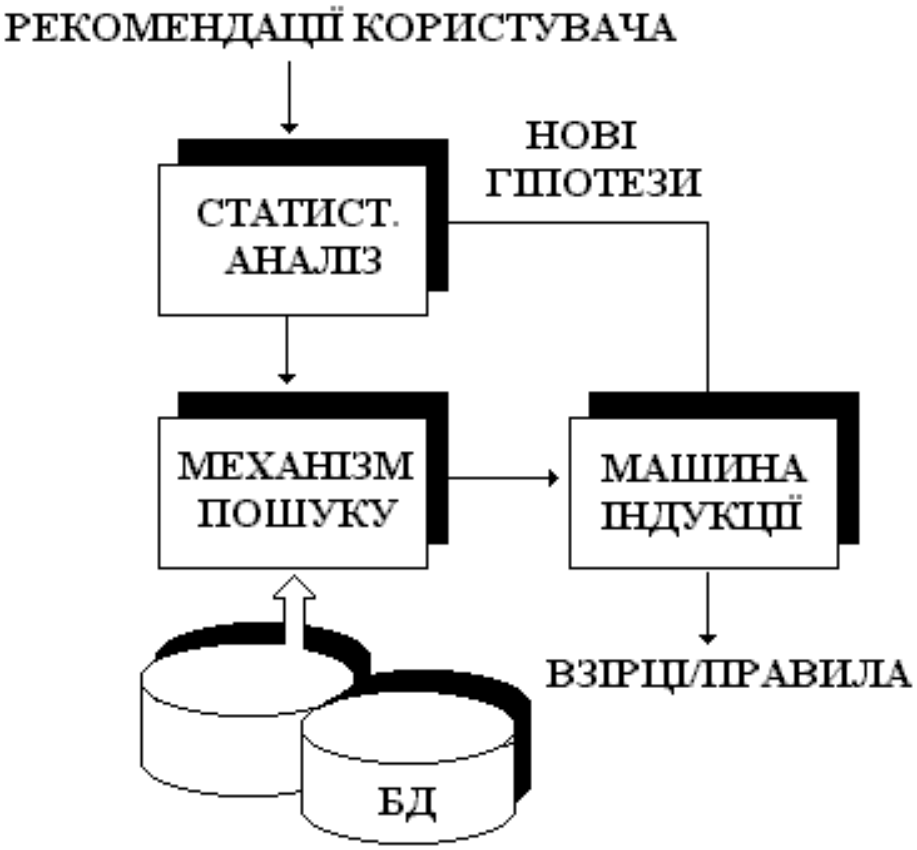


Рисунок 4. Схема використання правил індукції в системі дейтамайнінгу

Таблиця 1

Таблиця транзакцій

Номер транзакції	Елемент
123	Фарба
123	Пензель фарби
123	Цвяхи
124	Фарба
124	Пензель фарби
124	Лісоматеріал

125	….

Потрібно також відрізняти *нечіткі (fuzzy)* і *неточні (inexact)* правила. Неточні правила часто мають “фіксований" коефіцієнт довіри, тобто кожне

правило має специфічне ціле число або процент (як наприклад 70%), який представляє достовірність. Правила індукції може відкрити дуже загальні правила, які мають справу з цифровими і нецифровимиданими. Ці правила можуть комбінуватися з умовними і афінними (спорідненими) твердженнями в гібридних шаблонах (взірцях). Ключове питання полягає в переході від плоских баз даних до даних багатовимірних шаблонівOLAP-систем.

Найвідомішими прибічниками систем генерування правил є компанії Information Discovery, Inc. і Ultragem Corporation, кожна з яких має різний підхід до використання правил. Система Data Mining SuiteTM компанії InformationDiscovery використовує правила індукції (між іншими методами), в той час, як Ultragem покладається на генетичні алгоритми. Data Mining Suite генерує багатовимірні правила від баз даних багататабличнихSQL безпосередньо. Ultragemгенерує правила через генетичні мутації.

Дерева рішень. Дерева рішень (decision trees) є однимз найбільш популярних підходів до рішення задач Data Mining. Дерева рішень виражають просту форму умовної логіки, вони створюють ієрархічну структуру класифікуючих правил типу “ЯКЩО …ТО”. Система дерева рішень просто ділить таблицю для аналізу даних в менші таблиці за допомогою вибору підмножин, основаних на значенняхдля даного атрибута. Зважуючи на те, як ділиться таблиця, ми отримуємо різні алгоритми дерева рішень, як наприклад, CART (ClassificatfonandRegressionTrees), CHAID (ChiSquareAutomaticInteractionDetection), C4.5 , ID3, See5, Sipina та інші.

Для прикладу розглянемо набір записів (табл.2), що характеризує прибутковість збуту продуктів різними фірмами в різних регіонах. Дерево рішень, створене за цією таблицею, показане на рис.5. Для першого гілкування вибраний атрибут Штат, щоб почати виділення розділів розгалудження, потім атрибут - Фірма-виробник. Звичайно, якщо є 100 стовпців в таблиці, питання, які атрибути потрібно вибрати першими, стає критичним.

Таблиця 2

Характеристики збуту продуктів

Фірма-виробник	Штат	Місто	Колірпродукту	Прибуток
Smith	CA	LosAngeles	Голубий	Високий
Smith	AZ	Flagstaff	Зелений	Низький
Adams	NY	NYC	Голубий	Високий
Adams	AZ	Flagstaff	Червоний	Низький
Johnson	NY	NYC	Зелений	Середній
Johnson	CA	LosAngeles	Червоний	Середній

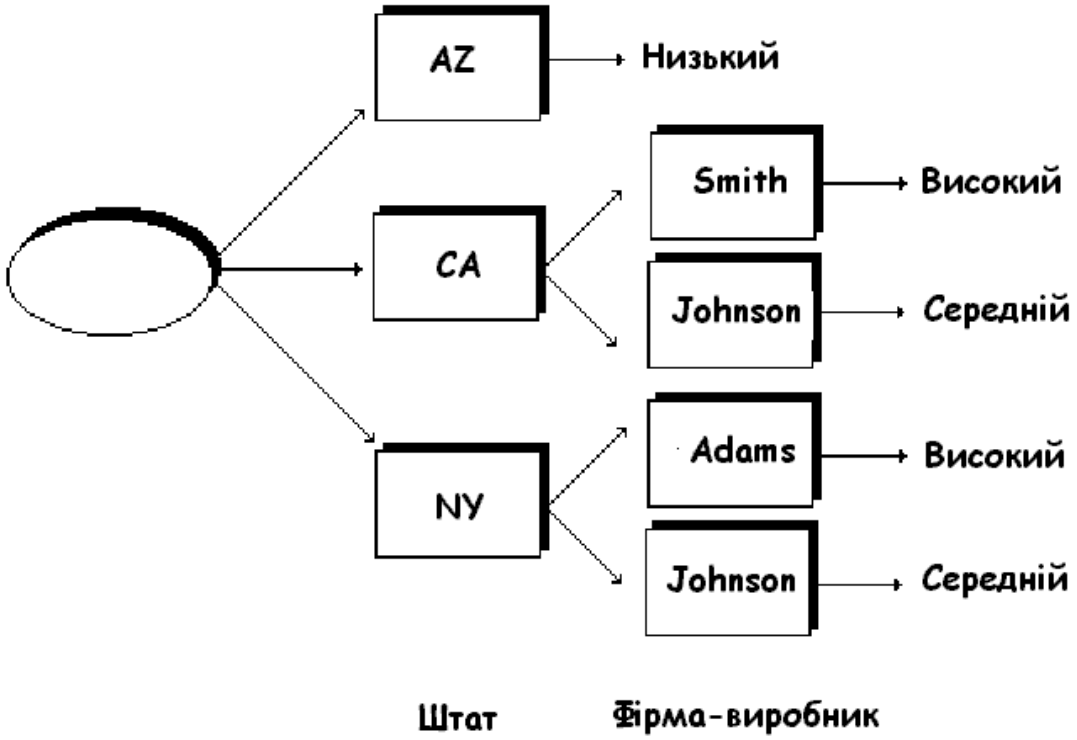


Рисунок 5. Приклад побудови дерева рішень

Фактично, в багатьох випадках, включаючи наведену вище таблицю, немає апріорі найкращих атрибутів, і який би атрибут для дерева рішень спершу не вибраний, завжди буде пошкодження інформації. Наприклад, два факти: (а) “Голубіпродукти мають високий прибуток" та (б) “Арізона має нижчий прибуток" не можуть ніколи бути одержані з дерева рішень, що відповідає таблиці. Ми можемо або отримати факт *(а)* або факт *(б)* з дерева, але не обидва, тому що дерево рішень вибирає один специфічний атрибутдля виділення розділів в кожній стадії. Правила і крос-табуляція, з другого боку, можуть відкрити обидва ці факти.

На даний час досить велике число продавців пропонують пакети програмного забезпечення, які основуються на методах дерева рішень як наприклад, CART. Сюди входять американські корпорації IBM, Pilot Software, Business Objects, Cognos, NeoVista, SAS, Angoss і Integral Solutions (ISL) та інші. Більшість цих систем дозволяє інтерактивне дослідження даних з деревами рішень. Самими поширеними програмними продуктами дейтамайнінгу, що основуються на деревах рішень, є See5/C5.0 (RuleQuest, Австралія), Clementine (Integral Solutions, Великобританія), SIPINA (University ofLyon, Франція), IDIS (Information Discovery, США). В програмному продукті дейтамайнінгу KnowledgeSTUDIO пропонується п'ять алгоритмів дерев рішень. Вартість систем варіюється від 1 до 10 тис. дол.

Генетичні алгоритми. Генетичні алгоритми також генерують правила з наборів даних, але не слідують дослідженням, орієнтованим протоколом правил індукції. Замість цього, вони покладаються на ідею “мутації" (“mutation”), щоб зробити зміни в шаблонах з метою отрамання підходящої форми шаблону завдяки селекції (відбору). Генетична операція кросовера (cross-over) є фактично дуже подібною до дій, пов’язаних з отриманням гібриду рослин і/або тварин. Обмін генетичним матеріалом хромосом (chromosomes) також базується

на тому ж методі. У випадкуправил, матеріал, який обмінюється, є частинашаблону, який правило описує.

Головний фокус в генетичних алгоритмах є комбінування шаблонів з правил, які були відкриті до цього, в той час як в правилах індукції головний фокусобробки є набори даних (детальніше див. розділ IV).

Візуалізаціяданих. Візуалізація даних (Datavisualization) – це інструментальні засоби графічного зображення комплексних зв'язків в багатовимірних даних з різних перспектив або точок зору,представлення даних і узагальнюючої інформації з використанням графіки, анімації, 3-D дисплеїв та інших мультимедійних засобів. Графічне подання інформації засобами візуалізації має на меті забезпечення спостерігача якісним розумінням контексту інформації.

Візуалізація даних відноситься до інструментальних засобів дейтамайнінгу, які трансформують комплексні формули, математичні зв'язки або інформацію сховища даних в діаграми або інші легко зрозумілі моделі. Статистичні інструментальні засоби подібно кластерному аналізу або дереву класифікації і регресії CART часто є компонентами інструментальних засобів візуалізації даних. Аналітики можуть візуалізувати кластери або досліджують бінарне дерево, яке створюється за допомогою класифікування записів.

Крос-табуляція (Cross Tabulation).Крос-табуляція (Cross Tabulation) або перехресна табуляція (перехресні табличні дані) є основна і дуже проста форма аналізу даних, добре відома в статистиці і широко використовувана для створення звітів. Двохвимірна крос-таблиця (cross-tab) подібна до електронної таблиці як щодо заголовків рядків і стовпців, та і щодо атрибутних значень. Комірки (cells) в таблиці являють собою агреговані операції, звичайно ряд атрибутних значень, що зустрічаються разом. Багато крос-таблиць за ефективністю рівноцінні до трьохвимірних ствопчатих гістограм (3D bar graph), що показують сумісно зустрічаючірахунки.

Наприклад, крос-таблицядля рівня прибутку, отримана шляхом аналізу вихідної табл. 2, може мати вигляд, як показано в табл. 3. В таблицю не включені поля “Фірма-виробник” і “Місто”, тому що крос-таблиця буде дуже великою. Однак, слід звернути увагу на той факт, що співпадання рахунків для полів “Голубий” і “Високий” перевищує інші і вказує на сильніший зв'язок.

Маючи справу з малим рядом нецифровихзначень, крос-таблиці є достатньо простими, щоб використовувати і знаходити деякі умовні логічні зв'язки (але не атрибутну логіку, афінну або інші форми логіки). Крос-таблиці звичайно виконуються для чотирьох класів проблем: коли число нецифровихзначень зростає; коли особа має справу з номерними значеннями; коли включаються декілька кон’юнкцій (логічних множень); коли відношення базуються не тільки на підрахунках. Агенти (Agents) і довірчі мережі (belief networks) є варіаціями теми крос-таблиць.

1.4.1.1 Таблиця 3.

Крос-таблиця

	CA	AZ	NY	Голубий	Зелений	Червоний
Прибуток високий	1	0	1	2	0	0

1.5

1.6 *Програмні агенти.* ***Термін "агент" інколи використовується (серед інших), щоб звернутися до крос-таблиць, які графічно показані в мережі і дозволяють тільки кон'юнкції (тобто операції логічного множення "І"). У цьому контексті термін агент є ефективним еквівалентом до терміну "пара: поле-значення ".***

Наприклад, якщо розглядати крос-таблицю (табл. 2), можна визначити 6 "агентів" (КОЛІР: голубий; КОЛІР: Червоний; КОЛІР: Зелений; ШТАТ: CA; ШТАТ: AZ; ШТАТ: NY) для мети (ПРИБУТОК: Високий) і графічно показати їх (рис. 6). Зауважимо, що тутваги 100 і 50 є просто відсотками кількості значень, що приєднуються з метою (тобто, вони представляють рівень впливу, а не ймовірність).

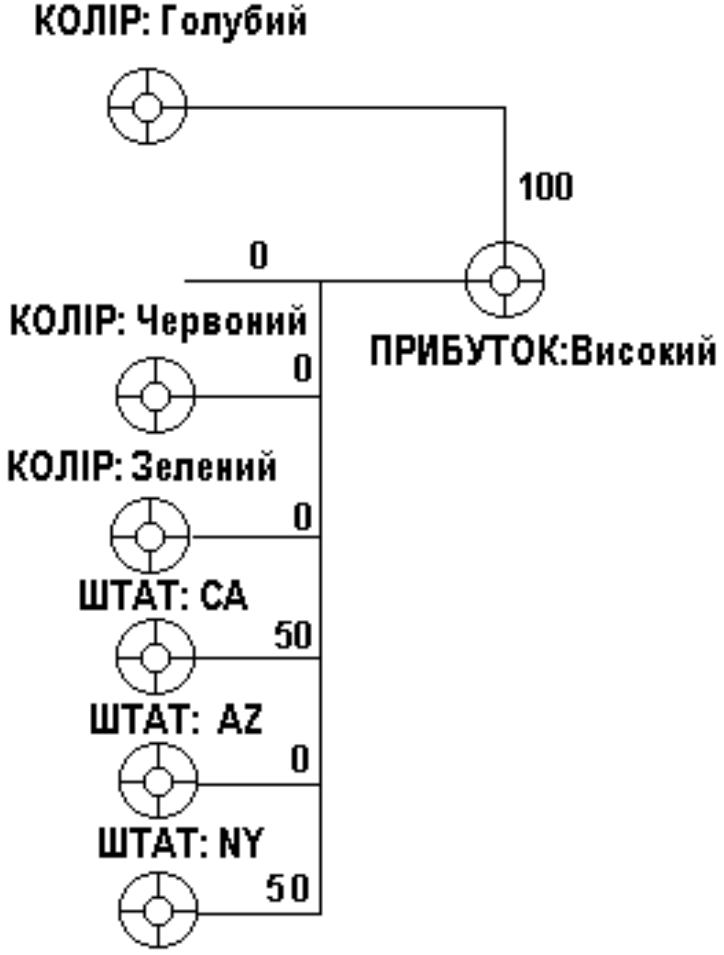
Подібно іншим методам крос-таблиць, коли мають справу з цифровимизначеннями, агенти вимагають розбити числа в фіксовані "блоки", (наприклад, розбити ВІК на три вікові класи: (1-30), (31-60), (61-100)). Звичайно, дані можуть утримувати шаблони, які перекривають будь-які з цих областей (наприклад, область (28-37)) і вони не будуть виявлені агентом. І, якщо діапазони вибрані дуже вузькі, то буде пропущено дуже багато з більших шаблонів. Крім того, ця нездатність мати справу з цифровими проблемами зберігається і для багатовимірних даних. Головним прибічником технологіїагента є корпорація DataMindTM, котра рекомендує використовувати мережі агентів, щоб обчислити "впливи". Фокус уваги в DataMind - аналізданих кінцевого користувача, показуючи при цьому впливи у вигляді мережі агентів. (Детальніше дивись розділ V).

Довірчі мережі. Довірчі мережі (Belief Networks), що інколи називаються *каузальними (причинними) мережами (causal networks*)), також покладаються на співпадання підрахунків (co-occurence counts), але як за графічним виконанням, так і відображенням імовірностей трошки відмінні від агентів.

Довірчі мережі звичайно ілюструються з використанням графічної презентації розподілу ймовірності (отриманого від підрахунків). Довірча мережа є орієнтованим графом (directed graph), що складається з вершин (змінні представлення) і дуг (представлення імовірносної залежності) між вершинами змінних.

Приклад довірчої мережі зображений на рис.7, де показано заради простоти тільки атрибут "колір". Рисунок відображає частину крос-таблиці, наведеної раніше. Кожна вершина містить умовнийрозподіл ймовірності, який описує зв'язок між вершиною і породжуючими елементами (parents) цієї вершини. Граф довірчої мережі ациклічний. Порівнюючи даний рисунок з рис. 6, можна побачити, що дуги в цій схемі означають імовірносну залежність між вершинами, скоріше, ніж "впливи" обчислень крос-таблиці.

Рисунок. 6. Схема впливу агентів на мету



Підходи на основі рівнянь EquationalApproaches). Основний методвиразу взірців (шаблонів) в цих системах є скоріше "поверхнева конструкція",ніж логічний вираз або обрахунки співпадання. Такі системи звичайно використовують множину рівнянь, щоб визначити „поверхню" всередині числового простору, потім вимірюють дистанцію (відхилення) від цієї поверхні.

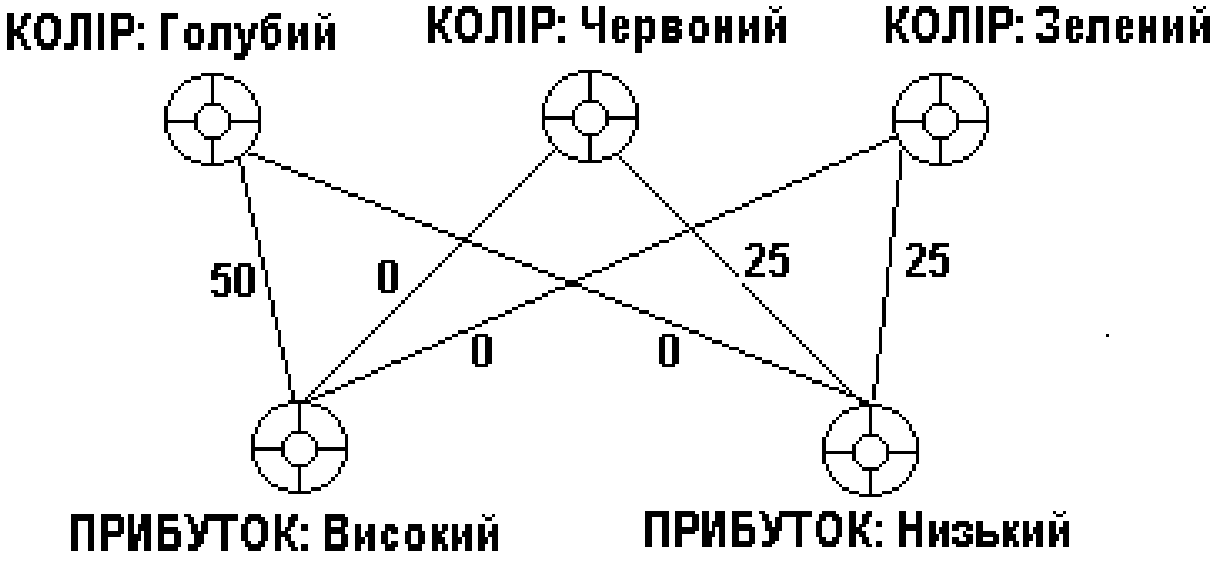


Рисунок. 7 . Приклад довірчої мережі

Підхід дейтамайнінгу на основі рівнянь включає статистичні методи і нейромережі. Оскільки висвітлення питань використання нейромереж в задачах дейтамайнінгу вимагає

досить багато місця, а з іншого боку ці питання опубліковані в низці україномовних видань, то наразі обмежимося декількома коментарями щодо статистичних методів.

Як правило, сучасні статистичні пакети, поряд з традиційними статистичними методами, включають також і елементи дейтамайнінгу. Відомим недоліком статистичних систем є високі вимоги щодо спеціальної підготовки користувачів. Крім того, потужні сучасні статистичні пакети (наприклад, SAS, SPSS, STATGRAPICS, STATISTICA, STADIA) є досить громіздкими для масового застосування в фінансах і бізнесі, до того ж вони досить дорогі – від $1000 до $8000.

Має місце ще принципово суттєвий недолік статистичних пакетів, котрий обмежує застосування їх в дейтамайнінгу. Мова йде про те, що більшість методів, що входять до статистичних пакетів, засновані на статистичній парадигмі, в якій головними фігурантами слугують усереднені характеристики вибірки. А ці характеристики при дослідженні реальних складних життєвих феноменів перетворюються в фіктивні характеристики.

Інші методи дейтамайнінгу. Зображене на рис.3. дерево методів дейтамайнінгу не покриває всієї множини використовуваних на даний час засобів видобування взірців інформації. Коротко зупинимося на деяких із методів, які не відображені на класифікаційній схемі, виділяючи при цьому аспекти впровадження в реально діючі системи дейтамайнінгу.

Нелінійні регресійні методи. Пошук залежності цільових змінних від інших ведеться в формі функцій якогось певного вигляду. Наприклад, в одномуз найбільш вдалих алгоритмів цього типу - методі групового обліку атрибутів (МГОА) залежність шукають в формі поліномів. Очевидно, що цей метод дає більш статистично значущі результати, ніж нейронні мережі. Це робить даний метод досить перспективним для аналізу фінансових і корпоративних даних. Прикладом системи, де реалізовані методи МГОА, є система NeuroShell компанії Ward Systems Group.

Еволюційне програмування. Сьогодні це сама молода і найбільш перспективна гілка data mining, реалізована, зокрема, в системі PolyAnalyst. Суть методу в тому, що гіпотези про вигляд залежності цільової змінної від інших змінних формулюються системою у вигляді програм на деякій внутрішній мові програмування. Процес побудови цих програм будується як еволюція в світі програм (цим метод трохи схожий на генетичні алгоритми). Коли система знаходить програму, досить точно виражаючу шукану залежність, вона починає вносити в неї невеликі модифікації і відбирає серед побудованих таким чином дочірніх програм ті, які підвищують точність. Таким способом система "вирощує" декілька генетичних ліній програм, які конкурують між собою в точності вираження шуканої залежності.

Спеціальний транслюючий модуль системи PolyAnalyst переводитьзнайдену залежність з внутрішньої мови системи на зрозумілу користувачеві мову (математичні формули, таблиці та інше.), роблячи їх легкодоступними. Всі ці заходи приводять до того, що PolyAnalyst показує в деяких задачах аналізу, зокрема, фінансових ринків Росії вельми високі показники.

Алгоритми обмеженого перебору. Алгоритми обмеженого перебору були запропоновані в середині 60-х років М.М. Бонгардом для пошуку логічних закономірностей в даних. Відтоді вони продемонстрували свою ефективність при розв’язуванні безлічі задач в самих різних областях.

Ці алгоритми обчислюють частоти комбінацій простих логічних подій в підгрупах даних. На основі аналізу обчислених частот робиться висновок про корисність тієї або іншої комбінації для встановлення асоціації в даних, для класифікації, прогнозування тощо.Найбільш яскравим сучасним представником цього підходу є система WizWhy підприємства WizSoft. Хоча автор системи Абрам Мейдан не розкриває специфіку алгоритму, покладеного в основу роботи WizWhy, за наслідками ретельного тестування системи були зроблені висновки про наявність тут обмеженого перебору (вивчалися результати, залежності часу їх отримання від числа аналізованих параметрів і ін.).

Автор WizWhy стверджує, що його система виявляє логічні правила типу if-then в даних. Насправді це, звичайно, не так. По-перше, максимальна довжина комбінації в if-then правила в системі WizWhy рівна 6, і, по-друге, з самого початку роботи алгоритму проводиться евристичний пошук простих логічних подій, на яких потім будується весь подальший аналіз. Зрозумівши ці особливості WizWhy, неважко було запропонувати просте тестове завдання, яке система не змогла взагалі вирішити. Інший момент - система видає рішення за прийнятний час тільки для порівняно невеликої розмірності даних.

Проте, система WizWhy є на сьогоднішній день одним з лідерів на ринку продуктів Data Mining. Це не позбавлено підстав. Система постійно демонструє вищі показники при рішенні практичних задач, чим решта всіх алгоритмів. Вартість системи біля $ 4000, кількість продажів - 30000.

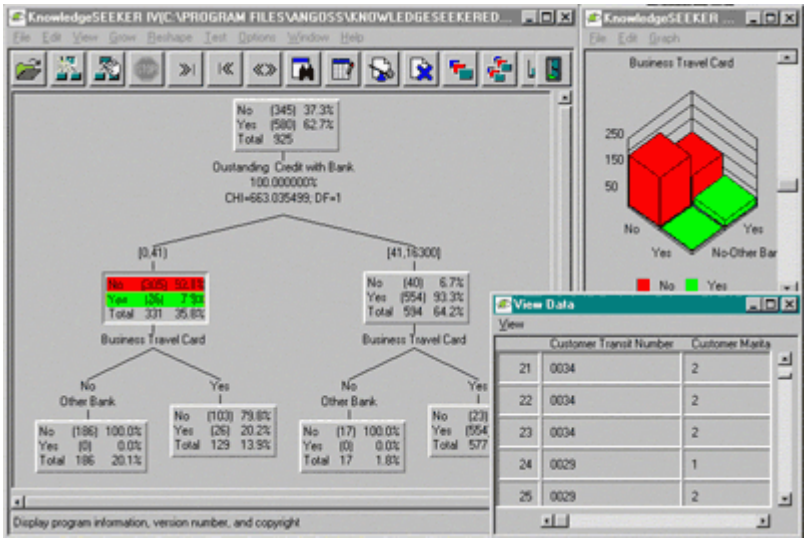


Рисунок 8 . Система WizWhy виявила правила, що пояснюють низьку врожайність деяких сільськогосподарських ділянок

IV. Генетичні алгоритми

4.1. Генетичні успадкування —концептуальна засада генетичних алгоритмів

У загальному значенні *генетичні алгоритми (GeneticAlgorithms)* — це тип алгоритмів, інспірованих механізмами еволюції живої природи, які застосовуються, головно, до задач глобальної оптимізації (зокрема, задач комбінаторної оптимізації) і деякою мірою для дейтамайнінгу, зокрема, для комбінування шаблонів з правил індукції, які були відкриті до цього, навчання нейромереж, пошуку зразків у даних, відкриття шаблонів у тексті тощо. Генетичні алгоритми належать нині до стандартного інструментарію методів дейтамайнінгу.

Ідея генетичних алгоритмів запозичена з живої природи і полягає в машинній організації еволюційного процесу створення, модифікації' і відбору кращих розв'язків, виходячи з того, що в процесі відтворення і модифікації розв'язків кращі з них (подібно До

процесу селекції в рослинництві й тваринництві) можуть дати Ще ліпших «нащадків», тобто нові, прийнятніші варіанти розв'язання задачі. Щоб краще зрозуміти концептуальні засади генетичних алгоритмів, зупинимося на короткому огляді механізмів природного добору і генетичного успадкування, що розглядаються в еволюційній теорії зародження і розвитку життя на нашій планеті. Ця теорія стверджує, що кожний біологічний вид ціле спрямовано розвивається й змінюється так, щоб у найкращий спосіб пристосуватися до навколишнього середовища.

Ключову роль в еволюції відіграє природний добір. Його суть полягає в тому, що найпристосованіші особи краще виживають і приносять більше потомства, ніж менш пристосовані. При цьому завдяки передаванню генетичної інформації, що називається *генетичним успадковуванням,* нащадки успадковують від батьків основні властивості. Проте слід зауважити, що сам по собі природний добір ще не забезпечує розвитку біологічного виду. Дійсно, якщо передбачити, що всі нащадки народжуються приблизно однаковими, то покоління будуть відрізнятися тільки за чисельністю, але не за пристосованістю. Тому дуже важливо вивчити, у який спосіб відбувається успадкування, тобто як властивості нащадка залежать від властивостей батьків.

Основний закон успадкування полягає в тому, що нащадки схожі на своїх батьків. Зокрема, нащадки пристосованіших батьків будуть напевно одними з найпристосованіших у своєму поколінні. Щоб зрозуміти, на чому грунтується ця схожість, нам потрібно буде трохи заглибитися в будову клітини тварин.

Майже в кожній клітині будь-якої тварини є ряд хромосом, що несуть інформацію про цю тварину. Основна частина хромосоми — нитка ДНК (молекула дезоксирибоза Нуклеїнової Кислоти), яка складається з чотирьох видів спеціальних з'єднань (молекул) — нуклеотидів, що чергуються в певній послідовності. Нуклеотиди позначають буквами A, T, C і G, і саме порядок їх розміщеня є кодом усіх генетичних властивостей даного організму. Кажучи точніше, ДНК визначає, які хімічні реакції будуть відбуватися в даній клітині, як вона буде розвиватися і які функції виконуватиме. Отже, генетичний код окремого індивідуума — це просто дуже довгий рядок комбінацій із чотирьох букв A, T, C і G, а сам ген — це відрізок ланцюга ДНК, що відповідає за певну властивість особи, наприклад за колір очей, тип волосся, колір шкіри і т. д. Різні значення генів називають *аллелями.* Вся сукупність генетичних ознак людини кодується за допомогою приблизно 60 тис. генів, які разом містять більше ніж 90 млн нуклеотидів.

Розрізняють два види клітин: статеві (такі, як сперматозоїд і яйцеклітина) і соматичні. У кожній соматичній клітині людини міститься 46 хромосом. Ці 46 хромосом насправді є 23 парами, причому в кожній парі одна з хромосом отримана від батька, а друга — від матері. Парні хромосоми відповідають за такі самі ознаки, наприклад, батьківська хромосома може містити ген чорного кольору очей, а парна їй материнська — ген блакитних очей. Існують певні закони, що керують участю тих або інших генів у розвитку особи. Зокрема, в нашому прикладі нащадок буде чорнооким, оскільки ген блакитних очей є «слабким» (рецесивним) \ пригнічується домінантним геном будь-якого іншого кольору.

У статевих клітинах хромосом тільки 23, і вони непарні. У момент запліднення відбувається злиття чоловічої і жіночої статевих клітин і утворюється клітина зародка, що містить якраз 46 хромосом. Які ж властивості нащадок отримає від батька, а які від матері? Це залежить від того, які саме статеві клітини брали участь у заплідненні. Річ у тім, що

процес вироблення статевих клітин (так званий *мейоз)* в організмі схильний до випадковості, із-за чого нащадки все ж багато чим відрізняються від своїх батьків.

У мейозі, зокрема, відбувається наступне: парні хромосоми соматичної клітини зближуються впритул, потім їх нитки ДНК розриваються в кількох випадкових місцях і хромосоми обмінюються своїми ідентичними ділянками. Цей процес забезпечує появу нових варіантів хромосом І називається *перехрещуванням хромосом* або *кросинговером* (від анг. *crossing-over).* Кожна з хромосом, що знову з'явилася, виявиться потім усередині однієї зі статевих клітин, і її генетична інформація може реалізуватися в нащадках даної особи.

Другим важливим чинником, що впливає на спадковість, ***£ мутації,*** тобто раптові спадкові зміни організму або його частин, ознак, властивостей, які виражаються у зміні деяких дільниць ДНК. Мутації також випадкові і можуть бути викликані різними зовнішніми чинниками, такими, наприклад, як радіоактивне опромінення. Якщо мутація сталася в статевій клітині, то змінений ген може передатися нащадку й виявитися у вигляді спадкової хвороби або в інших нових властивостях нащадка. Вважається, що саме мутації є причиною появи нових біологічних видів, а кросинговер визначає мінливість уже всередині виду (наприклад, генетичні відмінності між людьми).

Важливе місце в еволюційній теорії відводиться поняттю *популяції* як елементарній еволюційній одиниці. ***Популяція*** — це сукупність особин певного виду організмів, які здатні до вільного схрещування, населяють певну територію і деякою мірою ізольовані від сусідніх популяцій. У рамках кожної популяції відбувається процес розмноження — *репродукції (Reproduction),* що являє собою комбінацію послідовностей (strings, хромосом) у опуляци для створення нової послідовності (нащадка). За реродукції нащадок бере частини позицій генів від обох батьків, матиме частину ознак кожного із них. На рис. 9.13а) показана спрощена схема процесу репродукції, де ознаки батьків виражені хромосомою, котра складається з шести генів, що мають дві аллелі, позначені на схемі нулями і одиницями. Нащадок отримав чотири гени від другого батька (перша, друга, третя і шоста позиція) і два від першого (четверта і п'ята позиції).

У генетичних алгоритмах важливе значення мають: формування початкового ряду елементів (популяції), операції кросинговера, що в теорії генетичних алгоритмів частіше називають *кросовером (Cross-over),* і *мутації (Mutation).*

Кросовер -— це комбінування (змішування) хромосом шляхом замін значень генів і утворення нових хромосом на їх місцях. На рис. 9.13 *б)* наведена спрощена схема кросовера, де показано, як шляхом заміни ідентичних ділянок двох батьків отримані два нащадки з новими ознаками.

Мутація — спонтанне перетворення (видозміна) символів (характерних особливостей) у послідовності (хромосомі). На рис. 9 *в)* показано, як у результаті мутації п'ятого гена (значення 0 замінено 1) отримана нова хромосома.

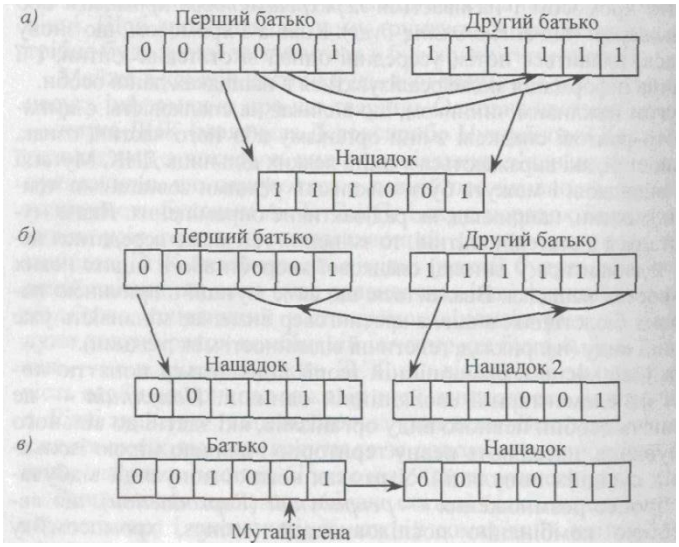


Рисунок 9. Схема генеративних процесів:

а) репродукції осіб популяції; *б)* кросовера осіб популяції; *в)* мутації хромосоми

ці процеси можуть комбінуватися для формування гібридних операторів, операцій репродукції (відтворення) і схрещування з тим, щоб бути спроможними створювати конкуренцію між популяціями.

4.2. Загальна схема генетичних алгоритмів

У концептуальному плані загальна схема генетичних алгоритмів досить проста. Спочатку генерується початкова популяція особин (індивідуумів, хромосом), тобто деякий ряд розв'язків задачі. Як правило, це робиться випадково. Потім необхідно змоделювати розмноження всередині цієї популяції. Для цього випадково підбираються кілька пар індивідуумів, проводиться схрещування хромосом у кожній парі, а отримані нові хромосоми поміщають у популяцію нового покоління. У генетичному алгоритмі зберігається засадний принцип природного добору: чим пристосованіший індивідуум (чим більше відповідне йому значення цільової функції), тим з більшою ймовірністю він буде брати участь у схрещуванні.

Потім моделюються мутації в кількох випадково вибраних особинах нового покоління, тобто змінюються деякі гени. Після цього стара популяція частково або повністю знищується і ми переходимо до розгляду наступного покоління. Популяція наступного покоління в більшості реалізацій генетичних алгоритмів містить стільки ж особин, скільки й початкова, але внаслідок відбору пристосованість (значення цільової функції) у ній в середньому вища. Операція доведення кількості особин поточної популяції до початково визначеної величини називається *редукцією.* Описані процеси відбору, схрещування і мутації повторюються вже Для цієї нової популяції.

У кожному наступному поколінні буде спостерігатися виникнення абсолютно нових розв'язків задачі. Серед них будуть як погані, так і кращі, але завдяки процедурі добору кількість кращих розв'язків буде зростати. Зауважимо, що в природі не буває абсолютних гарантій, і навіть найпристосованіший тигр може загинути від пострілу мисливця, не залишивши потомства. Імітуючи еволюцію в комп'ютері, можна уникати подібних

небажаних подій і завжди зберігати життя кращому з індивідуумів поточно-^{0} покоління. Така методика називається *«стратегією елітизму»,* коли в наступне покоління відбираються особини з найкращими показниками.

Описана послідовність дій за реалізації генетичних алгоритмів може перетворюватися в різні програмні реалізації залежно від типу розв'язуваної задачі і вибраних для цього підходів. Зокрема, в низці випадків може вводитися інша, ніж описана вище, єрархія базових понять, наприклад, кожний індивідуум може характеризуватися низкою хромосом, котрі, у свою чергу, містять різні типи генів. Пояснимо на прикладі.

Нехай розглядається завдання вибору плану вкладення коштів у вибрані наперед *N* інвестиційних проектів, причому потрібно визначити обсяги вкладень коштів у кожний проект так, щоб загальний їх обсяг в усі проекти не перевищував величину *D,* а вибраний критерій ефективності, наприклад рівень рентабельності інвестицій (прибуток на капітал, ROI — ReturnonInvestment), набував максимального значення. Розв'язуючи цю задачу за генетичним алгоритмом, вважатимемо, що кожен індивідуум — це інвестиційний план, який містить *N* хромосом, кожна з яких являє собою вектор із нулів та одиниць — двійковий вираз обсягу вкладень у даний проект. Якщо довжина хромосоми дорівнює вісьмом двійковим розрядам, то потрібне попереднє нормування всіх чисел на інтервалі від 0 до 255 (усього значень 2^{8}). Такі хромосоми називаються безперервними і уможливлюють подання значень довільних числових параметрів.

Мутації безперервних хромосом випадковим способом змінюють у них один біт (ген), впливаючи у такий спосіб на значення параметра. Кросовер також можна здійснювати стандартно, об'єднуючи частини відповідних хромосом (з однаковими номерами) різних індивідуумів. Особливістю цієї задачі є те, що загальний обсяг капіталу, що інвестується, фіксований і дорівнює *D.* Очевидно, що із-за мутацій і схрещувань можна отримувати розв'язки, для реалізації яких потрібний капітал, більший або менший ніж *D.* У генетичному алгоритмі використовується спеціальний механізм аналізування таких розв'язків, що дає змогу враховувати обмеження типу «сумарний капітал = *D*" за підрахунку пристосованості індивідуума. У процесі еволюції особини з суттєвим порушенням зазначених обмежень «вимирають». Унаслідок дії алгоритму отриманий розв'язок за сумарним капіталом може не дорівнювати точно, але бути близьким до заданої величини *D.* У процесі роботи генетичного алгоритму оцінюється значення цільової функції для кожного плану і здійснюється операція редукції для всієї популяції.

Цю саму задачу можна подати і в іншій генетичній інтерпретації, якщо ввести умову, що кожний із інвестиційних проектів або цілком приймається, або відхиляється. Тоді кожний варіант плану (хромосому) можна подати у вигляді послідовності з *N* нулів та одиниць, причому, якщо на цьому місці в хромосомі стоїть одиниця, то це означає, що і-й проект *(i* - 1, 2, ..., ЛО включений у план, а якщо нуль — не включений. Популяція складається із кількох варіантів планів. Визначення допустимості планів і оцінювання їх за вибраними критеріями проводиться аналогічно.

У загальному вигляді стратегію отримання рішень за допомогою генетичних алгоритмів можна реалізувати такими кроками:

1) ініціалізуйте популяцію;

2) виберіть батьків для репродукції і оператори мутації і кросовера;

3) виконайте операції, щоб згенерувати проміжну популяцію індивідуумів і оцінити їхні придатності;

4) виберіть членів популяції для отримання нової генерації

(версії);

5) повторюйте кроки 1—3, поки не буде досягнуте деяке правило зупинки.

На рис. 10 показана узагальнена схема реалізації генетичного алгоритму. До його основних характеристик належать: розмір популяції, оператор кросовера і ймовірність його використання, оператор мутації і її ймовірність, оператор селекції, оператор редукції, правило (критерій) зупинки процесу виконання генетичного алгоритму. Оператори селекції, кросовера, мутації і редукції ще називають *генетичними операторами.*

Критерієм зупинки процесу здійснення генетичного алгоритму може бути одна з трьох подій:

- сформовано задану користувачем кількість поколінь;

- популяція досягла заданої користувачем якості (наприклад, значення якості всіх особин перевищило задану порогову величину);

- досягнутий деякий рівень збіжності. Тобто особини в популяції стали настільки подібними, що дальше їх поліпшення відсувається надзвичайно повільно, і тому продовження здійснення ітерацій генетичного алгоритму стає недоцільним.

Після завершення роботи генетичного алгоритму з кінцевої популяції вибирається та особина, яка дає максимальне (або мінімальне) значення цільової функції і, отже, є результатом здійснення генетичного алгоритму. За рахунок того, що кінцева популяція краща, ніж початкова, отриманий результат являє собою поліпшене рішення.

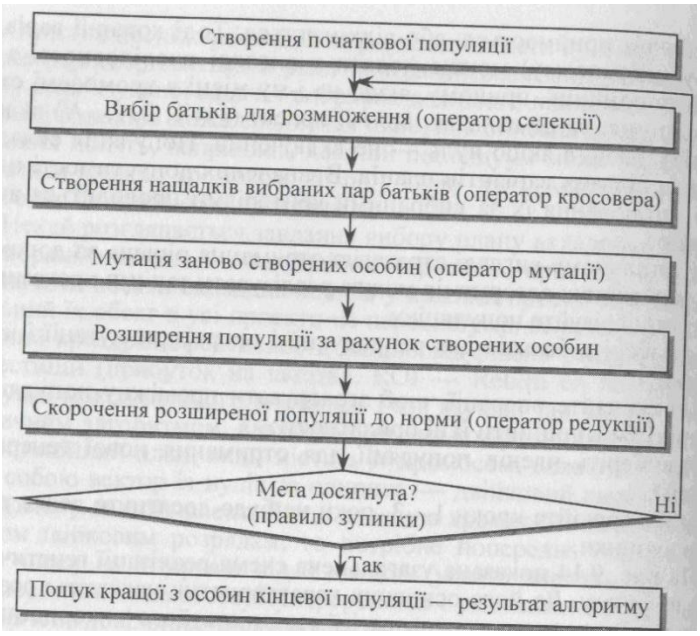


Рисунок 10. Узагальнена схема реалізації генеричного алгоритму

4.3. *Доступне програмне забезпечення генетичних алгоритмів*

Генетичні алгоритми нині можна застосовувати в різних галузях. їх успішно використовують для розв'язування низки великих і економічно важливих задач у бізнесі і в інженерних розробках. З їх допомогою були розроблені промислові проектні рішення, що уможливили багатомільйонну економію витрат. Фінансові компанії широко використовують ці засоби у разі прогнозування розвитку фінансових ринків для управління пакетами цінних паперів. Нарівні з іншими методами генетичні алгоритми, зазвичай, використовуються для оцінювання значень безперервних параметрів моделей великих розмірностей, для розв'язування комбінаторних задач, для задач з оптимізації, що містять одночасно безперервні і дискретні параметри. Іншою галуззю їх застосування є використання в системах добування нових знань із великих баз даних, створення і навчання стохастичних мереж, навчання нейронних мереж, оцінювання параметрів у задачах багатовимірного статистичного аналізу, отримання початкових даних для виконання інших алгоритмів пошуку і оптимізації. Все це зумовило зростання заінтересованості фірм-розробників комерційного програмного забезпечення стосовно генетичних алгоритмів, що в кінцевому результаті привело до появи на ринку багатьох програмних продуктів такого виду.

Незважаючи на те, що розв'язання конкретної оптимізаційної задачі часто потребує побудови генетичного алгоритму з унікальними значеннями параметрів, низка базових властивостей цих алгоритмів залишається постійною за розв'язання абсолютно різних задач. Тому здебільшого для реалізації конкретного генетичного алгоритму не потрібно створювати окремий програмний продукт.

Опишемо кілька прикладів програмного забезпечення, що дає змогу реалізовувати широкий набір генетичних алгоритмів, які можна застосовувати для розв'язування найрізноманітніших задач. Змінними параметрами генетичних алгоритмів у таких додатках, зазвичай, є різні значення ймовірностей, розмір популяції і низка специфічних властивостей алгоритму. Проте реалізація генетичних операторів, як правило, єдина для всіх алгоритмів і прихована від користувача.

Пакет Evolver 4.0 компанії Palisade Corp. Пакет Evolver являє собою доповнення до програми MSExcel версій 5.0 і 7.0. При цьому Excel використовується як засіб опису початкових даних алгоритму і розрахунків у процесі його виконання. У процесі установки Evolver додає в Excel додаткову панель інструментів, яка забезпечує доступ до пакета. Якщо Evolver не запущений для виконання, то панель інструментів не відображається. У разі запуску Evolver додаток Excel запускається автоматично.

Пакет GeneHunter 1.0 компанії WardSystemGroup. Пакет GeneHunter багато чим схожий з пакетом Evolver. Він також є надбудовою над MSExcel версій 5.0 і 7.0 і запускається з меню «Сервіс». Цей пакет русифікований і має низку додаткових настройок для генетичних алгоритмів: включення стратегій елітизму й різноманітності. Поля вікна GeneHunter практично такі самі як і в Evolver. Однак його вікно має низку відмінностей. Для установки параметрів алгоритму служить кнопка «Параметри...». Параметри генетичного алгоритму не зберігаються автоматично з файлом Excel. Для збереження параметрів служить кнопка «Модель», після натиснення на яку з'являється відповідне діалогове вікно.

Пакет Genetic Training Option (GTO) компанії California Scientific Software. Пакет GTO є додатковою утилітою, що поставляється для нейропакета BrainMaker виробництва компанії «California Scientific Software». Він застосовується як для побудови нейронних мереж, так і для поліпшення створеної за допомогою BrainMaker мережі. Але в обох випадках окремо від BrainMaker використовуватися не може.

Генетичні алгоритми складні для створення, але прості в застосуванні — потребують від користувача тільки формалізації задачі й формування початкових даних. Така ситуація багато в чому сприяє розширенню галузей застосування генетичних алгоритмів.

V. Програмні агенти в СППР

5.1. *Призначення і основні характеристики програмних агентів*

Програмні агенти (SoftwareAgents) — це автономні програми, котрі автоматично виконують конкретні завдання з моніторингу комп'ютерних систем і збору інформації в мережах, діють від імені користувача для забезпечення бажаних результатів, так само як людина-агент діє в інтересах замовника, щоб розширити його можливості (звідси й запозичений термін *«агент»).* Термін «програмні агенти» має низку синонімів: *«інтелектуальні (розумні) агенти»* (intelligentagents), *«інтелектуальні інтерфейси»* (intelligentinterfases), *«ноуботи»* (knowbots), *«персональні агенти»* (personalagents), *«програмні роботи»* (softwarerobots), *«аглети»* (aglets) — так називаються програмні агенти в продукті IBMAgletsWorkbench; часто вживаються скорочені терміни: *«агент», «робот»* тощо. Програмні агенти все більше вбудовуються у програмне забезпечення, щоб зробити дії користувачів ефективнішими та результативнішими.

Сучасні програмні агенти, котрі постійно еволюціонують, не тільки проводять спостереження і виконують різні вимірювання, але й розв'язують завдання щодо управління мережами. Зокрема, інтелектуальні агенти здатні автоматизувати численні операції керування мережами, наприклад, вибір оптимального трафіка, контроль за завантаженням, поновлення даних за спотворень у процесі обміну тощо. Крім того, інтелектуальні агенти можуть застосовуватися і в інших галузях: для передавання повідомлень, вибирання інформації, автоматизації ділових процедур (наприклад, агенти покупців і продавців, зустрічаючись у Web, можуть заключати комерційні угоди), у процесах постачання.

Існує багато типів програмних агентів (стаціонарні й мобільні, послужливі та ін.), котрі розроблені з застосуванням результатів досліджень у нейронних мережах, нечіткої логіки, інтерпретації текстів природною мовою, колаборативної фільтрації. Найвідомішими представниками цього виду програмних продуктів є «AgentWare 1.0» фірми «Autonomy».

З метою глибшого розуміння суті поняття «програмний агент» потрібно описати, яким він має бути?

- **Функції.** Агент виконує низку завдань за дорученнями користувача (або іншого агента).

- **Можливості щодо обміну інформацією.** Агент мусить мати можливість обмінюватися інформацією з користувачем (а також іноді з іншими агентами), щоб отримувати від нього інструкції, повідомляти йому про хід і завершення виконання завдань і передавати отримані результати.

• **Автономність.** Агент працює без прямого втручання користувача (наприклад, як фоновий процес у той час, коли комп'ютер виконує інші завдання). Завдання, що виконуються агентом, можуть бути найрізноманітнішими — від щонічного резервного копіювання даних до пошуку (за дорученням користувача) продавця, що пропонує зазначений продукт за найнижчою ціною.

• **Моніторинг.** Щоб мати можливість виконувати свої завдання в автономному режимі, агент має бути здатним контролювати середовище, в якому він діє.

• **Активація.** Щоб мати можливість працювати в автономному режимі, агент має бути здатним впливати на своє робоче середовище за допомогою механізму активізації.

• **«Розумність».** Агент має бути здатним інтерпретувати події, що контролюються ним, щоб ухвалювати належні рішення.

• **Безперервність роботи.** Багато агентів мають виконувати свої завдання постійно.

• **«Індивідуальність».** Деякі агенти можуть мати добре виражений індивідуальний «характер» і «емоційні стани».

• **Адаптивність.** Деякі агенти, грунтуючись на нагромадженому досвіді, автоматично пристосовуються до звичок і переваг своїх користувачів і можуть автоматично пристосовуватися до змін у навколишньому середовищі.

• **Мобільність.** Деякі агенти мають допускати можливість переміщення їх в інші комп'ютери, у тому числі й на системи іншої архітектури та інші платформи.

Програмні агенти значно різняться за їх складністю та функціями. *Як* простий приклад розглядають системи електронної пошти, які містять агентів, що допомагають оперувати великою кількістю повідомлень, котрі деякі люди отримують кожного дня. Агент фільтрує пошту, попереджує про небезпеку, про наявність пріоритетних повідомлень, перенаправляє повідомлення у разі відсутності користувача і відкидає повідомлення за його вказівками. Іншим, складнішим прикладом застосування агентів є особистий туристичний агент, який координує особисті туристичні плани, включаючи створення плану, наймання автомобіля, готелю і замовлення в ресторані.

Програмні агенти, що самі навчаються, спостерігають за тим, як користувач реально використовує програму, і пропонують виконувати це самі автоматично. Наприклад, якщо користувач читає всі повідомлення спершу від керівника (шефа), то агент міг би за-пропонувати помістити всі його повідомлення на початку списку.

Агент може керуватися часом, подією або алгоритмом чи деякою їх комбінацією. Наприклад, агент міг би бути запрограмованим так, щоб попередити користувача про небезпеку, коли ціна акцій компанії «Coca-Cola» перевищить S60 за акцію (приклад *керування за подією).* Або в кінці дня (приклад *керування за часом),* агент міг би перевіряти, чи не нижче зазначеного рівень запасів, щоб здійснювати нову закупівлю (приклад *керування за алгоритмом).* Ці приклади показують, що агенти можуть залишати програми і/або клієнтські місця, щоб виконувати свої завдання, і можуть навіть взаємодіяти з іншими агентами для пошуку інформації зовні.

Агенти можуть бути або попереджуючими або керованими користувачами. Попереджуючі агенти постійно переглядають середовище з метою пошуків певної інформації. Наприклад, агент може постійно шукати нові історії про клієнтів на електронних службах новин і надсилати ті, що знайшов до виконавчого менеджера через поштову систему компанії. Для порівняння, керований користувачем агент має шукати історії тільки тоді, коли йому дана на це вказівка.

5.2. Програмні агенти у СППР та BIC

В інформаційних системах, зокрема в СППР, програмні агенти можуть застосовуватися для пошуку в базах даних потрібної для користувача інформації, для її аналізу з метою виявлення тенденцій або моделей, які ОПР міг пропустити чи не помітити. Крім того, інтелектуальні агенти можуть швидше діставати інформацію для ідентифікації незвичайних ситуацій, що дасть змогу користувачеві негайно на них зреагувати. Наприклад, програмний агент DSSAgent компанії «MicroStrategy», відкритий екран якого показаний на рис. 11 (на рисунку з метою глибшого розуміння деякі позиції дано українською мовою), знаходить, інформацію в сховищі даних, підсумовує та аналізує її для ОПР. Активізована функція ФІЛЬТРИ: результати пошуку подані у вигляді стовпчикової діаграми, таблиці й карти. Система попереджує користувача, що в Новій Англії фактичний обсяг продажу на 9 % нижчий від запланованого, хоча по окремо виділеному продукту LY ситуація краща.

Рисунок 11. Вікно програмного агента DSSAgent (Реальна реляційна OLAP)

Користувачі інформаційних систем можуть знаходити інформацію з використанням розумних агентів на конкретну дату або здійснювати пошук за подіями. Наприклад, ОПР може виконувати регулярну перевірку браку або пропуску звітних даних для висвітлення через індикатори тих проблем, на які необхідно звернути увагу. ОПР за допомогою розумного агента може також знаходити інформацію про просування товарів вище запланованого рівня або після досягнення конкретним показником точно визначеного рівня.

Результати роботи агентів можуть поєднуватися з іншими блоками інформаційної системи. Наприклад, агент може знайти інформацію, яку слід автоматично імпортувати до

задачі з формування прогнозу для визначення майбутнього попиту. За бажання інший агент може бути переключений на результати розв'язання деяких задач для автоматичного Інформування управлінського персоналу.

Програмні агенти все більше включаються в програмне забезпечення додатків. Ключ до успіху для продавців —- переконатися, що користувачі легко можуть описати правила для агентів, яких ті мають дотримуватися. Широкі можливості програмних агентів закладені в їхній здатності брати участь у керуванні корпоративними знаннями. Виконавчі менеджери та інші працівники, які використовують знання, не страждають від браку інформації; скоріше, вони «плавають» у ній.їх проблема полягає у впорядкуванні існуючої інформації та виявленні тієї, яка є для них найважливішою. СППР, зокрема виконавчі інформаційні системи (ВІС), можуть значно допомогти їм у цьому завдяки процесам збирання, оброблення, структурування і подання інформації.

Програмні агенти застосовуються у ВІС, щоб допомогти менеджерам охопити великі масиви даних і інформації, які зберігаються в електронному вигляді. У контексті підтримки менеджерів та виконавців агента використовують як фоновий, базовий процес, який застосовує низку правил виявлення для того, щоб автоматично спостерігати за певною сукупністю даних, постійних або змінних, з метою пошуку зразка, що відповідає цим правилам, і інформувати зацікавлених користувачів, коли такі зразки виникають. Цей вид агентів також називають *«програмним агентом-фільтром»* чи *«програмним агентом-спостерігачем».*

Компанія «Comshare», є провідним продавцем продуктів СППР/ВІС, і є першою компанією, яка інтегрувала програмних агентів у свої продукти. «Виявляти і попереджувати» («DetectandAlert») — так коротко ця компанія називає можливості, що забезпечуються програмними агентами. Як зазначено у цій назві, агенти виявляють спеціальні умови і потім попереджують користувачів.

Програмні агенти є цінними інструментальними засобами для допомоги користувачам систем підтримки прийняття рішень та виконавчих інформаційних систем в аналізі великих баз даних на безперервній основі. Ринок продуктів програмних агентів постійно зростає, а виторг від їх продажу становить щорічно понад 2,6 млрд дол., незважаючи на відносно низьку реалізаційну ціну (в межах $50 за один програмний агент).

VI. *Доступне програмне забезпечення дейтамайнінгу*

Як уже зазначалося, нині на ринку програмних продуктів пропонуються десятки готових до використання систем дейтамайнінгу, причому деякі з них орієнтовані на широке охоплення технологічних засобів дейтамайнінгу, а інші грунтуються на специфічних методах (нейромережах, деревах рішень тощо). Охарактеризуємо найновіші системи ДМ з низкою різних підходів і методів дейтамайнінгу —MineSet, KnowlengeSTUDIO,PolyAnalyst. Вузькоорієнтовані на специфічні способи добування даних системи ДМ будуть згадуватися за ідентифікації найпоширеніших методів дейтамайнінгу в наступних параграфах даного розділу.

MineSet — візуальний інструмент аналітика

Компанія «SiliconGraphics» розробила систему дейтамайнінгу— *MineSet,* яка відрізняється специфічними особливостями як на концептуальному, так і на технологічному рівнях. Акцент при цьому робиться на унікальну процедуру візуальної інтерпретації складних взаємозв'язків у багатовимірних даних.

Система *MineSet*являє собою інструментарій для поглибленого інтелектуального аналізу даних на базі використання потужної візуальної парадигми. Характерною особливістю MineSet є комплексний підхід, що адаптує застосування не однієї, а кількох взаємодоповнюючих стратегій добування, аналізу й інтерпретації даних. Це дає користувачеві можливість вибирати найвідповідніший інструмент або ряд інструментів залежно від розв'язуваної задачі і видів використовуваних програмно-апаратних засобів. Архітектура MineSet має принципово відкритий характер — використовуючи стандартизований файловий формат, інші додатки можуть постачати дані для введення в MineSet, а також використовувати результати її роботи. Відкрита архітектура системи — це і основа для майбутнього її розширення, що передбачає можливість вбудовування нових компонентів на основі концепції інтеграції (plug-in). У свою чергу, інтерфейс прикладного програмування (API) дає змогу інкорпорувати елементи MineSet в автономні додатки.

KnowledgeSTUDIO

KnowledgeSTUDIO є новою версією дейтамайнінгу корпорації з програмного забезпечення «ANGOSS» (http://www.angoss. com/). Система впроваджує найрозвинутіші методи ДМ у корпоративне середовище з тим, щоб підприємства могли досягати максимальної вигоди від своїх інвестицій у дані. Вона забезпечує високу продуктивність користувачів щодо розв'язання ділових проблем без суттєвих зусиль на навчання, як це, наприклад, потрібно для освоєння статистичного програмного забезпечення. Крім того, це також потужний інструментальний засіб для аналітиків.

KnowledgeSTUDIO сумісна з основними статистичними пакетами програм. Наприклад, ця система не тільки читає і записує файли даних, але також і генерує коди статистичного пакета SAS. Із такими властивостями стосовно статистики моделювальники можуть швидко й легко адаптувати успадковані статистичні

аналізи.

Система *KnowledgeSTUDIO*тісно інтегрується зі сховищами і вітринами даних. У такому разі дані можуть добуватися в режимі *In-placeMining,* тобто коли вони залишаються у вітрині або сховищі даних «на місці», автоматично використовуючи для цього «хвилі запитів», тобто серію тверджень SQL. Завдяки тому, що дані отримуються безпосередньо від джерела, дублювання не потребується. Альтернативно, з метою оптимізації ДМ дані можна вибирати з їх форматом зберігання, а потім дейтамайнінг виконується сервером високої продуктивності, який орієнтований на формат файлів KnowledgeSTUDIO.

Технологія ДМ ANGOSSActiveX інтегрує моделі для прогнозування з Web-базовими додатками і бізнесовими клієнт/серверними додатками. Дослідження даних за допомогою використання дерев рішень і графіки може бути розширене через Інтранет і Інтернет, Можна також застосовувати Java-розв'язування для розгортання моделей. Для виконання алгоритмів обчислення в проекті KnowledgeSTUDIO передбачено використання або віддаленого «обчислювального» сервера, або локальної робочої станції. У KnowledgeSTUDIO реалізована велика кількість методів дейтамайнінгу. Пропонується п'ять алгоритмів дерев

рішень, три алгоритми нейромереж і алгоритм кластеризації «неконтрольованого навчання» (unsupervised). Має місце повне інтегрування з додатками і бізнесовими процесами. Можна створювати нові додатки або вставляти дейтамайнінг у наявні додатки. Програмований комплекс KnowledgeSTUDIOSoftware (SDK) надає можливість розроблення додатків, наприклад створення моделей для прогнозування, з можливим використанням VisualBasic, PowerBuilder, Delphi, C++, або Java. Формування, випробування і оцінювання нових моделей може бути також автоматизованим. KnowledgeSTUDIO забезпечує різні шляхи, щоб візуально виразити і дослідити у великих базах даних зразки прихованих закономірностей.

PolyAnalyst

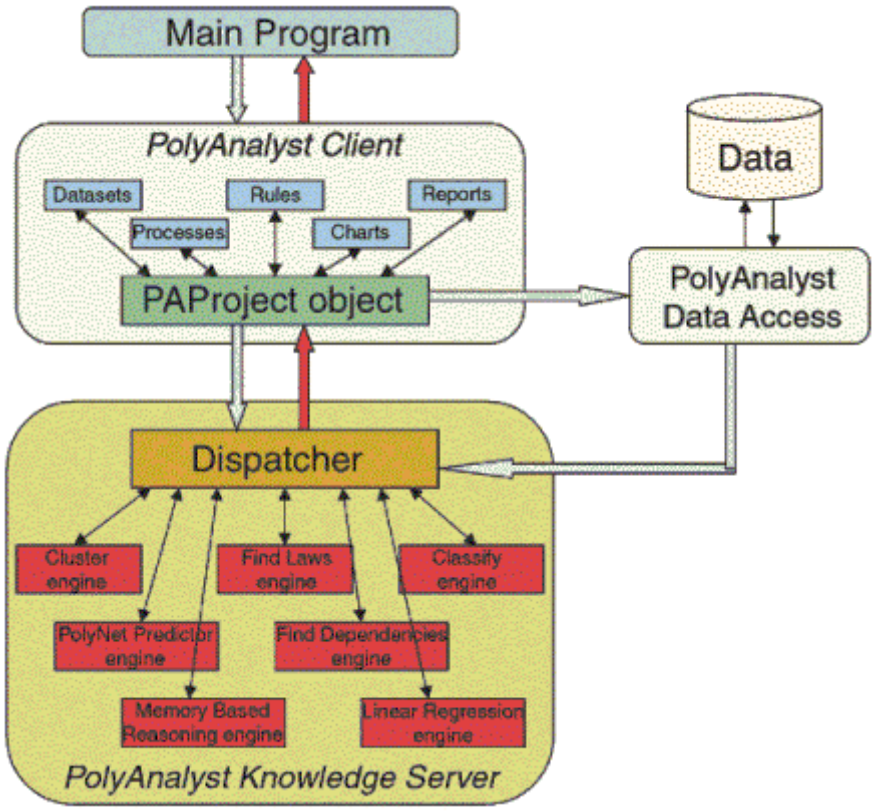
Компанія «Мегап'ютер» виробляє і пропонує на ринку сімейство продуктів для дейтамайнінгу — *PolyAnalyst.* Система PolyAnalyst призначена для автоматичного і напівавтоматичного аналізу числових баз даних і витягання з сирих даних практично корисних знань. PolyAnalyst знаходить багатофакторні залежності між змінними в базі даних, автоматично будує і тестує багатовимірні нелінійні моделі, що виражають знайдені залежності, виводить класифікаційні правила по повчальних прикладах, знаходить в даних багатовимірні кластери, будує алгоритми рішень.PolyAnalyst використовується в більш ніж 20 країнах світу для вирішення завдань з різних областей людської діяльності: бізнесу, фінансів, науки, медицини. В даний час - це одна з наймогутніших і в той же час доступних в ціновому відношенні комерційних систем для Data mining в світі. Основу PolyAnalyst складають так звані Exploration engines або Машини досліджень - математичні модулі, засновані на різних DM алгоритмах, і призначені для автоматичного аналізу даних. Компанія Megaputer Intelligence веде інтенсивні дослідження, направлені на розширення аналітичних функцій системи PolyAnalyst, розробку нових DM алгоритмів і нових математичних модулів системи.

В даний час PolyAnalyst є однією з наймогутніших систем DataMining в світі, реалізованих для Intel платформ і операційних систем MicrosoftWindows. Аналогічні системи DataMining таких провідних виробників, як IBM (IntelligentMiner, DataMiner), SiliconGraphics (SGIMiner), IntegralSolutions (Clementine), SASInstitute (SAS) працюють на середніх і великих машинах і коштують десятки і навіть сотні тисяч доларів. Завдяки унікальній технології "Еволюційного програмування", і іншим інноваційним математичним алгоритмам, PolyAnalyst поєднує в собі високу продуктивність "великих систем" з низькою вартістю, властивою програмам для Windows. PolyAnalyst - один з небагатьох комерційних продуктів, в якому реалізовані не тільки методи аналізу числових даних, але і алгоритми TextMining, - аналізу текстової інформації. Протягом своєї більш, ніж 10-річній історії, пакет безперервно розвивається, компанія-виробник додає нову функціональність, нові математичні модулі, планується портация системи на Unix платформи. PolyAnalyst набув широкого поширення в світі. Більше 500 інсталяцій в 20 країнах світу, серед користувачів системи значний список складають найбільші світові корпорації: Boeing, 3M, ChaseManhattanBank, Dupont, Siemens та інші. PolyAnalyst - універсальна система DataMining, вона з успіхом застосовується в різних областях: у рішенні бізнес-задач (directmarketing, cross-selling, customerretention), в соціологічних дослідженнях, в прикладних наукових і інженерних завданнях, в банківській справі, в страхуванні і медицині.

Архітектура системи

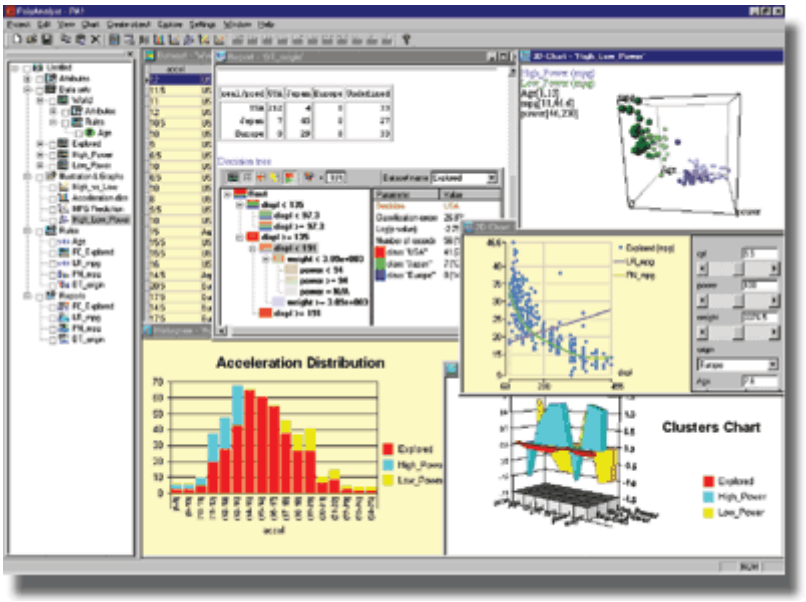
За своєю природою, PolyAnalyst є клієнт/серверним додатком. Користувач працює з клієнтською програмою PolyAnalyst Workplace. Математичні модулі виділені в серверну частину - PolyAnalyst Knowledge Server. Така архітектура надає природну можливість для масштабування системи: від однокористувацького варіанту до корпоративного рішення з декількома серверами. PolyAnalyst написаний на мові C++ з використанням специфікації

Microsoft's COM (ACTIVEX). Ця специфікація встановлює стандарт комунікації між програмними компонентами. Математичні модулі (Exploration Engines) і багато інших компонентів PolyAnalyst виділені в окремі динамічні бібліотеки і доступні з інших додатків. Це дає можливість інтегрувати математику PolyAnalyst в ті, що існують ІС, наприклад, в CRM або ERP системи.



PolyAnalyst Workplace - лабораторія аналітика

Workplace - це клієнтська частина програми, є повнофункціональним середовищем для аналізу даних. Розвинені можливості маніпулювання з даними, багата графіка для представлення даних і візуалізації результатів, майстри створення об'єктів, наскрізний логічний зв'язок між об'єктами, мова символьних правил, інтуїтивне управління через drop-down і pop-up меню, детальна контекстна довідка - ось тільки декілька основних рис призначеного для користувача інтерфейсу програми.



Одиницею Data Mining дослідження в PolyAnalyst є "проект". Проект об'єднує в собі всі об'єкти дослідження, дерево проекту, графіки, правила, звіти ітд. Проект зберігається у файлі внутрішнього формату системи. Звіти досліджень представляються у форматі HTML і доступні через інтернет.

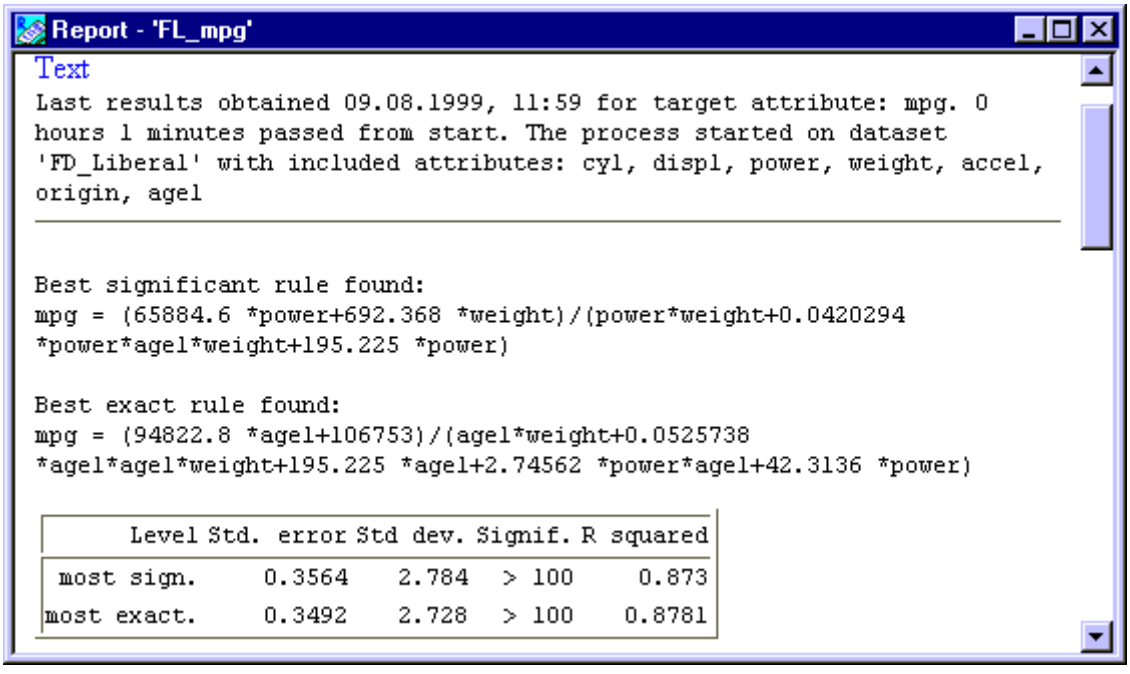
Аналітичний інструментарій PolyAnalyst

Версія PolyAnalyst 4.6 включає 18 математичних модулів, заснованих на різних алгоритмах Data і Text Mining. Більшість з цих алгоритмів є Know-How компанії Мегапьютер і не мають аналогів в інших системах. Алгоритми аналізу даних можна об'єднати в групи по їх функціональному призначенню: моделювання, прогнозування, кластеризація, класифікація, текстовий аналіз. Нижче наводиться коротка характеристика математичних алгоритмів PolyAnalyst.

Модулі для побудови числових моделей і прогнозу числових змінних

Модуль Find Laws (FL) - будівник моделей

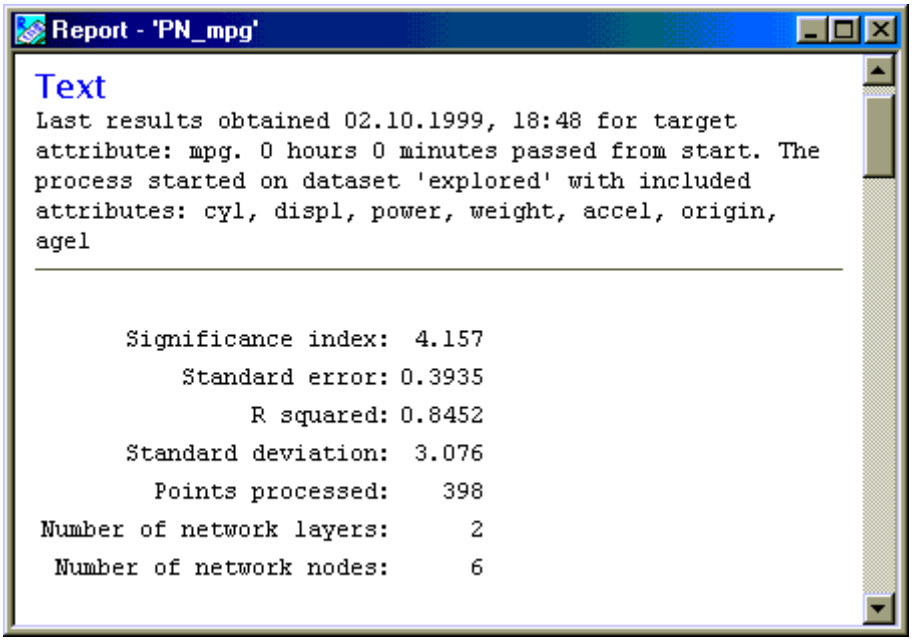
Модуль FL - це серце всієї системи. Алгоритм призначений для автоматичного знаходження в даних нелінійних залежностей (вид яких не задається користувачем) і представлення результатів у вигляді математичних формул, що включають і блоки умов. Здатність модуля FL автоматично будувати велике різноманіття математичних конструкцій робить його унікальним інструментом пошуку знання в символьному вигляді. Алгоритм заснований на технології еволюційного або як її ще називають генетичного програмування, вперше реалізованою в комерційних програмах компанією Мегапьютер.



PolyNet Predictor (PN) - поліноміальна нейронна мережа

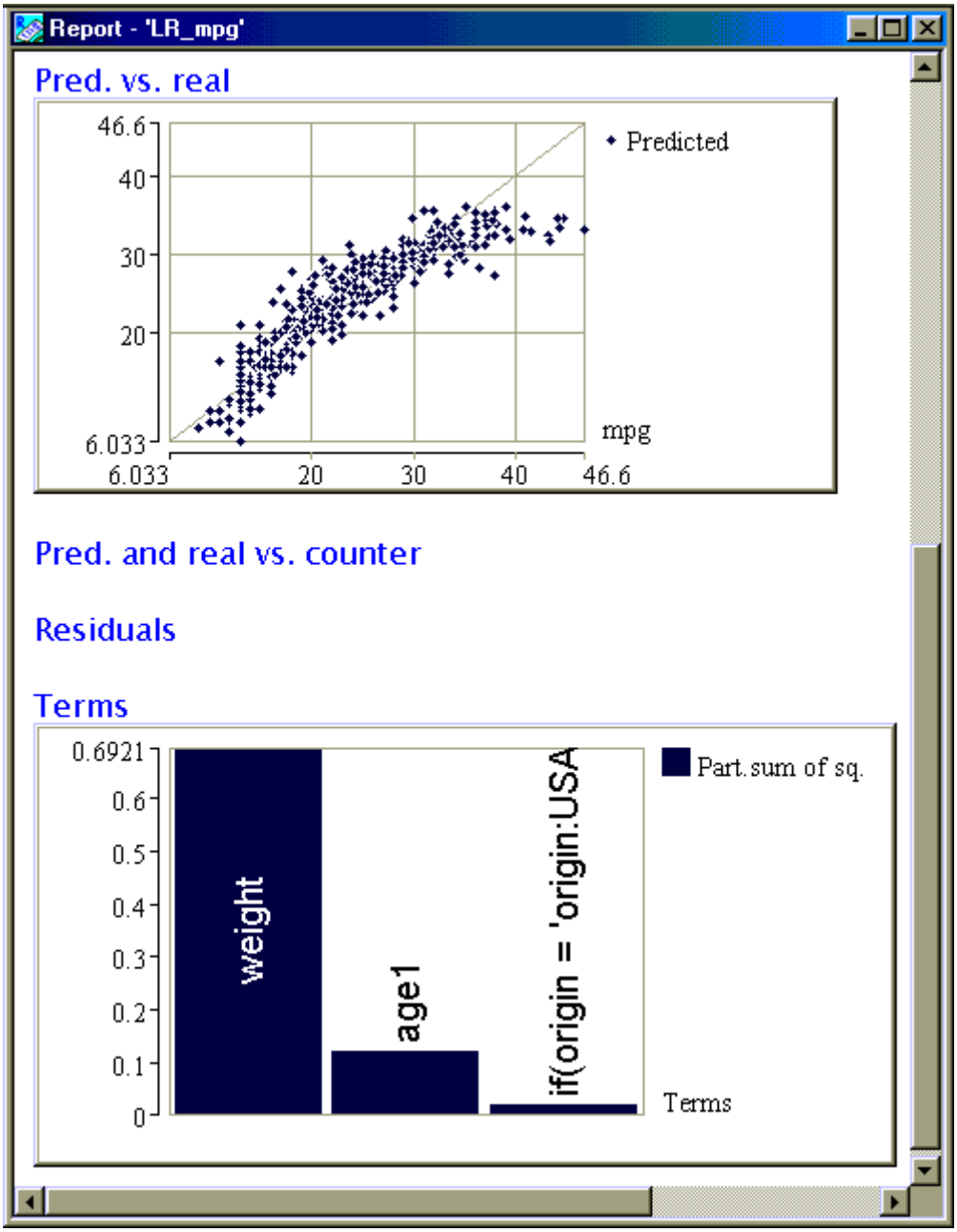
Робота цього алгоритму заснована на побудові ієрархічної структури, подібній нейронній мережі. При цьому складність цієї мережевої структури та інші її параметри підбираються динамічно на основі властивостей аналізованих даних. Якщо створювана мережева структура не є дуже складною, то може бути побудований еквівалентний нею вираз на мові символічних правил системи. Якщо ж мережа дуже велика, то правило не може бути показане, проте його можна обчислити, або іншими словами застосувати до початкових або

нових даних для побудови прогнозу.Даний алгоритм надзвичайно ефективний в інженерних і наукових завданнях, коли потрібно побудувати надійний прогноз для числової змінної.



Stepwise Linear Regression (LR) - покрокова багатопараметрична лінійна регресія

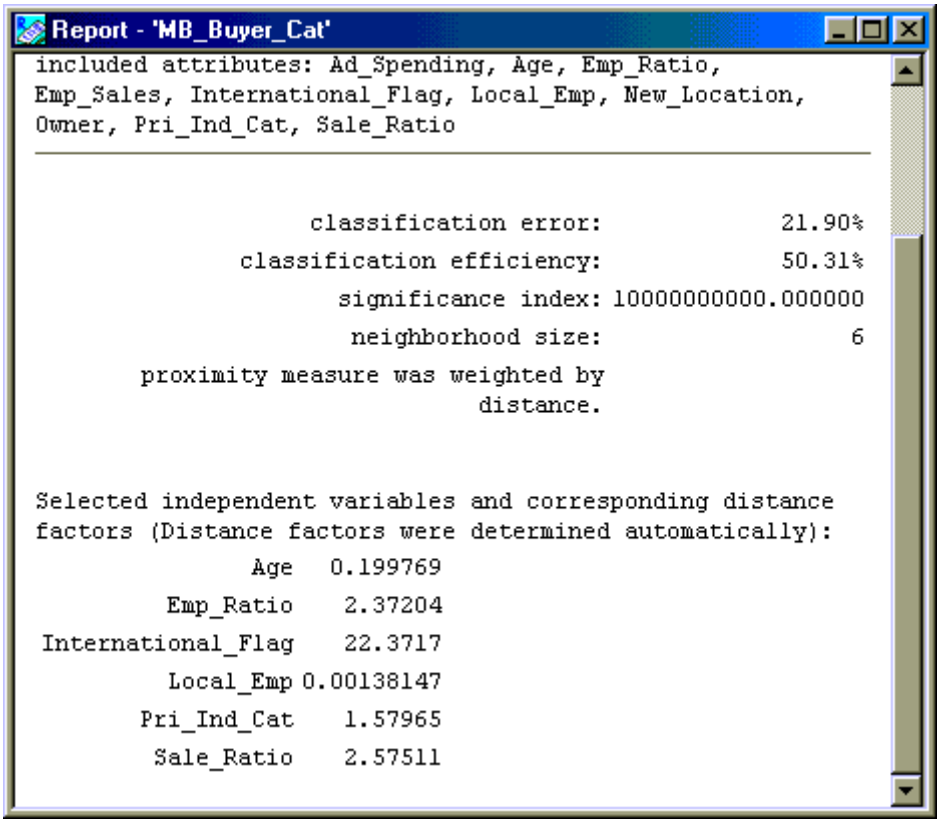
Лінійна регресія як широко поширений метод статистичного дослідження, включена в багато статистичних пакетів і електронні таблиці. Проте, реалізація цього модуля в системі PolyAnalyst має свої особливості, а саме, автоматичний вибір найбільш значущих незалежних змінних і ретельна оцінка статистичної значущості результатів. Потрібно відмітити, що в даному випадку значущість відрізняється від значущості одиничної регресійної моделі, оскільки протягом одного запуску даного обчислювального процесу може бути перевірене велике число регресійних моделей. Алгоритм працює дуже швидко і застосовується для побудови лінійних моделей на змішаних типах даних.



Memory based reasoning (MR) - метод "найближчихсусідів"

Усистемі PolyAnalyst використовуєтьсямодифікаціявідомогоалгоритму "методнайближчихсусідів". Ідеяметодудужепроста - дляпрогнозузначенняцільовоїзмінноїдляданогозапису, вповчальнійтаблицізісторичнимиданими, знаходяться "схожі" записи, дляякихвідомізначенняцільовоїзмінної, іобчислюєтьсясереднєзцихзначень, якеівважаєтьсяпрогнозом.

Напрактиціреалізаціяцієїідеїзустрічаєтьсязтрьомаосновнимитруднощами: 1. щовважатиміроюблизькостізаписів?, 2. скількизаписівбратидляусереднювання?, 3. якийметодусереднюваннявикористовувати (звичайнеабозваженеусереднювання?).У системі PolyAnalyst оптимізація цих параметрів проводиться на основі генетичних алгоритмів. У цьому і полягає відмінність даної реалізації алгоритму "найближчих сусідів" від відомих аналогів. Алгоритм MR використовується для прогнозу значень числових змінних і категоріальних змінних, включаючи текстові (string data type), а також для класифікації на два або декілька класів.



Алгоритми кластеризації

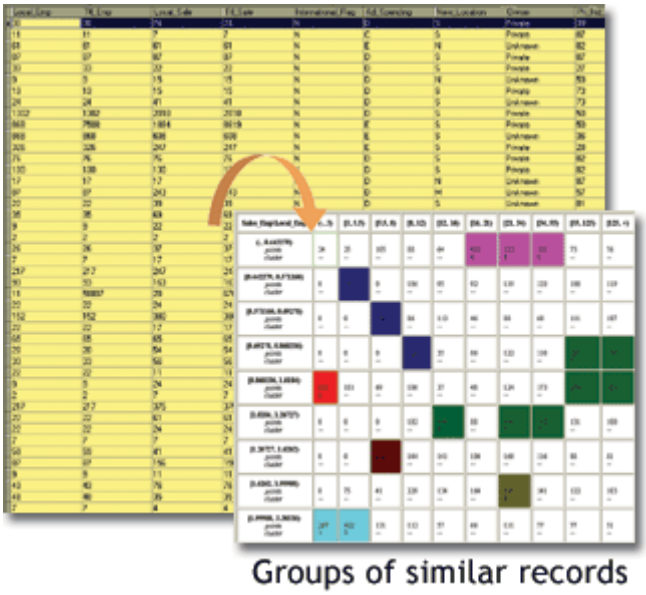
Find Dependecies (FD) - N-мірний аналіз розподілів

Даний алгоритм виявляє в початковій таблиці групи записів, для яких характерна наявність функціонального зв'язку між цільовою змінною і незалежними змінними, оцінює ступінь (силу) цієї залежності в термінах стандартної помилки, визначає набір найбільш впливаючих чинників, відсіває крапки, що відскочили. Цільова змінна для FD повинна бути числового типу, тоді як незалежні змінні можуть бути і числовими, і категоріями, і логічними.

Алгоритм працює дуже швидко і здатний обробляти великі об'єми даних. Його можна використовувати як препроцесор для алгоритмів FL, PN, LR, оскільки він зменшує простір пошуку, а також як фільтр крапок, що відскочили, або в зворотній постановці, як детектор виключень. FD створює правило табличного вигляду, проте як і всі правила PolyAnalyst воно може бути обчислене для будь-якого запису таблиці.

Find Clusters (FC) - N-мірнийкластерізатор

Цей метод застосовується тоді, коли треба виділити в деякій безлічі даних компактні типові підгрупи (кластери), що складаються з близьких по своїх характеристиках записів. Причому наперед може бути невідомо які змінні потрібно використовувати для такого розбиття. Алгоритм FC сам визначає набір змінних, для яких розбиття найбільш значуще. Результатом роботи алгоритму є опис областей (діапазонів значень змінних), що характеризують кожен виявлений кластер і розбиття досліджуваної таблиці на підмножини, відповідні кластерам. Якщо дані є достатньо однорідними по всіх своїх змінних і не містять "згущувань" крапок в якихось областях, цей метод не дасть результатів. Треба відзначити, що мінімальне число кластерів, що виявляються, рівне двом - згущування крапок тільки в одному місці в даному алгоритмі не розглядається як кластер. Крім того, цей метод більшою мірою, чим інші пред'являє вимоги до наявності достатньої кількості записів в досліджуваній таблиці, а саме, мінімальна кількість записів в таблиці, в якій може бути виявлено N кластерів, рівне (2N-1) 4.

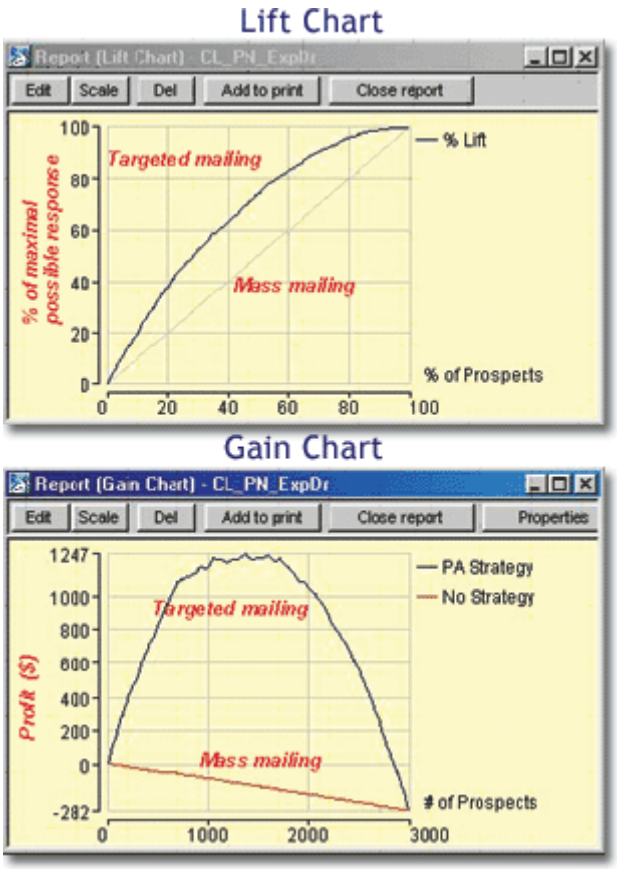


Алгоритми класифікації

У пакеті PolyAnalyst є багатий інструментарій для вирішення завдань класифікації, для знаходження правил віднесення записів до одного з двох або до одного з декількох класів.

Classify (CL) - класифікатор на основі нечіткої логіки

Алгоритм CL призначений для класифікації записів на два класи. У основі його роботи лежить побудова так званої функції належності і знаходження порогу розділення на класи. Функція належності приймає значення від 0 до 1. Якщо повернене значення функції для даного запису більше порогу, то цей запис належить до класу "1", якщо менше, то класу "0" відповідно. Цільова змінна для цього модуля повинна бути логічного типу.



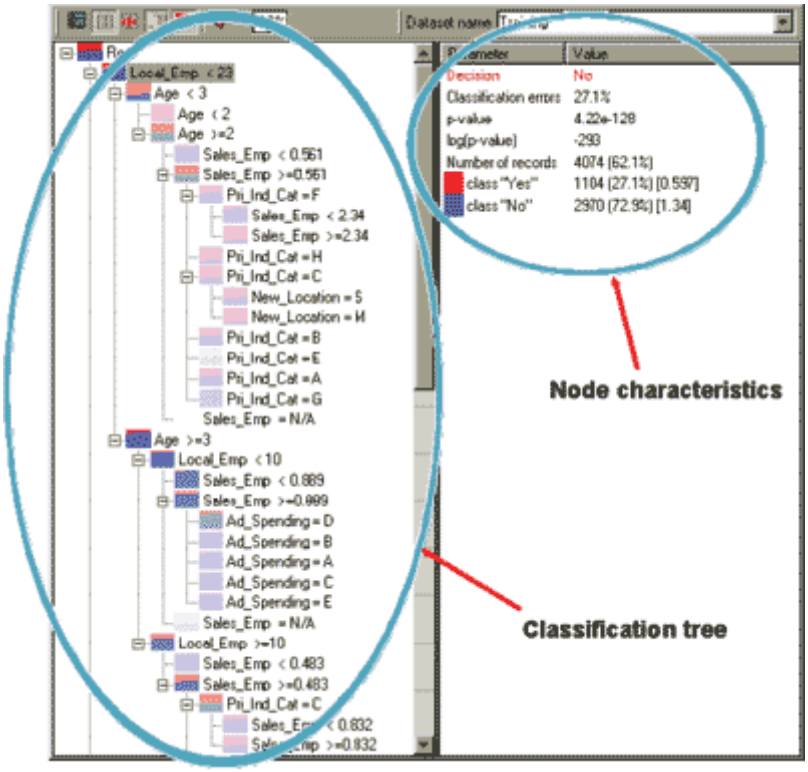
Discriminate (DS) - Дискримінація

Даний алгоритм є модифікацією алгоритму CL. Він призначений для того, щоб з'ясувати, чим дані з вибраної таблиці відрізняються від решти даних, включених в проект, іншими словами для виділення специфічних рис, що характеризують деяку підмножину записів проекту. На відміну від алгоритму CL, він не вимагає завдання цільовій змінній, досить вказати лише таблицю, для якої потрібно знайти відмінності.

Decision Tree (DT) - дерево рішень

Алгоритми "дерева рішень" широко поширені і реалізовані в багато Data Mining пакетах. Ці алгоритми використовуються в завданнях класифікації на два і більша кількість класів. Результатом їх роботи є ієрархічна деревоподібна структура, що складається з гілок, вузлів і листя. Для кожного вузла обчислюється критерій розщеплювання. Якщо дерево не дуже „розлоге", то таке уявлення є достатньо наочним. У системі PolyAnalyst, реалізований алгоритм, заснований на критерії максимізації взаємної інформації (information gain). Тобто для розщеплювання вибирається незалежна змінна, що несе максимальну (у сенсі Шеннона)

інформацію про залежну змінну. Цей критерій на відміну від багатьох критеріїв, вживаних в інших системах Data Mining, має ясну інтерпретацію і дає розумні результати при найрізноманітніших статистичних параметрах даних, що вивчаються. Алгоритм DT є одним з найшвидших в PolyAnalyst.



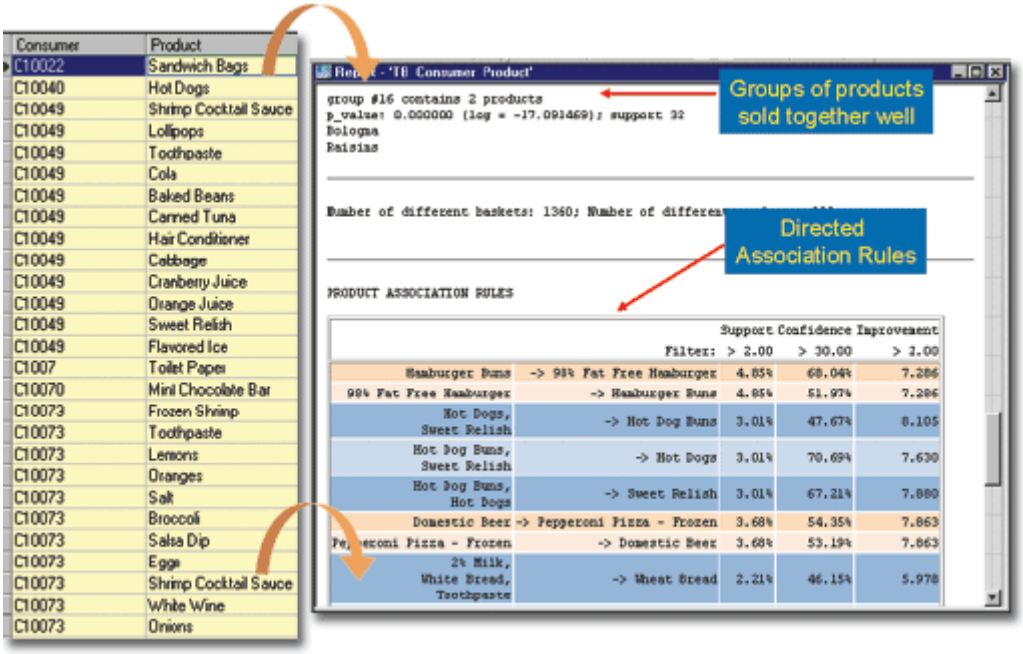
Decision Forest (DF) - ліси рішень

У разі, коли залежна змінна може приймати велику кількість різних значень, застосування методу дерев рішень стає неефективним. У такій ситуації в системі PolyAnalyst застосовується метод, названий лісом рішень (decision forest). При цьому будується сукупність дерев рішень - поодинці для кожного різного значення залежної змінної. Результатом прогнозу, заснованому на лісі рішень, є те значення залежної змінної, для якої відповідне дерево дає найбільш вірогідну оцінку.

Алгоритми асоціації

Market Basket Analysis (BA) - метод аналізу "корзини покупця"

Назва цього методу походить від завдання визначення які товари ймовірно купуються спільно. Проте реальна область його застосування значно ширша. Наприклад, продуктами можна вважати сторінки в Інтернеті, або ті або інші характеристики клієнта або відповіді респондентів в соціологічних і маркетингових дослідженнях ітд. Алгоритм BA отримує на вхід бінарну матрицю, в якій рядок - це одна корзина (касовий чек, наприклад), а стовпці заповнені логічними 0 і 1, такими, що позначають наявність або відсутність даної ознаки (товару). На виході формуються кластери ознак, що спільно зустрічаються, з оцінкою їх вірогідності і достовірності. Окрім цього формуються асоціативні направлені правила типу: якщо ознака "A", то з такою - то вірогідністю ще і ознака "B" ,і ще ознака "C". Алгоритм BA в PolyAnalyst працює виключно швидко і здатний обробляти величезні масиви даних.



Transactional Basket Analysis (ТБ) - транзакційнийаналіз "корзини"

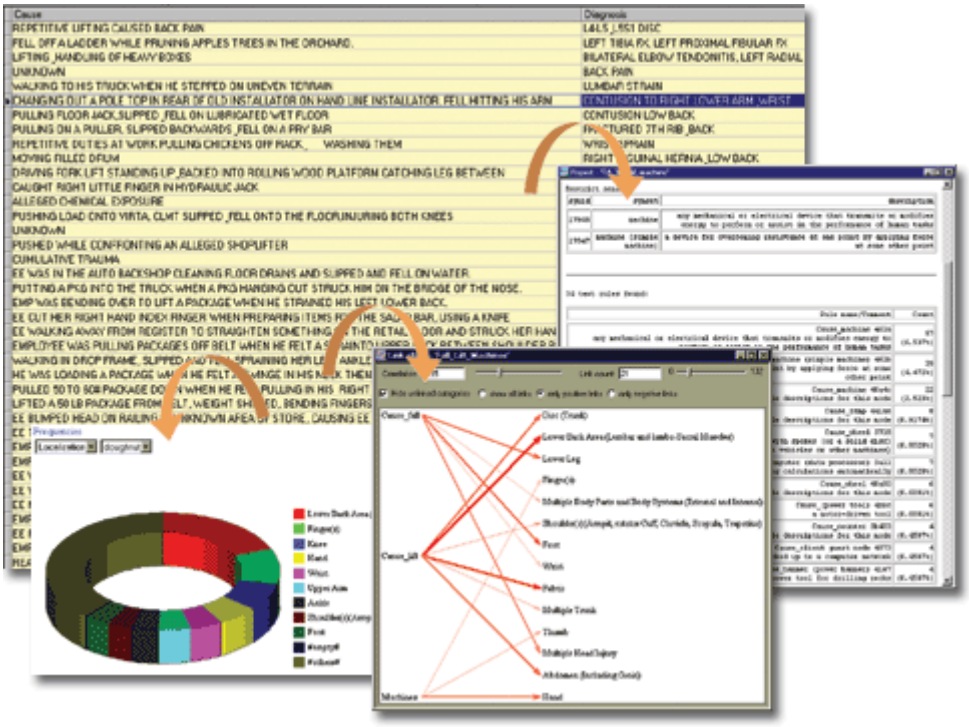
Transactional Basket Analysis - цемодифікаціяалгоритму ВА, вживанийдляаналізудужевеликихданих, щонерідкістьдляцьоготипузавдань. Він припускає, що кожен запис в базі даних відповідає одній транзакції, а не одній корзині (набору куплених за одну операцію товарів). На основі цього алгоритму компанія "Мегапьютер" розробила окремий продукт - X-SellAnalyst, призначений для on-line рекомендації продуктів в Інтернет магазинах.

Модулі текстового аналізу

Однією з унікальних особливостей PolyAnalyst є інтеграція інструментів Data Mining - засобів аналізу числової інформації з методами аналізу текстів на природній мові - алгоритмів Text Mining. На жаль, в поточній версії програми алгоритми Text Mining реалізовані тільки на англійській мові, проте в найближчих планах виробника забезпечити і підтримку російського та ряду інших європейських мов.

Text Analysis (ТА) - текстовий аналіз

Text Analysis є засобом формалізації неструктурованих текстових полів в базах даних. При цьому текстове поле представляється як набір булевих ознак, заснованих на наявності і/або частоті даного слова, стійкого словосполучення або поняття (з урахуванням відносин синонімії) в даному тексті. При цьому з'являється можливість розповсюдити на текстові поля всю потужність алгоритмів Data Mining, реалізованих в системі PolyAnalyst. Крім того, цей метод може бути використаний для кращого розуміння текстовою компоненти даних за рахунок автоматичного виділення найбільш ключових понять.

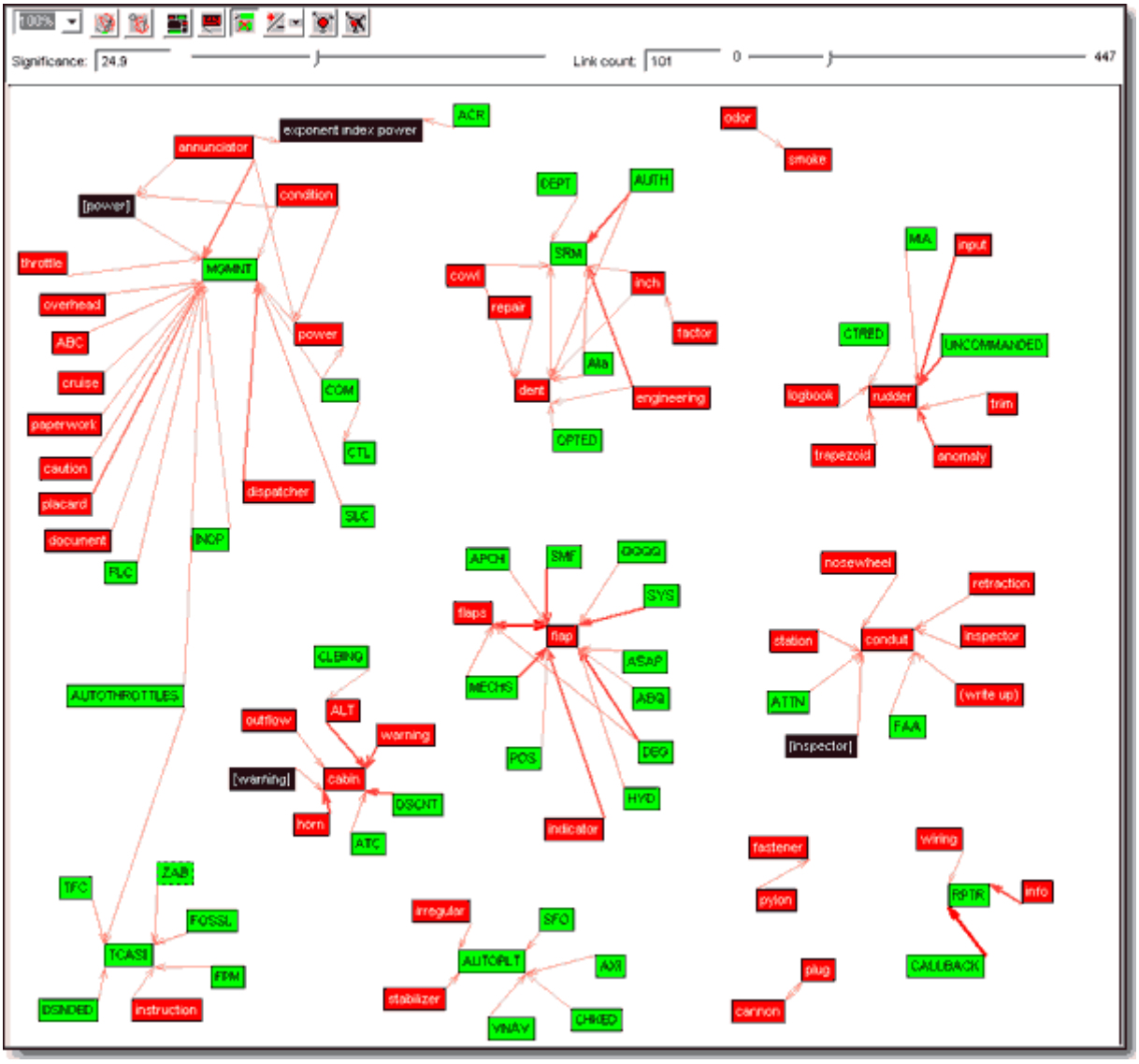


Text Categorizer (TC) - каталогізатор текстів

Цей модуль дозволяє автоматично створити ієрархічний деревовидний каталог наявних текстів і помітити кожен вузол цієї деревовидної структури найбільш індикативним для текстів, що відносяться до нього. Це потрібно для розуміння тематичної структури аналізованої сукупності текстових полів і ефективної навігації по ній.

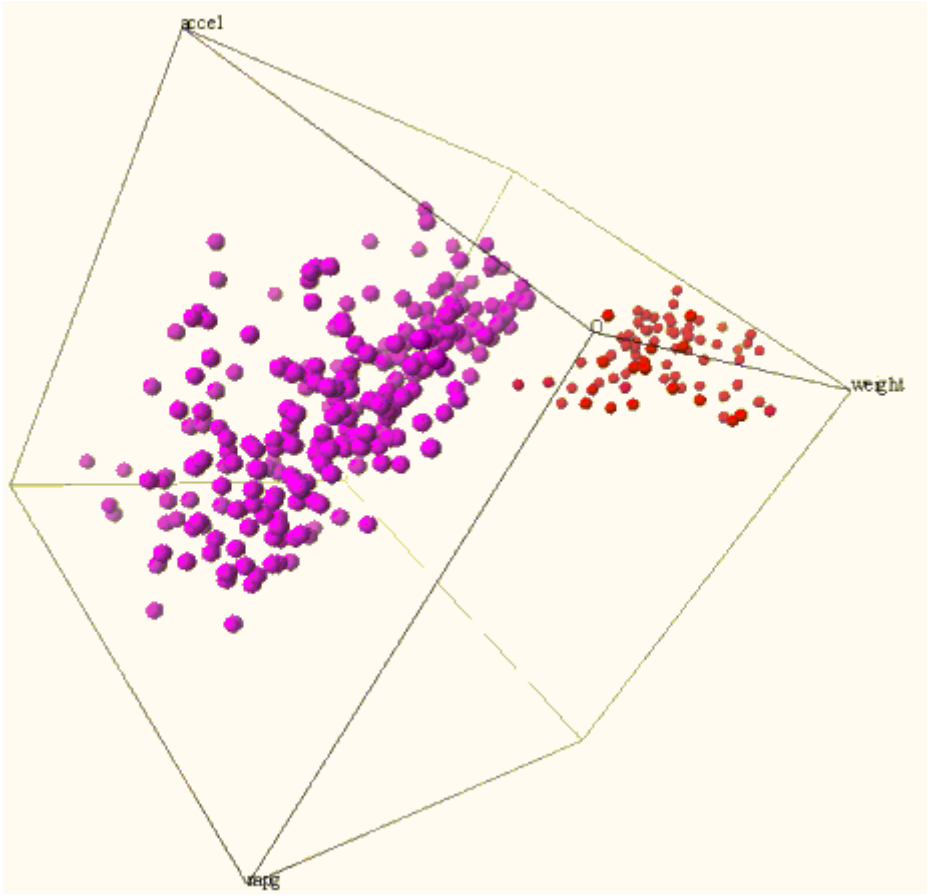
Link Terms (LT) - зв'язок понять

Цей модуль дозволяє виявляти зв'язки між поняттями, що зустрічаються в текстових полях бази даних, що вивчається, і представляти їх у вигляді графа. Цей граф також може бути використаний для виділення записів, що реалізовують вибраний зв'язок.



Візуалізація

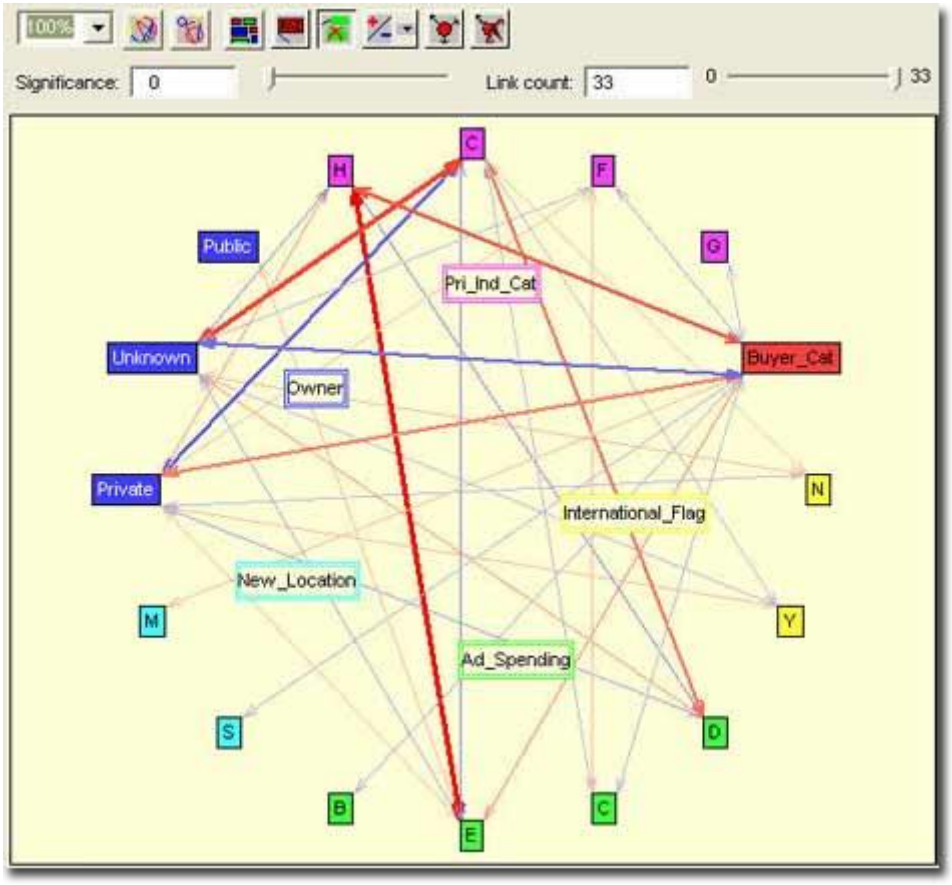
У PolyAnalyst є багатий набір інструментів для графічного уявлення і аналізу даних і результатів досліджень. Дані можуть представлятися в різних зорових форматах: гістограмах, двовимірних, псевдо- і реальних трьохвимірних графіках.



Знайдені в процесі Data Mining залежності можуть бути представлені як інтерактивні графіки із слайдерами для зміни значень представлених на них змінних. Ця особливість дозволяє користувачеві графічно моделювати результати. Є набір спеціальних графіків, широко вживаних в бізнесі, це так звані Lift, Gain charts, які використовуються для графічної оцінки якості класифікаційних моделей і вибору оптимального числа контактів (prospects). Окрім цього в останню версію програми включений новий візуальний метод Data Mining - аналіз зв'язків.

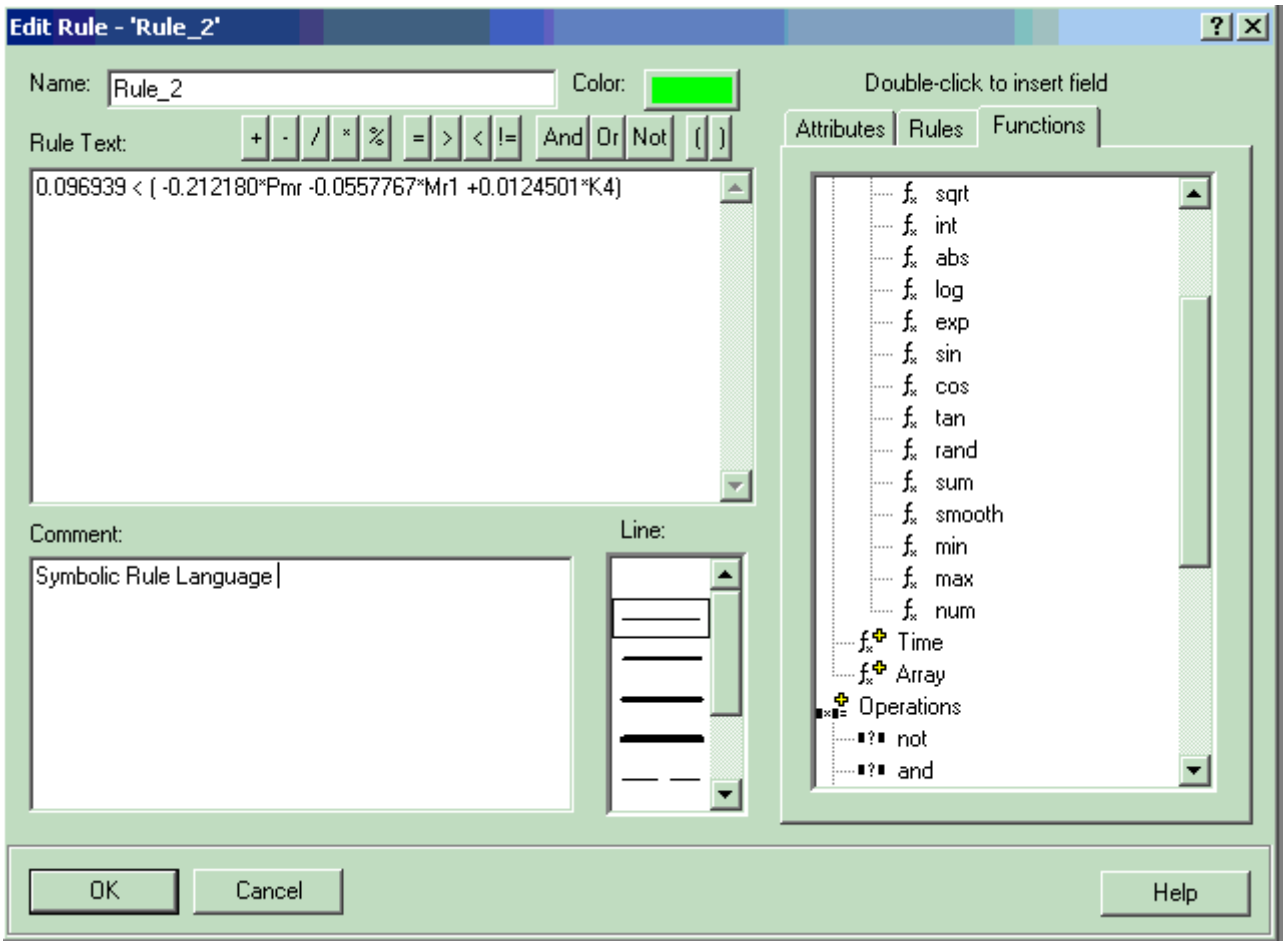
Link Analysis (LA) - аналіз зв'язків

Модуль Link Analysis дозволяє виявляти кореляційні і антикореляційні зв'язки між значеннями категоріальних і булевих полів і представляти їх у вигляді графа. Цей граф також може бути використаний для виділення записів, що реалізовують вибраний зв'язок.



Symbolic Rule Language (SRL) - мова символьних правил

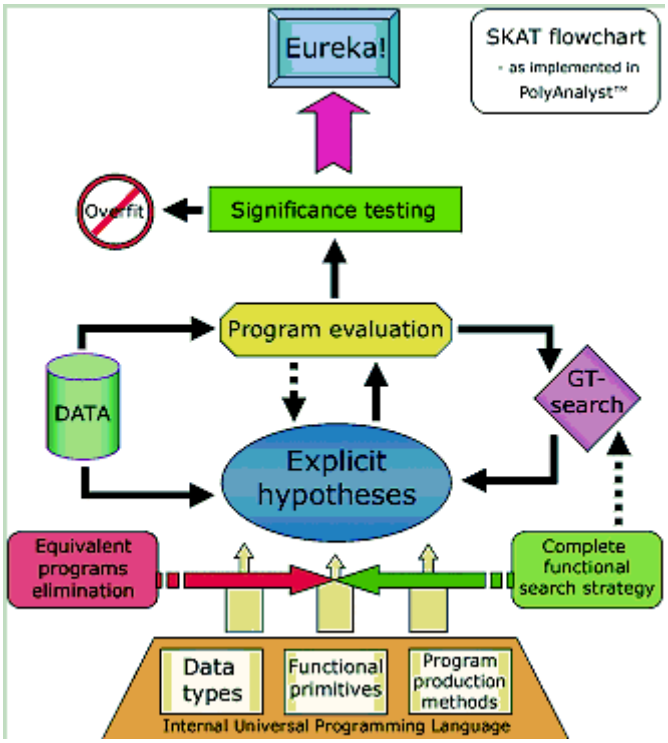
SRL - це універсальна алгоритмічна мова PolyAnalyst, яка використовується для символьного уявлення автоматично знайдених системою в процесі Data Mining правил, а також для створення користувачем своїх власних правил. На мові SRL можна виразити широкий спектр



математичних конструкцій, використовуючи операції алгебри, великий набір вбудованих функцій, операції з датами і часом, логічні і умовні конструкції. Для зручності написання виразів на SRL в програмі передбачений майстер створення правил.

Еволюційне програмування

Зараз Еволюційне програмування найбільш молода і багатообіцяюча технологія DataMining. Основна ідея методу полягає у формуванні гіпотез про залежність цільової змінної від інших змінних у вигляді, що автоматично синтезуються спеціальним модулем програм на внутрішній мові програмування. Використання універсальної мови програмування теоретично дозволяє виразити будь-яку залежність, причому вид цієї залежності наперед не відомий. Процес виробництва внутрішніх програм організовується як еволюція в просторі програм, в деякому роді що нагадує генетичні алгоритми. Коли система знаходить перспективну гіпотезу, що описує досліджувану залежність досить добре по цілому ряду критерій, в роботу включається механізм так званих "узагальнених перетворень" (GT-search). За допомогою цього механізму в "хорошу" програму вводяться незначні модифікації, не погіршуючі її якість, і проводиться відбір кращої дочірньої програми. До нової популяції потім знову застосовуються механізми синтезу нових програм, і цей процес рекурсивно повторюється. Таким чином, система створює деяке число генетичних ліній програм, що конкурують один з одним по точності, статистичній значущості і простоті виразу залежності.



Спеціальний модуль безперервно перетворить "кращу" на даний момент програму з внутрішнього уявлення в зовнішню мову PolyAnalyst - мову символьних правил (Symbolic Rule Language), зрозумілу людині: математичні формули, умовні конструкції і так далі. Це дозволяє користувачеві зрозуміти суть отриманої залежності, контролювати процес пошуку, а також отримувати графічну візуалізацію результатів. Контроль статистичної значущості отриманих результатів здійснюється цілим комплексом ефективних і сучасних статистичних методів, включаючи методи рандомізованого тестування.

Загальносистемні характеристики PolyAnalyst

Типи даних

PolyAnalyst працює з різними типами даних. Це - числа, булеві змінні (yes/no), категоріальні змінні, текстові рядки, дати, а також вільний англійський текст.

Доступ до даних

PolyAnalyst може отримувати початкові дані з різних джерел. Це: текстові файли з роздільником кома (.csv), файли Microsoft Excel 97/2000, будь-яка ODBC- сумісна СУБД, SAS data files, Oracle Express, IBM Visual Warehouse.

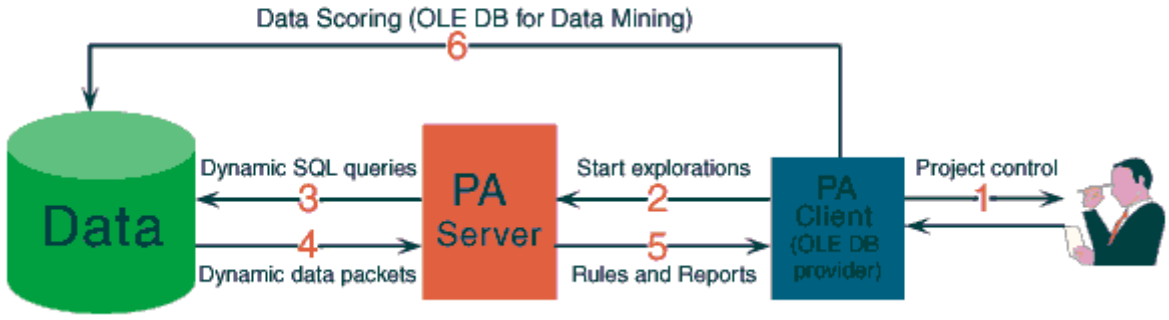
Підтримка OLE DB for Data Mining

Версія 4.6 PolyAnalyst підтримуєспецифікацію Microsoft OLE DB for Data Mining (Version 1.0). Привиконаннідослідженьдлябільшостіматематичнихмодулів (LR, FD, CL, FC, DT, DF, FL,PN, BA, ТБ) можнастворюватитакзвані "Mining Models" (MM). Післязавершенняаналізуціможеліможназастосовуватидозовнішніхданихчерезстандартніінтер фейси OLE DB або ADO зіншихпрограмабоскриптівтих, щопідтримуютьстворення ADO або COM-об'єктів. Застосування моделі здійснюється за допомогою виконання SQL-команд (Розширення SQL for DM). Mining Models можна також експортувати в PMML. У подальших планах розвитку програми намічається забезпечити інтеграцію "PolyAnalystDataMiningProvider" зMicrosoftAnalysisServices(ускладіSQLServer 2000)

In-place Data Mining

PolyAnalyst підтримуєзапускдослідженьназовнішніхданихчерез OLE DB інтерфейсиприбеззавантаженняцихданихвпроект PA. Привиконаннідослідження PolyAnalyst отримуєданіпорціямичерезвиконання SQL запитівдозовнішніхджерелданих. Це дозволяє подолати обмеження пам'яті при дослідженні великих масивів даних.

SQL-Mode Data Mining Workflow Processes



PolyAnalyst Scheduler - режим пакетної обробки

У PolyAnalyst передбачена можливість пакетного режиму аналізу даних. Для цього є спеціальна ськріптовий мова, на якій програмується всі аналітичні дії і тимчасова послідовність їх виконання, а також визначаються набори даних. Скрипт зберігається у файлі і автоматично ініціалізує дослідження у вказаний момент часу на певних даних. Для реалізації функції Scheduler в електронній ліцензії повинна бути включена відповідна опція.

Сімейство продуктів PolyAnalyst

Продукт	Конфігурація системи
Локальні продукти	

PolyAnalyst 4.6, однокористувацька версія	Математичні модулі: FL, FD, PN, FC, BA, TB, MB, CL, DS, DT, DF, LR, LA, TA, TC, LT, SS. Пакетна обрабка, підтримка OLE DB. Платформа - MSWindowsNT/2000/XP
PolyAnalyst 3.5 Professional (рос.)	Математичні модулі: FL, FD, PN, FC, CL, DS, LR, SS. Платформа - MS Windows NT/2000/XP
PolyAnalyst 3.5 Power (рос.)	Математичнімодулі: FD, PN, FC, CL, DS, LR, SS. Платформа - MS Windows 98/NT/2000/XP
PolyAnalyst 3.5 Lite – студентська версія (рос.)	Математичнімодулі: FD, FC, CL, DS, LR, SS. Платформа - MS Windows 98/NT/2000/XP
Мережеві продукти	
PolyAnalyst Knowledge Server 4.6, мережева версія	Математичнімодулі: FL, FD, PN, FC, BA, TB, MB, CL, DS, DT, DF, LR, LA, TA, TC, LT, SS. Пакетнаобрабка, підтримка OLE DB, In-Place Data Mining.Серверначастина- MS Windows NT/2000/XP server, клієнтськачастина - MS Windows 98/NT/2000/XP. Клієнт/сервернаверсіясистеми
Засоби розробки	
PolyAnalyst COM - SDK для створення власних додатків для Data Mining	Набір COM-об'єктів, бібліотеки, документація для розробників

Схема ліцензування PolyAnalyst

Для версії 4.6 діє покомпонентна система ліцензування, тобто можна вибрати тільки ті математичні модулі і ту функціональність системи, які необхідні користувачеві для вирішення даного конкретного класу завдань. Єдиним обов'язковим модулем є PolyAnalystWorkplace. Технічно вся конфігурація програми визначається в спеціальному криптованому файлі електронної ліцензії, який висилається при реєстрації продукту. З 2003 року для російських замовників діють спеціальні ціни на однокористувацьку (stand-alone) версію системи, які значно нижчі світових. Ця акція компанії-виробника направлена на просування технологій Data Mining на вітчизняний ринок. Мегапьютер також проводить активну освітню програму, в рамках якої для освітніх установ діють додаткові знижки.

PolyAnalyst COM SDK - засіб розробки, а також клієнт/серверна конфігурація системи - PolyAnalyst Knowledge Server ліцензуються на окремих умовах.

VII. Засоби Data Mining в Microsoft SQL Server 2000

З можливостей, SQL Server 2000, що надаються, перш за все виділимо наступні:

- побудова і обробка моделей Data Mining;
- витягання даних як з реляційних, так і з багатовимірних джерел;

- дваалгоритмиздобуванняданих - Microsoft Decision Trees і Microsoft Clustering;
- розширення мови запитів до багатовимірних даних (MDX);
- робота із зовнішніми додатками через об'єктну модель DSO (Decision Support Objects).

Моделі

Моделі Data Mining - це основа витягання даних в SQL Server 2000. По суті модель є сукупність метаданих, що відображають деякі правила і закономірності в початкових даних. При цьому структура моделі визначає набір ключових атрибутів аналізу, тоді як її зміст несе безпосередньо статистичну інформацію - тут простежується схожість з ідеологією звичайних таблиць. Проте варто мати на увазі, що на основі одного і того ж набору початкових даних можна побудувати декілька різних моделей. У цьому сенсі побудова правильної моделі гарантує нам отримання саме тих "прихованих" даних, які ми прагнемо виявити. На рис. 12 показана структура моделі, що містить дані про покупців магазина в розрізі товарів, що їх придбають.

Рисунок 12. Структура моделі Data Mining

Процес побудови моделі реалізований в AnalysisServices у вигляді майстра, що дозволяє крок за кроком задати параметри моделі і виконати її обробку, що, на думку розробників, спрощує проведення аналізу.

Вибір джерела даних

Перший крок в побудові моделі - вибір джерела даних для аналізу. Підтримуються два типи джерел даних: багатовимірні, використовувані в рамках технології OLAP (правда, поки як OLAP-джерело можна використовувати тільки сам модуль Analysis Services), і звичайні - реляційні. Наявність першого варіанту дає набагато більшу свободу вибору для аналізу, адже далеко не кожне підприємство має власне багатовимірне сховище даних.

Після вибору джерела можна приступати безпосередньо до формування структури моделі. Для цього потрібно визначити таблицю (або вимір, у разі багатовимірного джерела), що містить аналізовані дані, а також вибрати одне з полів таблиці (або показник багатовимірного куба), яке знаходитиметься у фокусі дослідження. Наприклад, якщо вам потрібно оцінити ризик кредиту для певних клієнтів банку, то величину цього ризику можна вибрати як предмет дослідження. Початковими даними для дослідження у такому разі можуть виступати дані про клієнта - вік, річний дохід, наявність автомобіля, місце проживання і т.п. Взагалі кажучи, вибір початкових даних і предмету аналізу - процес творчий, так що якщо не вдалося отримати необхідні оцінки відразу, то спробуйте змінити структуру моделі, ввівши в неї додаткові атрибути. Можливо, це дозволить оцінити ситуацію з іншої точки зору.

Вибір алгоритму аналізу

Наступний важливий крок - вибір одного з двох алгоритмів аналізу даних. Як вже зазначалось , Analysis Services підтримує два алгоритми - Microsoft Decision Trees і Microsoft Clustering. Оскільки області застосування і результати роботи кожного з них можуть сильно розрізнятися, на цьому кроці має сенс зупинитися докладніше. Алгоритм Microsoft Decision Trees заснований на відомому методі побудови дерев рішень. У його рамках значення кожного з досліджуваних атрибутів класифікується на основі значень решти атрибутів, з використанням правил вигляду “якщо -- то”. Результат роботи такого алгоритму - деревовидна структура, кожен вузол якої є якесь запитання. Щоб вирішити, до якого класу віднести деякий об'єкт або ситуацію, потрібно відповісти на питання, що стоять у вузлах цього дерева, починаючи з його кореня (найбільш близький аналог такої структури - дерево видів в біології). Головна перевага цього алгоритму - наочність і простота використання. Проте область застосування "деревовидного" методу обмежена в основному завданнями класифікації Другий алгоритм, Microsoft Clustering, використовує інший, не менш відомий метод пошуку логічних закономірностей - метод “найближчого сусіда”. В процесі роботи алгоритму початкові дані об'єднуються в групи (кластери) на основі аналогічних або схожих значень атрибутів. Отримані набори даних аналізуються, що дозволяє виявити приховані закономірності або побудувати імовірносний прогноз. Даний алгоритм дозволяє провести глибший аналіз даних, чим дерево рішень, але і він має свої обмеження. Його переважно застосовують для наборів даних із схожими атрибутами, значення яких належать певному інтервалу (наприклад, вік, річний дохід і т. п.). Проте у разі нетипових значень атрибутів алгоритм може давати невірну оцінку. Вибір правильного алгоритму залежить від класу завдання, яке потрібно вирішити, а також від складу початкових даних. Задачі класифікації неоднорідних даних краще вирішувати за допомогою алгоритму дерев рішень, а завдання прогнозування або виявлення неявних закономірностей - за допомогою методу кластеризації. Який би алгоритм ви не вибрали, на цьому побудова моделі закінчена, і можна переходити до наступного процесу - **тренування моделі.**

Тренування побудованої моделі - це не що інше, як процес обробки початкових даних згідно вибраного алгоритму. Цей процес може зайняти тривалий час, особливо при великих об'ємах даних. Після закінчення тренування початкові дані більше вам не знадобляться. В результаті тренування модель буде заповнена статистичними даними, які можуть бути представлені як в графічному, так і в цифровому вигляді.

Відображення результатів

Для відображення результатів аналізу використовуються вбудовані засоби Analysis Services. При цьому варіанти відображення різні для кожного з алгоритмів. Як приклад нижче приведені результати роботи алгоритму Microsoft Decision Trees.

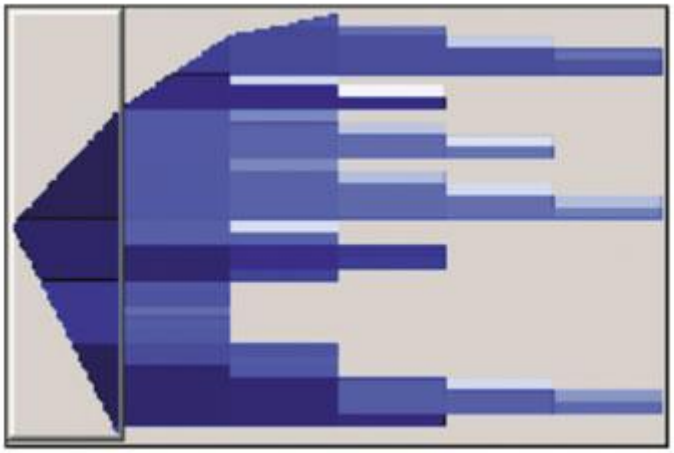


Рисунок 13. Дерево рішень.

Схема на рис.13 показує всі гілки побудованого дерева рішень. Темнішим кольором виділені гілки, відповідні найбільшій вірогідності (числу попадань), а світлішим - найменшою. У даному прикладі гілок у дерева небагато, проте в деяких випадках їх число може досягати декількох сотень. Виділена частина дерева відображається в режимі детального перегляду (рис.14).

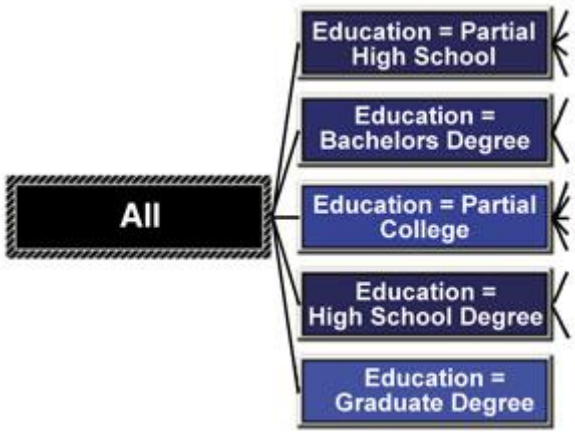


Рисунок 14. Вибрана частина дерева рішень

Будь-яку частину дерева рішень можна виділити для детального перегляду, але при цьому не можна проглядати більше двох рівнів одночасно. На збільшеній частині дерева можна бачити значення, привласнені кожному з вузлів в процесі роботи алгоритму. Як і в режимі проглядання всього дерева цілком, колір вузла тут сигналізує про кількість попадань початкових даних в цю гілку. Вибір певного вузла дерева дозволяє проглянути статистичну інформацію про даний вузол в числовому вигляді. Ця інформація включає значення вузла дерева, кількість значень початкових даних, що потрапили в дану гілку, і вірогідність попадання (рис.15).

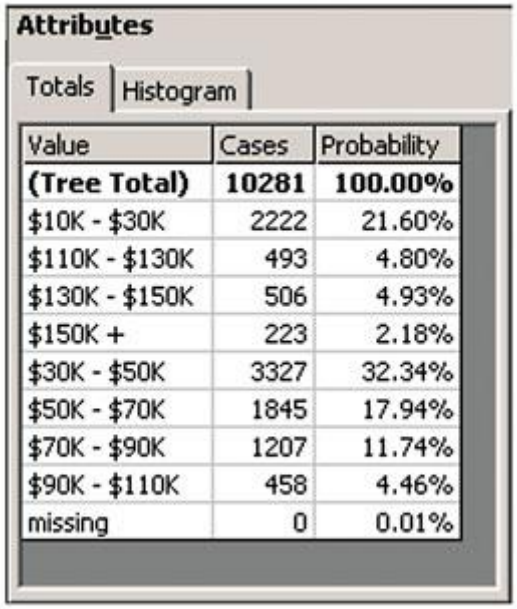


Рисунок 15. Інформація в числовій формі.

Отже, ми бачимо, що засоби витягання даних в SQL Server 2000 Analysis Services надають достатньо багатий набір функціональних можливостей для аналітиків і менеджерів підприємств. До того ж даний інструментарій відрізняється простотою у використанні і невисокою ціною, і, думається, він зможе знайти своїх користувачів в середовищі багатьох компаній.

VIII. Сфера застосування технологій інтелектуальних обчислень

8.1. Бізнес-застосування Data Mining

Для застосування продукту Data Mining, необхідно виконати ряд кроків:

1. Встановити масштаби проекту, що визначають, які дані необхідно зібрати. Важливо, щоб проект був направлений на реалізацію реальних бізнес-цілей.

2. Розробити базу даних для Data Mining. Необхідна інформація може бути розподілена по декількох базах, іноді вона навіть зберігається не в електронній формі. Дані з різних баз необхідно консолідувати і усунути невідповідності. Насправді розвиток технології баз даних вже не вимагає застосування алгоритмів Data Mining до окремої вітрини даних. Фактично, ефективний аналіз вимагає корпоративного сховища даних, що з погляду вкладень обходиться дешевше, ніж використання окремих вітрин.

Відзначимо, що у міру впровадження Data Mining - проектів в масштабі підприємства кількість користувачів зростає, тому все частіше виникає необхідність в доступі до великомасштабних інфраструктур даних. Сучасне сховище надає не тільки ефективний спосіб зберігання всіх корпоративних даних і усуває необхідність у використанні інших вітрин і джерел, але і стає ідеальною основою для Data Mining - проектів. Репозиторій даних підприємства забезпечує узгоджені і актуальні дані про клієнтів. Упроваджуючи Data Mining функції в сховищі, компанії скорочують витрати в двох напрямах. В цьому випадку, по-перше, вже не потрібно набувати і обслуговувати додаткове устаткування для Data Mining . По-друге, компанії не потрібно переносити дані зі сховища в спеціальні джерела для Data Mining - проектів, при цьому економляться час і матеріальні ресурси.

Ще один важливий момент - очищення даних. Тут розуміється перевірка на цілісність і обробка відсутніх значень. Точність методів Data Mining залежить від якості інформації, яка лежить в їх основі. Відмітимо, що перші два етапи можуть зайняти половину (а то і більше) часу, відведеного на весь проект.

3. Застосувати алгоритми Data Mining для визначення відносин між даними. І не виключено, що для виявлення потрібних залежностей доведеться використовувати декілька різних алгоритмів. Одні з них підійдуть на перших етапах процесу, інші на пізніших. У певних випадках має сенс запустити декілька алгоритмів паралельно, щоб проаналізувати дані з різних точок зору.

4. Досліджувати співвідношення, виявлені на попередніх етапах, на застосування в масштабах проекту. На цьому етапі можливо потрібна допомога експерта в певній області. Він визначить, чи є ті або інші відносини дуже специфічними або дуже загальними і вкаже, в яких областях слід продовжити аналіз.

5. Представити результати у вигляді звіту, в якому будуть перераховані всі відносини, що інтерпретуються. Такий звіт принесе тільки одномоментну вигоду, тоді таке як застосування, що дозволяє експертові творчо підходити до виявлення відносин, набагато корисніше. Тому фірма-постачальник повинна не тільки навчити клієнта методиці пошуку залежностей в даних, але і звернути особливу увагу на навчальній роботі з самою програмою.

Також на розподіл часу для Data Mining проекту впливають і інші чинники: тип кінцевого застосування, наявність і стан сховища даних. Наприклад, якщо взяти застосування для прогнозування продажів, то виявлені відносини між даними можна використовувати до тих пір, поки не зміниться діяльність компанії. І навпаки, при аналізі споживчої корзини компанія зазвичай шукає все нові залежності в даних. Для проекту прогнозування збуту більше часу доведеться витратити на перших трьох етапах, а для аналізу споживчої корзини - на останньому.

Сфера застосування Data Mining нічим не обмежена - вона скрізь, де є які-небудь дані. Але в першу чергу методи Data Mining сьогодні, м'яко кажучи, заінтригували комерційні підприємства, що розгортають проекти на основі інформаційних сховищ даних (Data Warehousing).

Data Mining представляють велику цінність для керівників і аналітиків в їх повсякденній діяльності. Ділові люди усвідомили, що за допомогою методів Data Mining вони можуть отримати відчутні переваги в конкурентній боротьбі. Стисло охарактеризуємо деякі можливі бізнес- застосування Data Mining .

Роздрібна торгівля

Ось типові завдання, які можна вирішувати за допомогою Data Mining у сфері роздрібної торгівлі:

· *аналіз купівельної корзини* (аналіз схожості) призначений для виявлення товарів, яких покупці прагнуть придбати разом. Знання купівельної корзини необхідне для поліпшення реклами, вироблення стратегії створення запасів товарів і способів їх розкладки в торгових залах.

· *дослідження тимчасових шаблонів* допомагає торговим підприємствам приймати рішення про створення товарних запасів.

· *створення прогнозуючих моделей* дає можливість торговим підприємствам дізнаватися характер потреб різних категорій клієнтів з певною поведінкою. Ці знання потрібні для розробки точно направлених, економічних заходів щодо просування товарів.

Банківська справа

Досягнення технології Data Mining використовуються в банківській справі для вирішення наступних поширених завдань:

· *виявлення шахрайства з кредитними картками*. Шляхом аналізу минулих транзакцій, які згодом виявилися шахрайськими, банк виявляє деякі стереотипи такого шахрайства.

· *сегментація клієнтів*. Розбиваючи клієнтів на різні категорії, банки роблять свою маркетингову політику більш цілеспрямованою і результативною, пропонуючи різні види послуг різним групам клієнтів.

· *прогнозування змін клієнтури*. Data Mining допомагає банкам будувати прогнозні моделі цінності своїх клієнтів, і відповідним чином обслуговувати кожну категорію.

Телекомунікації

В області телекомунікацій методи Data Mining допомагають компаніям енергійніше просувати свої програми маркетингу і ціноутворення, щоб утримувати існуючих клієнтів і привертати нових. Серед типових заходів відзначимо наступні:

· *аналіз записів про докладні характеристики викликів.* Призначення такого аналізу - виявлення категорій клієнтів з схожими стереотипами користування їх послугами і розробка привабливих наборів цін і послуг;

· *виявлення лояльності клієнтів*. Data Mining можна використовувати для визначення характеристик клієнтів, які, один раз скориставшись послугами даної компанії, з великою часткою вірогідність залишаться їй вірними. У результаті засоби, що виділяються на маркетинг, можна витрачати там, де віддача більше всього.

Спеціальні застосування

Медицина

Відомо багато експертних систем для постановки медичних діагнозів. Вони побудовані головним чином на основі правил, що описують поєднання різних симптомів різних захворювань. За допомогою таких правил дізнаються не тільки, на що хворий пацієнт, але і як потрібно його лікувати. Правила допомагають вибирати засоби медикаментозної дії, визначати свідчення - протипоказання, орієнтуватися в лікувальних процедурах, створювати умови найбільш ефективного лікування, передбачати результати призначеного курсу лікування і т.п. Технології Data Mining дозволяють виявляти в медичних даних шаблони, складаючи основу вказаних правил.

Молекулярна генетика і генна інженерія

Мабуть, найгостріше і разом з тим чітко завдання виявлення закономірностей в експериментальних даних, стоїть в молекулярній генетиці і генній інженерії. Тут вона формулюється як визначення так званих маркерів, під якими розуміють генетичні коди, контролюючі ті або інші фенотипічні ознаки живого організму. Такі коди можуть містити сотні, тисячі і більш зв'язаних елементів.

На розвиток генетичних досліджень виділяються великі кошти. Останнім часом в даній області виник особливий інтерес до застосування методів Data Mining. Прикладна хімія

Методи Data Mining знаходять широке застосування в прикладній хімії (органічній і неорганічній). Тут нерідко виникає питання про з'ясування особливостей хімічної будови тих або інших з'єднань, що визначають їх властивості. Особливо актуальне таке завдання при аналізі складних хімічних сполук, опис яких включає сотні і тисячі структурних елементів і їх зв'язків.

Можна привести ще багато прикладів різних областей знання, де методи Data Mining відіграють провідну роль. Особливість цих областей полягає в їх складній системній організації. Вони відносяться головним чином до організації систем понадкібернетичного рівня , закономірності якого не можуть бути достатньо точно описані на мові статистичних або інших аналітичних математичних моделей. Дані у вказаних областях неоднорідні, гетерогенні, нестаціонарні і часто відрізняються високою розмірністю.

Статистичні пакети

Останні версії майже всіх відомих статистичних пакетів включають разом з традиційними статистичними методами також елементи Data Mining. Але основна увага в них приділяється все ж таки класичним методикам - кореляційному, регресійному, факторному аналізу і іншим. Найсвіжіший детальний огляд пакетів для статистичного аналізу приведений на сторінках Інтернету http://is1.cemi.rssi.ru/ruswin/index.htm. Недоліком систем цього класу вважають вимогу до спеціальної підготовки користувача. Також відзначають, що могутні сучасні статистичні пакети є дуже "ваговитими" для масового застосування у фінансах і бізнесі. До того ж часто ці системи вельми дорогі - від $1000 до $15000.

Є ще серйозніший принциповий недолік статистичних пакетів, що обмежує їх застосування в Data Mining. Більшість методів, що входять до складу пакетів, спираються на статистичну парадигму, в якій головними фігурантами служать усереднені характеристики вибірки. А ці характеристики, як вказувалося вище, при дослідженні реальних складних життєвих феноменів часто є фіктивними величинами.

Розглянемо основні завдання, які успішно вирішуються з використанням інструментів DataMining.

Аналіз кредитного ризику
<http://www.inftech.webservis.ru/it/database/datamining/ar5.html>

Залучення і утримання клієнтів
<http://www.inftech.webservis.ru/it/database/datamining/ar5.html>

Прогнозування змін клієнтури
<http://www.inftech.webservis.ru/it/database/datamining/ar5.html>

Виявлення совокупностей банківських продуктів, що набувають клієнтами, і послуг
<http://www.inftech.webservis.ru/it/database/datamining/ar5.html>

Прогнозування залишку на рахунках клієнтів
<http://www.inftech.webservis.ru/it/database/datamining/ar5.html>

Управління портфелем цінних паперів
<http://www.inftech.webservis.ru/it/database/datamining/ar5.html>

Виявлення випадків шахрайства з кредитними картками
<http://www.inftech.webservis.ru/it/database/datamining/ar5.html>

Оцінка прибутковості інвестиційних проектів
<http://www.inftech.webservis.ru/it/database/datamining/ar5.html>

Оцінка інтенсивності конкуренції і найближчих конкурентів
<http://www.inftech.webservis.ru/it/database/datamining/ar5.html>

Профілізація якнайкращих досягнень
<http://www.inftech.webservis.ru/it/database/datamining/ar5.html>

Підвищення якості архівної фінансової інформації
<http://www.inftech.webservis.ru/it/database/datamining/ar5.html>

Верифікація даних по курсах валют
<http://www.inftech.webservis.ru/it/database/datamining/ar5.html>

8.2. ІАД та український ринок

На українському ринку технології інтелектуальних обчислень роблять лише перші кроки. Це можна пояснити їх високою вартістю, але, як показує історія розвитку інших галузей комп'ютерного ринку України, сам по собі цей фактор навряд чи є визначальним. Скоріше тут виявляється дія деяких специфічних для України негативних факторів, що різко зменшують ефективність застосування аналітичних технологій. Постараємося визначити ці фактори, проаналізувати ступінь притаманних їм різних класів систем інтелектуального аналізу даних та обчислень, а також виділити властивості таких систем, що полегшують українським покупцям їх застосування.

Почнемо з характеристики української специфіки. Комп'ютерні системи підтримки прийняття рішень, у принципі, можуть ґрунтуватися на двох підходах. Перший, більш традиційний, полягає в тому, що в системі фіксується досвід експерта, який використовується для вироблення, оптимального в даній ситуації, рішення. Системи інтелектуальних обчислень в основному реалізують другий підхід. Вони намагаються знайти рішення на основі аналізу історичних даних, що описують поведінку досліджуваного об'єкта, прийняті в минулому рішення, їхні результати і т.д. Усі ці дані можуть включати, наприклад, часові ряди цін на різні фінансові послуги, результати фінансово-господарської діяльності підприємства, статистику продажів тієї чи іншої продукції. Зрозуміло, щоб застосування цих систем у практиці виявилось виправданим, необхідно мати досить вагому множину цих даних - інакше прийняті на їхній основі рішення будуть безпідставними.

З цією очевидною обставиною зв'язані головні труднощі просування технологій інтелектуальних обчислень в Україні: відмінною рисою більшості вітчизняних підприємств є порівняно невеликий термін існування. Характерний "вік" накопичених ними баз даних складає 2-3 роки, і, як показує досвід, інформації, що міститься в цих базах даних, виявляється недостатньо для вироблення на її основі ефективної стратегії прийняття рішень за допомогою новітніх аналітичних систем. Небезпека тут складається не стільки в

неможливості виявлення цікавих взаємозв'язків у нечисленних даних і побудови моделей на їхній основі, скільки в одержанні статистично незначущих моделей і прийнятті на їхній основі невірних рішень. Якщо даних мало, а їх описова модель складна, то завжди можна підігнати цю модель під дані, навіть якщо це цілком випадкові числа. Той факт, що метод відмінно працює, коли потрібно пояснити те, що було в минулому, але зовсім непридатний для прийняття рішень "на майбутнє", народжує сумнів у здатності систем інтелектуальних обчислень вирішувати реальні задачі зі сфери бізнесу і фінансів. Таким чином, головна проблема застосування систем добування знань для України - це нечисленність аналізованих даних, а одне з головних вимог до цих систем -наявність жорсткого контролю статистичної значимості одержуваних результатів.

Іншою відмітною рисою української економіки, як на макрорівні, так і на рівні окремих підприємств, є її нестабільність; крім того, вона знаходиться під впливом дії численних, зненацька виникаючих факторів.

Нарешті, ще одна обставина впливає на застосування систем інтелектуальних обчислень в українських умовах. Воно зв'язано з тим, що особи, відповідальні за прийняття рішень у бізнесі і фінансах, звичайно не є фахівцями з статистики і штучного інтелекту, тому не можуть безпосередньо використовувати системи інтелектуального аналізу даних, що вимагають складного налаштування чи спеціальної підготовки даних. Якщо така система поставляється як складова частина загальної технології електронних сховищ даних, реалізованої на підприємстві (що стає самою розповсюдженою практикою в розвинутих країнах), то це не створює проблеми - всі налаштування і передпроцесорна обробка здійснюються автоматично. Однак українські підприємства, що використовують сховища даних з елементами інтелектуального аналізу,сьогодні вкрай нечисленні. Тому важливими факторами, що визначають комерційний успіх систем інтелектуального аналізу даних в України, є простота у використанні і високому ступені комп'ютеризації.

Передові підприємці усвідомили, що засоби інтелектуальних обчислень - це реальний спосіб підвищення ефективності роботи. Питання не в тому, чи потрібні нові технології, а в тому, як їх застосувати в кожному конкретному випадку. Витрати на постановку задачі і супровід інтелектуальних систем можуть на порядок перевищувати вартість окремого пакета програм. Очевидно, що варто витратити частину грошей на навчання фахівців - у підсумку вийде дешевше й ефективніше. Зростає роль спеціалізованих консалтингових фірм, що здійснюють комплексний супровід проектів, включаючи діагностику задачі, аналіз методів рішення, вироблення рекомендацій, реалізацію обраного підходу, супровід, оптимізацію тощо.

ЛІТЕРАТУРА:

1. Ситник В.Ф. Системи підтримки прийняття рішень: Навч. посібник. - К.: КНЕУ, 2004. - 614 с.
2. Федоров А., Елманова Н. Введение в OLAP - технологии Microsoft - М.: Диалог - МИФИ, 2002 - 268 с.
3. Дюк В.А. DataMining – состояние проблемы, новые решения. Wysiwyg: //38/ http://www.inftech.webservis.ru/database/datamining/ar1.html.
4. Дюк В.А. DataMining – интеллектуальный анализ данных. Wysiwyg: //18/ http://www.olap.ru/basic/dm2.asp.

5. Кречетов, П. Иванов. Продукти для интеллектуального анализа даних // *ComputerWeek-Москва*. - 1997. - N 14-15. - С. 32-39.

6E. F. Codd, S. B. Codd, C. T. Salley. Providing OLAP (On-Line Analytical Processing) to User-Analysts: An IT Mandate. - E. F. Codd & Associates, 1993.

7. J. Gray, S. Chaudhuri, A. Bosworth, A. Layman, D. Reichart, M. Venkatrao, F. Pellow, H. Pirahesh. Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab, and Sub-Totals // *Data Mining and Knowledge Discovery*. - 1997. - N 1. - P. 29-53.
8. D. Hackathorn. Reinventing Enterprise Systems Via Data Warehousing. - Washington, DC: The Data Warehousing Institute Annual Conference, 1995.
9. Корнеев В.В. и др. Базы данных. Интеллектуальная обработка информации. - М.: Нолидж, 2000. - 352 с.
10. Rob P. and C. Coronell. Database Systems: Design, Implementation, and Management, Course Technology, 1997.
11. Лесник А.А., Мальцев В.Н. Системы поддержки управленческих и проектных решений. - Л.: Машиностроение. Ленингр. отд-е, 1990. - 167 с.
12. Архипенков С.Я. Аналитические системы на базе Oracle Express OLAP. - М.: Диалог - МИФИ, 2000. - 320 с.