Оглавление

Что такое Data Mining?....7
Сравнение статистики, машинного обучения и Data Mining....9
Развитие технологии баз данных....9
Понятие Data Mining....10
Data Mining как часть рынка информационных технологий....11

Данные....17
Что такое данные?....17
Набор данных и их атрибутов....17
Измерения....19
Типы наборов данных....22
Форматы хранения данных....23
Базы данных. Основные положения....24
Классификация видов данных....27
Метаданные....28

Методы и стадии Data Mining....29

Задачи Data Mining. Информация и знания....39
Задачи Data Mining....39
От данных к решениям....43
От задачи к приложению....44
Информация....45
Знания....48
Сопоставление и сравнение понятий "информация", "данные", "знание"....48

Задачи Data Mining. Классификация и кластеризация....50
Задача классификации....50
Процесс классификации....53
Методы, применяемые для решения задач классификации....54
Точность классификации: оценка уровня ошибок....56
Оценивание классификационных методов....56
Задача кластеризации....57
Оценка качества кластеризации....60
Процесс кластеризации....60
Применение кластерного анализа....60
Выводы....61

Задачи Data Mining. Прогнозирование и визуализация....63
Задача прогнозирования....63
Сравнение задач прогнозирования и классификации....64
Прогнозирование и временные ряды....64
Задача визуализации....71

Сферы применения Data Mining....74
Применение Data Mining для решения бизнес-задач....75
Data Mining для научных исследований....79

Основы анализа данных....84
Анализ данных в Microsoft Excel....84
Описательная статистика....84
Корреляционный анализ....88

Регрессионный анализ....90
Выводы....96

Методы классификации и прогнозирования. Деревья решений....97
Преимущества деревьев решений....100
Процесс конструирования дерева решений....101
Алгоритмы....104
Выводы....106

Методы классификации и прогнозирования. Метод опорных векторов. Метод "ближайшего соседа". Байесовская классификация....107
Метод опорных векторов....107
Линейный SVM....108
Метод "ближайшего соседа" или системы рассуждений на основе аналогичных случаев....110
Решение задачи классификации новых объектов....112
Решение задачи прогнозирования....113
Оценка параметра k методом кросс-проверки....114
Байесовская классификация....115
Байесовская фильтрация по словам....116

Методы классификации и прогнозирования. Нейронные сети....118
Элементы нейронных сетей....119
Архитектура нейронных сетей....120
Обучение нейронных сетей....122
Модели нейронных сетей....124
Программное обеспечение для работы с нейронными сетями....126
Пример решения задачи....127
Пакет Matlab....132

Нейронные сети. Самоорганизующиеся карты Кохонена....134
Классификация нейронных сетей....134
Подготовка данных для обучения....135
Выбор структуры нейронной сети....136
Карты Кохонена....136
Пример решения задачи....141
Выводы....146

Методы кластерного анализа. Иерархические методы....147
Методы кластерного анализа....151
Меры сходства....153
Методы объединения или связи....154
Иерархический кластерный анализ в SPSS....155

Методы кластерного анализа. Итеративные методы....159
Алгоритм k-средних (k-means)....159
Алгоритм PAM (partitioning around Medoids)....162
Предварительное сокращение размерности....162
Факторный анализ....162
Итеративная кластеризация в SPSS....163
Сложности и проблемы, которые могут возникнуть при применении кластерного анализа..165
Новые алгоритмы и некоторые модификации алгоритмов кластерного анализа....167

Методы поиска ассоциативных правил....170
Часто встречающиеся приложения с применением ассоциативных правил:....170

Введение в ассоциативные правила....170
Часто встречающиеся шаблоны или образцы....171
Поддержка....172
Характеристики ассоциативных правил....173
Границы поддержки и достоверности ассоциативного правила....173
Методы поиска ассоциативных правил....174
Разновидности алгоритма Apriori....176
Пример решения задачи поиска ассоциативных правил....178

Способы визуального представления данных. Методы визуализации....184
Визуализация инструментов Data Mining....184
Визуализация Data Mining моделей....185
Методы визуализации....186
Представление данных в одном, двух и трех измерениях....186
Представление данных в 4 + измерениях....187
Параллельные координаты....187
"Лица Чернова"....188
Качество визуализации....190
Представление пространственных характеристик....191
Основные тенденции в области визуализации....191
Выводы....194

Комплексный подход к внедрению Data Mining, OLAP и хранилищ данных в СППР....195
Классификация СППР....197
OLAP-системы....198
OLAP-продукты....199
Интеграция OLAP и Data Mining....200
Хранилища данных....201
Преимущества использования хранилищ данных....202

Процесс Data Mining. Начальные этапы....205
Этап 1. Анализ предметной области....205
Этап 2. Постановка задачи....206
Этап 3. Подготовка данных....206
Выводы....214

Процесс Data Mining. Очистка данных....215
Инструменты очистки данных....215
Выводы по подготовке данных....221

Процесс Data Mining. Построение и использование модели....223
Моделирование....223
Виды моделей....224
Математическая модель....226
Этап 4. Построение модели....227
Этап 5. Проверка и оценка моделей....229
Этап 6. Выбор модели....230
Этап 7. Применение модели....230
Этап 8. Коррекция и обновление модели....231
Погрешности в процессе Data Mining....231
Выводы....233

Организационные и человеческие факторы в Data Mining. Стандарты Data Mining....234
Организационные Факторы....234

Человеческие факторы. Роли в Data Mining....235
CRISP-DM методология....238
SEMMA методология....240
Другие стандарты Data Mining....241
Стандарт PMML....241
Стандарты, относящиеся к унификации интерфейсов....242

Рынок инструментов Data Mining....244
Поставщики Data Mining....244
Классификация инструментов Data Mining....250
Программное обеспечение Data Mining для поиска ассоциативных правил....251
Программное обеспечение для решения задач кластеризации и сегментации....252
Программное обеспечение для решения задач классификации....253
Программное обеспечение Data Mining для решения задач оценивания и прогнозирования 253
Выводы....254

Инструменты Data Mining. SAS Enterprise Miner....255
Специализированное хранилище данных....266
Подход SAS к созданию информационно-аналитических систем....266
Технические требования пакета SASR Enterprise Miner....267

Инструменты Data Mining. Система PolyAnalyst....268
Архитектура системы....268
PolyAnalyst Workplace - лаборатория аналитика....269
Аналитический инструментарий PolyAnalyst....269
Алгоритмы кластеризации....271
Алгоритмы классификации....271
Алгоритмы ассоциации....272
Модули текстового анализа....273
Визуализация....274
Эволюционное программирование....275
Общесистемные характеристики PolyAnalyst....276
WebAnalyst....278

Инструменты Data Mining. Программные продукты Cognos и система STATISTICA Data Miner...280
Особенности методологии моделирования с применением Cognos 4Thought....282
Система STATISTICA Data Miner....286
Средства анализа STATISTICA Data Miner....288

Инструменты Oracle Data Mining и Deductor....295
Oracle Data Mining....295
Прогнозирующие модели....297
Дескрипторные модели....297
Аналитическая платформа Deductor....298

Инструмент KXEN....309

Data Mining консалтинг....318
Data Mining-услуги....318
Работа с клиентом....320
Примеры решения....322
Техническое описание решения....323
Выводы....326

Что такое Data Mining?

"За последние годы, когда, стремясь к повышению эффективности и прибыльности бизнеса, при создании БД все стали пользоваться средствами обработки цифровой информации, появился и побочный продукт этой активности - горы собранных данных: И вот все больше распространяется идея о том, что эти горы полны золота".

В прошлом процесс добычи золота в горной промышленности состоял из выбора участка земли и дальнейшего ее просеивания большое количество раз. Иногда искатель находил несколько ценных самородков или мог натолкнуться на золотоносную жилу, но в большинстве случаев он вообще ничего не находил и шел дальше к другому многообещающему месту или же вовсе бросал добывать золото, считая это занятие напрасной тратой времени.

Сегодня появились новые научные методы и специализированные инструменты, сделавшие горную промышленность намного более точной и производительной. Data Mining для данных развилась почти таким же способом. Старые методы, применявшиеся математиками и статистиками, отнимали много времени, чтобы в результате получить конструктивную и полезную информацию.

Сегодня на рынке представлено множество инструментов, включающих различные методы, которые делают Data Mining прибыльным делом, все более доступным для большинства компаний.

Термин Data Mining получил свое название из двух понятий: поиска ценной информации в большой базе данных (data) и добычи горной руды (mining). Оба процесса требуют или просеивания огромного количества сырого материала, или разумного исследования и поиска искомых ценностей.

Термин Data Mining часто переводится как добыча данных, извлечение информации, раскопка данных, интеллектуальный анализ данных, средства поиска закономерностей, извлечение знаний, анализ шаблонов, "извлечение зерен знаний из гор данных", раскопка знаний в базах данных, информационная проходка данных, "промывание" данных. Понятие "обнаружение знаний в базах данных" (Knowledge Discovery in Databases, KDD) можно считать синонимом Data Mining [1].

Понятие Data Mining, появившееся в 1978 году, приобрело высокую популярность в современной трактовке примерно с первой половины 1990-х годов. До этого времени обработка и анализ данных осуществлялся в рамках прикладной статистики, при этом в основном решались задачи обработки небольших баз данных.

О популярности Data Mining говорит и тот факт, что результат поиска термина "Data Mining" в поисковой системе Google (на сентябрь 2005 года) - более 18 миллионов страниц.

Что же такое Data Mining?

Data Mining - мультидисциплинарная область, возникшая и развивающаяся на базе таких наук как прикладная статистика, распознавание образов, искусственный интеллект, теория баз данных и др., см. рис. 1.1.

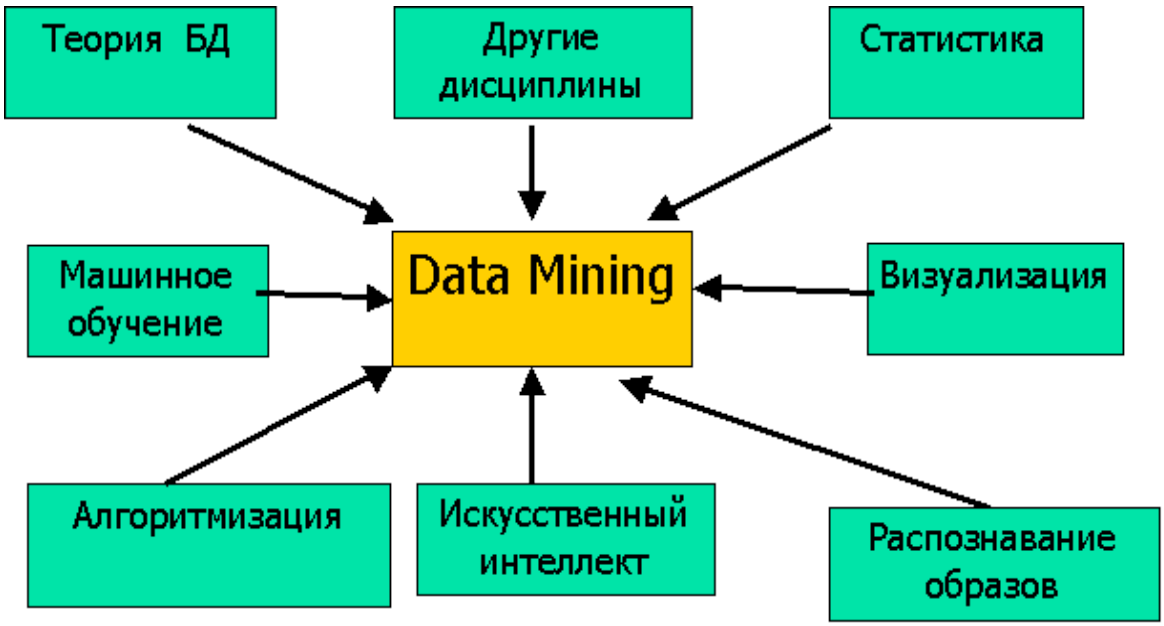

Рис. 1.1. Data Mining как мультидисциплинарная область

Приведем краткое описание некоторых дисциплин, на стыке которых появилась технология Data Mining.

Понятие Статистики

Статистика - это наука о методах сбора данных, их обработки и анализа для выявления закономерностей, присущих изучаемому явлению.

Статистика является совокупностью методов планирования эксперимента, сбора данных, их представления и обобщения, а также анализа и получения выводов на основании этих данных.

Статистика оперирует данными, полученными в результате наблюдений либо экспериментов. Одна из последующих глав будет посвящена понятию данных.

Понятие Машинного обучения

Единого определения машинного обучения на сегодняшний день нет.

Машинное обучение можно охарактеризовать как процесс получения программой новых знаний. Митчелл в 1996 году дал такое определение: "Машинное обучение - это наука, которая изучает компьютерные алгоритмы, автоматически улучшающиеся во время работы".

Одним из наиболее популярных примеров алгоритма машинного обучения являются нейронные сети.

Понятие Искусственного интеллекта

Искусственный интеллект - научное направление, в рамках которого ставятся и решаются задачи аппаратного или программного моделирования видов человеческой деятельности, традиционно считающихся интеллектуальными.

Термин интеллект (intelligence) происходит от латинского intellectus, что означает ум, рассудок, разум, мыслительные способности человека.

Соответственно, искусственный интеллект (AI, Artificial Intelligence) толкуется как свойство автоматических систем брать на себя отдельные функции интеллекта человека. Искусственным интеллектом называют свойство интеллектуальных систем выполнять творческие функции, которые традиционно считаются прерогативой человека.

Каждое из направлений, сформировавших Data Mining, имеет свои особенности. Проведем сравнение с некоторыми из них.

Сравнение статистики, машинного обучения и Data Mining

- Статистика
  - Более, чем Data Mining, базируется на теории.
  - Более сосредотачивается на проверке гипотез.
- Машинное обучение
  - Более эвристично.
  - Концентрируется на улучшении работы агентов обучения.
- Data Mining.
  - Интеграция теории и эвристик.
  - Сконцентрирована на едином процессе анализа данных, включает очистку данных, обучение, интеграцию и визуализацию результатов.

Понятие Data Mining тесно связано с **технологиями баз данных** и понятием данные, которые будут подробно рассмотрены в следующей лекции.

Развитие технологии баз данных

1960-е гг.

В 1968 году была введена в эксплуатацию первая промышленная СУБД система IMS фирмы IBM.

1970-е гг.

В 1975 году появился первый стандарт ассоциации по языкам систем обработки данных - Conference on Data System Languages (CODASYL), определивший ряд фундаментальных понятий в теории систем баз данных, которые до сих пор являются основополагающими для сетевой модели данных. В дальнейшее развитие теории баз данных большой вклад был сделан американским математиком Э.Ф. Коддом, который является создателем реляционной модели данных.

1980-е гг.

В течение этого периода многие исследователи экспериментировали с новым подходом в направлениях структуризации баз данных и обеспечения к ним доступа. Целью этих поисков было получение реляционных прототипов для более простого моделирования данных. В результате, в 1985 году был создан язык, названный SQL. На сегодняшний день практически все СУБД обеспечивают данный интерфейс.

1990-е гг.

Появились специфичные типы данных - "графический образ", "документ", "звук", "карта". Типы данных для времени, интервалов времени, символьных строк с двухбайтовым представлением символов были добавлены в язык SQL. Появились технологии DataMining, хранилища данных, мультимедийные базы данных и web-базы данных.

Возникновение и развитие Data Mining обусловлено различными факторами, основные среди которых являются следующие [2]:

- совершенствование аппаратного и программного обеспечения;
- совершенствование технологий хранения и записи данных;
- накопление большого количества ретроспективных данных;
- совершенствование алгоритмов обработки информации.

Понятие Data Mining

Data Mining - это процесс поддержки принятия решений, основанный на поиске в данных скрытых закономерностей (шаблонов информации) [3].

Технологию Data Mining достаточно точно определяет Григорий Пиатецкий-Шапиро (Gregory Piatetsky-Shapiro) - один из основателей этого направления:

Data Mining - это процесс обнаружения в сырых данных ранее неизвестных, нетривиальных, практически полезных и доступных интерпретации знаний, необходимых для принятия решений в различных сферах человеческой деятельности.

Суть и цель технологии Data Mining можно охарактеризовать так: это технология, которая предназначена для поиска в больших объемах данных неочевидных, объективных и полезных на практике закономерностей.

Неочевидных - это значит, что найденные закономерности не обнаруживаются стандартными методами обработки информации или экспертным путем.

Объективных - это значит, что обнаруженные закономерности будут полностью соответствовать действительности, в отличие от экспертного мнения, которое всегда является субъективным.

Практически полезных - это значит, что выводы имеют конкретное значение, которому можно найти практическое применение.

Знания - совокупность сведений, которая образует целостное описание, соответствующее некоторому уровню осведомленности об описываемом вопросе, предмете, проблеме и т.д.

Использование знаний (knowledge deployment) означает действительное применение найденных знаний для достижения конкретных преимуществ (например, в конкурентной борьбе за рынок).

Приведем еще несколько определений понятия Data Mining.

Data Mining - это процесс выделения из данных неявной и неструктурированной информации и представления ее в виде, пригодном для использования.

Data Mining - это процесс выделения, исследования и моделирования больших объемов данных для обнаружения неизвестных до этого структур (patterns) с целью достижения преимуществ в бизнесе (определение SAS Institute).

Data Mining - это процесс, цель которого - обнаружить новые значимые корреляции, образцы и тенденции в результате просеивания большого объема хранимых данных с использованием методик распознавания образцов плюс применение статистических и математических методов (определение Gartner Group).

В основу технологии Data Mining положена концепция шаблонов (patterns), которые представляют собой закономерности, свойственные подвыборкам данных, кои могут быть выражены в форме, понятной человеку.

"Mining" по-английски означает "добыча полезных ископаемых", а поиск закономерностей в огромном количестве данных действительно сродни этому процессу.

Цель поиска закономерностей - представление данных в виде, отражающем искомые процессы. Построение моделей прогнозирования также является целью поиска закономерностей.

Data Mining как часть рынка информационных технологий

Классификация аналитических систем

Агентство Gartner Group, занимающееся анализом рынков информационных технологий, в 1980-х годах ввело термин "Business Intelligence" (BI), деловой интеллект или бизнес-интеллект. Этот термин предложен для описания различных концепций и методов, которые улучшают бизнес решения путем использования систем поддержки принятия решений.

В 1996 году агентство уточнило определение данного термина.

Business Intelligence - программные средства, функционирующие в рамках предприятия и обеспечивающие функции доступа и анализа информации, которая находится в хранилище данных, а также обеспечивающие принятие правильных и обоснованных управленческих решений.

Понятие BI объединяет в себе различные средства и технологии анализа и обработки данных масштаба предприятия.

На основе этих средств создаются BI-системы, цель которых - повысить качество информации для принятия управленческих решений.

BI-системы также известны под названием Систем Поддержки Принятия Решений (СППР, DSS, Decision Support System). Эти системы превращают данные в информацию, на основе которой можно принимать решения, т.е. поддерживающую принятие решений.

Gartner Group определяет состав рынка систем Business Intelligence как набор программных продуктов следующих классов:

- средства построения хранилищ данных (data warehousing, ХД);
- системы оперативной аналитической обработки (OLAP);
- информационно-аналитические системы (Enterprise Information Systems, EIS);
- средства интеллектуального анализа данных (data mining);
- инструменты для выполнения запросов и построения отчетов (query and reporting tools).

Классификация Gartner базируется на методе функциональных задач, где программные продукты каждого класса выполняют определенный набор функций или операций с использованием специальных технологий.

Мнение экспертов о Data Mining

Приведем несколько кратких цитат [4] наиболее влиятельных членов бизнес-сообществ, которые являются экспертами в этой относительно новой технологии.

Руководство по приобретению продуктов Data Mining (Enterprise Data Mining Buying Guide) компании Aberdeen Group: "Data Mining - технология добычи полезной информации из баз данных. Однако в связи с существенными различиями между инструментами, опытом и финансовым состоянием поставщиков продуктов, предприятиям необходимо тщательно оценивать предполагаемых разработчиков Data Mining и партнеров.

Чтобы максимально использовать мощность масштабируемых инструментов Data Mining коммерческого уровня, предприятию необходимо выбрать, очистить и преобразовать данные, иногда интегрировать информацию, добытую из внешних источников, и установить специальную среду для работы Data Mining алгоритмов.

Результаты Data Mining в большой мере зависят от уровня подготовки данных, а не от "чудесных возможностей" некоего алгоритма или набора алгоритмов. Около 75% работы над Data Mining состоит в сборе данных, который совершается еще до того, как запускаются сами инструменты. Неграмотно применив некоторые инструменты, предприятие может бессмысленно растратить свой потенциал, а иногда и миллионы долларов".

Мнение Херба Эдельштайна (Herb Edelstein), известного в мире эксперта в области Data Mining, Хранилищ данных и CRM: "Недавнее исследование компании Two Crows показало, что Data Mining находится все еще на ранней стадии развития. Многие организации интересуются этой технологией, но лишь некоторые активно внедряют такие проекты. Удалось выяснить еще один важный момент: процесс реализации Data Mining на практике оказывается более сложным, чем ожидается.

IT-команды увлеклись мифом о том, что средства Data Mining просты в использовании. Предполагается, что достаточно запустить такой инструмент на терабайтной базе данных, и моментально появится полезная информация. На самом деле, успешный Data Mining-

проект требует понимания сути деятельности, знания данных и инструментов, а также процесса анализа данных".

Прежде чем использовать технологию Data Mining, необходимо тщательно проанализировать ее проблемы, ограничения и критические вопросы, с ней связанные, а также понять, чего эта технология не может.

Data Mining не может заменить аналитика

Технология не может дать ответы на те вопросы, которые не были заданы. Она не может заменить аналитика, а всего лишь дает ему мощный инструмент для облегчения и улучшения его работы.

Сложность разработки и эксплуатации приложения Data Mining

Поскольку данная технология является мультидисциплинарной областью, для разработки приложения, включающего Data Mining, необходимо задействовать специалистов из разных областей, а также обеспечить их качественное взаимодействие.

Квалификация пользователя

Различные инструменты Data Mining имеют различную степень "дружелюбности" интерфейса и требуют определенной квалификации пользователя. Поэтому программное обеспечение должно соответствовать уровню подготовки пользователя. Использование Data Mining должно быть неразрывно связано с повышением квалификации пользователя. Однако специалистов по Data Mining, которые бы хорошо разбирались в бизнесе, пока еще мало.

Извлечение полезных сведений невозможно без хорошего понимания сути данных

Необходим тщательный выбор модели и интерпретация зависимостей или шаблонов, которые обнаружены. Поэтому работа с такими средствами требует тесного сотрудничества между экспертом в предметной области и специалистом по инструментам Data Mining. Построенные модели должны быть грамотно интегрированы в бизнес-процессы для возможности оценки и обновления моделей. В последнее время системы Data Mining поставляются как часть технологии хранилищ данных.

Сложность подготовки данных

Успешный анализ требует качественной предобработки данных. По утверждению аналитиков и пользователей баз данных, процесс предобработки может занять до 80% процентов всего Data Mining-процесса.

Таким образом, чтобы технология работала на себя, потребуется много усилий и времени, которые уходят на предварительный анализ данных, выбор модели и ее корректировку.

Большой процент ложных, недостоверных или бессмысленных результатов

С помощью Data Mining можно отыскивать действительно очень ценную информацию, которая вскоре даст большие дивиденды в виде финансовой и конкурентной выгоды.

Однако Data Mining достаточно часто делает множество ложных и не имеющих смысла открытий. Многие специалисты утверждают, что Data Mining-средства могут выдавать огромное количество статистически недостоверных результатов. Чтобы этого избежать, необходима проверка адекватности полученных моделей на тестовых данных.

Высокая стоимость

Качественная Data Mining-программа может стоить достаточно дорого для компании. Вариантом служит приобретение уже готового решения с предварительной проверкой его использования, например на демо-версии с небольшой выборкой данных.

Наличие достаточного количества репрезентативных данных

Средства Data Mining, в отличие от статистических, теоретически не требуют наличия строго определенного количества ретроспективных данных. Эта особенность может стать причиной обнаружения недостоверных, ложных моделей и, как результат, принятия на их основе неверных решений. Необходимо осуществлять контроль статистической значимости обнаруженных знаний.

Отличия Data Mining от других методов анализа данных

Традиционные методы анализа данных (статистические методы) и OLAP в основном ориентированы на проверку заранее сформулированных гипотез (verification-driven data mining) и на "грубый" разведочный анализ, составляющий основу оперативной аналитической обработки данных (OnLine Analytical Processing, OLAP), в то время как одно из основных положений Data Mining - поиск неочевидных закономерностей. Инструменты Data Mining могут находить такие закономерности самостоятельно и также самостоятельно строить гипотезы о взаимосвязях. Поскольку именно формулировка гипотезы относительно зависимостей является самой сложной задачей, преимущество Data Mining по сравнению с другими методами анализа является очевидным.

Большинство статистических методов для выявления взаимосвязей в данных используют концепцию усреднения по выборке, приводящую к операциям над несуществующими величинами, тогда как Data Mining оперирует реальными значениями.

OLAP больше подходит для понимания ретроспективных данных, Data Mining опирается на ретроспективные данные для получения ответов на вопросы о будущем.

Перспективы технологии Data Mining

Потенциал Data Mining дает "зеленый свет" для расширения границ применения технологии. Относительно перспектив Data Mining возможны следующие направления развития:

- выделение типов предметных областей с соответствующими им эвристиками, формализация которых облегчит решение соответствующих задач Data Mining, относящихся к этим областям;
- создание формальных языков и логических средств, с помощью которых будет формализованы рассуждения и автоматизация которых станет инструментом решения задач Data Mining в конкретных предметных областях;

- создание методов Data Mining, способных не только извлекать из данных закономерности, но и формировать некие теории, опирающиеся на эмпирические данные;
- преодоление существенного отставания возможностей инструментальных средств Data Mining от теоретических достижений в этой области.

Если рассматривать будущее Data Mining в краткосрочной перспективе, то очевидно, что развитие этой технологии наиболее направлено к областям, связанным с бизнесом.

В краткосрочной перспективе продукты Data Mining могут стать такими же обычными и необходимыми, как электронная почта, и, например, использоваться пользователями для поиска самых низких цен на определенный товар или наиболее дешевых билетов.

В долгосрочной перспективе будущее Data Mining является действительно захватывающим - это может быть поиск интеллектуальными агентами как новых видов лечения различных заболеваний, так и нового понимания природы вселенной.

Однако Data Mining таит в себе и потенциальную опасность - ведь все большее количество информации становится доступным через всемирную сеть, в том числе и сведения частного характера, и все больше знаний возможно добыть из нее:

Не так давно крупнейший онлайновый магазин "Amazon" оказался в центре скандала по поводу полученного им патента "Методы и системы помощи пользователям при покупке товаров", который представляет собой не что иное как очередной продукт Data Mining, предназначенный для сбора персональных данных о посетителях магазина. Новая методика позволяет прогнозировать будущие запросы на основании фактов покупок, а также делать выводы об их назначении. Цель данной методики - то, о чем говорилось выше - получение как можно большего количества информации о клиентах, в том числе и частного характера (пол, возраст, предпочтения и т.д.). Таким образом, собираются данные о частной жизни покупателей магазина, а также членах их семей, включая детей. Последнее запрещено законодательством многих стран - сбор информации о несовершеннолетних возможен там только с разрешения родителей.

Исследования отмечают, что существуют как успешные решения, использующие Data Mining, так и неудачный опыт применения этой технологии [5]. Области, где применения технологии Data Mining, скорее всего, будут успешными, имеют такие особенности:

- требуют решений, основанных на знаниях;
- имеют изменяющуюся окружающую среду;
- имеют доступные, достаточные и значимые данные;
- обеспечивают высокие дивиденды от правильных решений.

Существующие подходы к анализу

Достаточно долго дисциплина Data Mining не признавалась полноценной самостоятельной областью анализа данных, иногда ее называют "задворками статистики" (Pregibon, 1997).

На сегодняшний день определилось несколько точек зрения на Data Mining. Сторонники одной из них считают его миражом, отвлекающим внимание от классического анализа данных. Сторонники другого направления - это те, кто принимает Data Mining как альтернативу традиционному подходу к анализу. Есть и середина, где рассматривается

возможность совместного использования современных достижений в области Data Mining и классическом статистическом анализе данных.

Технология Data Mining постоянно развивается, привлекает к себе все больший интерес как со стороны научного мира, так и со стороны применения достижений технологии в бизнесе.

Ежегодно проводится множество научных и практических конференций, посвященных Data Mining, одна из которых - Международная конференция по Knowledge Discovery Data Mining (International Conferences on Knowledge Discovery and Data Mining).

Среди наиболее известных **WWW-источников** - сайт www.kdnuggets.com, который ведет один из основателей Data Mining Григорий Пиатецкий-Шапиро.

Периодические издания по Data Mining: Data Mining and Knowledge Discovery, KDD Explorations, ACM-TODS, IEEE-TKDE, JIIS, J. ACM, Machine Learning, Artificial Intelligence.

Материалы конференций: ACM-SIGKDD, IEEE-ICDM, SIAM-DM, PKDD, PAKDD, Machine learning (ICML), AAAI, IJCAI, COLT (Learning Theory).

Данные

Что такое данные?

В широком понимании данные представляют собой факты, текст, графики, картинки, звуки, аналоговые или цифровые видео-сегменты.

Данные могут быть получены в результате измерений, экспериментов, арифметических и логических операций.

Данные должны быть представлены в форме, пригодной для хранения, передачи и обработки.

Иными словами, данные - это необработанный материал, предоставляемый поставщиками данных и используемый потребителями для формирования информации на основе данных.

Набор данных и их атрибутов

В таблице 2.1 представлена двухмерная таблица, представляющая собой набор данных.

Таблица 2.1. Двухмерная таблица "объект-атрибут"

	Атрибуты				
Объекты	Код клиента	Возраст	Семейное положение	Доход	Класс
	1	18	Single	125	1
	2	22	Married	100	1
	3	30	Single	70	1
	4	32	Married	120	1
	5	24	Divorced	95	2
	6	25	Married	60	1
	7	32	Divorced	220	1
	8	19	Single	85	2
	9	22	Married	75	1
	10	40	Single	90	2

По горизонтали таблицы располагаются атрибуты объекта или его признаки. По вертикали таблицы - объекты.

Объект описывается как набор атрибутов.

Объект также известен как запись, случай, пример, строка таблицы и т.д.

Атрибут - свойство, характеризующее объект.

Например: цвет глаз человека, температура воды и т.д.

Атрибут также называют переменной, полем таблицы, измерением, характеристикой.

В результате операционализации понятий [6], т.е. перехода от общих категорий к конкретным величинам, получается набор переменных изучаемого понятия.

Переменная (variable) - свойство или характеристика, общая для всех изучаемых объектов, проявление которой может изменяться от объекта к объекту.

Значение (value) переменной является проявлением признака.

При анализе данных, как правило, нет возможности рассмотреть всю интересующую нас совокупность объектов. Изучение очень больших объемов данных является дорогостоящим процессом, требующим больших временных затрат, а также неизбежно приводит к ошибкам, связанным с человеческим фактором.

Вполне достаточно рассмотреть некоторую часть всей совокупности, то есть выборку, и получить интересующую нас информацию на ее основании.

Однако размер выборки должен зависеть от разнообразия объектов, представленных в генеральной совокупности. В выборке должны быть представлены различные комбинации и элементы генеральной совокупности.

Генеральная совокупность (population) - вся совокупность изучаемых объектов, интересующая исследователя.

Выборка (sample) - часть генеральной совокупности, определенным способом отобранная с целью исследования и получения выводов о свойствах и характеристиках генеральной совокупности.

Параметры - числовые характеристики генеральной совокупности.

Статистики - числовые характеристики выборки.

Часто исследования основываются на гипотезах. Гипотезы проверяются с помощью данных. Гипотеза - предположение относительно параметров совокупности объектов, которое должно быть проверено на ее части.

Гипотеза - частично обоснованная закономерность знаний, служащая либо для связи между различными эмпирическими фактами, либо для объяснения факта или группы фактов.

Пример гипотезы: между показателями продолжительности жизни и качеством питания есть связь. В этом случае целью исследования может быть объяснение изменений конкретной переменной, в данном случае - продолжительности жизни. Допустим, существует гипотеза, что **зависимая переменная** (продолжительность жизни) изменяется в **зависимости** от некоторых причин (качество питания, образ жизни, место проживания и т.д.), которые и являются **независимыми переменными**.

Однако переменная изначально не является зависимой или независимой. Она становится таковой после формулировки конкретной гипотезы. Зависимая переменная в одной гипотезе может быть независимой в другой.

Измерения

Измерение - процесс присвоения чисел характеристикам изучаемых объектов согласно определенному правилу.

В процессе подготовки данных измеряется не сам объект, а его характеристики.

Шкала - правило, в соответствии с которым объектам присваиваются числа.

Многие инструменты Data Mining при импорте данных из других источников предлагают выбрать тип шкалы для каждой переменной и/или выбрать тип данных для входных и выходных переменных (символьные, числовые, дискретные и непрерывные). Пользователю такого инструмента необходимо владеть этими понятиями.

Переменные могут являться **числовыми** данными либо **символьными**.

Числовые данные, в свою очередь, могут быть дискретными и непрерывными.

Дискретные данные являются значениями признака, общее число которых конечно либо бесконечно, но может быть подсчитано при помощи натуральных чисел от одного до бесконечности.

Пример дискретных данных. Продолжительность маршрута троллейбуса (количество вариантов продолжительности конечно): 10, 15, 25 мин.

Непрерывные данные - данные, значения которых могут принимать какое угодно значение в некотором интервале. Измерение непрерывных данных предполагает большую точность.

Пример непрерывных данных: температура, высота, вес, длина и т.д.

Шкалы

Существует пять типов шкал измерений: номинальная, порядковая, интервальная, относительная и дихотомическая.

Номинальная шкала (nominal scale) - шкала, содержащая только категории; данные в ней не могут упорядочиваться, с ними не могут быть произведены никакие арифметические действия.

Номинальная шкала состоит из названий, категорий, имен для классификации и сортировки объектов или наблюдений по некоторому признаку.

Пример такой шкалы: профессии, город проживания, семейное положение.

Для этой шкалы применимы только такие операции: равно (=), не равно (≠).

Порядковая шкала (ordinal scale) - шкала, в которой числа присваивают объектам для обозначения относительной позиции объектов, но не величины различий между ними.

Шкала измерений дает возможность ранжировать значения переменных. Измерения же в порядковой шкале содержат информацию только о порядке следования величин, но не позволяют сказать "насколько одна величина больше другой", или "насколько она меньше другой".

Пример такой шкалы: место (1, 2, 3-е), которое команда получила на соревнованиях, номер студента в рейтинге успеваемости (1-й, 23-й, и т.д.), при этом неизвестно, насколько один студент успешней другого, известен лишь его номер в рейтинге.

Для этой шкалы применимы только такие операции: равно (=), не равно (≠), больше (>), меньше (<).

Интервальная шкала (interval scale) - шкала, разности между значениями которой могут быть вычислены, однако их отношения не имеют смысла.

Эта шкала позволяет находить разницу между двумя величинами, обладает свойствами номинальной и порядковой шкал, а также позволяет определить количественное изменение признака.

Пример такой шкалы: температура воды в море утром - 19 градусов, вечером - 24, т.е. вечерняя на 5 градусов выше, но нельзя сказать, что она в 1,26 раз выше.

Номинальная и порядковая шкалы являются дискретными, а интервальная шкала - непрерывной, она позволяет осуществлять точные измерения признака и производить арифметические операции сложения, вычитания, умножения, деления.

Для этой шкалы применимы только такие операции: равно (=), не равно (≠), больше (>), меньше (<), операции сложения (+) и вычитания (-).

Относительная шкала (ratio scale) - шкала, в которой есть определенная точка отсчета и возможны отношения между значениями шкалы.

Пример такой шкалы: вес новорожденного ребенка (4 кг и 3 кг). Первый в 1,33 раза тяжелее.

Цена на картофель в супермаркете выше в 1,2 раза, чем цена на базаре.

Относительные и интервальные шкалы являются числовыми.

Для этой шкалы применимы только такие операции: равно (=), не равно (≠), больше (>), меньше (<), операции сложения (+) и вычитания (-), умножения (*) и деления (/).

Дихотомическая шкала (dichotomous scale) - шкала, содержащая только две категории.

Пример такой шкалы: пол (мужской и женский).

Пример использования разных шкал для измерений свойств различных объектов, в данном случае температурных условий, приведен в таблице данных, изображенной в таблице 2.2.

Таблица 2.2. Множество измерений свойств различных объектов

Номер объекта	Профессия (номинальная шкала)	Средний балл (интервальная шкала)	Образование (порядковая шкала)
1	слесарь	22	среднее
2	ученый	55	высшее
3	учитель	47	высшее

Пример использования различных шкал для измерений свойств одной системы, в данном случае температурных условий, приведен в таблице данных, изображенной в таблице 2.3.

Таблица 2.3. Множество измерений свойств одной системы

Дата измерения	Облачность (номинальная шкала)	Температура в 8 часов утра (интервальная шкала)	Сила ветра (порядковая шкала)
1 сентября	облачно	22º С	Ветер сильный
2 сентября	пасмурно	17º С	Ветер слабый
3 сентября	ясно	23º С	Ветер очень сильный

Выводы. В этой части лекции мы рассмотрели понятие данных, объекта и атрибута, их характеристики.

Также мы обсудили типы шкал. Номинальная шкала описывает объекты или наблюдения в терминах качественных признаков. На один шаг далее идут порядковые шкалы, позволяющие упорядочивать наблюдения или объекты по определенной характеристике. Интервальные и относительные шкалы более сложны, в них возможно определение количественного значения признака.

Типы наборов данных

Данные, состоящие из записей

Наиболее часто встречающиеся данные - данные, состоящие из записей (record data) [7]. Примеры таких наборов данных: табличные данные, матричные данные, документальные данные, транзакционные или операционные.

Табличные данные - данные, состоящие из записей, каждая из которых состоит из фиксированного набора атрибутов.

Транзакционные данные представляют собой особый тип данных, где каждая запись, являющаяся транзакцией, включает набор значений.

Пример транзакционной базы данных, содержащей перечень покупок клиентов магазина, приведен на рис. 2.1.

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

Рис. 2.1. Пример транзакционных данных

Графические данные

Примеры графических данных: WWW-данные; молекулярные структуры; графы (рис. 2.2); карты.

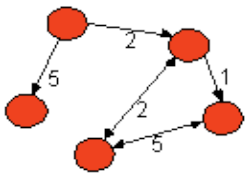

Рис. 2.2. Пример графа

С помощью карт, например, можно отследить изменения объектов во времени и пространстве, определить характер их распределения на плоскости или в пространстве. Преимуществом графического представления данных является большая простота их восприятия, чем, например, табличных данных.

Пример карты, являющейся картой Кохонена (моделью нейронных сетей, которые будут рассмотрены в одной из лекций нашего курса), представлен на рис. 2.3.

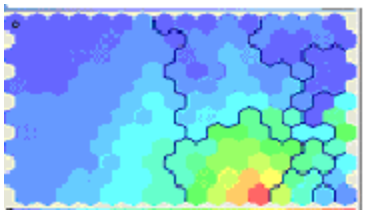

Рис. 2.3. Пример данных типа "Карта Кохонена"

Химические данные

Химические данные представляют собой особый тип данных. Пример таких данных: Benzene Molecule: C_6H_6 (рис. 2.4)

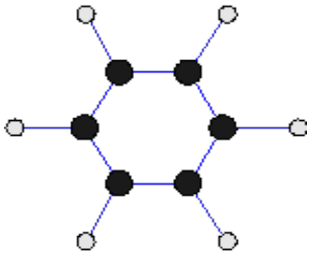

Рис. 2.4. Пример химических данных

Согласно опросу на сайте Kdnuggets, www.kdnuggets.com (апрель, 2004 г.) **"Типы анализируемых данных"**, наибольшее число опрошенных анализирует данные из "плоских" (flat table) и реляционных таблиц (26% и 24% соответственно), далее идут временные ряды (14%) и данные в виде текста (11%).

Остальные анализируемые типы данных в порядке убывания: web-контенты, XML, графика, аудио, видео и др.

Здесь и в следующих лекциях приводятся результаты опросов, проведенных на сайте Kdnuggets, который признан одним из наиболее авторитетных и известных сайтов в сфере Data Mining.

Форматы хранения данных

Одна из основных особенностей данных современного мира состоит в том, что их становится очень много. Возможны четыре аспекта работы с данными: определение данных, вычисление, манипулирование и обработка (сбор, передача и др.).

При манипулировании данными используется структура данных типа "файл". Файлы могут иметь различные форматы.

Как уже было отмечено ранее, большинство инструментов Data Mining позволяют импортировать данные из различных источников, а также экспортировать результирующие данные в различные форматы.

Данные для экспериментов удобно хранить в каком-то одном формате.

В некоторых инструментах Data Mining эти процедуры называются импорт/экспорт данных, другие позволяют напрямую открывать различные источники данных и сохранять результаты Data Mining в одном из предложенных форматов.

Наиболее распространенные форматы, согласно опросу "Форматы хранения данных", представлены на рис. 2.5.

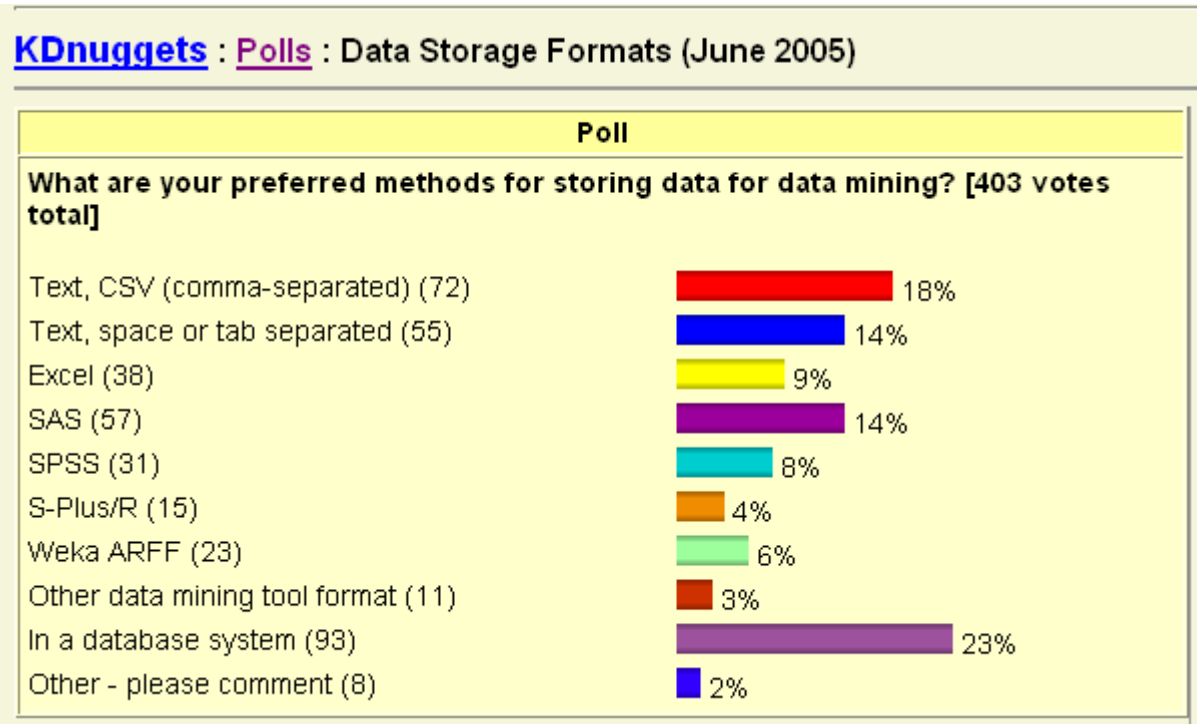

Рис. 2.5. Наиболее распространенные форматы хранения данных

Наибольшее число опрошенных (23%) предпочитают хранить данные в формате той базы данных, которую они используют. В формате Text, CSV - 18%, по 14% опрошенных хранят данные в формате Text, space or tab separated и SAS; в формате Excel - 9%, SPSS - 8%, S-Plus/R - 4%, Weka ARFF - 6%, в других форматах инструментов Data Mining - 2%.

Как видим из результатов опроса, наиболее распространенным форматом хранения данных для Data Mining выступают базы данных.

Базы данных. Основные положения

Для понимания организации данных в базе данных необходимо знание основных положений теории баз данных. Рассмотрим некоторые положения этой теории.

База данных (Database) - это особым образом организованные и хранимые в электронном виде данные.

Особым образом организованные означает, что данные организованы неким конкретным способом, способным облегчить их поиск и доступ к ним для одного или нескольких

приложений. Также такая организация данных предусматривает наличие минимальной избыточности данных.

Базы данных являются одной из разновидностей информационных технологий, а также формой хранения данных.

Целью создания баз данных является построение такой системы данных, которая бы не зависела от программного обеспечения, применяемых технических средств и физического расположения данных в ЭВМ. Построение такой системы данных должно обеспечивать непротиворечивую и целостную информацию. При проектировании базы данных предполагается многоцелевое ее использование.

База данных в простейшем случае представляется в виде системы двумерных таблиц.

Схема данных - описание логической структуры данных, специфицированное на языке описания данных и обрабатываемое СУБД.

Схема пользователя - зафиксированный для конкретного пользователя один вариант порядка полей таблицы.

Системы управления базами данных, СУБД

Система управления базой данных - это программное обеспечение, контролирующее организацию, хранение, целостность, внесение изменений, чтение и безопасность информации в базе данных.

СУБД (Database Management System, DBMS) представляет собой оболочку, с помощью которой при организации структуры таблиц и заполнения их данными получается та или иная база данных.

Система управления **реляционными базами данных** (Relational Database Management System) - это СУБД, основанная на реляционной модели данных.

В реляционной модели данных любое представление данных сводится к совокупности реляционных таблиц (двумерных таблиц особого типа). Системы управления реляционными базами данных используются для построения хранилищ данных.

СУБД имеет программные, технические и организационные составляющие.

Программные средства включают систему управления, обеспечивающую ввод-вывод, обработку и хранение информации, создание, модификацию и тестирование базы данных. Внутренними языками программирования СУБД являются языки четвертого поколения (C, C++, Pascal, Object Pascal). С помощью языков БД создаются приложения, базы данных и интерфейс пользователя, включающий экранные формы, меню, отчеты.

Аналитику при необходимости работы с конкретной СУБД, в частности, при экспорте данных в среду инструмента Data Mining, следует изучить особенности этой СУБД. Так, например, в базе данных СУБД FoxPro все таблицы и представления базы данных физически хранятся в отдельных файлах, которые объединяются в одном проекте. В СУБД Access все таблицы базы данных хранятся в одном файле.

Для работы с конкретной базой данных, в том числе с целью анализа, аналитику желательно знать описание всех таблиц и их структур (атрибутов, типов данных), количество записей в таблице, а также связи между таблицами. Иногда для этих целей используется словарь данных.

К базам данных, а также к СУБД предъявляются такие требования:

1. высокое быстродействие;
2. простота обновления данных;
3. независимость данных;
4. возможность многопользовательского использования данных;
5. безопасность данных;
6. стандартизация построения и эксплуатации БД (фактически СУБД);
7. адекватность отображения данных соответствующей предметной области;
8. дружелюбный интерфейс пользователя.

Высокое быстродействие предусматривает малое время отклика, т.е. малый промежуток времени от момента запроса к базе данных до момента реального получения данных.

Независимость данных - это возможность изменения логической и физической структуры базы данных без изменения представлений пользователей.

Независимость данных обеспечивает минимальные изменения структуры базы данных при изменениях стратегии доступа к данным и структуры самих исходных данных. Эти изменения должны быть предусмотрены на этапах концептуального и логического проектирования базы данных с обеспечением минимальных изменений на этапе физического ее проектирования.

Безопасность данных - это защита данных от преднамеренного или непреднамеренного нарушения секретности, искажения или разрушения. Безопасность включает два компонента: целостность и защиту данных от несанкционированного доступа.

Целостность данных - устойчивость хранимых данных к разрушению и уничтожению, связанным с неисправностями технических средств, системными ошибками и ошибочными действиями пользователей.

Целостность данных - точность и валидность данных. Целостность данных предполагает: отсутствие неточно введенных данных, защиту от ошибок при обновлении баз данных; невозможность удаления (или каскадное удаление) связанных данных разных таблиц; сохранность данных при сбоях техники (возможность восстановление данных) и др.

Защита данных от несанкционированного доступа предполагает ограничение доступа к определенным данным базы и достигается введением мер безопасности: разграничение прав доступа к данным различных пользователей в зависимости от выполняемых ими функций и/или должностных обязанностей; введением защиты в виде паролей; использованием представлений, т.е. таблиц, которые являются производными от исходных и предназначены для работы конкретных пользователей для решения конкретных задач.

Стандартизация обеспечивает преемственность поколений конкретной СУБД, упрощает взаимодействие баз данных одного поколения СУБД с одинаковыми и различными моделями данных.

СУБД отвечает за обработку запросов к базе данных и получение ответа. Способы хранения данных могут быть различными: модель данных может быть как реляционной, так и многомерной, сетевой или иерархической.

Классификация видов данных

Какими могут быть данные? Ниже приведено несколько классификаций.

Реляционные данные - это данные из реляционных баз (таблиц).

Многомерные данные - это данные, представленные в кубах OLAP.

Измерение (dimension) или ось - в многомерных данных - это собрание данных одного и того же типа, что позволяет структурировать многомерную базу данных.

По критерию постоянства своих значений в ходе решения задачи данные могут быть:

- переменными;
- постоянными;
- условно-постоянными.

Переменные данные - это такие данные, которые изменяют свои значения в процессе решения задачи.

Постоянные данные - это такие данные, которые сохраняют свои значения в процессе решения задачи (математические константы, координаты неподвижных объектов) и не зависят от внешних факторов.

Условно-постоянные данные - это такие данные, которые могут иногда изменять свои значения, но эти изменения не зависят от процесса решения задачи, а определяются внешними факторами.

Данные, в зависимости от тех функций, которые они выполняют, могут быть **справочными**, **оперативными**, **архивными**.

Следует различать данные за период и точечные данные. Эти различия важны при проектировании системы сбора информации, а также в процессе измерений.

- данные за период;
- точечные данные.

Данные за период характеризуют некоторый период времени. Примером данных за период могут быть: прибыль предприятия за месяц, средняя температура за месяц.

Точечные данные представляют значение некоторой переменной в конкретный момент времени. Пример точечных данных: остаток на счете на первое число месяца, температура в восемь часов утра.

Данные бывают первичными и вторичными. **Вторичные данные** - это данные, которые являются результатом определенных вычислений, примененных к **первичным данным**. Вторичные данные, как правило, приводят к ускоренному получению ответа на запрос пользователя за счет увеличения объема хранимой информации.

Метаданные

В завершение лекции о данных рассмотрим понятие метаданных.

Метаданные (Metadata) - это данные о данных.

В состав метаданных могут входить: каталоги, справочники, реестры.

Метаданные содержат сведения о составе данных, содержании, статусе, происхождении, местонахождении, качестве, форматах и формах представления, условиях доступа, приобретения и использования, авторских, имущественных и смежных с ними правах на данные и др.

Метаданные - важное понятие в управлении хранилищем данных.

Метаданные, применяемые при управлении хранилищем, содержат информацию, необходимую для его настройки и использования. Различают бизнес-метаданные и оперативные метаданные.

Бизнес-метаданные содержат бизнес-термины и определения, принадлежность данных и правила оплаты услуг хранилища.

Оперативные метаданные - это информация, собранная во время работы хранилища данных:

- происхождение перенесенных и преобразованных данных;
- статус использования данных (активные, архивированные или удаленные);
- данные мониторинга, такие как статистика использования, сообщения об ошибках и т.д.

Метаданные хранилища обычно размещаются в репозитории. Это позволяет использовать метаданные совместно различным инструментам, а также процессам при проектировании, установке, эксплуатации и администрировании хранилища.

Выводы. В лекции были рассмотрены понятие данных, объектов и атрибутов, их характеристики, типы шкал, понятие набора данных и его типы. Описаны возможные форматы хранения данных. Введены понятия базы данных, системы управления базами данных, метаданных.

Методы и стадии Data Mining

Основная особенность Data Mining - это сочетание широкого математического инструментария (от классического статистического анализа до новых кибернетических методов) и последних достижений в сфере информационных технологий. В технологии Data Mining гармонично объединились строго формализованные методы и методы неформального анализа, т.е. количественный и качественный анализ данных.

К методам и алгоритмам Data Mining относятся следующие: искусственные нейронные сети, деревья решений, символьные правила, методы ближайшего соседа и k-ближайшего соседа, метод опорных векторов, байесовские сети, линейная регрессия, корреляционно-регрессионный анализ; иерархические методы кластерного анализа, неиерархические методы кластерного анализа, в том числе алгоритмы k-средних и k-медианы; методы поиска ассоциативных правил, в том числе алгоритм Apriori; метод ограниченного перебора, эволюционное программирование и генетические алгоритмы, разнообразные методы визуализации данных и множество других методов.

Большинство аналитических методов, используемые в технологии Data Mining - это известные математические алгоритмы и методы. Новым в их применении является возможность их использования при решении тех или иных конкретных проблем, обусловленная появившимися возможностями технических и программных средств. Следует отметить, что большинство методов Data Mining были разработаны в рамках теории искусственного интеллекта.

Метод (method) представляет собой норму или правило, определенный путь, способ, прием решений задачи теоретического, практического, познавательного, управленческого характера.

Понятие алгоритма появилось задолго до создания электронных вычислительных машин. Сейчас алгоритмы являются основой для решения многих прикладных и теоретических задач в различных сферах человеческой деятельности, в большинстве - это задачи, решение которых предусмотрено с использованием компьютера.

Алгоритм (algorithm) - точное предписание относительно последовательности действий (шагов), преобразующих исходные данные в искомый результат.

Классификация стадий Data Mining

Data Mining может состоять из двух [8] или трех стадий [9]:

Стадия 1. Выявление закономерностей (свободный поиск).

Стадия 2. Использование выявленных закономерностей для предсказания неизвестных значений (прогностическое моделирование).

В дополнение к этим стадиям иногда вводят стадию валидации [10], следующую за стадией свободного поиска. Цель валидации - проверка достоверности найденных закономерностей. Однако, мы будем считать валидацию частью первой стадии, поскольку

в реализации многих методов, в частности, нейронных сетей и деревьев решений, предусмотрено деление общего множества данных на обучающее и проверочное, и последнее позволяет проверять достоверность полученных результатов.

Стадия 3. Анализ исключений - стадия предназначена для выявления и объяснения аномалий, найденных в закономерностях.

Итак, процесс Data Mining может быть представлен рядом таких последовательных стадий [11]:

СВОБОДНЫЙ ПОИСК (в том числе ВАЛИДАЦИЯ) ->

-> ПРОГНОСТИЧЕСКОЕ МОДЕЛИРОВАНИЕ ->

-> АНАЛИЗ ИСКЛЮЧЕНИЙ

1. Свободный поиск (Discovery)

На стадии свободного поиска осуществляется исследование набора данных с целью поиска скрытых закономерностей. Предварительные гипотезы относительно вида закономерностей здесь не определяются.

Закономерность (law) - существенная и постоянно повторяющаяся взаимосвязь, определяющая этапы и формы процесса становления, развития различных явлений или процессов.

Система Data Mining на этой стадии определяет шаблоны, для получения которых в системах OLAP, например, аналитику необходимо обдумывать и создавать множество запросов. Здесь же аналитик освобождается от такой работы - шаблоны ищет за него система. Особенно полезно применение данного подхода в сверхбольших базах данных, где уловить закономерность путем создания запросов достаточно сложно, для этого требуется перепробовать множество разнообразных вариантов.

Свободный поиск представлен такими **действиями**:

- выявление закономерностей условной логики (conditional logic);
- выявление закономерностей ассоциативной логики (associations and affinities);
- выявление трендов и колебаний (trends and variations).

Допустим, имеется база данных кадрового агентства с данными о профессии, стаже, возрасте и желаемом уровне вознаграждения. В случае самостоятельного задания запросов аналитик может получить приблизительно такие результаты: средний желаемый уровень вознаграждения специалистов в возрасте от 25 до 35 лет равен 1200 условных единиц. В случае свободного поиска система сама ищет закономерности, необходимо лишь задать целевую переменную. В результате поиска закономерностей система сформирует набор логических правил "если ..., то ...".

Могут быть найдены, например, такие закономерности "**Если** возраст < 20 лет и желаемый уровень вознаграждения > 700 условных единиц, то в 75% случаев соискатель ищет работу программиста" или "**Если** возраст >35 лет и желаемый уровень

вознаграждения > 1200 условных единиц, то в 90% случаев соискатель ищет руководящую работу". Целевой переменной в описанных правилах выступает профессия.

При задании другой целевой переменной, например, возраста, получаем такие правила: "**Если** соискатель ищет руководящую работу и его стаж > 15 лет, то возраст соискателя > 35 лет в 65 % случаев".

Описанные действия, в рамках стадии свободного поиска, выполняются при помощи :

- индукции правил условной логики (задачи классификации и кластеризации, описание в компактной форме близких или схожих групп объектов);
- индукции правил ассоциативной логики (задачи ассоциации и последовательности и извлекаемая при их помощи информация);
- определения трендов и колебаний (исходный этап задачи прогнозирования).

На стадии свободного поиска также должна осуществляться валидация закономерностей, т.е. проверка их достоверности на части данных, которые не принимали участие в формировании закономерностей. Такой прием разделения данных на обучающее и проверочное множество часто используется в методах нейронных сетей и деревьев решений и будет описан в соответствующих лекциях.

2. Прогностическое моделирование (Predictive Modeling)

Вторая стадия Data Mining - прогностическое моделирование - использует результаты работы первой стадии. Здесь обнаруженные закономерности используются непосредственно для прогнозирования.

Прогностическое моделирование включает такие **действия**:

- предсказание неизвестных значений (outcome prediction);
- прогнозирование развития процессов (forecasting).

В процессе прогностического моделирования решаются задачи классификации и прогнозирования.

При решении задачи классификации результаты работы первой стадии (индукции правил) используются для отнесения нового объекта, с определенной уверенностью, к одному из известных, предопределенных классов на основании известных значений.

При решении задачи прогнозирования результаты первой стадии (определение тренда или колебаний) используются для предсказания неизвестных (пропущенных или же будущих) значений целевой переменной (переменных).

Продолжая рассмотренный пример первой стадии, можем сделать следующий вывод.

Зная, что соискатель ищет руководящую работу и его стаж > 15 лет, на 65 % можно быть уверенным в том, что возраст соискателя > 35 лет. Или же, если возраст соискателя > 35 лет и желаемый уровень вознаграждения > 1200 условных единиц, на 90% можно быть уверенным в том, что соискатель ищет руководящую работу.

Сравнение свободного поиска и прогностического моделирования с точки зрения логики

Свободный поиск раскрывает общие закономерности. Он по своей природе индуктивен. Закономерности, полученные на этой стадии, формируются от частного к общему. В результате мы получаем некоторое общее знание о некотором классе объектов на основании исследования отдельных представителей этого класса.

Правило: "Если возраст соискателя < 20 лет и желаемый уровень вознаграждения > 700 условных единиц, то в 75% случаев соискатель ищет работу программиста"

На основании частного, т.е. информации о некоторых свойствах класса "возраст < 20 лет" и "желаемый уровень вознаграждения > 700 условных единиц", мы делаем вывод об общем, а именно: соискатели - программисты.

Прогностическое моделирование, напротив, дедуктивно. Закономерности, полученные на этой стадии, формируются от общего к частному и единичному. Здесь мы получаем новое знание о некотором объекте или же группе объектов на основании:

- знания класса, к которому принадлежат исследуемые объекты;
- знание общего правила, действующего в пределах данного класса объектов.

Знаем, что соискатель ищет руководящую работу и его стаж > 15 лет, на 65% можно быть уверенным в том, что возраст соискателя > 35 лет.

На основании некоторых общих правил, а именно: цель соискателя - руководящая работа и его стаж > 15 лет, мы делаем вывод о единичном - возраст соискателя > 35 лет.

Следует отметить, что полученные закономерности, а точнее, их конструкции, могут быть прозрачными, т.е. допускающими толкование аналитика (рассмотренные выше правила), и непрозрачными, так называемыми "черными ящиками". Типичный пример последней конструкции - нейронная сеть.

3. Анализ исключений (forensic analysis)

На третьей стадии Data Mining анализируются исключения или аномалии, выявленные в найденных закономерностях.

Действие, выполняемое на этой стадии, - выявление отклонений (deviation detection). Для выявления отклонений необходимо определить норму, которая рассчитывается на стадии свободного поиска.

Вернемся к одному из примеров, рассмотренному выше.

Найдено правило "Если возраст > 35 лет и желаемый уровень вознаграждения > 1200 условных единиц, то в 90 % случаев соискатель ищет руководящую работу". Возникает вопрос - к чему отнести оставшиеся 10 % случаев?

Здесь возможно два варианта. Первый из них - существует некоторое логическое объяснение, которое также может быть оформлено в виде правила. Второй вариант для оставшихся 10% - это ошибки исходных данных. В этом случае стадия анализа исключений может быть использована в качестве очистки данных [12].

Классификация методов Data Mining

Далее мы рассмотрим несколько известных классификаций методов Data Mining по различным признакам.

Классификация технологических методов Data Mining

Все методы Data Mining подразделяются на две большие группы по принципу работы с исходными обучающими данными. В этой классификации верхний уровень определяется на основании того, сохраняются ли данные после Data Mining либо они дистиллируются для последующего использования.

1. Непосредственное использование данных, или сохранение данных.

В этом случае исходные данные хранятся в явном детализированном виде и непосредственно используются на стадиях прогностического моделирования и/или анализа исключений. Проблема этой группы методов - при их использовании могут возникнуть сложности анализа сверхбольших баз данных.

Методы этой группы: кластерный анализ, метод ближайшего соседа, метод k-ближайшего соседа, рассуждение по аналогии.

2. Выявление и использование формализованных закономерностей, или дистилляция шаблонов.

При технологии дистилляции шаблонов один образец (шаблон) информации извлекается из исходных данных и преобразуется в некие формальные конструкции, вид которых зависит от используемого метода Data Mining. Этот процесс выполняется на стадии свободного поиска, у первой же группы методов данная стадия в принципе отсутствует. На стадиях прогностического моделирования и анализа исключений используются результаты стадии свободного поиска, они значительно компактнее самих баз данных. Напомним, что конструкции этих моделей могут быть трактуемыми аналитиком либо нетрактуемыми ("черными ящиками").

Методы этой группы: логические методы; методы визуализации; методы кросс-табуляции; методы, основанные на уравнениях.

Логические методы, или методы логической индукции, включают: нечеткие запросы и анализы; символьные правила; деревья решений; генетические алгоритмы.

Методы этой группы являются, пожалуй, наиболее интерпретируемыми - они оформляют найденные закономерности, в большинстве случаев, в достаточно прозрачном виде с точки зрения пользователя. Полученные правила могут включать непрерывные и дискретные переменные. Следует заметить, что деревья решений могут быть легко преобразованы в наборы символьных правил путем генерации одного правила по пути от корня дерева до его терминальной вершины. Деревья решений и правила фактически являются разными способами решения одной задачи и отличаются лишь по своим возможностям. Кроме того, реализация правил осуществляется более медленными алгоритмами, чем индукция деревьев решений.

Методы кросс-табуляции: агенты, баесовские (доверительные) сети, кросс-табличная визуализация. Последний метод не совсем отвечает одному из свойств Data Mining - самостоятельному поиску закономерностей аналитической системой. Однако, предоставление информации в виде кросс-таблиц обеспечивает реализацию основной задачи Data Mining - поиск шаблонов, поэтому этот метод можно также считать одним из методов Data Mining [13].

Методы на основе уравнений.

Методы этой группы выражают выявленные закономерности в виде математических выражений - уравнений. Следовательно, они могут работать лишь с численными переменными, и переменные других типов должны быть закодированы соответствующим образом. Это несколько ограничивает применение методов данной группы, тем не менее они широко используются при решении различных задач, особенно задач прогнозирования.

Основные методы данной группы: статистические методы и нейронные сети

Статистические методы наиболее часто применяются для решения задач прогнозирования. Существует множество методов статистического анализа данных, среди них, например, корреляционно-регрессионный анализ, корреляция рядов динамики, выявление тенденций динамических рядов, гармонический анализ.

Другая классификация разделяет все многообразие методов Data Mining на две группы: статистические и кибернетические методы. Эта схема разделения основана на различных подходах к обучению математических моделей [14].

Следует отметить, что существует два подхода отнесения статистических методов к Data Mining. Первый из них противопоставляет статистические методы и Data Mining, его сторонники считают классические статистические методы отдельным направлением анализа данных. Согласно второму подходу, статистические методы анализа являются частью математического инструментария Data Mining. Большинство авторитетных источников придерживается второго подхода [5, 14].

В этой классификации различают две группы методов:

- статистические методы, основанные на использовании усредненного накопленного опыта, который отражен в ретроспективных данных;
- кибернетические методы, включающие множество разнородных математических подходов.

Недостаток такой классификации: и статистические, и кибернетические алгоритмы тем или иным образом опираются на сопоставление статистического опыта с результатами мониторинга текущей ситуации.

Преимуществом такой классификации является ее удобство для интерпретации - она используется при описании математических средств современного подхода к извлечению знаний из массивов исходных наблюдений (оперативных и ретроспективных), т.е. в задачах Data Mining.

Рассмотрим подробнее представленные выше группы.

Статистические методы Data mining

В [14] эти методы представляют собой четыре взаимосвязанных раздела:

- предварительный анализ природы статистических данных (проверка гипотез стационарности, нормальности, независимости, однородности, оценка вида функции распределения, ее параметров и т.п.);
- выявление связей и закономерностей (линейный и нелинейный регрессионный анализ, корреляционный анализ и др.);
- многомерный статистический анализ (линейный и нелинейный дискриминантный анализ, кластерный анализ, компонентный анализ, факторный анализ и др.);
- динамические модели и прогноз на основе временных рядов.

Арсенал статистических методов Data Mining классифицирован на четыре группы методов:

1. Дескриптивный анализ и описание исходных данных.
2. Анализ связей (корреляционный и регрессионный анализ, факторный анализ, дисперсионный анализ).
3. Многомерный статистический анализ (компонентный анализ, дискриминантный анализ, многомерный регрессионный анализ, канонические корреляции и др.).
4. Анализ временных рядов (динамические модели и прогнозирование).

Кибернетические методы Data Mining

Второе направление Data Mining - это множество подходов, объединенных идеей компьютерной математики и использования теории искусственного интеллекта.

К этой группе относятся такие методы:

- искусственные нейронные сети (распознавание, кластеризация, прогноз);
- эволюционное программирование (в т.ч. алгоритмы метода группового учета аргументов);
- генетические алгоритмы (оптимизация);
- ассоциативная память (поиск аналогов, прототипов);
- нечеткая логика;
- деревья решений;
- системы обработки экспертных знаний.

Методы Data Mining также можно классифицировать по задачам Data Mining.

В соответствии с такой классификацией выделяем две группы. Первая из них - это подразделение методов Data Mining на решающие задачи сегментации (т.е. задачи классификации и кластеризации) и задачи прогнозирования.

В соответствии со второй классификацией по задачам методы Data Mining могут быть направлены на получение описательных и прогнозирующих результатов.

Описательные методы служат для нахождения шаблонов или образцов, описывающих данные, которые поддаются интерпретации с точки зрения аналитика.

К методам, направленным на получение описательных результатов, относятся итеративные методы кластерного анализа, в том числе: алгоритм k-средних, k-медианы, иерархические методы кластерного анализа, самоорганизующиеся карты Кохонена, методы кросс-табличной визуализации, различные методы визуализации и другие.

Прогнозирующие методы используют значения одних переменных для предсказания/прогнозирования неизвестных (пропущенных) или будущих значений других (целевых) переменных.

К методам, направленным на получение прогнозирующих результатов, относятся такие методы: нейронные сети, деревья решений, линейная регрессия, метод ближайшего соседа, метод опорных векторов и др.

Свойства методов Data Mining

Различные методы Data Mining характеризуются определенными свойствами, которые могут быть определяющими при выборе метода анализа данных. Методы можно сравнивать между собой, оценивая характеристики их свойств.

Среди основных свойств и характеристик методов Data Mining рассмотрим следующие: точность, масштабируемость, интерпретируемость, проверяемость, трудоемкость, гибкость, быстрота и популярность.

Масштабируемость - свойство вычислительной системы, которое обеспечивает предсказуемый рост системных характеристик, например, быстроты реакции, общей производительности и пр., при добавлении к ней вычислительных ресурсов.

В таблице 3.1 приведена сравнительная характеристика некоторых распространенных методов [15]. Оценка каждой из характеристик проведена следующими категориями, в порядке возрастания: чрезвычайно низкая, очень низкая, низкая/нейтральная, нейтральная/низкая, нейтральная, нейтральная/высокая, высокая, очень высокая.

Как видно из рассмотренной таблицы, каждый из методов имеет свои сильные и слабые стороны. Но ни один метод, какой бы не была его оценка с точки зрения присущих ему характеристик, не может обеспечить решение всего спектра задач Data Mining.

Большинство инструментов Data Mining, предлагаемых сейчас на рынке программного обеспечения, реализуют сразу несколько методов, например, деревья решений, индукцию правил и визуализацию, или же нейронные сети, самоорганизующиеся карты Кохонена и визуализацию.

В универсальных прикладных статистических пакетах (например, SPSS, SAS, STATGRAPHICS, Statistica, др.) реализуется широкий спектр разнообразнейших методов (как статистических, так и кибернетических). Следует учитывать, что для возможности их использования, а также для интерпретации результатов работы статистических методов (корреляционного, регрессионного, факторного, дисперсионного анализа и др.) требуются специальные знания в области статистики.

Универсальность того или иного инструмента часто накладывает определенные ограничения на его возможности. Преимуществом использования таких универсальных пакетов является возможность относительно легко сравнивать результаты построенных

моделей, полученные различными методами. Такая возможность реализована, например, в пакете Statistica, где сравнение основано на так называемой "конкурентной оценке моделей". Эта оценка состоит в применении различных моделей к одному и тому же набору данных и последующем сравнении их характеристик для выбора наилучшей из них.

Таблица 3.1. Сравнительная характеристика методов Data Mining

Алгоритм	Точность	Масштаби-руемость	Интерпрети-руемость	Пригод-ность к исполь-зованию	Трудо-емкость	Разносто-ронность	Быстрота	Популяр-ность, широта исполь-зования
класси-ческие методы (линейная регрессия)	нейтраль-ная	высокая	высокая / нейтраль-ная	высокая	нейтраль-ная	нейтраль-ная	высокая	низкая
нейронны е сети	высокая	низкая	низкая	низкая	нейтраль-ная	низкая	очень низкая	низкая
методы визуали-зации	высокая	очень низкая	высокая	высокая	очень высокая	низкая	чрезвы-чайно низкая	высокая / нейтраль-ная
деревья решений	низкая	высокая	высокая	высокая / нейтраль-ная	высокая	высокая	высокая / нейтраль-ная	высокая / нейтраль-ная
полино-миальные нейронны е сети	высокая	нейтральна я	низкая	высокая / нейтраль-ная	нейтраль-ная / низкая	нейтраль-ная	низкая / нейтраль-ная	нейтраль-ная

k-ближай-шего соседа	низкая	очень низкая	высокая / нейтраль-ная	нейтраль-ная	нейтраль-ная / низкая	низкая	высокая	низкая

Задачи Data Mining. Информация и знания

Напомним, что в основу технологии Data Mining положена концепция шаблонов, представляющих собой закономерности. В результате обнаружения этих, скрытых от невооруженного глаза закономерностей решаются задачи Data Mining. Различным типам закономерностей, которые могут быть выражены в форме, понятной человеку, соответствуют определенные задачи Data Mining.

Задачи (tasks) Data Mining иногда называют закономерностями (regularity) [16] или техниками (techniques) [17].

Единого мнения относительно того, какие задачи следует относить к Data Mining, нет. Большинство авторитетных источников перечисляют следующие: классификация, кластеризация, прогнозирование, ассоциация, визуализация, анализ и обнаружение отклонений, оценивание, анализ связей, подведение итогов.

Цель описания, которое следует ниже, - дать общее представление о задачах Data Mining, сравнить некоторые из них, а также представить некоторые методы, с помощью которых эти задачи решаются. Наиболее распространенные задачи Data Mining - классификация, кластеризация, ассоциация, прогнозирование и визуализация - будут подробно рассмотрены в последующих лекциях. Таким образом, задачи подразделяются по типам производимой информации [18], это наиболее общая классификация задач Data Mining. Дальнейшее детальное знакомство с методами решения задач Data Mining будет представлено в следующем разделе курса.

Задачи Data Mining

Классификация (Classification)

Краткое описание. Наиболее простая и распространенная задача Data Mining. В результате решения задачи классификации обнаруживаются признаки, которые характеризуют группы объектов исследуемого набора данных - классы; по этим признакам новый объект можно отнести к тому или иному классу.

Методы решения. Для решения задачи классификации могут использоваться методы: ближайшего соседа (Nearest Neighbor); k-ближайшего соседа (k-Nearest Neighbor); байесовские сети (Bayesian Networks); индукция деревьев решений; нейронные сети (neural networks).

Кластеризация (Clustering)

Краткое описание. Кластеризация является логическим продолжением идеи классификации. Это задача более сложная, особенность кластеризации заключается в том, что классы объектов изначально не предопределены. Результатом кластеризации является разбиение объектов на группы.

Пример метода решения задачи кластеризации: обучение "без учителя" особого вида нейронных сетей - самоорганизующихся карт Кохонена.

Ассоциация (Associations)

Краткое описание. В ходе решения задачи поиска ассоциативных правил отыскиваются закономерности между связанными событиями в наборе данных.

Отличие ассоциации от двух предыдущих задач Data Mining: поиск закономерностей осуществляется не на основе свойств анализируемого объекта, а между несколькими событиями, которые происходят одновременно.

Наиболее известный алгоритм решения задачи поиска ассоциативных правил - алгоритм Apriori.

Последовательность (Sequence), или последовательная ассоциация (sequential association)

Краткое описание. Последовательность позволяет найти временные закономерности между транзакциями. Задача последовательности подобна ассоциации, но ее целью является установление закономерностей не между одновременно наступающими событиями, а между событиями, связанными во времени (т.е. происходящими с некоторым определенным интервалом во времени). Другими словами, последовательность определяется высокой вероятностью цепочки связанных во времени событий. Фактически, ассоциация является частным случаем последовательности с временным лагом, равным нулю. Эту задачу Data Mining также называют задачей нахождения последовательных шаблонов (sequential pattern).

Правило последовательности: после события X через определенное время произойдет событие Y.

Пример. После покупки квартиры жильцы в 60% случаев в течение двух недель приобретают холодильник, а в течение двух месяцев в 50% случаев приобретается телевизор. Решение данной задачи широко применяется в маркетинге и менеджменте, например, при управлении циклом работы с клиентом (Customer Lifecycle Management).

Прогнозирование (Forecasting)

Краткое описание. В результате решения задачи прогнозирования на основе особенностей исторических данных оцениваются пропущенные или же будущие значения целевых численных показателей.

Для решения таких задач широко применяются методы математической статистики, нейронные сети и др.

Определение отклонений или выбросов (Deviation Detection), анализ отклонений или выбросов

Краткое описание. Цель решения данной задачи - обнаружение и анализ данных, наиболее отличающихся от общего множества данных, выявление так называемых нехарактерных шаблонов.

Оценивание (Estimation)

Задача оценивания сводится к предсказанию непрерывных значений признака.

Анализ связей (Link Analysis) - задача нахождения зависимостей в наборе данных.

Визуализация (Visualization, Graph Mining)

В результате визуализации создается графический образ анализируемых данных. Для решения задачи визуализации используются графические методы, показывающие наличие закономерностей в данных.

Пример методов визуализации - представление данных в 2-D и 3-D измерениях.

Подведение итогов (Summarization) - задача, цель которой - описание конкретных групп объектов из анализируемого набора данных.

Классификация задач Data Mining

Согласно классификации по стратегиям, задачи Data Mining подразделяются на следующие группы:

- обучение с учителем;
- обучение без учителя;
- другие.

Категория обучение с учителем представлена следующими задачами Data Mining: классификация, оценка, прогнозирование.

Категория обучение без учителя представлена задачей кластеризации.

В категорию другие входят задачи, не включенные в предыдущие две стратегии.

Задачи Data Mining, в зависимости от используемых моделей, могут быть дескриптивными и прогнозирующими. Эти типы моделей будут подробно описаны в лекции, посвященной процессу Data Mining.

В соответствии с этой классификацией, задачи Data Mining представлены группами описательных и прогнозирующих задач.

В результате решения описательных (descriptive) задач аналитик получает шаблоны, описывающие данные, которые поддаются интерпретации.

Эти задачи описывают общую концепцию анализируемых данных, определяют информативные, итоговые, отличительные особенности данных. Концепция описательных задач подразумевает характеристику и сравнение наборов данных.

Характеристика набора данных обеспечивает краткое и сжатое описание некоторого набора данных.

Сравнение обеспечивает сравнительное описание двух или более наборов данных.

Прогнозирующие (predictive) основываются на анализе данных, создании модели, предсказании тенденций или свойств новых или неизвестных данных.

Достаточно близким к вышеупомянутой классификации является подразделение задач Data Mining на следующие: исследования и открытия, прогнозирования и классификации, объяснения и описания.

Автоматическое исследование и открытие (свободный поиск)

Пример задачи: обнаружение новых сегментов рынка.

Для решения данного класса задач используются методы кластерного анализа.

прогнозирование и классификация

Пример задачи: предсказание роста объемов продаж на основе текущих значений.

Методы: регрессия, нейронные сети, генетические алгоритмы, деревья решений.

Задачи классификации и прогнозирования составляют группу так называемого индуктивного моделирования, в результате которого обеспечивается изучение анализируемого объекта или системы. В процессе решения этих задач на основе набора данных разрабатывается общая модель или гипотеза.

Объяснение и описание

Пример задачи: характеристика клиентов по демографическим данным и историям покупок.

Методы: деревья решения, системы правил, правила ассоциации, анализ связей.

Если доход клиента больше, чем 50 условных единиц, и его возраст - более 30 лет, тогда класс клиента - первый.

В интерпретации обобщенной модели аналитик получает новое знание. Группировка объектов происходит на основе их сходства.

Связь понятий

Итак, в предыдущей лекции нами были рассмотрены методы Data Mining и действия, выполняемые в рамках стадий Data Mining. Только что мы рассмотрели основные задачи Data Mining.

Напомним, что главная ценность Data Mining - это практическая направленность данной технологии, путь от сырых данных к конкретному знанию, от постановки задачи к готовому приложению, при поддержке которого можно принимать решения.

Многочисленность понятий, которые объединились в Data Mining, а также разнообразие методов, поддерживающих данную технологию, начинающему аналитику могут напомнить мозаику, части которой мало связаны между собой.

Как же мы можем связать в одно целое задачи, методы, действия, закономерности, приложения, данные, информацию, решения?

Рассмотрим два потока:

1. **ДАННЫЕ - ИНФОРМАЦИЯ - ЗНАНИЯ И РЕШЕНИЯ**
2. **ЗАДАЧИ - ДЕЙСТВИЯ И МЕТОДЫ РЕШЕНИЯ - ПРИЛОЖЕНИЯ**

Эти потоки являются "двумя сторонами одной медали", отображением одного процесса, результатом которого должно быть знание и принятие решения.

От данных к решениям

Для начала рассмотрим первый поток. На рис. 4.1.показана связь понятий "данные", "информация" и "решения", которая возникает в процессе принятия решений.

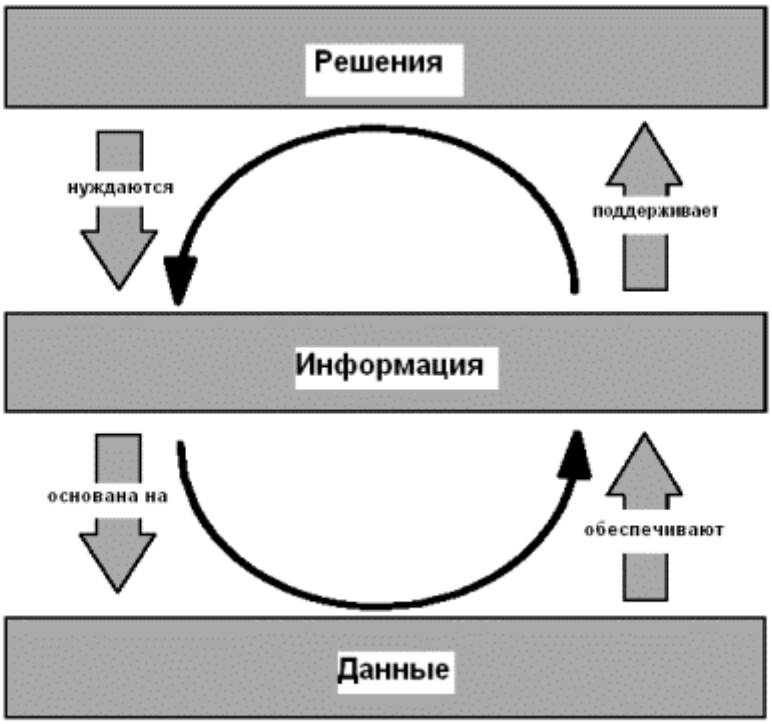

Рис. 4.1. Решения, информация и данные

Как видно из рисунка, данный процесс является циклическим. Принятие решений требует информации, которая основана на данных. Данные обеспечивают информацию, которая поддерживает решения, и т.д.

Рассмотренные понятия являются составной частью так называемой информационной пирамиды, в основании которой находятся данные, следующий уровень - это информация, затем идет решение, завершает пирамиду уровень знания. По мере продвижения вверх по информационной пирамиде объемы данных переходят в ценность решений, т.е. ценность для бизнеса. А, как известно, целью Business Intelligence является преобразование объемов данных в ценность бизнеса.

От задачи к приложению

Теперь подойдем к этому же процессу с другой стороны. Рассмотрим рис. 4.2. По словам авторов [17], он не претендует на полноту, зато отображает все уровни, которые затрагивает Data Mining.

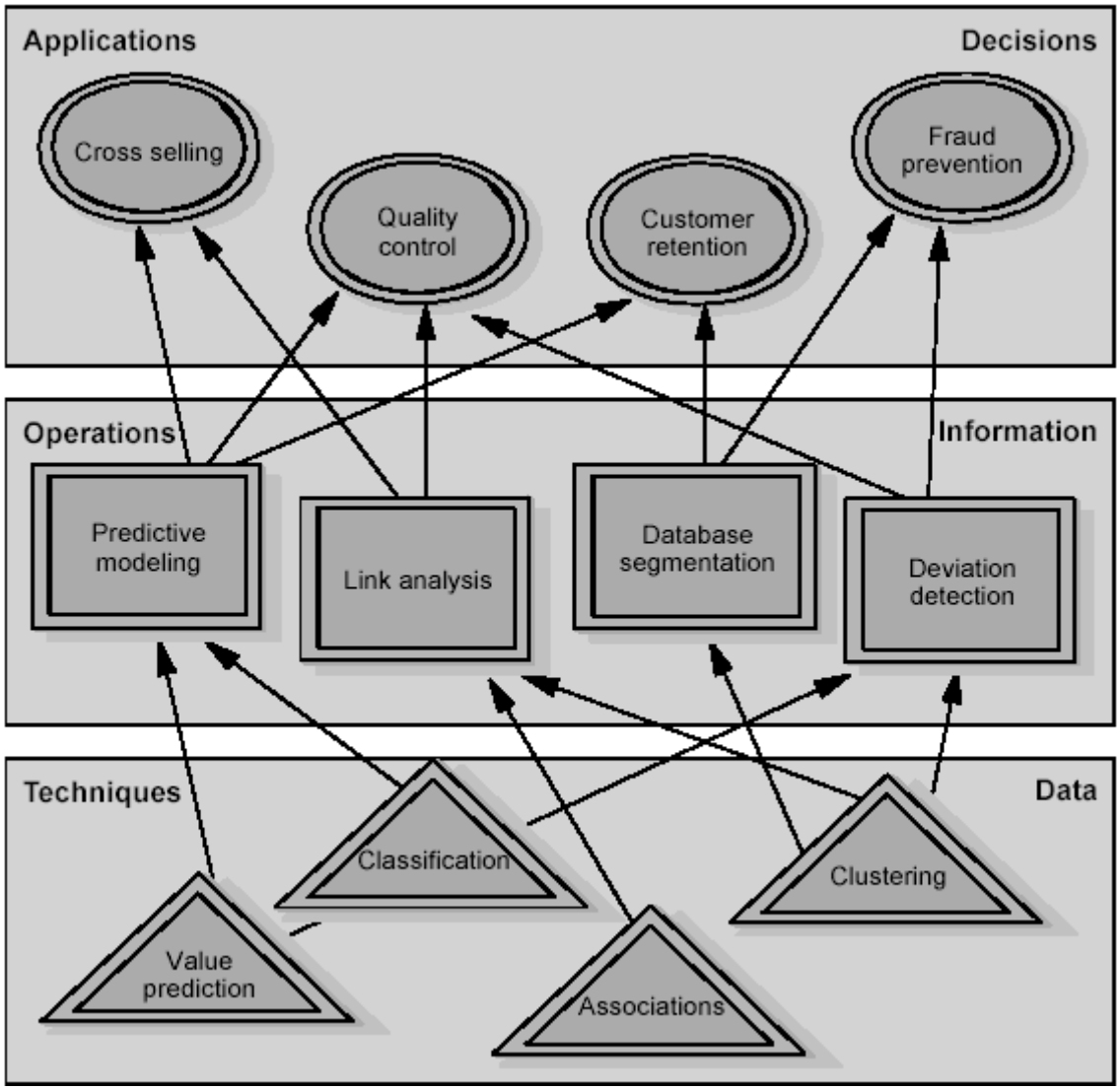

Рис. 4.2. Задачи, действия, приложения

Следует отметить, что уровни анализа (данные, информация, знания) практически соответствуют этапам эволюции анализа данных, которая происходила на протяжении последних лет.

Верхний - уровень приложений - является уровнем бизнеса (если мы имеем дело с задачей бизнеса), на нем менеджеры принимают решения. Приведенные примеры приложений: перекрестные продажи, контроль качества, удерживание клиентов.

Средний - уровень действий - по своей сути является уровнем информации, именно на нем выполняются действия Data Mining; на рисунке приведены такие действия:

прогностическое моделирование (было рассмотрено в предыдущей лекции), анализ связей, сегментация данных и другие.

Нижний - уровень определения задачи Data Mining, которую необходимо решить применительно к данным, имеющимся в наличии; на рисунке приведены задачи предсказания числовых значений, классификация, кластеризация, ассоциация.

Рассмотрим таблицу, демонстрирующую связь этих понятий.

Таблица 4.1. Уровни Data Mining

уровень 3	приложения	удержание клиентов	знания	Data Mining результат
уровень 2	действия	прогностическое моделирование	информация	метод анализа
уровень 1	задачи	классификация	данные	запросы

Напомним, что для решения задачи классификации результаты работы первой стадии (индукции правил) используются для отнесения нового объекта, с определенной уверенностью, к одному из известных, предопределенных классов на основании известных значений.

Рассмотрим задачу удержания клиентов (определения надежности клиентов фирмы).

Первый уровень. Данные - база данных по клиентам. Есть данные о клиенте (возраст, пол, профессия, доход). Определенная часть клиентов, воспользовавшись продуктом фирмы, осталась ей верна; другие клиенты больше не приобретали продукты фирмы. На этом уровне мы определяем тип задачи - это задача классификации.

На **втором уровне** определяем действие - прогностическое моделирование. С помощью прогностического моделирования мы с определенной долей уверенности можем отнести новый объект, в данном случае, нового клиента, к одному из известных классов - постоянный клиент, или это, скорее всего, его разовая покупка.

На **третьем уровне** мы можем воспользоваться приложением для принятия решения. В результате приобретения знаний, фирма может существенно снизить расходы, например, на рекламу, зная заранее, каким из клиентов следует активно рассылать рекламные материалы.

Таким образом, на протяжении нескольких лекций мы определились с понятиями "данные", "задачи", "методы", "действия".

Информация

Сейчас остановимся на еще не рассмотренном понятии информации. Несмотря на распространенность данного понятия, мы не всегда можем точно его определить и отличить от понятия данных. Информация, по своей сути, имеет многогранную природу. С развитием человечества, в том числе, с развитием компьютерных технологий, информация обретает все новые и новые свойства.

Обратимся к словарю. Информация (лат. informatio) -

1. любые сообщение о чем-либо;
2. сведения, являющиеся объектом хранения, переработки и передачи (например генетическая информация);
3. в математике (кибернетике) - количественная мера устранения неопределенности (энтропия), мера организации системы; в теории информации - раздел кибернетики, изучающий количественные закономерности, которые связаны со сбором, передачей, преобразованием и вычислением информации.

Информация - любые, неизвестные ранее сведения о каком-либо событии, сущности, процессе и т.п., являющиеся объектом некоторых операций, для которых существует содержательная интерпретация.

Под операциями здесь подразумевается восприятие, передача, преобразование, хранение и использование. Для восприятия информации необходима некоторая воспринимающая система, которая может интерпретировать ее, преобразовывать, определять соответствие определенным правилам и т.п. Таким образом, понятие информации следует рассматривать только при наличии источника и получателя информации, а также канала связи между ними.

Свойства информации

- Полнота информации.

  Это свойство характеризует качество информации и определяет достаточность данных для принятия решений, т.е. информация должна содержать весь необходимый набор данных.

  Пример. "Продажи товара А начнут сокращаться" Эта информация неполная, поскольку неизвестно, когда именно они начнут сокращаться.

  Пример полной информации. "Начиная с первого квартала, продажи товара А начнут сокращаться." Этой информации достаточно для принятия решений.

- Достоверность информации.

  Информация может быть достоверной и недостоверной. В недостоверной информации присутствует информационный шум, и чем он выше, тем ниже достоверность информации.

- Ценность информации.

  Ценность информации не может быть абстрактной. Информация должна быть полезной и ценной для определенной категории пользователей.

- Адекватность информации.

  Это свойство характеризует степень соответствия информации реальному объективному состоянию. Адекватная информация - это полная и достоверная информация.

- Актуальность информации.

  Информация должна быть актуальной, т.е. не устаревшей. Это свойство информации характеризует степень соответствия информации настоящему моменту времени.

- Ясность информации.

  Информация должна быть понятна тому кругу лиц, для которого она предназначена.

- Доступность информации.

  Доступность характеризует меру возможности получить определенную информацию. На это свойство информации влияют одновременно доступность данных и доступность адекватных методов.

- Субъективность информации.

  Информация носит субъективный характер, она определяется степенью восприятия субъекта (получателя информации).

Требования, предъявляемые к информации

- Динамический характер информации.

  Информация существует только в момент взаимодействия данных и методов, т.е. в момент информационного процесса. Остальное время она пребывает в состоянии данных.

- Адекватность используемых методов.

  Информация извлекается из данных. Однако в результате использования одних и тех же данных может появляться разная информация. Это зависит от адекватности выбранных методов обработки исходных данных.

Данные, по своей сути, являются объективными. Методы являются субъективными, в основе методов лежат алгоритмы, субъективно составленные и подготовленные. Таким образом, информация возникает и существует в момент диалектического взаимодействия объективных данных и субъективных методов.

Для бизнеса информация является исходной составляющей принятия решений.

Всю информацию, возникающую в процессе функционирования бизнеса и управления им, можно классифицировать определенным образом. В зависимости от источника получения, информацию разделяют на внутреннюю и внешнюю (например, информация, описывающая явления, происходящие за пределами фирмы, но имеющие к ней непосредственное отношение).

Также информация может быть классифицирована на фактическую и прогнозную. К фактической информации о бизнесе относится информация, характеризующая

свершившиеся факты; она является точной. Прогнозная информация является рассчитываемой или предполагаемой, поэтому ее нельзя считать точной, она может иметь определенную погрешность.

Знания

Знания - совокупность фактов, закономерностей и эвристических правил, с помощью которых решается поставленная задача.

Итак, формирование информации происходит в процессе сбора и передачи, т.е. обработки данных. Каким же образом из информации получают знания?

Все чаще истинные знания образуются на основе распределенных взаимосвязей разнородной информации [19]. Когда информация собрана и передана для получения явно не определенного заранее результата, то вы получаете знания. Сама по себе информация в чистом виде бессмысленна. Отсюда следует вывод, что информация - это чье-то тактическое знание, передаваемое в виде символов и при помощи каких-либо прикладных средств.

По определению Денхема Грэя, "знания - это абсолютное использование информации и данных, совместно с потенциалом практического опыта людей, способностями, идеями, интуицией, убежденностью и мотивациями".

Знания имеют определенные свойства, которые отличают их от информации [20].

1. **Структурированность**. Знания должны быть "разложены по полочкам".
2. **Удобство доступа и усвоения**. Для человека - это способность быстро понять и запомнить или, наоборот, вспомнить; для компьютерных знаний - средства доступа к знаниям.
3. **Лаконичность**. Лаконичность позволяет быстро осваивать и перерабатывать знания и повышает "коэффициент полезного содержания". В данный список лаконичность была добавлена из-за всем известной проблемы шума и мусорных документов, характерной именно для компьютерной информации - Internet и электронного документооборота.
4. **Непротиворечивость**. Знания не должны противоречить друг другу.
5. **Процедуры обработки**. Знания нужны для того, чтобы их использовать. Одно из главных свойств знаний - возможность их передачи другим и способность делать выводы на их основе. Для этого должны существовать процедуры обработки знаний. Способность делать выводы означает для машины наличие процедур обработки и вывода и подготовленность структур данных для такой обработки, т.е. наличие специальных форматов знаний.

Сопоставление и сравнение понятий "информация", "данные", "знание"

Для того чтобы уверенно оперировать понятиями "информация", "данные", "знание", необходимо не только понимать суть этих понятий, но и прочувствовать отличия между ними. Однако, одной интуитивной интерпретации этих понятий здесь недостаточно. Сложность понимания отличий вышеупомянутых понятий - в их кажущейся синонимичности. Вспомним, что понятие Data Mining переводится на русский язык при помощи этих же трех понятий: как добыча данных, извлечение информации, раскопка знаний.

Для того чтобы прочувствовать разницу, рассмотрим применение этих трех понятий на простом примере.

Для начала сделаем попытку разобраться в этих терминах на простых примерах.

1. Студент, который сдает экзамен, нуждается в данных.
2. Студент, который сдает экзамен, нуждается в информации.
3. Студент, который сдает экзамен, нуждается в знаниях.

При рассмотрении первого варианта - студент нуждается в данных - возникает мысль, что студенту нужны данные, например, для вычислений. Информацией во втором варианте может выступать конспект или учебник. В результате их использования студент получает лишь информацию, которая в определенных случаях может перейти в знания. Третий вариант звучит наиболее логично.

Информация, в отличие от данных, имеет смысл.

Понятия "информация" и "знания", с философской точки зрения, являются понятиями более высокого уровня, чем "данные", которое возникло относительно недавно.

Понятие "информации" непосредственно связано с сущностью процессов внутри информационной системы, тогда так понятие "знание" скорее ориентировано на качество процессов. Понятие "знание" тесно связано с процессом принятия решений.

Несмотря на различия, рассмотренные понятия, как уже отмечалось ранее, не являются разрозненными и несвязанными. Они есть часть одного потока: у истока его находятся данные, в процессе передачи которых возникает информация, и в результате использования информации, при определенных условиях, возникают знания.

В лекции уже отмечалось, что в процессе движения вверх по информационной пирамиде объемы данных переходят в ценность знаний. Однако большие объемы данных вовсе не означают и, тем более, не гарантируют получение знаний. Существует определенная зависимость ценности полученных знаний от качества и мощности процедур обработки данных. Типичным примером информации, которую нельзя превратить в знание, является текст на иностранном языке. При отсутствии словаря и переводчика эта информация вообще не имеет ценности, она не может перейти в знание. При наличии словаря процесс перехода от информации к знанию возможен, но длителен и трудоемок. При наличии переводчика информация действительно переходит в знания.

Таким образом, для получения ценных знаний необходимы качественные процедуры обработки. Процесс перехода от данных к знаниям занимает много времени и стоит дорого. Поэтому очевидно, что технология Data Mining с ее мощными и разнообразными алгоритмами является инструментом, при помощи которого, продвигаясь вверх по информационной пирамиде, мы можем получать действительно качественные и ценные знания.

Задачи Data Mining. Классификация и кластеризация

В предыдущей лекции мы кратко остановились на основных задачах Data Mining. Две из них - классификацию и кластеризацию - мы рассмотрим подробно в этой лекции.

Задача классификации

Классификация является наиболее простой и одновременно наиболее часто решаемой задачей Data Mining. Ввиду распространенности задач классификации необходимо четкое понимания сути этого понятия.

Приведем несколько определений.

Классификация - системное распределение изучаемых предметов, явлений, процессов по родам, видам, типам, по каким-либо существенным признакам для удобства их исследования; группировка исходных понятий и расположение их в определенном порядке, отражающем степень этого сходства.

Классификация - упорядоченное по некоторому принципу множество объектов, которые имеют сходные классификационные признаки (одно или несколько свойств), выбранных для определения сходства или различия между этими объектами.

Классификация требует соблюдения следующих правил:

- в каждом акте деления необходимо применять только одно основание;
- деление должно быть соразмерным, т.е. общий объем видовых понятий должен равняться объему делимого родового понятия;
- члены деления должны взаимно исключать друг друга, их объемы не должны перекрещиваться;
- деление должно быть последовательным.

Различают:

- вспомогательную (искусственную) классификацию, которая производится по внешнему признаку и служит для придания множеству предметов (процессов, явлений) нужного порядка;
- естественную классификацию, которая производится по существенным признакам, характеризующим внутреннюю общность предметов и явлений. Она является результатом и важным средством научного исследования, т.к. предполагает и закрепляет результаты изучения закономерностей классифицируемых объектов.

В зависимости от выбранных признаков, их сочетания и процедуры деления понятий классификация может быть:

- простой - деление родового понятия только по признаку и только один раз до раскрытия всех видов. Примером такой классификации является дихотомия, при которой членами деления бывают только два понятия, каждое из которых является противоречащим другому (т.е. соблюдается принцип: "А и не А");

- сложной - применяется для деления одного понятия по разным основаниям и синтеза таких простых делений в единое целое. Примером такой классификации является периодическая система химических элементов.

Под классификацией будем понимать отнесение объектов (наблюдений, событий) к одному из заранее известных классов.

Классификация - это закономерность, позволяющая делать вывод относительно определения характеристик конкретной группы. Таким образом, для проведения классификации должны присутствовать признаки, характеризующие группу, к которой принадлежит то или иное событие или объект (обычно при этом на основании анализа уже классифицированных событий формулируются некие правила).

Классификация относится к стратегии обучения с учителем (supervised learning), которое также именуют контролируемым или управляемым обучением.

Задачей классификации часто называют предсказание категориальной зависимой переменной (т.е. зависимой переменной, являющейся категорией) на основе выборки непрерывных и/или категориальных переменных.

Например, можно предсказать, кто из клиентов фирмы является потенциальным покупателем определенного товара, а кто - нет, кто воспользуется услугой фирмы, а кто - нет, и т.д. Этот тип задач относится к задачам бинарной классификации, в них зависимая переменная может принимать только два значения (например, да или нет, 0 или 1).

Другой вариант классификации возникает, если зависимая переменная может принимать значения из некоторого множества предопределенных классов. Например, когда необходимо предсказать, какую марку автомобиля захочет купить клиент. В этих случаях рассматривается множество классов для зависимой переменной.

Классификация может быть одномерной (по одному признаку) и многомерной (по двум и более признакам).

Многомерная классификация была разработана биологами при решении проблем дискриминации для классифицирования организмов. Одной из первых работ, посвященных этому направлению, считают работу Р. Фишера (1930 г.), в которой организмы разделялись на подвиды в зависимости от результатов измерений их физических параметров. Биология была и остается наиболее востребованной и удобной средой для разработки многомерных методов классификации.

Рассмотрим задачу классификации на простом примере. Допустим, имеется база данных о клиентах туристического агентства с информацией о возрасте и доходе за месяц. Есть рекламный материал двух видов: более дорогой и комфортный отдых и более дешевый, молодежный отдых. Соответственно, определены два класса клиентов: класс 1 и класс 2. База данных приведена в таблице 5.1.

Таблица 5.1. База данных клиентов туристического агентства

Код клиента	Возраст	Доход	Класс
1	18	25	1

2	22	100	1
3	30	70	1
4	32	120	1
5	24	15	2
6	25	22	1
7	32	50	2
8	19	45	2
9	22	75	1
10	40	90	2

Задача. Определить, к какому классу принадлежит новый клиент и какой из двух видов рекламных материалов ему стоит отсылать.

Для наглядности представим нашу базу данных в двухмерном измерении (возраст и доход), в виде множества объектов, принадлежащих классам 1 (оранжевая метка) и 2 (серая метка). На рис. 5.1 приведены объекты из двух классов.

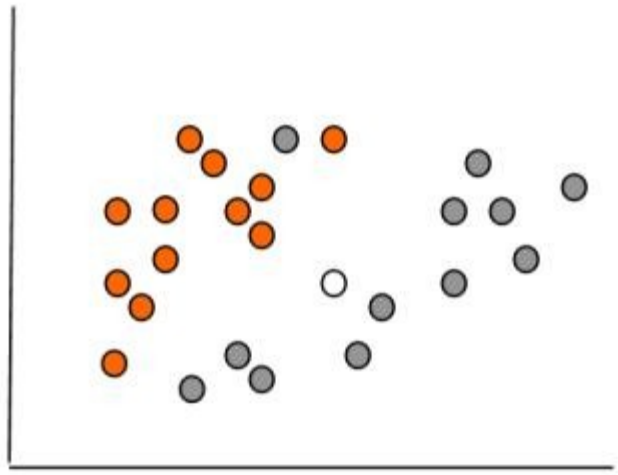

Рис. 5.1. Множество объектов базы данных в двухмерном измерениии

Решение нашей задачи будет состоять в том, чтобы определить, к какому классу относится новый клиент, на рисунке обозначенный белой меткой.

Процесс классификации

Цель процесса классификации состоит в том, чтобы построить модель, которая использует прогнозирующие атрибуты в качестве входных параметров и получает значение зависимого атрибута. Процесс классификации заключается в разбиении множества объектов на классы по определенному критерию.

Классификатором называется некая сущность, определяющая, какому из предопределенных классов принадлежит объект по вектору признаков.

Для проведения классификации с помощью математических методов необходимо иметь формальное описание объекта, которым можно оперировать, используя математический аппарат классификации. Таким описанием в нашем случае выступает база данных. Каждый объект (запись базы данных) несет информацию о некотором свойстве объекта.

Набор исходных данных (или выборку данных) разбивают на два множества: обучающее и тестовое.

Обучающее множество (training set) - множество, которое включает данные, использующиеся для обучения (конструирования) модели.

Такое множество содержит входные и выходные (целевые) значения примеров. Выходные значения предназначены для обучения модели.

Тестовое (test set) множество также содержит входные и выходные значения примеров. Здесь выходные значения используются для проверки работоспособности модели.

Процесс классификации состоит из двух этапов [21]: конструирования модели и ее использования.

1. Конструирование модели: описание множества предопределенных классов.
    - Каждый пример набора данных относится к одному предопределенному классу.
    - На этом этапе используется обучающее множество, на нем происходит конструирование модели.
    - Полученная модель представлена классификационными правилами, деревом решений или математической формулой.
2. Использование модели: классификация новых или неизвестных значений.
    - Оценка правильности (точности) модели.
        1. Известные значения из тестового примера сравниваются с результатами использования полученной модели.
        2. Уровень точности - процент правильно классифицированных примеров в тестовом множестве.
        3. Тестовое множество, т.е. множество, на котором тестируется построенная модель, не должно зависеть от обучающего множества.
    - Если точность модели допустима, возможно использование модели для классификации новых примеров, класс которых неизвестен.

Процесс классификации, а именно, конструирование модели и ее использование, представлен на рис. 5.2. - 5.3.

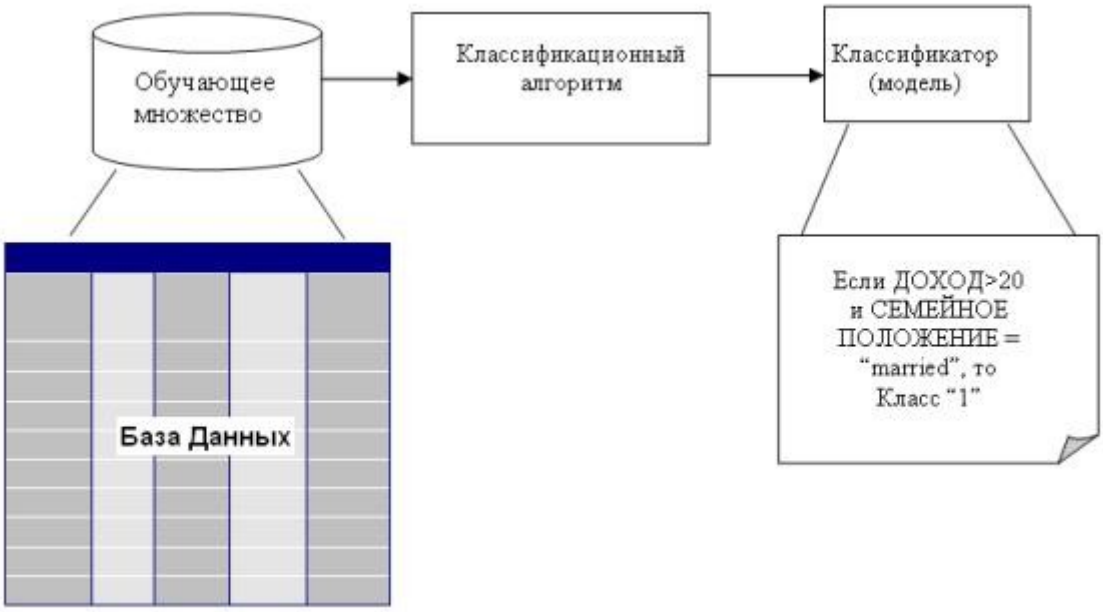

Рис. 5.2. Процесс классификации. Конструирование модели

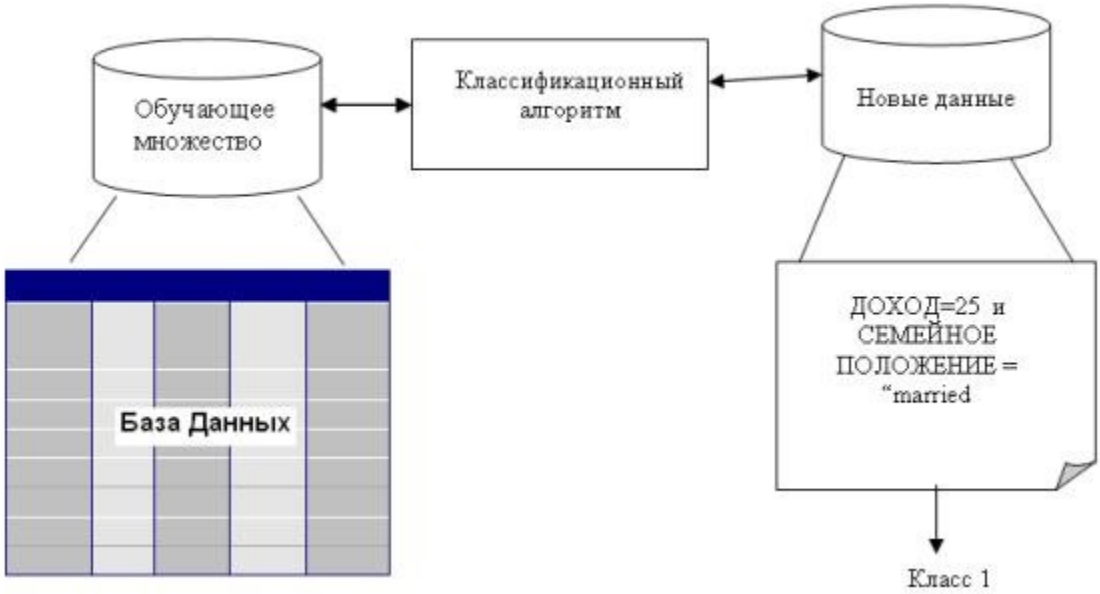

Рис. 5.3. Процесс классификации. Использование модели

Методы, применяемые для решения задач классификации

Для классификации используются различные методы. Основные из них:

- классификация с помощью деревьев решений;
- байесовская (наивная) классификация;
- классификация при помощи искусственных нейронных сетей;
- классификация методом опорных векторов;
- статистические методы, в частности, линейная регрессия;

- классификация при помощи метода ближайшего соседа;
- классификация CBR-методом;
- классификация при помощи генетических алгоритмов.

Схематическое решение задачи классификации некоторыми методами (при помощи линейной регрессии, деревьев решений и нейронных сетей) приведены на рис. 5.4 - 5.6.

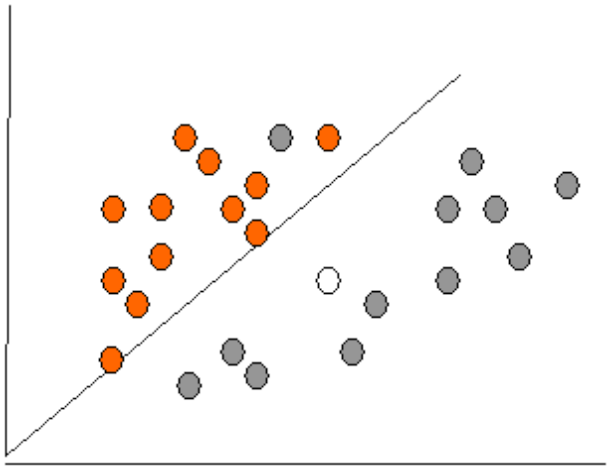

Рис. 5.4. Решение задачи классификации методом линейной регрессии

```
if X > 5 then grey
   else if Y > 3 then orange
      else if X > 2 then grey
         else orange
```

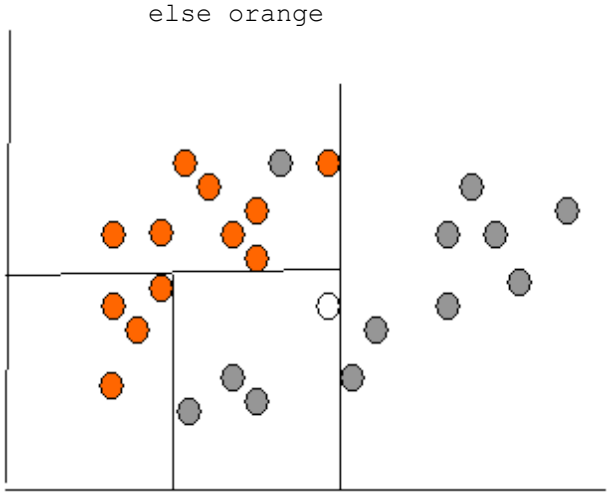

Рис. 5.5. Решение задачи классификации методом деревьев решений

Рис. 5.6. Решение задачи классификации методом нейронных сетей

Точность классификации: оценка уровня ошибок

Оценка точности классификации может проводиться при помощи кросс-проверки. Кросс-проверка (Cross-validation) - это процедура оценки точности классификации на данных из тестового множества, которое также называют кросс-проверочным множеством. Точность классификации тестового множества сравнивается с точностью классификации обучающего множества. Если классификация тестового множества дает приблизительно такие же результаты по точности, как и классификация обучающего множества, считается, что данная модель прошла кросс-проверку.

Разделение на обучающее и тестовое множества осуществляется путем деления выборки в определенной пропорции, например обучающее множество - две трети данных и тестовое - одна треть данных. Этот способ следует использовать для выборок с большим количеством примеров. Если же выборка имеет малые объемы, рекомендуется применять специальные методы, при использовании которых обучающая и тестовая выборки могут частично пересекаться.

Оценивание классификационных методов

Оценивание методов следует проводить, исходя из следующих характеристик [21]: скорость, робастность, интерпретируемость, надежность.

Скорость характеризует время, которое требуется на создание модели и ее использование.

Робастность, т.е. устойчивость к каким-либо нарушениям исходных предпосылок, означает возможность работы с зашумленными данными и пропущенными значениями в данных.

Интерпретируемость обеспечивает возможность понимания модели аналитиком.

Свойства классификационных правил:

- размер дерева решений;

- компактность классификационных правил.

Надежность методов классификации предусматривает возможность работы этих методов при наличии в наборе данных шумов и выбросов.

Задача кластеризации

Только что мы изучили задачу классификации, относящуюся к стратегии "обучение с учителем".

В этой части лекции мы введем понятия кластеризации, кластера, кратко рассмотрим классы методов, с помощью которых решается задача кластеризации, некоторые моменты процесса кластеризации, а также разберем примеры применения кластерного анализа.

Задача кластеризации сходна с задачей классификации, является ее логическим продолжением, но ее отличие в том, что классы изучаемого набора данных заранее не предопределены.

Синонимами термина "кластеризация" являются "автоматическая классификация", "обучение без учителя" и "таксономия".

Кластеризация предназначена для разбиения совокупности объектов на однородные группы (кластеры или классы). Если данные выборки представить как точки в признаковом пространстве, то задача кластеризации сводится к определению "сгущений точек".

Цель кластеризации - поиск существующих структур.

Кластеризация является описательной процедурой, она не делает никаких статистических выводов, но дает возможность провести разведочный анализ и изучить "структуру данных".

Само понятие "кластер" определено неоднозначно: в каждом исследовании свои "кластеры". Переводится понятие кластер (cluster) как "скопление", "гроздь".

Кластер можно охарактеризовать как группу объектов, имеющих общие свойства.

Характеристиками кластера можно назвать два признака:

- внутренняя однородность;
- внешняя изолированность.

Вопрос, задаваемый аналитиками при решении многих задач, состоит в том, как организовать данные в наглядные структуры, т.е. развернуть таксономии.

Наибольшее применение кластеризация первоначально получила в таких науках как биология, антропология, психология. Для решения экономических задач кластеризация длительное время мало использовалась из-за специфики экономических данных и явлений.

В таблице 5.2 приведено сравнение некоторых параметров задач классификации и кластеризации.

Таблица 5.2. Сравнение классификации и кластерзации

Характеристика	Классификация	Кластеризация
Контролируемость обучения	Контролируемое обучение	Неконтролируемое обучение
Стратегия	Обучение с учителем	Обучение без учителя
Наличие метки класса	Обучающее множество сопровождается меткой, указывающей класс, к которому относится наблюдение	Метки класса обучающего множества неизвестны
Основание для классификации	Новые данные классифицируются на основании обучающего множества	Дано множество данных с целью установления существования классов или кластеров данных

На рис. 5.7 схематически представлены задачи классификации и кластеризации.

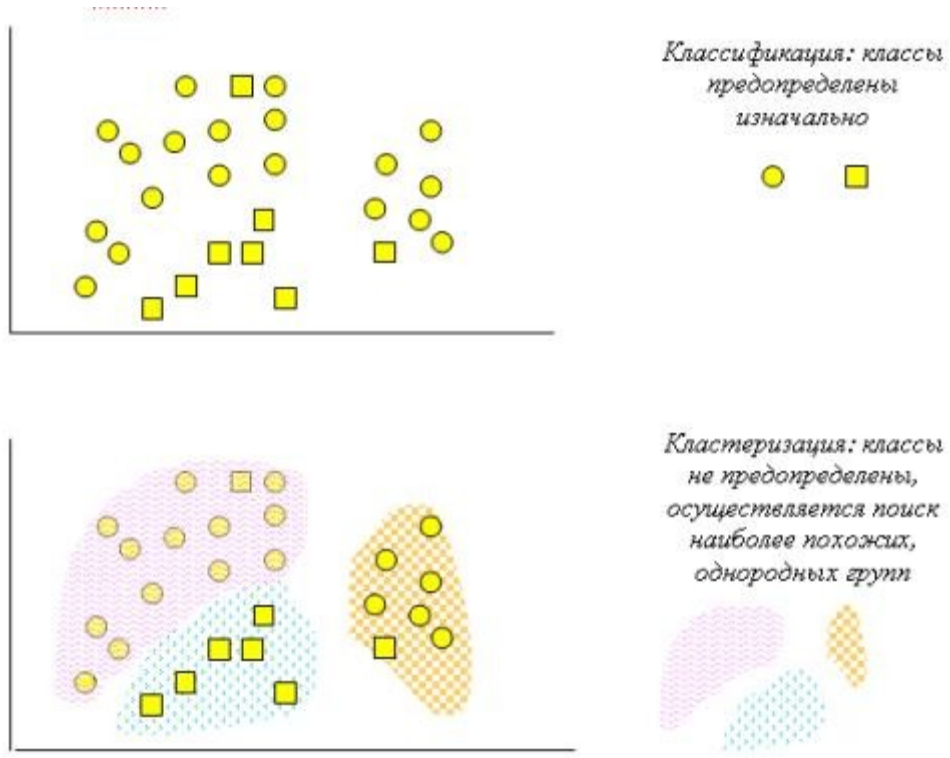

Рис. 5.7. Сравнение задач классификации и кластеризации

Кластеры могут быть непересекающимися, или эксклюзивными (non-overlapping, exclusive), и пересекающимися (overlapping) [22]. Схематическое изображение непересекающихся и пересекающихся кластеров дано на рис. 5.8.

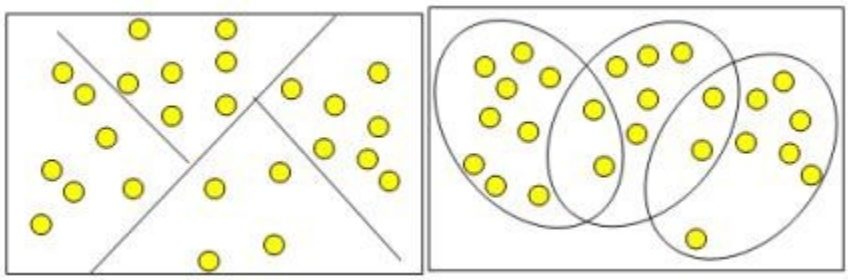

Рис. 5.8. Непересекающиеся и пересекающиеся кластеры

Следует отметить, что в результате применения различных методов кластерного анализа могут быть получены кластеры различной формы. Например, возможны кластеры "цепочного" типа, когда кластеры представлены длинными "цепочками", кластеры удлиненной формы и т.д., а некоторые методы могут создавать кластеры произвольной формы.

Различные методы могут стремиться создавать кластеры определенных размеров (например, малых или крупных) либо предполагать в наборе данных наличие кластеров различного размера.

Некоторые методы кластерного анализа особенно чувствительны к шумам или выбросам, другие - менее.

В результате применения различных методов кластеризации могут быть получены неодинаковые результаты, это нормально и является особенностью работы того или иного алгоритма.

Данные особенности следует учитывать при выборе метода кластеризации.

Подробнее обо всех свойствах кластерного анализа будет рассказано в лекции, посвященной его методам.

На сегодняшний день разработано более сотни различных алгоритмов кластеризации. Некоторые, наиболее часто используемые, будут подробно описаны во втором разделе курса лекций.

Приведем краткую характеристику подходов к кластеризации [21].

- Алгоритмы, основанные на разделении данных (Partitioning algorithms), в т.ч. итеративные:
  - разделение объектов на k кластеров;
  - итеративное перераспределение объектов для улучшения кластеризации.
- Иерархические алгоритмы (Hierarchy algorithms):
  - агломерация: каждый объект первоначально является кластером, кластеры, соединяясь друг с другом, формируют больший кластер и т.д.
- Методы, основанные на концентрации объектов (Density-based methods):

    - основаны на возможности соединения объектов;
    - игнорируют шумы, нахождение кластеров произвольной формы.
- Грид-методы (Grid-based methods):
    - квантование объектов в грид-структуры.
- Модельные методы (Model-based):
    - использование модели для нахождения кластеров, наиболее соответствующих данным.

Оценка качества кластеризации

Оценка качества кластеризации может быть проведена на основе следующих процедур:

- ручная проверка;
- установление контрольных точек и проверка на полученных кластерах;
- определение стабильности кластеризации путем добавления в модель новых переменных;
- создание и сравнение кластеров с использованием различных методов. Разные методы кластеризации могут создавать разные кластеры, и это является нормальным явлением. Однако создание схожих кластеров различными методами указывает на правильность кластеризации.

Процесс кластеризации

Процесс кластеризации зависит от выбранного метода и почти всегда является итеративным. Он может стать увлекательным процессом и включать множество экспериментов по выбору разнообразных параметров, например, меры расстояния, типа стандартизации переменных, количества кластеров и т.д. Однако эксперименты не должны быть самоцелью - ведь конечной целью кластеризации является получение содержательных сведений о структуре исследуемых данных. Полученные результаты требуют дальнейшей интерпретации, исследования и изучения свойств и характеристик объектов для возможности точного описания сформированных кластеров.

Применение кластерного анализа

Кластерный анализ применяется в различных областях. Он полезен, когда нужно классифицировать большое количество информации. Обзор многих опубликованных исследований, проводимых с помощью кластерного анализа, дал Хартиган (Hartigan, 1975).

Так, в медицине используется кластеризация заболеваний, лечения заболеваний или их симптомов, а также таксономия пациентов, препаратов и т.д. В археологии устанавливаются таксономии каменных сооружений и древних объектов и т.д. В маркетинге это может быть задача сегментации конкурентов и потребителей. В менеджменте примером задачи кластеризации будет разбиение персонала на различные группы, классификация потребителей и поставщиков, выявление схожих производственных ситуаций, при которых возникает брак. В медицине - классификация симптомов. В социологии задача кластеризации - разбиение респондентов на однородные группы.

Кластерный анализ в маркетинговых исследованиях

В маркетинговых исследованиях кластерный анализ применяется достаточно широко - как в теоретических исследованиях, так и практикующими маркетологами, решающими проблемы группировки различных объектов. При этом решаются вопросы о группах клиентов, продуктов и т.д.

Так, одной из наиболее важных задач при применении кластерного анализа в маркетинговых исследованиях является анализ поведения потребителя, а именно: группировка потребителей в однородные классы для получения максимально полного представления о поведении клиента из каждой группы и о факторах, влияющих на его поведение. Эта проблема подробно описана в работах Клакстона, Фрая и Портиса (1974), Киля и Лэйтона (1981).

Важной задачей, которую может решить кластерный анализ, является позиционирование, т.е. определение ниши, в которой следует позиционировать новый продукт, предлагаемый на рынке. В результате применения кластерного анализа строится карта, по которой можно определить уровень конкуренции в различных сегментах рынка и соответствующие характеристики товара для возможности попадания в этот сегмент. С помощью анализа такой карты возможно определение новых, незанятых ниш на рынке, в которых можно предлагать существующие товары или разрабатывать новые.

Кластерный анализ также может быть удобен, например, для анализа клиентов компании. Для этого все клиенты группируются в кластеры, и для каждого кластера вырабатывается индивидуальная политика. Такой подход позволяет существенно сократить объекты анализа, и, в то же время, индивидуально подойти к каждой группе клиентов.

Практика применения кластерного анализа в маркетинговых исследованиях

Приведем некоторые известные статьи, посвященные применению кластерного анализа для маркетинговых исследований.

В 1971 году была опубликована статья о сегментации клиентов по сфере интересов на основе данных, характеризующих предпочтения клиентов.

В 1974 году была опубликована статья Секстона (Sexton), целью которой была идентификация групп семей - потребителей продукта, в результате были разработаны стратегии позиционирования бренда. Основой для исследований были рейтинги, которые респонденты присваивали продуктам и брендам.

В 1981 году была опубликована статья, где проводился анализ поведения покупателей новых автомобилей на основе данных факторных нагрузок, полученных при анализе набора переменных.

Выводы

В этой лекции нами были подробно рассмотрены задачи классификации и кластеризации. Несмотря на кажущуюся похожесть этих задач, решаются они разными способами и при помощи разных методов. Различие задач прежде всего в исходных данных.

Классификация, являясь наиболее простой задачей Data Mining, относится к стратегии "обучение с учителем", для ее решения обучающая выборка должна содержать значения как входных переменных, так и выходных (целевых) переменных. Кластеризация, напротив, является задачей Data Mining, относящейся к стратегии "обучение без учителя", т.е. не требует наличия значения целевых переменных в обучающей выборке.

Задача классификации решается при помощи различных методов, наиболее простой - линейная регрессия. Выбор метода должен базироваться на исследовании исходного набора данных. Наиболее распространенные методы решения задачи кластеризации: метод k-средних (работает только с числовыми атрибутами), иерархический кластерный анализ (работает также с символьными атрибутами), метод SOM. Сложностью кластеризации является необходимость ее оценки.

Задачи Data Mining. Прогнозирование и визуализация

Мы продолжаем рассматривать наиболее распространенные и востребованные задачи Data Mining. В этой лекции мы подробно остановимся на задачах прогнозирования и визуализации.

Задача прогнозирования

Задачи прогнозирования решаются в самых разнообразных областях человеческой деятельности, таких как наука, экономика, производство и множество других сфер. Прогнозирование является важным элементом организации управления как отдельными хозяйствующими субъектами, так и экономики в целом.

Развитие методов прогнозирования непосредственно связано с развитием информационных технологий, в частности, с ростом объемов хранимых данных и усложнением методов и алгоритмов прогнозирования, реализованных в инструментах Data Mining.

Задача прогнозирования, пожалуй, может считаться одной из наиболее сложных задач Data Mining, она требует тщательного исследования исходного набора данных и методов, подходящих для анализа.

Прогнозирование (от греческого Prognosis), в широком понимании этого слова, определяется как опережающее отражение будущего. Целью прогнозирования является предсказание будущих событий.

Прогнозирование (forecasting) является одной из задач Data Mining и одновременно одним из ключевых моментов при принятии решений.

Прогностика (prognostics) - теория и практика прогнозирования.

Прогнозирование направлено на определение тенденций динамики конкретного объекта или события на основе ретроспективных данных, т.е. анализа его состояния в прошлом и настоящем. Таким образом, решение задачи прогнозирования требует некоторой обучающей выборки данных.

Прогнозирование - установление функциональной зависимости между зависимыми и независимыми переменными.

Прогнозирование является распространенной и востребованной задачей во многих областях человеческой деятельности. В результате прогнозирования уменьшается риск принятия неверных, необоснованных или субъективных решений.

Примеры его задач: прогноз движения денежных средств, прогнозирование урожайности агрокультуры, прогнозирование финансовой устойчивости предприятия.

Типичной в сфере маркетинга является задача прогнозирования рынков (market forecasting). В результате решения данной задачи оцениваются перспективы развития

конъюнктуры определенного рынка, изменения рыночных условий на будущие периоды, определяются тенденции рынка (структурные изменения, потребности покупателей, изменения цен).

Обычно в этой области решаются следующие практические задачи:

- прогноз продаж товаров (например, с целью определения нормы товарного запаса);
- прогнозирование продаж товаров, оказывающих влияние друг на друга;
- прогноза продаж в зависимости от внешних факторов.

Помимо экономической и финансовой сферы, задачи прогнозирования ставятся в самых разнообразных областях: медицине, фармакологии; популярным сейчас становится политическое прогнозирование.

В самых общих чертах решение задачи прогнозирования сводится к решению таких подзадач:

- выбор модели прогнозирования;
- анализ адекватности и точности построенного прогноза.

Сравнение задач прогнозирования и классификации

В предыдущей лекции нами была рассмотрена задача классификации. Прогнозирование сходно с задачей классификации.

Многие методы Data Mining используются для решения задач классификации и прогнозирования. Это, например, линейная регрессия, нейронные сети, деревья решений (которые иногда так и называют - деревья прогнозирования и классификации).

Задачи классификации и прогнозирования имеют сходства и различия.

Так в чем же сходство задач прогнозирования и классификации? При решении обеих задач используется двухэтапный процесс построения модели на основе обучающего набора и ее использования для предсказания неизвестных значений зависимой переменной.

Различие задач классификации и прогнозирования состоит в том, что в первой задаче предсказывается класс зависимой переменной, а во второй - числовые значения зависимой переменной, пропущенные или неизвестные (относящиеся к будущему).

Возвращаясь к примеру о туристическом агентстве, рассмотренном в предыдущей лекции, мы можем сказать, что определения класса клиента является решением задачи классификации, а прогнозирование дохода, который принесет этот клиент в будущем году, будет решением задачи прогнозирования.

Прогнозирование и временные ряды

Основой для прогнозирования служит историческая информация, хранящаяся в базе данных в виде временных рядов.

Существует понятие Data Mining временных рядов (Time-Series Data Mining).

Подробно с этим понятием можно ознакомиться в [23].

На основе ретроспективной информации в виде временных рядов возможно решение различных задач Data Mining. На рис. 6.1 представлены результаты опроса относительно Data Mining временных рядов. Как видим, наибольший процент (23%) среди решаемых задач занимает прогнозирование. Далее идут классификация и кластеризация (по 14%), сегментация и выявление аномалий (по 9%), обнаружение правил (8%). На другие задачи приходится менее чем по 6%.

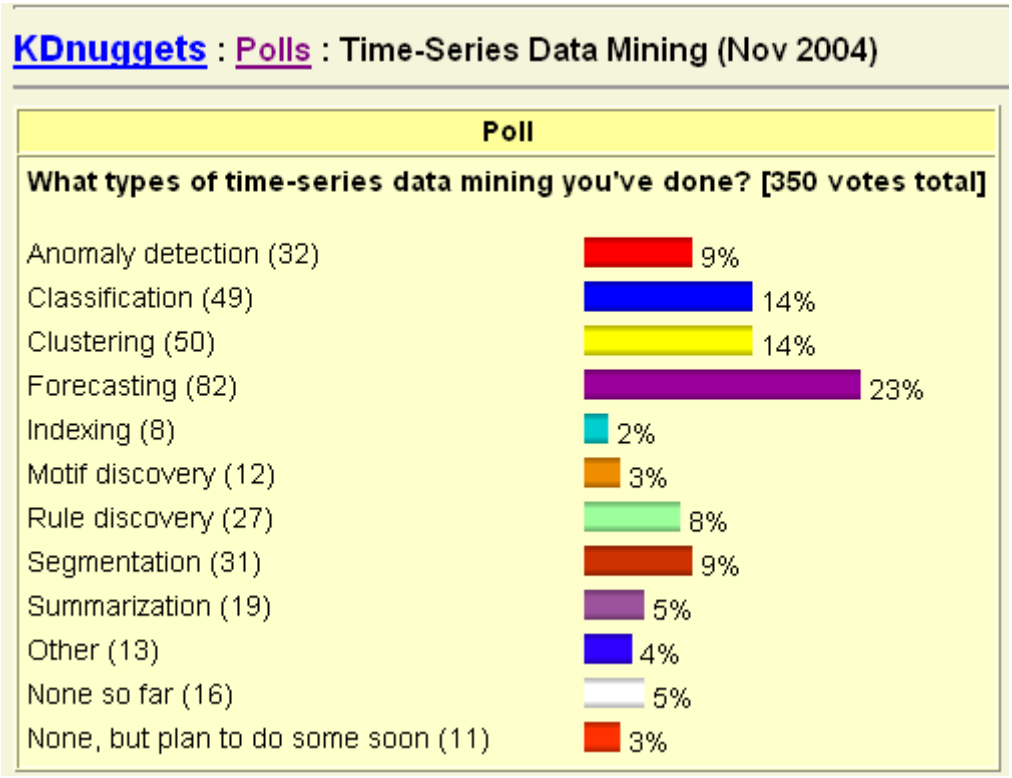

Рис. 6.1. Data Mining временных рядов

Однако чтобы сосредоточиться на понятии прогнозирования, мы будем рассматривать временные ряды лишь в рамках решения задачи прогнозирования.

Приведем два принципиальных отличия временного ряда от простой последовательности наблюдений:

- Члены временного ряда, в отличие от элементов случайной выборки, не являются статистически независимыми.
- Члены временного ряда не являются одинаково распределенными.

Временной ряд - последовательность наблюдаемых значений какого-либо признака, упорядоченных в неслучайные моменты времени.

Отличием анализа временных рядов от анализа случайных выборок является предположение о равных промежутках времени между наблюдениями и их хронологический порядок. Привязка наблюдений ко времени играет здесь ключевую роль, тогда как при анализе случайной выборки она не имеет никакого значения.

Типичный пример временного ряда - данные биржевых торгов.

Информация, накопленная в разнообразных базах данных предприятия, является временными рядами, если она расположена в хронологическом порядке и произведена в последовательные моменты времени.

Анализ временного ряда осуществляется с целью:

- определения природы ряда;
- прогнозирования будущих значений ряда.

В процессе определения структуры и закономерностей временного ряда предполагается обнаружение: шумов и выбросов, тренда, сезонной компоненты, циклической компоненты. Определение природы временного ряда может быть использовано как своеобразная "разведка" данных. Знание аналитика о наличии сезонной компоненты необходимо, например, для определения количества записей выборки, которое должно принимать участие в построении прогноза.

Шумы и выбросы будут подробно обсуждаться в последующих лекциях курса. Они усложняют анализ временного ряда. Существуют различные методы определения и фильтрации выбросов, дающие возможность исключить их с целью более качественного Data Mining.

Тренд, сезонность и цикл

Основными составляющими временного ряда являются тренд и сезонная компонента. Составляющие этих рядов могут представлять собой либо тренд, либо сезонную компоненту.

Тренд является систематической компонентой временного ряда, которая может изменяться во времени.

Трендом называют неслучайную функцию, которая формируется под действием общих или долговременных тенденций, влияющих на временной ряд.

Примером тенденции может выступать, например, фактор роста исследуемого рынка.

Автоматического способа обнаружения трендов во временных рядах не существует. Но если временной ряд включает монотонный тренд (т.е. отмечено его устойчивое возрастание или устойчивое убывание), анализировать временной ряд в большинстве случаев нетрудно.

Существует большое разнообразие постановок задач прогнозирования, которое можно подразделить на две группы [24]: прогнозирование односерийных рядов и прогнозирование мультисерийных, или взаимовлияющих, рядов.

Группа прогнозирование односерийных рядов включает задачи построения прогноза одной переменной по ретроспективным данным только этой переменной, без учета влияния других переменных и факторов.

Группа прогнозирования мультисерийных, или взаимовлияющих, рядов включает задачи анализа, где необходимо учитывать взаимовлияющие факторы на одну или несколько переменных.

Кроме деления на классы по односерийности и многосерийности, ряды также бывают сезонными и несезонными.

Последнее деление подразумевает наличие или отсутствие у временного ряда такой составляющей как сезонность, т.е. включение сезонной компоненты.

Сезонная составляющая временного ряда является периодически повторяющейся компонентой временного ряда.

Свойство сезонности означает, что через примерно равные промежутки времени форма кривой, которая описывает поведение зависимой переменной, повторяет свои характерные очертания.

Свойство сезонности важно при определении количества ретроспективных данных, которые будут использоваться для прогнозирования.

Рассмотрим простой пример. На рис. 6.2. приведен фрагмент ряда, который иллюстрирует поведение переменной "объемы продажи товара X" за период, составляющий один месяц. При изучении кривой, приведенной на рисунке, аналитик не может сделать предположений относительно повторяемости формы кривой через равные промежутки времени.

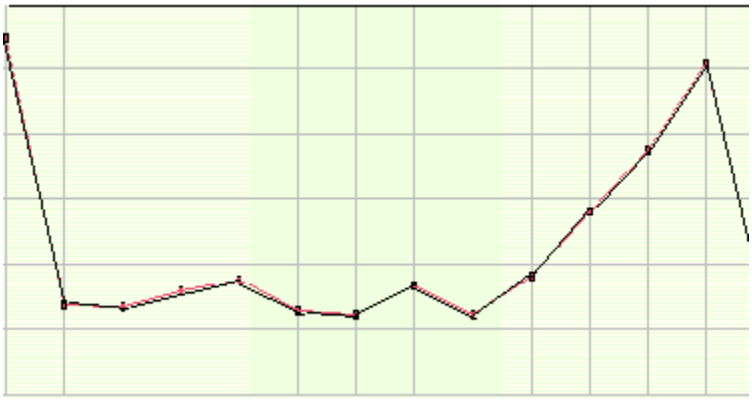

Рис. 6.2. Фрагмент временного ряда за сезонный период

Однако при рассмотрении более продолжительного ряда (за 12 месяцев), изображенного на рис. 6.3, можно увидеть явное наличие сезонной компоненты. Следовательно, о сезонности продаж можно говорить только, когда рассматриваются данные за несколько месяцев.

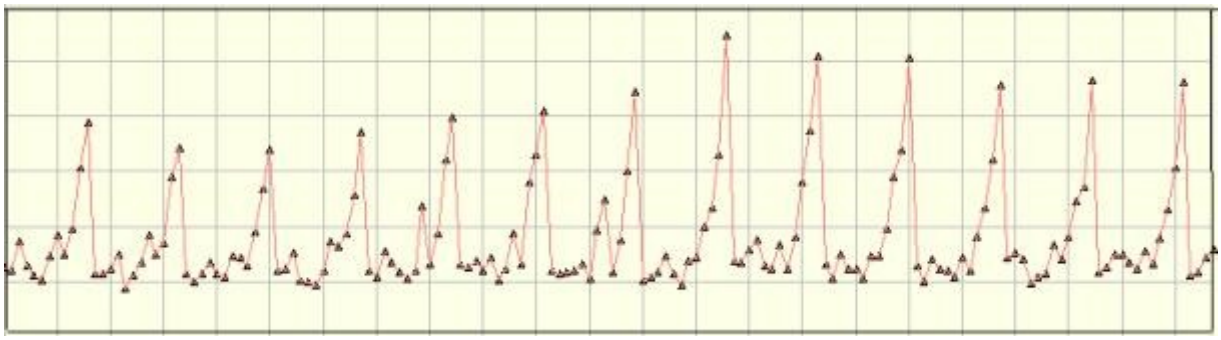

Рис. 6.3. Фрагмент временного ряда за 12-ти сезонных периодов

Таким образом, в процессе подготовки данных для прогнозирования аналитику следует определить, обладает ли ряд, который он анализирует, свойством сезонности.

Определение наличия компоненты сезонности необходимо для того, чтобы входная информация обладала свойством репрезентативности.

Ряд можно считать несезонным, если при рассмотрении его внешнего вида нельзя сделать предположений о повторяемости формы кривой через равные промежутки времени.

Иногда по внешнему виду кривой ряда нельзя определить, является он сезонным или нет.

Существует понятие сезонного мультиряда. В нем каждый ряд описывает поведение факторов, которые влияют на зависимую (целевую) переменную.

Пример такого ряда - ряды продаж нескольких товаров, подверженных сезонным колебаниям.

При сборе данных и выборе факторов для решения задачи по прогнозированию в таких случаях следует учитывать, что влияние объемов продаж товаров друг на друга здесь намного меньше, чем воздействие фактора сезонности.

Важно не путать понятия сезонной компоненты ряда и сезонов природы. Несмотря на близость их звучания, эти понятия разнятся. Так, например, объемы продаж мороженого летом намного больше, чем в другие сезоны, однако это является тенденцией спроса на данный товар.

Очень часто тренд и сезонность присутствуют во временном ряде одновременно.

Пример. Прибыль фирмы растет на протяжении нескольких лет (т.е. во временном ряде присутствует тренд); ряд также содержит сезонную компоненту.

Отличия циклической компоненты от сезонной:

1. Продолжительность цикла, как правило, больше, чем один сезонный период;
2. Циклы, в отличие от сезонных периодов, не имеют определенной продолжительности.

При выполнении каких-либо преобразований понять природу временного ряда значительно проще, такими преобразованиями могут быть, например, удаление тренда и сглаживание ряда.

Перед началом прогнозирования необходимо ответить на следующие вопросы:

1. Что нужно прогнозировать?
2. В каких временных элементах (параметрах)?
3. С какой точностью прогноза?

При ответе на первый вопрос, мы определяем переменные, которые будут прогнозироваться. Это может быть, например, уровень производства конкретного вида продукции в следующем квартале, прогноз суммы продажи этой продукции и т.д.

При выборе переменных следует учитывать доступность ретроспективных данных, предпочтения лиц, принимающих решения, окончательную стоимость Data Mining.

Часто при решении задач прогнозирования возникает необходимость предсказания не самой переменной, а изменений ее значений.

Второй вопрос при решении задачи прогнозирования - определение следующих параметров:

- периода прогнозирования;
- горизонта прогнозирования;
- интервала прогнозирования.

Период прогнозирования - основная единица времени, на которую делается прогноз.

Например, мы хотим узнать доход компании через месяц. Период прогнозирования для этой задачи - месяц.

Горизонт прогнозирования - это число периодов в будущем, которые покрывает прогноз.

Если мы хотим узнать прогноз на 12 месяцев вперед, с данными по каждому месяцу, то период прогнозирования в этой задаче - месяц, горизонт прогнозирования - 12 месяцев.

Интервал прогнозирования - частота, с которой делается новый прогноз.

Интервал прогнозирования может совпадать с периодом прогнозирования.

Рекомендации по выбору параметров прогнозирования.

При выборе параметров необходимо учитывать, что горизонт прогнозирования должен быть не меньше, чем время, которое необходимо для реализации решения, принятого на основе этого прогноза. Только в этом случае прогнозирование будет иметь смысл.

С увеличением горизонта прогнозирования точность прогноза, как правило, снижается, а с уменьшением горизонта - повышается.

Мы можем улучшить качество прогнозирования, уменьшая время, необходимое на реализацию решения, для которого реализуется прогноз, и, следовательно, уменьшив при этом горизонт и ошибку прогнозирования.

При выборе интервала прогнозирования следует выбирать между двумя рисками: вовремя не определить изменения в анализируемом процессе и высокой стоимостью прогноза. При длительном интервале прогнозирования возникает риск не идентифицировать изменения, произошедшие в процессе, при коротком - возрастают издержки на прогнозирование.

При выборе интервала необходимо также учитывать стабильность анализируемого процесса и стоимость проведения прогноза.

Точность прогноза

Точность прогноза, требуемая для решения конкретной задачи, оказывает большое влияние на прогнозирующую систему. Ошибка прогноза зависит от используемой системы прогноза.

Чем больше ресурсов имеет такая система, тем больше шансов получить более точный прогноз. Однако прогнозирование не может полностью уничтожить риски при принятии решений. Поэтому всегда учитывается возможная ошибка прогнозирования.

Точность прогноза характеризуется ошибкой прогноза.

Наиболее распространенные виды ошибок:

- **Средняя ошибка (СО).** Она вычисляется простым усреднением ошибок на каждом шаге. Недостаток этого вида ошибки - положительные и отрицательные ошибки аннулируют друг друга.
- **Средняя абсолютная ошибка (САО)**. Она рассчитывается как среднее абсолютных ошибок. Если она равна нулю, то мы имеем совершенный прогноз. В сравнении со средней квадратической ошибкой, эта мера "не придает слишком большого значения" выбросам.
- **Сумма квадратов ошибок (SSE)**, среднеквадратическая ошибка. Она вычисляется как сумма (или среднее) квадратов ошибок. Это наиболее часто используемая оценка точности прогноза.
- **Относительная ошибка (ОО).** Предыдущие меры использовали действительные значения ошибок. Относительная ошибка выражает качество подгонки в терминах относительных ошибок.

Виды прогнозов

Прогноз может быть краткосрочным, среднесрочным и долгосрочным.

Краткосрочный прогноз представляет собой прогноз на несколько шагов вперед, т.е. осуществляется построение прогноза не более чем на 3% от объема наблюдений или на 1-3 шага вперед.

Среднесрочный прогноз - это прогноз на 3-5% от объема наблюдений, но не более 7-12 шагов вперед; также под этим типом прогноза понимают прогноз на один или половину сезонного цикла. Для построения краткосрочных и среднесрочных прогнозов вполне подходят статистические методы.

Долгосрочный прогноз - это прогноз более чем на 5% от объема наблюдений.

При построении данного типа прогнозов статистические методы практически не используются, кроме случаев очень "хороших" рядов, для которых прогноз можно просто "нарисовать".

До сих пор мы рассматривали аспекты прогнозирования, так или иначе связанные с процессом принятия решения. Существуют и другие факторы, которые необходимо учитывать при прогнозировании.

Задача 1. Известно, что анализируемый процесс относительно стабилен во времени, изменения происходят медленно, процесс не зависит от внешних факторов.

Задача 2. Анализируемый процесс нестабилен и очень сильно зависит от внешних факторов.

Решение первой задачи должно быть сосредоточено на использовании большого количества ретроспективных данных. При решении второй задачи особое внимание следует обратить на оценки специалиста в предметной области, эксперта, чтобы иметь возможность отразить в прогнозирующей модели все необходимые внешние факторы, а также уделить время для сбора данных по этим факторам (сбор внешних данных часто намного сложнее сбора внутренних данных информационной системы). Доступность данных, на основе которых будет осуществляться прогнозирование, - важный фактор построения прогнозной модели. Для возможности выполнения качественного прогноза данные должны быть представительными, точными и достоверными.

Методы прогнозирования

Методы Data Mining, при помощи которых решаются задачи прогнозирования, будут рассмотрены во втором разделе курса. Среди распространенных методов Data Mining, используемых для прогнозирования, отметим нейронные сети и линейную регрессию.

Выбор метода прогнозирования зависит от многих факторов, в том числе от параметров прогнозирования. Выбор метода следует производить с учетом всех специфических особенностей набора ретроспективных данных и целей, с которыми он строится.

Программное обеспечение Data Mining, используемое для прогнозирования, должно обеспечивать пользователя точным и достоверным прогнозом. Однако получение такого прогноза зависит не только от программного обеспечения и методов, заложенных в его основу, но также и от других факторов, среди которых полнота и достоверность исходных данных, своевременность и оперативность их пополнения, квалификация пользователя.

Задача визуализации

Визуализация - это инструментарий, который позволяет увидеть конечный результат вычислений, организовать управление вычислительным процессом и даже вернуться назад к исходным данным, чтобы определить наиболее рациональное направление дальнейшего движения [25].

С задачей визуализации можно подробно ознакомиться по материалам конференций, среди которых, например, CHI и ACM-SIGGraph, а также в периодической литературе, в частности, по материалам журнала "IEEE Trans. visualization and computer graphics".

В результате использования визуализации создается графический образ данных. Применение визуализации помогает в процессе анализа данных увидеть аномалии, структуры, тренды. При рассмотрении задачи прогнозирования мы использовали графическое представление временного ряда и увидели, что в нем присутствует сезонная компонента. В предыдущей лекции мы рассматривали задачи классификации и кластеризации, и для иллюстрации распределения объектов в двухмерном пространстве также использовали визуализацию.

Можно говорить о том, что применение визуализации является более экономичным: линия тренда или скопления точек на диаграмме рассеивания позволяет аналитику намного быстрее определить закономерности и прийти к нужному решению. Таким образом, здесь идет речь об использовании в Data Mining не символов, а образов.

Главное преимущество визуализации - практически полное отсутствие необходимости в специальной подготовке пользователя. При помощи визуализации ознакомиться с информацией очень легко, достаточно всего лишь бросить на нее взгляд.

Хотя простейшие виды визуализации появились достаточно давно, ее использование сейчас только набирает силу. Визуализации не направлена исключительно на совершенствование техники анализа - по словам Скотта Лейбса, в некоторых случаях визуализация может даже заменить его.

Визуализации данных может быть представлена в виде: графиков, схем, гистограмм, диаграмм и т.д.

Кратко роль визуализации можно описать такими ее возможностями:

- поддержка интерактивного и согласованного исследования;
- помощь в представлении результатов;
- использование глаз (зрения), чтобы создавать зрительные образы и осмысливать их.

Плохая визуализация

Результаты визуализации иногда могут вводить пользователя в заблуждение. Приведем простой пример плохой визуализации. Допустим, мы имеем базу "Прибыль компании А" за период с 2000 по 2005 года, она представлена в табличном виде в таблице 6.1.

Таблица 6.1. Прибыль компании А

год	прибыль
2000	1100
2001	1101
2002	1104

2003	1105
2004	1106
2005	1107

Построим гистограмму в Excel по этим данным.

Гистограмма представляет собой визуальное изображение распределения данных.

Эта информация отображается при помощи серии прямоугольников или полос одинаковой ширины, высота которых указывает количество данных в каждом классе.

Используя все значения построения графика, принятые по умолчанию, получаем гистограмму, приведенную на рис. 6.4.

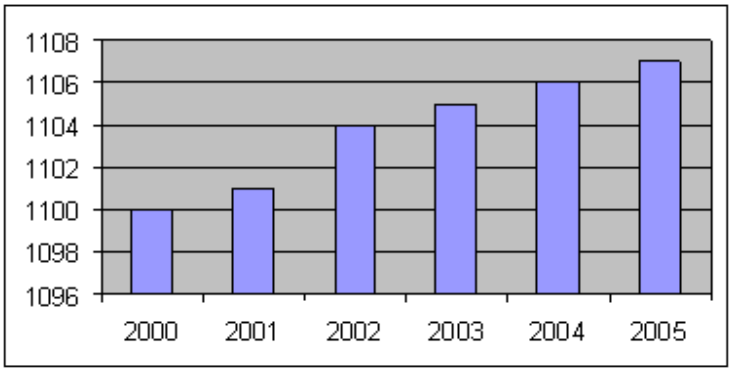

Рис. 6.4. Гистограмма, минимальное значение оси y равно 1096

Данный рисунок демонстрирует значительный рост прибыли компании А за период с 2000 по 2005 года. Однако, если мы обратим внимание на ось y, показывающую величину прибыли, то увидим, что эта ось пересекает ось x в значении, равном 1096. Фактически, ось y со значениями от 1096 до 1108 вводит пользователя в заблуждение. Изменив значения параметров, отвечающих за формат оси y, получаем график, приведенный на рис. 6.5.

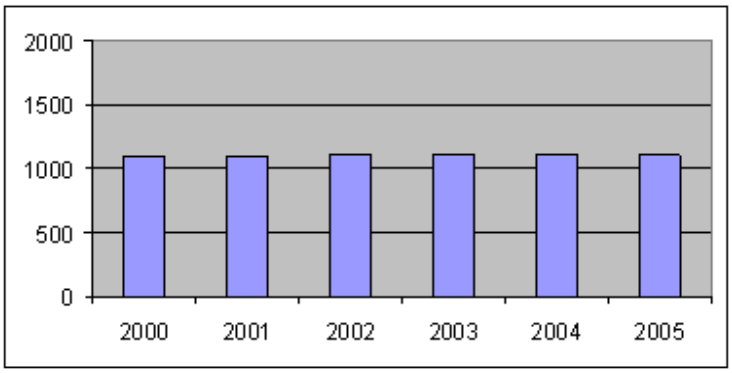

Рис. 6.5. Гистограмма, минимальное значение оси y равно 0

Ось y со значениями от 0 до 2000 дает пользователю правильную информацию о незначительном изменении прибыли компании.

Если речь идет о большой размерности и сложности исходных данных, средства визуализации обеспечивают их резкое уменьшение, конденсируя, быть может, миллионы записей данных в простые, легкие для понимания и манипулирования представления [26]. Такие представления называют визуальным или графическим способом представления информации. Визуализацию можно считать ключевым фактором в исследовании данных, полученных при помощи инструментов Data Mining. В таких случаях говорят о визуальном Data Mining.

Методы визуализации, среди которых представления информации в одно-, двух-, трехмерном и более измерениях, а также другие способы отображения информации, например, параллельные координаты, "лица Чернова", будут рассмотрены в следующем разделе курса.

Сферы применения Data Mining

В предыдущих лекциях мы рассмотрели задачи и методы Data Mining. Однако вводная часть не будет полной, если не рассмотреть, для каких конкретных задач и в каких сферах жизнедеятельности человека можно использовать эту технологию. Следует сразу сказать, что область использования Data Mining ничем не ограничена - она везде, где имеются какие-либо данные. В этой лекции мы рассмотрим всевозможные сферы применения Data Mining.

Цель этого обзора есть не перечисление абсолютно всех сфер применения, а знакомство с теми направлениями, где Data Mining работает и дает реальные результаты.

В [16] выделены два направления применения систем Data Mining: как массового продукта и как инструмента для проведения уникальных исследований.

Следует отметить, что на сегодняшний день наибольшее распространение технология Data Mining получила при решении бизнес-задач. Возможно, причина в том, что именно в этом направлении отдача от использования инструментов Data Mining может составлять, по некоторым источникам, до 1000% и затраты на ее внедрение могут достаточно быстро окупиться.

Сейчас технология Data Mining используется практически во всех сферах деятельности человека, где накоплены ретроспективные данные.

Мы будем рассматривать четыре основные сферы применения технологии Data Mining подробно [22, 27]: наука, бизнес, исследования для правительства и Web-направление.

- Применение Data Mining для решения бизнес-задач. Основные направления: банковское дело, финансы, страхование, CRM, производство, телекоммуникации, электронная коммерция, маркетинг, фондовый рынок и другие.

- Применение Data Mining для решения задач государственного уровня. Основные направления: поиск лиц, уклоняющихся от налогов; средства в борьбе с терроризмом.
- Применение Data Mining для научных исследований. Основные направления: медицина, биология, молекулярная генетика и генная инженерия, биоинформатика, астрономия, прикладная химия, исследования, касающиеся наркотической зависимости, и другие.
- Применение Data Mining для решения Web-задач. Основные направления: поисковые машины (search engines), счетчики и другие.

Применение Data Mining для решения бизнес-задач

Банковское дело

Технология Data Mining используется в банковской сфере для решения ряда типичных задач.

Задача "Выдавать ли кредит клиенту?"

Классический пример применения Data Mining в банковском деле - решение задачи определения возможной некредитоспособности клиента банка. Эту задачу также называют анализом кредитоспособности клиента или "Выдавать ли кредит клиенту?".

Без применения технологии Data Mining задача решается сотрудниками банковского учреждения на основе их опыта, интуиции и субъективных представлений о том, какой клиент является благонадежным. По похожей схеме работают системы поддержки принятия решений и на основе методов Data Mining. Такие системы на основе исторической (ретроспективной) информации и при помощи методов классификации выявляют клиентов, которые в прошлом не вернули кредит.

Задача "Выдавать ли кредит клиенту?" при помощи методов Data Mining решается следующим образом. Совокупность клиентов банка разбивается на два класса (вернувшие и не вернувшие кредит); на основе группы клиентов, не вернувших кредит, определяются основные "черты" потенциального неплательщика; при поступлении информации о новом клиенте определяется его класс ("вернет кредит", "не вернет кредит").

Задача привлечения новых клиентов банка.

С помощью инструментов Data Mining возможно провести классификацию на "более выгодных" и "менее выгодных" клиентов. После определения наиболее выгодного сегмента клиентов банку есть смысл проводить более активную маркетинговую политику по привлечению клиентов именно среди найденной группы.

Другие задачи сегментации клиентов.

Разбивая клиентов при помощи инструментов Data Mining на различные группы, банк имеет возможность сделать свою маркетинговую политику более целенаправленной, а потому - эффективной, предлагая различным группам клиентов именно те виды услуг, в которых они нуждаются.

Задача управления ликвидностью банка. Прогнозирование остатка на счетах клиентов.

Проводя прогнозирования временного ряда с информацией об остатках на счетах клиентов за предыдущие периоды, применяя методы Data Mining, можно получить прогноз остатка на счетах в определенный момент в будущем. Полученные результаты могут быть использованы для оценки и управления ликвидностью банка.

Задача выявления случаев мошенничества с кредитными карточками.

Для выявления подозрительных операций с кредитными карточками применяются так называемые "подозрительные стереотипы поведения", определяемые в результате анализа банковских транзакций, которые впоследствии оказались мошенническими. Для определения подозрительных случаев используется совокупность последовательных операций на определенном временном интервале. Если система Data Mining считает очередную операцию подозрительной, банковский работник может, ориентируясь на эту информацию, заблокировать операции с определенной карточкой.

Страхование

Страховой бизнес связан с определенным риском. Здесь задачи, решаемые при помощи Data Mining, сходны с задачами в банковском деле.

Информация, полученная в результате сегментации клиентов на группы, используется для определения групп клиентов. В результате страховая компания может с наибольшей выгодой и наименьшим риском предлагать определенные группы услуг конкретным группам клиентов.

Задача выявление мошенничества решается путем нахождения некого общего стереотипа поведения клиентов-мошенников.

Телекоммуникации

В сфере телекоммуникаций достижения Data Mining могут использоваться для решения задачи, типичной для любой компании, которая работает с целью привлечения постоянных клиентов, - определения лояльности этих клиентов. Необходимость решения таких задач обусловлена жесткой конкуренцией на рынке телекоммуникаций и постоянной миграцией клиентов от одной компании в другую. Как известно, удержание клиента намного дешевле его возврата. Поэтому возникает необходимость выявления определенных групп клиентов и разработка наборов услуг, наиболее привлекательных именно для них. В этой сфере, так же как и во многих других, важной задачей является выявление фактов мошенничества.

Помимо таких задач, являющихся типичными для многих областей деятельности, существует группа задач, определяемых спецификой сферы телекоммуникаций.

Электронная коммерция

В сфере электронной коммерции Data Mining применяется для формирования рекомендательных систем и решения задач классификации посетителей Web-сайтов. Такая классификация позволяет компаниям выявлять определенные группы клиентов и проводить маркетинговую политику в соответствии с обнаруженными интересами и потребностями клиентов. Технология Data Mining для электронной коммерции тесно связана с технологией Web Mining [28].

Промышленное производство

Особенности промышленного производства и технологических процессов создают хорошие предпосылки для возможности использования технологии Data Mining в ходе решения различных производственных задач. Технический процесс по своей природе должен быть контролируемым, а все его отклонения находятся в заранее известных пределах;

т.е. здесь мы можем говорить об определенной стабильности, которая обычно не присуща большинству задач, встающих перед технологией Data Mining.

Основные задачи Data Mining в промышленном производстве [29]:

- комплексный системный анализ производственных ситуаций;
- краткосрочный и долгосрочный прогноз развития производственных ситуаций;
- выработка вариантов оптимизационных решений;
- прогнозирование качества изделия в зависимости от некоторых параметров технологического процесса;
- обнаружение скрытых тенденций и закономерностей развития производственных процессов;
- прогнозирование закономерностей развития производственных процессов;
- обнаружение скрытых факторов влияния;
- обнаружение и идентификация ранее неизвестных взаимосвязей между производственными параметрами и факторами влияния;
- анализ среды взаимодействия производственных процессов и прогнозирование изменения ее характеристик;
- выработку оптимизационных рекомендаций по управлению производственными процессами;
- визуализацию результатов анализа, подготовку предварительных отчетов и проектов допустимых решений с оценками достоверности и эффективности возможных реализаций.

Маркетинг

В сфере маркетинга Data Mining находит очень широкое применение.

Основные вопросы маркетинга "Что продается?", "Как продается?", "Кто является потребителем?"

В лекции, посвященной задачам классификации и кластеризации, подробно описано использование кластерного анализа для решения задач маркетинга, как, например, сегментация потребителей.

Другой распространенный набор методов для решения задач маркетинга - методы и алгоритмы поиска ассоциативных правил.

Также успешно здесь используется поиск временных закономерностей.

Розничная торговля

В сфере розничной торговли, как и в маркетинге, применяются:

- алгоритмы поиска ассоциативных правил (для определения часто встречающихся наборов товаров, которые покупатели покупают одновременно). Выявление таких правил помогает размещать товары на прилавках торговых залов, вырабатывать стратегии закупки товаров и их размещения на складах и т.д.
- использование временных последовательностей, например, для определения необходимых объемов запасов товаров на складе.
- методы классификации и кластеризации для определения групп или категорий клиентов, знание которых способствует успешному продвижению товаров.

Фондовый рынок

Вот список задач фондового рынка, которые можно решать при помощи технологии Data Mining [30]:

- прогнозирование будущих значений финансовых инструментов и индикаторов по их прошлым значениям;
- прогноз тренда (будущего направления движения - рост, падение, флэт) финансового инструмента и его силы (сильный, умеренно сильный и т.д.);
- выделение кластерной структуры рынка, отрасли, сектора по некоторому набору характеристик;
- динамическое управление портфелем;
- прогноз волатильности;
- оценка рисков;
- предсказание наступления кризиса и прогноз его развития;
- выбор активов и др.

Кроме описанных выше сфер деятельности, технология Data Mining может применяться в самых разнообразных областях бизнеса, где есть необходимость в анализе данных и накоплен некоторый объем ретроспективной информации.

Применение Data Mining в CRM

Одно из наиболее перспективных направлений применения Data Mining - использование данной технологии в аналитическом CRM.

CRM (Customer Relationship Management) - управление отношениями с клиентами.

При совместном использовании этих технологий добыча знаний совмещается с "добычей денег" из данных о клиентах.

Важным аспектом в работе отделов маркетинга и отдела продаж является составление целостного представления о клиентах, информация об их особенностях, характеристиках, структуре клиентской базы. В CRM используется так называемое профилирование клиентов, дающее полное представление всей необходимой информации о клиентах. Профилирование клиентов включает следующие компоненты: сегментация клиентов, прибыльность клиентов, удержание клиентов, анализ реакции клиентов. Каждый из этих компонентов может исследоваться при помощи Data Mining, а анализ их в совокупности, как компонентов профилирования, в результате может дать те знания, которые из каждой отдельной характеристики получить невозможно.

В результате использования Data Mining решается задача сегментации клиентов на основе их прибыльности. Анализ выделяет те сегменты покупателей, которые приносят наибольшую прибыль. Сегментация также может осуществляться на основе лояльности клиентов. В результате сегментации вся клиентская база будет поделена на определенные сегменты, с общими характеристиками. В соответствии с этими характеристиками компания может индивидуально подбирать маркетинговую политику для каждой группы клиентов.

Также можно использовать технологию Data Mining для прогнозирования реакции определенного сегмента клиентов на определенный вид рекламы или рекламных акций - на основе ретроспективных данных, накопленных в предыдущие периоды.

Таким образом, определяя закономерности поведения клиентов при помощи технологии Data Mining, можно существенно повысить эффективность работы отделов маркетинга, продаж и сбыта. При объединении технологий CRM и Data Mining и грамотном их внедрении в бизнес компания получает значительные преимущества перед конкурентами.

Исследования для правительства

В планах правительства США стоит создание системы, которая позволит отслеживать всех иностранцев, приезжающих в страну. Задача этого комплекса: начиная с пограничного терминала, на основе технологии биометрической идентификации личности и различных других баз данных контролировать, насколько реальные планы иностранцев соответствуют заявленным ранее (включая перемещения по стране, сроки отъезда и др.). Предварительная стоимость системы составляет более 10 млрд. долларов, разработчик комплекса - компания Accenture.

По данным аналитического отчета Главного контрольного управления американского Конгресса, правительственные ведомства США участвуют приблизительно в двухстах проектах на основе анализа данных (Data Mining), собирающих разнообразную информацию о населении. Более ста из этих проектов направлены на сбор персональной информации (имена, фамилии, адреса e-mail, номера соцстрахования и удостоверений водительских прав), и на основе этой информации осуществляют предсказания возможного поведения людей. Поскольку в упомянутом отчете не приведена информация о секретных отчетах, надо полагать, что общее число таких систем значительно больше.

Несмотря на пользу, которую приносят системы отслеживания, эксперты упомянутого управления, так же как и независимые эксперты, предупреждают о значительном риске, с которым связаны подобные проекты. Причина опасений - проблемы, которые могут возникнуть при управлении и надзоре за такими базами.

Data Mining для научных исследований

Биоинформатика

Одна из научных областей применения технологии Data Mining - биоинформатика, направление, целью которого является разработка алгоритмов для анализа и систематизации генетической информации. Полученные алгоритмы используются для определения структур макромолекул, а также их функций, с целью объяснения различных биологических явлений.

Медицина

Несмотря на консервативность медицины во многих ее аспектах, технология Data Mining в последние годы активно применяется для различных исследований и в этой сфере человеческой деятельности. Традиционно для постановки медицинских диагнозов используются экспертные системы, которые построены на основе символьных правил, сочетающих, например, симптомы пациента и его заболевание. С использованием Data Mining при помощи шаблонов можно разработать базу знаний для экспертной системы.

Фармацевтика

В области фармацевтики методы Data Mining также имеют достаточно широкое применение. Это задачи исследования эффективности клинического применения определенных препаратов, определение групп препаратов, которые будут эффективны для конкретных групп пациентов. Актуальными здесь также являются задачи продвижения лекарственных препаратов на рынок.

Молекулярная генетика и генная инженерия

В молекулярной генетике и генной инженерии выделяют отдельное направление Data Mining, которое имеет название анализ данных в микро-массивах (Microarray Data Analysis, MDA). Подробно с применением Microarray Data Analysis можно ознакомиться в [22].

Некоторые применения этого направления:

- ранняя и более точная диагностика;
- новые молекулярные цели для терапии;
- улучшенные и индивидуально подобранные виды лечения;
- фундаментальные биологические открытия.

Примеры использования Data Mining - молекулярный диагноз некоторых серьезнейших заболеваний; открытие того, что генетический код действительно может предсказывать вероятность заболевания; открытие некоторых новых лекарств и препаратов.

Основные понятия, которыми оперирует Data Mining в областях "Молекулярная генетика и генная инженерия" - маркеры, т.е. генетические коды, которые контролируют различные признаки живого организма.

На финансирование проектов с использованием Data Mining в рассматриваемых сферах выделяют значительные финансовые средства.

Химия

Технология Data Mining активно используется в исследованиях органической и неорганической химии. Одно из возможных применений Data Mining в этой сфере - выявление каких-либо специфических особенностей строения соединений, которые могут включать тысячи элементов.

Далее мы рассмотрим технологии, в основу которых также положено понятие Mining или "добыча".

Web Mining

Web Mining можно перевести как "добыча данных в Web". Web Intelligence или Web Интеллект готов "открыть новую главу" в стремительном развитии электронного бизнеса. Способность определять интересы и предпочтения каждого посетителя, наблюдая за его поведением, является серьезным и критичным преимуществом конкурентной борьбы на рынке электронной коммерции.

Системы Web Mining могут ответить на многие вопросы, например, кто из посетителей является потенциальным клиентом Web-магазина, какая группа клиентов Web-магазина приносит наибольший доход, каковы интересы определенного посетителя или группы посетителей.

Технология Web Mining охватывает методы, которые способны на основе данных сайта обнаружить новые, ранее неизвестные знания и которые в дальнейшем можно будет использовать на практике. Другими словами, технология Web Mining применяет технологию Data Mining для анализа неструктурированной, неоднородной, распределенной и значительной по объему информации, содержащейся на Web-узлах.

Согласно таксономии Web Mining [31], здесь можно выделить два основных направления: Web Content Mining и Web Usage Mining.

Web Content Mining подразумевает автоматический поиск и извлечение качественной информации из разнообразных источников Интернета, перегруженных "информационным шумом". Здесь также идет речь о различных средствах кластеризации и аннотировании документов.

В этом направлении, в свою очередь, выделяют два подхода: подход, основанный на агентах, и подход, основанный на базах данных.

Подход, основанный на агентах (Agent Based Approach), включает такие системы:

- интеллектуальные поисковые агенты (Intelligent Search Agents);
- фильтрация информации / классификация;
- персонифицированные агенты сети.

Примеры систем интеллектуальных агентов поиска:

- Harvest (Brown и др., 1994),
- FAQ-Finder (Hammond и др., 1995),
- Information Manifold (Kirk и др., 1995),
- OCCAM (Kwok and Weld, 1996), and ParaSite (Spertus, 1997),
- ILA (Information Learning Agent) (Perkowitz and Etzioni, 1995),
- ShopBot (Doorenbos и др., 1996).

Подход, основанный на базах данных (Database Approach), включает системы:

- многоуровневые базы данных;
- системы web-запросов (Web Query Systems);

Примеры систем web-запросов:

- W3QL (Konopnicki и Shmueli, 1995),
- WebLog (Lakshmanan и др., 1996),
- Lorel (Quass и др., 1995),
- UnQL (Buneman и др., 1995 and 1996),
- TSIMMIS (Chawathe и др.., 1994).

Второе направление Web Usage Mining подразумевает обнаружение закономерностей в действиях пользователя Web-узла или их группы.

Анализируется следующая информация:

- какие страницы просматривал пользователь;
- какова последовательность просмотра страниц.

Анализируется также, какие группы пользователей можно выделить среди общего их числа на основе истории просмотра Web-узла.

Web Usage Mining включает следующие составляющие:

- предварительная обработка;
- операционная идентификация;
- инструменты обнаружения шаблонов;
- инструменты анализа шаблонов.

При использовании Web Mining перед разработчиками возникает два типа задач. Первая касается сбора данных, вторая - использования методов персонификации. В результате сбора некоторого объема персонифицированных ретроспективных данных о конкретном клиенте, система накапливает определенные знания о нем и может рекомендовать ему, например, определенные наборы товаров или услуг. На основе информации о всех посетителях сайта Web-система может выявить определенные группы посетителей и также рекомендовать им товары или же предлагать товары в рассылках.

Задачи Web Mining согласно [31] можно подразделить на такие категории:

- Предварительная обработка данных для Web Mining.
- Обнаружение шаблонов и открытие знаний с использованием ассоциативных правил, временных последовательностей, классификации и кластеризации;
- Анализ полученного знания.

Text Mining

Text Mining охватывает новые методы для выполнения семантического анализа текстов, информационного поиска и управления. Синонимом понятия Text Mining является KDT (Knowledge Discovering in Text - поиск или обнаружение знаний в тексте).

В отличие от технологии Data Mining, которая предусматривает анализ упорядоченной в некие структуры информации, технология Text Mining анализирует большие и сверхбольшие массивы неструктурированной информации.

Программы, реализующие эту задачу, должны некоторым образом оперировать естественным человеческим языком и при этом понимать семантику анализируемого

текста. Один из методов, на котором основаны некоторые Text Mining системы, - поиск так называемой подстроки в строке.

Call Mining

По словам Энн Беднарц [32], "добыча звонков" может стать популярным инструментом корпоративных информационных систем.

Технология Call Mining объединяет в себя распознавание речи, ее анализ и Data Mining. Ее цель - упрощение поиска в аудио-архивах, содержащих записи переговоров между операторами и клиентами. При помощи этой технологии операторы могут обнаруживать недостатки в системе обслуживания клиентов, находить возможности увеличения продаж, а также выявлять тенденции в обращениях клиентов.

Среди разработчиков новой технологии Call Mining ("добыча" и анализ звонков) - компании CallMiner, Nexidia, ScanSoft, Witness Systems. В технологии Call Mining разработано два подхода - на основе преобразования речи в текст и на базе фонетического анализа.

Примером реализации первого подхода, основанного на преобразовании речи, является система CallMiner. В процессе Call Mining сначала используется система преобразования речи, затем следует ее анализ, в ходе которого в зависимости от содержания разговоров формируется статистика телефонных вызовов. Полученная информация хранится в базе данных, в которой возможен поиск, извлечение и обработка.

Пример реализации второго подхода - фонетического анализа - продукция компании Nexidia. При этом подходе речь разбивается на фонемы, являющиеся звуками или их сочетаниями. Такие элементы образуют распознаваемые фрагменты. При поиске определенных слов и их сочетаний система идентифицирует их с фонемами.

Аналитики отмечают, что за последние годы интерес к системам на основе Call Mining значительно возрос. Это объясняется тем фактом, что менеджеры высшего звена компаний, работающих в различных сферах, в т.ч. в области финансов, мобильной связи, авиабизнеса, не хотят тратить много времени на прослушивание звонков с целью обобщения информации или же выявления каких-либо фактов нарушений.

По словам Дэниэла Хонг, аналитика компании Datamonitor: "Использование этих технологий повышает оперативность и снижает стоимость обработки информации".

Типичная инсталляция продукции от разработчика Nexidia обходится в сумму от 100 до 300 тыс. долл. Стоимость внедрения системы CallMiner по преобразованию речи и набора аналитических приложений составляет около 450 тыс. долл.

По мнению Шоллера, приложения Audio Mining и Video Mining найдут со временем гораздо более широкое применение, например, при индексации учебных видеофильмов и презентаций в медиабиблиотеках компаний. Однако технологии Audio Mining и Video Mining находятся сейчас на уровне становления, а практическое их применение - на самой начальной стадии.

Основы анализа данных

В этой лекции мы рассмотрим некоторые аспекты статистического анализа данных, в частности, описательную статистику, корреляционный и регрессионный анализы. Статистический анализ включает большое разнообразие методов, даже для поверхностного знакомства с которыми объема одной лекции слишком мало. Цель данной лекции - дать самое общее представление о понятиях корреляции, регрессии, а также познакомиться с описательной статистикой. Примеры, рассмотренные в лекции, намеренно упрощены.

Существует большое разнообразие прикладных пакетов, реализующих широкий спектр статистических методов, их также называют универсальными пакетами или инструментальными наборами. О таких наборах мы подробно поговорим в последнем разделе курса. В Microsoft Excel также реализован широкий арсенал методов математической статистики, реализация примеров данной лекции продемонстрирована именно на этом программном обеспечении.

Следует заметить, что существует сложность использования статистических методов, так же как и статистического программного обеспечения, - для этого пользователю необходимы специальные знания.

Анализ данных в Microsoft Excel

Microsoft Excel имеет большое число статистических функций. Некоторые являются встроенными, некоторые доступны после установки пакета анализа. В данной лекции мы воспользуемся именно этим программным обеспечением.

Обращение к Пакету анализа. Средства, включенные в пакет анализа данных, доступны через команду **Анализ данных** меню **Сервис**. Если эта команда отсутствует в меню, в меню **Сервис/Надстройки** необходимо активировать пункт **"Пакет анализа"**.

Далее мы рассмотрим некоторые инструменты, включенные в Пакет анализа.

Описательная статистика

Описательная статистика (Descriptive statistics) - техника сбора и суммирования количественных данных, которая используется для превращения массы цифровых данных в форму, удобную для восприятия и обсуждения.

Цель описательной статистики - обобщить первичные результаты, полученные в результате наблюдений и экспериментов.

Пусть дан набор данных А, представленный в таблице 8.1.

Таблица 8.1. Набор данных А

x	y

3	9
2	7
4	12
5	15
6	17
7	19
8	21
9	23,4
10	25,6
11	27,8

Выбрав в меню **Сервис "Пакет анализа"** и выбрав инструмент анализа **"Описательная статистика"**, получаем одномерный статистический отчет, содержащий информацию о центральной тенденции и изменчивости или вариации входных данных.

В состав описательной статистики входят такие характеристики: среднее; стандартная ошибка; медиана; мода; стандартное отклонение; дисперсия выборки; эксцесс; асимметричность; интервал; минимум; максимум; сумма; счет.

Отчет "Описательная статистика" для двух переменных их набора данных А приведен в таблице 8.2.

Таблица 8.2. Описательная статистика для набора данных А

	x	y
Среднее	6,5	17,68
Стандартная ошибка	0,957427108	2,210922382
Медиана	6,5	18
Стандартное отклонение	3,027650354	6,991550456
Дисперсия выборки	9,166666667	48,88177778
Эксцесс	-1,2	-1,106006058
Асимметричность	0	-0,128299221
Интервал	9	20,8

Минимум	2	7
Максимум	11	27,8
Сумма	65	176,8
Счет	10	10
Наибольший (1)	11	27,8
Наименьший (1)	2	7
Уровень надежности (95,0%)	2,16585224	5,001457714

Рассмотрим, что же представляют собой характеристики описательной статистики.

Центральная тенденция

Измерение центральной тенденции заключается в выборе числа, которое наилучшим способом описывает все значения признака набора данных. Такое число имеет как свои достоинства, так и недостатки. Мы рассмотрим две характеристики этого измерения, а именно: среднее значение и медиану, эти понятия будут использоваться нами в последующих лекциях.

Главная цель среднего - представление набора данных для последующего анализа, сопоставления и сравнения.

Значение среднего легко вычисляется и может быть использовано для последующего анализа. Оно может быть вычислено для данных, измеряемых по интервальной шкале, и для некоторых данных, измеряемых по порядковой шкале. Среднее значение рассчитывается как среднее арифметическое набора данных: сумма всех значений выборки, деленная на объем выборки. "Сжимая" данные таким образом, мы теряем много информации.

Среднее значение очень информативно и позволяет делать вывод относительно всего исследуемого набора данных. При помощи среднего мы получаем возможность сравнивать несколько наборов данных или их частей.

При анализе данных средним не следует злоупотреблять, необходимо учитывать его свойства и ограничения. Известны характеристики "средняя температура по больнице" или "средняя высота дома", показывающие некорректность использования этой меры центральной тенденции для некоторых случаев.

Свойства среднего

- При расчете среднего не допускаются пропущенные значения данных.
- Среднее может вычисляться только для числовых данных и для дихотомических шкал.
- Для одного набора данных может быть рассчитано одно и только одно значение среднего.

Информативность среднего значения переменной высока, если известен ее доверительный интервал. Доверительным интервалом для среднего значения является интервал значений вокруг оценки, где с данным уровнем доверия находится "истинное" среднее популяции. Вычисление доверительных интервалов основывается на предположении нормальности наблюдаемых величин.

Ширина доверительного интервала зависит от размера выборки и от разброса данных.

С увеличением размера выборки точность оценки среднего возрастает. С увеличением разброса значений выборки надежность среднего падает. Если размер выборки достаточно большой, качество среднего увеличивается независимо от выполнения предположения нормальности выборки.

Медиана - точная середина выборки, которая делит ее на две равные части по числу наблюдений.

Обязательным условием нахождения медианы является упорядоченность выборки.

Таким образом, для нечетного количества наблюдений медианой выступает наблюдение с номером (n+1)/2, где n - количество наблюдений в выборке.

Для четного числа наблюдений медианой является среднее значение наблюдений n/2 и (n+2)/2.

Некоторые свойства медианы

- Для одного набора данных может быть рассчитано одно и только одно значение медианы.
- Медиана может быть рассчитана для неполного набора данных, для этого необходимо знать номера наблюдений по порядку, общее количество наблюдений и несколько значений в середине набора данных.

Характеристики вариации данных

Наиболее простыми характеристиками выборки являются максимум и минимум.

Минимум - наименьшее значение выборки.

Максимум - наибольшее значение выборки.

Размах - разница между наибольшим и наименьшим значениями выборки.

Дисперсия - среднее арифметическое квадратов отклонений значений от их среднего.

Стандартное отклонение - квадратный корень из дисперсии выборки - мера того, насколько широко разбросаны точки данных относительно их среднего.

Эксцесс показывает "остроту пика" распределения, характеризует относительную остроконечность или сглаженность распределения по сравнению с нормальным распределением. Положительный эксцесс обозначает относительно остроконечное распределение (пик заострен). Отрицательный эксцесс обозначает относительно сглаженное распределение (пик закруглен).

Если эксцесс существенно отличается от нуля, то распределение имеет или более закругленный пик, чем нормальное, или, напротив, имеет более острый пик (возможно, имеется несколько пиков). Эксцесс нормального распределения равен нулю.

Асимметрия или асимметричность показывает отклонение распределения от симметричного. Если асимметрия существенно отличается от нуля, то распределение несимметрично, нормальное распределение абсолютно симметрично. Если распределение имеет длинный правый хвост, асимметрия положительна; если длинный левый хвост - отрицательна.

Выбросы (outliers) - данные, резко отличающиеся от основного числа данных.

При обнаружении выбросов перед исследователем стоит дилемма: оставить наблюдения-выбросы либо от них отказаться. Второй вариант требует серьезной аргументации и описания. Полезным будет провести анализ данных с выбросами и без и сравнить результаты.

Следует помнить, что при применении классических методов статистического анализа, которые, как правило, не являются робастными (устойчивыми), наличие выбросов в наборе данных приводит к некорректным результатам. Если набор данных относительно мал, исключение данных, которые считаются выбросами, может заметно повлиять на результаты анализа.

Наличие выбросов в наборе данных может быть связано с появлением так называемых "сдвинутых" значений, связанных с систематической ошибкой, ошибок ввода, ошибок сбора данных и т.д. Иногда к выбросам могут относиться наименьшие и наибольшие значения набора данных.

Корреляционный анализ

Корреляционный анализ применяется для количественной оценки взаимосвязи двух наборов данных, представленных в безразмерном виде. Корреляционный анализ дает возможность установить, ассоциированы ли наборы данных по величине. Коэффициент корреляции, всегда обозначаемый латинской буквой r, используется для определения наличия взаимосвязи между двумя свойствами.

Связь между признаками (по шкале Чеддока) может быть сильной, средней и слабой. Тесноту связи определяют по величине коэффициента корреляции, который может принимать значения от -1 до +1 включительно. Критерии оценки тесноты связи показаны на рис. 8.1.

Величина коэффициента корреляции	0.1 - 0.3	0.3 - 0.5	0.5 – 0.7	0.7 – 0.9	0.9 – 1.0
Характеристика силы связи	слабая	умеренная	заметная	высокая	весьма высокая

средняя сильная

Рис. 8.1. Количественные критерии оценки тесноты связи

Коэффициент корреляции Пирсона

Коэффициент корреляции Пирсона r, который является безразмерным индексом в интервале от -1,0 до 1,0 включительно, отражает степень линейной зависимости между двумя множествами данных.

Показатель тесноты связи между двумя признаками определяется по формуле линейного коэффициента корреляции:

$$r_{xy} = \frac{n\sum xy - \sum x \sum y}{\sqrt{[n\sum x^2 - (\sum x)^2]\cdot[n\sum y^2 - (\sum y)^2]}}$$

где x - значение факторного признака;

y - значение результативного признака;

n - число пар данных.

Парная корреляция - это связь между двумя признаками: результативным и факторным или двумя факторными.

Варианты связи, характеризующие наличие или отсутствие линейной связи между признаками:

- большие значения из одного набора данных связаны с большими значениями другого набора (положительная корреляция) - наличие прямой линейной связи;
- малые значения одного набора связаны с большими значениями другого (отрицательная корреляция) - наличие отрицательной линейной связи;
- данные двух диапазонов никак не связаны (нулевая корреляция) - отсутствие линейной связи.

В качестве примера возьмем набор данных А (таблица 8.1). Необходимо определить наличие линейной связи между признаками x и y.

Для графического представления связи двух переменных использована система координат с осями, соответствующими переменным x и y. Построенный график, называемый диаграммой рассеивания, показан на рис. 8.2. Данная диаграмма показывает, что низкие значения переменной x соответствуют низким значениям переменной y, высокие значения переменной x соответствуют высоким значениям переменной y. Этот пример демонстрирует наличие явной связи.

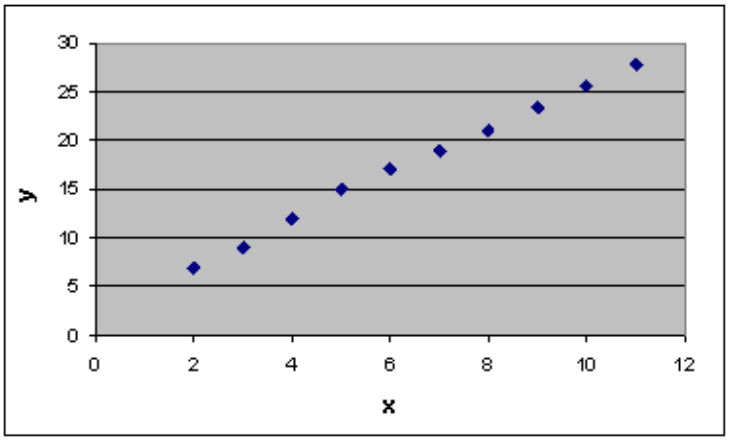

Рис. 8.2. Диаграмма рассеивания

Таким образом, мы можем установить зависимость между переменными x и y. Рассчитаем коэффициент корреляции Пирсона между двумя массивами (x и y) при помощи функции MS Excel ПИРСОН(массив1;массив2). В результате получаем значение коэффициент корреляции равный 0,998364, т.е. связь между переменными x и y является весьма высокой. Используя пакет анализа MS Excel и инструмент анализа "Корреляция", можем построить корреляционную матрицу.

Любая зависимость между переменными обладает двумя важными свойствами: величиной и надежностью. Чем сильнее зависимость между двумя переменными, тем больше величина зависимости и тем легче предсказать значение одной переменной по значению другой переменной. Величину зависимости легче измерить, чем надежность.

Надежность зависимости не менее важна, чем ее величина. Это свойство связано с представительностью исследуемой выборки. Надежность зависимости характеризует, насколько вероятно, что эта зависимость будет снова найдена на других данных.

С ростом величины зависимости переменных ее надежность обычно возрастает.

Регрессионный анализ

Основная особенность регрессионного анализа: при его помощи можно получить конкретные сведения о том, какую форму и характер имеет зависимость между исследуемыми переменными.

Последовательность этапов регрессионного анализа

Рассмотрим кратко этапы регрессионного анализа.

1. Формулировка задачи. На этом этапе формируются предварительные гипотезы о зависимости исследуемых явлений.
2. Определение зависимых и независимых (объясняющих) переменных.
3. Сбор статистических данных. Данные должны быть собраны для каждой из переменных, включенных в регрессионную модель.
4. Формулировка гипотезы о форме связи (простая или множественная, линейная или нелинейная).

5. Определение функции регрессии (заключается в расчете численных значений параметров уравнения регрессии)
6. Оценка точности регрессионного анализа.
7. Интерпретация полученных результатов. Полученные результаты регрессионного анализа сравниваются с предварительными гипотезами. Оценивается корректность и правдоподобие полученных результатов.
8. Предсказание неизвестных значений зависимой переменной.

При помощи регрессионного анализа возможно решение задачи прогнозирования и классификации. Прогнозные значения вычисляются путем подстановки в уравнение регрессии параметров значений объясняющих переменных. Решение задачи классификации осуществляется таким образом: линия регрессии делит все множество объектов на два класса, и та часть множества, где значение функции больше нуля, принадлежит к одному классу, а та, где оно меньше нуля, - к другому классу.

Задачи регрессионного анализа

Рассмотрим основные задачи регрессионного анализа: установление формы зависимости, определение функции регрессии, оценка неизвестных значений зависимой переменной.

Установление формы зависимости.

Характер и форма зависимости между переменными могут образовывать следующие разновидности регрессии:

- положительная линейная регрессия (выражается в равномерном росте функции);
- положительная равноускоренно возрастающая регрессия;
- положительная равнозамедленно возрастающая регрессия;
- отрицательная линейная регрессия (выражается в равномерном падении функции);
- отрицательная равноускоренно убывающая регрессия;
- отрицательная равнозамедленно убывающая регрессия.

Однако описанные разновидности обычно встречаются не в чистом виде, а в сочетании друг с другом. В таком случае говорят о комбинированных формах регрессии.

Определение функции регрессии.

Вторая задача сводится к выяснению действия на зависимую переменную главных факторов или причин, при неизменных прочих равных условиях, и при условии исключения воздействия на зависимую переменную случайных элементов. Функция регрессии определяется в виде математического уравнения того или иного типа.

Оценка неизвестных значений зависимой переменной.

Решение этой задачи сводится к решению задачи одного из типов:

- Оценка значений зависимой переменной внутри рассматриваемого интервала исходных данных, т.е. пропущенных значений; при этом решается задача интерполяции.
- Оценка будущих значений зависимой переменной, т.е. нахождение значений вне заданного интервала исходных данных; при этом решается задача экстраполяции.

Обе задачи решаются путем подстановки в уравнение регрессии найденных оценок параметров значений независимых переменных. Результат решения уравнения представляет собой оценку значения целевой (зависимой) переменной.

Рассмотрим некоторые предположения, на которые опирается регрессионный анализ.

Предположение линейности, т.е. предполагается, что связь между рассматриваемыми переменными является линейной. Так, в рассматриваемом примере мы построили диаграмму рассеивания и смогли увидеть явную линейную связь. Если же на диаграмме рассеивания переменных мы видим явное отсутствие линейной связи, т.е. присутствует нелинейная связь, следует использовать нелинейные методы анализа.

Предположение о нормальности остатков. Оно допускает, что распределение разницы предсказанных и наблюдаемых значений является нормальным. Для визуального определения характера распределения можно воспользоваться гистограммами остатков.

При использовании регрессионного анализа следует учитывать его основное ограничение. Оно состоит в том, что регрессионный анализ позволяет обнаружить лишь зависимости, а не связи, лежащие в основе этих зависимостей.

Регрессионный анализ дает возможность оценить степень связи между переменными путем вычисления предполагаемого значения переменной на основании нескольких известных значений.

Уравнение регрессии.

Уравнение регрессии выглядит следующим образом: Y=a+b*X

При помощи этого уравнения переменная Y выражается через константу a и угол наклона прямой (или угловой коэффициент) b, умноженный на значение переменной X. Константу a также называют свободным членом, а угловой коэффициент - коэффициентом регрессии или B-коэффициентом.

В большинстве случав (если не всегда) наблюдается определенный разброс наблюдений относительно регрессионной прямой.

Остаток - это отклонение отдельной точки (наблюдения) от линии регрессии (предсказанного значения).

Для решения задачи регрессионного анализа в MS Excel выбираем в меню **Сервис "Пакет анализа"** и инструмент анализа "Регрессия". Задаем входные интервалы X и Y. Входной интервал Y - это диапазон зависимых анализируемых данных, он должен включать один столбец. Входной интервал X - это диапазон независимых данных, которые необходимо проанализировать. Число входных диапазонов должно быть не больше 16.

На выходе процедуры в выходном диапазоне получаем отчет, приведенный в таблице 8.3а - 8.3в.

ВЫВОД ИТОГОВ

Таблица 8.3а. Регрессионная статистика	
Регрессионная статистика	
Множественный R	0,998364
R-квадрат	0,99673
Нормированный R-квадрат	0,996321
Стандартная ошибка	0,42405
Наблюдения	10

Сначала рассмотрим верхнюю часть расчетов, представленную в таблице 8.3а, - регрессионную статистику.

Величина R-квадрат, называемая также мерой определенности, характеризует качество полученной регрессионной прямой. Это качество выражается степенью соответствия между исходными данными и регрессионной моделью (расчетными данными). Мера определенности всегда находится в пределах интервала [0;1].

В большинстве случаев значение R-квадрат находится между этими значениями, называемыми экстремальными, т.е. между нулем и единицей.

Если значение R-квадрата близко к единице, это означает, что построенная модель объясняет почти всю изменчивость соответствующих переменных. И наоборот, значение R-квадрата, близкое к нулю, означает плохое качество построенной модели.

В нашем примере мера определенности равна 0,99673, что говорит об очень хорошей подгонке регрессионной прямой к исходным данным.

множественный R - коэффициент множественной корреляции R - выражает степень зависимости независимых переменных (X) и зависимой переменной (Y).

Множественный R равен квадратному корню из коэффициента детерминации, эта величина принимает значения в интервале от нуля до единицы.

В простом линейном регрессионном анализе множественный R равен коэффициенту корреляции Пирсона. Действительно, множественный R в нашем случае равен коэффициенту корреляции Пирсона из предыдущего примера (0,998364).

Таблица 8.3б. Коэффициенты регрессии			
	Коэффициенты	Стандартная ошибка	t-статистика
Y-пересечение	2,694545455	0,33176878	8,121757129
Переменная X 1	2,305454545	0,04668634	49,38177965
* Приведен усеченный вариант расчетов			

Теперь рассмотрим среднюю часть расчетов, представленную в таблице 8.3б. Здесь даны коэффициент регрессии b (2,305454545) и смещение по оси ординат, т.е. константа a (2,694545455).

Исходя из расчетов, можем записать уравнение регрессии таким образом:

```
Y= x*2,305454545+2,694545455
```

Направление связи между переменными определяется на основании знаков (отрицательный или положительный) коэффициентов регрессии (коэффициента b).

Если знак при коэффициенте регрессии - положительный, связь зависимой переменной с независимой будет положительной. В нашем случае знак коэффициента регрессии положительный, следовательно, связь также является положительной.

Если знак при коэффициенте регрессии - отрицательный, связь зависимой переменной с независимой является отрицательной (обратной).

В таблице 8.3в. представлены результаты вывода остатков. Для того чтобы эти результаты появились в отчете, необходимо при запуске инструмента "Регрессия" активировать чекбокс "Остатки".

ВЫВОД ОСТАТКА

Таблица 8.3в. Остатки

Наблюдение	Предсказанное Y	Остатки	Стандартные остатки
1	9,610909091	-0,610909091	-1,528044662
2	7,305454545	-0,305454545	-0,764022331
3	11,91636364	0,083636364	0,209196591
4	14,22181818	0,778181818	1,946437843
5	16,52727273	0,472727273	1,182415512
6	18,83272727	0,167272727	0,418393181
7	21,13818182	-0,138181818	-0,34562915
8	23,44363636	-0,043636364	-0,109146047
9	25,74909091	-0,149090909	-0,372915662
10	28,05454545	-0,254545455	-0,636685276

При помощи этой части отчета мы можем видеть отклонения каждой точки от построенной линии регрессии. Наибольшее абсолютное значение остатка в нашем случае - 0,778, наименьшее - 0,043. Для лучшей интерпретации этих данных воспользуемся графиком исходных данных и построенной линией регрессии, представленными на рис. 8.3. Как видим, линия регрессии достаточно точно "подогнана" под значения исходных данных.

Следует учитывать, что рассматриваемый пример является достаточно простым и далеко не всегда возможно качественное построение регрессионной прямой линейного вида.

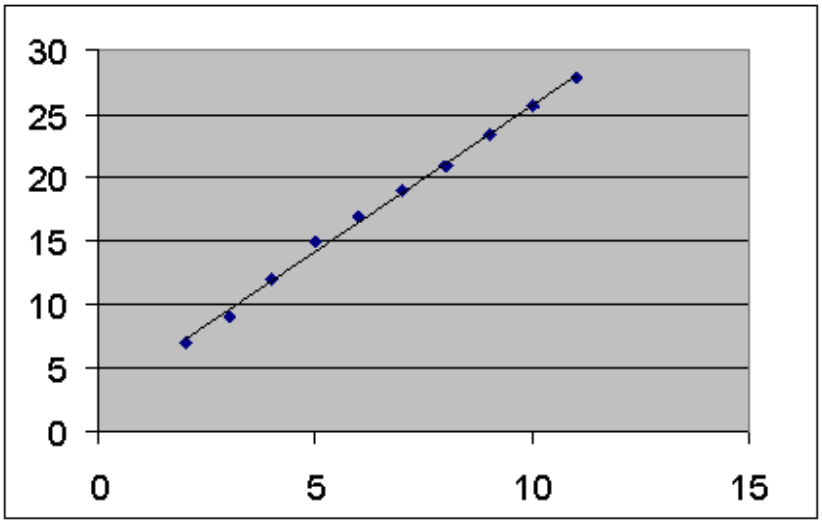

Рис. 8.3. Исходные данные и линия регрессии

Осталась нерассмотренной задача оценки неизвестных будущих значений зависимой переменной на основании известных значений независимой переменной, т.е. задача прогнозирования.

Имея уравнение регрессии, задача прогнозирования сводится к решению уравнения Y= x*2,305454545+2,694545455 с известными значениями x. Результаты прогнозирования зависимой переменной Y на шесть шагов вперед представлены в таблице 8.4.

Таблица 8.4. Результаты прогнозирования переменной Y

x	Y(прогнозируемое)
11	28,05455
12	30,36
13	32,66545
14	34,97091
15	37,27636
16	39,58182

Таким образом, в результате использования регрессионного анализа в пакете Microsoft Excel мы:

- построили уравнение регрессии;
- установили форму зависимости и направление связи между переменными - положительная линейная регрессия, которая выражается в равномерном росте функции;
- установили направление связи между переменными;
- оценили качество полученной регрессионной прямой;
- смогли увидеть отклонения расчетных данных от данных исходного набора;
- предсказали будущие значения зависимой переменной.

Если функция регрессии определена, интерпретирована и обоснована, и оценка точности регрессионного анализа соответствует требованиям, можно считать, что построенная модель и прогнозные значения обладают достаточной надежностью.

Прогнозные значения, полученные таким способом, являются средними значениями, которые можно ожидать.

Выводы

В этой части лекции мы рассмотрели основные характеристики описательной статистики и среди них такие понятия, как среднее значение, медиана, максимум, минимум и другие характеристики вариации данных. Также было кратко рассмотрено понятие выбросов. Рассмотренные в лекции характеристики относятся к так называемому исследовательскому анализу данных, его выводы могут относиться не к генеральной совокупности, а лишь к выборке данных. Исследовательский анализ данных используется для получения первичных выводов и формирования гипотез относительно генеральной совокупности. Также были рассмотрены основы корреляционного и регрессионного анализа, их задачи и возможности практического использования.

Методы классификации и прогнозирования. Деревья решений

Метод деревьев решений (decision trees) является одним из наиболее популярных методов решения задач классификации и прогнозирования. Иногда этот метод Data Mining также называют деревьями решающих правил, деревьями классификации и регрессии.

Как видно из последнего названия, при помощи данного метода решаются задачи классификации и прогнозирования.

Если зависимая, т.е. целевая переменная принимает дискретные значения, при помощи метода дерева решений решается задача классификации.

Если же зависимая переменная принимает непрерывные значения, то дерево решений устанавливает зависимость этой переменной от независимых переменных, т.е. решает задачу численного прогнозирования.

Впервые деревья решений были предложены Ховилендом и Хантом (Hoveland, Hunt) в конце 50-х годов прошлого века. Самая ранняя и известная работа Ханта и др., в которой излагается суть деревьев решений - "Эксперименты в индукции" ("Experiments in Induction") - была опубликована в 1966 году.

В наиболее простом виде дерево решений - это способ представления правил в иерархической, последовательной структуре. Основа такой структуры - ответы "Да" или "Нет" на ряд вопросов.

На рис. 9.1 приведен пример дерева решений, задача которого - ответить на вопрос: "Играть ли в гольф?" Чтобы решить задачу, т.е. принять решение, играть ли в гольф, следует отнести текущую ситуацию к одному из известных классов (в данном случае - "играть" или "не играть"). Для этого требуется ответить на ряд вопросов, которые находятся в узлах этого дерева, начиная с его корня.

Первый узел нашего дерева "Солнечно?" является узлом проверки, т.е. условием. При положительном ответе на вопрос осуществляется переход к левой части дерева, называемой левой ветвью, при отрицательном - к правой части дерева. Таким образом, внутренний узел дерева является узлом проверки определенного условия. Далее идет следующий вопрос и т.д., пока не будет достигнут конечный узел дерева, являющийся узлом решения. Для нашего дерева существует два типа конечного узла: "играть" и "не играть" в гольф.

В результате прохождения от корня дерева (иногда называемого корневой вершиной) до его вершины решается задача классификации, т.е. выбирается один из классов - "играть" и "не играть" в гольф.

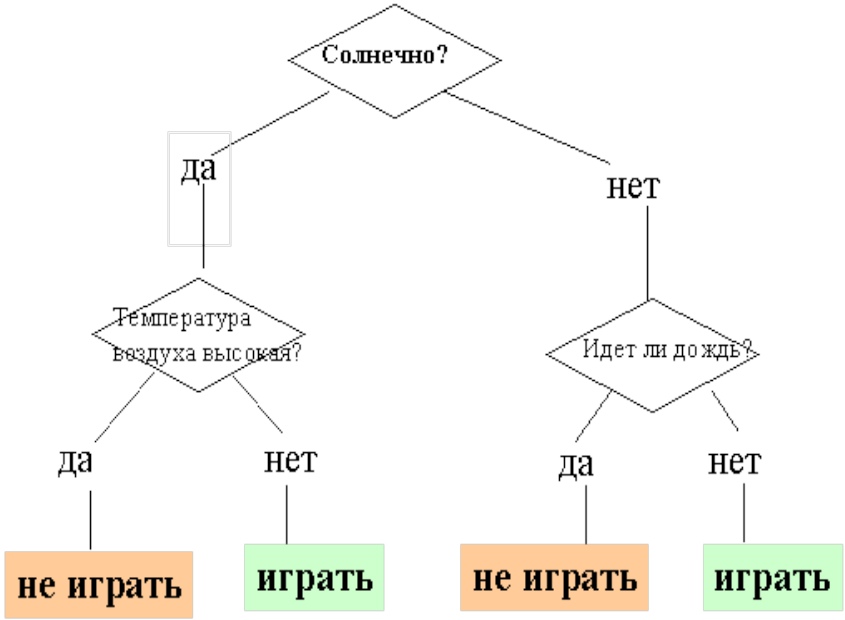

Рис. 9.1. Дерево решений "Играть ли в гольф?"

Целью построения дерева решения в нашем случае является определение значения категориальной зависимой переменной.

Итак, для нашей задачи основными элементами дерева решений являются:

Корень дерева: "Солнечно?"

Внутренний узел дерева или узел проверки: "Температура воздуха высокая?", "Идет ли дождь?"

Лист, конечный узел дерева, узел решения или вершина: "Играть", "Не играть"

Ветвь дерева (случаи ответа): "Да", "Нет".

В рассмотренном примере решается задача бинарной классификации, т.е. создается дихотомическая классификационная модель. Пример демонстрирует работу так называемых бинарных деревьев.

В узлах бинарных деревьев ветвление может вестись только в двух направлениях, т.е. существует возможность только двух ответов на поставленный вопрос ("да" и "нет").

Бинарные деревья являются самым простым, частным случаем деревьев решений. В остальных случаях, ответов и, соответственно, ветвей дерева, выходящих из его внутреннего узла, может быть больше двух.

Рассмотрим более сложный пример. База данных, на основе которой должно осуществляться прогнозирование, содержит следующие ретроспективные данные о клиентах банка, являющиеся ее атрибутами: возраст, наличие недвижимости, образование, среднемесячный доход, вернул ли клиент вовремя кредит. Задача состоит в том, чтобы на

основании перечисленных выше данных (кроме последнего атрибута) определить, стоит ли выдавать кредит новому клиенту.

Как мы уже рассматривали в лекции, посвященной задаче классификации, такая задача решается в два этапа: построение классификационной модели и ее использование.

На этапе построения модели, собственно, и строится дерево классификации или создается набор неких правил. На этапе использования модели построенное дерево, или путь от его корня к одной из вершин, являющийся набором правил для конкретного клиента, используется для ответа на поставленный вопрос "Выдавать ли кредит?"

Правилом является логическая конструкция, представленная в виде "если : то :".

На рис. 9.2. приведен пример дерева классификации, с помощью которого решается задача "Выдавать ли кредит клиенту?". Она является типичной задачей классификации, и при помощи деревьев решений получают достаточно хорошие варианты ее решения.

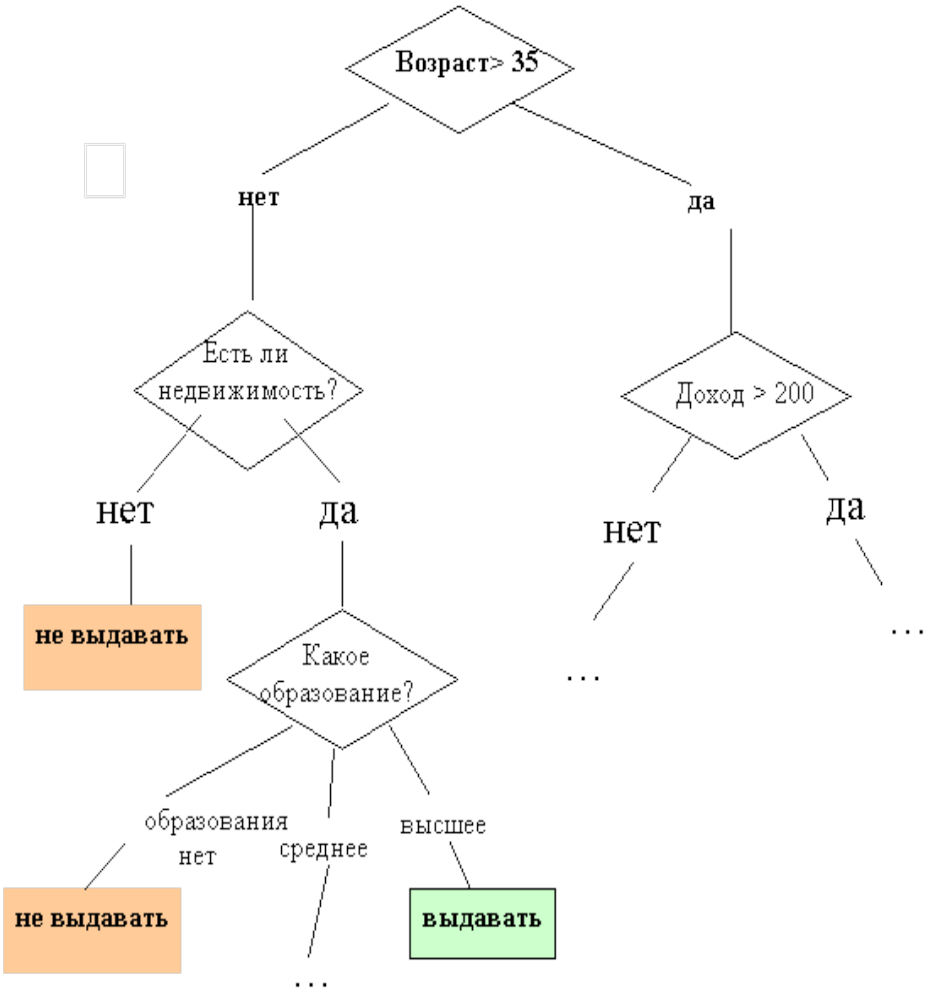

Рис. 9.2. Дерево решений "Выдавать ли кредит?"

Как мы видим, внутренние узлы дерева (возраст, наличие недвижимости, доход и образование) являются атрибутами описанной выше базы данных. Эти атрибуты

называют прогнозирующими, или атрибутами расщепления (splitting attribute). Конечные узлы дерева, или листы, именуются метками класса, являющимися значениями зависимой категориальной переменной "выдавать" или "не выдавать" кредит.

Каждая ветвь дерева, идущая от внутреннего узла, отмечена предикатом расщепления. Последний может относиться лишь к одному атрибуту расщепления данного узла. Характерная особенность предикатов расщепления: каждая запись использует уникальный путь от корня дерева только к одному узлу-решению. Объединенная информация об атрибутах расщепления и предикатах расщепления в узле называется критерием расщепления (splitting criterion) [33].

На рис. 9.2. изображено одно из возможных деревьев решений для рассматриваемой базы данных. Например, критерий расщепления "Какое образование?", мог бы иметь два предиката расщепления и выглядеть иначе: образование "высшее" и "не высшее". Тогда дерево решений имело бы другой вид.

Таким образом, для данной задачи (как и для любой другой) может быть построено множество деревьев решений различного качества, с различной прогнозирующей точностью.

Качество построенного дерева решения весьма зависит от правильного выбора критерия расщепления. Над разработкой и усовершенствованием критериев работают многие исследователи.

Метод деревьев решений часто называют "наивным" подходом [34]. Но благодаря целому ряду преимуществ, данный метод является одним из наиболее популярных для решения задач классификации.

Преимущества деревьев решений

Интуитивность деревьев решений. Классификационная модель, представленная в виде дерева решений, является интуитивной и упрощает понимание решаемой задачи. Результат работы алгоритмов конструирования деревьев решений, в отличие, например, от нейронных сетей, представляющих собой "черные ящики", легко интерпретируется пользователем. Это свойство деревьев решений не только важно при отнесении к определенному классу нового объекта, но и полезно при интерпретации модели классификации в целом. Дерево решений позволяет понять и объяснить, почему конкретный объект относится к тому или иному классу.

Деревья решений дают возможность извлекать правила из базы данных на **естественном языке**. Пример правила: Если Возраст > 35 и Доход > 200, то выдать кредит.

Деревья решений позволяют создавать классификационные модели в тех областях, где аналитику достаточно сложно формализовать знания.

Алгоритм конструирования дерева решений **не требует от пользователя выбора входных атрибутов** (независимых переменных). На вход алгоритма можно подавать все существующие атрибуты, алгоритм сам выберет наиболее значимые среди них, и только они будут использованы для построения дерева. В сравнении, например, с нейронными сетями, это значительно облегчает пользователю работу, поскольку в нейронных сетях выбор количества входных атрибутов существенно влияет на время обучения.

Точность моделей, созданных при помощи деревьев решений, сопоставима с другими методами построения классификационных моделей (статистические методы, нейронные сети).

Разработан ряд **масштабируемых алгоритмов**, которые могут быть использованы для построения деревьев решения на сверхбольших базах данных; масштабируемость здесь означает, что с ростом числа примеров или записей базы данных время, затрачиваемое на обучение, т.е. построение деревьев решений, растет линейно. Примеры таких алгоритмов: SLIQ, SPRINT.

Быстрый процесс обучения. На построение классификационных моделей при помощи алгоритмов конструирования деревьев решений требуется значительно меньше времени, чем, например, на обучение нейронных сетей.

Большинство алгоритмов конструирования деревьев решений имеют возможность специальной обработки **пропущенных значений**.

Многие классические статистические методы, при помощи которых решаются задачи классификации, могут работать только с числовыми данными, в то время как деревья решений работают и с числовыми, и с **категориальными** типами данных.

Многие статистические методы являются параметрическими, и пользователь должен заранее владеть определенной информацией, например, знать вид модели, иметь гипотезу о виде зависимости между переменными, предполагать, какой вид распределения имеют данные. Деревья решений, в отличие от таких методов, строят непараметрические модели. Таким образом, деревья решений способны решать такие задачи Data Mining, в которых отсутствует априорная информация о виде зависимости между исследуемыми данными.

Процесс конструирования дерева решений

Напомним, что рассматриваемая нами задача классификации относится к стратегии обучения с учителем, иногда называемого индуктивным обучением. В этих случаях все объекты тренировочного набора данных заранее отнесены к одному из предопределенных классов.

Алгоритмы конструирования деревьев решений состоят из этапов "построение" или "создание" дерева (tree building) и "сокращение" дерева (tree pruning). В ходе создания дерева решаются вопросы выбора критерия расщепления и остановки обучения (если это предусмотрено алгоритмом). В ходе этапа сокращения дерева решается вопрос отсечения некоторых его ветвей.

Рассмотрим эти вопросы подробней.

Критерий расщепления

Процесс создания дерева происходит сверху вниз, т.е. является нисходящим. В ходе процесса алгоритм должен найти такой критерий расщепления, иногда также называемый критерием разбиения, чтобы разбить множество на подмножества, которые бы ассоциировались с данным узлом проверки. Каждый узел проверки должен быть помечен определенным атрибутом. Существует правило выбора атрибута: он должен разбивать исходное множество данных таким образом, чтобы объекты подмножеств, получаемых в

результате этого разбиения, являлись представителями одного класса или же были максимально приближены к такому разбиению. Последняя фраза означает, что количество объектов из других классов, так называемых "примесей", в каждом классе должно стремиться к минимуму.

Существуют различные критерии расщепления. Наиболее известные - мера энтропии и индекс Gini.

В некоторых методах для выбора атрибута расщепления используется так называемая **мера информативности** подпространств атрибутов, которая основывается на энтропийном подходе и известна под названием "мера информационного выигрыша" (information gain measure) или мера энтропии.

Другой критерий расщепления, предложенный Брейманом (Breiman) и др., реализован в алгоритме CART и называется **индексом Gini**. При помощи этого индекса атрибут выбирается на основании расстояний между распределениями классов.

Если дано множество T, включающее примеры из n классов, индекс Gini, т.е. gini(T), определяется по формуле:

$$gini(T) = 1 - \sum_{j=1}^{n} p_j^2$$

где T - текущий узел, pj - вероятность класса j в узле T, n - количество классов.

Большое дерево не означает, что оно "подходящее"

Чем больше частных случаев описано в дереве решений, тем меньшее количество объектов попадает в каждый частный случай. Такие деревья называют "ветвистыми" или "кустистыми", они состоят из неоправданно большого числа узлов и ветвей, исходное множество разбивается на большое число подмножеств, состоящих из очень малого числа объектов. В результате "переполнения" таких деревьев их способность к обобщению уменьшается, и построенные модели не могут давать верные ответы.

В процессе построения дерева, чтобы его размеры не стали чрезмерно большими, используют специальные процедуры, которые позволяют создавать оптимальные деревья, так называемые деревья "подходящих размеров" (Breiman,1984).

Какой размер дерева может считаться оптимальным? Дерево должно быть достаточно сложным, чтобы учитывать информацию из исследуемого набора данных, но одновременно оно должно быть достаточно простым [39]. Другими словами, дерево должно использовать информацию, улучшающую качество модели, и игнорировать ту информацию, которая ее не улучшает.

Тут существует две возможные стратегии. Первая состоит в наращивании дерева до определенного размера в соответствии с параметрами, заданными пользователем. Определение этих параметров может основываться на опыте и интуиции аналитика, а

также на некоторых "диагностических сообщениях" системы, конструирующей дерево решений.

Вторая стратегия состоит в использовании набора процедур, определяющих "подходящий размер" дерева, они разработаны Бриманом, Куилендом и др. в 1984 году. Однако, как отмечают авторы, нельзя сказать, что эти процедуры доступны начинающему пользователю.

Процедуры, которые используют для предотвращения создания чрезмерно больших деревьев, включают: сокращение дерева путем отсечения ветвей; использование правил остановки обучения.

Следует отметить, что не все алгоритмы при конструировании дерева работают по одной схеме. Некоторые алгоритмы включают два отдельных последовательных этапа: построение дерева и его сокращение; другие чередуют эти этапы в процессе своей работы для предотвращения наращивания внутренних узлов.

Остановка построения дерева

Рассмотрим правило остановки. Оно должно определить, является ли рассматриваемый узел внутренним узлом, при этом он будет разбиваться дальше, или же он является конечным узлом, т.е. узлом решением.

Остановка - такой момент в процессе построения дерева, когда следует прекратить дальнейшие ветвления.

Один из вариантов правил остановки - "ранняя остановка" (prepruning), она определяет целесообразность разбиения узла. Преимущество использования такого варианта - уменьшение времени на обучение модели. Однако здесь возникает риск снижения точности классификации. Поэтому рекомендуется "вместо остановки использовать отсечение" (Breiman, 1984).

Второй вариант остановки обучения - ограничение глубины дерева. В этом случае построение заканчивается, если достигнута заданная глубина.

Еще один вариант остановки - задание минимального количества примеров, которые будут содержаться в конечных узлах дерева. При этом варианте ветвления продолжаются до того момента, пока все конечные узлы дерева не будут чистыми или будут содержать не более чем заданное число объектов.

Существует еще ряд правил, но следует отметить, что ни одно из них не имеет большой практической ценности, а некоторые применимы лишь в отдельных случаях [35].

Сокращение дерева или отсечение ветвей

Решением проблемы слишком ветвистого дерева является его сокращение путем отсечения (pruning) некоторых ветвей.

Качество классификационной модели, построенной при помощи дерева решений, характеризуется двумя основными признаками: точностью распознавания и ошибкой.

Точность распознавания рассчитывается как отношение объектов, правильно классифицированных в процессе обучения, к общему количеству объектов набора данных, которые принимали участие в обучении.

Ошибка рассчитывается как отношение объектов, неправильно классифицированных в процессе обучения, к общему количеству объектов набора данных, которые принимали участие в обучении.

Отсечение ветвей или замену некоторых ветвей поддеревом следует проводить там, где эта процедура не приводит к возрастанию ошибки. Процесс проходит снизу вверх, т.е. является восходящим. Это более популярная процедура, чем использование правил остановки. Деревья, получаемые после отсечения некоторых ветвей, называют усеченными.

Если такое усеченное дерево все еще не является интуитивным и сложно для понимания, используют извлечение правил, которые объединяют в наборы для описания классов. Каждый путь от корня дерева до его вершины или листа дает одно правило. Условиями правила являются проверки на внутренних узлах дерева.

Алгоритмы

На сегодняшний день существует большое число алгоритмов, реализующих деревья решений: CART, C4.5, CHAID, CN2, NewId, ITrule и другие.

Алгоритм CART

Алгоритм CART (Classification and Regression Tree), как видно из названия, решает задачи классификации и регрессии. Он разработан в 1974-1984 годах четырьмя профессорами статистики - Leo Breiman (Berkeley), Jerry Friedman (Stanford), Charles Stone (Berkeley) и Richard Olshen (Stanford).

Атрибуты набора данных могут иметь как дискретное, так и числовое значение.

Алгоритм CART предназначен для построения бинарного дерева решений. Бинарные деревья также называют двоичными. Пример такого дерева рассматривался в начале лекции.

Другие особенности алгоритма CART:

- функция оценки качества разбиения;
- механизм отсечения дерева;
- алгоритм обработки пропущенных значений;
- построение деревьев регрессии.

Каждый узел бинарного дерева при разбиении имеет только двух потомков, называемых дочерними ветвями. Дальнейшее разделение ветви зависит от того, много ли исходных данных описывает данная ветвь. На каждом шаге построения дерева правило, формируемое в узле, делит заданное множество примеров на две части. Правая его часть (ветвь right) - это та часть множества, в которой правило выполняется; левая (ветвь left) - та, для которой правило не выполняется.

Функция оценки качества разбиения, которая используется для выбора оптимального правила, - индекс Gini - был описан выше. Отметим, что данная оценочная функция основана на идее уменьшения неопределенности в узле. Допустим, есть узел, и он разбит на два класса. Максимальная неопределенность в узле будет достигнута при разбиении его на два подмножества по 50 примеров, а максимальная определенность - при разбиении на 100 и 0 примеров.

Правила разбиения. Напомним, что алгоритм CART работает с числовыми и категориальными атрибутами. В каждом узле разбиение может идти только по одному атрибуту. Если атрибут является числовым, то во внутреннем узле формируется правило вида xi <= c, Значение c в большинстве случаев выбирается как среднее арифметическое двух соседних упорядоченных значений переменной xi обучающего набора данных. Если же атрибут относится к категориальному типу, то во внутреннем узле формируется правило xi V(xi), где V(xi) - некоторое непустое подмножество множества значений переменной xi в обучающем наборе данных.

Механизм отсечения. Этим механизмом, имеющим название minimal cost-complexity tree pruning, алгоритм CART принципиально отличается от других алгоритмов конструирования деревьев решений. В рассматриваемом алгоритме отсечение - это некий компромисс между получением дерева "подходящего размера" и получением наиболее точной оценки классификации. Метод заключается в получении последовательности уменьшающихся деревьев, но деревья рассматриваются не все, а только "лучшие представители".

Перекрестная проверка (V-fold cross-validation) является наиболее сложной и одновременно оригинальной частью алгоритма CART. Она представляет собой путь выбора окончательного дерева, при условии, что набор данных имеет небольшой объем или же записи набора данных настолько специфические, что разделить набор на обучающую и тестовую выборку не представляется возможным.

Итак, основные характеристики алгоритма CART: бинарное расщепление, критерий расщепления - индекс Gini, алгоритмы minimal cost-complexity tree pruning и V-fold cross-validation, принцип "вырастить дерево, а затем сократить", высокая скорость построения, обработка пропущенных значений.

Алгоритм C4.5

Алгоритм C4.5 строит дерево решений с неограниченным количеством ветвей у узла. Данный алгоритм может работать только с дискретным зависимым атрибутом и поэтому может решать только задачи классификации. C4.5 считается одним из самых известных и широко используемых алгоритмов построения деревьев классификации.

Для работы алгоритма C4.5 необходимо соблюдение следующих требований:

- Каждая запись набора данных должна быть ассоциирована с одним из предопределенных классов, т.е. один из атрибутов набора данных должен являться меткой класса.
- Классы должны быть дискретными. Каждый пример должен однозначно относиться к одному из классов.
- Количество классов должно быть значительно меньше количества записей в исследуемом наборе данных.

Последняя версия алгоритма - алгоритм C4.8 - реализована в инструменте Weka как J4.8 (Java). Коммерческая реализация метода: C5.0, разработчик RuleQuest, Австралия.

Алгоритм C4.5 медленно работает на сверхбольших и зашумленных наборах данных.

Мы рассмотрели два известных алгоритма построения деревьев решений CART и C4.5. Оба алгоритма являются робастными, т.е. устойчивыми к шумам и выбросам данных.

Алгоритмы построения деревьев решений различаются следующими характеристиками:

- вид расщепления - бинарное (binary), множественное (multi-way)
- критерии расщепления - энтропия, Gini, другие
- возможность обработки пропущенных значений
- процедура сокращения ветвей или отсечения
- возможности извлечения правил из деревьев.

Ни один алгоритм построения дерева нельзя априори считать наилучшим или совершенным, подтверждение целесообразности использования конкретного алгоритма должно быть проверено и подтверждено экспериментом.

Разработка новых масштабируемых алгоритмов

Наиболее серьезное требование, которое сейчас предъявляется к алгоритмам конструирования деревьев решений - это масштабируемость, т.е. алгоритм должен обладать масштабируемым методом доступа к данным.

Разработан ряд новых масштабируемых алгоритмов, среди них - алгоритм Sprint, предложенный Джоном Шафером и его коллегами [36]. Sprint, являющийся масштабируемым вариантом рассмотренного в лекции алгоритма CART, предъявляет минимальные требования к объему оперативной памяти.

Выводы

В лекции мы рассмотрели метод деревьев решений; определить его кратко можно как иерархическое, гибкое средство предсказания принадлежности объектов к определенному классу или прогнозирования значений числовых переменных.

Качество работы рассмотренного метода деревьев решений зависит как от выбора алгоритма, так и от набора исследуемых данных. Несмотря на все преимущества данного метода, следует помнить, что для того, чтобы построить качественную модель, необходимо понимать природу взаимосвязи между зависимыми и независимыми переменными и подготовить достаточный набор данных.

Методы классификации и прогнозирования. Метод опорных векторов. Метод "ближайшего соседа". Байесовская классификация

В предыдущих лекциях мы рассмотрели такие методы классификации и прогнозирования как линейная регрессия и деревья решений; в этой лекции мы продолжим знакомство с методами этой группы и рассмотрим следующие из них: метод опорных векторов, метод ближайшего соседа (метод рассуждений на основе прецедентов) и байесовскую классификацию.

Метод опорных векторов

Метод опорных векторов (Support Vector Machine - SVM) относится к группе граничных методов. Она определяет классы при помощи границ областей.

При помощи данного метода решаются задачи бинарной классификации.

В основе метода лежит понятие плоскостей решений.

Плоскость (plane) решения разделяет объекты с разной классовой принадлежностью.

На рис.10.1 приведен пример, в котором участвуют объекты двух типов. Разделяющая линия задает границу, справа от которой - все объекты типа brown (коричневый), а слева - типа yellow (желтый). Новый объект, попадающий направо, классифицируется как объект класса brown или - как объект класса yellow, если он расположился по левую сторону от разделяющей прямой. В этом случае каждый объект характеризуется двумя измерениями.

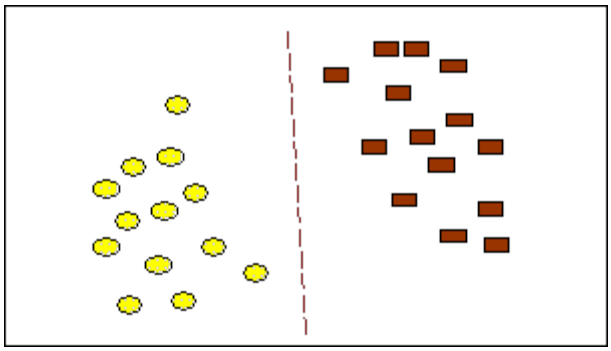

Рис. 10.1. Разделение классов прямой линией

Цель метода опорных векторов - найти плоскость, разделяющую два множества объектов; такая плоскость показана на рис. 10.2. На этом рисунке множество образцов поделено на два класса: желтые объекты принадлежат классу A, коричневые - классу B.

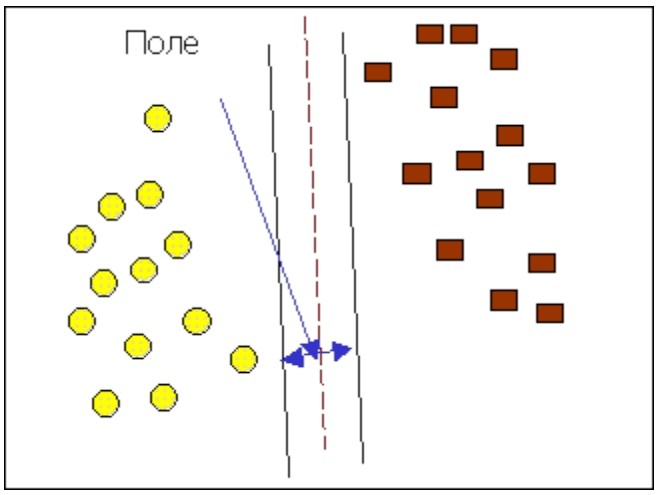

Рис. 10.2. К определению опорных векторов

Метод отыскивает образцы, находящиеся на границах между двумя классами, т.е. опорные вектора; они изображены на рис. 10.3.

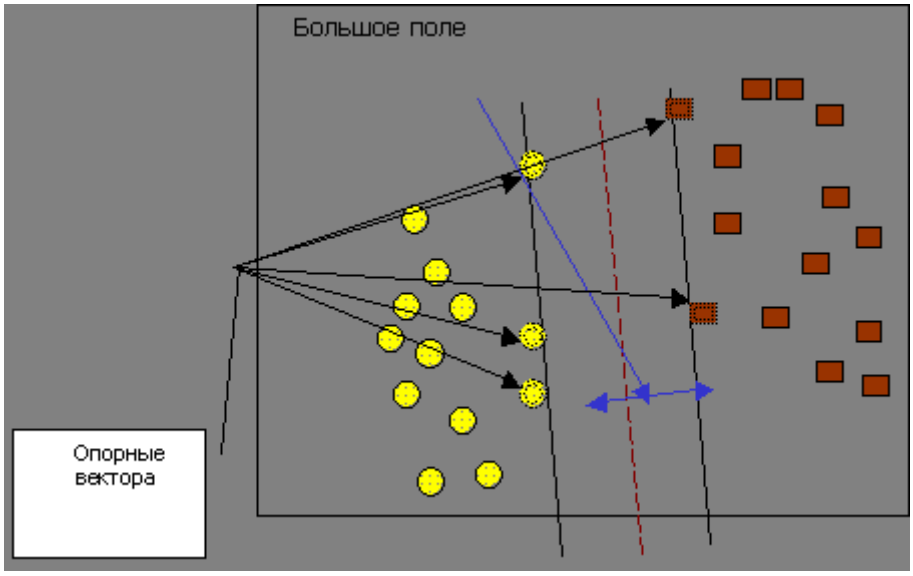

Рис. 10.3. Опорные векторы

Опорными векторами называются объекты множества, лежащие на границах областей.

Классификация считается хорошей, если область между границами пуста.

На рис. 10.3.показано пять векторов, которые являются опорными для данного множества.

Линейный SVM

Решение задачи бинарной классификации при помощи метода опорных векторов заключается в поиске некоторой линейной функции, которая правильно разделяет набор данных на два класса. Рассмотрим задачу классификации, где число классов равно двум.

Задачу можно сформулировать как поиск функции f(x), принимающей значения меньше нуля для векторов одного класса и больше нуля - для векторов другого класса. В качестве исходных данных для решения поставленной задачи, т.е. поиска классифицирующей функции f(x), дан тренировочный набор векторов пространства, для которых известна их принадлежность к одному из классов. Семейство классифицирующих функций можно описать через функцию f(x). Гиперплоскость определена вектором a и значением b, т.е. f(x)=ax+b. Решение данной задачи проиллюстрировано на рис. 10.4.

В результате решения задачи, т.е. построения SVM-модели, найдена функция, принимающая значения меньше нуля для векторов одного класса и больше нуля - для векторов другого класса. Для каждого нового объекта отрицательное или положительное значение определяет принадлежность объекта к одному из классов.

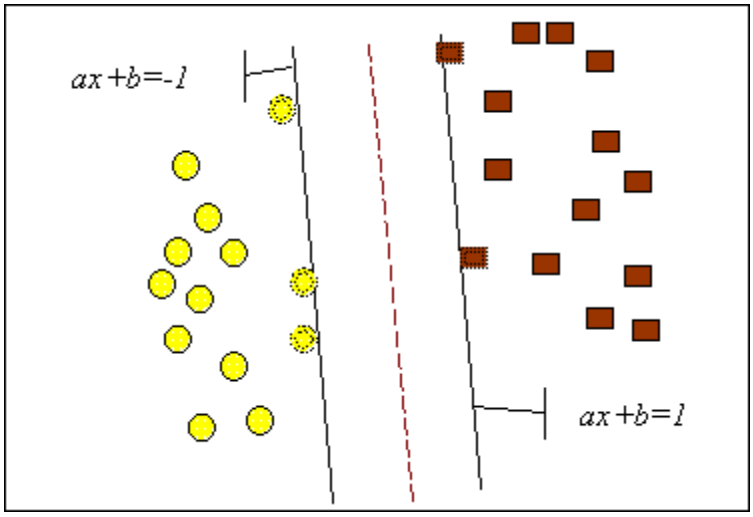

Рис. 10.4. Линейный SVM

Наилучшей функцией классификации является функция, для которой ожидаемый риск минимален. Понятие ожидаемого риска в данном случае означает ожидаемый уровень ошибки классификации.

Напрямую оценить ожидаемый уровень ошибки построенной модели невозможно, это можно сделать при помощи понятия эмпирического риска. Однако следует учитывать, что минимизация последнего не всегда приводит к минимизации ожидаемого риска. Это обстоятельство следует помнить при работе с относительно небольшими наборами тренировочных данных.

Эмпирический риск - уровень ошибки классификации на тренировочном наборе.

Таким образом, в результате решения задачи методом опорных векторов для линейно разделяемых данных мы получаем функцию классификации, которая минимизирует верхнюю оценку ожидаемого риска.

Одной из проблем, связанных с решением задач классификации рассматриваемым методом, является то обстоятельство, что не всегда можно легко найти линейную границу между двумя классами.

В таких случаях один из вариантов - увеличение размерности, т.е. перенос данных из плоскости в трехмерное пространство, где возможно построить такую плоскость, которая идеально разделит множество образцов на два класса. Опорными векторами в этом случае будут служить объекты из обоих классов, являющиеся экстремальными.

Таким образом, при помощи добавления так называемого оператора ядра и дополнительных размерностей, находятся границы между классами в виде гиперплоскостей.

Однако следует помнить: сложность построения SVM-модели заключается в том, что чем выше размерность пространства, тем сложнее с ним работать. Один из вариантов работы с данными высокой размерности - это предварительное применение какого-либо метода понижения размерности данных для выявления наиболее существенных компонент, а затем использование метода опорных векторов.

Как и любой другой метод, метод SVM имеет свои сильные и слабые стороны, которые следует учитывать при выборе данного метода.

Недостаток метода состоит в том, что для классификации используется не все множество образцов, а лишь их небольшая часть, которая находится на границах.

Достоинство метода состоит в том, что для классификации методом опорных векторов, в отличие от большинства других методов, достаточно небольшого набора данных. При правильной работе модели, построенной на тестовом множестве, вполне возможно применение данного метода на реальных данных.

Метод опорных векторов позволяет [37, 38]:

- получить функцию классификации с минимальной верхней оценкой ожидаемого риска (уровня ошибки классификации);
- использовать линейный классификатор для работы с нелинейно разделяемыми данными, сочетая простоту с эффективностью.

Метод "ближайшего соседа" или системы рассуждений на основе аналогичных случаев

Следует сразу отметить, что метод "ближайшего соседа" ("nearest neighbour") относится к классу методов, работа которых основывается на хранении данных в памяти для сравнения с новыми элементами. При появлении новой записи для прогнозирования находятся отклонения между этой записью и подобными наборами данных, и наиболее подобная (или ближний сосед) идентифицируется.

Например, при рассмотрении нового клиента банка, его атрибуты сравниваются со всеми существующими клиентами данного банка (доход, возраст и т.д.). Множество "ближайших соседей" потенциального клиента банка выбирается на основании ближайшего значения дохода, возраста и т.д.

При таком подходе используется термин "k-ближайший сосед" ("k-nearest neighbour"). Термин означает, что выбирается k "верхних" (ближайших) соседей для их рассмотрения в качестве множества "ближайших соседей". Поскольку не всегда удобно хранить все данные, иногда хранится только множество "типичных" случаев. В таком случае

используемый метод называют рассуждением по аналогии (Case Based Reasoning, CBR), рассуждением на основе аналогичных случаев, рассуждением по прецедентам.

Прецедент - это описание ситуации в сочетании с подробным указанием действий, предпринимаемых в данной ситуации.

Подход, основанный на прецедентах, условно можно поделить на следующие этапы:

- сбор подробной информации о поставленной задаче;
- сопоставление этой информации с деталями прецедентов, хранящихся в базе, для выявления аналогичных случаев;
- выбор прецедента, наиболее близкого к текущей проблеме, из базы прецедентов;
- адаптация выбранного решения к текущей проблеме, если это необходимо;
- проверка корректности каждого вновь полученного решения;
- занесение детальной информации о новом прецеденте в базу прецедентов.

Таким образом, вывод, основанный на прецедентах, представляет собой такой метод анализа данных, который делает заключения относительно данной ситуации по результатам поиска аналогий, хранящихся в базе прецедентов.

Данный метод по своей сути относится к категории "обучение без учителя", т.е. является "самообучающейся" технологией, благодаря чему рабочие характеристики каждой базы прецедентов с течением времени и накоплением примеров улучшаются. Разработка баз прецедентов по конкретной предметной области происходит на естественном для человека языке, следовательно, может быть выполнена наиболее опытными сотрудниками компании - экспертами или аналитиками, работающими в данной предметной области.

Однако это не означает, что CBR-системы самостоятельно могут принимать решения. Последнее всегда остается за человеком, данный метод лишь предлагает возможные варианты решения и указывает на самый "разумный" с ее точки зрения.

Преимущества метода

- Простота использования полученных результатов.
- Решения не уникальны для конкретной ситуации, возможно их использование для других случаев.
- Целью поиска является не гарантированно верное решение, а лучшее из возможных.

Недостатки метода "ближайшего соседа"

- Данный метод не создает каких-либо моделей или правил, обобщающих предыдущий опыт, - в выборе решения они основываются на всем массиве доступных исторических данных, поэтому невозможно сказать, на каком основании строятся ответы.
- Существует сложность выбора меры "близости" (метрики). От этой меры главным образом зависит объем множества записей, которые нужно хранить в памяти для достижения удовлетворительной классификации или прогноза. Также существует высокая зависимость результатов классификации от выбранной метрики.
- При использовании метода возникает необходимость полного перебора обучающей выборки при распознавании, следствие этого - вычислительная трудоемкость.
- Типичные задачи данного метода - это задачи небольшой размерности по количеству классов и переменных.

С помощью данного метода решаются задачи классификации и регрессии.

Рассмотрим подробно принципы работы метода k-ближайших соседей для решения задач классификации и регрессии (прогнозирования).

Решение задачи классификации новых объектов

Эта задача схематично изображена на рис. 10.5. Примеры (известные экземпляры) отмечены знаком "+" или "-", определяющим принадлежность к соответствующему классу ("+" или "-"), а новый объект, который требуется классифицировать, обозначен красным кружочком. Новые объекты также называют точками запроса.

Наша цель заключается в оценке (классификации) отклика точек запроса с использованием специально выбранного числа их ближайших соседей. Другими словами, мы хотим узнать, к какому классу следует отнести точку запроса: как знак "+" или как знак "-".

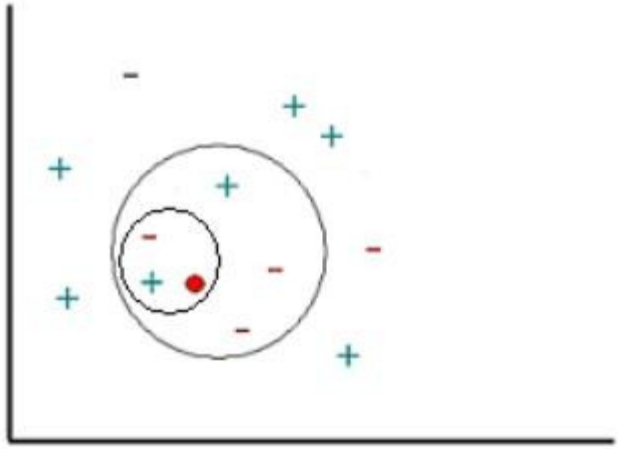

Рис. 10.5. Классификация объектов множества при разном значении параметра k

Для начала рассмотрим результат работы метода k-ближайших соседей с использованием одного ближайшего соседа. В этом случае отклик точки запроса будет классифицирован как знак плюс, так как ближайшая соседняя точка имеет знак плюс.

Теперь увеличим число используемых ближайших соседей до двух. На этот раз метод k-ближайших соседей не сможет классифицировать отклик точки запроса, поскольку вторая ближайшая точка имеет знак минус и оба знака равноценны (т.е. победа с одинаковым количеством голосов).

Далее увеличим число используемых ближайших соседей до 5. Таким образом, будет определена целая окрестность точки запроса (на графике ее граница отмечена красной(серой) окружностью). Так как в области содержится 2 точки со знаком "+" и 3 точки со знаком "-" , алгоритм k-ближайших соседей присвоит знак "-" отклику точки запроса.

Решение задачи прогнозирования

Далее рассмотрим принцип работы метода k-ближайших соседей для решения задачи регрессии. Регрессионные задачи связаны с прогнозированием значения зависимой переменной по значениям независимых переменных набора данных.

Рассмотрим график, показанный на рис. 10.6. Изображенный на ней набор точек (зеленые прямоугольники) получен по связи между независимой переменной x и зависимой переменной y (кривая красного цвета). Задан набор зеленых объектов (т.е. набор примеров); мы используем метод k-ближайших соседей для предсказания выхода точки запроса X по данному набору примеров (зеленые прямоугольники).

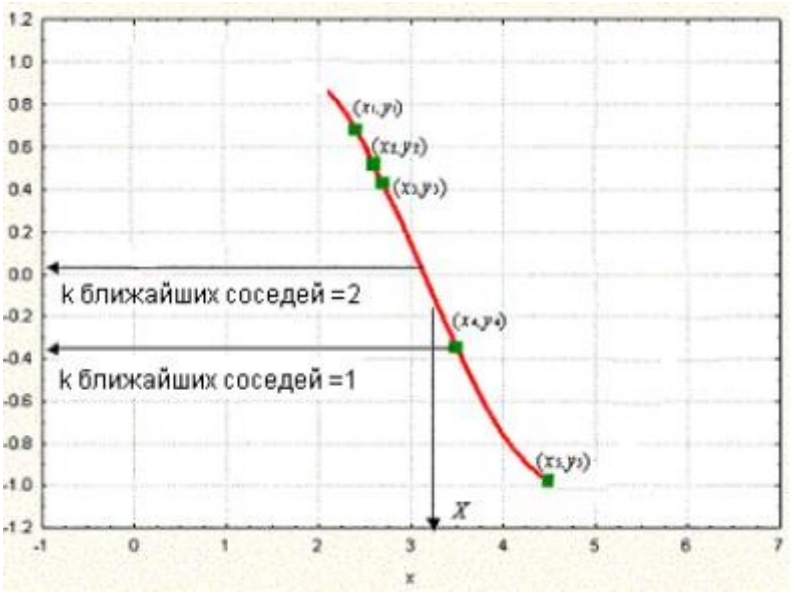

Рис. 10.6. Решение задачи прогнозирования при разных значениях параметра k

Сначала рассмотрим в качестве примера метод k-ближайших соседей с использованием одного ближайшего соседа, т.е. при k, равном единице. Мы ищем набор примеров (зеленые прямоугольники) и выделяем из их числа ближайший к точке запроса X. Для нашего случая ближайший пример - точка (x_4 ;y_4). Выход x_4 (т.е. y_4), таким образом, принимается в качестве результата предсказания выхода X (т.е. Y). Следовательно, для одного ближайшего соседа можем записать: выход Y равен y_4 ($Y = y_4$).

Далее рассмотрим ситуацию, когда k равно двум, т.е. рассмотрим двух ближайших соседей. В этом случае мы выделяем уже две ближайшие к X точки. На нашем графике это точки y_3 и y_4 соответственно. Вычислив среднее их выходов, записываем решение для Y в виде $Y = (y_3 + y_4)/2$.

Решение задачи прогнозирования осуществляется путем переноса описанных выше действий на использование произвольного числа ближайших соседей таким образом, что выход Y точки запроса X вычисляется как среднеарифметическое значение выходов k-ближайших соседей точки запроса.

Независимые и зависимые переменные набора данных могут быть как непрерывными, так и категориальными. Для непрерывных зависимых переменных задача рассматривается как задача прогнозирования, для дискретных переменных - как задача классификации.

Предсказание в задаче прогнозирования получается усреднением выходов k-ближайших соседей, а решение задачи классификации основано на принципе "по большинству голосов".

Критическим моментом в использовании метода k-ближайших соседей является выбор параметра k. Он один из наиболее важных факторов, определяющих качество прогнозной либо классификационной модели.

Если выбрано слишком маленькое значение параметра k, возникает вероятность большого разброса значений прогноза. Если выбранное значение слишком велико, это может привести к сильной смещенности модели. Таким образом, мы видим, что должно быть выбрано оптимальное значение параметра k. То есть это значение должно быть настолько большим, чтобы свести к минимуму вероятность неверной классификации, и одновременно, достаточно малым, чтобы k соседей были расположены достаточно близко к точке запроса.

Таким образом, мы рассматриваем k как сглаживающий параметр, для которого должен быть найден компромисс между силой размаха (разброса) модели и ее смещенностью.

Оценка параметра k методом кросс-проверки

Один из вариантов оценки параметра k - проведение кросс-проверки (Bishop, 1995).

Такая процедура реализована, например, в пакете STATISTICA (StatSoft) [39].

Кросс-проверка - известный метод получения оценок неизвестных параметров модели. Основная идея метода - разделение выборки данных на v "складок". V "складки" здесь суть случайным образом выделенные изолированные подвыборки.

По фиксированному значению k строится модель k-ближайших соседей для получения предсказаний на v-м сегменте (остальные сегменты при этом используются как примеры) и оценивается ошибка классификации. Для регрессионных задач наиболее часто в качестве оценки ошибки выступает сумма квадратов, а для классификационных задач удобней рассматривать точность (процент корректно классифицированных наблюдений).

Далее процесс последовательно повторяется для всех возможных вариантов выбора v. По исчерпании v "складок" (циклов), вычисленные ошибки усредняются и используются в качестве меры устойчивости модели (т.е. меры качества предсказания в точках запроса). Вышеописанные действия повторяются для различных k, и значение, соответствующее наименьшей ошибке (или наибольшей классификационной точности), принимается как оптимальное (оптимальное в смысле метода кросс-проверки).

Следует учитывать, что кросс-проверка - вычислительно емкая процедура, и необходимо предоставить время для работы алгоритма, особенно если объем выборки достаточно велик.

Второй вариант выбора значения параметра k - самостоятельно задать его значение. Однако этот способ следует использовать, если имеются обоснованные предположения относительно возможного значения параметра, например, предыдущие исследования сходных наборов данных.

Метод k-ближайших соседей показывает достаточно неплохие результаты в самых разнообразных задачах.

Примером реального использования описанного выше метода является программное обеспечение центра технической поддержки компании Dell, разработанное компанией Inference. Эта система помогает сотрудникам центра отвечать на большее число запросов, сразу предлагая ответы на распространенные вопросы и позволяя обращаться к базе во время разговора по телефону с пользователем. Сотрудники центра технической поддержки, благодаря реализации этого метода, могут отвечать одновременно на значительное число звонков. Программное обеспечение CBR сейчас развернуто в сети Intranet компании Dell.

Инструментов Data Mining, реализующих метод k-ближайших соседей и CBR-метод, не слишком много. Среди наиболее известных: CBR Express и Case Point (Inference Corp.), Apriori (Answer Systems), DP Umbrella (VYCOR Corp.), KATE tools (Acknosoft, Франция), Pattern Recognition Workbench (Unica, США), а также некоторые статистические пакеты, например, Statistica.

Байесовская классификация

Альтернативные названия: байесовское моделирование, байесовская статистика, метод байесовских сетей.

Ознакомиться детально с байесовской классификацией можно в [11]. Изначально байесовская классификация использовалась для формализации знаний экспертов в экспертных системах [40], сейчас баесовская классификация также применяется в качестве одного из методов Data Mining.

Так называемая наивная классификация или наивно-байесовский подход (naive-bayes approach) [43] является наиболее простым вариантом метода, использующего байесовские сети. При этом подходе решаются задачи классификации, результатом работы метода являются так называемые "прозрачные" модели.

"Наивная" классификация - достаточно прозрачный и понятный метод классификации. "Наивной" она называется потому, что исходит из предположения о взаимной независимости признаков.

Свойства наивной классификации:

1. Использование всех переменных и определение всех зависимостей между ними.
2. Наличие двух предположений относительно переменных:
    - все переменные являются одинаково важными;
    - все переменные являются статистически независимыми, т.е. значение одной переменной ничего не говорит о значении другой.

Большинство других методов классификации предполагают, что перед началом классификации вероятность того, что объект принадлежит тому или иному классу, одинакова; но это не всегда верно.

Допустим, известно, что определенный процент данных принадлежит конкретному классу. Возникает вопрос, можем ли мы использовать эту информацию при построении модели классификации? Существует множество реальных примеров использования этих априорных знаний, помогающих классифицировать объекты. Типичный пример из медицинской практики. Если доктор отправляет результаты анализов пациента на дополнительное исследование, он относит пациента к какому-то определенному классу. Каким образом можно применить эту информацию? Мы можем использовать ее в качестве дополнительных данных при построении классификационной модели.

Отмечают такие достоинства байесовских сетей как метода Data Mining [41]:

- в модели определяются зависимости между всеми переменными, это позволяет легко обрабатывать ситуации, в которых значения некоторых переменных неизвестны;
- байесовские сети достаточно просто интерпретируются и позволяют на этапе прогностического моделирования легко проводить анализ по сценарию "что, если";
- байесовский метод позволяет естественным образом совмещать закономерности, выведенные из данных, и, например, экспертные знания, полученные в явном виде;
- использование байесовских сетей позволяет избежать проблемы переучивания (overfitting), то есть избыточного усложнения модели, что является слабой стороной многих методов (например, деревьев решений и нейронных сетей).

Наивно-байесовский подход имеет следующие недостатки:

- перемножать условные вероятности корректно только тогда, когда все входные переменные действительно статистически независимы; хотя часто данный метод показывает достаточно хорошие результаты при несоблюдении условия статистической независимости, но теоретически такая ситуация должна обрабатываться более сложными методами, основанными на обучении байесовских сетей [42];
- невозможна непосредственная обработка непрерывных переменных - требуется их преобразование к интервальной шкале, чтобы атрибуты были дискретными; однако такие преобразования иногда могут приводить к потере значимых закономерностей [43];
- на результат классификации в наивно-байесовском подходе влияют только индивидуальные значения входных переменных, комбинированное влияние пар или троек значений разных атрибутов здесь не учитывается [43]. Это могло бы улучшить качество классификационной модели с точки зрения ее прогнозирующей точности, однако,увеличило бы количество проверяемых вариантов.

Байесовская классификация нашла широкое применение на практике.

Байесовская фильтрация по словам

Не так давно баесовская классификация была предложена для персональной фильтрации спама. Первый фильтр был разработан Полем Грахемом (Paul Graham). Для работы алгоритма требуется выполнение двух требований.

Первое требование - необходимо, чтобы у классифицируемого объекта присутствовало достаточное количество признаков. Этому идеально удовлетворяют все слова писем пользователя, за исключением совсем коротких и очень редко встречающихся.

Второе требование - постоянное переобучение и пополнение набора "спам - не спам". Такие условия очень хорошо работают в локальных почтовых клиентах, так как поток "не спама" у конечного клиента достаточно постоянен, а если изменяется, то не быстро.

Однако для всех клиентов сервера точно определить поток "не спама" довольно сложно, поскольку одно и то же письмо, являющееся для одного клиента спамом, для другого спамом не является. Словарь получается слишком большим, не существует четкого разделения на спам и "не спам", в результате качество классификации, в данном случае решение задачи фильтрации писем, значительно снижается.

Методы классификации и прогнозирования. Нейронные сети

Идея нейронных сетей родилась в рамках теории искусственного интеллекта, в результате попыток имитировать способность биологических нервных систем обучаться и исправлять ошибки.

Нейронные сети (Neural Networks) - это модели биологических нейронных сетей мозга, в которых нейроны имитируются относительно простыми, часто однотипными, элементами (искусственными нейронами).

Нейронная сеть может быть представлена направленным графом с взвешенными связями, в котором искусственные нейроны являются вершинами, а синаптические связи - дугами.

Нейронные сети широко используются для решения разнообразных задач.

Среди областей применения нейронных сетей - автоматизация процессов распознавания образов, прогнозирование, адаптивное управление, создание экспертных систем, организация ассоциативной памяти, обработка аналоговых и цифровых сигналов, синтез и идентификация электронных цепей и систем.

С помощью нейронных сетей можно, например, предсказывать объемы продаж изделий, показатели биржевого рынка, выполнять распознавание сигналов, конструировать самообучающиеся системы.

Модели нейронных сетей могут быть программного и аппаратного исполнения. Мы будем рассматривать сети первого типа.

Если говорить простым языком, слоистая нейронная сеть представляет собой совокупность нейронов, которые составляют слои. В каждом слое нейроны между собой никак не связаны, но связаны с нейронами предыдущего и следующего слоев. Информация поступает с первого на второй слой, со второго - на третий и т.д.

Среди задач Data Mining, решаемых с помощью нейронных сетей, будем рассматривать такие:

- Классификация (обучение с учителем). Примеры задач классификации: распознавание текста, распознавание речи, идентификация личности.
- Прогнозирование. Для нейронной сети задача прогнозирования может быть поставлена таким образом: найти наилучшее приближение функции, заданной конечным набором входных значений (обучающих примеров). Например, нейронные сети позволяют решать задачу восстановления пропущенных значений.
- Кластеризация (обучение без учителя). Примером задачи кластеризации может быть задача сжатия информации путем уменьшения размерности данных. Задачи кластеризации решаются, например, самоорганизующимися картами Кохонена. Этим сетям будет посвящена отдельная лекция.

Рассмотрим три примера задач, для решения которых возможно применение нейронных сетей.

Медицинская диагностика. В ходе наблюдения за различными показателями состояния пациентов накоплена база данных. Риск наступления осложнений может соответствовать сложной нелинейной комбинации наблюдаемых переменных, которая обнаруживается с помощью нейросетевого моделирования.

Прогнозирование показателей деятельности фирмы (объемы продаж). На основе ретроспективной информации о деятельности организации возможно определение объемов продаж на будущие периоды.

Предоставление кредита. Используя базу данных о клиентах банка, применяя нейронные сети, можно установить группу клиентов, которые относятся к группе потенциальных "неплательщиков".

Элементы нейронных сетей

Искусственный нейрон (формальный нейрон) - элемент искусственных нейронных сетей, моделирующий некоторые функции биологического нейрона.

Главная функция искусственного нейрона - формировать выходной сигнал в зависимости от сигналов, поступающих на его входы.

В самой распространенной конфигурации входные сигналы обрабатываются адаптивным сумматором, затем выходной сигнал сумматора поступает в нелинейный преобразователь, где преобразуется функцией активации, и результат подается на выход (в точку ветвления).

Общий вид искусственного нейрона приведен на рис. 11.1.

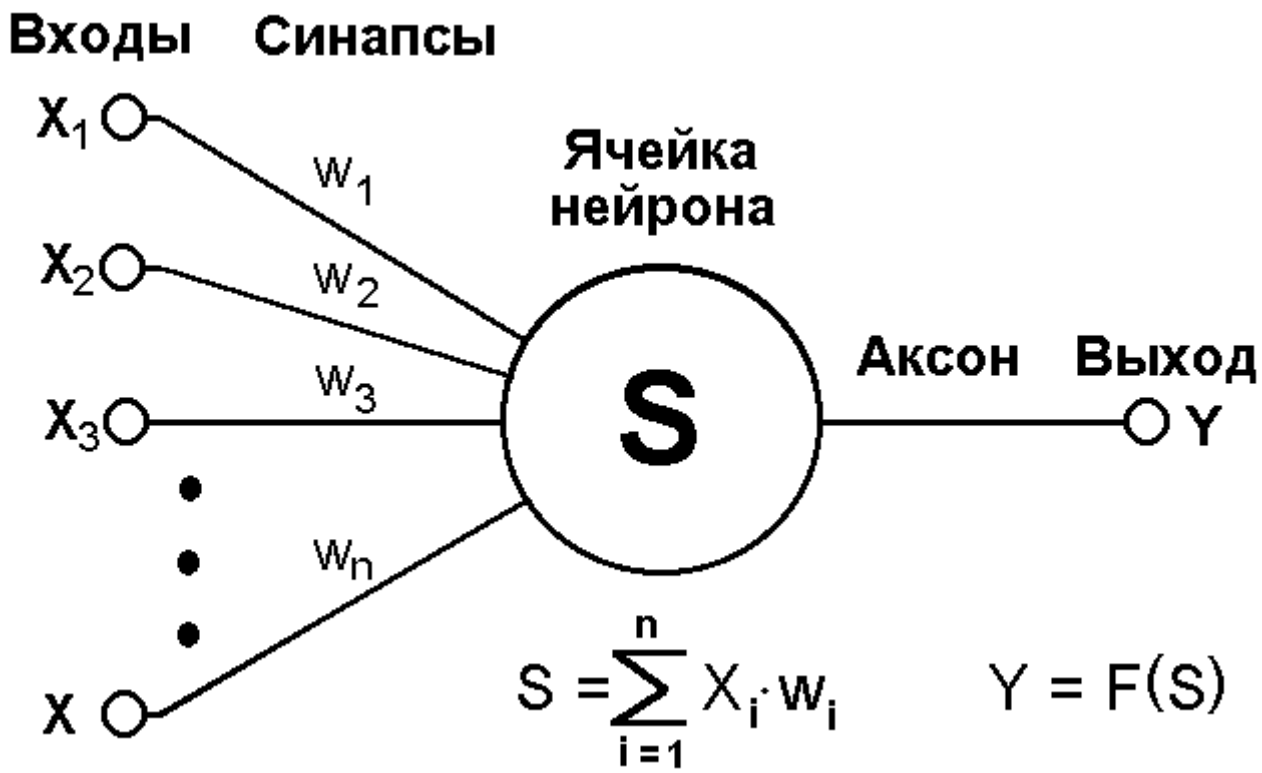

Рис. 11.1. Искусственный нейрон

Нейрон характеризуется текущим состоянием и обладает группой синапсов - однонаправленных входных связей, соединенных с выходами других нейронов.

Нейрон имеет аксон - выходную связь данного нейрона, с которой сигнал (возбуждения или торможения) поступает на синапсы следующих нейронов.

Каждый синапс характеризуется величиной синаптической связи (ее весом wi).

Текущее состояние нейрона определяется как взвешенная сумма его входов:

$$s = \sum_{i=1}^{n} x_i \cdot w_i$$

Выход нейрона есть функция его состояния:

```
y = f(s).
```

Активационная функция, которую также называют характеристической функцией, - это нелинейная функция, вычисляющая выходной сигнал формального нейрона.

Часто используемые активационные функции:

- Жесткая пороговая функция.
- Линейный порог.
- Сигмоидальная функция.

Выбор активационной функции определяется спецификой поставленной задачи либо ограничениями, накладываемыми некоторыми алгоритмами обучения.

Нелинейный преобразователь - это элемент искусственного нейрона, преобразующий текущее состояние нейрона (выходной сигнал адаптивного сумматора) в выходной сигнал нейрона по некоторому нелинейному закону (активационной функции).

Точка ветвления (выход) - это элемент формального нейрона, посылающий его выходной сигнал по нескольким адресам и имеющий один вход и несколько выходов.

На вход точки ветвления обычно подается выходной сигнал нелинейного преобразователя, который затем посылается на входы других нейронов.

Архитектура нейронных сетей

Нейронные сети могут быть синхронные и асинхронные.

В синхронных нейронных сетях в каждый момент времени свое состояние меняет лишь один нейрон.

В асинхронных - состояние меняется сразу у целой группы нейронов, как правило, у всего слоя [44].

Можно выделить две базовые архитектуры - слоистые и полносвязные сети [45, 46].

Ключевым в слоистых сетях является понятие слоя.

Слой - один или несколько нейронов, на входы которых подается один и тот же общий сигнал.

Слоистые нейронные сети - нейронные сети, в которых нейроны разбиты на отдельные группы (слои) так, что обработка информации осуществляется послойно.

В слоистых сетях нейроны i-го слоя получают входные сигналы, преобразуют их и через точки ветвления передают нейронам (i+1) слоя. И так до k-го слоя, который выдает выходные сигналы для интерпретатора и пользователя. Число нейронов в каждом слое не связано с количеством нейронов в других слоях, может быть произвольным.

В рамках одного слоя данные обрабатываются параллельно, а в масштабах всей сети обработка ведется последовательно - от слоя к слою. К слоистым нейронным сетям относятся, например, многослойные персептроны, сети радиальных базисных функций, когнитрон, некогнитрон, сети ассоциативной памяти.

Однако сигнал не всегда подается на все нейроны слоя. В когнитроне, например, каждый нейрон текущего слоя получает сигналы только от близких ему нейронов предыдущего слоя.

Слоистые сети, в свою очередь, могут быть однослойными и многослойными [46].

Однослойная сеть - сеть, состоящая из одного слоя.

Многослойная сеть - сеть, имеющая несколько слоев.

В многослойной сети первый слой называется входным, последующие - внутренними или скрытыми, последний слой - выходным. Таким образом, промежуточные слои - это все слои в многослойной нейронной сети, кроме входного и выходного.

Входной слой сети реализует связь с входными данными, выходной - с выходными.

Таким образом, нейроны могут быть входными, выходными и скрытыми.

Входной слой организован из входных нейронов (input neuron), которые получают данные и распространяют их на входы нейронов скрытого слоя сети.

Скрытый нейрон (hidden neuron) - это нейрон, находящийся в скрытом слое нейронной сети.

Выходные нейроны (output neuron), из которых организован выходной слой сети, выдает результаты работы нейронной сети.

В полносвязных сетях каждый нейрон передает свой выходной сигнал остальным нейронам, включая самого себя. Выходными сигналами сети могут быть все или некоторые выходные сигналы нейронов после нескольких тактов функционирования сети. Все входные сигналы подаются всем нейронам.

Обучение нейронных сетей

Перед использованием нейронной сети ее необходимо обучить.

Процесс обучения нейронной сети заключается в подстройке ее внутренних параметров под конкретную задачу.

Алгоритм работы нейронной сети является итеративным, его шаги называют эпохами или циклами.

Эпоха - одна итерация в процессе обучения, включающая предъявление всех примеров из обучающего множества и, возможно, проверку качества обучения на контрольном множестве.

Процесс обучения осуществляется на обучающей выборке.

Обучающая выборка включает входные значения и соответствующие им выходные значения набора данных. В ходе обучения нейронная сеть находит некие зависимости выходных полей от входных.

Таким образом, перед нами ставится вопрос - какие входные поля (признаки) нам необходимо использовать. Первоначально выбор осуществляется эвристически, далее количество входов может быть изменено.

Сложность может вызвать вопрос о количестве наблюдений в наборе данных. И хотя существуют некие правила, описывающие связь между необходимым количеством наблюдений и размером сети, их верность не доказана.

Количество необходимых наблюдений зависит от сложности решаемой задачи. При увеличении количества признаков количество наблюдений возрастает нелинейно, эта проблема носит название "проклятие размерности". При недостаточном количестве данных рекомендуется использовать линейную модель.

Аналитик должен определить количество слоев в сети и количество нейронов в каждом слое.

Далее необходимо назначить такие значения весов и смещений, которые смогут минимизировать ошибку решения. Веса и смещения автоматически настраиваются таким образом, чтобы минимизировать разность между желаемым и полученным на выходе сигналами, которая называется ошибка обучения.

Ошибка обучения для построенной нейронной сети вычисляется путем сравнения выходных и целевых (желаемых) значений. Из полученных разностей формируется функция ошибок.

Функция ошибок - это целевая функция, требующая минимизации в процессе управляемого обучения нейронной сети.

С помощью функции ошибок можно оценить качество работы нейронной сети во время обучения. Например, часто используется сумма квадратов ошибок.

От качества обучения нейронной сети зависит ее способность решать поставленные перед ней задачи.

Переобучение нейронной сети

При обучении нейронных сетей часто возникает серьезная трудность, называемая проблемой переобучения (overfitting).

Переобучение, или чрезмерно близкая подгонка - излишне точное соответствие нейронной сети конкретному набору обучающих примеров, при котором сеть теряет способность к обобщению.

Переобучение возникает в случае слишком долгого обучения, недостаточного числа обучающих примеров или переусложненной структуры нейронной сети.

Переобучение связано с тем, что выбор обучающего (тренировочного) множества является случайным. С первых шагов обучения происходит уменьшение ошибки. На последующих шагах с целью уменьшения ошибки (целевой функции) параметры подстраиваются под особенности обучающего множества. Однако при этом происходит "подстройка" не под общие закономерности ряда, а под особенности его части - обучающего подмножества. При этом точность прогноза уменьшается.

Один из вариантов борьбы с переобучением сети - деление обучающей выборки на два множества (обучающее и тестовое).

На обучающем множестве происходит обучение нейронной сети. На тестовом множестве осуществляется проверка построенной модели. Эти множества не должны пересекаться.

С каждым шагом параметры модели изменяются, однако постоянное уменьшение значения целевой функции происходит именно на обучающем множестве. При разбиении множества на два мы можем наблюдать изменение ошибки прогноза на тестовом множестве параллельно с наблюдениями над обучающим множеством. Какое-то количество шагов ошибки прогноза уменьшается на обоих множествах. Однако на определенном шаге ошибка на тестовом множестве начинает возрастать, при этом ошибка на обучающем множестве продолжает уменьшаться. Этот момент считается концом реального или настоящего обучения, с него и начинается переобучение.

Описанный процесс проиллюстрирован на рис. 11.2.

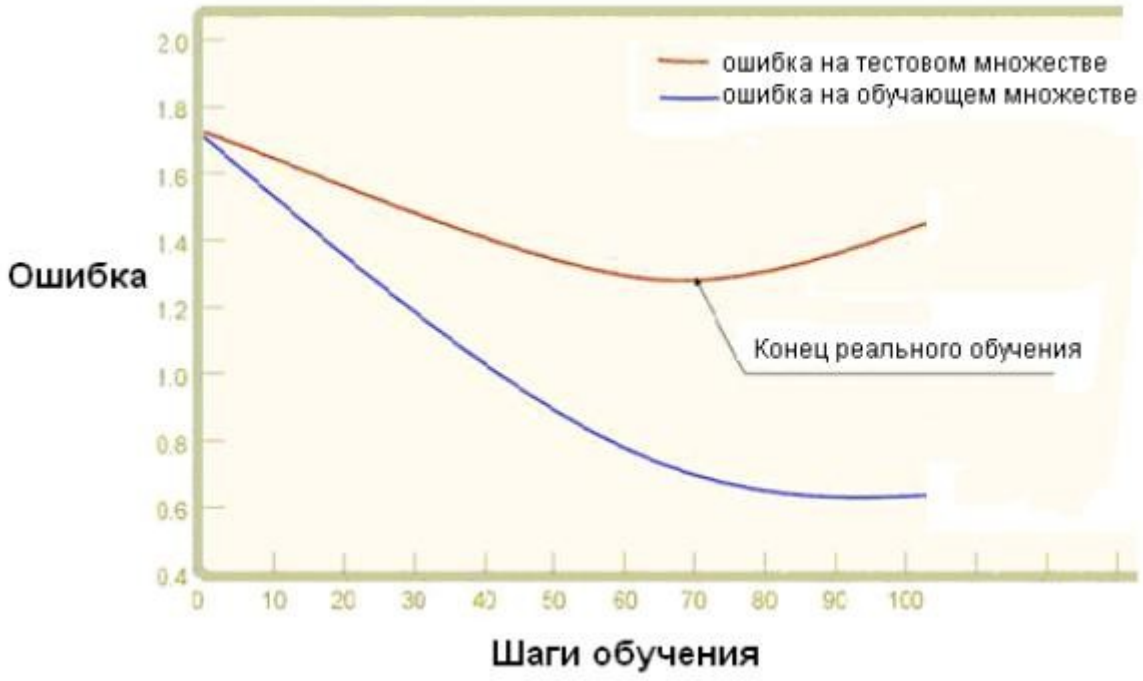

Рис. 11.2. Процесс обучений сети. Явление переобучения

На первом шаге ошибки прогноза для обучающего и тестового множества одинаковы. На последующих шагах значения обеих ошибок уменьшаются, однако с семидесятого шага ошибка на тестовом множестве начинает возрастать, т.е. начинается процесс переобучения сети.

Прогноз на тестовом множестве является проверкой работоспособности построенной модели. Ошибка на тестовом множестве может являться ошибкой прогноза, если тестовое множество максимально приближено к текущему моменту.

Модели нейронных сетей

Рассмотрим наиболее простые модели нейронных сетей: однослойный и многослойный персептрон.

Персептрон

Большое количество моделей персептрона рассмотрено в основополагающей работе Розенблатта [47]. Простейшая модель нейронной сети - однослойный персептрон.

Однослойный персептрон (персептрон Розенблатта) - однослойная нейронная сеть, все нейроны которой имеют жесткую пороговую функцию активации.

Однослойный персептрон имеет простой алгоритм обучения и способен решать лишь самые простые задачи. Эта модель вызвала к себе большой интерес в начале 1960-х годов и стала толчком к развитию искусственных нейронных сетей.

Классический пример такой нейронной сети - однослойный трехнейронный персептрон - представлен на рис. 11.3.

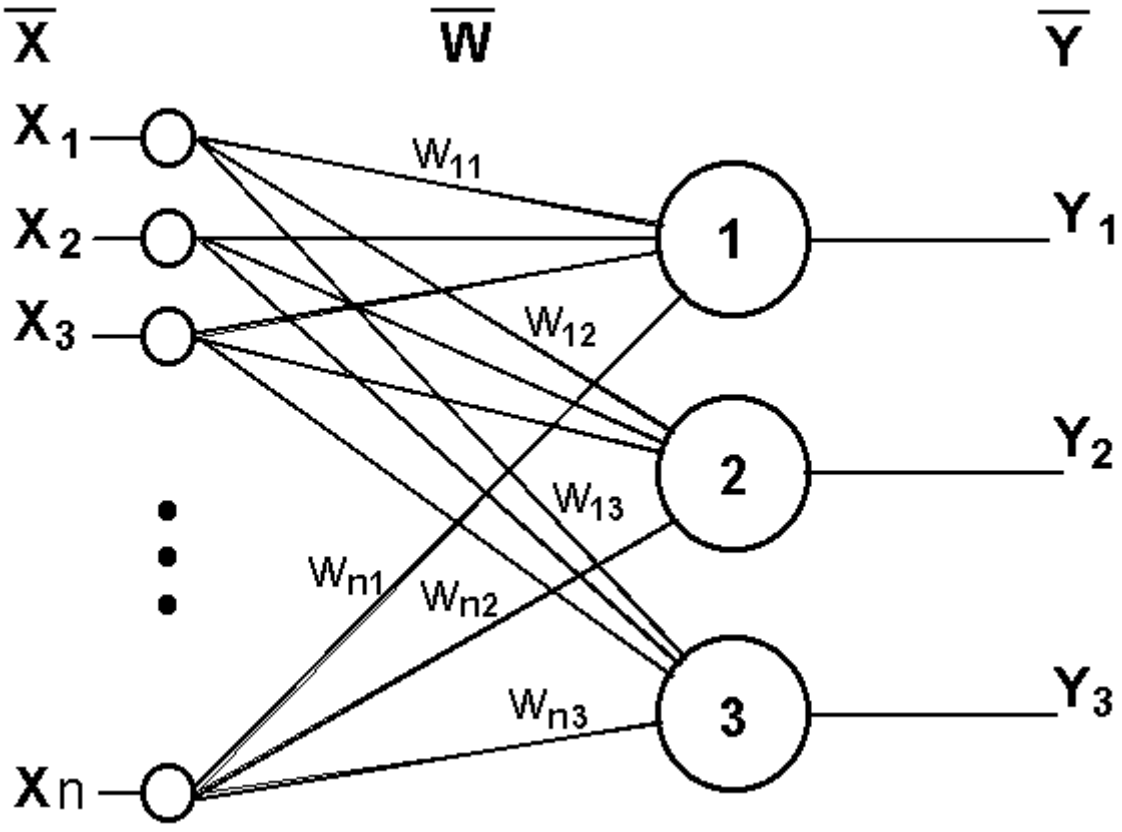

Рис. 11.3. Однослойный трехнейронный персептрон

Сеть, изображенная на рисунке, имеет n входов, на которые поступают сигналы, идущие по синапсам на 3 нейрона. Эти три нейрона образуют единственный слой данной сети и выдают три выходных сигнала.

Многослойный персептрон (MLP) - нейронная сеть прямого распространения сигнала (без обратных связей), в которой входной сигнал преобразуется в выходной, проходя последовательно через несколько слоев.

Первый из таких слоев называют входным, последний - выходным. Эти слои содержат так называемые вырожденные нейроны и иногда в количестве слоев не учитываются. Кроме входного и выходного слоев, в многослойном персептроне есть один или несколько промежуточных слоев, которые называют скрытыми.

В этой модели персептрона должен быть хотя бы один скрытый слой. Присутствие нескольких таких слоев оправдано лишь в случае использования нелинейных функций активации.

Пример двухслойного персептрона представлен на рис. 11.4.

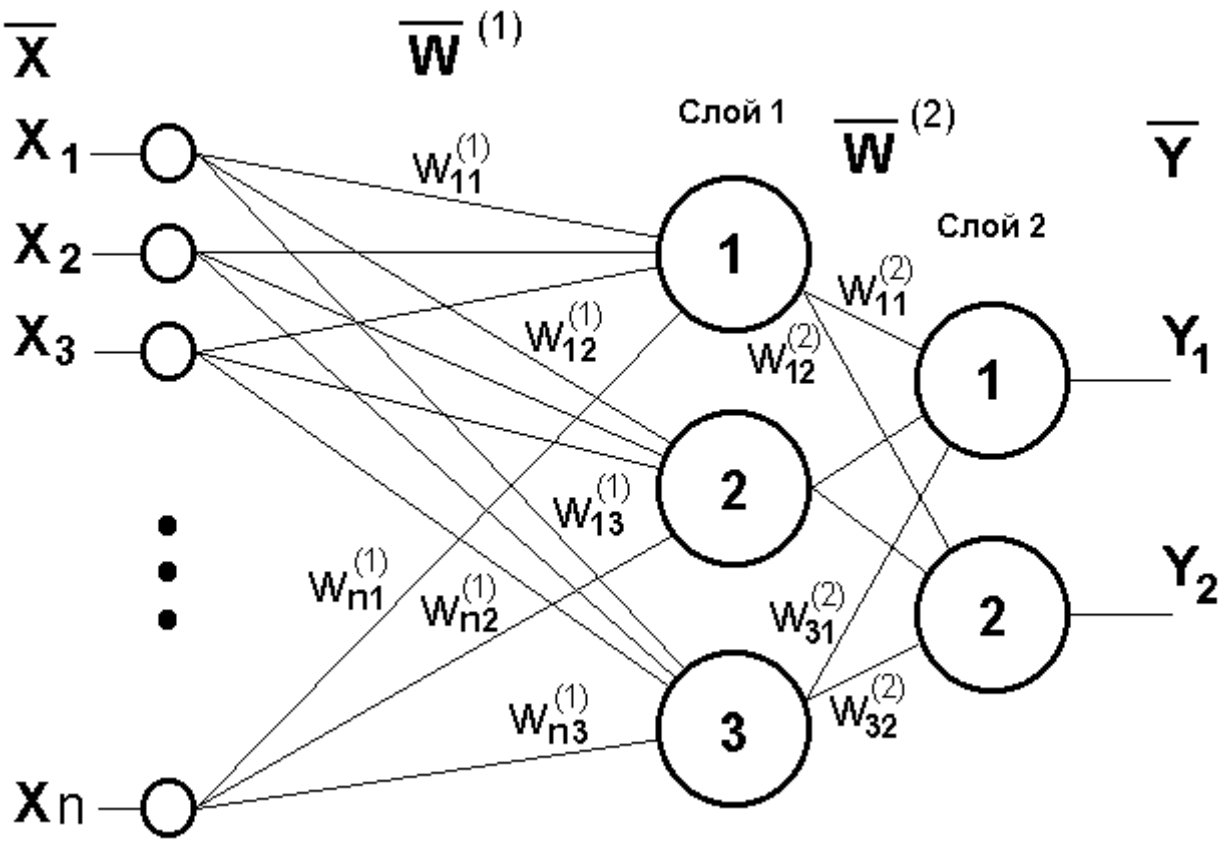

Рис. 11.4. Двухслойный перцептрон

Сеть, изображенная на рисунке, имеет n входов. На них поступают сигналы, идущие далее по синапсам на 3 нейрона, которые образуют первый слой. Выходные сигналы первого слоя передаются двум нейронам второго слоя. Последние, в свою очередь, выдают два выходных сигнала.

Метод обратного распространения ошибки (Back propagation, backprop) - алгоритм обучения многослойных персептронов, основанный на вычислении градиента функции ошибок. В процессе обучения веса нейронов каждого слоя нейросети корректируются с учетом сигналов, поступивших с предыдущего слоя, и невязки каждого слоя, которая вычисляется рекурсивно в обратном направлении от последнего слоя к первому.

Другие модели нейронных сетей будут рассмотрены в следующей лекции.

Программное обеспечение для работы с нейронными сетями

Программное обеспечение, имитирующее работу нейронной сети, называют нейросимулятором либо нейропакетом.

Большинство нейропакетов включают следующую последовательность действий:

- Создание сети (выбор пользователем параметров либо одобрение установленных по умолчанию).
- Обучение сети.

- Выдача пользователю решения.

Существует огромное разнообразие нейропакетов, возможность использования нейросетей включена также практически во все известные статистические пакеты.

Среди специализированных нейропакетов можно назвать такие: BrainMaker, NeuroOffice, NeuroPro, и др.

Критерии сравнения нейропакетов: простота применения, наглядность представляемой информации, возможность использовать различные структуры, скорость работы, наличие документации. Выбор определяется квалификацией и требованиями пользователя.

Пример решения задачи

Рассмотрим решение задачи "Выдавать ли кредит клиенту" в аналитическом пакете Deductor (BaseGroup).

В качестве обучающего набора данных выступает база данных, содержащая информацию о клиентах, в частности: Сумма кредита, Срок кредита, Цель кредитования, Возраст, Пол, Образование, Частная собственность, Квартира, Площадь квартиры. На основе этих данных необходимо построить модель, которая сможет дать ответ, входит ли Клиент, желающий получить кредит, в группу риска невозврата кредита, т.е. пользователь должен получить ответ на вопрос "Выдавать ли кредит?". Задача относится к группе задач классификации, т.е. обучения с учителем.

Данные для анализа находятся в файле credit.txt. Импортируем данные из файла при помощи мастера импорта. Запускаем мастер обработки

и выбираем метод обработки данных - нейронная сеть. Задаем назначения исходных столбцов данных. Выходной столбец в нашей задаче - "Давать кредит", все остальные - входные. Этот шаг проиллюстрирован на рис. 11.5.

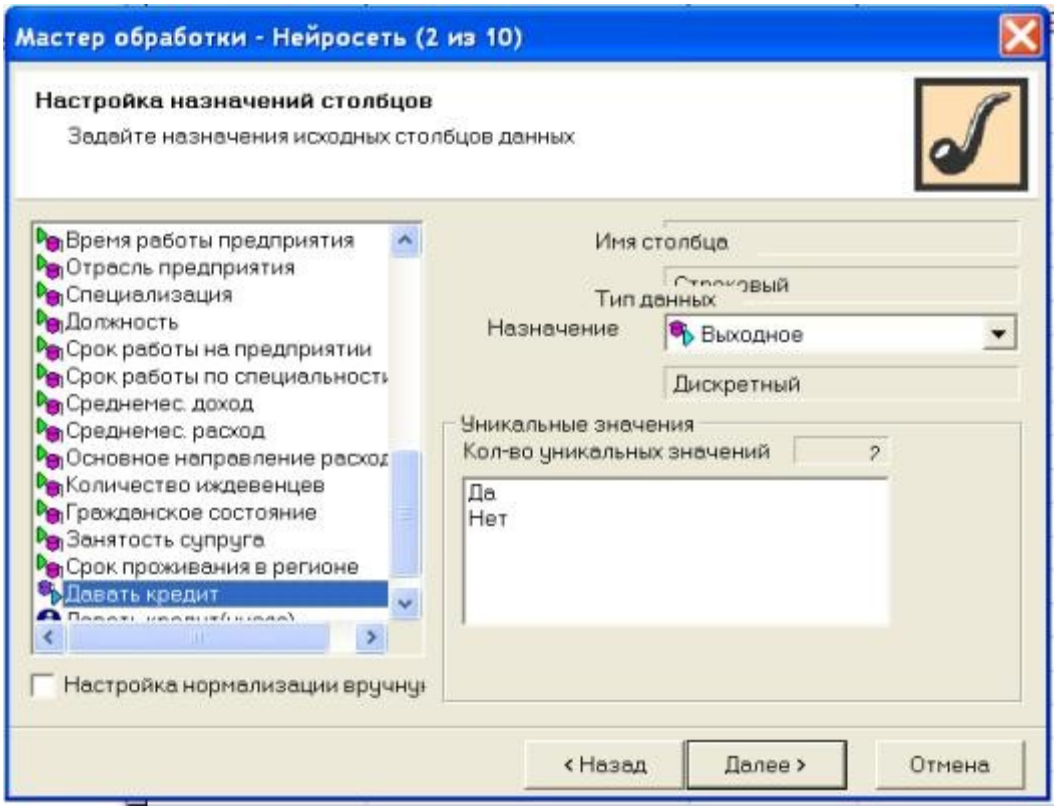

Рис. 11.5. Шаг "Настройка назначений столбцов"

На следующем шаге мастер предлагает разбить исходное множество данных на обучающее и тестовое. Способ разбиения исходного множества данных по умолчанию задан "Случайно". Этот шаг представлен на рис. 11.6.

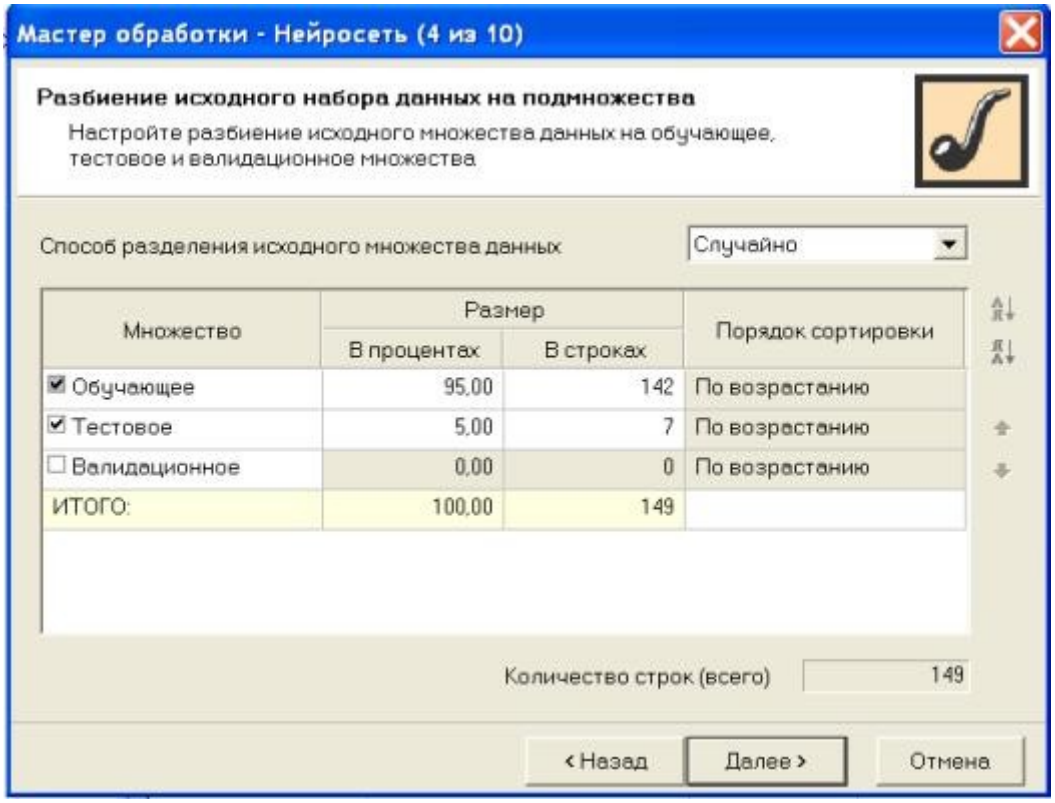

Рис. 11.6. Шаг "Разбиение исходного набора данных на подмножества"

На следующем шаге необходимо определить структуру нейронной сети, т.е. указать количество нейронов в входом слое - 33 (количество входных переменных), в скрытом слое - 1, в выходном слое - 1 (количество выходных переменных). Активационная функция - Сигмоида, и ее крутизна равна единице. Этот шаг проиллюстрирован на рис. 11.7.

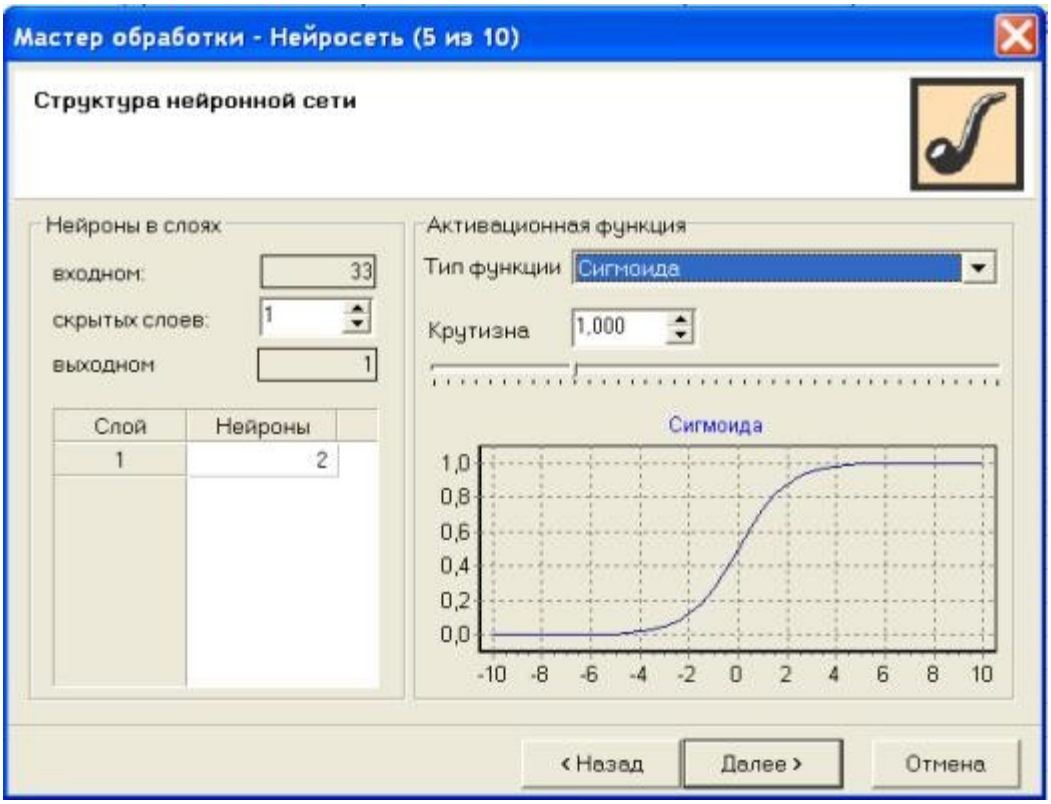

Рис. 11.7. Шаг "Структура нейронной сети"

Далее выбираем алгоритм и параметры обучения нейронной сети. Этот шаг имеет название "Настройка процесса обучения нейронной сети", он представлен на рис. 11.8.

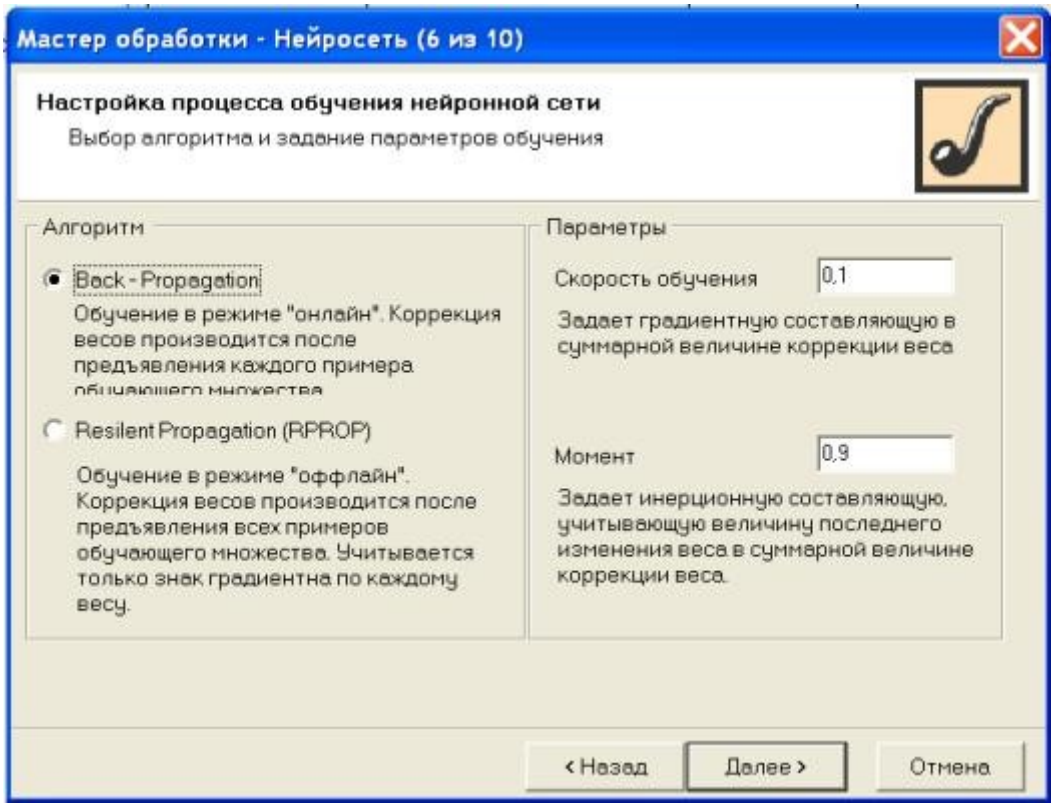

Рис. 11.8. Шаг "Настройка процесса обучения нейронной сети"

На следующем шаге настраиваем условия остановки обучения. Будем считать пример распознанным, если ошибка меньше 0,005, и укажем условие остановки обучения при достижении эпохи 10000.

На следующем шаге запускаем процесс обучения и наблюдаем за изменением величины ошибки и процентом распознанных примеров в обучающем и тестовом множествах. В нашем случае мы видим, что на эпохе № 4536 в обучающем множестве распознано 83,10% примеров, а на тестовом - 85,71% примеров. Фрагмент этого процесса проиллюстрирован на рис. 11.9.

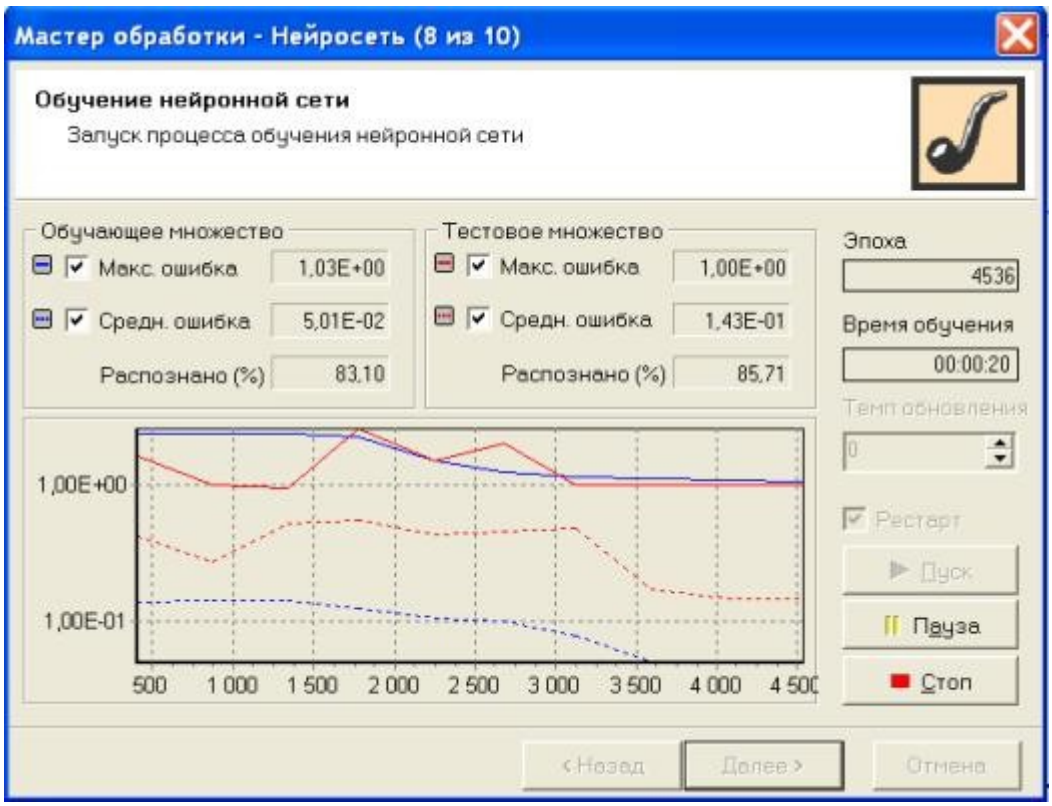

Рис. 11.9. Шаг "Обучение нейронной сети"

После окончания процесса обучения для интерпретации полученных результатов мы имеем возможность выбрать визуализаторы из списка предложенных. Выберем такие: таблица сопряженности, граф нейросети, анализ "что, если", и при помощи них проанализируем полученные данные [48].

На рис. 11.10 показана таблица сопряженности. По ее диагонали расположены примеры, которые были правильно распознаны, т.е. 55 клиентов, которым можно выдавать кредит, и 89 клиентов, которым выдавать кредит не стоит. В остальных ячейках расположены те клиенты, которые были отнесены к другому классу (1 и 4). Можно считать, что правильно классифицированы практически все примеры - 96,64%.

Кросс таблица | Кросс диаграмма | Граф нейро-сети | Что-если | Таблица сопряженности

Давать кредит

	Классифицировано		
Фактически	Да	Нет	Итого
Да	55	4	59
Нет	1	89	90
Итого	56	93	149

Рис. 11.10. Таблица сопряженности

Визуализатор "что-если" позволяет провести эксперимент. Данные по потенциальному получателю кредита следует ввести в соответствующие поля, и построенная модель

рассчитает значение поля "Давать кредит" - "Да" или "Нет", т.е. решит поставленную задачу.

Пакет Matlab

Пакет MATLAB (The MathWorks) также предоставляет пользователям возможность работы с нейронными сетями. Входящий в стандартную поставку MATLAB "Neural Network Toolbox" предоставляет широкие возможности для работы с нейронными сетями всех типов.

Преимущество пакета MATLAB состоит в том, что при его использовании пользователь не ограничен моделями нейронных сетей и их параметрами, жестко заложенными в нейросимуляторе, а имеет возможность самостоятельно сконструировать ту сеть, которую считает оптимальной для решения поставленной задачи.

Рассмотрим пример конструирования нейронной сети в пакете Matlab.

Пусть имеется 15 независимых переменных - показателей деятельности фирмы и одна зависимая переменная - объем продаж. Имеем базу данных за прошедший год. Необходимо построить понедельный прогноз объемов продаж на месяц. Для решения задачи предлагается использовать трехслойную сеть обратного распространения.

Сформируем такую сеть, которая включает 15 нейронов во входом слое (по количеству входных переменных), 8 нейронов во втором слое и 1 нейрон в выходном слое (по количеству выходных переменных).

Для каждого слоя выберем передаточную функцию: первый слой - logsig, второй - logsig, третий - purelin.

В среде Matlab синтаксис такой нейронной сети выглядит следующим образом:

```
Net=netff(PR, [S1,S2, : , Sn],{TF1,TF2, : , TFn},btf, blf, pf),
```

где PR - массив минимальных и максимальных значений для R векторов входа;

Si - количество нейронов в i-м слое;

TFi - функция активации слоя i;

btf - обучающая функция, реализующая метод обратного распространения;

blf - функция настройки, реализующая метод обратного распространения;

pf - критерий качества обучения.

Активационной функцией может выступать любая дифференцируемая функция, например, tansig, logsig, purelin.

```
Net=netff(minmax (P), [n,m, l],{ logsig, logsig, purelin },trainpr),
```

где P - множество входных векторов;

n - количество входов НС;

m - количество нейронов в скрытом слое;

l - количество выходов НС.

Необходимо также установить метод расчета значения ошибки. Например, если выбран метод наименьших квадратов, то эта функция будет выглядеть так: Net.performFcn='SSE'.

Для установления максимального количества эпох равным 10000 воспользуемся функцией: net.trainParam.epochs=10000.

Запустить процесс обучения можно таким образом:

```
[net,tr]=train(net,P,T);
```

После окончания обучения сети ее можно сохранить в файле, например, с именем nn1.mat.

Для этого необходимо выполнить команду:

```
save nn1 net;
```

Таким образом, в пакете возможно конструирование сети любой сложности и нет необходимости привязываться к ограничениям, накладываемым нейросимуляторами. Однако для работы с нейронными сетями в пакете Matlab необходимо изучить как саму среду, так и большинство функций Neural Network Toolbox. Для более детального изучения конструирования нейронных сетей в Neural Network Toolbox можно порекомендовать [49, 50].

Нейронные сети. Самоорганизующиеся карты Кохонена.

Классификация нейронных сетей

Одна из возможных классификаций нейронных сетей - по направленности связей.

Нейронные сети бывают с обратными связями и без обратных связей.

Сети без обратных связей

- Сети с обратным распространением ошибки.

  Сети этой группы характеризуются фиксированной структурой, итерационным обучением, корректировкой весов по ошибкам. Такие сети были рассмотрены в предыдущей лекции.

- Другие сети (когнитрон, неокогнитрон, другие сложные модели).

Преимуществами сетей без обратных связей является простота их реализации и гарантированное получение ответа после прохождения данных по слоям.

Недостатком этого вида сетей считается минимизация размеров сети - нейроны многократно участвуют в обработке данных.

Меньший объем сети облегчает процесс обучения.

Сети с обратными связями

- Сети Хопфилда (задачи ассоциативной памяти).
- Сети Кохонена (задачи кластерного анализа).

Преимуществами сетей с обратными связями является сложность обучения, вызванная большим числом нейронов для алгоритмов одного и того же уровня сложности.

Недостатки этого вида сетей - требуются специальные условия, гарантирующие сходимость вычислений.

Другая классификация нейронных сетей: сети прямого распространения и рекуррентные сети.

Сети прямого распространения

- Персептроны.
- Сеть Back Propagation.
- Сеть встречного распространения.
- Карта Кохонена.

Рекуррентные сети. Характерная особенность таких сетей - наличие блоков динамической задержки и обратных связей, что позволяет им обрабатывать динамические модели.

- Сеть Хопфилда.
- Сеть Элмана - сеть, состоящая из двух слоев, в которой скрытый слой охвачен динамической обратной связью, что позволяет учесть предысторию наблюдаемых процессов и накопить информацию для выработки правильной стратегии управления. Эти сети применяются в системах управления движущимися объектами.

Нейронные сети могут обучаться с учителем или без него.

При **обучении с учителем** для каждого обучающего входного примера требуется знание правильного ответа или функции оценки качества ответа. Такое обучение называют управляемым. Нейронной сети предъявляются значения входных и выходных сигналов, а она по определенному алгоритму подстраивает веса синаптических связей. В процессе обучения производится корректировка весов сети по результатам сравнения фактических выходных значений с входными, известными заранее.

При **обучении без учителя** раскрывается внутренняя структура данных или корреляции между образцами в наборе данных. Выходы нейронной сети формируются самостоятельно, а веса изменяются по алгоритму, учитывающему только входные и производные от них сигналы. Это обучение называют также неуправляемым. В результате такого обучения объекты или примеры распределяются по категориям, сами категории и их количество могут быть заранее не известны.

Подготовка данных для обучения

При подготовке данных для обучения нейронной сети необходимо обращать внимание на следующие существенные моменты.

Количество наблюдений в наборе данных. Следует учитывать тот фактор, что чем больше размерность данных, тем больше времени потребуется для обучения сети.

Работа с выбросами. Следует определить наличие выбросов и оценить необходимость их присутствия в выборке.

Обучающая выборка должна быть представительной (репрезентативной).

Обучающая выборка не должна содержать противоречий, так как нейронная сеть однозначно сопоставляет выходные значения входным.

Нейронная сеть работает только с числовыми входными данными, поэтому важным этапом при подготовке данных является преобразование и кодирование данных.

При использовании на вход нейронной сети следует подавать значения из того диапазона, на котором она обучалась. Например, если при обучении нейронной сети на один из ее входов подавались значения от 0 до 10, то при ее применении на вход следует подавать значения из этого же диапазона или близлежащие.

Существует понятие нормализации данных. Целью нормализации значений является преобразование данных к виду, который наиболее подходит для обработки, т.е. данные, поступающие на вход, должны иметь числовой тип, а их значения должны быть распределены в определенном диапазоне. Нормализатор может приводить дискретные данные к набору уникальных индексов либо преобразовывать значения, лежащие в

произвольном диапазоне, в конкретный диапазон, например, [0..1]. Нормализация выполняется путем деления каждой компоненты входного вектора на длину вектора, что превращает входной вектор в единичный.

Выбор структуры нейронной сети

Выбор структуры нейронной сети обуславливается спецификой и сложностью решаемой задачи. Для решения некоторых типов задач разработаны оптимальные конфигурации [44, 51, 52].

В большинстве случаев выбор структуры нейронной сети определяется на основе объединения опыта и интуиции разработчика.

Однако существуют основополагающие **принципы**, которыми следует руководствоваться при разработке новой конфигурации [53]:

1. возможности сети возрастают с увеличением числа ячеек сети, плотности связей между ними и числом выделенных слоев;
2. введение обратных связей наряду с увеличением возможностей сети поднимает вопрос о динамической устойчивости сети;
3. сложность алгоритмов функционирования сети (в том числе, например, введение нескольких типов синапсов - возбуждающих, тормозящих и др.) также способствует усилению мощи НС.

Вопрос о необходимых и достаточных свойствах сети для решения того или иного рода задач представляет собой целое направление нейрокомпьютерной науки. Так как проблема синтеза нейронной сети сильно зависит от решаемой задачи, дать общие подробные рекомендации затруднительно. Очевидно, что процесс функционирования НС, то есть сущность действий, которые она способна выполнять, зависит от величин синаптических связей, поэтому, задавшись определенной структурой НС, отвечающей какой-либо задаче, разработчик сети должен найти оптимальные значения всех переменных весовых коэффициентов (некоторые синаптические связи могут быть постоянными).

Карты Кохонена

Самоорганизующиеся карты (Self-Organizing Maps, SOM)

Сети, называемые картами Кохонена, - это одна из разновидностей нейронных сетей, однако они принципиально отличаются от рассмотренных выше, поскольку используют неконтролируемое обучение. Напомним, что при таком обучении обучающее множество состоит лишь из значений входных переменных, в процессе обучения нет сравнивания выходов нейронов с эталонными значениями. Можно сказать, что такая сеть учится понимать структуру данных.

Идея сети Кохонена принадлежит финскому ученому Тойво Кохонену (1982 год). Основной принцип работы сетей - введение в правило обучения нейрона информации относительно его расположения.

В основе идеи сети Кохонена лежит аналогия со свойствами человеческого мозга. Кора головного мозга человека представляет собой плоский лист и свернута складками. Таким

образом, можно сказать, что она обладает определенными топологическими свойствами (участки, ответственные за близкие части тела, примыкают друг к другу и все изображение человеческого тела отображается на эту двумерную поверхность).

Задачи, решаемые при помощи карт Кохонена

Самоорганизующиеся карты могут использоваться для решения таких задач, как моделирование, прогнозирование, поиск закономерностей в больших массивах данных, выявление наборов независимых признаков и сжатие информации.

Наиболее распространенное применение сетей Кохонена - решение задачи классификации без учителя, т.е. кластеризации.

Напомним, что при такой постановке задачи нам дан набор объектов, каждому из которых сопоставлена строка таблицы (вектор значений признаков). Требуется разбить исходное множество на классы, т.е. для каждого объекта найти класс, к которому он принадлежит.

В результате получения новой информации о классах возможна коррекция существующих правил классификации объектов.

Вот два из распространенных применений карт Кохонена: разведочный анализ данных и обнаружение новых явлений [39].

Разведочный анализ данных. Сеть Кохонена способна распознавать кластеры в данных, а также устанавливать близость классов. Таким образом, пользователь может улучшить свое понимание структуры данных, чтобы затем уточнить нейросетевую модель. Если в данных распознаны классы, то их можно обозначить, после чего сеть сможет решать задачи классификации. Сети Кохонена можно использовать и в тех задачах классификации, где классы уже заданы, - тогда преимущество будет в том, что сеть сможет выявить сходство между различными классами.

Обнаружение новых явлений. Сеть Кохонена распознает кластеры в обучающих данных и относит все данные к тем или иным кластерам. Если после этого сеть встретится с набором данных, непохожим ни на один из известных образцов, то она не сможет классифицировать такой набор и тем самым выявит его новизну.

Обучение сети Кохонена

Сеть Кохонена, в отличие от многослойной нейронной сети, очень проста; она представляет собой два слоя: входной и выходной. Ее также называют самоорганизующей картой. Элементы карты располагаются в некотором пространстве, как правило, двумерном. Сеть Кохонена изображена на рис. 12.1

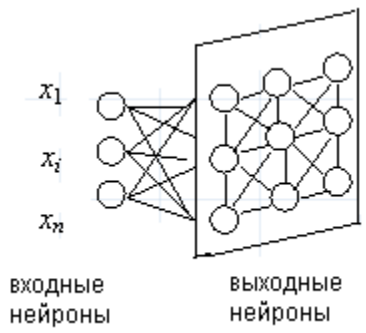

Рис. 12.1. Сеть Кохонена

Сеть Кохонена обучается методом последовательных приближений. В процессе обучения таких сетей на входы подаются данные, но сеть при этом подстраивается не под эталонное значение выхода, а под закономерности во входных данных. Начинается обучение с выбранного случайным образом выходного расположения центров.

В процессе последовательной подачи на вход сети обучающих примеров определяется наиболее схожий нейрон (тот, у которого скалярное произведение весов и поданного на вход вектора минимально). Этот нейрон объявляется победителем и является центром при подстройке весов у соседних нейронов. Такое правило обучения предполагает "соревновательное" обучение с учетом расстояния нейронов от "нейрона-победителя".

Обучение при этом заключается не в минимизации ошибки, а в подстройке весов (внутренних параметров нейронной сети) для наибольшего совпадения с входными данными.

Основной итерационный алгоритм Кохонена последовательно проходит ряд эпох, на каждой из которых обрабатывается один пример из обучающей выборки. Входные сигналы последовательно предъявляются сети, при этом желаемые выходные сигналы не определяются. После предъявления достаточного числа входных векторов синаптические веса сети становятся способны определить кластеры. Веса организуются так, что топологически близкие узлы чувствительны к похожим входным сигналам.

В результате работы алгоритма центр кластера устанавливается в определенной позиции, удовлетворительным образом кластеризующей примеры, для которых данный нейрон является "победителем". В результате обучения сети необходимо определить меру соседства нейронов, т.е. окрестность нейрона-победителя.

Окрестность представляет собой несколько нейронов, которые окружают нейрон-победитель [39].

Сначала к окрестности принадлежит большое число нейронов, далее ее размер постепенно уменьшается. Сеть формирует топологическую структуру, в которой похожие примеры образуют группы примеров, близко находящиеся на топологической карте.

Полученную карту можно использовать как средство визуализации при анализе данных. В результате обучения карта Кохонена классифицирует входные примеры на кластеры

(группы схожих примеров) и визуально отображает многомерные входные данные на плоскости нейронов.

Уникальность метода самоорганизующихся карт состоит в преобразовании n-мерного пространства в двухмерное. Применение двухмерных сеток связано с тем, что существует проблема отображения пространственных структур большей размерности.

Имея такое представление данных, можно визуально определить наличие или отсутствие взаимосвязи во входных данных.

Нейроны карты Кохонена располагают в виде двухмерной матрицы, раскрашивают эту матрицу в зависимости от анализируемых параметров нейронов.

На рис. 12.2 приведен пример карты Кохонена

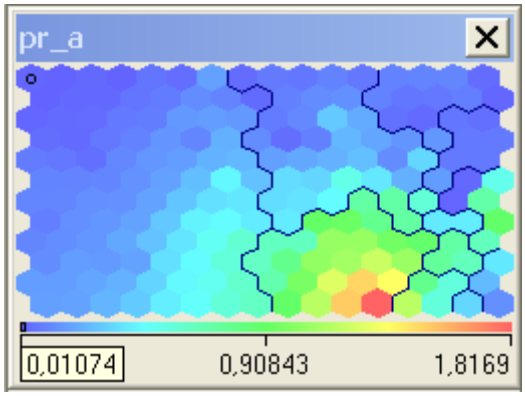

Рис. 12.2. Пример карты Кохонена

Что же означает ее раскраска? На рис.12.3 приведена раскраска карты, а точнее, ее i-го признака (показателя pr_a), в трехмерном представлении. Как мы видим, темно-синие участки на карте соответствуют наименьшим значениям показателя, красные - самым высоким.

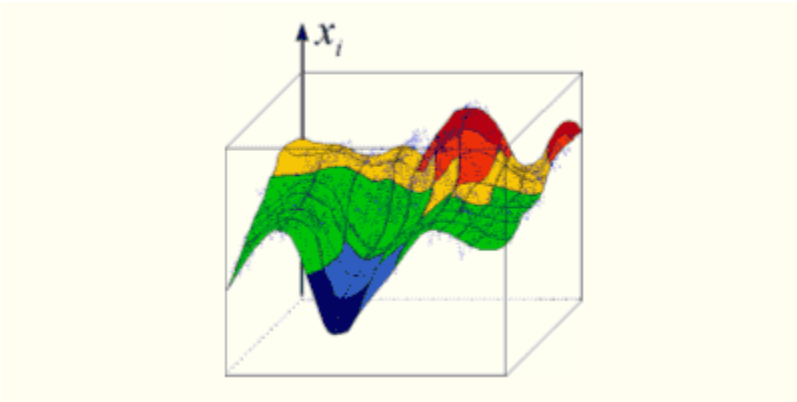

Рис. 12.3. Раскраска i-го признака в трехмерном пространстве

Теперь, возвращаясь к рисунку рис.12.2, мы можем сказать, какие объекты имеют наибольшие значения рассматриваемого показателя (группа объектов, обозначенная красным цветом), а какие - наименьшие значения (группа объектов, обозначенная синим цветом).

Таким образом, карты Кохонена (как и географические карты) можно отображать:

- в двухмерном виде, тогда карта раскрашивается в соответствии с уровнем выхода нейрона;
- в трехмерном виде.

В результате работы алгоритма получаем такие карты:

- карта входов нейронов;
- карта выходов нейронов;
- специальные карты.

Координаты каждой карты определяют положение одного нейрона. Так, координаты [15:30] определяют нейрон, который находится на пересечении 15-го столбца с 30-м рядом в матрице нейронов. Рассмотрим, что же представляют собой эти карты.

Карта входов нейронов.

Веса нейронов подстраиваются под значения входных переменных и отображают их внутреннюю структуру. Для каждого входа рисуется своя карта, раскрашенная в соответствии со значением конкретного веса нейрона.

При анализе данных используют несколько карт входов.

На одной из карт выделяют область определенного цвета - это означает, что соответствующие входные примеры имеют приблизительно одинаковое значение соответствующего входа. Цветовое распределение нейронов из этой области анализируется на других картах для определения схожих или отличительных характеристик. Пример рассмотренных карт входов будет приведен ниже.

Карта выходов нейронов.

На карту выходов нейронов проецируется взаимное расположение исследуемых входных данных. Нейроны с одинаковыми значениями выходов образуют кластеры - замкнутые области на карте, которые включают нейроны с одинаковыми значениями выходов.

Специальные карты. Это карта кластеров, матрица расстояний, матрица плотности попадания и другие карты, которые характеризуют кластеры, полученные в результате обучения сети Кохонена.

Важно понимать, что между всеми рассмотренными картами существует взаимосвязь - все они являются разными раскрасками одних и тех же нейронов. Каждый пример из обучающей выборки имеет одно и то же расположение на всех картах.

Пример решения задачи

Программное обеспечение, позволяющее работать с картами Кохонена, сейчас представлено множеством инструментов. Это могут быть как инструменты, включающие только реализацию метода самоорганизующихся карт, так и нейропакеты с целым набором структур нейронных сетей, среди которых - и карты Кохонена; также данный метод реализован в некоторых универсальных инструментах анализа данных.

К инструментарию, включающему реализацию метода карт Кохонена, относятся SoMine, Statistica, NeuroShell, NeuroScalp, Deductor и множество других. Для решения задачи будем использовать аналитический пакет Deductor.

Пусть имеется база данных коммерческих банков с показателями деятельности за текущий период. Необходимо провести их кластеризацию, т.е. выделить однородные группы банков на основе показателей из базы данных, всего показателей - 21.

Исходная таблица находится в файле "banks.xls". Она содержит показатели деятельности коммерческих банков за отчетный период.

Сначала импортируем данные из xls-файла в среду аналитического пакета.

На первом шаге мастера запускаем мастер обработки и выбираем из списка метод обработки "Карта Кохонена". Далее следует настроить назначения столбцов, т.е. для каждого столбца выбрать одно из назначений: входное, выходное, не используется и информационное. Укажем всем столбцам, соответствующим показателям деятельности банков, назначение "Входной". "Выходной" не назначаем.

Следующий шаг предлагает разбить исходное множество на обучающее, тестовое и валидационное. По умолчанию,программа предлагает разбить множество на обучающее - 95% и тестовое - 5%.

Эти шаги аналогичны шагам в мастере обработки для нейронных сетей, описанным в предыдущей Лекции.

На шаге № 5, изображенном на рис. 12.4 предлагается настроить параметры карты: количество ячеек по X и по Y их форму (шестиугольную или четырехугольную).

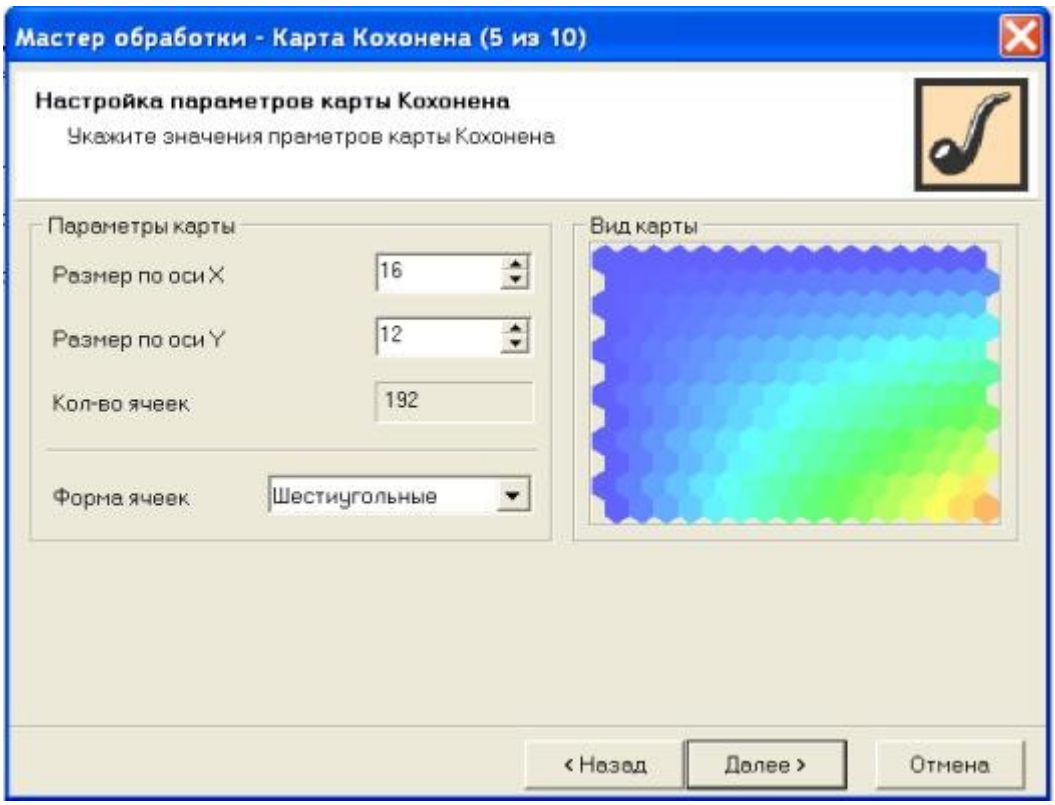

Рис. 12.4. Шаг № 5 "Настройка параметров карты Кохонена"

На шестом шаге "Настройка параметров остановки обучения", проиллюстрированном на рис. 12.5, устанавливаем параметры остановки обучения и устанавливаем эпоху, по достижению которой обучение будет прекращено.

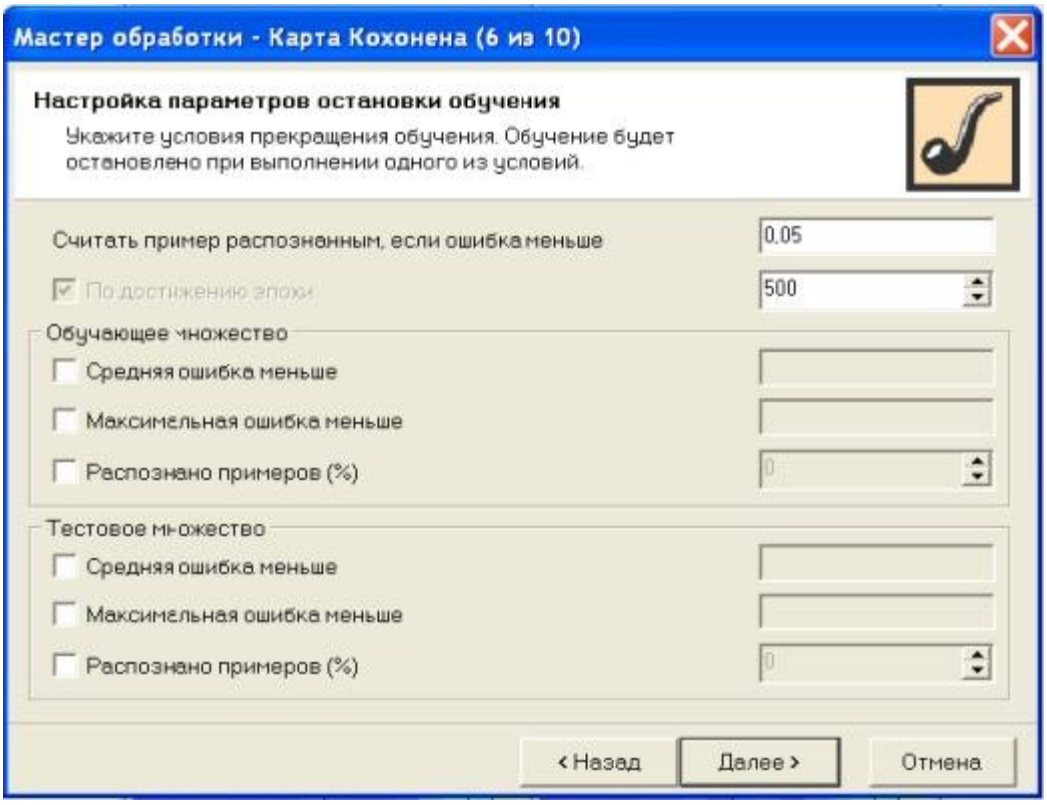

Рис. 12.5. Шаг № 6 "Настройка параметров остановки обучения"

На седьмом шаге, представленном на рис. 12.6, настраиваются другие параметры обучения: способ начальной инициализации, тип функции соседства. Возможны два варианта кластеризации: автоматическое определение числа кластеров с соответствующим уровнем значимости и фиксированное количество кластеров (определяется пользователем). Поскольку нам неизвестно количество кластеров, выберем автоматическое определение их количества.

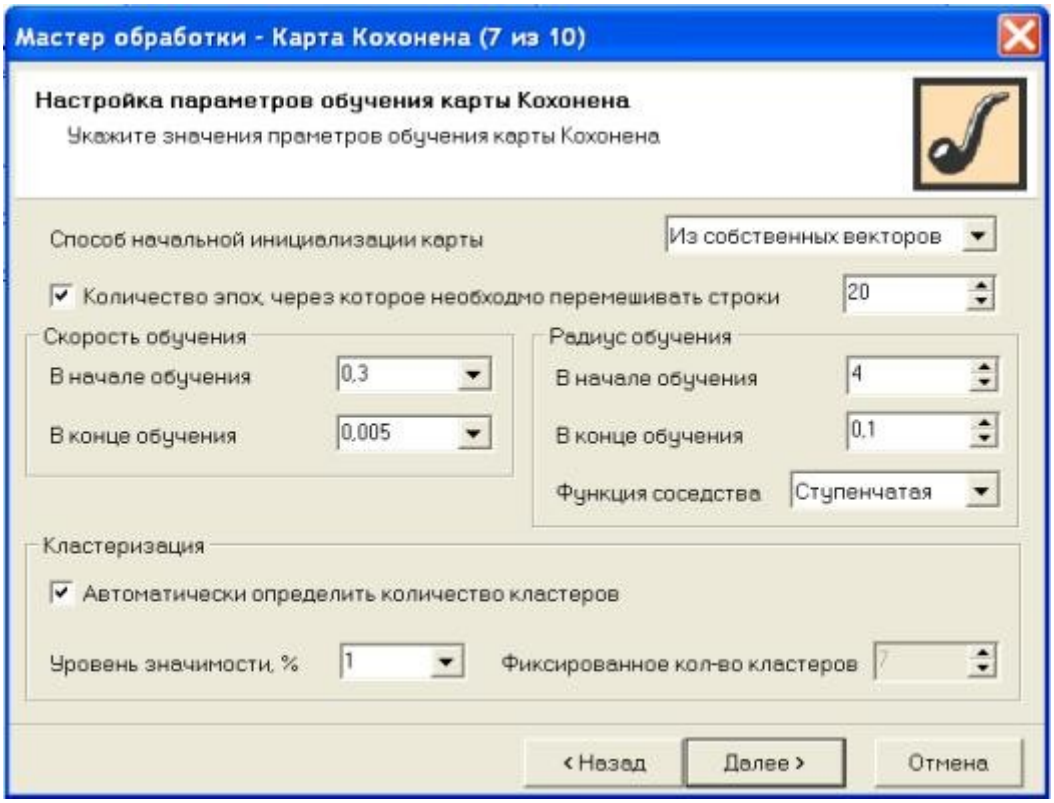

Рис. 12.6. Шаг № 7 "Настройка параметров остановки обучения"

На восьмом шаге запускаем процесс обучения сети - необходимо нажать на кнопку "Пуск" и дождаться окончания процесса обучения. Во время обучения можем наблюдать изменение количества распознанных примеров и текущие значения ошибок. Этот процесс аналогичен тому, что мы рассматривали при обучении нейронных сетей в предыдущей лекции.

По окончании обучения в списке визуализаторов выберем "Карту Кохонена" и визуализатор "Что-если". На последнем шаге настраиваем отображения карты Кохонена, этот шаг проиллюстрирован на рис. 12.7.

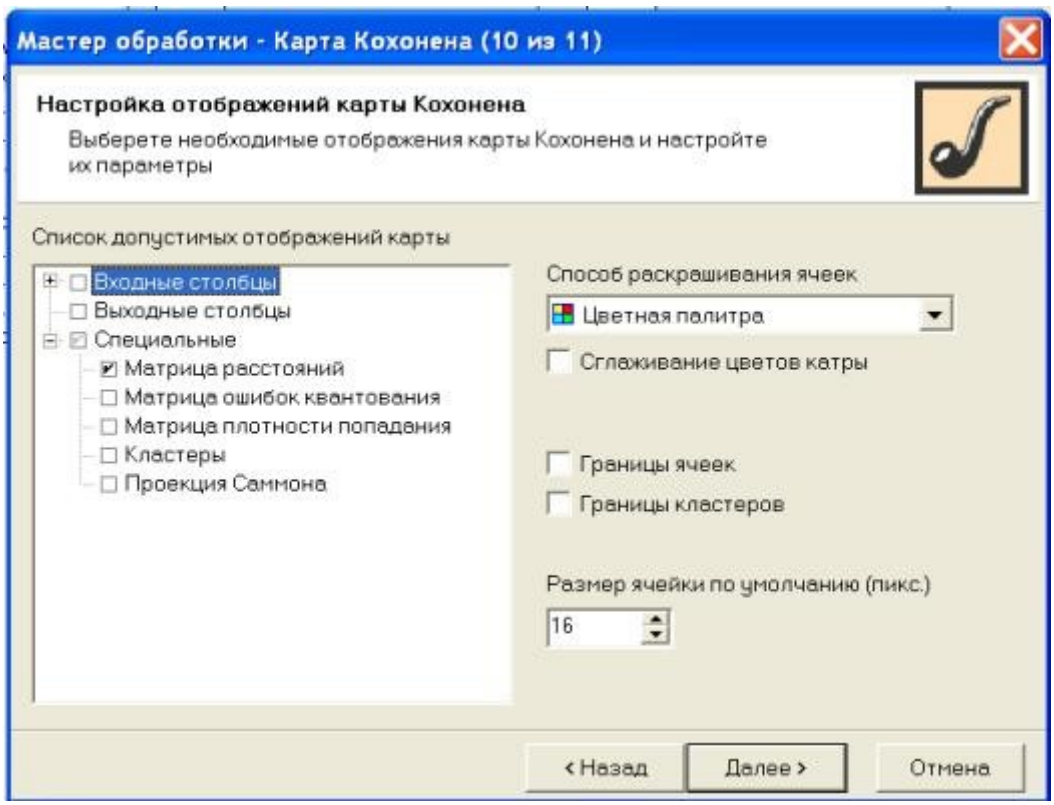

Рис. 12.7. "Шаг № 10 Настройка отображений карты Кохонена"

Укажем отображения всех входных, выходных столбцов, кластеров, а также поставим флажок "Границы кластеров" для четкого отображения границ.

Карты входов

При анализе карт входов рекомендуют использовать сразу несколько карт. Исследуем фрагмент карты, состоящий из карт трех входов, который приведен на рис. 12.8

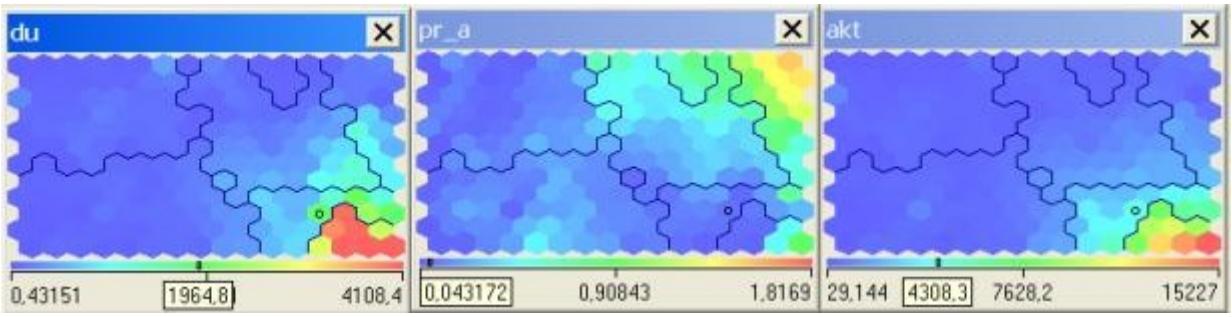

Рис. 12.8. Карты трех входов

На одной из карт выделяем область с наибольшими значениями показателя. Далее имеет смысл изучить эти же нейроны на других картах.

На первой карте наибольшие значения имеют объекты, расположенные в правом нижнем углу. Рассматривая одновременно три карты, мы можем сказать, что эти же объекты

имеют наибольшие значения показателя, изображенного на третьей карте. Также по раскраске первой и третьей карты можно сделать вывод, что существует взаимосвязь между этими показателями.

Также мы можем определить, например, такую характеристику: кластер, расположенный в правом верхнем углу, характеризуется низкими значениями показателей du (депозиты юридических лиц) и akt (активы банка) и высокими значениями показателей pr_a (прибыльность активов).

Эта информация позволяет так охарактеризовать кластер, находящийся в правом верхнем углу: это банки с небольшими активами, небольшими привлеченными депозитными средствами от юридических лиц, но с наиболее прибыльными активами, т.е. это группа небольших, но наиболее прибыльных банков.

Это лишь фрагмент вывода, который можно сделать, исследуя карту.

На следующем рисунке (рис. 12.9) приведена иллюстрация карт входов и выходов, последняя - эта карта кластеров. Здесь мы видим несколько карт входов (показателей деятельности банков) и сформированные кластеры, каждый из которых выделен отдельным цветом.

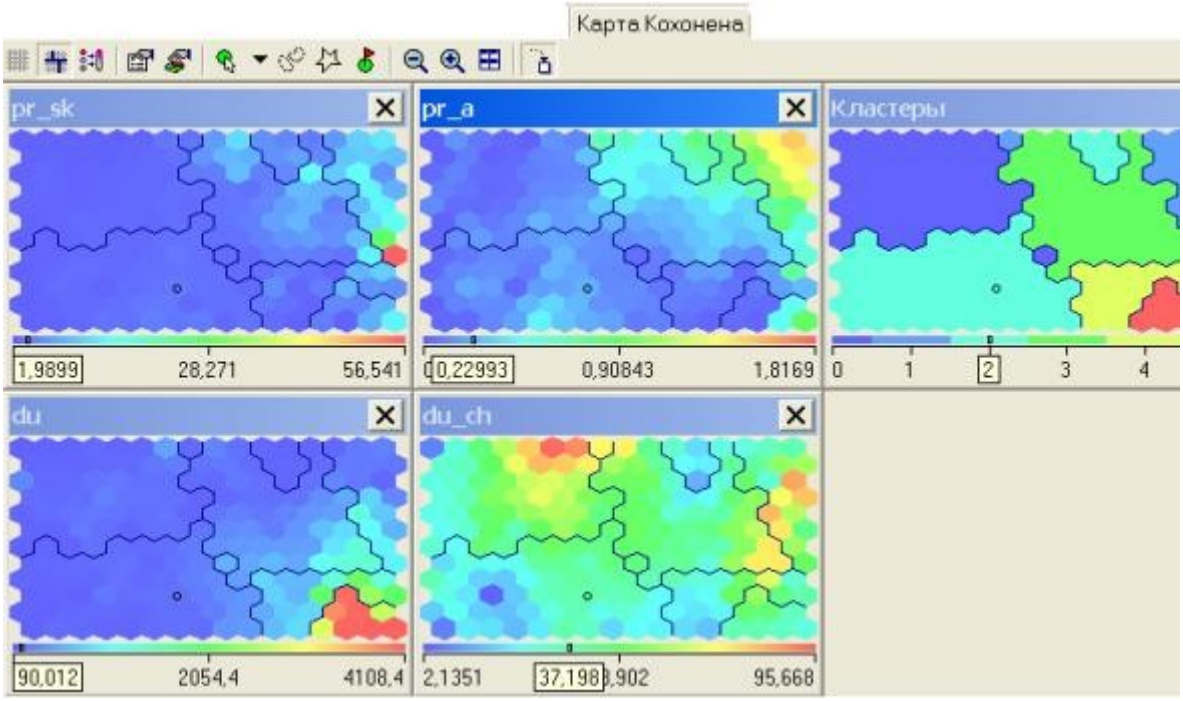

Рис. 12.9. Карты входов и выходов

Для нахождения конкретного объекта на карте необходимо нажать правой кнопкой мыши на исследуемом объекте и выбрать пункт "Найти ячейку на карте". Выполнение этой процедуры показано на рис. 12.10. В результате мы можем видеть как сам объект, так и значение того измерения, которое мы просматриваем. Таким образом, мы можем оценить положение анализируемого объекта, а также сравнить его с другими объектами.

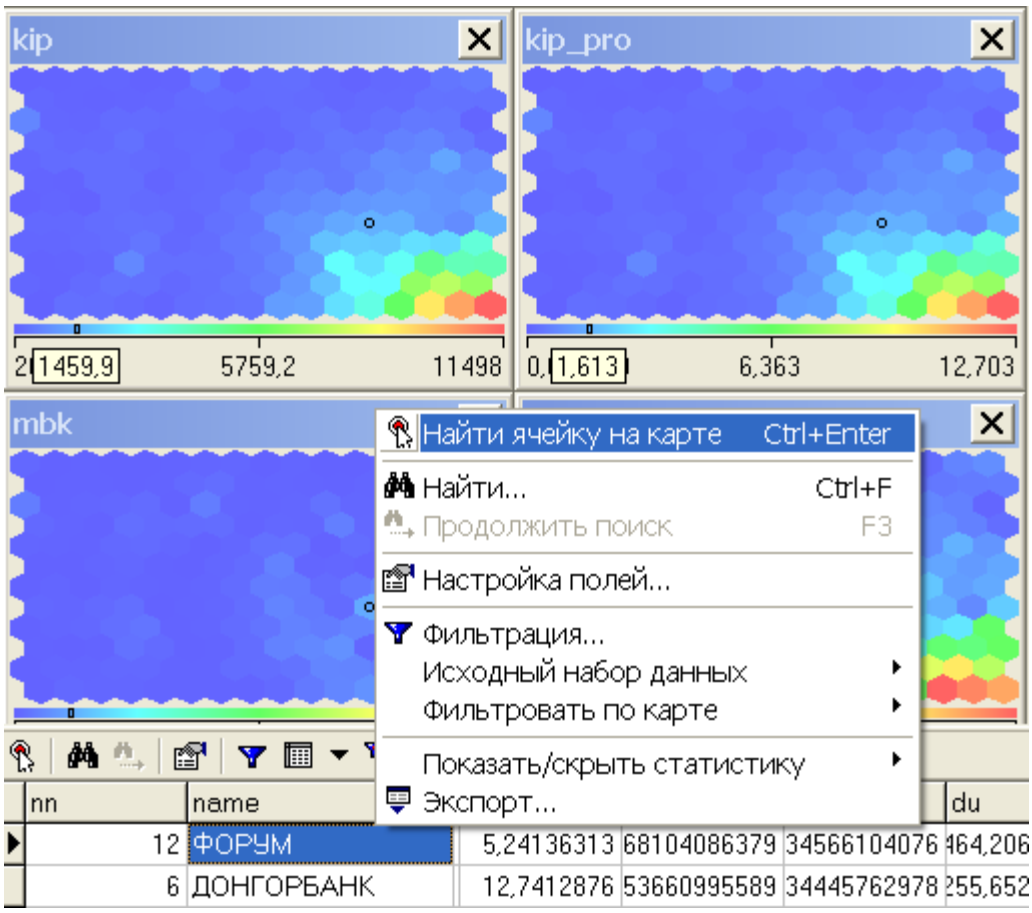

Рис. 12.10. Ячейка на карте

В результате применения самоорганизующихся карт многомерное пространство входных факторов было представлено в двухмерном виде, в котором его достаточно удобно анализировать.

Банки были классифицированы на 7 групп, для каждой из которых возможно определение конкретных характеристик, исходя из раскраски соответствующих показателей.

Выводы

В этой лекции мы подробно рассмотрели такую парадигму нейронных сетей как карты Кохонена. Основное отличие этих сетей от других моделей состоит в наглядности и удобстве использования. Эти сети позволяют упростить многомерную структуру, их можно считать одним из методов проецирования многомерного пространства в пространство с более низкой размерностью. Интенсивность цвета в определенной точке карты определяется данными, которые туда попали: ячейки с минимальными значениями изображаются темно-синим цветом, ячейки с максимальными значениями - красным.

Другое принципиальное отличие карт Кохонена от других моделей нейронных сетей - иной подход к обучению, а именно - неуправляемое или неконтролируемое обучение. Этот тип обучения позволяет данным обучающей выборки содержать значения только входных переменных. Сеть Кохонена учится понимать саму структуру данных и решает задачи кластеризации.

Методы кластерного анализа. Иерархические методы

С понятием кластеризации мы познакомились в первом разделе курса. В этой лекции мы опишем понятие "кластер" с математической точки зрения, а также рассмотрим методы решения задач кластеризации - методы кластерного анализа.

Термин кластерный анализ, впервые введенный Трионом (Tryon) в 1939 году, включает в себя более 100 различных алгоритмов.

В отличие от задач классификации, кластерный анализ не требует априорных предположений о наборе данных, не накладывает ограничения на представление исследуемых объектов, позволяет анализировать показатели различных типов данных (интервальным данным, частотам, бинарным данным). При этом необходимо помнить, что переменные должны измеряться в сравнимых шкалах.

Кластерный анализ позволяет сокращать размерность данных, делать ее наглядной.

Кластерный анализ может применяться к совокупностям временных рядов, здесь могут выделяться периоды схожести некоторых показателей и определяться группы временных рядов со схожей динамикой.

Кластерный анализ параллельно развивался в нескольких направлениях, таких как биология, психология, др., поэтому у большинства методов существует по два и более названий. Это существенно затрудняет работу при использовании кластерного анализа.

Задачи кластерного анализа можно объединить в следующие группы:

1. Разработка типологии или классификации.
2. Исследование полезных концептуальных схем группирования объектов.
3. Представление гипотез на основе исследования данных.
4. Проверка гипотез или исследований для определения, действительно ли типы (группы), выделенные тем или иным способом, присутствуют в имеющихся данных.

Как правило, при практическом использовании кластерного анализа одновременно решается несколько из указанных задач.

Рассмотрим пример процедуры кластерного анализа.

Допустим, мы имеем набор данных A, состоящий из 14-ти примеров, у которых имеется по два признака X и Y. Данные по ним приведены в таблице 13.1.

Таблица 13.1. Набор данных A

№ примера	признак X	признак Y
1	27	19
2	11	46

3	25	15
4	36	27
5	35	25
6	10	43
7	11	44
8	36	24
9	26	14
10	26	14
11	9	45
12	33	23
13	27	16
14	10	47

Данные в табличной форме не носят информативный характер. Представим переменные X и Y в виде диаграммы рассеивания, изображенной на рис. 13.1.

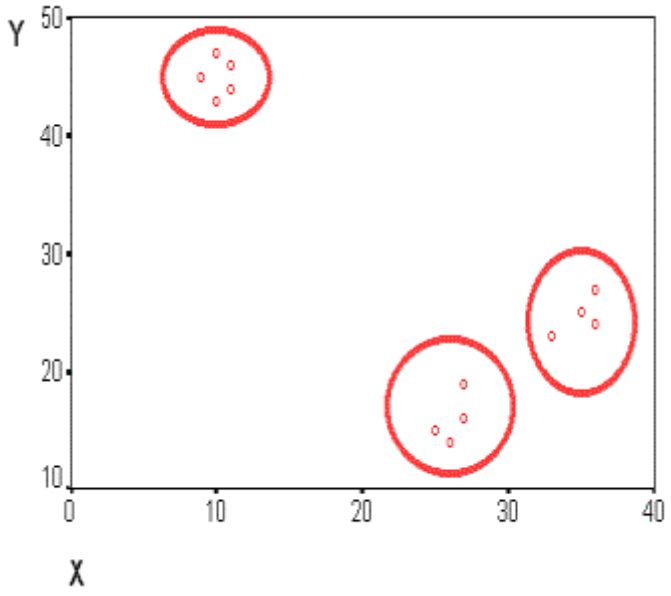

Рис. 13.1. Диаграмма рассеивания переменных X и Y

На рисунке мы видим несколько групп "похожих" примеров. Примеры (объекты), которые по значениям X и Y "похожи" друг на друга, принадлежат к одной группе (кластеру); объекты из разных кластеров не похожи друг на друга.

Критерием для определения схожести и различия кластеров является расстояние между точками на диаграмме рассеивания. Это сходство можно "измерить", оно равно расстоянию между точками на графике. Способов определения меры расстояния между кластерами, называемой еще мерой близости, существует несколько. Наиболее распространенный способ - вычисление евклидова расстояния между двумя точками i и j на плоскости, когда известны их координаты X и Y:

$$D_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2},$$

(13.1)

Примечание: чтобы узнать расстояние между двумя точками, надо взять разницу их координат по каждой оси, возвести ее в квадрат, сложить полученные значения для всех осей и извлечь квадратный корень из суммы.

Когда осей больше, чем две, расстояние рассчитывается таким образом: сумма квадратов разницы координат состоит из стольких слагаемых, сколько осей (измерений) присутствует в нашем пространстве. Например, если нам нужно найти расстояние между двумя точками в пространстве трех измерений (такая ситуация представлена на рис. 13.2), формула (13.1) приобретает вид:

$$D = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2},$$

(13.2)

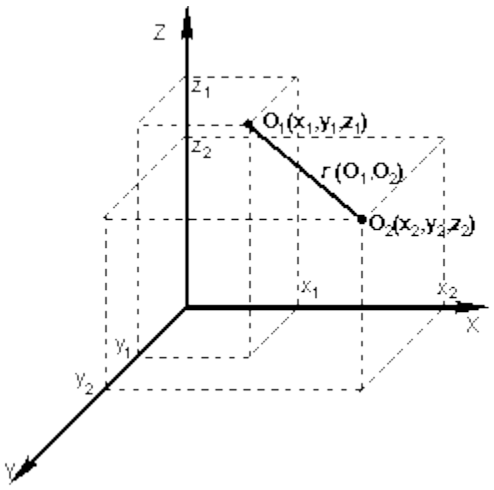

Рис. 13.2. Расстояние между двумя точками в пространстве трех измерений

Кластер имеет следующие математические характеристики: центр, радиус, среднеквадратическое отклонение, размер кластера.

Центр кластера - это среднее геометрическое место точек в пространстве переменных.

Радиус кластера - максимальное расстояние точек от центра кластера.

Как было отмечено в одной из предыдущих лекций, кластеры могут быть перекрывающимися. Такая ситуация возникает, когда обнаруживается перекрытие кластеров. В этом случае невозможно при помощи математических процедур однозначно отнести объект к одному из двух кластеров. Такие объекты называют спорными.

Спорный объект - это объект, который по мере сходства может быть отнесен к нескольким кластерам.

Размер кластера может быть определен либо по радиусу кластера, либо по среднеквадратичному отклонению объектов для этого кластера. Объект относится к кластеру, если расстояние от объекта до центра кластера меньше радиуса кластера. Если это условие выполняется для двух и более кластеров, объект является спорным.

Неоднозначность данной задачи может быть устранена экспертом или аналитиком.

Работа кластерного анализа опирается на два предположения. Первое предположение - рассматриваемые признаки объекта в принципе допускают желательное разбиение пула (совокупности) объектов на кластеры. В начале лекции мы уже упоминали о сравнимости шкал, это и есть второе предположение - правильность выбора масштаба или единиц измерения признаков.

Выбор масштаба в кластерном анализе имеет большое значение. Рассмотрим пример. Представим себе, что данные признака x в наборе данных A на два порядка больше данных признака y: значения переменной x находятся в диапазоне от 100 до 700, а значения переменной y - в диапазоне от 0 до 1.

Тогда, при расчете величины расстояния между точками, отражающими положение объектов в пространстве их свойств, переменная, имеющая большие значения, т.е. переменная x, будет практически полностью доминировать над переменной с малыми значениями, т.е. переменной y. Таким образом из-за неоднородности единиц измерения признаков становится невозможно корректно рассчитать расстояния между точками.

Эта проблема решается при помощи предварительной стандартизации переменных. Стандартизация (standardization) или нормирование (normalization) приводит значения всех преобразованных переменных к единому диапазону значений путем выражения через отношение этих значений к некой величине, отражающей определенные свойства конкретного признака. Существуют различные способы нормирования исходных данных.

Два наиболее распространенных способа:

- деление исходных данных на среднеквадратичное отклонение соответствующих переменных;
- вычисление Z-вклада или стандартизованного вклада.

Наряду со стандартизацией переменных, существует вариант придания каждой из них определенного коэффициента важности, или веса, который бы отражал значимость соответствующей переменной. В качестве весов могут выступать экспертные оценки, полученные в ходе опроса экспертов - специалистов предметной области. Полученные произведения нормированных переменных на соответствующие веса позволяют получать

расстояния между точками в многомерном пространстве с учетом неодинакового веса переменных.

В ходе экспериментов возможно сравнение результатов, полученных с учетом экспертных оценок и без них, и выбор лучшего из них.

Методы кластерного анализа

Методы кластерного анализа можно разделить на две группы:

- иерархические;
- неиерархические.

Каждая из групп включает множество подходов и алгоритмов.

Используя различные методы кластерного анализа, аналитик может получить различные решения для одних и тех же данных. Это считается нормальным явлением.

Рассмотрим иерархические и неиерархические методы подробно.

Иерархические методы кластерного анализа

Суть иерархической кластеризации состоит в последовательном объединении меньших кластеров в большие или разделении больших кластеров на меньшие.

Иерархические агломеративные методы (Agglomerative Nesting, AGNES)

Эта группа методов характеризуется последовательным объединением исходных элементов и соответствующим уменьшением числа кластеров.

В начале работы алгоритма все объекты являются отдельными кластерами. На первом шаге наиболее похожие объекты объединяются в кластер. На последующих шагах объединение продолжается до тех пор, пока все объекты не будут составлять один кластер.

Иерархические дивизимные (делимые) методы (DIvisive ANAlysis, DIANA)

Эти методы являются логической противоположностью агломеративным методам. В начале работы алгоритма все объекты принадлежат одному кластеру, который на последующих шагах делится на меньшие кластеры, в результате образуется последовательность расщепляющих групп.

Принцип работы описанных выше групп методов в виде дендрограммы показан на рис. 13.3.

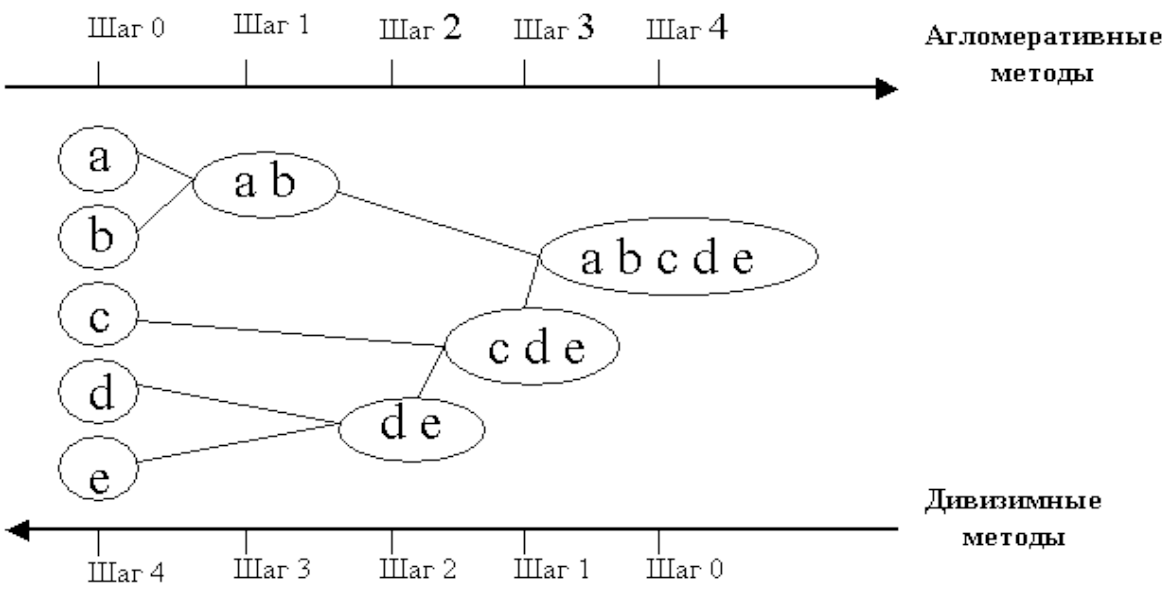

Рис. 13.3. Дендрограмма агломеративных и дивизимных методов

Программная реализация алгоритмов кластерного анализа широко представлена в различных инструментах Data Mining, которые позволяют решать задачи достаточно большой размерности. Например, агломеративные методы реализованы в пакете SPSS, дивизимные методы - в пакете Statgraf.

Иерархические методы кластеризации различаются правилами построения кластеров. В качестве правил выступают критерии, которые используются при решении вопроса о "схожести" объектов при их объединении в группу (агломеративные методы) либо разделения на группы (дивизимные методы).

Иерархические методы кластерного анализа используются при небольших объемах наборов данных.

Преимуществом иерархических методов кластеризации является их наглядность.

Иерархические алгоритмы связаны с построением дендрограмм (от греческого dendron - "дерево"), которые являются результатом иерархического кластерного анализа. Дендрограмма описывает близость отдельных точек и кластеров друг к другу, представляет в графическом виде последовательность объединения (разделения) кластеров.

Дендрограмма (dendrogram) - древовидная диаграмма, содержащая n уровней, каждый из которых соответствует одному из шагов процесса последовательного укрупнения кластеров.

Дендрограмму также называют древовидной схемой, деревом объединения кластеров, деревом иерархической структуры.

Дендрограмма представляет собой вложенную группировку объектов, которая изменяется на различных уровнях иерархии.

Существует много способов построения дендограмм. В дендограмме объекты могут располагаться вертикально или горизонтально. Пример вертикальной дендрограммы приведен на рис. 13.4.

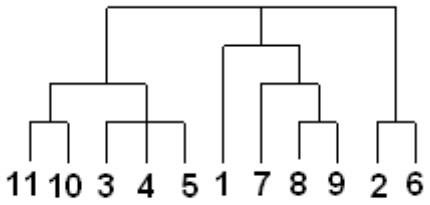

Рис. 13.4. Пример дендрограммы

Числа 11, 10, 3 и т.д. соответствуют номерам объектов или наблюдений исходной выборки. Мы видим, что на первом шаге каждое наблюдение представляет один кластер (вертикальная линия), на втором шаге наблюдаем объединение таких наблюдений: 11 и 10; 3, 4 и 5; 8 и 9; 2 и 6. На втором шаге продолжается объединение в кластеры: наблюдения 11, 10, 3, 4, 5 и 7, 8, 9. Данный процесс продолжается до тех пор, пока все наблюдения не объединятся в один кластер.

Меры сходства

Для вычисления расстояния между объектами используются различные меры сходства (меры подобия), называемые также метриками или функциями расстояний. В начале лекции мы рассмотрели евклидово расстояние, это наиболее популярная мера сходства.

Квадрат евклидова расстояния.

Для придания больших весов более отдаленным друг от друга объектам можем воспользоваться квадратом евклидова расстояния путем возведения в квадрат стандартного евклидова расстояния.

Манхэттенское расстояние (расстояние городских кварталов), также называемое "хэмминговым" или "сити-блок" расстоянием.

Это расстояние рассчитывается как среднее разностей по координатам. В большинстве случаев эта мера расстояния приводит к результатам, подобным расчетам расстояния евклида. Однако, для этой меры влияние отдельных выбросов меньше, чем при использовании евклидова расстояния, поскольку здесь координаты не возводятся в квадрат.

Расстояние Чебышева. Это расстояние стоит использовать, когда необходимо определить два объекта как "различные", если они отличаются по какому-то одному измерению.

Процент несогласия. Это расстояние вычисляется, если данные являются категориальными.

Методы объединения или связи

Когда каждый объект представляет собой отдельный кластер, расстояния между этими объектами определяются выбранной мерой. Возникает следующий вопрос - как определить расстояния между кластерами? Существуют различные правила, называемые методами объединения или связи для двух кластеров.

Метод ближнего соседа или **одиночная связь**. Здесь расстояние между двумя кластерами определяется расстоянием между двумя наиболее близкими объектами (ближайшими соседями) в различных кластерах. Этот метод позволяет выделять кластеры сколь угодно сложной формы при условии, что различные части таких кластеров соединены цепочками близких друг к другу элементов. В результате работы этого метода кластеры представляются длинными "цепочками" или "волокнистыми" кластерами, "сцепленными вместе" только отдельными элементами, которые случайно оказались ближе остальных друг к другу.

Метод наиболее удаленных соседей или полная связь. Здесь расстояния между кластерами определяются наибольшим расстоянием между любыми двумя объектами в различных кластерах (т.е. "наиболее удаленными соседями"). Метод хорошо использовать, когда объекты действительно происходят из различных "рощ". Если же кластеры имеют в некотором роде удлиненную форму или их естественный тип является "цепочечным", то этот метод не следует использовать.

Метод Варда (Ward's method). В качестве расстояния между кластерами берется прирост суммы квадратов расстояний объектов до центров кластеров, получаемый в результате их объединения (Ward, 1963). В отличие от других методов кластерного анализа для оценки расстояний между кластерами, здесь используются методы дисперсионного анализа. На каждом шаге алгоритма объединяются такие два кластера, которые приводят к минимальному увеличению целевой функции, т.е. внутригрупповой суммы квадратов. Этот метод направлен на объединение близко расположенных кластеров и "стремится" создавать кластеры малого размера.

Метод невзвешенного попарного среднего (метод невзвешенного попарного арифметического среднего - unweighted pair-group method using arithmetic averages, UPGMA (Sneath, Sokal, 1973)).

В качестве расстояния между двумя кластерами берется среднее расстояние между всеми парами объектов в них. Этот метод следует использовать, если объекты действительно происходят из различных "рощ", в случаях присутствия кластеров "цепочного" типа, при предположении неравных размеров кластеров.

Метод взвешенного попарного среднего (метод взвешенного попарного арифметического среднего - weighted pair-group method using arithmetic averages, WPGM A (Sneath, Sokal, 1973)). Этот метод похож на метод невзвешенного попарного среднего, разница состоит лишь в том, что здесь в качестве весового коэффициента используется размер кластера (число объектов, содержащихся в кластере).

Этот метод рекомендуется использовать именно при наличии предположения о кластерах разных размеров.

Невзвешенный центроидный метод (метод невзвешенного попарного центроидного усреднения - unweighted pair-group method using the centroid average (Sneath and Sokal, 1973)).

В качестве расстояния между двумя кластерами в этом методе берется расстояние между их центрами тяжести.

Взвешенный центроидный метод (метод взвешенного попарного центроидного усреднения - weighted pair-group method using the centroid average, WPGMC (Sneath, Sokal 1973)). Этот метод похож на предыдущий, разница состоит в том, что для учета разницы между размерами кластеров (числе объектов в них), используются веса. Этот метод предпочтительно использовать в случаях, если имеются предположения относительно существенных отличий в размерах кластеров.

Иерархический кластерный анализ в SPSS

Рассмотрим процедуру иерархического кластерного анализа в пакете SPSS (SPSS). Процедура иерархического кластерного анализа в SPSS предусматривает группировку как объектов (строк матрицы данных), так и переменных (столбцов) [54]. Можно считать, что в последнем случае роль объектов играют переменные, а роль переменных - столбцы.

В этом методе реализуется иерархический агломеративный алгоритм, смысл которого заключается в следующем. Перед началом кластеризации все объекты считаются отдельными кластерами, в ходе алгоритма они объединяются. Вначале выбирается пара ближайших кластеров, которые объединяются в один кластер. В результате количество кластеров становится равным N-1. Процедура повторяется, пока все классы не объединятся. На любом этапе объединение можно прервать, получив нужное число кластеров. Таким образом, результат работы алгоритма агрегирования зависит от **способов вычисления расстояния между объектами и определения близости между кластерами**.

Для определения расстояния между парой кластеров могут быть сформулированы различные подходы. С учетом этого в SPSS предусмотрены следующие методы:

- Среднее расстояние между кластерами (Between-groups linkage), устанавливается по умолчанию.
- Среднее расстояние между всеми объектами пары кластеров с учетом расстояний внутри кластеров (Within-groups linkage).
- Расстояние между ближайшими соседями - ближайшими объектами кластеров (Nearest neighbor).
- Расстояние между самыми далекими соседями (Furthest neighbor).
- Расстояние между центрами кластеров (Centroid clustering) или центроидный метод. Недостатком этого метода является то, что центр объединенного кластера вычисляется как среднее центров объединяемых кластеров, без учета их объема.
- Метод медиан - тот же центроидный метод, но центр объединенного кластера вычисляется как среднее всех объектов (Median clustering).
- Метод Варда.

Пример иерархического кластерного анализа

Порядок агломерации (протокол объединения кластеров) представленных ранее данных приведен в таблице 13.2. В протоколе указаны такие позиции:

- Stage - стадии объединения (шаг);
- Cluster Combined - объединяемые кластеры (после объединения кластер принимает минимальный номер из номеров объединяемых кластеров);
- Coefficients - коэффициенты.

Таблица 13.2. Порядок агломерации

	Cluster Combined		Coefficients
	Cluster 1	Cluster 2	
1	9	10	,000
2	2	14	1,461E-02
3	3	9	1,461E-02
4	5	8	1,461E-02
5	6	7	1,461E-02
6	3	13	3,490E-02
7	2	11	3,651E-02
8	4	5	4,144E-02
9	2	6	5,118E-02
10	4	12	,105
11	1	3	,120
12	1	4	1,217
13	1	2	7,516

Так, в колонке Cluster Combined можно увидеть порядок объединения в кластеры: на первом шаге были объединены наблюдения 9 и 10, они образовывают кластер под номером 9, кластер 10 в обзорной таблице больше не появляется. На следующем шаге происходит объединение кластеров 2 и 14, далее 3 и 9, и т.д.

В колонке Coefficients приведено количество кластеров, которое следовало бы считать оптимальным; под значением этого показателя подразумевается расстояние между двумя кластерами, определенное на основании выбранной меры расстояния. В нашем случае это квадрат евклидова расстояния, определенный с использованием стандартизированных

значений. Процедура стандартизации используется для исключения вероятности того, что классификацию будут определять переменные, имеющие наибольший разброс значений. В SPSS применяются следующие виды стандартизации:

- Z-шкалы (Z-Scores). Из значений переменных вычитается их среднее, и эти значения делятся на стандартное отклонение.
- Разброс от -1 до 1. Линейным преобразованием переменных добиваются разброса значений от -1 до 1.
- Разброс от 0 до 1. Линейным преобразованием переменных добиваются разброса значений от 0 до 1.
- Максимум 1. Значения переменных делятся на их максимум.
- Среднее 1. Значения переменных делятся на их среднее.
- Стандартное отклонение 1. Значения переменных делятся на стандартное отклонение.

Кроме того, возможны преобразования самих расстояний, в частности, можно расстояния заменить их абсолютными значениями, это актуально для коэффициентов корреляции. Можно также все расстояния преобразовать так, чтобы они изменялись от 0 до 1.

Определение количества кластеров

Существует проблема определения числа кластеров. Иногда можно априорно определить это число. Однако в большинстве случаев число кластеров определяется в процессе агломерации/разделения множества объектов.

Процессу группировки объектов в иерархическом кластерном анализе соответствует постепенное возрастание коэффициента, называемого критерием E. Скачкообразное увеличение значения критерия E можно определить как характеристику числа кластеров, которые действительно существуют в исследуемом наборе данных. Таким образом, этот способ сводится к определению скачкообразного увеличения некоторого коэффициента, который характеризует переход от сильно связанного к слабо связанному состоянию объектов.

В таблице 13.2 мы видим, что значение поля Coefficients увеличивается скачкообразно, следовательно, объединение в кластеры следует остановить, иначе будет происходить объединение кластеров, находящихся на относительно большом расстоянии друг от друга.

В нашем примере это скачок с 1,217 до 7,516. Оптимальным считается количество кластеров, равное разности количества наблюдений (14) и количества шагов до скачкообразного увеличения коэффициента (12).

Следовательно, после создания двух кластеров объединений больше производить не следует, хотя визуально мы ожидали появления трех кластеров.

Агрегирование данных может быть представлено графически в виде дендрограммы. Она определяет объединенные кластеры и значения коэффициентов на каждом шаге агломерации (отображены значения коэффициентов, приведенные к шкале от 0 до 25).

Дендрограмма для нашего примера приведена на рис. 13.5. Разрез дерева агрегирования вертикальной чертой дал нам два кластера, состоящих из 9 и 5 объектов.

На верхней линии по горизонтали отмечены номера шагов алгоритма, всего алгоритму потребовалось 25 шагов для объединения всех объектов в один кластер.

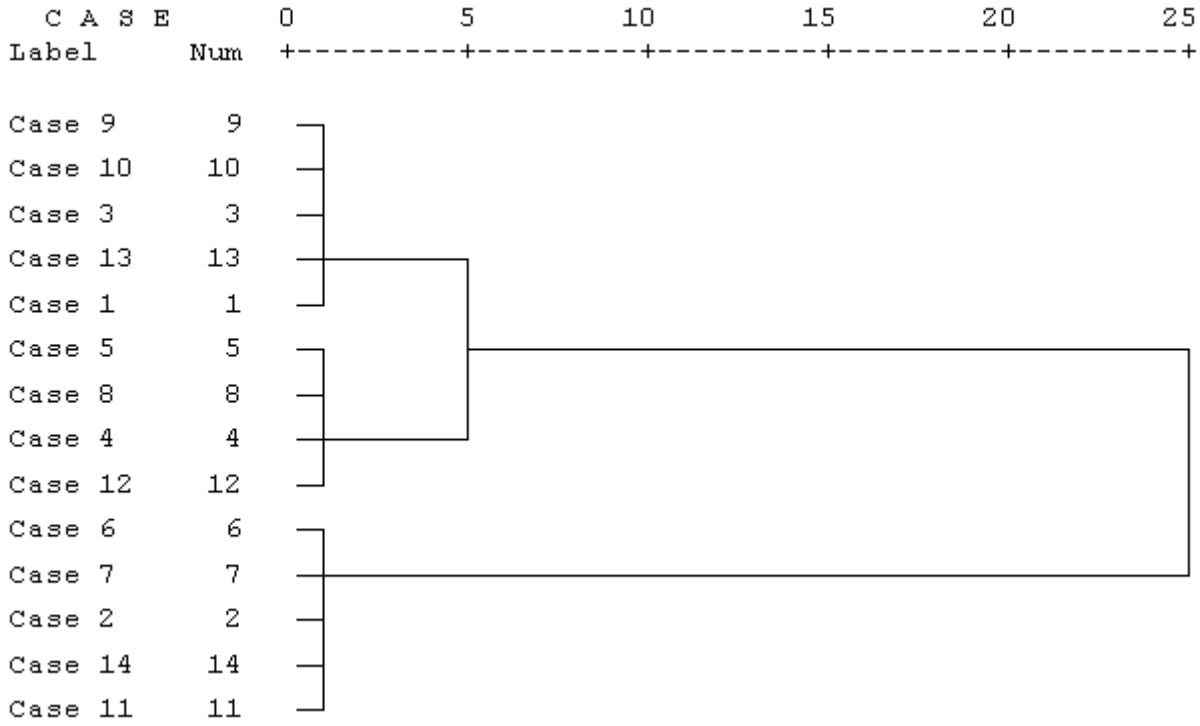

Рис. 13.5. Дендрограмма процесса слияния

Методы кластерного анализа. Итеративные методы.

При большом количестве наблюдений иерархические методы кластерного анализа не пригодны. В таких случаях используют неиерархические методы, основанные на разделении, которые представляют собой итеративные методы дробления исходной совокупности. В процессе деления новые кластеры формируются до тех пор, пока не будет выполнено правило остановки.

Такая неиерархическая кластеризация состоит в разделении набора данных на определенное количество отдельных кластеров. Существует два подхода. Первый заключается в определении границ кластеров как наиболее плотных участков в многомерном пространстве исходных данных, т.е. определение кластера там, где имеется большое "сгущение точек". Второй подход заключается в минимизации меры различия объектов

Алгоритм k-средних (k-means)

Наиболее распространен среди неиерархических методов алгоритм k-средних, также называемый **быстрым кластерным анализом**. Полное описание алгоритма можно найти в работе Хартигана и Вонга (Hartigan and Wong, 1978). В отличие от иерархических методов, которые не требуют предварительных предположений относительно числа кластеров, для возможности использования этого метода необходимо иметь гипотезу о наиболее вероятном количестве кластеров.

Алгоритм k-средних строит k кластеров, расположенных на возможно больших расстояниях друг от друга. Основной тип задач, которые решает алгоритм k-средних, - наличие предположений (гипотез) относительно числа кластеров, при этом они должны быть различны настолько, насколько это возможно. Выбор числа k может базироваться на результатах предшествующих исследований, теоретических соображениях или интуиции.

Общая идея алгоритма: заданное фиксированное число k кластеров наблюдения сопоставляются кластерам так, что средние в кластере (для всех переменных) максимально возможно отличаются друг от друга.

Описание алгоритма

1. Первоначальное распределение объектов по кластерам.

   Выбирается число k, и на первом шаге эти точки считаются "центрами" кластеров. Каждому кластеру соответствует один центр.

   Выбор начальных центроидов может осуществляться следующим образом:

   - выбор k-наблюдений для максимизации начального расстояния;
   - случайный выбор k-наблюдений;
   - выбор первых k-наблюдений.

   В результате каждый объект назначен определенному кластеру.

2. Итеративный процесс.

Вычисляются центры кластеров, которыми затем и далее считаются покоординатные средние кластеров. Объекты опять перераспределяются.

Процесс вычисления центров и перераспределения объектов продолжается до тех пор, пока не выполнено одно из условий:

- кластерные центры стабилизировались, т.е. все наблюдения принадлежат кластеру, которому принадлежали до текущей итерации;
- число итераций равно максимальному числу итераций.

На рис. 14.1 приведен пример работы алгоритма k-средних для k, равного двум.

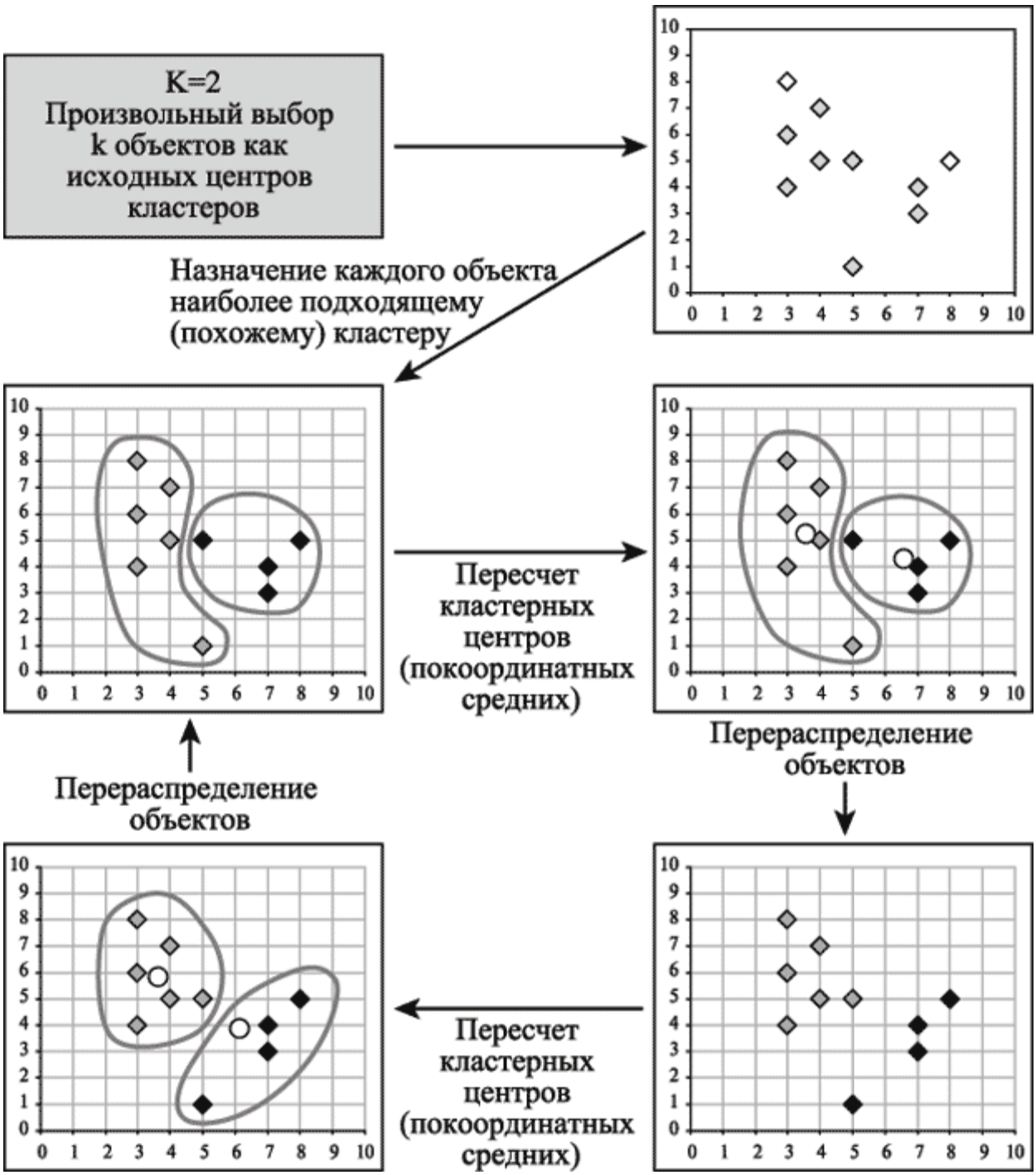

Рис. 14.1. Пример работы алгоритма k-средних (k=2)

Выбор числа кластеров является сложным вопросом. Если нет предположений относительно этого числа, рекомендуют создать 2 кластера, затем 3, 4, 5 и т.д., сравнивая полученные результаты.

Проверка качества кластеризации

После получений результатов кластерного анализа методом k-средних следует проверить правильность кластеризации (т.е. оценить, насколько кластеры отличаются друг от друга). Для этого рассчитываются средние значения для каждого кластера. При хорошей

кластеризации должны быть получены сильно отличающиеся средние для всех измерений или хотя бы большей их части.

Достоинства алгоритма k-средних:

- простота использования;
- быстрота использования;
- понятность и прозрачность алгоритма.

Недостатки алгоритма k-средних:

- алгоритм слишком чувствителен к выбросам, которые могут искажать среднее. Возможным решением этой проблемы является использование модификации алгоритма - алгоритм k-медианы;
- алгоритм может медленно работать на больших базах данных. Возможным решением данной проблемы является использование выборки данных.

Алгоритм PAM (partitioning around Medoids)

PAM является модификацией алгоритма k-средних, алгоритмом k-медианы (k-medoids).

Алгоритм менее чувствителен к шумам и выбросам данных, чем алгоритм k-means, поскольку медиана меньше подвержена влияниям выбросов.

PAM эффективен для небольших баз данных, но его не следует использовать для больших наборов данных.

Предварительное сокращение размерности

Рассмотрим пример. Есть база данных клиентов фирмы, которых следует разбить на однородные группы. Каждый клиент описывается при помощи 25 переменных. Использование такого большого числа переменных приводит к выделению кластеров нечеткой структуры. В результате аналитику достаточно сложно интерпретировать полученные кластеры.

Более понятные и прозрачные результаты кластеризации могут быть получены, если вместо множества исходных переменных использовать некие обобщенные переменные или критерии, содержащие в сжатом виде информацию о связях между переменными. Т.е. возникает задача понижения размерности данных. Она может решаться при помощи различных методов; один из наиболее распространенных - факторный анализ. Остановимся на нем более подробно.

Факторный анализ

Факторный анализ - это метод, применяемый для изучения взаимосвязей между значениями переменных.

Вообще, факторный анализ преследует две цели:

- сокращение числа переменных;

- классификацию переменных - определение структуры взаимосвязей между переменными.

Соответственно, факторный анализ может использоваться для решения задач сокращения размерности данных или для решения задач классификации.

Критерии или главные факторы, выделенные в результате факторного анализа, содержат в сжатом виде информацию о существующих связях между переменными. Эта информация позволяет получить лучшие результаты кластеризации и лучше объяснить семантику кластеров. Самим факторам может быть сообщен определенный смысл.

При помощи факторного анализа большое число переменных сводится к меньшему числу независимых влияющих величин, которые называются факторами.

Фактор в "сжатом" виде содержит информацию о нескольких переменных. В один фактор объединяются переменные, которые сильно коррелируют между собой. В результате факторного анализа отыскиваются такие комплексные факторы, которые как можно более полно объясняют связи между рассматриваемыми переменными.

На первом шаге факторного анализа осуществляется стандартизация значений переменных, необходимость которой была рассмотрена в предыдущей лекции.

Факторный анализ опирается на гипотезу о том, что анализируемые переменные являются косвенными проявлениями сравнительно небольшого числа неких скрытых факторов.

Факторный анализ - это совокупность методов, ориентированных на выявление и анализ скрытых зависимостей между наблюдаемыми переменными. Скрытые зависимости также называют латентными.

Один из методов факторного анализа - метод главных компонент - основан на предположении о независимости факторов друг от друга.

Итеративная кластеризация в SPSS

Обычно в статистических пакетах реализован широкий арсенал методов, что позволяет сначала провести сокращение размерности набора данных (например, при помощи факторного анализа), а затем уже собственно кластеризацию (например, методом быстрого кластерного анализа). Рассмотрим этот вариант проведения кластеризации в пакете SPSS.

Для сокращения размерности исходных данных воспользуемся факторным анализом. Для этого выберем в меню: Analyze (Анализ)/Data Reduction (Преобразование данных)/Factor (Факторный анализ):

При помощи кнопки Extraction:(Отбор) следует выбрать метод отбора. Мы оставим выбранный по умолчанию анализ главных компонентов, который упоминался выше. Также следует выбрать метод вращения - выберем один из наиболее популярных - метод варимакса. Для сохранения значений факторов в виде переменных в закладке "Значения" необходимо поставить отметку "Save as variables" (Сохранить как переменные).

В результате этой процедуры пользователь получает отчет "Объясненная суммарная дисперсия", по которой видно число отобранных факторов - это те компоненты, собственные значения которых превосходят единицу.

Полученные значения факторов, которым обычно присваиваются названия fact1_1, fact1_2 и т.д., используем для проведения кластерного анализа методом k-средних. Для проведения быстрого кластерного анализа выберем в меню:

Analyze (Анализ)/Classify(Классифицировать)/K-Means Cluster: (Кластерный анализ методом k-средних).

В диалоговом окне K Means Cluster Analysis (Кластерный анализ методом k-средних) необходимо поместить факторные переменные fact1_1, fact1_2 и т.д. в поле тестируемых переменных. Здесь же необходимо указать количество кластеров и количество итераций.

В результате этой процедуры получаем отчет с выводом значений центров сформированных кластеров, количестве наблюдений в каждом кластере, а также с дополнительной информацией, заданной пользователем.

Таким образом, алгоритм k-средних делит совокупность исходных данных на заданное количество кластеров. Для возможности визуализации полученных результатов следует воспользоваться одним из графиков, например, диаграммой рассеивания. Однако традиционная визуализация возможна для ограниченного количества измерений, ибо, как известно, человек может воспринимать только трехмерное пространство. Поэтому, если мы анализируем более трех переменных, следует использовать специальные многомерные методы представления информации, о них будет рассказано в одной из последующих лекций курса.

Итеративные методы кластеризации различаются выбором следующих параметров:

- начальной точки;
- правилом формирования новых кластеров;
- правилом остановки.

Выбор метода кластеризации зависит от количества данных и от того, есть ли необходимость работать одновременно с несколькими типами данных.

В пакете SPSS, например, при необходимости работы как с количественными (например, доход), так и с категориальными (например, семейное положение) переменными, а также если объем данных достаточно велик, используется метод Двухэтапного кластерного анализа, который представляет собой масштабируемую процедуру кластерного анализа, позволяющую работать с данными различных типов.

Для этого на первом этапе работы записи предварительно кластеризуются в большое количество суб-кластеров. На втором этапе полученные суб-кластеры группируются в необходимое количество. Если это количество неизвестно, процедура сама автоматически определяет его. При помощи этой процедуры банковский работник может, например,выделять группы людей, одновременно используя такие показатели как возраст, пол и уровень дохода. Полученные результаты позволяют определить клиентов, входящих в группы риска невозврата кредита.

Процесс кластерного анализа. Рекомендуемые этапы

В общем случае все этапы кластерного анализа взаимосвязаны, и решения, принятые на одном из них, определяют действия на последующих этапах.

Аналитику следует решить, использовать ли все наблюдения либо же исключить некоторые данные или выборки из набора данных.

Выбор метрики и метода стандартизации исходных данных.

Определение количества кластеров (для итеративного кластерного анализа).

Определение метода кластеризации (правила объединения или связи).

По мнению многих специалистов, выбор метода кластеризации является решающим при определении формы и специфики кластеров.

Анализ результатов кластеризации. Этот этап подразумевает решение таких вопросов: не является ли полученное разбиение на кластеры случайным; является ли разбиение надежным и стабильным на подвыборках данных; существует ли взаимосвязь между результатами кластеризации и переменными, которые не участвовали в процессе кластеризации; можно ли интерпретировать полученные результаты кластеризации.

Проверка результатов кластеризации. Результаты кластеризации также должны быть проверены формальными и неформальными методами. Формальные методы зависят от того метода, который использовался для кластеризации. Неформальные включают следующие процедуры проверки качества кластеризации:

- анализ результатов кластеризации, полученных на определенных выборках набора данных;
- кросс-проверка;
- проведение кластеризации при изменении порядка наблюдений в наборе данных;
- проведение кластеризации при удалении некоторых наблюдений;
- проведение кластеризации на небольших выборках.

Один из вариантов проверки качества кластеризации - использование нескольких методов и сравнение полученных результатов. Отсутствие подобия не будет означать некорректность результатов, но присутствие похожих групп считается признаком качественной кластеризации.

Сложности и проблемы, которые могут возникнуть при применении кластерного анализа

Как и любые другие методы, методы кластерного анализа имеют определенные слабые стороны, т.е. некоторые сложности, проблемы и ограничения.

При проведении кластерного анализа следует учитывать, что результаты кластеризации зависят от критериев разбиения совокупности исходных данных. При понижении размерности данных могут возникнуть определенные искажения, за счет обобщений могут потеряться некоторые индивидуальные характеристики объектов.

Существует ряд сложностей, которые следует продумать перед проведением кластеризации.

- Сложность выбора характеристик, на основе которых проводится кластеризация. Необдуманный выбор приводит к неадекватному разбиению на кластеры и, как следствие, - к неверному решению задачи.
- Сложность выбора метода кластеризации. Этот выбор требует неплохого знания методов и предпосылок их использования. Чтобы проверить эффективность конкретного метода в определенной предметной области, целесообразно применить следующую процедуру: рассматривают несколько априори различных между собой групп и перемешивают их представителей между собой случайным образом. Далее проводится кластеризация для восстановления исходного разбиения на кластеры. Доля совпадений объектов в выявленных и исходных группах является показателем эффективности работы метода.
- Проблема выбора числа кластеров. Если нет никаких сведений относительно возможного числа кластеров, необходимо провести ряд экспериментов и, в результате перебора различного числа кластеров, выбрать оптимальное их число.
- Проблема интерпретации результатов кластеризации. Форма кластеров в большинстве случаев определяется выбором метода объединения. Однако следует учитывать, что конкретные методы стремятся создавать кластеры определенных форм, даже если в исследуемом наборе данных кластеров на самом деле нет.

Сравнительный анализ иерархических и неиерархических методов кластеризации

Перед проведением кластеризации у аналитика может возникнуть вопрос, какой группе методов кластерного анализа отдать предпочтение. Выбирая между иерархическими и неиерархическими методами, необходимо учитывать следующие их особенности.

Неиерархические методы выявляют более высокую устойчивость по отношению к шумам и выбросам, некорректному выбору метрики, включению незначимых переменных в набор, участвующий в кластеризации. Ценой, которую приходится платить за эти достоинства метода, является слово "априори". Аналитик должен заранее определить количество кластеров, количество итераций или правило остановки, а также некоторые другие параметры кластеризации. Это особенно сложно начинающим специалистам.

Если нет предположений относительно числа кластеров, рекомендуют использовать иерархические алгоритмы. Однако если объем выборки не позволяет это сделать, возможный путь - проведение ряда экспериментов с различным количеством кластеров, например, начать разбиение совокупности данных с двух групп и, постепенно увеличивая их количество, сравнивать результаты. За счет такого "варьирования" результатов достигается достаточно большая гибкость кластеризации.

Иерархические методы, в отличие от неиерархических, отказываются от определения числа кластеров, а строят полное дерево вложенных кластеров.

Сложности иерархических методов кластеризации: ограничение объема набора данных; выбор меры близости; негибкость полученных классификаций.

Преимущество этой группы методов в сравнении с неиерархическими методами - их наглядность и возможность получить детальное представление о структуре данных.

При использовании иерархических методов существует возможность достаточно легко идентифицировать выбросы в наборе данных и, в результате, повысить качество данных. Эта процедура лежит в основе двухшагового алгоритма кластеризации. Такой набор данных в дальнейшем может быть использован для проведения неиерархической кластеризации.

Существует еще одни аспект, о котором уже упоминалось в этой лекции. Это вопрос кластеризации всей совокупности данных или же ее выборки. Названный аспект существенен для обеих рассматриваемых групп методов, однако он более критичен для иерархических методов. Иерархические методы не могут работать с большими наборами данных, а использование некоторой выборки, т.е. части данных, могло бы позволить применять эти методы.

Результаты кластеризации могут не иметь достаточного статистического обоснования. С другой стороны, при решении задач кластеризации допустима нестатистическая интерпретация полученных результатов, а также достаточно большое разнообразие вариантов понятия кластера. Такая нестатистическая интерпретация дает возможность аналитику получить удовлетворяющие его результаты кластеризации, что при использовании других методов часто бывает затруднительным.

Новые алгоритмы и некоторые модификации алгоритмов кластерного анализа

Методы, которые мы рассмотрели в этой и предыдущей лекциях, являются "классикой" кластерного анализа. До последнего времени основным критерием, по которому оценивался алгоритм кластеризации, было качество кластеризации: полагалось, чтобы весь набор данных умещался в оперативной памяти.

Однако сейчас, в связи с появлением сверхбольших баз данных, появились новые требования, которым должен удовлетворять алгоритм кластеризации. Основное из них, как уже упоминалось в предыдущих лекциях, - это масштабируемость алгоритма.

Отметим также другие свойства, которым должен удовлетворять алгоритм кластеризации: независимость результатов от порядка входных данных; независимость параметров алгоритма от входных данных.

В последнее время ведутся активные разработки новых алгоритмов кластеризации, способных обрабатывать сверхбольшие базы данных. В них основное внимание уделяется масштабируемости. К таким алгоритмам относятся обобщенное представление кластеров (summarized cluster representation), а также выборка и использование структур данных, поддерживаемых нижележащими СУБД [33].

Разработаны алгоритмы, в которых методы иерархической кластеризации интегрированы с другими методами. К таким алгоритмам относятся: BIRCH, CURE, CHAMELEON, ROCK.

Алгоритм BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies)

Алгоритм предложен Тьян Зангом и его коллегами [55].

Благодаря обобщенным представлениям кластеров, скорость кластеризации увеличивается, алгоритм при этом обладает большим масштабированием.

В этом алгоритме реализован двухэтапный процесс кластеризации.

В ходе первого этапа формируется предварительный набор кластеров. На втором этапе к выявленным кластерам применяются другие алгоритмы кластеризации - пригодные для работы в оперативной памяти.

В [33] приведена следующая аналогия, описывающая этот алгоритм. Если каждый элемент данных представить себе как бусину, лежащую на поверхности стола, то кластеры бусин можно "заменить" теннисными шариками и перейти к более детальному изучению кластеров теннисных шариков. Число бусин может оказаться достаточно велико, однако диаметр теннисных шариков можно подобрать таким образом, чтобы на втором этапе можно было, применив традиционные алгоритмы кластеризации, определить действительную сложную форму кластеров.

Алгоритм WaveCluster

WaveCluster представляет собой алгоритм кластеризации на основе волновых преобразований [56]. В начале работы алгоритма данные обобщаются путем наложения на пространство данных многомерной решетки. На дальнейших шагах алгоритма анализируются не отдельные точки, а обобщенные характеристики точек, попавших в одну ячейку решетки. В результате такого обобщения необходимая информация умещается в оперативной памяти. На последующих шагах для определения кластеров алгоритм применяет волновое преобразование к обобщенным данным.

Главные особенности WaveCluster:

1. сложность реализации;
2. алгоритм может обнаруживать кластеры произвольных форм;
3. алгоритм не чувствителен к шумам;
4. алгоритм применим только к данным низкой размерности.

Алгоритм CLARA (Clustering LARge Applications)

Алгоритм CLARA был разработан Kaufmann и Rousseeuw в 1990 году для кластеризации данных в больших базах данных. Данный алгоритм строится в статистических аналитических пакетах, например, таких как S+.

Изложим кратко суть алгоритма. Алгоритм CLARA извлекает множество образцов из базы данных. Кластеризация применяется к каждому из образцов, на выходе алгоритма предлагается лучшая кластеризация.

Для больших баз данных этот алгоритм эффективнее, чем алгоритм PAM. Эффективность алгоритма зависит от выбранного в качестве образца набора данных. Хорошая кластеризация на выбранном наборе может не дать хорошую кластеризацию на всем множестве данных.

Алгоритмы Clarans, CURE, DBScan

Алгоритм Clarans (Clustering Large Applications based upon RANdomized Search) [14] формулирует задачу кластеризации как случайный поиск в графе. В результате работы этого алгоритма совокупность узлов графа представляет собой разбиение множества

данных на число кластеров, определенное пользователем. "Качество" полученных кластеров определяется при помощи критериальной функции. Алгоритм Clarans сортирует все возможные разбиения множества данных в поисках приемлемого решения. Поиск решения останавливается в том узле, где достигается минимум среди предопределенного числа локальных минимумов.

Среди новых масштабируемых алгоритмов также можно отметить алгоритм CURE [57] - алгоритм иерархической кластеризации, и алгоритм DBScan [58], где понятие кластера формулируется с использованием концепции плотности (density).

Основным недостатком алгоритмов BIRCH, Clarans, CURE, DBScan является то обстоятельство, что они требуют задания некоторых порогов плотности точек, а это не всегда приемлемо. Эти ограничения обусловлены тем, что описанные алгоритмы ориентированы на сверхбольшие базы данных и не могут пользоваться большими вычислительными ресурсами [59].

Над масштабируемыми методами сейчас активно работают многие исследователи, основная задача которых - преодолеть недостатки алгоритмов, существующих на сегодняшний день.

Методы поиска ассоциативных правил

Как уже упоминалось в первом разделе курса, ассоциация - одна из задач Data Mining. Целью поиска ассоциативных правил (association rule) является нахождение закономерностей между связанными событиями в базах данных.

В этой лекции мы подробно рассмотрим следующие вопросы:

- Что такое ассоциативные правила?
- Какие существуют алгоритмы поиска ассоциативных правил?
- Что такое часто встречающиеся наборы товаров?
- Применение задачи поиска ассоциативных правил?

Очень часто покупатели приобретают не один товар, а несколько. В большинстве случаев между этими товарами существует взаимосвязь. Так, например, покупатель, приобретающий макаронные изделия, скорее всего, захочет приобрести также кетчуп. Эта информация может быть использована для размещения товара на прилавках.

Часто встречающиеся приложения с применением ассоциативных правил:

- розничная торговля: определение товаров, которые стоит продвигать совместно; выбор местоположения товара в магазине; анализ потребительской корзины; прогнозирование спроса;
- перекрестные продажи: если есть информация о том, что клиенты приобрели продукты А, Б и В, то какие из них вероятнее всего купят продукт Г?
- маркетинг: поиск рыночных сегментов, тенденций покупательского поведения;
- сегментация клиентов: выявление общих характеристик клиентов компании, выявление групп покупателей;
- оформление каталогов, анализ сбытовых кампаний фирмы, определение последовательностей покупок клиентов (какая покупка последует за покупкой товара А);
- анализ Web-логов.

Приведем простой пример ассоциативного правила: покупатель, приобретающий банку краски, приобретет кисточку для краски с вероятностью 50%.

Введение в ассоциативные правила

Впервые задача поиска ассоциативных правил (association rule mining) была предложена для нахождения типичных шаблонов покупок, совершаемых в супермаркетах, поэтому иногда ее еще называют анализом рыночной корзины (market basket analysis).

Рыночная корзина - это набор товаров, приобретенных покупателем в рамках одной отдельно взятой транзакции.

Транзакции являются достаточно характерными операциями, ими, например, могут описываться результаты посещений различных магазинов.

Транзакция - это множество событий, которые произошли одновременно.

Регистрируя все бизнес-операции в течение всего времени своей деятельности, торговые компании накапливают огромные собрания транзакций. Каждая такая транзакция представляет собой набор товаров, купленных покупателем за один визит.

Полученные в результате анализа шаблоны включают перечень товаров и число транзакций, которые содержат данные наборы.

Транзакционная или операционная база данных (Transaction database) представляет собой двумерную таблицу, которая состоит из номера транзакции (TID) и перечня покупок, приобретенных во время этой транзакции.

TID - уникальный идентификатор, определяющий каждую сделку или транзакцию.

Пример транзакционной базы данных, состоящей из покупательских транзакций, приведен в таблице 15.1. В таблице первая колонка (TID) определяет номер транзакции, во второй колонке таблицы приведены товары, приобретенные во время определенной транзакции.

Таблица 15.1. Транзакционная база данных

TID	Приобретенные покупки
100	Хлеб, молоко, печенье
200	Молоко, сметана
300	Молоко, хлеб, сметана, печенье
400	Колбаса, сметана
500	Хлеб, молоко, печенье, сметана

На основе имеющейся базы данных нам нужно найти закономерности между событиями, то есть покупками.

Часто встречающиеся шаблоны или образцы

Допустим, имеется транзакционная база данных D. Присвоим значениям товаров переменные (таблица 15.2).

Хлеб = a

Молоко = b

Печенье = c

Сметана = d

Колбаса = e

Конфеты = f

Таблица 15.2. Часто встречающиеся наборы товаров

TID	Приобретенные покупки	→	TID	Приобретенные покупки
100	Хлеб, молоко, печенье		100	a, b, c
200	Молоко, сметана		200	b, d
300	Молоко, хлеб, сметана, печенье		300	b, a, d, c
400	Колбаса, сметана		400	e, d
500	Хлеб, молоко, печенье, сметана		500	a, b, c, d
600	Конфеты		600	f

Рассмотрим набор товаров (Itemset), включающий, например, {Хлеб, молоко, печенье}. Выразим этот набор с помощью переменных:

```
abc={a,b,c}
```

Поддержка

Этот набор товаров встречается в нашей базе данных три раза, т.е. поддержка этого набора товаров равна 3:

```
SUP(abc)=3.
```

При минимальном уровне поддержки, равной трем, набор товаров abc является часто встречающимся шаблоном.

min_sup=3, {Хлеб, молоко, печенье} - часто встречающийся шаблон.

Поддержкой называют количество или процент транзакций, содержащих определенный набор данных.

Для данного набора товаров поддержка, выраженная в процентном отношении, равна 50%.

```
SUP(abc)=(3/6)*100%=50%
```

Поддержку иногда также называют обеспечением набора.

Таким образом, набор представляет интерес, если его поддержка выше определенного пользователем минимального значения (min support). Эти наборы называют часто встречающимися (frequent).

Характеристики ассоциативных правил

Ассоциативное правило имеет вид: "Из события А следует событие В".

В результате такого вида анализа мы устанавливаем закономерность следующего вида: "Если в таранзакции встретился набор товаров (или набор элементов) А, то можно сделать вывод, что в этой же транзакции должен появиться набор элементов В)" Установление таких закономерностей дает нам возможность находить очень простые и понятные правила, называемые ассоциативными.

Основными характеристиками ассоциативного правила являются поддержка и достоверность правила.

Рассмотрим правило "из покупки молока следует покупка печенья" для базы данных, которая была приведена выше в таблице 15.1. Понятие поддержки набора мы уже рассмотрели. Существует понятие поддержки правила.

Правило имеет поддержку s, если s% транзакций из всего набора содержат одновременно наборы элементов А и В или, другими словами, содержат оба товара.

Молоко - это товар А, печенье - это товар В. Поддержка правила "из покупки молока следует покупка печенья" равна 3, или 50%.

Достоверность правила показывает, какова вероятность того, что из события А следует событие В.

Правило "Из А следует В" справедливо с достоверностью с, если с% транзакций из всего множества, содержащих набор элементов А, также содержат набор элементов В.

Число транзакций, содержащих молоко, равно четырем, число транзакций, содержащих печенье, равно трем, достоверность правила равна (3/4)*100%, т.е. 75%.

Достоверность правила "из покупки молока следует покупка печенья" равна 75%, т.е. 75% транзакций, содержащих товар А, также содержат товар В.

Границы поддержки и достоверности ассоциативного правила

При помощи использования алгоритмов поиска ассоциативных правил аналитик может получить все возможные правила вида "Из А следует В", с различными значениями поддержки и достоверности. Однако в большинстве случаев, количество правил необходимо ограничивать заранее установленными минимальными и максимальными значениями поддержки и достоверности.

Если значение поддержки правила слишком велико, то в результате работы алгоритма будут найдены правила очевидные и хорошо известные. Слишком низкое значение поддержки приведет к нахождению очень большого количества правил, которые, возможно, будут в большей части необоснованными, но не известными и не очевидными для аналитика. Таким образом, необходимо определить такой интервал, "золотую середину", который с одной стороны обеспечит нахождение неочевидных правил, а с другой - их обоснованность.

Если уровень достоверности слишком мал, то ценность правила вызывает серьезные сомнения. Например, правило с достоверностью в 3% только условно можно назвать правилом.

Методы поиска ассоциативных правил

Алгоритм AIS. Первый алгоритм поиска ассоциативных правил, называвшийся AIS [62], (предложенный Agrawal, Imielinski and Swami) был разработан сотрудниками исследовательского центра IBM Almaden в 1993 году. С этой работы начался интерес к ассоциативным правилам; на середину 90-х годов прошлого века пришелся пик исследовательских работ в этой области, и с тех пор каждый год появляется несколько новых алгоритмов.

В алгоритме AIS кандидаты множества наборов генерируются и подсчитываются "на лету", во время сканирования базы данных.

Алгоритм SETM. Создание этого алгоритма было мотивировано желанием использовать язык SQL для вычисления часто встречающихся наборов товаров. Как и алгоритм AIS, SETM также формирует кандидатов "на лету", основываясь на преобразованиях базы данных. Чтобы использовать стандартную операцию объединения языка SQL для формирования кандидата, SETM отделяет формирование кандидата от их подсчета.

Неудобство алгоритмов AIS и SETM - излишнее генерирование и подсчет слишком многих кандидатов, которые в результате не оказываются часто встречающимися. Для улучшения их работы был предложен алгоритм Apriori [63].

Работа данного алгоритма состоит из нескольких этапов, каждый из этапов состоит из следующих шагов:

- формирование кандидатов;
- подсчет кандидатов.

Формирование кандидатов (candidate generation) - этап, на котором алгоритм, сканируя базу данных, создает множество i-элементных кандидатов (i - номер этапа). На этом этапе поддержка кандидатов не рассчитывается.

Подсчет кандидатов (candidate counting) - этап, на котором вычисляется поддержка каждого i-элементного кандидата. Здесь же осуществляется отсечение кандидатов, поддержка которых меньше минимума, установленного пользователем (min_sup). Оставшиеся i-элементные наборы называем часто встречающимися.

Рассмотрим работу алгоритма Apriori на примере базы данных D. Иллюстрация работы алгоритма приведена на рис. 15.1. Минимальный уровень поддержки равен 3.

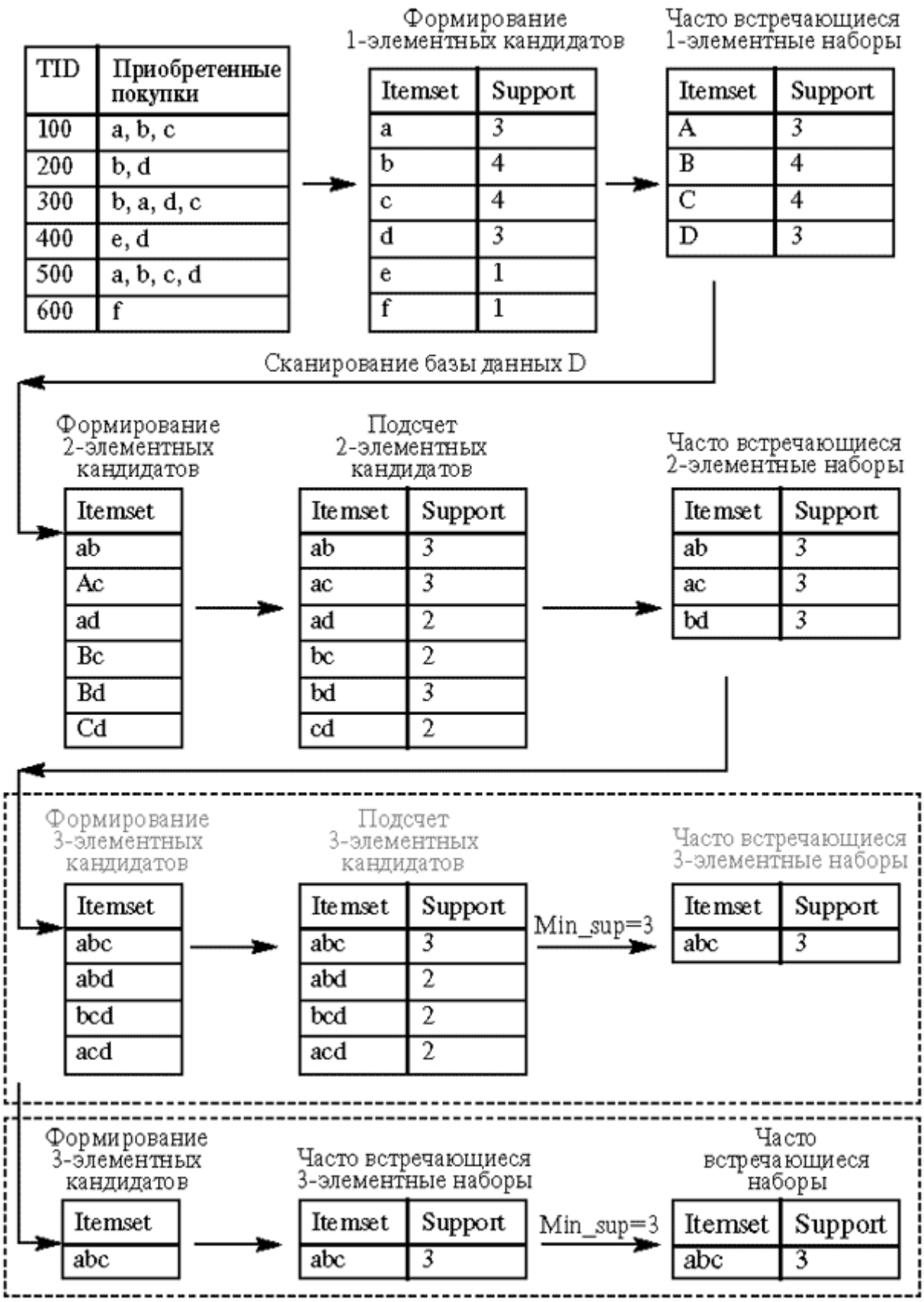

Рис. 15.1. Алгоритм Apriori

На первом этапе происходит формирование одноэлементных кандидатов. Далее алгоритм подсчитывает поддержку одноэлементных наборов. Наборы с уровнем поддержки меньше установленного, то есть 3, отсекаются. В нашем примере это наборы e и f, которые имеют поддержку, равную 1. Оставшиеся наборы товаров считаются часто встречающимися одноэлементными наборами товаров: это наборы a, b, c, d.

Далее происходит формирование двухэлементных кандидатов, подсчет их поддержки и отсечение наборов с уровнем поддержки, меньшим 3. Оставшиеся двухэлементные наборы товаров, считающиеся часто встречающимися двухэлементными наборами ab, ac, bd, принимают участие в дальнейшей работе алгоритма.

Если смотреть на работу алгоритма прямолинейно, на последнем этапе алгоритм формирует трехэлементные наборы товаров: abc, abd, bcd, acd, подсчитывает их поддержку и отсекает наборы с уровнем поддержки, меньшим 3. Набор товаров abc может быть назван часто встречающимся.

Однако алгоритм Apriori уменьшает количество кандидатов, отсекая - априори - тех, которые заведомо не могут стать часто встречающимися, на основе информации об отсеченных кандидатах на предыдущих этапах работы алгоритма.

Отсечение кандидатов происходит на основе предположения о том, что у часто встречающегося набора товаров все подмножества должны быть часто встречающимися. Если в наборе находится подмножество, которое на предыдущем этапе было определено как нечасто встречающееся, этот кандидат уже не включается в формирование и подсчет кандидатов.

Так наборы товаров ad, bc, cd были отброшены как нечасто встречающиеся, алгоритм не рассматривал товаров abd, bcd, acd.

При рассмотрении этих наборов формирование трехэлементных кандидатов происходило бы по схеме, приведенной в верхнем пунктирном прямоугольнике. Поскольку алгоритм априори отбросил заведомо нечасто встречающиеся наборы, последний этап алгоритма сразу определил набор abc как единственный трехэлементный часто встречающийся набор (этап приведен в нижнем пунктирном прямоугольнике).

Алгоритм Apriori рассчитывает также поддержку наборов, которые не могут быть отсечены априори. Это так называемая негативная область (negative border), к ней принадлежат наборы-кандидаты, которые встречаются редко, их самих нельзя отнести к часто встречающимся, но все подмножества данных наборов являются часто встречающимися.

Разновидности алгоритма Apriori

В зависимости от размера самого длинного часто встречающегося набора алгоритм Apriori сканирует базу данных определенное количество раз. Разновидности алгоритма Apriori, являющиеся его оптимизацией, предложены для сокращения количества сканирований базы данных, количества наборов-кандидатов или того и другого [31]. Были предложены следующие разновидности алгоритма Apriori: AprioriTID и AprioriHybrid.

AprioriTid

Интересная особенность этого алгоритма - то, что база данных D не используется для подсчета поддержки кандидатов набора товаров после первого прохода.

С этой целью используется кодирование кандидатов, выполненное на предыдущих проходах. В последующих проходах размер закодированных наборов может быть намного меньше, чем база данных, и таким образом экономятся значительные ресурсы.

AprioriHybrid

Анализ времени работы алгоритмов Apriori и AprioriTid показывает, что в более ранних проходах Apriori добивается большего успеха, чем AprioriTid; однако AprioriTid работает лучше Apriori в более поздних проходах. Кроме того, они используют одну и ту же процедуру формирования наборов-кандидатов. Основанный на этом наблюдении, алгоритм AprioriHybrid предложен, чтобы объединить лучшие свойства алгоритмов Apriori и AprioriTid. AprioriHybrid использует алгоритм Apriori в начальных проходах и переходит к алгоритму AprioriTid, когда ожидается, что закодированный набор первоначального множества в конце прохода будет соответствовать возможностям памяти. Однако, переключение от Apriori до AprioriTid требует вовлечения дополнительных ресурсов.

Некоторыми авторами были предложены другие алгоритмы поиска ассоциативных правил, целью которых также было усовершенствование алгоритма Apriori. Кратко изложим суть нескольких, для более подробной информации можно рекомендовать [31, 33].

Один из них - **алгоритм DHP**, также называемый алгоритмом хеширования (J. Park, M. Chen and P. Yu, 1995 год). В основе его работы - вероятностный подсчет наборов-кандидатов, осуществляемый для сокращения числа подсчитываемых кандидатов на каждом этапе выполнения алгоритма Apriori [63, 64]. Сокращение обеспечивается за счет того, что каждый из k-элементных наборов-кандидатов помимо шага сокращения проходит шаг хеширования. В алгоритме на k-1 этапе во время выбора кандидата создается так называемая хеш-таблица. Каждая запись хеш-таблицы является счетчиком всех поддержек k-элементных наборов, которые соответствуют этой записи в хеш-таблице. Алгоритм использует эту информацию на этапе k для сокращения множества k-элементных наборов-кандидатов. После сокращения подмножества, как это происходит в Apriori, алгоритм может удалить набор-кандидат, если его значение в хеш-таблице меньше порогового значения, установленного для обеспечения.

К другим усовершенствованным алгоритмам относятся: PARTITION, DIC, алгоритм "выборочного анализа".

PARTITION алгоритм (A. Savasere, E. Omiecinski and S. Navathe, 1995 год). Этот алгоритм разбиения (разделения) заключается в сканировании транзакционной базы данных путем разделения ее на непересекающиеся разделы, каждый из которых может уместиться в оперативной памяти [65]. На первом шаге в каждом из разделов при помощи алгоритма Apriori определяются "локальные" часто встречающиеся наборы данных. На втором подсчитывается поддержка каждого такого набора относительно всей базы данных. Таким образом, на втором этапе определяется множество всех потенциально встречающихся наборов данных.

Алгоритм DIC, Dynamic Itemset Counting (S. Brin R. Motwani, J. Ullman and S. Tsur, 1997 год). Алгоритм разбивает базу данных на несколько блоков, каждый из которых отмечается так называемыми "начальными точками" (start point), и затем циклически сканирует базу данных [64].

Пример решения задачи поиска ассоциативных правил

Дана транзакционная база данных, необходимо найти наиболее часто встречающиеся наборы товаров и набор ассоциативных правил с определенными границами значений поддержки и доверия.

Рассмотрим процесс построения ассоциативных правил в аналитическом пакете Deductor.

Транзакционная база данных, которая содержит в каждой записи номер чека и товар, приобретенный по этому чеку, имеет формат MS Excel. Для начала импортируем данные из файла MS Excel в среду Deductor, этот процесс аналогичен тому, что был рассмотрен в лекции о нейронных сетях. Единственное отличие - в назначении столбцов. Для номера транзакции (обычно в базе данных - это поле "номер чека") указываем тип "идентификатор транзакции (ID)", а для наименований товара - тип "элемент". Результат импорта базы данных из файла MS Excel в среду Deductor видим на рис. 15.2. На рисунке приведен фрагмент базы данных, которая содержит более 140 записей.

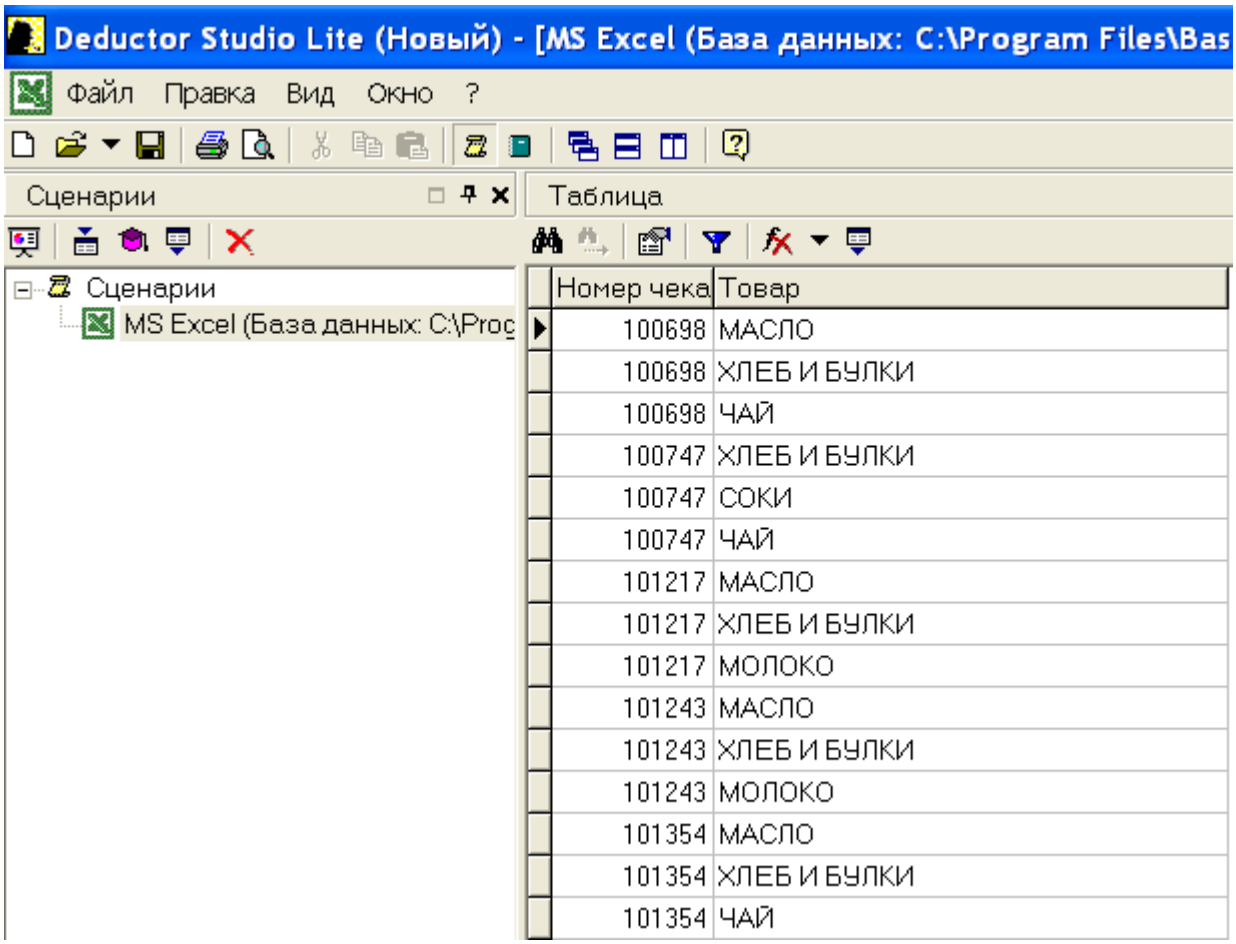

Рис. 15.2. Транзакционная база данных, импортированная в Deductor из файла MS Excel

Далее вызываем мастер обработки и выбираем метод "Ассоциативные правила". На втором шаге мастера проверяем назначения исходных столбцов данных, они должны иметь тип "ID" и "элемент".

На третьем шаге, проиллюстрированном на рис. 15.3, необходимо настроить параметры поиска правил, т.е. установить минимальные и максимальные характеристики поддержки и достоверности. Это наиболее "ответственный" момент формирования набора правил, о важности выбора границ значений поддержки и достоверности уже говорилось в начале лекции. Выбор можно сделать на основе каких-либо соображений, имеющегося опыта анализа подобных данных, интуиции или же определить в ходе экспериментов.

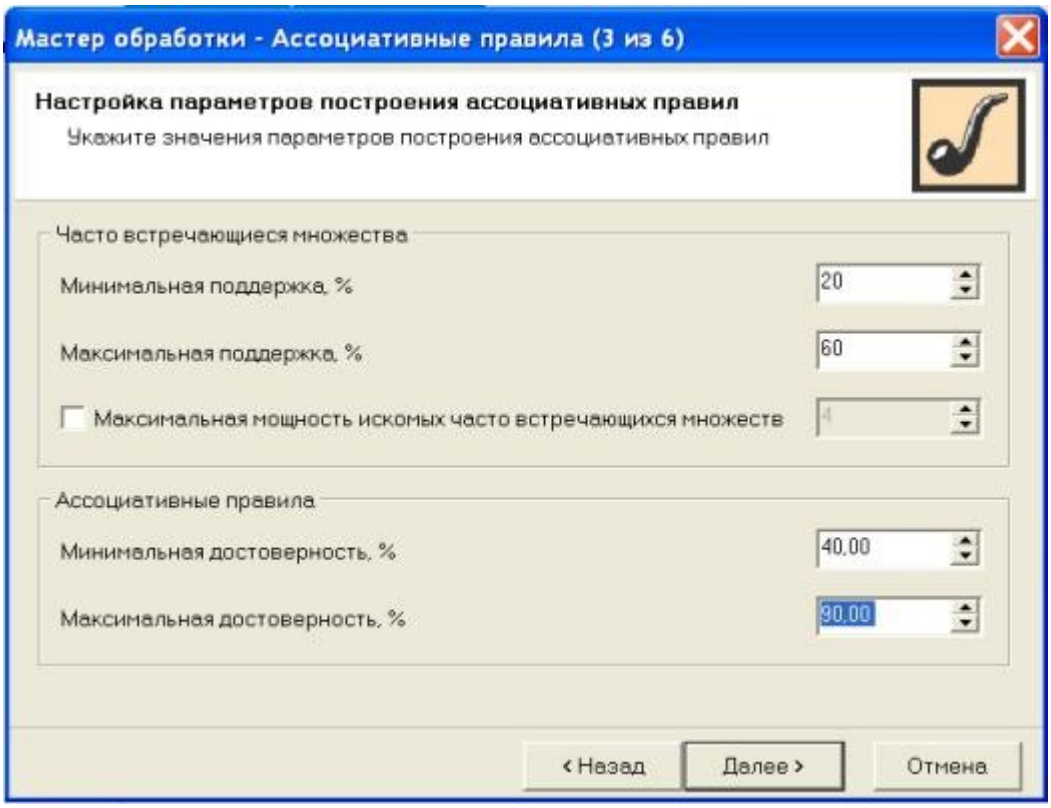

Рис. 15.3. Настройка параметров построения ассоциативных правил

Мы установим такие границы для параметров поиска: минимальный и максимальный уровень поддержки равны 20% и 60% соответственно, минимальный и максимальный уровень значения достоверности равны 40% и 90% соответственно. Эти значения были выявлены в ходе проведения нескольких экспериментов, и оказалось, что именно при таких значениях формируется требуемый набор правил. При указании некоторых значений, например, уровня поддержки от 30% до 50%, набор правил не формируется, поскольку ни одно правило по параметрам поддержки не входит в этот интервал.

На следующем шаге мастера запускается процесс поиска ассоциативных правил. В результате видим информацию о количестве множеств и найденных правил в виде гистограммы распределения часто встречающихся множеств по их мощности. Данный процесс проиллюстрирован на рис. 15.4.

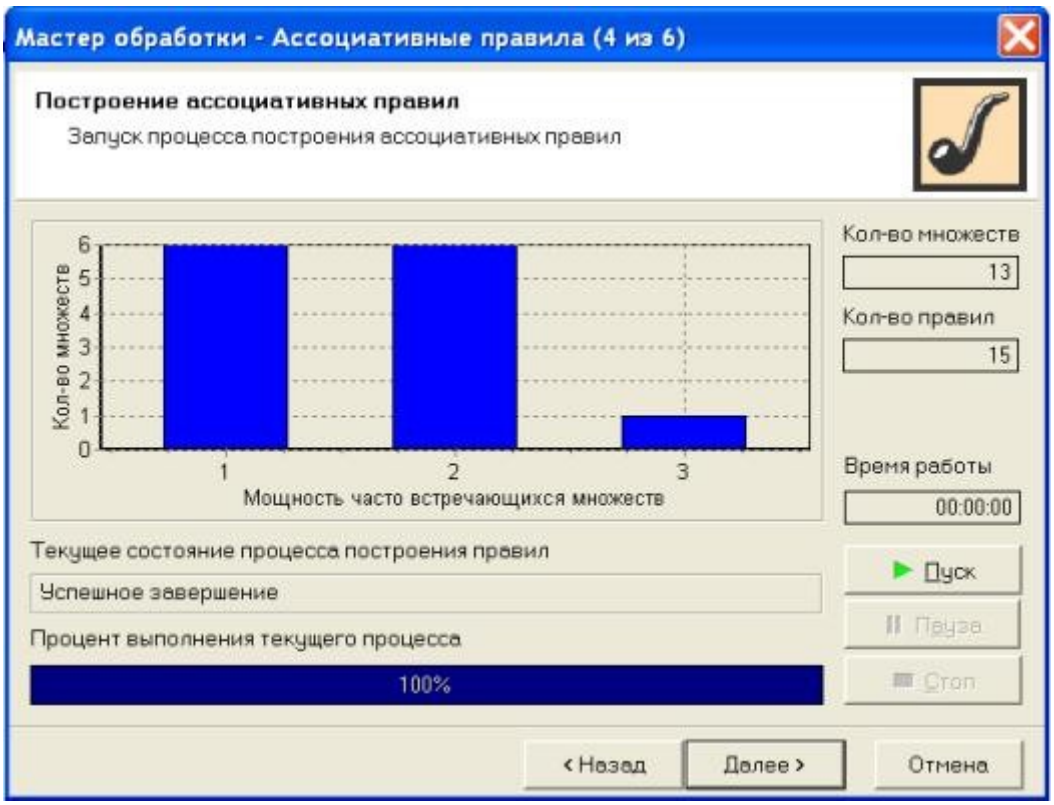

Рис. 15.4. Процесс построения ассоциативных правил

Здесь мы видим, что количество сформированных множеств равно тринадцати - это популярные наборы, количество сформированных правил - пятнадцать.

На следующем шаге для просмотра полученных результатов предлагается выбрать визуализаторы из списка; мы выберем такие: "Популярные наборы", "Правила", "Дерево правил", "Что-если". Рассмотрим, что они из себя представляют.

Визуализатор "Популярные наборы". Популярные наборы или часто встречающиеся наборы - это наборы, состоящие из одного или нескольких товаров, которые в транзакциях наиболее часто встречаются одновременно. Характеристикой, насколько часто набор встречается в анализируемом наборе данных, является поддержка.

Популярные наборы нашего набора данных, найденные при заданных параметрах, приведены в таблице 15.3. Есть возможность отсортировать данную таблицу по разным ее характеристикам. Для определения наиболее популярных товаров и их наборов удобно отсортировать ее по уровню поддержки. Таким образом, мы видим, что наибольшей популярностью пользуются такие товары: хлеб и булки, масло, соки.

N	Множество	↑Поддержка	
		%	Кол-во
6	ХЛЕБ И БУЛКИ	54,55	24

3	МАСЛО	52,27	23
5	СОКИ	50,00	22
10	МАСЛО **И** ХЛЕБ И БУЛКИ	45,45	20
4	МОЛОКО	43,18	19
2	КЕФИР	31,82	14
1	ЙОГУРТЫ	31,82	14
12	СОКИ **И** ХЛЕБ И БУЛКИ	22,73	10
11	МОЛОКО **И** ХЛЕБ И БУЛКИ	22,73	10
8	МАСЛО **И** МОЛОКО	22,73	10
7	ЙОГУРТЫ **И** КЕФИР	22,73	10
13	МАСЛО **И** МОЛОКО **И** ХЛЕБ И БУЛКИ	20,45	9
9	МАСЛО **И** СОКИ	20,45	9

Визуализатор "Правила"

Правила в данном визуализаторе размещены в виде списка. Каждое правило, представленное как "условие-следствие", характеризуется значением поддержки в абсолютном и процентном выражении, а также достоверностью. Таким образом, аналитик видит поведение покупателей, описанное в виде набора правил. Набор правил для решаемой нами задачи приведен в таблице 15.4. Например, первое правило говорит о том, что если покупатель купил йогурт, то с достоверностью или вероятностью 71% он купит также кефир. Эта информация полезна с различных точек зрения. Она, например, помогает решить задачу расположения товаров в магазине.

N	Условие	Следствие	Поддержка		Достоверность, %
			%	Кол-во	
1	ЙОГУРТЫ	КЕФИР	22,73	10	71,43
2	КЕФИР	ЙОГУРТЫ	22,73	10	71,43
3	МАСЛО	МОЛОКО	22,73	10	43,48
4	МОЛОКО	МАСЛО	22,73	10	52,63
5	СОКИ	МАСЛО	20,45	9	40,91

6	МАСЛО	ХЛЕБ И БУЛКИ	45,45	20	86,96
7	ХЛЕБ И БУЛКИ	МАСЛО	45,45	20	83,33
8	МОЛОКО	ХЛЕБ И БУЛКИ	22,73	10	52,63
9	ХЛЕБ И БУЛКИ	МОЛОКО	22,73	10	41,67
10	СОКИ	ХЛЕБ И БУЛКИ	22,73	10	45,45
11	ХЛЕБ И БУЛКИ	СОКИ	22,73	10	41,67
12	МАСЛО **И** МОЛОКО	ХЛЕБ И БУЛКИ	20,45	9	90,00
13	МАСЛО **И** ХЛЕБ И БУЛКИ	МОЛОКО	20,45	9	45,00
14	МОЛОКО **И** ХЛЕБ И БУЛКИ	МАСЛО	20,45	9	90,00
15	МОЛОКО	МАСЛО **И** ХЛЕБ И БУЛКИ	20,45	9	47,37

При большом количестве найденных правил и широком ассортименте товаров анализировать полученные правила достаточно сложно. Для удобства анализа таких наборов правил предлагаются визуализаторы "Дерево правил" и "Что-если".

Визуализатор "Дерево правил" представляет собой двухуровневое дерево, которое может быть построено по двум критериям: по условию и по следствию. Если дерево построено по условию, то вверху списка отображается условие правила, а список, прилагающийся к данному условию, состоит из его следствий. При выборе определенного условия, в правой части визуализатора отображаются следствия условия, уровень поддержки и достоверности.

В случае построения дерева по следствию, вверху списка отображается следствие правила, а список состоит из его условий. При выборе определенного следствия, в правой части визуализатора мы видим условия этого правила с указанием уровня поддержки и достоверности.

Визуализатор "что-если" удобен, если нам необходимо ответить на вопрос, какие следствия могут получиться из данного условия.

Например, выбрав условие "МОЛОКО", в левой части экрана получаем три следствия "МАСЛО", "ХЛЕБ И БУЛКИ", "МАСЛО И ХЛЕБ И БУЛКИ", для которых указаны уровень поддержки и достоверности. Этот визуализатор представлен на рис. 15.5.

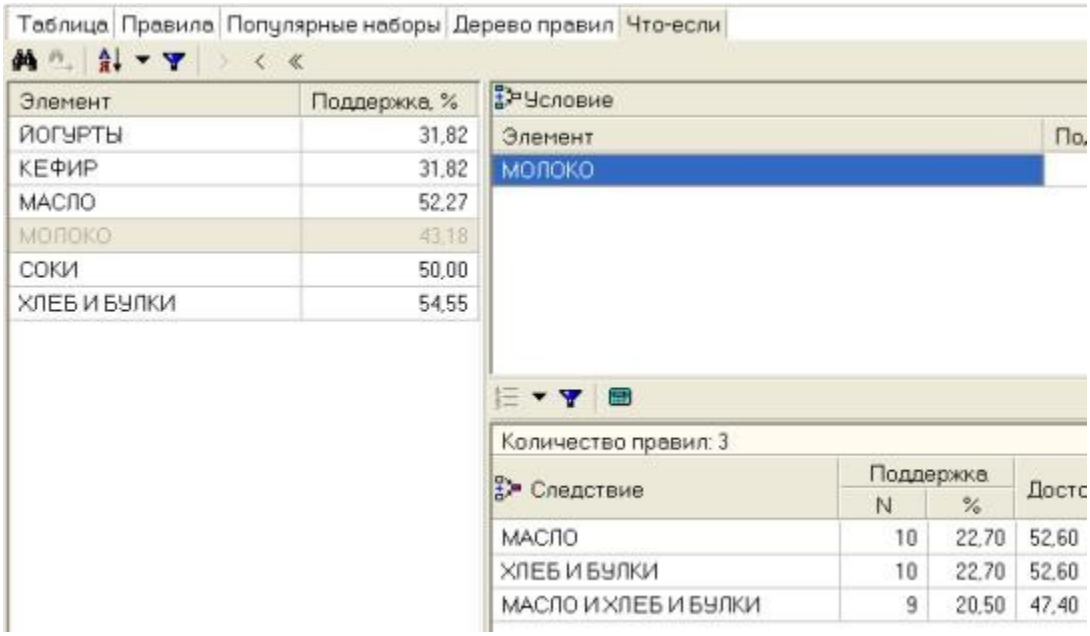

Рис. 15.5. Визуализатор "Что-если"

Рассмотренный пример поиска ассоциативных правил является типичной иллюстрацией задачи анализа покупательской корзины. В результате ее решения определяются часто встречающиеся наборы товаров, а также наборы товаров, совместно приобретаемые покупателями. Найденные правила могут быть использованы для решения различных задач, в частности для размещения товаров на прилавках магазинов, предоставления скидок на пары товаров для повышения объема продаж и, следовательно, прибыли и других задач.

Способы визуального представления данных. Методы визуализации

"Говорят, один рисунок стоит тысячи слов, и это действительно так, но при условии, что рисунок хороший." Боумена [65]

С возрастанием количества накапливаемых данных, даже при использовании сколь угодно мощных и разносторонних алгоритмов Data Mining, становится все сложнее "переваривать" и интерпретировать полученные результаты. А, как известно, одно из положений Data Mining - поиск практически полезных закономерностей. Закономерность может стать практически полезной, только если ее можно осмыслить и понять.

В 1987 году по инициативе ACM SIGGRAPH IEEE Computer Society Technical Committee of Computer Graphics, в связи с необходимостью использования новых методов, средств и технологий данных, были сформулированы соответствующие задачи направления визуализации.

К способам визуального или графического представления данных относят графики, диаграммы, таблицы, отчеты, списки, структурные схемы, карты и т.д.

Визуализация традиционно рассматривалась как вспомогательное средство при анализе данных, однако сейчас все больше исследований говорит о ее самостоятельной роли.

Традиционные методы визуализации могут находить следующее применение:

- представлять пользователю информацию в наглядном виде;
- компактно описывать закономерности, присущие исходному набору данных;
- снижать размерность или сжимать информацию;
- восстанавливать пробелы в наборе данных;
- находить шумы и выбросы в наборе данных.

Визуализация инструментов Data Mining

Каждый из алгоритмов Data Mining использует определенный подход к визуализации. В предыдущих лекциях мы рассмотрели ряд методов Data Mining. В ходе использования каждого из методов, а точнее, его программной реализации, мы получали некие визуализаторы, при помощи которых нам удавалось интерпретировать результаты, полученные в результате работы соответствующих методов и алгоритмов.

- Для деревьев решений это визуализатор дерева решений, список правил, таблица сопряженности.
- Для нейронных сетей в зависимости от инструмента это может быть топология сети, график изменения величины ошибки, демонстрирующий процесс обучения.
- Для карт Кохонена: карты входов, выходов, другие специфические карты.
- Для линейной регрессии в качестве визуализатора выступает линия регрессии.
- Для кластеризации: дендрограммы, диаграммы рассеивания.

Диаграммы и графики рассеивания часто используются для оценки качества работы того или иного метода.

Все эти способы визуального представления или отображения данных могут выполнять одну из функций:

- являются иллюстрацией построения модели (например, представление структуры (графа) нейронной сети);
- помогают интерпретировать полученный результат;
- являются средством оценки качества построенной модели;
- сочетают перечисленные выше функции (дерево решений, дендрограмма).

Визуализация Data Mining моделей

Первая функция (иллюстрация построения модели), по сути, является визуализацией Data Mining модели. Существует много различных способов представления моделей, но графическое ее представление дает пользователю максимальную "ценность". Пользователь, в большинстве случаев, не является специалистом в моделировании, чаще всего он эксперт в своей предметной области. Поэтому модель Data Mining должна быть представлена на наиболее естественном для него языке или, хотя бы, содержать минимальное количество различных математических и технических элементов.

Таким образом, доступность является одной из основных характеристик модели Data Mining. Несмотря на это, существует и такой распространенный и наиболее простой способ представления модели, как "черный ящик". В этом случае пользователь не понимает поведения той модели, которой пользуется. Однако, несмотря на непонимание, он получает результат - выявленные закономерности. Классическим примером такой модели является модель нейронной сети.

Другой способ представления модели - представление ее в интуитивном, понятном виде. В этом случае пользователь действительно может понимать то, что происходит "внутри" модели. Таким образом, можно обеспечить его непосредственное участие в процессе. Такие модели обеспечивают пользователю возможность обсуждать ее логику с коллегами, клиентами и другими пользователями, или объяснять ее.

Понимание модели ведет к пониманию ее содержания. В результате понимания возрастает доверие к модели. Классическим примером является дерево решений. Построенное дерево решений действительно улучшает понимание модели, т.е. используемого инструмента Data Mining.

Кроме понимания, такие модели обеспечивают пользователя возможностью взаимодействовать с моделью, задавать ей вопросы и получать ответы. Примером такого взаимодействия является средство "что, если". При помощи диалога "система-пользователь" пользователь может получить понимание модели.

Теперь перейдем к функциям, которые помогают интерпретировать и оценить результаты построения Data Mining моделей. Это всевозможные графики, диаграммы, таблицы, списки и т.д.

Примерами средств визуализации, при помощи которых можно оценить качество модели, являются диаграмма рассеивания, таблица сопряженности, график изменения величины ошибки.

Диаграмма рассеивания представляет собой график отклонения значений, прогнозируемых при помощи модели, от реальных. Эти диаграммы используют для непрерывных величин. Визуальная оценка качества построенной модели возможна только по окончанию процесса построения модели.

Таблица сопряженности используется для оценки результатов классификации. Такие таблицы применяются для различных методов классификации. Они уже использовались нами в предыдущих лекциях. Оценка качества построенной модели возможно только по окончанию процесса построения модели.

График изменения величины ошибки. График демонстрирует изменение величины ошибки в процессе работы модели. Например, в процессе работы нейронных сетей пользователь может наблюдать за изменением ошибки на обучающем и тестовом множествах и остановить обучение для недопущения "переобучения" сети. Здесь оценка качества модели и его изменения может оцениваться непосредственно в процессе построения модели.

Примерами средств визуализации, которые помогают интерпретировать результат, являются: линия тренда в линейной регрессии, карты Кохонена, диаграмма рассеивания в кластерном анализе.

Методы визуализации

Методы визуализации, в зависимости от количества используемых измерений, принято классифицировать на две группы [22]:

- представление данных в одном, двух и трех измерениях;
- представление данных в четырех и более измерениях.

Представление данных в одном, двух и трех измерениях

К этой группе методов относятся хорошо известные способы отображения информации, которые доступны для восприятия человеческим воображением. Практически любой современный инструмент Data Mining включает способы визуального представления из этой группы.

В соответствии с количеством измерений представления это могут быть следующие способы:

- одномерное (univariate) измерение, или 1-D;
- двумерное (bivariate) измерение, или 2-D;
- трехмерное или проекционное (projection) измерение, или 3-D.

Следует заметить, что наиболее естественно человеческий глаз воспринимает двухмерные представления информации.

При использовании двух- и трехмерного представления информации пользователь имеет возможность увидеть закономерности набора данных:

- его кластерную структуру и распределение объектов на классы (например, на диаграмме рассеивания);

- топологические особенности;
- наличие трендов;
- информацию о взаимном расположении данных;
- существование других зависимостей, присущих исследуемому набору данных.

Если набор данных имеет более трех измерений, то возможны такие варианты:

- использование многомерных методов представления информации (они рассмотрены ниже);
- снижение размерности до одно-, двух- или трехмерного представления. Существуют различные способы снижения размерности, один из них - факторный анализ - был рассмотрен в одной из предыдущих лекций. Для снижения размерности и одновременного визуального представления информации на двумерной карте используются самоорганизующиеся карты Кохонена.

Представление данных в 4 + измерениях

Представления информации в четырехмерном и более измерениях недоступны для человеческого восприятия. Однако разработаны специальные методы для возможности отображения и восприятия человеком такой информации.

Наиболее известные способы многомерного представления информации:

- параллельные координаты;
- "лица Чернова";
- лепестковые диаграммы.

Параллельные координаты

В параллельных координатах переменные кодируются по горизонтали, вертикальная линия определяет значение переменной. Пример набора данных, представленного в декартовых координатах и параллельных координатах, дан на рис. 16.1 [22]. Этот метод представления многомерных данных был изобретен Альфредом Инселбергом (Alfred Inselberg) в 1985 году.

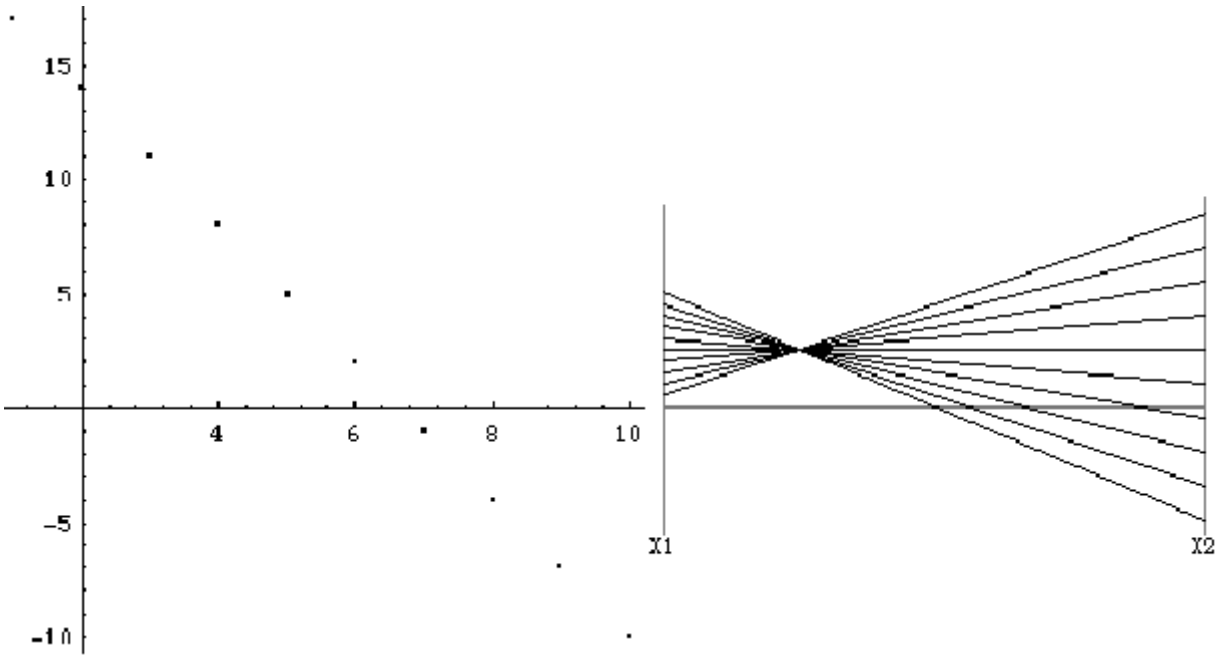

Рис. 16.1. Набор данных в декартовых координатах и в параллельных координатах

"Лица Чернова"

Основная идея представления информации в "лицах Чернова" состоит в кодировании значений различных переменных в характеристиках или чертах человеческого лица [66]. Пример такого "лица" приведен на рис.16.2.

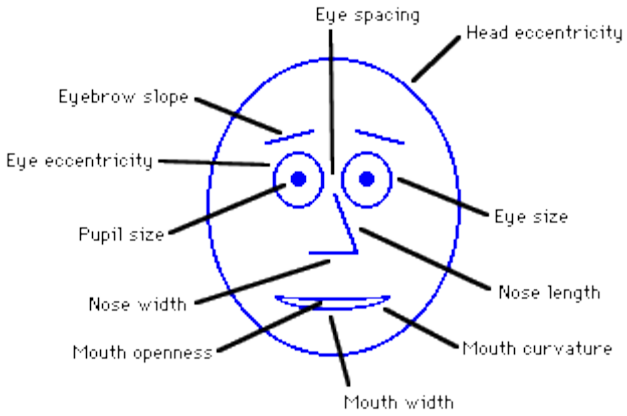

Рис. 16.2. "Лицо Чернова"

Для каждого наблюдения рисуется отдельное "лицо". На каждом "лице" относительные значения переменных представлены как формы и размеры отдельных черт лица (например, длина и ширина носа, размер глаз, размер зрачка, угол между бровями).

Анализ информации при помощи такого способа отображения основан на способности человека интуитивно находить сходства и различия в чертах лица.

На рис. 16.3 представлен набор данных, каждая запись которого выражена в виде "лица Чернова".

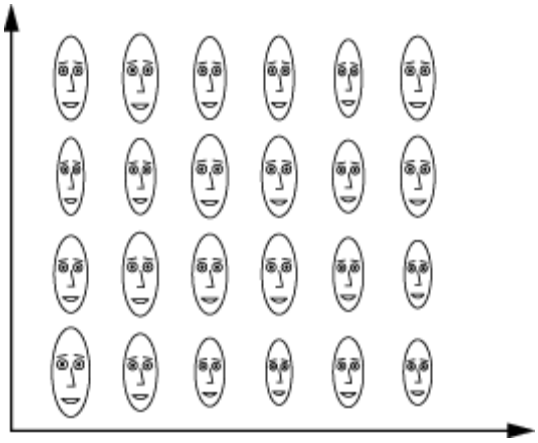

Рис. 16.3. Пример многомерного изображения данных при помощи "лиц Чернова"

Перед использованием методов визуализации необходимо:

- Проанализировать, следует ли изображать все данные или же какую-то их часть.
- Выбрать размеры, пропорции и масштаб изображения.
- Выбрать метод, который может наиболее ярко отобразить закономерности, присущие набору данных.

Многие современные средства анализа данных позволяют строить сотни типов различных графиков и диаграмм. Поэтому выбор метода визуализации, если он самостоятельно осуществляется пользователем, не так прост и легок, как может показаться на первый взгляд. Наличие большого количества средств визуализации, представленных в инструменте, который применяет пользователь, может даже вызвать растерянность.

Одну и ту же информацию можно представить при помощи различных средств. Для того чтобы средство визуализации могло выполнять свое основное назначение - представлять информацию в простом и доступном для человеческого восприятия виде - необходимо придерживаться законов соответствия выбранного решения содержанию отображаемой информации и ее функциональному назначению. Иными словами, нужно сделать так, чтобы при взгляде на визуальное представление информации можно было сразу выявить закономерности в исходных данных и принимать на их основе решения.

Среди двухмерных и трехмерных средств наиболее широко известны линейные графики, линейные, столбиковые, круговые секторные и векторные диаграммы.

Приведем рекомендации по использованию этих наиболее простых и популярных средств визуализации.

При помощи **линейного графика** можно отобразить тенденцию, передать изменения какого-либо признака во времени. Для сравнения нескольких рядов чисел такие графики наносятся на одни и те же оси координат.

Гистограмму применяют для сравнения значений в течение некоторого периода или же соотношения величин.

Круговые диаграммы используют, если необходимо отобразить соотношение частей и целого, т.е. для анализа состава или структуры явлений. Составные части целого изображаются секторами окружности. Секторы рекомендуют размещать по их величине: вверху - самый крупный, остальные - по движению часовой стрелки в порядке уменьшения их величины. Круговые диаграммы также применяют для отображения результатов факторного анализа, если действия всех факторов являются однонаправленными. При этом каждый фактор отображается в виде одного из секторов круга.

Выбор того или иного средства визуализации зависит от поставленной задачи (например, нужно определить структуру данных или же динамику процесса) и от характера набора данных.

Качество визуализации

Современные аналитические средства, в том числе и Data Mining, немыслимы без качественной визуализации. В результате использования средств визуализации должны быть получены наглядные и выразительные, ясные и простые изображения, за счет использования разнообразных средств: цвета, контраста, границ, пропорций, масштаба и т.д.

В связи с ростом требований к средствам визуализации, а также необходимости сравнивания их между собой, в последние годы был сформирован ряд принципов качественного визуального представления информации.

Принципы Тафта (Tufte's Principles) графического представления данных высокого качества [67] гласят:

- предоставляйте пользователю самое большое количество идей, в самое короткое время, с наименьшим количеством чернил на наименьшем пространстве;
- говорите правду о данных.

В [65] описаны основные принципы компоновки визуальных средств представления информации:

1. Принцип лаконичности.
2. Принцип обобщения и унификации.
3. Принцип акцента на основных смысловых элементах.
4. Принцип автономности.
5. Принцип структурности.
6. Принцип стадийности.
7. Принцип использования привычных ассоциаций и стереотипов.

Принцип лаконичности говорит о том, что средство визуализации должно содержать лишь те элементы, которые необходимы для сообщения пользователю существенной информации, точного понимания ее значения или принятия (с вероятностью не ниже допустимой величины) соответствующего оптимального решения.

Кроме обозначенных выше принципов, средство визуализации должно обладать высокой надежностью и скоростью, которая устроит пользователя, принимающего на основе этой информации решения.

Представление пространственных характеристик

Отдельным направлением визуализации является наглядное представление пространственных характеристик объектов. В большинстве случаев такие средства выделяют на карте отдельные регионы и обозначают их различными цветами в зависимости от значения анализируемого показателя.

На рис. 16.4 приведен пример такой визуализации в среде MineSet [26], являющейся, в данном случае, инструментом визуального Data Mining. Карта представлена в виде графического интерфейса, отображающего данные в виде трехмерного ландшафта произвольно определенных и позиционированных форм (столбчатых диаграмм, каждая с индивидуальными высотой и цветом). Такой способ позволяет наглядно показывать количественные и реляционные характеристики пространственно-ориентированных данных и быстро идентифицировать в них тренды.

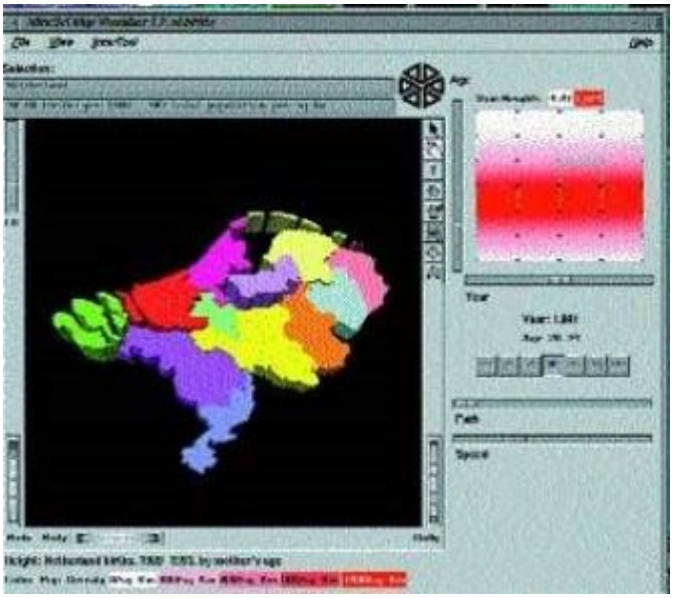

Рис. 16.4. MineSet. Ландшафтный визуализатор

Основные тенденции в области визуализации

Как уже отмечалось, при помощи средств визуализации поддерживаются важные задачи бизнеса, среди которых - процесс принятия решений. В связи с этим возникает необходимость перехода средств визуализации на более качественный уровень, который характеризуется появлением абсолютно новых средств визуализации и взглядов на ее функции, а также развитием ряда тенденций в этой области.

Среди основных тенденций в области визуализации Филип Рассом (Philip Russom) выделяет [68]:

- Разработка сложных видов диаграмм.

- Повышение уровня взаимодействия с визуализацией пользователя.
- Увеличение размеров и сложности структур данных, представляемых визуализацией.

1. Разработка сложных видов диаграмм.

   Большинство визуализаций данных построено на основе диаграмм стандартного типа (секторные диаграммы, графики рассеяния и.т.д.). Эти способы являются одновременно старейшими, наиболее элементарными и распространенными. В последние годы перечень видов диаграмм, поддерживаемых инструментальными средствами визуализации, существенно расширился. Поскольку потребности пользователей весьма многообразны, инструменты визуализации поддерживают самые различные типы диаграмм. Например, известно, что бизнес-пользователи предпочитают секторные диаграммы и гистограммы, тогда как ученых больше устраивают визуализации в виде графиков рассеяния и диаграмм констелляции. Пользователи, работающие с геопространственными данными, сильнее заинтересованы в картах и прочих трехмерных представлениях данных. Электронные инструментальные панели, в свою очередь, более популярны среди руководителей, использующих бизнес-аналитические технологии для контроля за показателями работы компании. Такие пользователи нуждаются в наглядной визуализации в виде "спидометров", "термометров" и "светофоров".

   Средства создания диаграмм и презентационной графики предназначены главным образом для визуализации данных. Однако возможности такой визуализации обычно встроены и во множество различных других программ и систем - в инструменты репортинга и OLAP, средства для Text Mining и Data Mining, а также в CRM-приложения и приложения для управления бизнесом. Для создания встроенной визуализации многие поставщики реализуют визуализационную функциональность в виде компонент, встраиваемых в различные инструменты, приложения, программы и web-страницы (в том числе инструментальные панели и персонализированные страницы порталов).

2. Повышение уровня взаимодействия с визуализацией пользователя.

   Еще совсем недавно большая часть средств визуализации представляла собой статичные диаграммы, предназначенные исключительно для просмотра. Сейчас широко используются динамические диаграммы, уже сами по себе являющиеся пользовательским интерфейсом, в котором пользователь может напрямую и интерактивно манипулировать визуализацией, подбирая новое представление информации.

   Например, базовое взаимодействие позволяет пользователю вращать диаграмму или изменять ее тип в поисках наиболее полного представления данных. Кроме того, пользователь может менять визуальные свойства - к примеру, шрифты, цвета и рамки. В визуализациях сложного типа (графиках рассеяния или диаграммах констелляции) пользователь может выбирать информационные точки с помощью мыши и перемещать их, облегчая тем самым понимание представления данных.

   Более совершенные методы визуализации данных часто включают в себя диаграмму или любую другую визуализацию как составной уровень. Пользователь может углубляться (drill down) в визуализацию, исследуя подробности

обобщенных ею данных, или углубляться в OLAP, Data Mining или другие сложные технологии.

Сложное взаимодействие позволяет пользователю изменять визуализацию для нахождения альтернативных интерпретаций данных. Взаимодействие с визуализацией подразумевает минимальный по своей сложности пользовательский интерфейс, в котором пользователь может управлять представлением данных, просто "кликая" на элементы визуализации, перетаскивая и помещая представления объектов данных или выбирая пункты меню. Инструменты OLAP или Data Mining превращают непосредственное взаимодействие с визуализацией в один из этапов итерационного анализа данных. Средства Text Mining или управления документами придают такому непосредственному взаимодействию характер навигационного механизма, помогающего пользователю исследовать библиотеки документов.

Визуальный запрос является наиболее современной формой сложного взаимодействия пользователя с данными. В нем пользователь может, например, видеть крайние информационные точки графика рассеяния, выбирать их мышкой и получать новые визуализации, представляющие именно эти точки. Приложение визуализации данных генерирует соответствующий язык запроса, управляет принятием запроса базой данных и визуально представляет результирующее множество. Пользователь может сфокусироваться на анализе, не отвлекаясь на составление запроса.

3. Увеличение размеров и сложности структур данных, представляемых визуализацией.

Элементарная секторная диаграмма или гистограмма визуализирует простые последовательности числовых информационных точек. Однако новые усовершенствованные типы диаграмм способны визуализировать тысячи таких точек и даже сложные структуры данных - например, нейронные сети.

Скажем, средства OLAP (а также инструменты генерации запросов и выпуска отчетов) уже давно поддерживают диаграммы для своих онлайновых отчетов. Новые визуализационные программы обновляют контент за счет периодически повторяющегося считывания данных. Фактически пользователи визуализационных программ, отслеживающие линейные процессы (колебания фондового рынка, показатели работы компьютерных систем, сейсмограммы, сетки полезности и др.), нуждаются в загрузке данных в режиме реального времени или близком к нему режиме.

Пользователи инструментов Data Mining обычно анализируют очень большие наборы численных данных. Традиционные типы диаграмм для бизнеса (секторные диаграммы и гистограммы) плохо справляются с представлением тысяч информационных точек. Поэтому инструменты Data Mining почти всегда поддерживают некую форму визуализации данных, способную отражать структуры и закономерности исследуемых наборов данных, в соответствии с тем аналитическим подходом, который используется в инструменте.

Помимо того, что визуализация поддерживает обработку структурированных данных, она также является ключевым средством представления схем так называемых неструктурированных данных, например текстовых документов, т.е.

Text Mining. В частности, средства Text Mining могут осуществлять парсинг больших пакетов документов и формировать предметные указатели понятий и тем, освещенных в этих документах. Когда предметные указатели созданы с помощью нейросетевой технологии, пользователю непросто продемонстрировать их без некоторой формы визуализации данных. Визуализация в таком случае преследует две цели:

- визуальное представление контента библиотеки документов;
- навигационный механизм, который пользователь может применять при исследовании документов и их тем.

Выводы

Как показывают многие исследования, визуализация является одним из наиболее перспективных направлений анализа данных, в т.ч. Data Mining. Однако в этом направлении можно выделить проблемы, такие как сложность ориентации среди огромного количества инструментов, предлагающих решения по визуализации, а также непризнание рядом специалистов методов визуализации как полноценных средств анализа и навязывание им вспомогательной роли при использовании других методов. Однако у визуализации есть неоспоримые преимущества: она может служить источником информации для пользователя, не требуя теоретических знаний и специальных навыков работы, может выступить тем языком, который объединит профессионалов из различных проблемных областей, может превратить исходный набор данных в изображение, благодаря которому у исследователя могут появиться абсолютно новые, неожиданные решения.

Комплексный подход к внедрению Data Mining, OLAP и хранилищ данных в СППР

В одной из предыдущих лекций мы рассматривали информационную пирамиду, в ходе движения по которой, от данных к решениям, объемы знаний переходят в ценность бизнеса. Процесс Data Mining, который как раз и заключается в движении вверх по этой информационной пирамиде, неразрывно связан с процессом принятия решений, его можно рассматривать как неотъемлемую часть систем поддержки принятия решений (СППР).

Таким образом, Data Mining можно рассматривать как процесс поддержки принятия решений, при этом накопленные сведения автоматически обобщаются до информации, которая может быть охарактеризована как знания [11].

С понятием решений и принятием решений мы уже кратко познакомились в одной из первых лекций курса.

СППР возникли в результате развития управленческих информационных систем и систем управления базами данных в начале 70-х годов прошлого века.

На данный момент существует огромное количество СППР, разработанных и внедренных в различных областях человеческой деятельности. Темпы их разработок постоянно возрастают.

Однако на сегодняшний день, несмотря на распространенность данных систем, общепризнанное определение данного термина пока не найдено. Следует отметить, что хотя СППР широко применяется во всем мире, на просторах СНГ системам этого типа пока еще не уделяется должное внимание.

Рассмотрим, что же представляет собой система поддержки принятия решений. Как уже было отмечено, данный вопрос является дискуссионным, так же как и вопрос отнесения различных типов систем к классу СППР; мнения по этому поводу часто даже противоречат друг другу. Приведем несколько определений СППР.

Основу СППР составляет комплекс взаимосвязанных моделей с соответствующей информационной поддержкой исследования, экспертные и интеллектуальные системы, включающие опыт решения задач управления и обеспечивающие участие коллектива экспертов в процессе выработки рациональных решений [71].

Система поддержки принятия решений - это диалоговая автоматизированная система, использующая правила принятия решений и соответствующие модели с базами данных, а также интерактивный компьютерный процесс моделирования.

СППР - это средство для "вычисления решений", которое основано "на использовании ряда процедур по обработке данных и суждений, помогающих лицу, принимающему решение (далее - ЛПР), в принятии решения" [72].

СППР - "интерактивные автоматизированные системы, которые помогают ЛПР использовать данные и модели, чтобы решать неструктурированные проблемы" [73].

СППР - "компьютерная информационная система, используемая для поддержки различных видов деятельности при принятии решения в ситуациях, где невозможно или нежелательно иметь автоматические системы, которые полностью выполняют весь процесс принятия решения". СППР не заменяет ЛПР, автоматизируя процесс принятия решения, а оказывает ему помощь в ходе решения поставленной задачи [74].

Следует заметить, что, начиная с первых определений СППР, круг задач, решаемых при их помощи, ограничился слабоструктурированными и неструктурированными.

Определим СППР таким образом: СППР - интерактивная компьютерная система, предназначенная для поддержки принятия решений в слабоструктурированных и неструктурированных проблемах различных видов человеческой деятельности [75].

Существенными концепциями этого определения являются:

- компьютерная интерактивная (т.е. не обуславливающая обязательного непосредственного использования ЛПР системы поддержки принятия решений);
- поддержка принятия решений (решение принимает человек);
- слабоструктурированных и неструктурированных проблем (именно такими проблемами занимаются руководители).

Рассмотрим, что же представляет собой классификация проблем на слабоструктурированные, неструктурированные и структурированные [75, 76].

Неструктурированные задачи имеют только качественное описание, основанное на суждениях ЛПР, количественные зависимости между основными характеристиками задачи не известны.

Структурированные задачи характеризуются существенными зависимостями, которые могут быть выражены количественно.

Слабоструктурированные задачи занимают промежуточное положение и являются "сочетающими количественные и качественные зависимости, причем малоизвестные и неопределенные стороны задачи имеют тенденцию доминировать" [76].

Можно выделить три компонента, составляющие основу классической структуры СППР, которыми она отличается от других типов информационных систем: подсистему интерфейса пользователя, подсистему управления базой данных и подсистему управления базой моделей [75].

Если посмотреть на СППР с функциональной стороны, можно выделить следующие ее компоненты [11, 77]:

- сервер хранилища данных;
- инструментарий OLAP;
- инструментарий Data Mining.

Эти компоненты СППР рассматривают такие основные вопросы: вопрос накопления данных и их моделирования на концептуальном уровне, вопрос эффективной загрузки данных из нескольких независимых источников и вопрос анализа данных.

Можно сказать, что использование оперативной аналитической обработки (систем OLAP) на сегодня ограничивается обеспечением доступа к многомерным данным.

Технология Data Mining представляет в СППР наибольший интерес, поскольку с ее помощью можно провести наиболее глубокий и всесторонний анализ данных и, следовательно, принимать наиболее взвешенные и обоснованные решения.

Классификация СППР

Вопрос классификаций СППР на сегодняшний день является актуальным, продолжаются разработки новых таксономий. Рассмотрим две из них.

Ниже приведена классификация СППР по сходству некоторых признаков (D.J. Power, 2000). С подробным описанием групп можно ознакомиться в [75].

- СППР, ориентированные на данные (Data-driven DSS, Data-oriented DSS);
- СППР, ориентированные на модели (Model-driven DSS);
- СППР, ориентированные на знания (Knowledge-driven DSS);
- СППР, ориентированные на документы (Document-driven DSS);
- СППР, ориентированные на коммуникации и групповые СППР (Communications-Driven ? Group DSS);
- Интер-организованные и Интра-организованные СППР (Inter-Organizational або Intra-Organizational DSS);
- Специфически функциональные СППР или СППР общего назначения (Function-Specific або General Purpose DSS);
- СППР на базе Web (Web-Based DSS).

В зависимости от данных, с которыми работают СППР, выделяют два основных их типа СППР: EIS и DSS [75,78].

EIS (Execution Information System) - информационная система Руководства, ИСР.

СППР этого типа являются оперативными, предназначенными для немедленного реагирования на текущую ситуацию. В большинстве они ориентированы на неподготовленного пользователя, потому имеют упрощенный интерфейс, базовый набор предлагаемых возможностей, фиксированные формы представления информации и перечень решаемых задач. Такие системы основаны на типичных запросах, число которых относительно невелико; отчеты, полученные в результате таких запросов, представляются в максимально удобном виде.

DSS (Decision Support System). К системам этого типа относят многофункциональные системы анализа и исследования данных. Они предполагают глубокую проработку данных, которую можно использовать в процессе принятий решений.

Системы этого типа, в отличие от EIS, рассчитаны на пользователей, имеющих как знания в предметной области, так и возможности использования современных компьютерных технологий. Этим системам присущи черты искусственного интеллекта, за счет

возможности проработки исходных данных в конкретные выводы по поставленной задаче. Такие системы имеет смысл создавать, если есть основания для обобщения и анализа данных и процессов их обработки.

В последнее время к СППР относят только второй тип, т.е. DSS.

Системы этого типа иногда называют динамическими, т.е. они должны быть ориентированы на обработку неожиданных (ad hoc) запросов. Поддержка принятия решений на основе накопленных данных может выполняться в трех базовых сферах [79]. Более подробно с этим материалом можно ознакомиться в [11].

1. Область детализированных данных (OLTP-системы).

   Целью большинства таких систем является поиск информации, это так называемые информационно-поисковые системы. Они могут использоваться в качестве надстроек над системами обработки данных или как хранилища данных.

2. Сфера агрегированных показателей (OLAP-системы).

   Задачами OLAP систем является обобщение, агрегация, гиперкубическое представление информации и многомерный анализ. Это могут быть многомерные СУБД или же реляционные базы с предварительной агрегацией данных.

3. Сфера закономерностей (Data Mining).

Такое деление систем на EIS и DSS не обязательно означает реализацию СППР одного из типов. Они могут существовать параллельно, когда каждая из систем предоставляет свои функции определенной категории пользователей.

Общая схема поддержки принятия решений, предлагаемая в [80], включает:

- помощь ЛПР при оценке состояния управляемой системы и воздействий на нее; выявление предпочтений ЛПР;
- генерацию возможных решений;
- оценку возможных альтернатив, исходя из предпочтений ЛПР;
- анализ последствий принимаемых решений и выбор лучшего с точки зрения ЛПР.

OLAP-системы

В основе концепции OLAP, или оперативной аналитической обработки данных (On-Line Analytical Processing), лежит многомерное концептуальное представление данных (Multidimensional conceptual view).

Термин OLAP введен Коддом (E. F. Codd) в 1993 году. Главная идея данной системы заключается в построении многомерных таблиц, которые могут быть доступны для запросов пользователей. Эти многомерные таблицы или так называемые многомерные кубы строятся на основе исходных и агрегированных данных. И исходные, и агрегированные данные для многомерных таблиц могут храниться как в реляционных, так и в многомерных базах данных. Взаимодействуя с OLAP-системой, пользователь может осуществлять гибкий просмотр информации, получать различные срезы данных, выполнять аналитические операции детализации, свертки, сквозного распределения,

сравнения во времени. Вся работа с OLAP-системой происходит в терминах предметной области.

OLAP-продукты

Сейчас на рынке представлено огромное многообразие OLAP-систем. Разработано несколько классификаций продуктов этого типа: например, классификация по способу хранения данных, по месту нахождения OLAP-машины, по степени готовности к применению. Рассмотрим первую из приведенных классификаций.

Существует три способа хранения данных в OLAP-системах или три архитектуры OLAP-серверов [77]:

- MOLAP (Multidimensional OLAP);
- ROLAP (Relational OLAP);
- HOLAP (Hybrid OLAP).

Таким образом, согласно этой классификации OLAP-продукты могут быть представлены тремя классами систем.

- В случае MOLAP, исходные и многомерные данные хранятся в многомерной БД или в многомерном локальном кубе. Такой способ хранения обеспечивает высокую скорость выполнения OLAP-операций. Но многомерная база в этом случае чаще всего будет избыточной. Куб, построенный на ее основе, будет сильно зависеть от числа измерений. При увеличении количества измерений объем куба будет экспоненциально расти. Иногда это может привести к "взрывному росту" объема данных, парализующему в результате запросы пользователей.
- В ROLAP-продуктах исходные данные хранятся в реляционных БД или в плоских локальных таблицах на файл-сервере. Агрегатные данные могут помещаться в служебные таблицы в той же БД. Преобразование данных из реляционной БД в многомерные кубы происходит по запросу OLAP-средства. При этом скорость построения куба будет сильно зависеть от типа источника данных, и поэтому время отклика системы порой становится неприемлемо большим.
- В случае использования гибридной архитектуры, т.е. в HOLAP-продуктах, исходные данные остаются в реляционной базе, а агрегаты размещаются в многомерной. Построение OLAP-куба выполняется по запросу OLAP-средства на основе реляционных и многомерных данных. Такой подход позволяет избежать взрывного роста данных. При этом можно достичь оптимального времени исполнения клиентских запросов.

Следующая классификация - по месту размещения OLAP-машины. По этому признаку OLAP-продукты делятся на OLAP-серверы и OLAP-клиенты.

В серверных OLAP-средствах вычисления и хранение агрегатных данных выполняются отдельным процессом - сервером. Клиентское приложение получает только результаты запросов к многомерным кубам, которые хранятся на сервере. Некоторые OLAP-серверы поддерживают хранение данных только в реляционных базах, другие - только в многомерных. Многие современные OLAP-серверы поддерживают все три способа хранения данных: MOLAP, ROLAP и HOLAP. Одним из самых распространенным в настоящее время серверным решением является OLAP-сервер корпорации Microsoft. OLAP-клиент устроен по-другому. Построение многомерного куба и OLAP-вычисления выполняются в памяти клиентского компьютера.

С помощью OLAP-сервера может быть организовано физическое хранение обработанной многомерной информации [81], что позволяет быстро выдавать ответы на запросы пользователя. Кроме того, предусматривается преобразование данных из реляционных и других баз в многомерные структуры в режиме реального времени. Каким образом реляционные и многомерные средства работают совместно? OLAP продукты вливаются в существующую корпоративную инфраструктуру путем интегрирования с реляционными системами. Администраторы баз данных либо загружают реляционные данные в многомерный кэш, либо настраивают кэш для доступа к SQL-данным.

В таблице 17.1 приведены сравнительные характеристики различных моделей управления данными [81]:

Таблица 17.1. Сравнительные характеристики различных моделей управления данными

Характеристики	Реляционные СУБД OLTP	Реляционные СУБД СППР/Хранилища данных	Многомерные СУБД OLAP
Типовая операция	Обновление	Отчет	Анализ
Уровень аналитических требований	Низкий	Средний	Высокий
Экраны	Неизменяемые	Определяемые пользователем	Определяемые пользователем
Объем данных на транзакцию	Небольшой	От малого до большого	Большой
Уровень данных	Детальные	Детальные и суммарные	В основном суммарные
Сроки хранения данных	Только текущие	Исторические и текущие	Исторические, текущие и прогнозируемые
Структурные элементы	Записи	Записи	Массивы

Интеграция OLAP и Data Mining

Обе технологии можно рассматривать как составные части процесса поддержки принятия решений. Однако эти технологии как бы движутся в разных направлениях: OLAP сосредотачивает внимание исключительно на обеспечении доступа к многомерным данным, а методы Data Mining в большинстве случаев работают с плоскими одномерными таблицами и реляционными данными.

Интеграция технологий OLAP и Data Mining "обогащает" функциональность и одной, и другой технологии. Эти два вида анализа должны быть тесно объединены, чтобы интегрированная технология могла обеспечивать одновременно многомерный доступ и поиск закономерностей. По словам N. Raden, "многие компании создали ... прекрасные хранилища данных, идеально разложив по полочкам горы неиспользуемой информации,

которая сама по себе не обеспечивает ни быстрой, ни достаточно грамотной реакции на рыночные события" [82].

K. Parsaye [83] вводит составной термин "OLAP Data Mining" (многомерный Data Mining) для обозначения такого объединения.

Средство многомерного интеллектуального анализа данных должно находить закономерности как в детализированных, так и в агрегированных с различной степенью обобщения данных. Анализ многомерных данных должен строиться над специального вида гиперкубом, ячейки которого содержат не произвольные численные значения (количество событий, объем продаж, сумма собранных налогов), а числа, определяющие вероятность соответствующего сочетания значений атрибутов. Проекции такого гиперкуба (исключающие из рассмотрения отдельные измерения) также должны исследоваться на предмет поиска закономерностей. J. Han предлагает еще более простое название - "OLAP Mining" и выдвигает несколько вариантов интеграции двух технологий.

1. "Cubing then mining". Возможность выполнения интеллектуального анализа должна обеспечиваться над любым результатом запроса к многомерному концептуальному представлению, то есть над любым фрагментом любой проекции гиперкуба показателей.
2. "Mining then cubing". Подобно данным, извлеченным из хранилища, результаты интеллектуального анализа должны представляться в гиперкубической форме для последующего многомерного анализа.
3. "Cubing while mining". Этот гибкий способ интеграции позволяет автоматически активизировать однотипные механизмы интеллектуальной обработки над результатом каждого шага многомерного анализа (перехода между уровнями обобщения, извлечения нового фрагмента гиперкуба и т.д.).

На сегодняшний день немногие производители реализуют Data Mining для многомерных данных. Кроме того, некоторые методы Data Mining, например, метод ближайших соседей или байесовская классификация, в силу их неспособности работать с агрегированными данными неприменимы к многомерным данным.

Хранилища данных

Информационные системы современных предприятий часто организованы таким образом, чтобы минимизировать время ввода и корректировки данных, т.е. организованы не оптимально с точки зрения проектирования базы данных. Такой подход усложняет доступ к историческим (архивным) данным. Изменения структур в базах данных информационных систем очень трудоемки, а иногда попросту невозможны.

В то же время, для успешного ведения современного бизнеса необходима актуальная информация, предоставляемая в удобном для анализа виде и в реальном масштабе времени. Доступность такой информации позволяет как оценивать текущее положение дел, так и делать прогнозы на будущее, следовательно, принимать более взвешенные и обоснованные решения. К тому же, основой для принятия решений должны быть реальные данные.

Если данные хранятся в базах данных различных информационных систем предприятия, при их анализе возникает ряд сложностей, в частности, значительно возрастает время, необходимое для обработки запросов; могут возникать проблемы с поддержкой

различных форматов данных, а также с их кодированием; невозможность анализа длительных рядов ретроспективных данных и т.д.

Эта проблема решается путем создания хранилища данных. Задачей такого хранилища является интеграция, актуализация и согласование оперативных данных из разнородных источников для формирования единого непротиворечивого взгляда на объект управления в целом. На основе хранилищ данных возможно составление всевозможной отчетности, а также проведение оперативной аналитической обработки и Data Mining.

Билл Инмон (Bill Inmon) определяет хранилища данных как "предметно ориентированные, интегрированные, неизменчивые, поддерживающие хронологию наборы данных, организованные с целью поддержки управления" и призванные выступать в роли "единого и единственного источника истины", который обеспечивает менеджеров и аналитиков достоверной информацией, необходимой для оперативного анализа и принятия решений [84].

Предметная ориентация хранилища данных означает, что данные объединены в категории и сохраняются соответственно областям, которые они описывают, а не применениям, их использующим.

Интегрированность означает, что данные удовлетворяют требованиям всего предприятия, а не одной функции бизнеса. Этим хранилище данных гарантирует, что одинаковые отчеты, сгенерированные для разных аналитиков, будут содержать одинаковые результаты.

Привязка ко времени означает, что хранилище можно рассматривать как совокупность "исторических" данных: возможно восстановление данных на любой момент времени. Атрибут времени явно присутствует в структурах хранилища данных.

Неизменность означает, что, попав один раз в хранилище, данные там сохраняются и не изменяются. Данные в хранилище могут лишь добавляться.

Ричард Хакаторн, другой основоположник этой концепции, писал, что цель Хранилищ Данных - обеспечить для организации "единый образ существующей реальности" [86].

Другими словами, хранилище данных представляет собой своеобразный накопитель информации о деятельности предприятия.

Данные в хранилище представлены в виде многомерных структур под названием "звезда" или "снежинка".

Преимущества использования хранилищ данных

Хранилище данных имеет преимущества в сравнении с использованием оперативных систем или баз данных, в [88] приведены следующие из них:

- В отличие от оперативных систем, хранилище данных содержит информацию за весь требуемый временной интервал - вплоть до нескольких десятилетий - в едином информационном пространстве, что делает такие хранилища идеальной основой для выявления трендов, сезонных зависимостей и других важных аналитических показателей.

- Как правило, информационные системы предприятия хранят и представляют аналогичные данные по-разному. Например, одни и те же показатели могут храниться в различных единицах измерения. Одна и та же продукция или одни и те же клиенты могут именоваться по-разному. В системах хранилищ несоответствия в данных устраняются на этапе сбора информации и погружения ее в единую базу данных. При этом организуются единые справочники, все показатели в которых приводятся к одинаковым единицам измерения.
- Очень часто оперативные системы вследствие ошибок операторов содержат некоторое количество неверных данных. На этапе помещения в хранилище данных информация предварительно обрабатывается. Данные по специальной технологии проверяются на соответствие заданным ограничениям и при необходимости корректируются (очищаются). Технология обеспечивает построение аналитических отчетов на основе надежных данных и своевременное оповещение администратора хранилища об ошибках во входящей информации.
- Универсализация доступа к данным. Хранилище данных предоставляет уникальную возможность получать любые отчеты о деятельности предприятия на основе одного источника информации. Это позволяет интегрировать данные, вводимые и накапливаемые в различных оперативных системах, легко и просто сравнивать их. При этом в процессе создания отчетов пользователь не связан различиями в доступе к данным оперативных систем.
- Ускорение получения аналитических отчетов. Получение отчетов при помощи средств, предоставляемых оперативными системами, - способ неоптимальный. Эти системы затрачивают значительное время на агрегирование информации (расчет суммарных, средних, минимальных, максимальных значений). Кроме того, в текущей базе оперативной системы находятся только самые необходимые и свежие данные, в то время как информация за прошлые периоды помещается в архив. Если данные приходится получать из архива, продолжительность построения отчета возрастает еще в два-три раза. Следует также учитывать, что сервер оперативной системы зачастую не обеспечивает необходимую производительность при одновременном построении сложных отчетов и вводе информации. Это может катастрофически сказываться на работе предприятия, так как операторы не смогут оформлять накладные, фиксировать отгрузку или получение продукции в то время, когда выполняется построение очередного отчета. Хранилище данных позволяет решить эти проблемы. Во-первых, работа сервера хранилища не мешает работе операторов. Во-вторых, в хранилище помимо детальной информации содержатся и заранее рассчитанные агрегированные значения. В-третьих, в хранилище архивная информация всегда доступна для включения в отчеты. Все это позволяет значительно сократить время создания отчетов и избежать проблем в оперативной работе.
- Построение произвольных запросов. Информацию в хранилище данных недостаточно только централизовать и структурировать. Аналитику нужны средства визуализации этой информации, инструмент, с помощью которого легко получать данные, необходимые для принятия своевременных решений. Одно из главных требований любого аналитика - простота формирования отчетов и их наглядность. В случае оперативных систем построение отчетов часто лишено гибкости; чтобы создать новый отчет, приходится задействовать специалистов ИТ-отдела, которые объединяют данные нескольких систем. В случае же использования хранилища данных решение проблемы предоставляет технология OLAP (On-Line Analytical Processing). Эта технология обеспечивает доступ к данным в терминах, привычных для аналитика. Технология OLAP базируется на концепции многомерного представления данных. Действительно, каждое числовое значение, содержащееся в хранилище данных, имеет до нескольких десятков атрибутов (например, количество продаж определенным менеджером в определенном регионе на определенную дату и т.п.). Таким образом, можно считать, что работа идет с многомерными структурами данных (многомерными кубами), в которых числовые

значения расположены на пересечении нескольких измерений. Именно этот подход используется в OLAP-системах. Они предоставляют гибкие средства навигации по многомерным структурам - так называемые OLAP-манипуляции. С их помощью аналитик может получать различные срезы данных, "крутить" данные.

Как видно из перечисленных преимуществ использования технологии хранилищ данных, большая их часть может существенно упростить, повысить скорость и качественно улучшить процесс Data Mining. Таким образом, комплексное внедрение этих технологий дает разработчикам и пользователям неоспоримые преимущества перед использованием разрозненных баз данных различных информационных систем при создании систем поддержки принятия решений.

Процесс Data Mining. Начальные этапы

Процесс Data Mining является своего рода исследованием. Как любое исследование, этот процесс состоит из определенных этапов, включающих элементы сравнения, типизации, классификации, обобщения, абстрагирования, повторения.

Процесс Data Mining неразрывно связан с процессом принятия решений.

Процесс Data Mining строит модель, а в процессе принятия решений эта модель эксплуатируется.

Рассмотрим традиционный процесс Data Mining. Он включает следующие этапы:

- анализ предметной области;
- постановка задачи;
- подготовка данных;
- построение моделей;
- проверка и оценка моделей;
- выбор модели;
- применение модели;
- коррекция и обновление модели.

В этой лекции мы подробно рассмотрим первые три этапа процесса Data Mining, остальные этапы будут рассмотрены в следующей лекции.

Этап 1. Анализ предметной области

Исследование - это процесс познания определенной предметной области, объекта или явления с определенной целью.

Процесс исследования заключается в наблюдении свойств объектов с целью выявления и оценки важных, с точки зрения субъекта-исследователя, закономерных отношений между показателями данных свойств.

Решение любой задачи в сфере разработки программного обеспечения должно начинаться с изучения предметной области.

Предметная область - это мысленно ограниченная область реальной действительности, подлежащая описанию или моделированию и исследованию.

Предметная область состоит из объектов, различаемых по свойствам и находящихся в определенных отношениях между собой или взаимодействующих каким-либо образом.

Предметная область - это часть реального мира, она бесконечна и содержит как существенные, так и не значащие данные, с точки зрения проводимого исследования.

Исследователю необходимо уметь выделить существенную их часть. Например, при решении задачи "Выдавать ли кредит?" важными являются все данные про частную жизнь клиента, вплоть до того, имеет ли работу супруг, есть ли у клиента несовершеннолетние

дети, каков уровень его образования и т.д. Для решения другой задачи банковской деятельности эти данные будут абсолютно неважны. Существенность данных, таким образом, зависит от выбора предметной области.

В процессе изучения предметной области должна быть создана ее модель. Знания из различных источников должны быть формализированы при помощи каких-либо средств.

Это могут быть текстовые описания предметной области или специализированные графические нотации. Существует большое количество методик описания предметной области: например, методика структурного анализа SADT и основанная на нем IDEF0, диаграммы потоков данных Гейна-Сарсона, методика объектно-ориентированного анализа UML и другие. Модель предметной области описывает процессы, происходящие в предметной области, и данные, которые в этих процессах используются.

Это первый этап процесса Data Mining. Но от того, насколько верно смоделирована предметная область, зависит успех дальнейшей разработки приложения Data Mining.

Этап 2. Постановка задачи

Постановка задачи Data Mining включает следующие шаги:

- формулировка задачи;
- формализация задачи.

Постановка задачи включает также описание статического и динамического поведения исследуемых объектов.

Пример задачи. При продвижении нового товара на рынок необходимо определить, какая группа клиентов фирмы будет наиболее заинтересована в данном товаре.

Описание статики подразумевает описание объектов и их свойств.

Пример. Клиент является объектом. Свойства объекта "клиент": семейное положение, доход за предыдущий год, место проживания.

При описании динамики описывается поведение объектов и те причины, которые влияют на их поведение.

Пример. Клиент покупает товар А. При появлении нового товара В клиент уже не покупает товар А, а покупает только товар В. Появление товара В изменило поведение клиента. Динамика поведения объектов часто описывается вместе со статикой.

Технология Data Mining не может заменить аналитика и ответить на те вопросы, которые не были заданы. Поэтому постановка задачи является необходимым этапом процесса Data Mining, поскольку именно на этом этапе мы определяем, какую же задачу необходимо решить. Иногда этапы анализа предметной области и постановки задачи объединяют в один этап.

Этап 3. Подготовка данных

Цель этапа: разработка базы данных для Data Mining.

Понятие данных было рассмотрено в лекции № 2 этого курса лекций.

Подготовка данных является важнейшим этапом, от качества выполнения которого зависит возможность получения качественных результатов всего процесса Data Mining. Кроме того, следует помнить, что на этап подготовки данных, по некоторым оценкам, может быть потрачено до 80% всего времени, отведенного на проект.

Рассмотрим подробно, что же представляет собой этот этап.

1. Определение и анализ требований к данным

На этом этапе осуществляется так называемое моделирование данных, т.е. определение и анализ требований к данным, которые необходимы для осуществления Data Mining. При этом изучаются вопросы распределения пользователей (географическое, организационное, функциональное); вопросы доступа к данным, которые необходимы для анализа, необходимость во внешних и/или внутренних источниках данных; а также аналитические характеристики системы (измерения данных, основные виды выходных документов, последовательность преобразования информации и др.).

2. Сбор данных

Наличие в организации хранилища данных делает анализ проще и эффективней, его использование, с точки зрения вложений, обходится дешевле, чем использование отдельных баз данных или витрин данных. Однако далеко не все предприятия оснащены хранилищами данных. В этом случае источником для исходных данных являются оперативные, справочные и архивные БД, т.е. данные из существующих информационных систем.

Также для Data Mining может потребоваться информация из информационных систем руководителей, внешних источников, бумажных носителей, а также знания экспертов или результаты опросов.

Следует помнить, что в процессе подготовки данных аналитики и разработчики не должны привязываться к показателям, которые есть в наличии, и описать максимальное количество факторов и признаков, влияющих на анализируемый процесс.

На этом этапе осуществляется кодирование некоторых данных. Допустим, одним из атрибутов клиента является уровень дохода, который должен быть представлен в системе одним из значений: очень низким, низким, средним, высоким, очень высоким. Необходимо определить градации уровня дохода, в этом процессе потребуется сотрудничество аналитика с экспертом в предметной области. Возможно, для таких преобразований данных потребуется написание специальных процедур.

Определение необходимого количества данных

При определении необходимого количества данных следует учитывать, являются ли данные упорядоченными или нет.

Если данные упорядочены и мы имеем дело с временными рядами, желательно знать, включает ли такой набор данных сезонную/цикличную компоненту. В случае присутствия

в наборе данных сезонной/цикличной компоненты, необходимо иметь данные как минимум за один сезон/цикл.

Если данные не упорядочены, то есть события из набора данных не связаны по времени, в ходе сбора данных следует соблюдать следующие правила.

Количество записей в наборе. Недостаточное количество записей в наборе данных может стать причиной построения некорректной модели. С точки зрения статистики, точность модели увеличивается с увеличением количества исследуемых данных. Возможно, некоторые данные являются устаревшими или описывают какую-то нетипичную ситуацию, и их нужно исключить из базы данных. Алгоритмы, используемые для построения моделей на сверхбольших базах данных, должны быть масштабируемыми.

Соотношение количества записей в наборе и количества входных переменных. При использовании многих алгоритмов необходимо определенное (желательное) соотношение входных переменных и количества наблюдений. Количество записей (примеров) в наборе данных должно быть значительно больше количества факторов (переменных).

Набор данных должен быть репрезентативным и представлять как можно больше возможных ситуаций. Пропорции представления различных примеров в наборе данных должны соответствовать реальной ситуации.

3. Предварительная обработка данных

Анализировать можно как качественные, так и некачественные данные. Результат будет достигнут и в том, и в другом случае. Для обеспечения качественного анализа необходимо проведение предварительной обработки данных, которая является необходимым этапом процесса Data Mining.

Оценивание качества данных. Данные, полученные в результате сбора, должны соответствовать определенным критериям качества. Таким образом, можно выделить важный подэтап процесса Data Mining - оценивание качества данных.

Качество данных (Data quality) - это критерий, определяющий полноту, точность, своевременность и возможность интерпретации данных.

Данные могут быть высокого качества и низкого качества, последние - это так называемые грязные или "плохие" данные.

Данные высокого качества - это полные, точные, своевременные данные, которые поддаются интерпретации.

Такие данные обеспечивают получение качественного результата: знаний, которые смогут поддерживать процесс принятия решений.

О важности обсуждаемой проблемы говорит тот факт, что "серьезное отношение к качеству данных" занимает первое место среди десяти основных тенденций, прогнозирующихся в начале 2005 года в области Business Intelligence и Хранилищ данных компанией Knightsbridge Solutions. Этот прогноз был сделан в январе 2005 года, а в июне 2005 года Даффи Брансон (Duffie Brunson), один из руководителей компании Knightsbridge Solutions, проанализировал состоятельность данных ранее прогнозов.

Сокращенное изложение его анализа представлено в [90]. Ниже изложен прогноз и его анализ полгода спустя.

Прогноз. Многие компании стали обращать больше внимания на качество данных, поскольку низкое качество стоит денег в том смысле, что ведет к снижению производительности, принятию неправильных бизнес-решений и невозможности получить желаемый результат, а также затрудняет выполнение требований законодательства. Поэтому компании действительно намерены предпринимать конкретные действия для решения проблем качества данных.

Реальность. Данная тенденция сохраняется, особенно в индустрии финансовых услуг. В первую очередь это относится к фирмам, старающимся выполнять соглашение Basel II. Некачественные данные не могут использоваться в системах оценки рисков, которые применяются для установки цен на кредиты и вычисления потребностей организации в капитале. Интересно отметить, что существенно изменились взгляды на способы решения проблемы качества данных. Вначале менеджеры обращали основное внимание на инструменты оценки качества, считая, что "собственник" данных должен решать проблему на уровне источника, например, очищая данные и переобучая сотрудников. Но сейчас их взгляды существенно изменились. Понятие качества данных гораздо шире, чем просто их аккуратное введение в систему на первом этапе. Сегодня уже многие понимают, что качество данных должно обеспечиваться процессами извлечения, преобразования и загрузки (Extraction, Transformation, Loading - ETL), а также получения данных из источников, которые подготавливают данные для анализа.

Рассмотрим понятия качества данных более детально.

Данные низкого качества, или грязные данные - это отсутствующие, неточные или бесполезные данные с точки зрения практического применения (например, представленные в неверном формате, не соответствующем стандарту). Грязные данные появились не сегодня, они возникли одновременно с системами ввода данных.

Грязные данные могут появиться по разным причинам, таким как ошибка при вводе данных, использование иных форматов представления или единиц измерения, несоответствие стандартам, отсутствие своевременного обновления, неудачное обновление всех копий данных, неудачное удаление записей-дубликатов и т.д. Необходимо оценить стоимость наличия грязных данных; другими словами, наличие грязных данных может действительно привести к финансовым потерям и юридической ответственности, если их присутствие не предотвращается или они не обнаруживаются и не очищаются [91].

Для более подробного знакомства с грязными данными можно рекомендовать [92], где представлена таксономия 33 типов грязных данных и также разработана таксономия методов предотвращения или распознавания и очистки данных. Описаны различные типы грязных данных, среди них выделены следующие группы:

- грязные данные, которые могут быть автоматически обнаружены и очищены;
- данные, появление которых может быть предотвращено;
- данные, которые непригодны для автоматического обнаружения и очистки;
- данные, появление которых невозможно предотвратить.

Поэтому важно понимать, что специальные средства очистки могут справиться не со всеми видами грязных данных.

Рассмотрим наиболее распространенные виды грязных данных:

- пропущенные значения;
- дубликаты данных;
- шумы и выбросы.

Пропущенные значения (Missing Values).

Некоторые значения данных могут быть пропущены в связи с тем, что:

- данные вообще не были собраны (например, при анкетировании скрыт возраст);
- некоторые атрибуты могут быть неприменимы для некоторых объектов (например, атрибут "годовой доход" неприменим к ребенку).

Как мы можем поступить с пропущенными данными?

- Исключить объекты с пропущенными значениями из обработки.
- Рассчитать новые значения для пропущенных данных.
- Игнорировать пропущенные значения в процессе анализа.
- Заменить пропущенные значения на возможные значения.

Дублирование данных (Duplicate Data).

Набор данных может включать продублированные данные, т.е. дубликаты.

Дубликатами называются записи с одинаковыми значениями всех атрибутов.

Наличие дубликатов в наборе данных может являться способом повышения значимости некоторых записей. Такая необходимость иногда возникает для особого выделения определенных записей из набора данных. Однако в большинстве случаев, продублированные данные являются результатом ошибок при подготовке данных.

Как мы можем поступить с продублированными данными?

Существует два варианта обработки дубликатов. При первом варианте удаляется вся группа записей, содержащая дубликаты. Этот вариант используется в том случае, если наличие дубликатов вызывает недоверие к информации, полностью ее обесценивает.

Второй вариант состоит в замене группы дубликатов на одну уникальную запись.

Шумы и выбросы.

Выбросы - резко отличающиеся объекты или наблюдения в наборе данных.

Шумы и выбросы являются достаточно общей проблемой в анализе данных. Выбросы могут как представлять собой отдельные наблюдения, так и быть объединенными в некие группы. Задача аналитика - не только их обнаружить, но и оценить степень их влияния на

результаты дальнейшего анализа. Если выбросы являются информативной частью анализируемого набора данных, используют робастные методы и процедуры.

Достаточно распространена практика проведения двухэтапного анализа - с выбросами и с их отсутствием - и сравнение полученных результатов.

Различные методы Data Mining имеют разную чувствительность к выбросам, этот факт необходимо учитывать при выборе метода анализа данных. Также некоторые инструменты Data Mining имеют встроенные процедуры очистки от шумов и выбросов.

Визуализация данных позволяет представить данные, в том числе и выбросы, в графическом виде. Пример наличия выбросов изображен на диаграмме рассеивания на рис. 18.1. Мы видим несколько наблюдений, резко отличающихся от других (находящихся на большом расстоянии от большинства наблюдений).

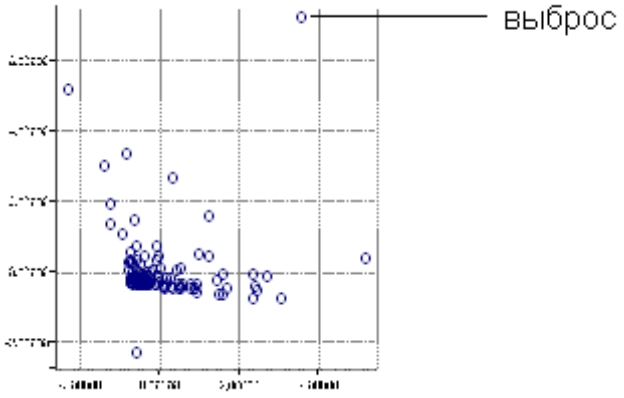

Рис. 18.1. Пример набора данных с выбросами

Очевидно, что результаты Data Mining на основе грязных данных не могут считаться надежными и полезными. Однако наличие таких данных не обязательно означает необходимость их очистки или же предотвращения появления. Всегда должен быть разумный выбор между наличием грязных данных и стоимостью и/или временем, необходимым для их очистки.

Очистка данных

Очистка данных (data cleaning, data cleansing или scrubbing) занимается выявлением и удалением ошибок и несоответствий в данных с целью улучшения качества данных.

Проблемы с качеством встречаются в отдельных наборах данных - таких как файлы и базы данных. Когда интеграции подлежит множество источников данных (например в Хранилищах, интегрированных системах баз данных или глобальных информационных Интернет-системах), необходимость в очистке данных существенно возрастает. Это происходит оттого, что источники часто содержат разрозненные данные в различном представлении. Для обеспечения доступа к точным и согласованным данным необходима консолидация различных представлений данных и исключение дублирующейся информации. Специальные средства очистки обычно имеют дело с конкретными областями - в основном это имена и адреса - или же с исключением дубликатов.

Преобразования обеспечиваются либо в форме библиотеки правил, либо пользователем в интерактивном режиме. Преобразования данных могут быть автоматически получены с помощью средств согласования схемы [93].

Метод очистки данных должен удовлетворять ряду критериев [93].

1. Он должен выявлять и удалять все основные ошибки и несоответствия, как в отдельных источниках данных, так и при интеграции нескольких источников.
2. Метод должен поддерживаться определенными инструментами, чтобы сократить объемы ручной проверки и программирования, и быть гибким в плане работы с дополнительными источниками.
3. Очистка данных не должна производиться в отрыве от связанных со схемой преобразования данных, выполняемых на основе сложных метаданных.
4. Функции маппирования для очистки и других преобразований данных должны быть определены декларативным образом и подходить для использования в других источниках данных и в обработке запросов.
5. Инфраструктура технологического процесса должна особенно интенсивно поддерживаться для Хранилищ данных, обеспечивая эффективное и надежное выполнение всех этапов преобразования для множества источников и больших наборов данных.

На сегодняшний день интерес к очистке данных возрастает. Целый ряд исследовательских групп занимается общими проблемами, связанными с очисткой данных, в том числе, со специфическими подходами к Data Mining и преобразованию данных на основании сопоставления схемы. В последнее время некоторые исследования коснулись единого, более сложного подхода к очистке данных, включающего ряд аспектов преобразования данных, специфических операторов и их реализации.

Этапы очистки данных

В целом, очистка данных включает следующие этапы [93] (ниже изложено краткое описание содержание этих этапов, в этом же источнике можно найти подробное их описание).

1. Анализ данных.
2. Определение порядка и правил преобразования данных.
3. Подтверждение.
4. Преобразования.
5. Противоток очищенных данных.

Этап № 1. Анализ данных.

Подробный анализ данных необходим для выявления подлежащих удалению видов ошибок и несоответствий. Здесь можно использовать как ручную проверку данных или их шаблонов, так и специальные программы для получения метаданных о свойствах данных и определения проблем качества.

Этап № 2. Определение порядка и правил преобразования данных.

В зависимости от числа источников данных, степени их неоднородности и загрязненности, данные могут требовать достаточно обширного преобразования и

очистки. Иногда для отображения источников общей модели данных используется трансляция схемы; для Хранилищ данных обычно используется реляционное представление. Первые шаги по очистке могут уточнить или изменить описание проблем отдельных источников данных, а также подготовить данные для интеграции. Дальнейшие шаги должны быть направлены на интеграцию схемы/данных и устранение проблем множественных элементов, например, дубликатов. Для Хранилищ в процессе работы по определению ETL должны быть определены методы контроля и поток данных, подлежащий преобразованию и очистке.

Преобразования данных, связанные со схемой, так же как и этапы очистки, должны, насколько возможно, определяться с помощью декларативного запроса и языка маппирования, обеспечивая, таким образом, автоматическую генерацию кода преобразования. К тому же, в процессе преобразования должна существовать возможность запуска написанного пользователем кода очистки и специальных средств. Этапы преобразования могут требовать обратной связи с пользователем по тем элементам данных, для которых отсутствует встроенная логика очистки.

Этап № 3. Подтверждение.

На этом этапе определяется правильность и эффективность процесса и определений преобразования. Это осуществляется путем тестирования и оценивания, например, на примере или на копии данных источника, - чтобы выяснить, необходимо ли как-то улучшить эти определения. При анализе, проектировании и подтверждении может потребоваться множество итераций, например, в связи с тем, что некоторые ошибки становятся заметны только после проведения определенных преобразований.

Этап № 4. Преобразования.

На этом этапе осуществляется выполнение преобразований либо в процессе ETL для загрузки и обновления Хранилища данных, либо при ответе на запросы по множеству источников.

Этап № 5. Противоток очищенных данных.

После того как ошибки отдельного источника удалены, загрязненные данные в исходных источниках должны замениться на очищенные, для того чтобы улучшенные данные попали также в унаследованные приложения и в дальнейшем при извлечении не требовали дополнительной очистки. Для Хранилищ очищенные данные находятся в области хранения данных.

Такой процесс преобразования требует больших объемов метаданных (схем, характеристик данных уровня схемы, определений технологического процесса и др.). Для согласованности, гибкости и упрощения использования в других случаях, эти метаданные должны храниться в репозитории на основе СУБД. Для поддержки качества данных подробная информация о процессе преобразования должна записываться как в репозиторий, так и в трансформированные элементы данных, в особенности информация о полноте и свежести исходных данных и происхождения информации о первоисточнике трансформированных объектов и произведенных с ними изменениях. Например, на рис. 3 производная таблица Потребители содержит атрибуты Идентификатор и Номер, позволяя проследить путь исходных записей.

Далее подробно описываются возможные методы анализа данных (выявления конфликтов), определения преобразований и разрешения конфликтов. Конфликты наименований обычно разрешаются путем переименования; структурные конфликты требуют частичного перестроения и унификации исходных схем.

Выводы

В этой лекции мы начали рассматривать этапы процесса Data Mining, в частности, уделили много внимания этапу подготовки данных и их предварительной обработке, подробно остановились на понятии грязных данных и этапах очистки данных.

Внимание, отведенное обсуждению этой проблемы, вызвано необходимостью использования при непосредственном проведении Data Mining максимально полных, точных, своевременных данных, поддающихся интерпретации, т.е. данных высокого качества.

В следующей лекции мы рассмотрим инструменты очистки данных, их сильные стороны и проблемы.

Процесс Data Mining. Очистка данных

Инструменты очистки данных

На сегодняшний день рынок программного обеспечения предлагает большой выбор средств, целью которых является преобразование и очистка данных.

Рассмотрим две классификации таких средств.

Эрхард Рам (Erhard Ram) и Хонг Хай До (Hong Hai Do) определяют следующую классификацию средств очистки и соответствующие им инструментов.

1. Средства анализа и модернизации данных.
2. Специальные средства очистки:
    - очистка специфической области;
    - исключение дубликатов.
3. Инструменты ETL.

В [93] изложено подробное описание этой классификации, ниже приведено ее краткое описание.

1. Средства анализа и модернизации данных

Средства анализа и модернизации, обрабатывающие данные с целью выявления ошибок, несоответствий и определения необходимых очищающих преобразований, согласно этой классификации, могут быть разделены на средства профайлинга данных и средства Data Mining.

Профайлинг данных. MIGRATIONARCHITECT (Evoke Software) является одним из немногих коммерческих инструментов этой категории. Для каждого атрибута он определяет следующие метаданные: тип данных, длину, множество элементов, дискретные значения и их процентное отношение, минимальные и максимальные значения, утраченные значения и уникальность. MIGRATIONARCHITECT также может помочь в разработке целевой схемы для миграции данных.

Средства Data Mining. Например, WIZRULE (WizSoft) и DATAMININGSUITE (Information Discovery) выводят отношения между атрибутами и их значениями, вычисляют уровень достоверности, отражающий число квалифицирующих рядов.

WIZRULE может отражать три вида правил: математическую формулу, правила if-then ("если-то") и правила правописания, отсеивающие неверно написанные имена, - например, "значение Edinburgh 52 раза встречается в поле Потребитель; 2 случая содержат одинаковые значения". WIZRULE также автоматически указывает на отклонения от набора обнаруженных правил как на возможные ошибки.

Средства модернизации данных, например, INTEGRITY (Vality), используют обнаруженные шаблоны и правила для определения и выполнения очищающих преобразований, т.е. модернизируют унаследованные данные. В INTEGRITY элементы данных подвергаются ряду обработок - разбору, типизации, анализу шаблонов и частот.

Результатом этих действий является табличное представление содержимого полей, их шаблонов и частот, в зависимости от того, какие шаблоны можно выбрать для стандартизации данных. Для определения очищающих преобразований INTEGRITY предлагает язык с набором операторов для преобразований столбцов (например, перемещения, расщепления, удаления) и рядов. INTEGRITY идентифицирует и консолидирует записи с помощью метода статистического соответствия. При вычислении оценок для упорядочивания соответствий, по которым пользователь отбирает настоящие дубликаты, используются взвешенные коэффициенты.

2. Специальные средства очистки

Специальные средства очистки обычно имеют дело с конкретными областями - в основном это имена и адреса - или же с исключением дубликатов. Преобразования либо обеспечиваются заранее, в форме библиотеки правил, либо в интерактивном режиме, пользователем. Преобразования данных могут быть автоматически получены и с помощью средств согласования схемы.

Ряд средств ориентирован на специфическую область - например, на очистку данных по именам и адресам или на специфические фазы очистки - например, анализ данных или исключение дубликатов. Благодаря своей ограниченной области применения, специализированные средства обычно очень эффективны, однако для работы с широким спектром проблем преобразования и очистки они нуждаются в дополнении другими инструментами.

2.1. Очистка специфической области

Имена и адреса записаны в различных источниках и обычно имеют множество элементов, поэтому поиск соответствий их конкретному потребителю имеет большое значение для управления отношениями с клиентами. Ряд коммерческих инструментов, например IDCENTRIC (First Logic), PUREINTEGRATE (Oracle), QUICKADDRESS (QAS Systems), REUNION (Pitney Bowes) и TRILLIUM (Trillium Software), предназначены для очистки именно таких данных. Они содержат соответствующие методы: например извлечения и преобразования имен и адресов в отдельные стандартные элементы, проверку допустимости названий улиц, городов и индексов, вместе с возможностями сопоставления на основе очищенных данных. Они включают огромную библиотеку предопределенных правил относительно проблем, часто встречающихся в данных такого рода. К примеру, модуль извлечение TRILLIUM (парсер) и модуль сопоставления содержат свыше 200000 бизнес-правил. Эти инструменты обеспечивают и возможности настройки или расширения библиотеки правил за счет правил, определенных пользователем для собственных специфических случаев.

2.2. Исключение дубликатов

Примерами средств для выявления и удаления дубликатов являются DATACLEANSER (EDD), MERGE/PURGELIBRARY (Sagent/QMSoftware), MATCHIT (HelpITSystems) и MASTERMERGE (Pitney Bowes). Обычно они требуют, чтобы источник данных уже был очищен и подготовлен для согласования. Ими поддерживается несколько подходов к согласованию значений атрибутов; а такие средства как DATACLEANSER и MERGE/PURGE LIBRARY позволяют также интегрировать правила согласования, определенные пользователем.

3. Инструменты ETL

Средства ETL обеспечивают возможность сложных преобразований и большей части технологического процесса преобразования и очистки данных. Общей проблемой средств ETL являются ограниченные за счет собственных API и форматов метаданных возможности взаимодействия, усложняющие совместное использование различных средств.

Многие коммерческие инструменты поддерживают процесс ETL для Хранилищ данных на комплексном уровне, например, COPYMANAGER (Information Builders), DATASTAGE (Informix/Ardent), EXTRACT (ETI), POWERMART (Informatica), DECISIONBASE (CA/Platinum), DATATRANSFORMATIONSERVICE (Microsoft), METASUITE (Minerva/Carleton), SAGENTSOLUTIONPLATFORM (Sagent) и WAREHOUSEADMINISTRATOR (SAS). Для единообразного управления всеми метаданными по источникам данных, целевым схемам, маппированиям, скриптам и т.д. они используют репозиторий на основе СУБД. Схемы и данные извлекаются из оперативных источников данных как через "родной" файл и шлюзы СУБД DBMS, так и через стандартные интерфейсы - например ODBC и EDA. Преобразования данных определяются через простой графический интерфейс. Для определения индивидуальных шагов маппирования обычно существует собственный язык правил и комплексная библиотека предопределенных функций преобразования. Эти средства поддерживают и повторное использование существующих преобразованных решений, например внешних процедур C/C++ с помощью имеющегося в них интерфейса для их интеграции во внутреннюю библиотеку преобразований. Процесс преобразования выполняется либо системой, интерпретирующей специфические преобразования в процессе работы, либо откомпилированным кодом. Все средства на базе системы (например, COPYMANAGER, DECISIONBASE, POWERMART, DATASTAGE, WAREHOUSEADMINISTRATOR), имеют планировщик и поддерживают технологические процессы со сложными зависимостями выполнения между этапами преобразования. Технологический процесс может также помогать работе внешних средств (скажем, в специфических задачах очистки это будут очистка имен/адресов или исключение дубликатов).

Средства ETL обычно содержат мало встроенных возможностей очистки, но позволяют пользователю определять функциональность очистки через собственный API. Как правило, анализ данных для автоматического выявления ошибок и несоответствий в данных не поддерживается. Тем не менее, пользователи могут реализовывать такую логику при работе с метаданными и путем определения характеристик содержимого с помощью функций агрегации (sum, count, min, max, median, variance, deviation,:). Поставляемая библиотека преобразований отвечает различным потребностям преобразования и очистки - например конверсию типов данных (в частности, переформатирование данных), строковые функции (расщепление, слияние, замена, поиск по подстроке), арифметические, научные и статистические функции и т.д. Извлечение значений из атрибутов свободного формата автоматизировано неполностью, и пользователю приходится определять разделители, разграничивающие фрагменты значений.

Языки правил обычно охватывают конструкции if-then и case, способствующие обработке исключений в значениях данных, - неверных написаний, аббревиатур, утраченных или зашифрованных значений и значений вне допустимого диапазона. Эти проблемы могут также решаться с помощью функциональных возможностей по выборке данных из таблиц. Поддержка согласования элементов данных обычно ограничена использованием

возможностей объединения и нескольких простых строковых функций соответствия, например точного или группового соответствия или soundex. Тем не менее, определенные пользователем функции соответствия полей, так же как и функции корреляции сходства полей, могут программироваться и добавляться во внутреннюю библиотеку преобразований.

Другая классификация средств очистки данных, предложенная Джули Борт, подразделяет инструменты очистки данных на две условные категории:

- универсальные системы, предназначенные для обслуживания всей базы данных целиком;
- верификаторы имени/адреса для очистки только данных о клиентах.

Суть этой классификации, изложенная в [94], приведена ниже.

Универсальные системы. К этой категории относится большая часть продуктов, имеющихся на рынке. Это: Enterprise Integrator компании Apertus; Integrity Data Reengineering Tool производства Validy Technology; Data Quality Administrator от Gladstone Computer Services; Inforefiner фирмы Platinium Technology; QDB Analyze (производство QDB Solutions) Trillium Software System компании Hart-Hanks Data Technologies.

Эти системы следует выбирать тогда, когда речь идет о создании банков данных всего предприятия и, соответственно, о сплошной очистке данных. Каждая система использует собственную технологию и имеет собственную сферу приложений. Некоторые из них работают в пакетном режиме, например Trillium, которая просматривает данные в поисках определенных образов и обучается на основе найденной информации. Образы, подлежащие распознаванию (скажем, названия фирм или городские адреса), задаются на этапе предварительного программирования. Другие продукты, как то системы компаний Apertus и Validy, представляют собой средства разработки. В первой применяются правила, написанные на языке Object Query Language. С ней довольно легко работать, но для написания правил требуется настоящее мастерство.

Система компании Validy при отборе записей использует алгоритмы нечеткой логики и делает этот очень эффективно, выуживая такое, что человеку просто в голову не пришло бы проверять. Но эту систему труднее освоить.

Верификаторы имени/адреса. В простых системах, наподобие систем анализа рынка, вполне можно обойтись очисткой имен и адресов. Примеры продуктов этой категории: Nadis компании Group 1 Software и пакет компании Postalsoft. Последний содержит три библиотеки: исправления и кодировки адресов, оформления правильных имен и слияния/очистки. Первая библиотека корректирует адреса, вторая предлагает способ их стандартизации, третья выполняет консолидирующие функции.

Эти продукты проще использовать, и, поскольку область применения их не так широка, работу по очистке они выполнят значительно быстрее. В качестве дополнительной функции это программное обеспечение придает адресам вид, отвечающий требованиям почты. К примеру, Nadis автоматически преобразует имя и адрес в стандарт Universal Name and Address data standard.

Дополнительный продукт компании Group 1, Code-1 Plus, проверяет список адресов на соответствие требованиям. Сертификация гарантирует корректность ZIP-кода и

используется при больших объемах исходящей почты. Те, кто применял эти средства, говорят, что автоматизация работы по обеспечению соответствия адресов различным правилам, установленным почтовым ведомством, стоит затраченных усилий и средств, даже если приходится дополнять названные пакеты другими средствами очистки.

Выше мы рассмотрели одну сторону медали - решение проблемы некачественных или грязных данных путем использования специальных средств очистки и редактирования данных. Однако есть и другая сторона - автоматизированный процесс очистки данных иногда может приводить к ошибкам в данных, которых ранее в них не было.

Рич Олшефски (Rich Olshefski) предлагает классификацию ошибок в данных, которые возникают в результате использования средств очистки [95]. Эти ошибки являются двумя крайностями очистки данных. Качественные, правильно очищенные данные находятся где-то на "золотой середине" между этими между этими крайностями по очистке и редактированию данных.

Ошибка Типа 1 возникает, когда инструмент очистки пытается решить проблему, которой на самом деле не существует. Ошибки Типа 1 имеют место в случае, когда инструмент очистки данных начинает исправлять несоответствия в данных там, где их нет.

Ошибка Типа 2 возникает, когда инструменты очистки полностью упускают существующую проблему.

Ошибка Типа 2 случается при упущении программой неверных данных. Такие данные беспрепятственно проходят проверку, являясь при этом ошибочными. Эту ошибку еще называют "утраченной ошибкой". Программа очистки данных пропускает данные, которые на самом деле должна была исправить. Это может происходить из-за случайной кажущейся правильности ошибочных данных, или же потому, что программа просто упустила их или не была предназначена для очистки таких данных.

Проблема

Самая сложная задача, стоящая перед программой очистки данных, заключается в минимизации ошибок Типа 1 и 2. Для устранения ошибок Типа 1 программа должна стараться не исправлять то, что и так верно. Это сразу же закономерным образом повышает вероятность возникновения ошибки Типа 2. Ошибок Типа 2 можно избежать путем скрупулезной работы с данными, что, конечно же, незамедлительно приводит к излишней очистке и, соответственно, - к допущению ошибки Типа 1.

Некоторые программы очистки стараются так или иначе поддерживать баланс между излишней тщательностью и излишним доверием, создавая объемистые отчеты о "подозрительных" записях. Эти программы собирают все подозрительное в одну большую кучу, которая и является таким отчетом. Такая методика существенно увеличивает затраты на уточнение данных, поскольку требует участия дорогостоящих человеческих ресурсов.

Другим путем чрезмерной компенсации ошибок Типа 1 является внесение слишком малого числа исправлений. А самые примитивные - и поэтому наиболее опасные - программы очистки данных стараются компенсировать и ошибки Типа 2, выдавая на выходе нечто гораздо более скверное, чем то, что было до "очистки".

Определение качественной программы очистки данных, по словам Рича Олшефски, состоит из четырех элементов. Программа должна:

- не затрагивать правильные данные;
- исправлять неверные;
- создавать небольшой по объему отчет о подозрительных записях;
- требовать минимальных затрат на установку, обслуживание и ручные проверки.

Именно такая программа будет "золотой серединой" между ошибками Типа 1 и 2.

Каким же образом можно достичь такого равновесия?

Каждая программа очистки данных имеет некую базу знаний, используемую для поиска и исправления ошибок. Чем она больше и разнообразнее по составу информации, тем лучше результаты очистки.

Рич Олшефски предлагает советы по выбору программного обеспечения, поддерживающего равновесие между двумя возможными перегибами в процессе очистки данных.

- Самым важным является объем базы знаний. Отметьте для себя размер базы знаний и различные виды хранящейся в ней информации. Обратите внимание на количество подтверждений, использующихся при принятии решений по элементам данных. Потребуйте от поставщика продемонстрировать вам примеры работы программы с различными неоднозначными данными.
- Проведите небольшой опыт. Попросите вашего потенциального поставщика описать, как его решение обрабатывает ошибки Типов 1 и 2. Если он не понимает, о чем речь, даже после того как вы объяснили ему терминологию, вы явно обращаетесь не по адресу - либо это не специалист, либо вообще не тот поставщик, что вам нужен.
- Следует тщательно оценить возможности решений. Готовые демо-версии, изображающие решение всех ваших проблем с помощью продукта конкретного поставщика, всегда чрезвычайно подозрительны. Понятно, что демо-версии предназначены для отражения наиболее сильных сторон решений, однако вам следует проявить дотошность. Хорошая демо-версия берет данные "с лету", в идеале - просто ваши собственные.
- Необходимо удостовериться, что вы владеете полной информацией о продукте. Убедитесь, что вы ясно представляете себе стоимость установки, обслуживания и объем организационной работы. Цена самого продукта может оказаться лишь вершиной айсберга. Вы должны точно представлять себе, что именно вас ожидает.

"Очистка данных" может иметь множество значений, но в целом она означает обеспечение поддержки очистки данных, так или иначе связанных с потребителем. Инструменты очистки данных обычно выполняют одну или несколько из следующих функций [96].

Парсинг. Имя и адрес клиента часто хранится в текстовых полях свободного формата. Текст свободного формата иногда труден для разбиения на самостоятельные подстроки, соответствующие типу поля, к которому они относятся (номер улицы, улица адреса, город, штат, индекс и другие данные подобного характера). Программное обеспечение, осуществляющее парсинг, распознает такие подстроки и назначает им соответствующие поля. К тому же, парсинг фирм и стандартизация слов, связанных с описанием фирм, позволяет программе полностью проверить данные о фирмах - (включая сокращения и

аббревиатуры) и стандартизировать фирмы в едином согласованном формате. Большинство поставщиков обеспечивают возможность изменение словарей парсинга в своих инструментах для обработки специфических имен и данных о фирмах, имеющихся у клиента.

Стандартизация. Данные имен и адресов могут вводиться в различных форматах, многие из которых вполне грамматически корректны. Например, "Улица", "Ул." и "Ул" обозначают одно и то же очевидное понятие в составе адреса. У Почтовой службы Соединенных Штатов Америки существуют стандарты для этих и других подобных случаев. Программы стандартизации трансформируют такие поля в согласованный набор обозначений, подходящих для Почтовой службы. Самым важным объектом стандартизации являются записи по клиентам, точность которых может быть существенно повышена за счет использования процесса согласования, описанного далее.

Проверка допустимости. Множество поставщиков предлагают средства распознавания допустимых международных и американских адресов. Некоторые приложения объединяются с программами проверки допустимости и файлами почтовых адресов, проверяющих допустимость международных адресных данных.

Улучшение. Ряд поставщиков предлагают программы, которые добавляют к данным дополнительные факты о записях, изначально в них не содержавшиеся, например, может содержать возможность присвоения клиентам пола на основании анализа его имени и других показателей его профайла. Некоторые поставщики могут устанавливать географическую информацию, обозначающую гео-код, долготу и широту указанной местности. Наиболее же ценным дополнением клиентского профайла являются данные третьих фирм, содержащие демографическую и психографическую информацию.

Согласование и консолидация. Как только имя и адрес очищены, для устранения дублирования клиентов в рамках каждого списка и соединения данных из различных источников применяется программа согласования. Большинство средств содержат алгоритмы расстановки приоритетов между полями (в процессе согласования) и контроля очередности сравнения полей.

Выводы по подготовке данных

В этой лекции мы закончили изучение этапа подготовки данных. Рассмотрели две классификации инструментов очистки и редактирования данных, изучили советы по выбору программного обеспечения, основные функции инструментов очистки данных, классификацию ошибок в данных, которые возникают в результате использования средств очистки данных.

Эти знания являются необходимой составляющей знаний, обеспечивающих возможность проведения процесса Data Mining на данных высокого качества.

Инструменты очистки данных не избавляют пользователя от работы, пользователю достаточно сложно их освоить. Некоторые грязные данные вообще не поддаются автоматической очистке. Перед тем как принимать решение об очистке данных, необходимо рассчитать ее стоимость, т.е. определить, оправдан ли будет этот процесс. Если принято решение, что очистка данных необходима, аналитик получает гарантию того, что процесс Data Mining будет проведен на основе достоверных и качественных данных.

Напомним, что рассмотренные этапы могут занять до 80% всего времени, отведенного на весь проект.

Процесс Data Mining. Построение и использование модели

В предыдущих двух лекциях мы рассмотрели такие этапы процесса Data Mining как анализ предметной области, постановка задачи и подготовка данных. В этой лекции мы уделим внимание оставшимся этапам процесса Data Mining, а именно:

- построению модели;
- проверке и оценке моделей;
- выбору модели;
- применению модели;
- коррекцию и обновлению модели.

Ключевым словом в названии всех этих этапов является понятие "модель". В связи с этим необходимо уделить некоторое время определениям понятий "модель" и "моделирование".

Моделирование

В широком смысле слова моделирование - это научная дисциплина, цель которой - изучение методов построения и использования моделей для познания реального мира.

Моделирование - единственный к настоящему времени систематизированный способ увидеть варианты будущего и определить потенциальные последствия альтернативных решений, что позволяет их объективно сравнивать [97].

Моделирование - достаточно популярный и эффективный метод исследования данных, который является основой анализа данных.

Существует огромное количество ситуаций, когда экспериментировать в реальной жизни не представляется возможным. В этих случаях как раз и применяется моделирование.

Моделирование как процесс представляет собой построение модели и изучение ее свойств, которые подобны наиболее важным, с точки зрения аналитика, свойствам исследуемых объектов.

Таким образом, при помощи моделирования изучаются свойства объектов путем исследования соответствующих свойств построенных моделей.

Моделирование есть метод, процесс и научная дисциплина.

Моделирование широко применяется при использовании методов Data Mining. Путем использования моделей Data Mining осуществляется анализ данных. С помощи моделей Data Mining обнаруживается полезная, ранее неизвестная, доступная интерпретации информация, используемая для принятия решений.

Модель представляет собой упрощенное представление о реальном объекте, процессе или явлении.

Создание и использование Data Mining модели является ключевым моментом для начала понимания, осмысления и прогнозирования тенденций анализируемого объекта.

Построение моделей Data Mining осуществляется с целью исследования или изучения моделируемого объекта, процесса, явления и получения новых знаний, необходимых для принятия решений. Использование моделей Data Mining позволяет определить наилучшее решение в конкретной ситуации.

Аналитик создает модель как подобие изучаемого объекта. Модели могут быть записаны в виде различных изображений, схем, математических формул и т.д. Схематический пример модели был рассмотрен в лекции, посвященной задаче классификации, в первом разделе курса.

Преимуществом использования моделей при исследованиях является простота модели в сравнении с исследуемым объектом. При этом модели позволяют выделить в объекте наиболее существенные факторы с точки зрения цели исследования, и не отвлекаться на маловажные детали.

Из последнего замечания следует, что модель обладает свойством неполноты, поскольку является по своему определению абстрактной.

Приведем простой пример. Пусть имеется база данных клиентов фирмы, содержащая информацию о доходах клиента, семейном положении, предпочтениях и т.д. На основании этой информации нужно определить, является ли определенный клиент потенциальным покупателем нового товара фирмы.

Строя модель, мы предполагаем, что выбор клиента будет определяться характеристиками, которые имеются в базе данных (и мы считаем их существенными для данной задачи). Однако на решение клиента могут оказывать влияние ряд других факторов (например, мода, влияние рекламы, появление на рынке аналогичных товаров других производителей). Эти факторы являются неучтенными. Следовательно, в процессе использования модели ее структура должна совершенствоваться путем уточнения факторов.

Виды моделей

Построенные модели могут иметь различную сложность. Сложность построенной модели зависит от используемых методов, а также от сложности объекта, который анализируется.

Под сложным объектом понимается объект сложной структуры, который характеризуется большим количеством входных переменных, изменчивостью внутренней структуры и внешних факторов, нелинейностью взаимосвязей и др.

Классификация типов моделей в зависимости от характерных свойств, присущих изучаемому объекту или системе, такова

1. динамические (системы, изменяющиеся во времени) и статические;
2. стохастические и детерминированные;
3. непрерывные и дискретные;
4. линейные и нелинейные;
5. статистические; экспертные; модели, основанные на методах Data Mining;

6. прогнозирующие (классификационные) и описательные.

Рассмотрим подробно прогнозирующие и описательные модели. Именно такое подразделение соответствует делению задач Data Mining на два класса: прогнозирующие и описательные.

Прогнозирующие и классификационные (predictive) модели.

Эти модели в явном виде содержат информацию для прогноза, т.е. позволяют прогнозировать числовые значения либо класс (категорию).

Модели, с помощью которых осуществляется прогноз числовых значений атрибутов, будем называть прогнозирующими. Прогнозирование новых значений осуществляется на основе известных (существующих) значений. Прогнозирующие модели Data Mining позволяют выявить особенности функционирования конкретного объекта и на их основе предсказывать будущее поведение объекта. При использовании моделирования (в отличие, например, от предположений, основанных на интуиции) взаимосвязи переменных могут быть оценены количественно, что позволяет выбрать наиболее точную модель и получить более надежный прогноз.

В отличие от классификации, в задачах прогнозирования целевыми являются непрерывные переменные.

Примеры прогнозирующих моделей - это модели линейной регрессии (простейшие модели) и модели на основе нейронных сетей.

Модели, с помощью которых осуществляется прогнозирование класса объекта, будем называть классификационными.

Таким образом, с помощью описанных выше моделей решают задачи классификации и прогнозирования. Такое решение подразумевает двухэтапный процесс: создание модели и ее использование.

Создание моделей Data Mining этого типа означает поиск правил, которые объясняют зависимость выходных параметров от входных.

Примеры классификационных моделей - модели на основе деревьев решений, а также байесовский метод. При помощи классификационной модели решаются следующие задачи:

- принадлежит ли новый клиент к одному из набора существующих классов;
- подходит ли пациенту определенный курс лечения;
- выявление групп ненадежных клиентов;
- определение групп клиентов, которым следует рассылать каталог с новой продукцией.

Класс в этом случае является целевой (выходной) переменной модели.

Дескриптивные или описательные (descriptive) модели описывают общие закономерности предметной области. С помощью дескриптивных моделей решают задачи поиска ассоциативных правил, задачи кластеризации, группировки, обобщения.

Модели кластеризации используются для классификации объектов, при условии, что набор целевых классов неизвестен; они создают так называемые сегментированные модели.

При помощи модели кластеризации, например, решается задача разбиения клиентов фирмы на группы (кластеры) по критерию "близости".

Модели правил ассоциаций используются для нахождения закономерностей между связанными событиями в базах данных.

При помощи модели правил ассоциаций решается задача определения часто встречающихся наборов товаров.

Модели могут быть физическими, концептуальными, математическими, аналоговыми.

Рассмотрим, что же представляет собой математическая модель (ее также называют символической).

Математическая модель

Математическая модель объекта - это его отображение в виде совокупности уравнений, неравенств, логических отношений, графиков и т.д.

При помощи математической модели создается образ исследуемого объекта/системы, выраженный в математических формулах с целью изучения определенных свойств данного объекта. После построения математической модели необходимо наполнить ее данными и провести соответствующие расчеты.

При построении модели следует определить экзогенные и эндогенные переменные .

Экзогенные переменные - переменные, которые задаются вне модели, они известны заранее.

Эндогенные переменные - переменные, которые определяются по ходу расчетов в модели, они не задаются извне.

Далее описывается формализация условий задачи и целевая функция, если она имеется.

Наиболее простое формальное описание модели выражается через функциональную зависимость:

$$Y=f(x_1,...,x_n),$$

где $x_1,...,x_n$ - независимые переменные, Y - зависимая или целевая переменная.

Более сложное описание модели выглядит следующим образом:

$$Y=f(x_1,...,x_n,z_1,...,z_r,w_1,...,w_s),$$

где $x_1,...,x_n$ - независимые переменные, являющиеся внутренними свойствами изучаемого объекта;

$z_1,...,z_r$ - независимые переменные, являющиеся внешними факторами, влияющими на изучаемый объект;

$w_1,...,w_s$ - неучтенные свойства или факторы.

Y - зависимая или целевая переменная.

Необходимо по возможности выяснить все закономерности между целевой переменной и всеми учитываемыми факторами. В результате будет составлена математическая модель, в которой следует отображать те переменные и факторы, которые являются существенными для решения поставленной задачи.

Следует также помнить, что данные, на основе которых строится модель, практически всегда содержат ошибки, поэтому математическая модель является лишь приближенным описанием свойств изучаемого объекта.

В случаях, когда зависимость неизвестна, задача аналитика заключается в том, чтобы определить эту функциональную зависимость. Большинство задач Data Mining относятся как раз к подобной категории задач.

Этап 4. Построение модели

После этого отступления займемся снова этапами процесса Data Mining. После окончания этапа подготовки данных можно переходить к построению модели.

Вернемся к задаче, приведенной в лекции № 6 для более подробного изучения процесса моделирования. Напомним, что в примере рассматривалась задача классификации клиентов туристического агентства на два класса: класс 1 (клиенты, предпочитающие более дорогой, семейный отдых) и класс 2 (клиенты, ориентированные на более дешевый, молодежный отдых).

Задача классификации была выбрана для иллюстрации процесса моделирования, поскольку именно этот тип задач предусматривает обязательное деление процесса моделирования на два отдельных этапа: конструирование (построение) модели и ее использование.

На этапе построения модели при помощи некоего классификационного метода или алгоритма была создана модель (классификатор клиентов). В результате построения модели одно из правил, которые мы получили, гласит: "Если ДОХОД>20 и СЕМЕЙНОЕ ПОЛОЖЕНИЕ = "married", то класс "1".

С одной стороны, можно говорить, что построенная модель выделила наиболее существенные (или значимые) факторы с точки зрения решаемой задачи. Для решения задачи классификации наиболее значимыми оказались переменные "доход" и "семейное положение", остальные факторы (т.е. остальные показатели исследуемой базы данных), сколько бы их ни было, оказались маловажными и не были включены в модель.

С другой стороны, данная модель, как и любая другая, может обладать свойством неполноты. Примером неучтенного фактора могут быть, скажем, природные катаклизмы, которые повлияли на желание клиента пользоваться услугами туристического агентства.

Для построения моделей используются различные методы и алгоритмы Data Mining. Некоторые задачи могут быть решены при помощи моделей, построенных на основе различных методов. Идеальной модели, которая бы позволила решать разнообразные задачи, не существует. Поэтому многие разработчики включают в инструменты Data Mining возможность построения различных моделей, многие также обеспечивают возможность расширяемости моделей. Некоторые инструменты Data Mining создаются специально для конкретных областей применения.

Не так давно рабочей группой Data Mining Group был предложен стандарт PMML (Predictive Model Markup Language), который позволяет осуществлять обмен моделями, созданными в приложениях различных поставщиков программного обеспечения Data Mining. Этот стандарт будет подробно рассмотрен в одной из следующих лекций Курса.

Среди большого разнообразия методов Data Mining должен быть выбран метод или же комбинация методов, при использовании которых построенная модель будет наилучшим образом описывать исследуемый объект.

Иногда для выявления искомых закономерностей требуется использование нескольких методов и алгоритмов. В таком случае одни методы используются в начале моделирования, другие - на дальнейших этапах. Пример: для определения однотипных групп клиентов применялся один из методов кластеризации, в результате клиенты были разбиты на группы, каждой из которых, присвоен код; далее мы пользовались методом деревьев решений. Код группы (результат работы предыдущего метода) использовался для интерпретации полученных закономерностей.

Выбор метода, на основе которого будет построена модель, должен осуществляться с учетом постановки задачи, особенностей набора исходных данных, специфики решаемой задачи, результатов, которые должны быть получены на выходе.

Постановка задачи формализует суть задачи, так, наличие входных и выходных переменных при решении задачи классификации определяет выбор одного из методов "обучение с учителем". Наличие лишь входных переменных определяет выбор другого - метода "обучение без учителя".

Среди особенностей исходного набора данных, например, могут быть следующие его характеристики:

- количество записей в наборе;
- соотношение количества записей в наборе данных и количества входных переменных;
- наличие выбросов, ибо некоторые методы особенно чувствительны к наличию выбросов в данных. Этот факт следует учитывать при построении модели на подобных данных.

Как уже упоминалось выше, Data Mining является итеративным процессом.

Итерация - это циклическая управляющая структура, она содержит выбор между альтернативами и следование избранной.

Выбор между альтернативами в нашем случае - это этап оценки модели.

Если модель приемлема, возможно ее использование.

Этапы подготовки данных, построения модели, оценки модели и выбора лучшей представляют собой цикл.

Если по каким-либо причинам построенная модель оказалось неприемлемой, цикл повторяется и следует один из следующих этапов:

- подготовка данных (если причина некорректности модели - в данных);
- построение модели (если причина некорректности - во внутренних параметрах самой модели).

Для определения специфических свойств исследуемых данных иногда требуется несколько итераций.

Цикл № t-1. Подготовка данных -> построение модели № t-1-> оценка и выбор модели.

Цикл № t. Подготовка данных -> построение модели № t -> оценка и выбор модели.

Цикл № t+1. Подготовка данных -> построение модели № t+1 -> оценка и выбор модели.

Иногда имеет смысл использовать несколько методов параллельно для возможности сравнения и анализа данных с различных точек зрения.

Этап 5. Проверка и оценка моделей

Проверка модели подразумевает проверку ее достоверности или адекватности. Эта проверка заключается в определении степени соответствия модели реальности. Адекватность модели проверяется путем тестирования.

Адекватность модели (adequacy of a model) - соответствие модели моделируемому объекту или процессу.

Понятия достоверности и адекватности являются условными, поскольку мы не можем рассчитывать на полное соответствие модели реальному объекту, иначе это был бы сам объект, а не модель. Поэтому в процессе моделирования следует учитывать адекватность не модели вообще, а именно тех ее свойств, которые являются существенными с точки зрения проводимого исследования. В процессе проверки модели необходимо установить включение в модель всех существенных факторов. Сложность решения этой проблемы зависит от сложности решаемой задачи.

Проверка модели также подразумевает определение той степени, в которой она действительно помогает менеджеру при принятии решений.

Оценка модели подразумевает проверку ее правильности. Оценка построенной модели осуществляется путем ее тестирования.

Тестирование модели заключается в "прогонке" построенной модели, заполненной данными, с целью определения ее характеристик, а также в- проверке ее работоспособности. Тестирование модели включает в себя проведение множества экспериментов. На вход модели могут подаваться выборки различного объема. С точки зрения статистики, точность модели увеличивается с увеличением количества

исследуемых данных. Алгоритмы, являющиеся основой для построения моделей на сверхбольших базах данных, должны обладать свойством масштабирования.

Если модель достаточно сложна, а значит, требуется много времени на ее обучение и последующую оценку, то иногда бывает можно построить и протестировать модель на небольшой части выборки. Однако этот вариант подходит только для однородных данных, в противном случае необходимо использовать все доступные данные [98]. Построенные модели рекомендуется тестировать на различных выборках для определения их обобщающих способностей. В ходе экспериментов можно варьировать объем выборки (количество записей), набор входных и выходных переменных, использовать выборки различной сложности.

Выявленные соотношения и закономерности должны быть проанализированы экспертом в предметной области - он поможет определить, как являются выясненные закономерности (возможно, слишком общими или узкими и специфическими).

Для оценки результатов полученных моделей следует использовать знания специалистов предметной области. Если результаты полученной модели эксперт считает неудовлетворительными, следует вернуться на один из предыдущих шагов процесса Data Mining, а именно: подготовка данных, построение модели, выбор модели.

Если же результаты моделирования эксперт считает приемлемыми, ее можно применять для решения реальных задач.

Этап 6. Выбор модели

Если в результате моделирования нами было построено несколько различных моделей, то на основании их оценки мы можем осуществить выбор лучшей из них. В ходе проверки и оценки различных моделей на основании их характеристик, а также с учетом мнения экспертов, следует выбор наилучшей. Достаточно часто это оказывается непростой задачей.

Основные характеристики модели, которые определяют ее выбор, - это точность модели и эффективность работы алгоритма [77].

В некоторых программных продуктах реализован ряд методов, разработанных для выбора модели. Многие из них основаны на так называемой "конкурентной оценке моделей", которая состоит в применении различных моделей к одному и тому же набору данных и последующем сравнении их характеристик.

Например, в пакете Statistica (Statsoft) [39] эти методы рассматриваются как ядро "предсказывающей добычи данных", они включают: накопление (голосование, усреднение); бустинг; мета-обучение.

Этап 7. Применение модели

После тестирования, оценки и выбора модели следует этап применения модели. На этом этапе выбранная модель используется применительно к новым данным с целью решения задач, поставленных в начале процесса Data Mining. Для классификационных и прогнозирующих моделей на этом этапе прогнозируется целевой (выходной) атрибут (target attribute).

Этап 8. Коррекция и обновление модели

По прошествии определенного установленного промежутка времени с момента начала использования модели Data Mining следует проанализировать полученные результаты, определить, действительно ли она "успешна" или же возникли проблемы и сложности в ее использовании.

Однако даже если модель с успехом используется, ее не следует считать абсолютно верной на все времена. Дело в том, что необходимо периодически оценивать адекватность модели набору данных, а также текущей ситуации (следует учитывать возможность изменения внешних факторов). Даже самая точная модель со временем перестает быть таковой. Для того чтобы построенная модель выполняла свою функцию, следует работать над ее коррекцией (улучшением). При появлении новых данных требуется повторное обучение модели. Этот процесс называют обновлением модели. Работы, проводимые с моделью на этом этапе, также называют контролем и сопровождением модели.

Существует много причин, требующих обучить модель заново, т.е. обновить ее, чтобы отразить определенные изменения.

Основными причинами являются следующие:

- изменились входящие данные или их поведение;
- появились дополнительные данные для обучения;
- изменились требования к форме и количеству выходных данных;
- изменились цели бизнеса, которые повлияли на критерии принятия решений;
- изменилось внешнее окружение или среда (макроэкономика, политическая ситуация, научно-технический прогресс, появление новых конкурентов и товаров и т.д.).

Причины, перечисленные выше, могут обесценить допущения и исходную информацию, на которых основывалась модель при построении.

Приведем простой пример из задачи о туристическом агентстве.

Рассматриваемое правило гласит: "Если ДОХОД>20 и СЕМЕЙНОЕ ПОЛОЖЕНИЕ = "married", то класс "1". Эта модель может успешно работать на протяжении какого-то периода, но затем, например, в силу инфляции в стране, модель должна быть скорректирована. В результате рассматриваемое правило может выглядеть таким образом: "Если ДОХОД>30 и СЕМЕЙНОЕ ПОЛОЖЕНИЕ = "married", то класс "1".

Погрешности в процессе Data Mining

Процесс Data Mining может быть успешным и неуспешным. Использование Data Mining не является гарантией получения исключительно достоверных знаний и принятия на основе этих знаний абсолютно верных решений.

Построенная модель может обладать рядом погрешностей. Вот некоторые из них: недостоверные исходные допущения при построении модели; ограниченные возможности при сборе необходимых данных; неуверенность и страхи пользователя системы, и, в силу этого, слабое их применение; неоправданно высокая стоимость.

Наиболее распространенной погрешностью модели являются **неверные** или **недостоверные исходные допущения**. Некоторые допущения поддаются объективной предварительной проверке, другие не могут быть заранее проверены. Если модель Data Mining основана на допущениях, естественно, ее точность зависит от точности допущений. Если допущения предыдущих периодов при использовании модели не оправдались, т.е. оказались неточны, то следует отказаться от "продления" этих допущений на будущие периоды.

Допустим ситуацию, когда модель хорошо работает в 18 из 20 филиалов компании. В двух филиалах, скорее всего, причина ошибок кроется не в погрешностях или неточностях модели, а в совсем других причинах, например, в данных. Если же модель плохо работает во всех филиалах без исключения, то, скорее всего, построенная модель некорректна.

Довольно сложно и установить время, которое необходимо для определения качества оценки модели. Этот отрезок времени обусловливается спецификой задачи и определяется индивидуально.

Ограниченные возможности при сборе необходимых данных

Как говорилось в одной из предыдущих Лекций, при формировании переменных модели следует абстрагироваться от тех данных, которые есть в наличии. Однако, не всегда есть возможность получить именно те данные, которые необходимы, а также быть уверенными в их качестве. Тем не менее, следует учитывать, что точность построенной модели определяется точностью входных данных.

Если внешние факторы, включенные в модель, изменяются очень часто, эти изменения должны отражаться в системе. Следует учитывать, что это не всегда возможно, а иногда - нецелесообразно.

Неуверенность пользователей

По словам Шеннона, ни одну модель "нельзя считать успешно выстроенной, пока она не принята, не понята и не применена на практике". Однако во многих исследованиях, касающихся использования моделей, отмечается, что в процессе принятия решений далеко не все построенные модели используются в полной мере, а некоторые вовсе не используются. Основными причинами этого является недоверие к моделям либо их непонимание. Для того чтобы избежать подобных явлений, лица, принимающие решения, должны принимать участие в постановке той задачи, для которой строится модель. В дальнейшем следует научить руководителя работать с моделью (т.е. ее программной реализацией), в частности, объяснить ему функции модели, возможности, ограничения и т.д.

Неоправданно высокая стоимость

В результате процесса Data Mining должна быть получена выгода (конечно, если речь не идет о научных исследованиях). Полученная прибыль должна оправдать расходы на процесс Data Mining, а это не только стоимость программного обеспечения для Data Mining, но и затраты на подготовку данных, обучение, консультирование и т.д. Стоимость проекта зависит от его длительности, типа конечного приложения, уровня подготовки пользователей, варианта внедрения (готовый продукт, разработка "под ключ", адаптация под конкретную задачу).

Выводы

Важным этапом в процессе Data Mining является предварительная подготовка данных, в том числе их очистка. От качества подготовленных данных будут зависеть результаты всего процесса.

В процессе построения и выбора модели Data Mining следует пробовать использовать различные методы и алгоритмы, а также их сочетания. При отсутствии опыта использования методов Data Mining лучше начинать с более простых, поддающихся интерпретации моделей. Далее можно постепенно усложнять модели, т.е. использовать более сложные методы. Не следует требовать от модели абсолютной точности, модель можно начинать использовать при получении первых приемлемых результатов.

Следует помнить, что процесс Data Mining является итеративным. При невозможности получения результатов, которые эксперт предметной области считает приемлемыми, необходимо вернуться на один из предыдущих этапов процесса.

Организационные и человеческие факторы в Data Mining. Стандарты Data Mining

Бизнес конкретной фирмы не является изолированным, он - часть рынка. Успешность бизнеса зависит не столько от того, как работает форма, сколько от того, как она работает в сравнении с подобными фирмами рынка. Существует множество различий, нас интересует одно из них - программное обеспечение или инструменты, которые используются для управления бизнесом и принятия решений.

Первый вопрос, который в связи с этим замечанием можно задать менеджеру: "Устраивает ли Вас то программное обеспечение, которое Вы используете для получения новых знаний о делах фирмы?". Если ответ "да", то, возможно, Вы не нуждаетесь в дополнительных инструментах. Но, возможно, у Вас есть вопросы, на которые Вы бы хотели получить ответы, например, почему некоторые Ваши клиенты перешли к конкурирующим фирмам. Ответ на этот и другие вопросы может дать инструмент Data Mining.

В предыдущих лекциях нами был рассмотрен процесс Data Mining с точки зрения этапов, которые должны быть пройдены для получения определенного знания и в итоге - для принятия наиболее верного решения.

Процесс Data Mining можно рассматривать с другой стороны, а именно, с точки зрения организационных и человеческих факторов, которые играют далеко не последнюю роль при внедрении проекта Data Mining.

Организационные Факторы

Когда в организации принято решение использовать Data Mining, первый вопрос, который возникает: "С чего начать?" После того как в организации принято решение использовать технологию Data Mining, необходимо потратить определенное время и усилия, чтобы подготовиться к этому. Необходимо создать определенную организационную окружающую среду.

Поток данных (flow of Data) в организации должен быть приспособлен к Data Mining [17], т.е. сотрудники должны быть заинтересованы в открытом сотрудничестве по обмену информацией. Особенно важно это во взаимодействии между бизнес-отделами и техническими отделами.

Рассмотрим два аспекта, касающихся организационных факторов процесса Data Mining: организационную культуру и деловую окружающую среду.

Чтобы сотрудники могли работать на максимально высоком уровне, организация должна обеспечить свободный поток нужной информации к тому сотруднику, которому она требуется, в четкие сроки и в правильной форме; только тогда возможно будет выработать своевременное оптимальное решение. Лидирующие компании обеспечивают это путем инвестиций в свою информационную инфраструктуру, которая поддерживает бизнес-процессы предприятия [99].

Организационная культура подразумевает активное открытое сотрудничество по обмену информацией между отделами компании и ее сотрудниками.

Это особенно важно во взаимодействии между бизнес-отделами и техническими отделами. Люди должны желать принимать новую информацию и, на основе этого, изменять условия и методы своего труда. Если сотрудники скрывают или защищают свои данные и не желают активно участвовать в обмене информацией и создании новой информации, организация, скорее всего, будет нуждаться во внутреннем или внешнем консультировании для изменения этих фактов. Это всегда непростая задача, но это существенный фактор для достижения успехов при внедрении Data Mining.

Деловая Окружающая среда. Направлять Ваши действия по Data Mining должен бизнес. Руководители высшего звена должны быть заинтересованы во вложении средств в Data Mining, поскольку этот процесс всегда требует значительных затрат. Необходимо четкое понимание проблемы или задачи, которую нужно решить. В организации должна присутствовать готовность открыть доступ к данным и показателям, а также к другим аспектам деятельности.

Интеграция Data Mining в бизнес всегда означает интеграцию соответствующего инструмента в деловую среду организации.

Человеческие факторы. Роли в Data Mining

Человеческий фактор при внедрении Data Mining - это наличие и квалификационное соответствие специалистов, готовых работать с Data Mining.

Специалисты компании, вовлеченные в процесс Data Mining, исполняют одну из ролей, которые показаны на рис. 21.1: специалист предметной области, администратор баз данных, специалист по добыче данных.

Рис. 21.1. Роли в Data Mining

Роли между специалистами распределены следующим образом.

Специалист предметной области (Domain experts) - специалист, имеющий знания о окружении бизнеса, процессах, заказчиках, клиентах, потребителях, конкурентах, т.е. о предметной области.

Знания о предметной области включают факты, которые к данной области относятся, закономерности, характерные для нее, гипотезы о возможных связях между явлениями, процессами и фактами в ней, процедуры для решения типовых задач. Экспертные знания - это те знания, которыми располагает специалист в некоторой предметной области.

Администратор баз данных (Database administrator) - специалист, имеющий знания о том, где и каким образом хранятся данные, как получить к ним доступ и как связать между собой эти данные.

Администратор базы данных отвечает за выработку требований к базе данных, за ее проектирование, реализацию, эффективное использование и сопровождение.

Другими обязанностями администратора баз данных могут быть: определение статуса информации и статуса пользователей; модификация данных; обеспечение целостности данных; загрузка данных и ведение БД; защита данных; обеспечение восстановления баз данных; сбор и статистическая обработка обращений к БД; анализ эффективности функционирования базы данных.

Специалист по добыче данных (Mining specialists) - специалист по анализу данных, который имеет, как минимум, основы статистических знаний.

Этот специалист должен быть способен применять технологии Data Mining и интерпретировать полученные результаты. Он должен уметь устанавливать связи со специалистом по предметной области для управления полученными результатами и с администратором БД для получения доступа к данным в запрос на свои действия.

Специалист по добыче данных ответственен за получение необходимых для Data Mining сведений из различных источников, а также за получение информации от специалистов в данной предметной области. Специалист по добыче данных должен быть также своего рода постановщиком задач. Он должен уметь получать необходимую информацию и входные данные для Data Mining-системы у специалистов по предметной области, задавать вопросы с целью уточнения сведений и т.д.

Первые две роли из описанных выше в том или ином виде присутствуют в любой компании. Третья роль в первое время внедрения Data Mining может исполняться консультантом другой компании. После приобретения соответствующих знаний, это место может занять человек из Вашей компании, например - маркетинговый аналитик.

Одной из основных трудностей при выборе специалистов либо внутри Вашей организации, либо сторонних консультантов является разнообразие областей, которые должны быть объединены в одном процессе. Процесс Data Mining требует наличия связей между бизнесом, анализом и информационными технологиями, чтобы обеспечить непрерывный двунаправленный поток информации (данные - информация - решения), который был рассмотрен в одной из начальных лекций курса.

Три роли, рассмотренные выше, являются основными, и без них процесс Data Mining не может быть осуществлен. Часто в процесс также вовлечены другие специалисты по информационным технологиям и менеджеры проектов.

Среди них могут быть:

- менеджер проектов (Project Manager);
- специалист по IT Архитектуре (IT Architect);
- специалист по Архитектуре Решений (Solution Architect);
- специалист по Архитектуре Данных (Data Architect);
- специалист по Моделированию данных (Data Modeler);
- эксперт Data Mining (Data Mining Expert);
- деловой Аналитик (Business Analyst).

Каждая из этих ролей может быть отведена специалисту внутри организации либо стороннему специалисту. Процесс найма третьих лиц, т.е. сторонних специалистов для выполнения определенных работ, называют аутсорсингом (outsourcing). Воспользовавшись услугами приглашенных специалистов, компании могут добиться существенного уменьшения затрат на оплату труда. О других преимуществах аутсорсинга для Data Mining будет рассказано в следующем разделе курса.

Роли Data Mining, в зависимости от конечной цели работ, распределяются следующим образом:

- исследователи (написание исследовательских докладов и статей);
- практикующие аналитики (решение реальных и практических задач анализа данных);
- разработчики программного обеспечения (написание Data Mining- программного обеспечения);
- студенты (в настоящее время обучающиеся в учебных заведениях);
- бизнес-аналитики (главным образом, оценивающие результаты использования data mining);
- менеджеры (управляют одним или большим количеством проектов);
- другие.

Согласно последним опросам на KDnuggets, наибольшее число из голосующих - это практикующие аналитики, использующие технологию Data Mining для анализа реальных данных (34%), и исследователи (19%), далее идут студенты, бизнес-аналитики, разработчики программного обеспечения и менеджеры.

Теперь мы рассмотрим процесс Data Mining в разрезе работ, выполняемых описанными выше специалистами, коснемся распределения их обязанностей, укажем, где эти работы пересекаются в процессе достижения бизнес-цели.

Напомним, что процесс Data Mining практически никогда не является линейным, в большинстве случаев это итеративный циклический процесс. Именно итеративность гарантируют процессу Data Mining такой результат, который будет адаптирован под решение конкретной задачи.

Процесс Data Mining, с точки зрения человеческого фактора, является постоянным взаимодействием трех основных специалистов.

Взаимодействие специалиста по добыче данных и специалиста по предметной области осуществляется в двух точках соприкосновения (не забываем при этом, что Data Mining - итеративный процесс).

Первая точка - анализ предметной области, где определяются задачи и требования к будущей системе. Специалист по добыче данных должен вникнуть в предметную область,

изучить ее базовые термины, другими словами, он должен провести **анализ предметной области**. На основании знаний методов и инструментов Data Mining специалист по добыче данных предлагает вариант решения проблемы.

Второй точкой соприкосновения указанных выше специалистов является интерпретация результатов, полученных в результате Data Mining.

Взаимодействие специалиста по добыче данных и администратора баз данных осуществляется на этапах анализа требований к данным и сбора данных. Непосредственно подготовка данных для Data Mining может осуществляться специалистом по добыче данных самостоятельно либо во взаимодействии с администратором баз данных.

Взаимодействие трех специалистов осуществляется на завершающих этапах Data Mining при проверке работоспособности системы, например, при сравнении прогнозных результатов с реальными. При необходимости процесс Data Mining возвращается на один из предыдущих этапов.

От того, насколько консолидированы будут действия специалистов из разных областей, зависит длительность проекта и качество полученных результатов.

Если в проекте Data Mining присутствует роль руководителя, на него возлагается координация и контроль работ, проводимых описанными выше специалистами.

CRISP-DM методология

Мы рассмотрели процесс Data Mining с двух сторон: как последовательность этапов и как последовательность работ, выполняемых исполнителями ролей Data Mining.

Существует еще одна сторона - это стандарты, описывающие методологию Data Mining. Последние рассматривают организацию процесса Data Mining и разработку Data Mining-систем.

CRISP-DM [100] (The Cross Industrie Standard Process for Data Mining - Стандартный межотраслевой процесс Data Mining) является наиболее популярной и распространенной методологией. Членами консорциума CRISP-DM являются NCR, SPSS и DaimlerChrysler.

В соответствии со стандартом CRISP, **Data Mining является непрерывным процессом со многими циклами и обратными связями**.

Data Mining по стандарту CRISP-DM включает следующие фазы:

1. Осмысление бизнеса (Business understanding).
2. Осмысление данных (Data understanding).
3. Подготовка данных (Data preparation).
4. Моделирование (Modeling).
5. Оценка результатов (Evaluation).
6. Внедрение (Deployment).

К этому набору фаз иногда добавляют седьмой шаг - Контроль, он заканчивает круг. Фазы Data Mining по стандарту CRISP-DM изображены на рис. 21.2.

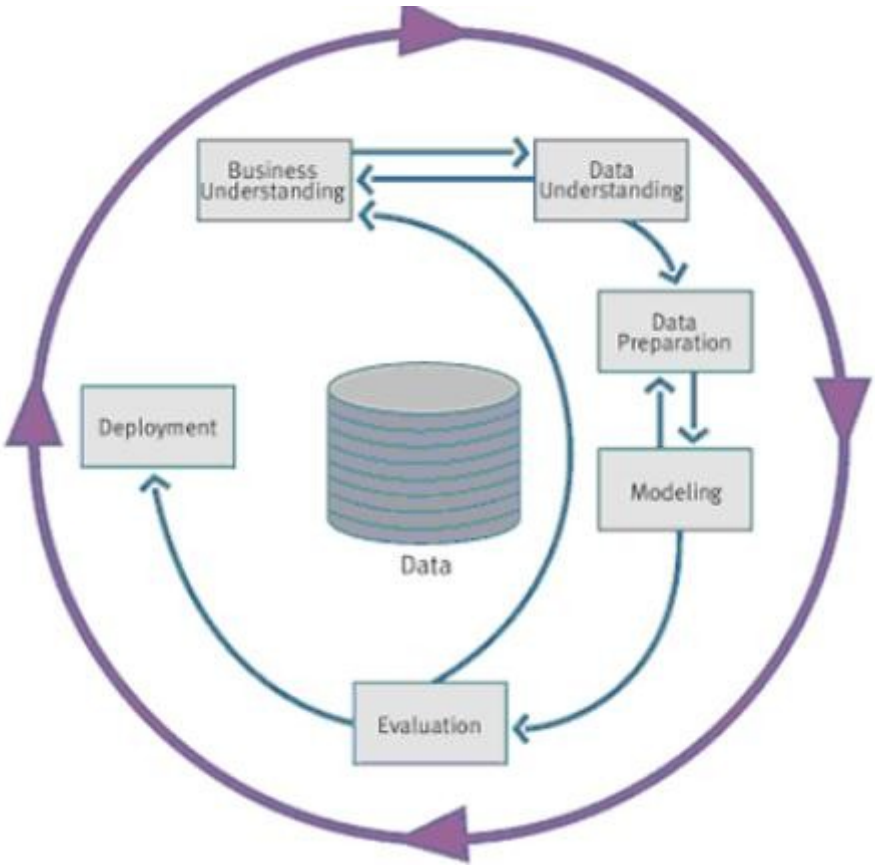

Рис. 21.2. Фазы, рекомендуемые моделью CRISP-DM

При помощи методологии CRISP-DM Data Mining превращается в бизнес-процесс, в ходе которого технология Data Mining фокусируется на решении конкретных проблем бизнеса. Методология CRISP-DM, которая разработана экспертами в индустрии Data Mining, представляет собой пошаговое руководство, где определены задачи и цели для каждого этапа процесса Data Mining.

Методология CRISP-DM описывается в терминах иерархического моделирования процесса [101], который состоит из набора задач, описанных четырьмя уровнями обобщения (от общих к специфическим): фазы, общие задачи, специализированные задачи и запросы.

На верхнем уровне процесс Data Mining организовывается в определенное количество **фаз**, на втором уровне каждая фаза разделяется на несколько **общих задач**. Задачи второго уровня называются общими, потому что они являются обозначением (планированием) достаточно широких задач, которые охватывают все возможные Data Mining-ситуации. Третий уровень является **уровнем специализации задачи**, т.е. тем местом, где действия общих задач переносятся на конкретные специфические ситуации. Четвертый уровень является **отчетом** по действиям, решениям и результатам фактического использования Data Mining.

CRISP-DM - это не единственный стандарт, описывающий методологию Data Mining. Помимо него, можно применять такие известные методологии, являющиеся мировыми стандартами, как Two Crows, SEMMA, а также методологии организации или свои собственные.

SEMMA методология

SEMMA методология реализована в среде SAS Data Mining Solution (SAS) [102]. Ее аббревиатура образована от слов Sample ("Отбор данных", т.е. создание выборки), Explore ("Исследование отношений в данных"), Modify ("Модификация данных"), Model ("Моделирование взаимозависимостей"), Assess ("Оценка полученных моделей и результатов"). Методология разработки проекта Data Mining в соответствии с методологией SEMMA изображена на рис. 21.3.

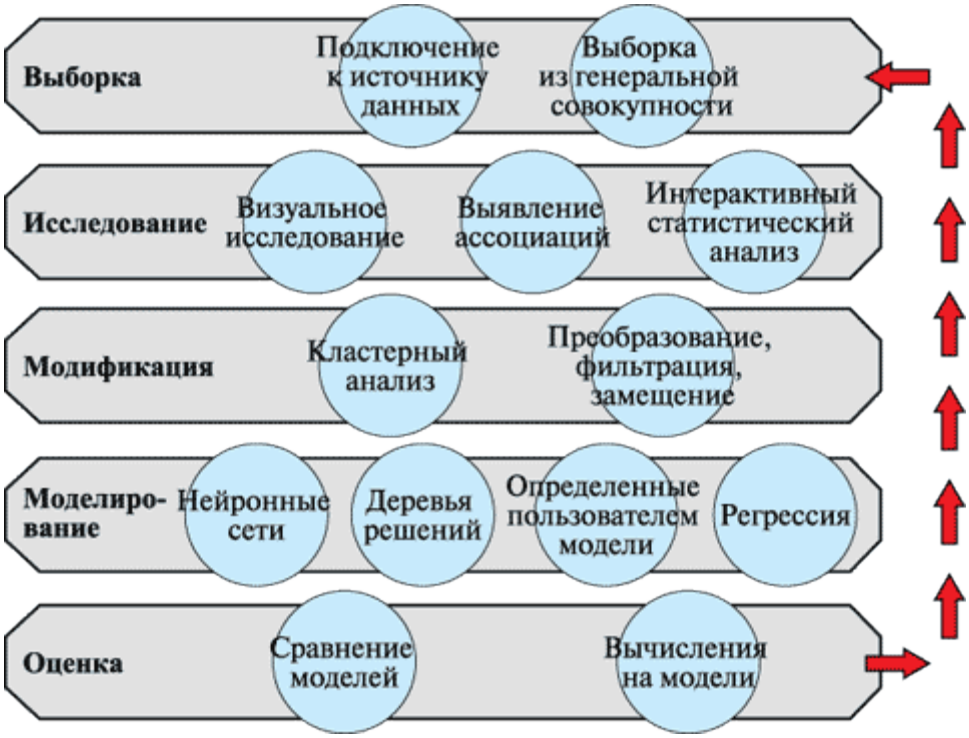

Рис. 21.3. Методология разработки проекта Data Mining в соответствии с методологией SEMMA

Подход SEMMA подразумевает, что все процессы выполняются в рамках гибкой оболочки, поддерживающей выполнение всех необходимых работ по обработке и анализу данных. Подход SEMMA сочетает структурированность процесса и логическую организацию инструментальных средств, поддерживающих выполнение каждого из шагов. Благодаря диаграммам процессов обработки данных, подход SEMMA упрощает применение методов статистического исследования и визуализации, позволяет выбирать и преобразовывать наиболее значимые переменные, создавать модели с этими переменными, чтобы предсказать результаты, подтвердить точность модели и подготовить модель к развертыванию.

Эта методология не навязывает каких-либо жестких правил. В результате использования методологии SEMMA разработчик может располагать научными методами построения концепции проекта, его реализации, а также оценки результатов проектирования.

По результатам последних опросов KDnuggets (2004 г.), 42% опрошенных лиц использует методологию CRISP-DM, 10% - методологию SEMMA, 6% - собственную методологию организации, 28% - свою собственную методологию, другими методологиями пользуется 6% опрошенных. Не пользуются никакой методологией 7% опрошенных.

Другие стандарты Data Mining

Как уже отмечалось, описанные стандарты являются методологиями Data Mining, т.е. рассматривают организацию процесса и разработку систем Data Mining. Помимо этой группы, в последние годы появился ряд стандартов, цель которых - согласовать достижения в Data Mining, упростить управление моделированием процессов и дальнейшее использование созданных моделей. Эти стандарты условно можно поделить на две категории:

1. Стандарты, относящиеся к выработке единого соглашения по хранению и передаче моделей Data Mining.
2. Стандарты, относящиеся к унификации интерфейсов.

Стандарт PMML

В предыдущих лекциях мы уже упоминали о стандарте PMML (Predictive Modeling mark-up Language) - языке описания предикторных (или прогнозных) моделей или языке разметки для прогнозного моделирования.

PMML относится к группе стандартов **по хранению и передаче моделей Data Mining**.

Разработка и внедрение этого стандарта ведется IT-консорциумом DMG (Data Mining Group). DMG [103] - группа, в которую входят все лидирующие компании, разрабатывающие программное обеспечение в области анализа данных.

Основа этого стандарта - язык XML. Примером другого стандарта, также основанного на языке XML, является стандарт обмена статистическими данными и метаданными. Стандарт PMML используется для описания моделей Data Mining и статистических моделей.

Основная цель стандарта PMML - обеспечение возможности обмена моделями данных между программным обеспечением разных разработчиков.

При помощи стандарта PMML-совместимые приложения могут легко обмениваться моделями данных с другими PMML-инструментами. Таким образом, модель, созданная в одном программном продукте, может использоваться для прогнозного моделирования в другом.

По словам сторонников PMML, этот стандарт "делает Data Mining более демократичным", позволяет все большому количеству пользователей пользоваться продуктами Data Mining. Это достигается за счет возможности использования ранее созданных моделей данных. PMML позволяет использовать модели данных сколь угодно часто и существенно помогает в практической работе с ними.

Стандарт PMML включает:

- описание анализируемых данных (структура и типы данных);
- описание схемы анализа (используемые поля данных);
- описание трансформаций данных (например, преобразования типов данных);
- описание статистик, прогнозируемых полей и самих прогнозных моделей.

Стандарт PMML обеспечивает поддержку наиболее распространенных прогнозных моделей, созданных при помощи алгоритмов и методов анализа данных, в частности - нейронных сетей, деревьев решений, алгоритмов ассоциативных правил, кластерного анализа, логических правил и др.

Стандарты, относящиеся к унификации интерфейсов

С помощью стандартов этой группы любое приложение может получить доступ к функциональности Data Mining. Здесь можно выделить стандарты, направленные на стандартизацию интерфейсов для объектных языков программирования, и стандарты, направленные на разработку надстройки над языком SQL.

К стандартам, направленным на стандартизацию интерфейсов для объектных языков программирования, можно отнести: **CWM Data Mining**, **JDM**.

В 2000 году организации MDC (MetaData Coalition, www.mdcinfo.com) и OMG (Object Management Group, www.omg.org), разрабатывающие два конкурирующих стандарта - в области интеллектуальных технологий для бизнеса - **OIM** (Open Information Model) и **CWM** (Common Warehouse Metamodel) - общую метамодель хранилищ данных решили объединить свои достижения и усилия под управлением OMG. Стандарт CWM включает описание базовых элементов объектной модели, реляционных отношений, языка XML, структуры семантики предметной области, архитектуры OLAP, добычи данных, технологии перегрузки данных и некоторых расширений.

JDM (The Java Data Mining standard - Java Specification Request 73, JSR-73). Стандарт, разработанный группой JSR 73, Java Data Mining API (JDM) - это первая попытка создать стандартный Java API (программный интерфейс приложения) для получения доступа к инструментам Data Mining из Java-приложений.

Вторая группа стандартов направлена на разработку надстройки над языком SQL, которая позволяла бы обращаться к инструментарию Data Mining, встроенному непосредственно в реляционную базу данных. К этой группе можно отнести следующие стандарты: SQL/MM, OLE DB for Data Mining.

Стандарт SQL/MM представляет собой набор определенных пользователем SQL процедур для возможностей вычислений и использований моделей Data Mining.

The OLE DB for Data Mining standard of Microsoft. Этот стандарт позволяет, подобно SQL/MM, применять методы Data Mining в структуре реляционных баз данных. Этот стандарт является расширением OLE DB.

Стандарты, имеющие прямое или опосредованное отношение к Data Mining, можно объединить в группы:

- стандарты, базирующиеся на услугах Data Mining (услуги создания модели управления, скоринговые услуги, услуги анализа данных, услуги исследования данных, статистические услуги моделирования);
- стандарты web-службы (SOAP/XML, WSRF, и т.д), Grid-Услуги (OGSA, OGSA/DAI, и т.д.), Семантические Стандарты Web (RDF, OWL, и т.д.);

- стандарты, которые должны появиться в ближайшее время: стандарты для технологического процесса, стандарты для преобразований данных, стандарты для оперативного (real time) Data Mining, стандарты для сетей данных (data webs).

Как мы видим, стандарты Data Mining развиваются, появляются также новые, имеющие как прямое, так и опосредованное отношение к этой технологии. Это свидетельствует о достаточной "зрелости" Data Mining и вступлении ее в новый этап развития.

Рынок инструментов Data Mining

На рынке программного обеспечения Data Mining существует огромное разнообразие продуктов, относящихся к этой категории. И не растеряться в нем достаточно сложно. Для выбора продукта следует тщательно изучить задачи, поставленные перед Вами, и обозначить те результаты, которые необходимо получить.

Приведем цитату из Руководства по приобретению продуктов Data Mining (Enterprise Data Mining Buying Guide) компании Aberdeen Group: "Data Mining - технология добычи полезной информации из баз данных. Однако в связи с существенными различиями между инструментами, опытом и финансовым состоянием поставщиков продуктов, предприятиям необходимо тщательно оценивать предполагаемых разработчиков Data Mining и партнеров".

Существуют различные варианты решений по внедрению инструментов Data Mining, например:

- покупка готового программного обеспечения Data Mining;
- покупка программного обеспечения Data Mining, адаптированного под конкретный бизнес;
- разработка Data Mining-продукта на заказ сторонней компанией;
- разработка Data Mining-продукта своими силами;
- различные комбинации вариантов, описанных выше, в том числе использование различных библиотек, компонентов и инструментальные наборы для разработчиков создания встроенных приложений Data Mining.

В этой лекции мы рассмотрим, что предлагает рынок готового программного обеспечения, в частности, оценим рынок в разрезе задач Data Mining.

Поставщики Data Mining

В начале 90-х годов прошлого столетия рынок Data Mining насчитывал около десяти поставщиков. В средине 90-х число поставщиков, представленных компаниями малого, среднего и большого размера, насчитывало более 50 фирм.

Сейчас к аналитическим технологиям, в том числе к Data Mining, проявляется огромный интерес. На этом рынке работает множество фирм, ориентированных на создание инструментов Data Mining, а также комплексного внедрения Data Mining, OLAP и хранилищ данных. Инструменты Data Mining во многих случаях рассматриваются как составная часть BI-платформ, в состав которых также входят средства построения хранилищ и витрин данных, средства обработки неожиданных запросов (ad-hoc query), средства отчетности (reporting), а также инструменты OLAP.

Разработкой в секторе Data Mining всемирного рынка программного обеспечения заняты как всемирно известные лидеры, так и новые развивающиеся компании. Инструменты Data Mining могут быть представлены либо как самостоятельное приложение, либо как дополнения к основному продукту.

Последний вариант реализуется многими лидерами рынка программного обеспечения. Так, уже стало традицией, что разработчики универсальных статистических пакетов, в дополнение к традиционным методам статистического анализа, включают в пакет определенный набор методов Data Mining. Это такие пакеты как SPSS (SPSS, Clementine), Statistica (StatSoft), SAS Institute (SAS Enterprise Miner). Некоторые разработчики OLAP-решений также предлагают набор методов Data Mining, например, семейство продуктов Cognos. Есть поставщики, включающие Data Mining решения в функциональность СУБД: это Microsoft (Microsoft SQL Server), Oracle, IBM (IBM Intelligent Miner for Data).

Рынок поставщиков Data Mining активно развивается. Постоянно появляются новые фирмы-разработчики и новые инструменты.

Интересными являются данные опроса "Инструменты Data Mining, которые Вы регулярно используете", проведенного в мае 2005 года на Kdnuggets. Его результаты представлены на рис. 22.1.

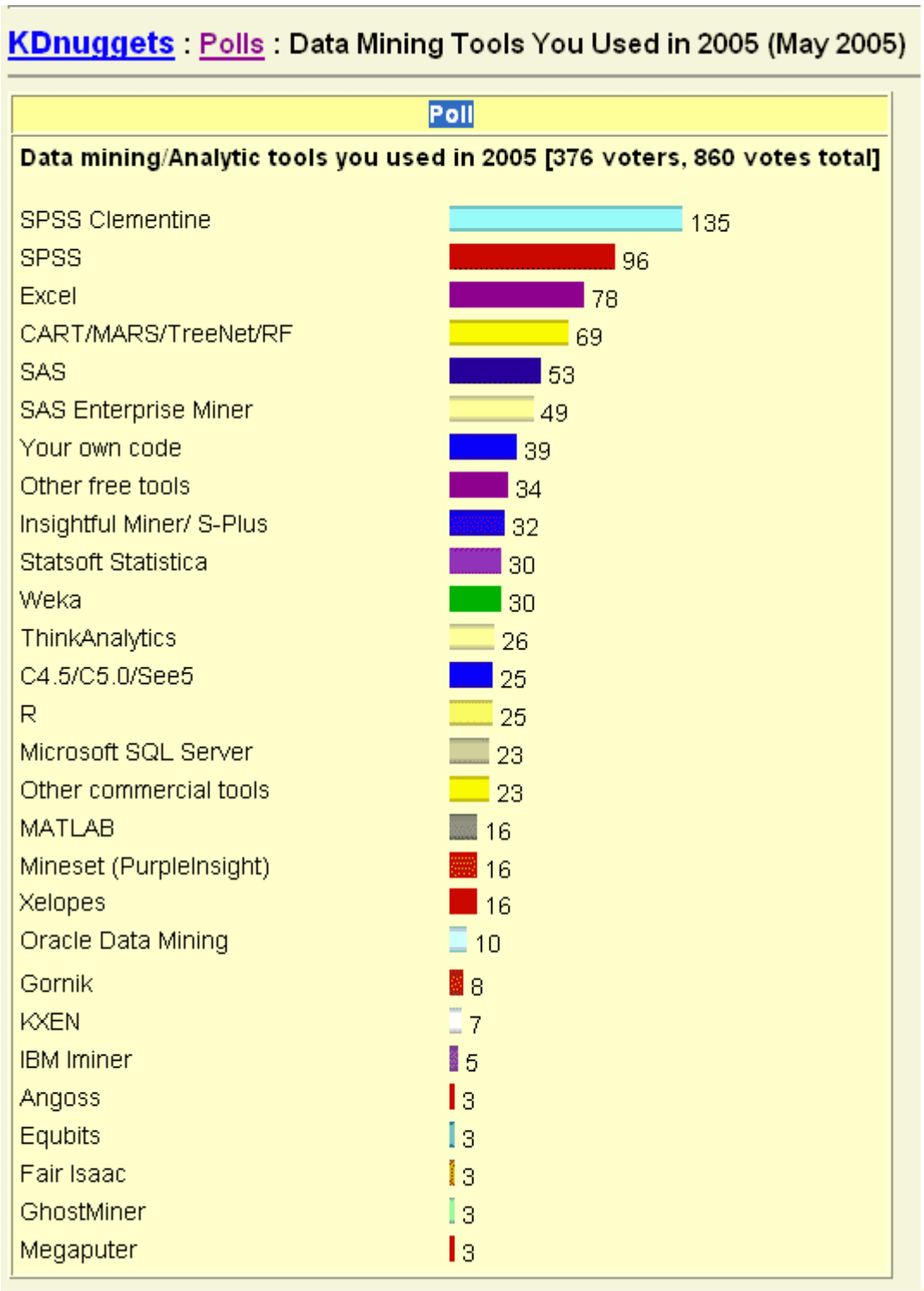

Рис. 22.1. Инструменты Data Mining, используемые голосовавшими в 2005 году

Сравнивая данные этого опроса с подобными опросами 2002 и 2003 годов, можно сказать, что популярность некоторых продуктов возрастает, а некоторых - падает. Это касается как коммерческих, так и свободно распространяемых инструментов. Например, что касается бесплатного инструментария: в 2003 году, по сравнению с 2002 годом, часть голосов от инструмента Weka ушли к инструментам Prudsys Xelopes и R, в 2005 же году количество голосов за инструмент Weka увеличилось, а за Xelopes проголосовало существенно

меньше пользователей. Подобный пример можно привести и из коммерческого программного обеспечения: популярность Microsoft Сервер SQL для Data Mining в 2003 году, по сравнению с 2002 годом, возросла, а в 2005 году - снизилась.

Таким же образом изменялись позиции большинства инструментов, но результаты всех трех опросов представлены практически одним и тем же списком поставщиков.

Как видно из опроса, число респондентов вдвое меньше числа голосов, и каждый голосовавший мог выбрать несколько инструментов. Числа, представленные в опросе, означают фактическое число голосов. Процент по каждому инструменту не определяется, поскольку он будет отличаться в зависимости от того, вычислен ли он относительно числа респондентов или от числа голосов.

В комментариях к этому опросу по поводу участия в нем продавцов, редактор сайта отмечает, что при голосовании были использованы механизмы против двойного голосования, но его нельзя считать научным, поскольку за некоторые продукты представители компаний разработчиков голосовали намного более активно, чем за другие (некоторые очевидные двойные голоса продавцов были удалены). Однако эти опросы, по оценкам редактора, действительно дают ощущение разнообразия существующих инструментов Data Mining.

Относительно цен на инструменты, редактор отмечает, что они имеют тенденцию изменяться, а также отличаются по стоимости для бизнес-пользователей и научных работников, так как последние иногда могут получить бесплатную лицензию для исследований.

Представленные выше продукты, согласно предполагаемой цене для бизнес-пользователей на май 2005 года, сгруппированы следующим образом:

- Уровень предприятия: (US $10000 и больше)

  Fair Isaac, IBM, Insightful, KXEN, Oracle, SAS, SPSS.

- Уровень отдела: (от $1000 до $9999)

  Angoss, CART/MARS/TreeNet/Random Forests, Equbits, GhostMiner, Gornik, Mineset, MATLAB, Megaputer, Microsoft SQL Server, Statsoft Statistica, ThinkAnalytics.

- Личный уровень: (от $1 до $999): Excel, See5.
- Свободно распространяемое программное обеспечение: C4.5, R, Weka, Xelopes.

Инструменты Data Mining можно оценивать по различным критериям. Оценка программных средств Data Mining с точки зрения конечного пользователя определяется путем оценки набора его характеристик. Их можно поделить на две группы: бизнес-характеристики и технические характеристики. Это деление является достаточно условным, и некоторые характеристики могут попадать одновременно в обе категории.

Характеристика № 1. Интуитивный интерфейс.

Интерфейс - среда передачи информации между программной средой и пользователем, диалоговая система, которая позволяет передать человеку все необходимые данные, полученные на этапе формализации и вычисления.

Интерфейс подразумевает расположение различных элементов, в т.ч. блоков меню, информационных полей, графических блоков, блоков форм, на экранных формах.

Для удобства работы пользователя необходимо, чтобы интерфейс был интуитивным.

Интуитивный интерфейс позволяет пользователю легко и быстро воспринимать элементы интерфейса, благодаря чему диалог "программная среда-пользователь" становится проще и доступней.

Понятие интуитивного интерфейса включает также понятие знакомой окружающей среды и наличие внятной нетехнической терминологии (например, для сообщения пользователю о совершенной ошибке).

Характеристика № 2. Удобство экспорта/импорта данных.

При работе с инструментом Data Mining-пользователь часто применяет разнообразные наборы данных, работает с различными источниками данных. Это могут быть текстовые файлы, файлы электронных таблиц, файлы баз данных. Инструмент Data Mining должен иметь удобный способ загрузки (импорта) данных. По окончании работы пользователь также должен иметь удобный способ выгрузки (экспорта) данных в удобную для него среду. Программа должна поддерживать наиболее распространенные форматы данных: txt, dbf, xls, csv и другие.

Дополнительное удобство для пользователя создается при возможности загрузки и выгрузки определенной части (по выбору пользователя) импортируемых или экспортируемых полей.

Характеристика № 3. Наглядность и разнообразие получаемой отчетности

Эта характеристика подразумевает получение отчетности в терминах предметной области, а также в качественно спроектированных выходных формах в том количестве, которое может предоставить пользователю всю необходимую результативную информацию.

Характеристика № 4. Легкость обучения работы с инструментарием

Характеристика № 5. Прозрачные и понятные шаги Data Mining-процесса

Характеристика № 6. Руководство пользователя.Существенно упрощает работу пользователя наличие руководства пользователя, с пошаговым описанием шагов генерации моделей Data Mining.

Характеристика № 7. Удобство и простота использования. Существенно облегчает работу начинающего пользователя возможность использовать Мастер или Визард (Wizard).

Характеристика № 8. Для пользователей, не владеющих английским языком, важной характеристикой является **наличие русифицированной версии инструмента**, а также документации на русском языке.

Характеристика № 9. Наличие демонстрационной версии с решением конкретного примера.

Характеристика № 10. Возможности визуализации. Наличие графического представления информации существенно облегчает интерпретируемость полученных результатов.

Характеристика № 11. Наличие значений **параметров, заданных по умолчанию**. Для начинающих пользователей - это достаточно существенная характеристика, так как при выполнении многих алгоритмов от пользователя требуется задание или выбор большого числа параметров. Особенно много их в инструментах, реализующих метод нейронных сетей. В нейросимуляторах чаще всего заранее заданы значения основных параметров, иной раз неопытным пользователям даже не рекомендуется изменять эти значения. Если же такие значения отсутствуют, пользователю приходится перепробовать множество вариантов, прежде чем получить приемлемый результат.

Характеристика № 12. Количество реализуемых методов и алгоритмов. Во многих инструментах Data Mining реализовано сразу несколько методов, позволяющих решать одну или несколько задач. Если для решения одной задачи (классификации) предусмотрена возможность использования нескольких методов (деревьев решений и нейронных сетей), пользователь получает возможность сравнивать характеристики моделей, построенных при помощи этих методов.

Характеристика № 13. Скорость вычислений и скорость представления результатов.

Характеристика № 14. Наличие квалифицированного ассистента (консультации по выбору методов и алгоритмов), консультационная поддержка.

Характеристика № 15. Возможности поиска, сортировки, фильтрации.

Такая возможность полезна как для входных данных, так и для выходной информации. Применяется сортировка по различным критериям (полям), с возможностью накладывания условий.

При условии фильтрации входных данных появляется возможность построения модели Data Mining на одной из выборок набора данных. Необходимость и польза от проведения такого анализа была описана в одной из лекций, посвященных процессу Data Mining. Фильтрация выходной информации полезна с точки зрения интерпретации результатов. Так, например, иногда при построении деревьев решений результаты получаются слишком громоздкими, и здесь могут оказаться полезными функция как фильтрации, так и поиска и сортировки. Дополнительное удобство для пользователя - цветовая подсветка некоторых категорий записей.

Характеристика № 16. Защита, пароль. Очень часто при помощи Data Mining анализируется конфиденциальная информация, поэтому наличие пароля доступа в систему является желательной характеристикой для инструмента.

Характеристика № 17. Платформы, на которых поддерживается работа инструмента, в частности: PC Standalone (95/98/2000/NT), Unix Server, Unix Standalone, PC Client, NT Server.

Описанные характеристики являются критериями функциональности, удобства, безопасности инструмента Data Mining. При выборе инструмента следует руководствоваться потребностями, а также задачами, которые необходимо решить.

Так, например, если точно известно, что фирме необходимо решать исключительно задачи классификации, то возможность решения инструментом других задач совсем не является критичной. Однако, следует учитывать, что внедрение Data Mining при серьезном подходе требует серьезных финансовых вложений, поэтому необходимо учитывать все возможные задачи, которые могут возникнуть в перспективе.

Классификация инструментов Data Mining

Рынок инструментов Data Mining определяется широтой этой технологии и вследствие этого - огромным многообразием программного обеспечения. Приведем классификацию инструментов Data Mining согласно KDnuggets: инструменты общего и специфического назначения; бесплатные и коммерческие инструменты.

Наиболее популярная группа инструментов содержит следующие категории:

- наборы инструментов;
- классификация данных;
- кластеризация и сегментация;
- инструменты статистического анализа;
- анализ текстов (Text Mining), извлечение отклонений (Information Retrieval (IR));
- инструменты визуализации.

Наборы инструментов. К этой категории относятся универсальные инструменты, которые включают методы классификации, кластеризации и предварительной подготовки данных. К этой группе относятся такие известные коммерческие инструменты как:

- Clementine (http://www.spss.com/clementine). Data Mining с использованием Clementine является бизнес-процессом, разработанным для минимизации времени решения задач. Clementine поддерживает процесс Data Mining: доступ к данным, преобразования, моделирование, оценивание и внедрение. При помощи Clementine Data Mining выполняется с методологией CRISP-DM.
- DBMiner 2.0 Enterprise (http://www.dbminer.com), мощный инструмент для исследования больших баз данных; использует Microsoft Сервер SQL 7.0 Plato.
- IBM Intelligent Miner for Data (http://www.ibm.com/software/data/iminer/fordata/). Инструмент предлагает последние Data Mining-методы, поддерживает полный Data Mining процесс: от подготовки данных до презентации результатов. Поддержка языков XML и PMML.
- KXEN (Knowledge eXtraction ENgines). Инструмент, работающий на основе теории Вапника (Vapnik) SVM. Решает задачи подготовки данных, сегментации, временных рядов и SVM-классификации.
- Oracle Data Mining (ODM) (http://otn.oracle.com/products/bi/9idmining.html). Инструмент обеспечивает GUI, PL/SQL-интерфейсы, Java-интерфейс. Используемые методы:

байесовская классификация, алгоритмы поиска ассоциативных правил, кластерные методы, SVM и другие.
- Polyanalyst (http://www.megaputer.com/). Набор, обеспечивающий всесторонний Data Mining. Сейчас, помимо методов прежних версий, также включает анализ текстов, лес решений, анализ связей. Поддерживает OLE DB for Data Mining и DCOM-технологию.
- SAS Enterprise Miner (http://www.sas.com/). Интегрированный набор, который обеспечивает дружественный GUI. Поддерживается методология SEMMA.
- SPSS (http://www.spss.com/clementine/). Один из наиболее популярных инструментов, поддерживается множество методов Data Mining.
- Statistica Data Miner (http://www.StatSoft.com/). Инструмент обеспечивает всесторонний, интегрированный статистический анализ данных, имеет мощные графические возможности, управление базами данных, а также приложение разработки систем.

Примером российской разработки инструментального набора, кроме Polyanalyst, является пакет Deductor, при помощи которого в предыдущих лекциях были решены некоторые задачи. Deductor будет подробно рассмотрен в одной из последующих лекций.

Наиболее известный представитель свободно распространяемого набора инструментов - пакет Weka (http://www.cs.waikato.ac.nz/ml/weka/index.html). Weka представляет собой набор алгоритмов машинного обучения для решения реальных Data Mining-проблем. Weka написана на Java и запускается практически со всех платформ.

Вторая группа задач представлена инструментами, реализующими следующие решения:

- инструментарий для поиска ассоциативных правил;
- агенты;
- оценивание, регрессии и прогнозирование;
- анализ связей;
- последовательные шаблоны и временные ряды;
- инструменты BI (Business Intelligence), Database and OLAP software;
- инструменты преобразования и очистки данных;
- библиотеки, компоненты и инструментальные наборы для разработчиков создания встроенных приложений Data Mining;
- Web Mining: анализ поведения сайтов, XML mining;
- поиск на Web;
- Audio and Video Mining.

Некоторые из этих групп инструментов будут более детально рассмотрены далее.

Среди поставщиков Data Mining можно выделить ряд компаний, основная цель которых - консультирование по применению Data Mining. Одна из наиболее известных среди них - компания Two Crows.

Программное обеспечение Data Mining для поиска ассоциативных правил

Коммерческие инструменты:

- Azmy SuperQuery (http://www.azmy.com/), поисковик ассоциативных правил;
- Clementine, набор от SPSS, включающий анализ рыночной корзины;
- IBM Intelligent Miner for Data (http://www.software.ibm.com/data/intelli-mine/);
- IREX (http://www.giwebb.com), сегментирование данных с целью оптимизации числовых результатов, например, прибыли;

- The LPA Data Mining Toolkit (http://www.lpa.co.uk/dtm.htm) поддерживает поиск ассоциативных правил в реляционных базах данных.
- Magnum Opus (http://www.rulequest.com/MagnumOpus-info.html) является быстрым инструментом поиска ассоциативных правил в данных, поддерживается операционными системами Windows, Linux и Solaris;
- Nuggets (http://www.data-mine.com/) - это набор, включающий поиск ассоциативных правил и другие алгоритмы;
- Megaputer Polyanalyst Suite (http://www.megaputer.com/), включает машину поиска ассоциативных правил;
- Purple Insight MineSet является набором визуального Data Mining, включающим визуализатор ассоциативных правил;
- Wizsoft модуль WizRule: нахождение ассоциативных правил и потенциальных ошибок данных; модуль WizWhy: использует ассоциативные правила для Data Mining;
- Xpertrule Miner 4.0 (http://www.attar.com/);
- XAffinity(TM), используется для идентификации сходств или шаблонов в транзакциях.

Свободно распространяемые инструменты:

- Apriori, инструмент для нахождения ассоциативных правил при помощи алгоритма Apriori;
- Apriori, FP-growth, Eclat and DIC implementations (http://www.adrem.ua.ac.be/) by Bart Goethals;
- ARtool (http://www.cs.umb.edu/), инструмент содержит набор алгоритмов для поиска ассоциативных правил в бинарных базах данных (binary databases);
- DM-II system (http://www.comp.nus.edu.sg/), инструмент включает алгоритм CBA для выполнения классификации на основе ассоциативных правил и некоторых других характеристик;
- FIMI, Frequent Itemset Mining Implementations (http://fimi.cs.helsinki.fi/) - является репозиторием, включающим программное обеспечение и базы данных.

Программное обеспечение для решения задач кластеризации и сегментации

Коммерческие инструменты:

- ClustanGraphics3, (http://www.clustan.com/) иерархический кластерный анализ "сверху вниз", поддерживаются мощные графические возможности, www.clustan.com;
- CViz Cluster Visualization, (http://www.alphaworks.ibm.com/tech/cviz)-продукт для анализа наборов данных с большой размерностью, обеспечивает визуализацию наполнения кластеров объектами;
- IBM Intelligent Miner for Data, (http://www-4.ibm.com/software/data/iminer/), включает два кластерных алгоритма;
- Neusciences aXi.Kohonen, (http://www.neusciences.com/), ActiveX Control для кластеризации алгоритмом Кохонена, включает Delphi-интерфейс;
- PolyAnalyst, (http://www.megaputer.com/), предлагает кластеризацию, основанную на алгоритме локализации аномалий (Localization of Anomalies, LA);
- StarProbe, (http://www.roselladb.com/starprobe.htm) основан на Web кросс-платформенной системе, включает методы кластеризации, нейронные сети, деревья решений, визуализацию и т.д.;
- Visipoint (http://www.visipoint.fi/). Кластеризация методом Самоорганизующихся Карт Кохонена (Self-Organizing Map clustering) и визуализация.

Свободно распространяемые инструменты:

- Autoclass C (http://ic.arc.nasa.gov/projects/bayes-group/autoclass/autoclass-c-program.html, http://ic.arc.nasa.gov), "обучение без учителя" при помощи Байесовских сетей от NASA, работает из-под операционных систем Unix и Windows;
- CLUTO (http://www.cs.umn.edu/~karypis/cluto, http://www.cs.umn.edu/~karypis/cluto). В инструменте реализован набор алгоритмов кластеризации, основанных на разделении данных;
- Databionic ESOM Tools (http://databionic-esom.sourceforge.net/). Инструмент представлен набором программ для кластеризации, визуализации и классификации, реализован алгоритм ESOM - выходящие самоорганизующиеся карты;
- MCLUST/EMCLUST (http://www.stat.washington.edu/fraley/mclust_home.html). В инструменте реализовано создание кластеров при помощи модельного подхода (model-based) и дискриминантного анализа, иерархическая кластеризация. Программная реализация инструмента - на Фортране с интерфейсом к S-PLUS;
- PermutMatrix (http://www.lirmm.fr/). Программное обеспечение для кластерного анализа, с хорошими графическими возможностями, здесь реализовано несколько методов иерархического кластерного анализа;
- PROXIMUS (http://www.cs.purdue.edu/homes/koyuturk/proximus/). Инструмент для сжатия размерности, кластеризации и обнаружения образцов в дискретных наборах данных;
- ReCkless (http://cde.iiit.net/RNNs/) является набором кластерных алгоритмов, основанных на концепции k-ближайших соседей. Инструмент перед проведением кластеризации выполняет поиск и идентификацию шумов и выбросов для уменьшения их влияния на результаты кластеризации;
- Snob (http://www.csse.monash.edu.au/), программа кластеризации на основе MML (Minimum Message Length - Минимальная Длина Сообщения);
- SOM in Excel (http://www.geocities.com/adotsaha/NN/SOMinExcel.html), реализация метода самоорганизующихся карт Кохонена в Microsoft Excel от Angshuman Saha.

Как видим из описания, многие программные продукты совмещают в себе реализацию нескольких методов, в частности, очень часто вместе с кластерными методами также реализованы и методы визуализации. Некоторые инструменты ориентированы на работу только с дискретными данными. Это следует учитывать при выборе программного обеспечения.

Программное обеспечение для решения задач классификации

Существует множество инструментов для решения задач классификации. Инструменты этой группы строят модели, которые делят исходный набор данных на 2 или более дискретных класса. Инструменты классификации, в соответствии с используемыми методами, делятся на следующие категории: правила, деревья решений, нейронные сети, Байовские сети, метод опорных векторов и другие. Этот список практически соответствует тому набору методов классификации, который был рассмотрен во втором разделе курса лекций.

Программное обеспечение Data Mining для решения задач оценивания и прогнозирования

Примером коммерческого программного обеспечения этой группы является инструмент Alyuda Forecaster XL (http://www.alyuda.com/forecasting-tool-for-excel.htm).

Инструмент реализован в виде Excel-надстройки и предназначен для решения задач прогнозирования и оценивания с использованием нейронных сетей.

Подобный инструмент от российских разработчиков - фирмы НейрОК - Excel-надстройка ExcelNeuralPackage (http://www.neurok.ru/demo/enp/demo_enp.htm).

В инструменте реализованы две базовые парадигмы нейронных сетей - многослойный персептрон и сети Кохонена. С указанной страницы можно загрузить free-версию и подробное руководство пользователя.

Выводы

Как мы видим, рынок программного обеспечения Data Mining представлен множеством инструментов, на нем идет постоянная конкурентная борьба за потребителя. Такая конкуренция порождает новые качественные решения. Все большее число поставщиков стремятся объединить в своих инструментах как можно большее число современных методов и технологий. Data Mining-инструменты чаще всего рассматриваются как составная часть рынка Business Intelligence, который, несмотря на некоторый общий спад в индустрии информационных технологий, уверенно и постоянно развивается.

В то же время некоторые специалисты отмечают отставание существующего программного обеспечения от теоретических разработок в связи со сложностью программной реализации некоторых новых теоретических разработок методов и алгоритмов Data Mining.

В целом, можно резюмировать, что рынок Business Intelligence, в том числе рынок инструментов Data Mining, настолько широк и разнообразен, что любая компания может выбрать для себя инструмент, который подойдет ей по функциональности и по возможностям бюджета.

Инструменты Data Mining. SAS Enterprise Miner

Программный продукт SAS Enterprise Miner (разработчик SAS Institute Inc., [102]) - это интегрированный компонент системы SAS, созданный специально для выявления в огромных массивах данных информации, которая необходима для принятия решений. Разработанный для поиска и анализа глубоко скрытых закономерностей в данных SAS, Enterprise Miner включает в себя методы статистического анализа, соответствующую методологию выполнения проектов Data Mining (SEMMA) и графический интерфейс пользователя. Важной особенностью SAS Enterprise Miner является его полная интеграция с программным продуктом SAS Warehouse Administrator, предназначенным для разработки и эксплуатации информационных хранилищ, и другими компонентами системы SAS. Разработка проектов Data Mining может выполняться как локально, так и в архитектуре клиент-сервер.

Назначение пакета SAS Enterprise Miner. Пакет SAS Enterprise Miner позволяет оптимизировать процесс Data Mining в целом, начиная от организации доступа к данным и заканчивая оценкой готовой модели [104].

Пакет поддерживает выполнение всех необходимых процедур в рамках единого интегрированного решения с возможностями коллективной работы и поставляется как распределенное клиент-серверное приложение, что особенно удобно для осуществления анализа данных в масштабах крупных организаций. Пакет SAS Enterprise Miner предназначен для специалистов по анализу данных, маркетинговых аналитиков, маркетологов, специалистов по анализу рисков, специалистов по выявлению мошеннических действий, а также инженеров и ученых, ответственных за принятие ключевых решений в бизнесе или исследовательской деятельности.

Пакет SAS Enterprise Miner обеспечивает эффективную обработку огромных объемов данных и предоставляет простые способы публикации результатов анализа для различных аудиторий, что позволяет встраивать эти модели в бизнес-процессы предприятия.

Обзор программного продукта

Пакет SAS Enterprise Miner 5.1 поставляется в виде современной распределенной клиент-серверной системы для Data Mining или для углубленного анализа данных в крупных организациях. Пакет позволяет оптимизировать процессы анализа данных, поддерживая все необходимые шаги в рамках единого решения, а также возможности гибкого сотрудничества больших рабочих групп в рамках единого проекта. Система обеспечивает расширенную интеграцию с системами управления данными и развертывания моделей, а благодаря широкому спектру выбора конфигурации пакета в зависимости от требований бизнеса нет необходимости приобретать системы специализированных решений.

Графический интерфейс (GUI) для анализа данных

В пакете SAS Enterprise Miner реализован подход, основанный на создании диаграмм процессов обработки данных и позволяющий устранить необходимость ручного кодирования и ускорить разработку моделей благодаря методике Data Mining SEMMA. Среда для формирования диаграмм процессов обработки данных пакета SAS Enterprise

Miner устраняет необходимость ручного кодирования, диаграммы выступают в качестве самоописательных шаблонов, которые можно изменять или применять для решения новых проблем, не повторяя анализ с самого начала. Существует возможность обмена диаграммами между аналитиками в масштабах предприятия.

Графический пользовательский интерфейс пакета является интерфейсом типа "указать и щелкнуть". С его помощью пользователи могут выполнить все стадии процесса Data Mining от выбора источников данных, их исследования и модификации до моделирования и оценки качества моделей с последующим применением полученных моделей, как для обработки новых данных, так и для поддержки процессов принятия решений. Главное окно SAS Enterprise Miner представлено на рис. 23.1.

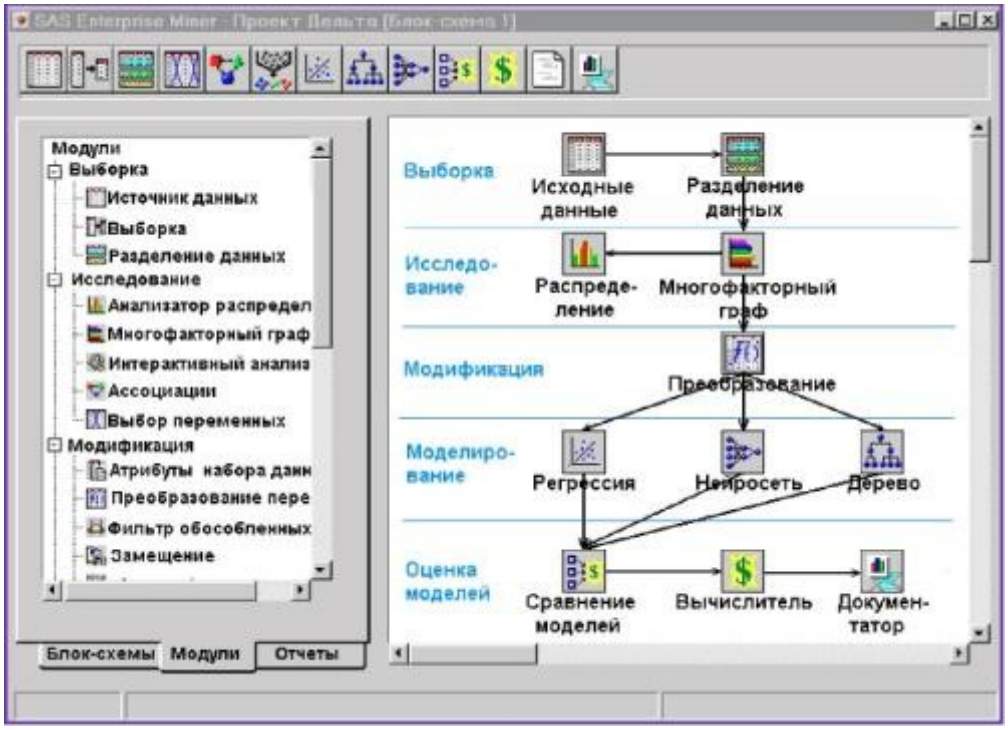

Рис. 23.1. Главное окно SAS Enterprise Miner

Инструментарий для углубленного интеллектуального анализа данных

Новая версия пакета SAS Enterprise Miner 5.1 спроектирована с использованием архитектуры Java-клиент / SAS-сервер, которая позволяет отделить вычислительный сервер, выполняющий обработку данных, от пользовательского интерфейса. Это обеспечивает гибкость в выборе конфигурации эффективного решения - от однопользовательской системы до крупнейших решений корпоративного масштаба. Обработку данных можно выполнять на мощных серверах, а конечные пользователи могут перемещаться из офиса домой или в отдаленные филиалы, не теряя связи с аналитическими проектами и сервисами. Некоторые серверные задачи, интенсивно использующие ресурсы процессора, например сортировка и агрегация данных, отбор переменных и регрессионный анализ, сделаны многопоточными, что позволяет распределить их выполнение между несколькими процессорами.

Процессы в Enterprise Miner могут работать параллельно и в асинхронном режиме. Масштабные или повторяющиеся процессы обучения модели или скоринга могут быть выполнены в виде пакетного задания, назначенного на наименее загруженные часы работы аналитического сервера.

Набор инструментов для подготовки, агрегации и исследования данных

Пакет SAS Enterprise Miner предлагает различные инструменты для осуществления подготовки данных, которые дают возможность, например, сделать выборку или разбивку данных, осуществить вставку недостающих значений, провести кластеризацию, объединить источники данных, устранить лишние переменные, выполнить обработку на языке SAS посредством специализированного узла SAS code, осуществить преобразование переменных и фильтрацию недостоверных данных. Пакет оснащен функциями описательной статистики, а также расширенными средствами визуализации, которые позволяют исследовать сверхбольшие объемы данных, представленных в виде многомерных графиков, и производить графическое сравнение результатов моделирования.

Платформенно-независимый пользовательский интерфейс пакета SAS Enteprise Miner 5.1 создан на базе Java и предоставляет пользователям широкий набор средств статистической графики с гибкими возможностями настройки и управления. Для создания специальных графиков предусмотрен Java-мастер. Все графики и лежащие в их основе таблицы динамически связаны между собой и поддерживают интерактивные режимы работы.

Интегрированный комплекс разнообразных методов моделирования

Пакет SAS Enterprise Miner предоставляет набор инструментов и алгоритмов прогностического и описательного моделирования, включающий деревья решений, нейронные сети, самоорганизующиеся нейронные сети, методы рассуждения, основанные на механизмах поиска в памяти (memorybased reasoning), линейную и логистическую регрессии, кластеризацию, ассоциации, временные ряды и многое другое.

Интеграция различных моделей и алгоритмов в пакете Enterprise Miner позволяет производить последовательное сравнение моделей, созданных на основе различных методов, и оставаться при этом в рамках единого графического интерфейса. Встроенные средства оценки формируют единую среду для сравнения различных методов моделирования, как с точки зрения статистики, так и с точки зрения бизнеса, позволяя выявить наиболее подходящие методы для имеющихся данных. Результатом является качественный анализ данных, выполненный с учетом специфических проблем конкретного бизнеса.

Интегрированные средства сравнения моделей и пакеты результатов

Пакет SAS Enterprise Miner оснащен рядом встроенных функций контроля, работающих в рамках единой оболочки и обеспечивающих сравнение результатов различных методов моделирования как с точки зрения статистики, так и с точки зрения бизнеса.

Полученные модели можно публиковать для совместного использования в рамках предприятия при помощи репозитария моделей, представляющего собой первую на рынке систему управления моделями. Управление моделями обеспечивает модуль Enterprise

Miner Repository. Пакет предоставляет ряд встроенных оценочных функций, позволяющих сравнить результаты различных методов моделирования, как в терминах бизнеса, так и с использованием статистической диагностики. Это дает возможность измерить эффективность модели в терминах ее прибыльности. Аналитики могут наблюдать за обновляемыми моделями и отслеживать улучшение их точности с течением времени. Созданные диаграммы можно сохранять и импортировать в виде XML-файлов, что облегчает процесс их передачи другим аналитикам. SAS Enterprise Miner позволяет создавать сжатые пакеты с результатами моделирования, в которых хранится вся информация о процессе обработки данных, включая предварительную обработку данных, логику моделирования, результаты моделирования и оценочный код. Эти пакеты результатов могут быть зарегистрированы на сервере метаданных (SAS Metadata Server), откуда их потом могут извлекать для изучения специалисты по анализу данных и представители бизнеса. Специальный модуль с Web-интерфейсом предусмотрен для просмотра репозитария моделей.

Скоринг по модели и простота развертывания модели

Итогом работ по интеллектуальному анализу данных является развертывание созданной модели - это заключительная стадия, на которой реализуется экономическая отдача от проведенных исследований. Процесс применения модели к новым данным, известный как скоринг, часто требует ручного написания или преобразования программного кода. Пакет SAS Enterprise Miner автоматизирует процесс подбора коэффициентов и предоставляет готовый программный код для скоринга на всех стадиях создания модели, поддерживает создание различных программных сред для развертывания модели на языках SAS, C, Java и PMML. Этот программный код может использоваться в различных средах (в пакетном режиме или в реальном времени) в системе SAS, в Web или непосредственно в реляционных базах данных. Пакет создает код для аналитических моделей и для предварительной обработки данных.

Когда оценочный код создан, можно проводить скоринг наборов данных как непосредственно в Enterprise Miner, так и экспортировать скоринг-код и выполнить скоринг на другой машине, а также отторгнуть формулу для скоринга для применения в пакетном режиме или в режиме реального времени в Web или непосредственно в реляционных базах данных.

Гибкость благодаря открытости и расширяемости

Пакет Enterprise Miner предоставляет настраиваемую и расширяемую среду интеллектуального анализа данных, позволяющую добавлять инструментальные средства и интегрировать персонифицированный код на языке SAS. Стандартную инструментальную библиотеку, входящую в состав пакета SAS Enterprise Miner 5.1, легко расширить при помощи средств настройки, использующих язык SAS и XML-логику. Кроме того, есть возможность использования экспериментального интерфейса Java API, позволяющего встраивать процессы пакета Enterprise Miner в различные пользовательские приложения. Эта возможность может оказаться особенно плодотворной для компаний, стремящихся создать собственное аналитическое приложение, которое будет сочетать в себе, например, возможности создания OLAP-отчетов и выполнения интеллектуального анализа данных в рамках единого интерфейса.

Встроенная стратегия обнаружения данных

Интеллектуальный анализ данных становится особенно эффективным, если он является составной частью интегрированной стратегии предоставления информации. Пакет Enterprise Miner органично интегрируется с другими предложениями SAS, например, пакетом SAS ETL Studio, средствами аналитической обработки OLAP, прогностическим и другими аналитическими модулями, а также с приложением SAS Text Miner. Подход SAS к созданию информационно-аналитических систем кратко будет изложен в конце лекции.

Распределенная система интеллектуального анализа данных, ориентированная на крупные предприятия

Пакет SAS Enterprise Miner может быть развернут с использованием Web-портала для тонких клиентов, что обеспечивает удобный доступ к пакету для множества пользователей при минимальных затратах на обслуживание клиентских программ. Пакет SAS Enterprise Miner поддерживает серверные системы Windows, а также различные UNIX~платформы. Технические характеристики пакета изложены в конце этой лекции.

Основные характеристики пакета SAS Enterprise Miner 5.1

Интерфейсы

Простой графический интерфейс, создающий диаграммы процессов обработки данных:

- Быстрое создание большого числа качественных моделей.
- Возможность доступа через Web-интерфейс.
- Доступ к среде программирования SAS.
- Возможность обмена диаграммами в формате XML.
- Возможность повторного использования диаграмм в качестве шаблонов для других проектов и пользователей.

Пакетная обработка:

- Включает в себя все те же возможности, что и графический интерфейс.
- Основана на языке SAS macro.

Экспериментальный интерфейс Java API.

Репозитарий моделей с Web-интерфейсом:

- Управление большими портфелями моделей.
- Поиск моделей по заданному алгоритму, целевой переменной и т.п.
- Публикация результатов в виде ступенчатых диаграмм, деревьев и скоринг-кодов, удобных для специалистов в области бизнеса и анализа данных.

Масштабируемая обработка

- Серверная обработка - обучение модели в асинхронном режиме. Аккуратная остановка обработки (по заданным критериям).
- Параллельная обработка - одновременный запуск нескольких диаграмм или инструментов.

- Многопоточные прогностические алгоритмы.
- Все хранение и обработка данных - на серверах.

Доступ к данным

Доступ более чем к 50 различным файловым структурам.

Интеграция с пакетом SAS ETL Studio посредством SAS Metadata Server:

- SAS ETL Studio можно использовать для определения исходных, обучающих таблиц для пакета Enterprise Miner.
- SAS ETL Studio можно использовать для извлечения и развертывания скоринг-кода пакета Enterprise Miner.

Выборки

- Простая случайная.
- Стратифицированная.
- Взвешенная.
- Кластерная.
- Систематическая.
- Первые N наблюдений.
- Выборка редких событий.

Разбивка данных

- Создание обучающих, проверочных и тестовых наборов данных.
- Обеспечение качественного обобщения моделей на основании контрольных данных.
- Стандартная стратификация по целевому классу.
- Сбалансированная разбивка по любой классовой переменной.

Преобразования

- Простые: логарифмическое, квадратный корень, обратное, квадратичное, экспоненциальное, стандартизованное.
- Накопительные: bucketed (с разбивкой по областям), квантильное, оптимизированная разбивка по взаимосвязи с целевымм значениями.
- Оптимизированные: максимизация нормализации, максимизация корреляции с целевыми значениями, выравнивание распределения по целевым уровням.

Фильтрация недостоверных данных

- Применение различных распределительных порогов, позволяющих исключить значения из экстремальных интервалов.
- Объединение классовых значений, встречающихся менее n раз.

Замена данных

- С использованием мер центрированности.
- На основе распределения.
- Заполнение дерева суррогатными значениями.
- Методом усреднения расстояний.

- С использованием устойчивых М-оценок.
- С использованием стандартных констант.

Описательная статистика

Одномерные статистические таблицы и графики:

- Интервальные переменные n, среднее, медиана, минимум, максимум, стандартное отклонение, масштабированное отклонение и процент отсутствия.
- Классовые переменные число категорий, счетчики, модальные, процентные модальные, процент отсутствия.
- Графики распределения.
- Статистическая разбивка для каждого уровня целевых классов.

Двумерные статистические таблицы и графики:

- Упорядоченные графики корреляции Пирсона и Спирмана.
- Упорядоченный график хи-квадрат с возможностью группировки непрерывных исходных данных по n группам.
- График коэффициентов вариации.

Отбор переменных по logworth-критерию.

Другие интерактивные графики:

- "Тепловые" карты, отражающие корреляцию или ассоциацию типа хи-квадрат первоначальных значений с целевыми признаками по сегментам.
- Графики стоимости переменных, ранжирующие первоначальные значенияна основании их стоимости по целевому признаку.
- Распределения классовых переменных по целевым признакам и/или сегментным переменным.

Графики масштабированного среднего отклонения.

Графика/визуализация

Графики, создаваемые в пакетном и интерактивном режимах: графики разброса, гистограммы, многомерные графики, круговые диаграммы, диаграммы с областями, пузырьковые диаграммы.

Удобный Java-мастер для построения графиков:

- Заголовки и сноски.
- Возможность применения к данным предложения WHERE.
- Возможность выбора из нескольких цветовых схем.
- Простота масштабирования осей.
- Использование данных, полученных в результате анализа в пакете Enterprise Miner, для создания специализированных графиков.

Динамическая загрузка данных в клиентское приложение при помощи нескольких методик выборки.

Графики и таблицы интерактивно связаны между собой и поддерживают выполнение таких операций как очистка и связывание.

Удобное копирование данных и графиков в другие приложения, а также возможность их сохранения в виде файлов GIF или TIF.

Кластеризация

- По выбору пользователя или автоматический - выбор k лучших кластеров.
- Различные стратегии кодирования классовых переменных в процессе анализа.
- Управление недостающими данными.
- Графики профилей переменных сегментов, отражающие распределение исходных данных и других факторов в рамках каждого кластера.
- Профиль дерева решений, использующий исходные данные для составления прогноза о принадлежности кластеру.
- Оценочный код PMML.

Анализ рыночной корзины

Выявление ассоциаций и причинно-следственных связей:

- Сетевой график правил, упорядоченный по степени достоверности.
- Статистические графики подъема, достоверности, прогноза достоверности и поддержки правил.
- Статистическая гистограмма частотных показателей в заданных границах поддержки и достоверности.
- График зависимости разброса достоверности от прогнозируемой достоверности.
- Таблица описания правил.
- Сетевой график правил.

Органичная интеграция правил с другими исходными данными обеспечивает расширенное прогностическое моделирование.

Удобный вывод правил обеспечивает кластеризацию клиентов по их покупательным и поведенческим характеристикам.

Оценочный код PMML.

Анализ Web-активности

- Масштабируемое и эффективное выявление наиболее популярных Интернет-маршрутов на основе анализа данных об Интернет-активности пользователей.
- Выявление наиболее частых последовательностей в последовательных данных любого типа.

Уменьшение размерности

Выбор переменных:

- Удаление переменных, не связанных с целевыми признаками, на основе критериев отбора хи-квадрат или R2.

- Удаление переменных из иерархий.
- Удаление переменных со многими недостающими значениями.
- Сокращение числа классовых переменных с большим количеством уровней.
- Группировка непрерывных исходных данных для выявления нелинейных взаимосвязей.
- Выявление взаимодействий.

Главные компоненты:

- Вычисление собственных значений и собственных векторов на основании матриц корреляции и ковариации.
- Графики: масштабированное отклонение, логарифмические собственные значения, кумулятивные пропорциональные собственные значения.
- Исследование выбранных основных компонентов при помощи методов предиктивного моделирования.

Исследование временных рядов:

- Сокращение объемов транзакционных данных на основе формирования временных рядов с использованием разнообразных методов аккумуляции и преобразования.
- Методы анализа включают сезонный анализ, анализ тенденций, анализ временных областей, сезонную декомпозицию.
- Исследование сокращенных временных рядов при помощи методов кластерного и предиктивного моделирования.

Управление временными метриками при помощи описательных данных.

Утилита SAS Code Node

- Обеспечивает запись кода SAS для упрощения сложных процедур подготовки и преобразования данных.
- Позволяет использовать процедуры других продуктов SAS.
- Поддерживает импорт внешних моделей.
- Позволяет создавать собственные модели и узлы Enterprise Miner.
- Содержит макропеременные, упрощающие ссылку на источники данных, переменные и т.п.
- Имеет расширяемую логику формирования оценочного кода.

Исчерпывающие средства моделирования

- Выбор моделей на базе обучающей, проверочной или тестовой выборки данных с использованием различных критериев, таких как: прибыли или убытки, AIC, SBC, среднеквадратичная ошибка, частота ошибок классификации, ROC, Джини, KS (Колмогорова-Смирнова).
- Поддерживает двоичные, номинальные, порядковые и интервальные исходные данные и целевые признаки.
- Удобный доступ к оценочному коду и всем источникам данных.
- Отображение нескольких результатов в одном окне позволяет лучше оценить эффективность модели.

Регрессии

- Линейная и логистическая.

- Пошаговая, с прямой и обратной выборкой.
- Построитель условий для уравнений: полиномиальных, основных взаимодействий, поддержка иерархии эффектов.
- Перекрестная проверка.
- Правила для иерархии эффектов.
- Методы оптимизации: сопряженные градиенты, метод двойных ломаных, метод Ньютона-Рафсона с линейным или гребневым поиском, квазиньютоновский метод, метод доверительных областей.
- Оценочный код PMML.

Деревья решений

Общая методология:

- CHAID (автоматическое выявление взаимодействия по методу хи-квадрат).
- Деревья классификации и регрессии.
- C 4.5.
- Отбор деревьев на основе целевых значений прибыльности или роста с соответствующим отсечением ветвей.

Критерии расщепления: вероятностный критерий хи-квадрат, вероятностный

F-критерий, критерий Джини, критерий энтропии, уменьшение дисперсии.

Автоматический вывод идентификаторов листьев дерева в качестве входных

значений для последующего моделирования.

Отображение правил на английском языке.

Вычисление значимости переменных для предварительного отбора.

Уникальное представление консолидированной диаграммы дерева.

Интерактивная работа с деревом на настольном ПК:

- Интерактивное расширение и обрезание деревьев.
- Задание специальных точек разбиения, включая двоичные или многовариантные разбиения.
- Свыше 13 динамически связанных таблиц и графиков, позволяющих произвести более качественную оценку дерева.
- Возможность распечатать диаграмму дерева на одном или нескольких листах.

В основе - новая быстрая процедура ARBORETUM.

Нейронные сети

Узел нейронной сети:

- Гибкие архитектуры сетей с развитыми функциями комбинирования и активации.
- 10 методов обучения сети.

- Предварительная оптимизация.
- Автоматическая стандартизация входных параметров.
- Поддержка направленных связей.

Узел самоорганизующейся нейронной сети:

- Автоматизированное создание многоуровневых персептронов для поиска оптимальной конфигурации.
- Выбор функций типа и активации из четырех различных типов архитектур.
- Оценочный код PMML.

Узел нейронной сети анализа данных (DM Neural node):

- Создание модели с уменьшением размерности и выбором функций.
- Быстрое обучение сети.
- Линейное и нелинейное оценивание.

Двухуровневое моделирование

- Последовательное и параллельное моделирование для классовых и интервальных целевых признаков.
- Выбор модели в виде дерева решений, регрессии или нейронной сети на каждом уровне.
- Управление применением прогноза для классов к прогнозу интервалов.
- Точная оценка экономической выгодности клиентов.

Методы вывода путем сопоставления

- Метод отбора ближайших k-соседей для категоризации или прогноза наблюдений.
- Запатентованные методы создания дерева и поиска с уменьшенной размерностью.

Множества моделей

- Объединение прогнозов моделей для создания потенциально более сильного решения.
- Среди методов: усреднение, мажоритарная выборка, выбор максимального значения.

Сравнение моделей

- Сравнение нескольких моделей в рамках единой инструментальной оболочки для всех источников данных.
- Автоматический выбор лучшей модели на основе заданного пользователем критерия.
- Расширенная статистика соответствия и диагностики.
- Ступенчатые диаграммы.
- Кривые ROC.
- Диаграммы прибылей и убытков с возможностью выбора решения.
- Матрица неточностей (классификации).
- График распределения вероятностных оценок классовых целевых признаков.
- Ранжирование и распределение оценок интервальных целевых признаков.

Количественная оценка

- Интерактивная количественная оценка узла в рамках графического интерфейса.
- Автоматическая генерация оценочного кода на языках SAS, C, Java и PMML.

- Моделирование сбора, кластеризации, преобразования и вычисления недостающих значений для оценочных кодов на языках SAS, C и Java.
- Развертывание моделей в нескольких средах.

Инструментальные средства

- Узел удаления переменных.
- Узел слияния данных.
- Узел метаданных, позволяющий изменять столбцы метаданных, например роль, уровень измерений и порядок.

Специализированное хранилище данных

Важность использования технологий хранилищ данных как информационной основы для Data Mining уже рассматривалась нами. Структура хранилища, оптимизированная под задачи аналитической обработки, позволяет свести к минимуму потери времени на поиск нужных данных и получение промежуточных результатов.

Подход SAS к созданию информационно-аналитических систем

Подход компании SAS к созданию информационно-аналитических систем стандартизован в рамках SAS Intelligent Warehousing solutions, рис. 23.2.

Рис. 23.2. Структура SAS Intelligent Warehousing solutions

Этот подход предусматривает:

- простые в использовании эффективные методы извлечения данных из ERP/OLTP-систем, баз данных и других источников без применения микропрограммирования на языке управления данными ERP/OLTP-системы (семейство программных продуктов SAS/ACCESS).
- высокотехнологичные методы очистки исходных данных и их подготовки для загрузки в хранилище (SAS Data Quality-Cleanse).
- средства проектирования и администрирования хранилищ данных (SAS/Warehouse Administrator).
- технологию физического хранения больших объемов данных (SAS Scalable Performance Data Server).
- методы интеллектуального анализа данных:

    - OLAP-анализа (SAS OLAP Server),
    - эконометрического моделирования и расчета временных рядов (SAS/ETS),
    - исследования операций и оптимизация (SAS/OR),
    - имитационного моделирования (SAS/IML),
    - статистического анализа (SAS/STAT),
    - нейросетевого и других методов углубленного анализа данных (SAS Enterprise Miner).
- дружественные к пользователю эффективные средства отчетности (SAS/Enterprise Guide, SAS/EIS, SAS/InterNet, AppDevStudio),
- быстрое получение результата за счет специальной методологии проектирования (SAS/ Rapid Result) и, как следствие,
- быстрый возврат инвестиций системы коллективного доступа к информационному хранилищу (хранилищу данных) посредством Web-технологий (Web-порталов). Для разработки Web-порталов компания SAS предлагает решение SAS Information Delivery Portal.

Технические требования пакета SASR Enterprise Miner

Поддерживаемые клиентские платформы Microsoft Windows (32-разрядная)

Windows NT 4 Workstation, Windows 2000 Professional, Windows XP Professional, AIX (64-разрядная) релиз 5.1, HPUX (64-разрядная) релиз 11 i (11.11), Solaris 8 или 9 (64-разрядная)

Поддерживаемые серверные платформы Microsoft Windows (32-разрядная, 64-разрядная) Windows NT 4 Server 4.0, Windows 2000, Windows Server 2003, AIX (64-разрядная) релиз 5.1.

HPUX (64-разрядная), релиз 11 i (11.11), Linux для Intel (32-разрядная)

Red Hat Linux 8.0, Red Hat Advanced Server 2.1, SuSE Linux Enterprise Server 8 Solaris 8 или 9 (64-разрядная), Tru64 UNIX (64-разрядная) Version 5.1A или 5.1 B.

Минимальная тактовая частота процессора 1 ГГц.

Требования к оперативной памяти: 512 МБ для клиента, 512 МБ для сервера.

Требования к объему дисковой памяти: не менее 40 МБ для клиента 3 ГБ для сервера (в среднем для Win XP см. замечания по инсталляции SAS).

Требования к ПО

Основной пакет SAS, SAS/STAT, Webклиент Java 1.4.1, в противном случае не нужен (в состав SAS входит частная версия JRE 1.4.1), что делает пакет предпочтительным для организаций, ведущих крупномасштабные проекты в области интеллектуального анализа данных.

Инструменты Data Mining. Система PolyAnalyst

Назначение системы. Система PolyAnalyst предназначена для автоматического и полуавтоматического анализа числовых баз данных и извлечения из сырых данных практически полезных знаний. PolyAnalyst находит многофакторные зависимости между переменными в базе данных, автоматически строит и тестирует многомерные нелинейные модели, выражающие найденные зависимости, выводит классификационные правила по обучающим примерам, находит в данных многомерные кластеры, строит алгоритмы решений. Разработчик системы PolyAnalyst - российская компания Megaputer Intelligence или "Мегапьютер" [105].

Архитектура системы

По своей природе PolyAnalyst является клиент-серверным приложением. Пользователь работает с клиентской программой PolyAnalyst Workplace. Математические модули выделены в серверную часть - PolyAnalyst Knowledge Server. Такая архитектура предоставляет естественную возможность для масштабирования системы: от однопользовательского варианта до корпоративного решения с несколькими серверами. PolyAnalyst написан на языке C++ с использованием спецификации Microsoft's COM (ActiveX). Эта спецификация устанавливает стандарт коммуникации между программными компонентами. Архитектура системы PolyAnalyst представлена на рис. 24.1.

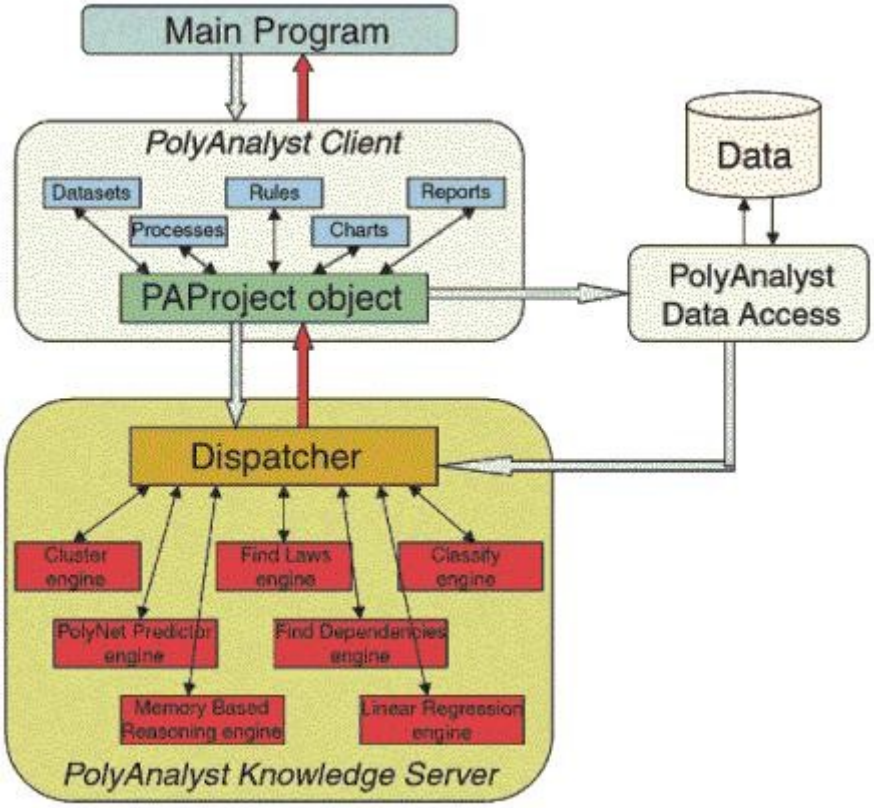

Рис. 24.1. Архитектура системы PolyAnalyst

Математические модули (Exploration Engines) и многие другие компоненты PolyAnalyst выделены в отдельные динамические библиотеки и доступны из других приложений. Это дает возможность интегрировать математику PolyAnalyst в существующие ИС, например, в CRM- или ERP- системы.

PolyAnalyst Workplace - лаборатория аналитика

Workplace - это клиентская часть программы, ее пользовательский интерфейс. Workplace представляет собой полнофункциональную среду для анализа данных, которая показана на рис. 24.2.

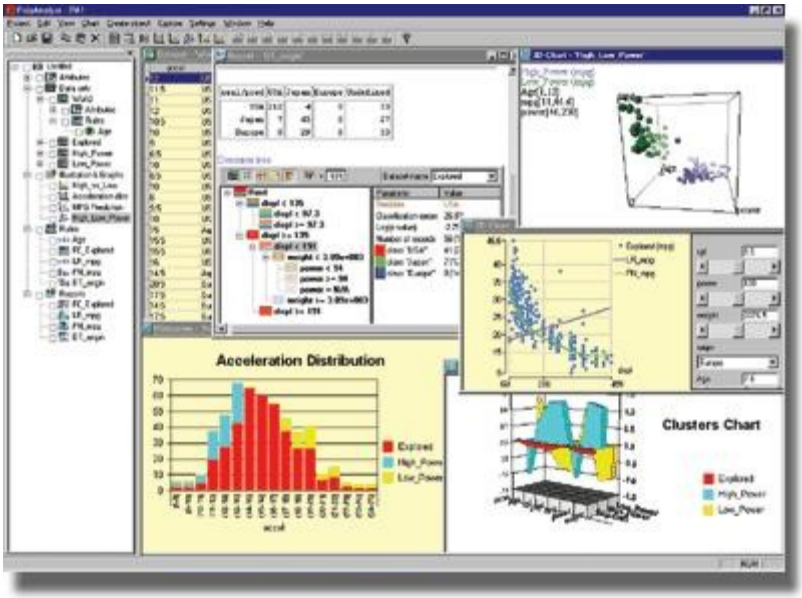

Рис. 24.2. Пользовательский интерфейс PolyAnalyst

Основные черты пользовательского интерфейса программы: развитые возможности манипулирования с данными, графика для представления данных и визуализации результатов, мастера создания объектов, сквозная логическая связь между объектами, язык символьных правил, интуитивное управление через drop-down и pop-up меню, подробная контекстная справка.

Единицей Data Mining исследования в PolyAnalyst является "проект". Проект объединяет в себе все объекты исследования, дерево проекта, графики, правила, отчеты и т.д. Проект сохраняется в файле внутреннего формата системы. Отчеты исследований представляются в формате HTML и доступны через Интернет.

Аналитический инструментарий PolyAnalyst

Версия PolyAnalyst 4.6 включает 18 математических модулей, основанных на различных алгоритмах Data и Text Mining. Большинство из этих алгоритмов являются Know-How компании Мегапьютер и не имеют аналогов в других системах.

- моделирование,
- прогнозирование,
- кластеризация,

- классификация,
- текстовый анализ.

Ниже дается краткая характеристика математическим алгоритмам PolyAnalyst.

Модули для построения числовых моделей и прогноза числовых переменных

Модуль Find Laws (FL) - построитель моделей

Модуль FL - это сердце всей системы. Алгоритм предназначен для автоматического нахождения в данных нелинейных зависимостей (вид которых не задается пользователем) и представления результатов в виде математических формул, включающих в себя и блоки условий. Способность модуля FL автоматически строить большое многообразие математических конструкций делает его уникальным инструментом поиска знания в символьном виде. Алгоритм основан на технологии эволюционного, или как ее еще называют, генетического программирования, впервые реализованной в коммерческих программах компанией "Мегапьютер".

PolyNet Predictor (PN) - полиномиальная нейронная сеть

Работа этого алгоритма основана на построении иерархической структуры, подобной нейронной сети. При этом сложность этой сетевой структуры и другие ее параметры подбираются динамически на основе свойств анализируемых данных. Если создаваемая сетевая структура не является слишком сложной, то может быть построено эквивалентное ей выражение на языке символических правил системы. Если же сеть слишком большая, то правило не может быть показано, однако его можно вычислить, или - иными словами, применить к исходным или новым данным для построения прогноза. Данный алгоритм чрезвычайно эффективен в инженерных и научных задачах, когда требуется построить надежный прогноз для числовой переменной.

Stepwise Linear Regression (LR) - пошаговая многопараметрическая линейная регрессия

Линейная регрессия, как широко распространенный метод статистического исследования, включена во многие статистические пакеты и электронные таблицы. Однако, реализация этого модуля в системе PolyAnalyst имеет свои особенности, а именно: автоматический выбор наиболее значимых независимых переменных и тщательная оценка статистической значимости результатов. Нужно заметить, что в данном случае значимость отличается от значимости единичной регрессионной модели, так как в течение одного запуска данного вычислительного процесса может быть проверено большое число регрессионных моделей.

Алгоритм работает очень быстро и применим для построения линейных моделей на смешанных типах данных.

Memory based Reasoning (MR) - метод "ближайших соседей"

В системе PolyAnalyst используется модификация известного алгоритма "метод ближайших соседей".

Идея метода была рассмотрена нами ранее. Особенность и отличие реализации алгоритма "ближайших соседей" в системе PolyAnalyst от известных аналогов этого метода

заключается в оптимизации меры близости и количества записей для усреднения на основе генетических алгоритмов. Алгоритм MR используется для предсказания значений числовых переменных и категориальных переменных, включая текстовые (string data type), а также для классификации на два или несколько классов.

Алгоритмы кластеризации

Find Dependencies (FD) - N-мерный анализ распределений

Данный алгоритм обнаруживает в исходной таблице группы записей, для которых характерно наличие функциональной связи между целевой переменной и независимыми переменными, оценивает степень (силу) этой зависимости в терминах стандартной ошибки, определяет набор наиболее влияющих факторов, отсеивает отскочившие точки. Целевая переменная для FD должна быть числового типа, в то время как независимые переменные могут быть и числовыми, и категориями, и логическими.

Алгоритм работает очень быстро и способен обрабатывать большие объемы данных. Его можно использовать как препроцессор для алгоритмов FL, PN, LR, так как он уменьшает пространство поиска, а также как фильтр отскочивших точек или, в обратной постановке, как детектор исключений. FD создает правило табличного вида, однако, как и все правила PolyAnalyst, оно может быть вычислено для любой записи таблицы.

Find Clusters (FC) - N-мерный кластеризатор

Этот метод применяется тогда, когда надо выделить в некотором множестве данных компактные типичные подгруппы (кластеры), состоящие из близких по своим характеристикам записей. Алгоритм FC сам определяет набор переменных, для которых разбиение наиболее значимо. Результатом работы алгоритма является описание областей (диапазонов значений переменных), характеризующих каждый обнаруженный кластер, и разбиение исследуемой таблицы на подмножества, соответствующие кластерам. Если данные являются достаточно однородными по всем своим переменным и не содержат "сгущений" точек в каких-то областях, этот метод не даст результатов. Надо отметить, что минимальное число обнаруживаемых кластеров равно двум - сгущение точек только в одном месте в данном алгоритме не рассматривается как кластер. Кроме того, этот метод в большей степени, чем остальные, предъявляет требования к наличию достаточного количества записей в исследуемой таблице, а именно: минимальное количество записей в таблице, в которой может быть обнаружено N кластеров, равно (2N-1)4.

Алгоритмы классификации

В пакете PolyAnalyst имеется богатый инструментарий для решения задач классификации, т.е. для нахождения правил отнесения записей к одному из двух или к одному из нескольких классов.

Classify (CL) - классификатор на основе нечеткой логики

Алгоритм CL предназначен для классификации записей на два класса. В основе его работы лежит построение так называемой функции принадлежности и нахождения порога разделения на классы. Функция принадлежности принимает значения от окрестности 0 до окрестности 1. Если возвращаемое значение функции для данной записи больше порога,

то эта запись принадлежит к классу "1", если меньше, то классу "0" соответственно. Целевая переменная для этого модуля должна быть логического типа.

Discriminate (DS) - дискриминация

Данный алгоритм является модификацией алгоритма CL. Он предназначен для того, чтобы выяснить, чем данные из выбранной таблицы отличаются от остальных данных, включенных в проект, иными словами, для выделения специфических черт, характеризующих некоторое подмножество записей проекта. В отличие от алгоритма CL, он не требует задания целевой переменной, достаточно указать лишь таблицу, для которой требуется найти отличия.

Decision Tree (DT) - дерево решений

В системе PolyAnalyst реализован алгоритм, основанный на критерии максимизации взаимной информации (information gain). То есть для расщепления выбирается независимая переменная, несущая максимальную (в смысле Шеннона) информацию о зависимой переменной. Этот критерий имеет ясную интерпретацию и дает разумные результаты при самых разнообразных статистических параметрах изучаемых данных. Алгоритм DT является одним из самых быстрых в PolyAnalyst.

Decision Forest (DF) - леса решений

В случае, когда зависимая переменная может принимать большое количество разных значений, применение метода деревьев решений становится неэффективным. В такой ситуации в системе PolyAnalyst применяется метод, называемый лесом решений (decision forest). При этом строится совокупность деревьев решений - по одному для каждого различного значения зависимой переменной. Результатом прогноза, основанного на лесе решений, является то значение зависимой переменной, для которой соответствующее дерево дает наиболее вероятную оценку.

Алгоритмы ассоциации

Market Basket Analysis (BA) - метод анализа "корзины покупателя"

Название этого метода происходит от задачи определения вероятности, какие товары покупаются совместно. Однако реальная область его применения значительно шире. Например, продуктами можно считать страницы в Интернете, или те или иные характеристики клиента, или ответы респондентов в социологических и маркетинговых исследованиях и т.д. Алгоритм BA получает на вход бинарную матрицу, в которой строка - это одна корзина (кассовый чек, например), а столбцы заполнены логическими 0 и 1, обозначающими наличие или отсутствие данного признака (товара). На выходе формируются кластеры совместно встречаемых признаков с оценкой их вероятности и достоверности. Кроме этого, формируются ассоциативные направленные правила типа: если признак "A", то с такой-то вероятностью еще и признак "B" и еще признак "C". Алгоритм BA в PolyAnalyst работает исключительно быстро и способен обрабатывать огромные массивы данных.

Transactional Basket Analysis (TB) - транзакционный анализ "корзины"

Transactional Basket Analysis - это модификация алгоритма BA, применяемый для анализа очень больших данных, что не редкость для этого типа задач. Он предполагает, что каждая запись в базе данных соответствует одной транзакции, а не одной корзине (набору купленных за одну операцию товаров). На основе этого алгоритма компания "Мегапьютер" создала отдельный продукт - X-SellAnalyst, предназначенный для on-line рекомендации продуктов в Интернет-магазинах.

Модули текстового анализа

В системе PolyAnalyst реализована интеграция инструментов Data Mining с методами анализа текстов на естественном языке - алгоритмов Text Mining. Иллюстрация работы модулей текстового анализа показана на рис. 24.3.

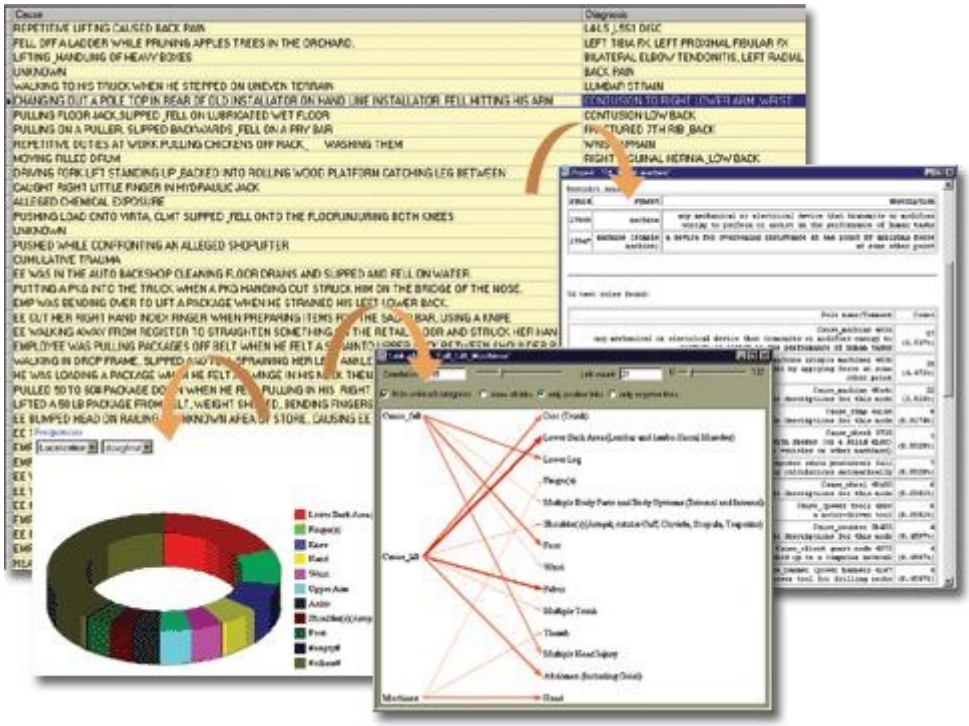

Рис. 24.3. Иллюстрация работы модулей текстового анализа

Text Analysis (TA) - текстовый анализ

Text Analysis представляет собой средство формализации неструктурированных текстовых полей в базах данных. При этом текстовое поле представляется как набор булевых признаков, основанных на наличии и/или частоте данного слова, устойчивого словосочетания или понятия (с учетом отношений синонимии и "общее-частное") в данном тексте. Тем самым появляется возможность распространить на текстовые поля всю мощь алгоритмов Data Mining, реализованных в системе PolyAnalyst. Кроме того, этот метод может быть использован для лучшего понимания текстовой компоненты данных за счет автоматического выделения наиболее распространенных ключевых понятий.

Text Categorizer (TC) - каталогизатор текстов

Этот модуль позволяет автоматически создать иерархический древовидный каталог имеющихся текстов и пометить каждый узел этой древовидной структуры наиболее индикативным для относящихся к нему текстов. Это нужно для понимания тематической структуры анализируемой совокупности текстовых полей и для эффективной навигации по ней.

Link Terms (LT) - связь понятий

Этот модуль позволяет выявлять связи между понятиями, встречающимися в текстовых полях изучаемой базы данных, и представлять их в виде графа. Граф также может быть использован для выделения записей, реализующих выбранную связь.

В PolyAnalyst встроены алгоритмы работы с текстовыми данными двух видов:

1. Алгоритмы, извлекающие ключевые понятия и работающие с ними.
2. Алгоритмы, сортирующие тексты на классы, которые определяются пользователем с помощью языка запросов.

Первый вид алгоритмов работает только с текстами на английском языке - при этом используется специальный словарь понятий английского языка. Алгоритмы второго типа могут работать с текстами и на английском, и на русском языках.

Text OLAP (матрицы измерений) и **Taxonomies** (таксономии) - это похожие друг на друга методы категоризации текстов. В Text OLAP пользователь создает именованные столбцы (измерения), состоящие из текстовых запросов. Например: "[добыча] и [нефть] и не ([руда] или [уголь] или [газ])". В процессе работы алгоритма PolyAnalyst применяет каждое из условий к каждому документу в базе данных и в случае удовлетворения условия относит этот документ к соответствующей категории. После работы модуля пользователь может выбирать различные элементы матрицы измерений и просматривать на экране тексты, удовлетворяющие выбранным условиям. Найденные слова будут в этих документах подкрашены разным цветом.

Работа с таксономиями очень похожа на работу с Text OLAP, только здесь пользователь строит иерархическую структуру из таких же условий, как и в матрицах измерений. Система пытается соотнести каждый документ с узлами этого дерева. После работы модуля пользователь также может перемещаться по узлам наполненной таксономии, просматривая отфильтрованные документы с подкрашенными словами.

Матрицы измерений и таксономии дают возможность пользователю взглянуть на коллекцию его документов под самыми разными углами. Но это не все: на основе этих объектов можно делать и другие, более сложные методы анализа, (например, анализ связей (Link Analysis), который показывает, насколько связаны друг с другом различные категории текстов, описанные пользователем) или включать тексты как независимые сущности в другие методы линейного и нелинейного анализа. Все это приводит к плотной интеграции подходов Data Mining и Text Mining в единую концепцию анализа информации.

Визуализация

В PolyAnalyst имеется богатый набор инструментов для графического представления и анализа данных и результатов исследований. Данные могут представляться в различных

зрительных форматах: гистограммах , двумерных, псевдо- и реальных трехмерных графиках.

Найденные в процессе Data Mining зависимости могут быть представлены как интерактивные графики со слайдерами для изменения значений представленных на них переменных. Эта особенность позволяет пользователю графически моделировать результаты. Имеется набор специальных графиков, широко применяемых в бизнесе, - это так называемые Lift, Gain charts, которые используются для графической оценки качества классификационных моделей и выбора оптимального числа контактов. Кроме этого, в последнюю версию программы включен новый визуальный метод Data Mining: анализ связей.

Link Analysis (LA) - анализ связей

Модуль Link Analysis позволяет выявлять корреляционные и антикорреляционные связи между значениями категориальных и булевых полей и представлять их в виде графа Этот граф также может быть использован для выделения записей, реализующих выбранную связь.

Symbolic Rule Language (SRL) - язык символьных правил

SRL - это универсальный алгоритмический язык PolyAnalyst, который используется для символьного представления автоматически найденных системой в процессе Data Mining правил, а также для создания пользователем своих собственных правил. На языке SRL можно выразить широкий спектр математических конструкций, используя алгебраические операции, большой набор встроенных функций, операции с датами и временем, логические и условные конструкции. Для удобства написания выражений на SRL в программе предусмотрен мастер создания правил.

Эволюционное программирование

В данное время эволюционное программирование является наиболее молодой и одной из многообещающих технологий Data Mining. Основная идея метода состоит в формировании гипотез о зависимости целевой переменной от других переменных в виде автоматически синтезируемых специальным модулем программ на внутреннем языке программирования.

Использование универсального языка программирования теоретически позволяет выразить любую зависимость, причем вид этой зависимости заранее не известен.

Процесс производства внутренних программ организуется как эволюция в пространстве программ, в некотором роде напоминающая генетические алгоритмы. Когда система находит перспективную гипотезу, описывающую исследуемую зависимость достаточно хорошо по целому ряду критериев, в работу включается механизм так называемых "обобщенных преобразований" (GT-search). С помощью этого механизма в "хорошую" программу вводятся незначительные модификации, не ухудшающие ее качество, и производится отбор лучшей дочерней программы. К новой популяции затем опять применяются механизмы синтеза новых программ, и этот процесс рекурсивно повторяется. Таким образом, система создает некоторое число генетических линий программ, конкурирующих друг с другом по точности, статистической значимости и простоте выражения зависимости.

Специальный модуль непрерывно преобразует "лучшую" на данный момент программу с внутреннего представления во внешний язык PolyAnalyst - язык символьных правил (Symbolic Rule Language), понятный человеку: математические формулы, условные конструкции и так далее. Это позволяет пользователю уяснить суть полученной зависимости, контролировать процесс поиска, а также получать графическую визуализацию результатов. Контроль статистической значимости полученных результатов осуществляется комплексом эффективных и современных статистических методов, включая методы рандомизированного тестирования.

Общесистемные характеристики PolyAnalyst

Типы данных

PolyAnalyst работает с разными типами данных. Это: числа, булевы переменные (yes/no), категориальные переменные, текстовые строки, даты, а также свободный английский текст.

Доступ к данным

PolyAnalyst может получать исходные данные из различных источников. Это: текстовые файлы с разделителем "запятая" (.csv), файлы Microsoft Excel 97/2000, любая ODBC-совместимая СУБД, SAS data files, Oracle Express, IBM Visual Warehouse.

Поддержка OLE DB for Data Mining

Версия 4.6 PolyAnalyst поддерживает спецификацию Microsoft OLE DB for Data Mining (Version 1.0). При выполнении исследований для большинства математических модулей (LR, FD, CL, FC, DT, DF, FL,PN, BA, TB) можно создавать так называемые "Mining Models" (MM). После завершения анализа эти модели можно применять к внешним данным через стандартные интерфейсы OLE DB или ADO из других программ или скриптов, поддерживающих создание ADO или COM-объектов. Применение модели осуществляется при помощи выполнения SQL-команд (Расширение SQL for DM). Mining Models можно также экспортировать в PMML. В планах развития программы намечается обеспечить интеграцию "PolyAnalyst DataMining Provider" с Microsoft Analysis Services(в составе SQL Server 2000).

In-place Data Mining

PolyAnalyst поддерживает запуск исследований на внешних данных через OLE DB интерфейсы без загрузки этих данных в проект PA. При выполнении исследования PolyAnalyst получает данные порциями через исполнение SQL-запросов к внешним источникам данных. Это позволяет преодолеть ограничения памяти при исследовании больших массивов данных. Данный процесс продемонстрирован на рис. 24.4.

SQL-Mode Data Mining Workflow Processes

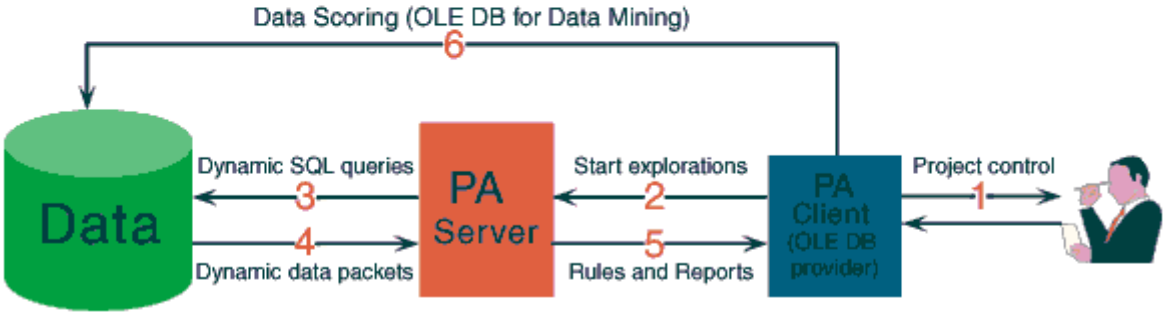

Рис. 24.4. In-place Data Mining

PolyAnalyst Scheduler - режим пакетной обработки

В PolyAnalyst предусмотрена возможность пакетного режима анализа данных. Для этого имеется специальный скриптовый язык, на котором программируется все аналитические действия и временная последовательность их выполнения, а также определяются наборы данных. Скрипт сохраняется в файле и автоматически инициализирует исследование в указанный момент времени на определенных данных. Для реализации функции Scheduler в электронной лицензии должна быть включена соответствующая опция.

В таблице 24.1 описано семейство продуктов PolyAnalyst6: продукты и соответствующие конфигурации системы.

Таблица 24.1. Семейство продуктов PolyAnalyst

Продукт	Конфигурация системы
Локальные продукты	
PolyAnalyst 4.6, однопользовательская версия	Математические модули: FL, FD, PN, FC, BA, TB, MB, CL, DS, DT, DF, LR, LA, TA, TC, LT, SS. Пакетная обработка, поддержка OLE DB. Платформа - MS Windows NT/2000/XP
PolyAnalyst 3.5 Professional (русс.)	Математические модули: FL, FD, PN, FC, CL, DS, LR, SS. Платформа - MS Windows NT/2000/XP
PolyAnalyst 3.5 Power (русс.)	Математические модули: FD, PN, FC, CL, DS, LR, SS. Платформа - MS Windows 98/NT/2000/XP
PolyAnalyst 3.5 Lite - студенческая версия (русс.)	Математические модули: FD, FC, CL, DS, LR, SS. Платформа - MS Windows 98/NT/2000/XP
Сетевые продукты	

PolyAnalyst Knowledge Server 4.6, сетевая версия	Математические модули: FL, FD, PN, FC, BA, TB, MB, CL, DS, DT, DF, LR, LA, TA, TC, LT, SS. Пакетная обработка, поддержка OLE DB, In-Place Data Mining. Серверная часть - MS Windows NT/2000/XP server, клиентская часть - MS Windows 98/NT/2000/XP. Клиент/серверная версия системы
Средства разработки	
PolyAnalyst COM - SDK для создания собственных приложений для Data Mining	Набор COM-объектов, библиотеки, документация для разработчиков

WebAnalyst

Помимо разработок PolyAnalyst и TextAnalyst, предназначенных соответственно для добычи данных и текстов (Data Mining и Text Mining), фирма Мегапьютер реализует третий продукт - WebAnalyst.

WebAnalyst - это корпоративный аналитический сервер, представляющий собой интегрированную платформу для хранения и обработки информации и адаптированный для работы с web-данными и для решения задач e-business.

WebAnalyst является масштабируемым сервером приложений с открытой архитектурой, который автоматизирует задачи сбора информации, ее преобразования, анализа и генерации персонализированного контента для потребителей. Кроме этого, клиентское приложение WebAnalyst предоставляет гибкий инструмент для визуального проектирования.

- Обрабатывает данные из различных источников, таких как каналы передачи данных (HTTP), внешние базы данных и лог-файлы web-серверов.
- Хранит связанную информацию в собственной единой универсальной базе данных.
- Содержит набор встроенных аналитических инструментов и инструментов для работы с данными (модули WebAnalyst), предоставляет пользователю визуальное средство для разработки процедур обработки и анализа данных и для генерации контента.

WebAnalyst уже включает в себя все математические модули для Data и Text Mining систем PolyAnalyst и TextAnalyst, а также специальную аналитическую математику.

WebAnalyst может быть полезен при решении следующих задач [106]:

- регистрации взаимодействия посетителя с Web-сайтом;
- преобразовании и хранении аналитической информации;
- использовании собранных данных для изучения интересов посетителя и его предпочтений;
- анализе эффективности ресурсов сайта и его архитектуры;
- составлении отчетов для руководства;
- использовании полученной информации для персонифицированного диалога с каждым посетителем.

В качестве "сырья" для своей работы WebAnalyst может использовать: информационные потоки от Web-серверов; базы данных информационного наполнения; базы данных клиентов, продуктов и транзакций; накопленные регистрационные файлы Web-серверов; другие внешние источники данных.

Инструменты Data Mining. Программные продукты Cognos и система STATISTICA Data Miner

Программные продукты Cognos (разработчик - компания Cognos [107]) - это инструменты интеллектуального или делового анализа данных (от англ. Business Intelligence Tools), или BI-инструменты. Представление о комплексе программных средств компании Cognos дает следующий рис. 25.1 [108].

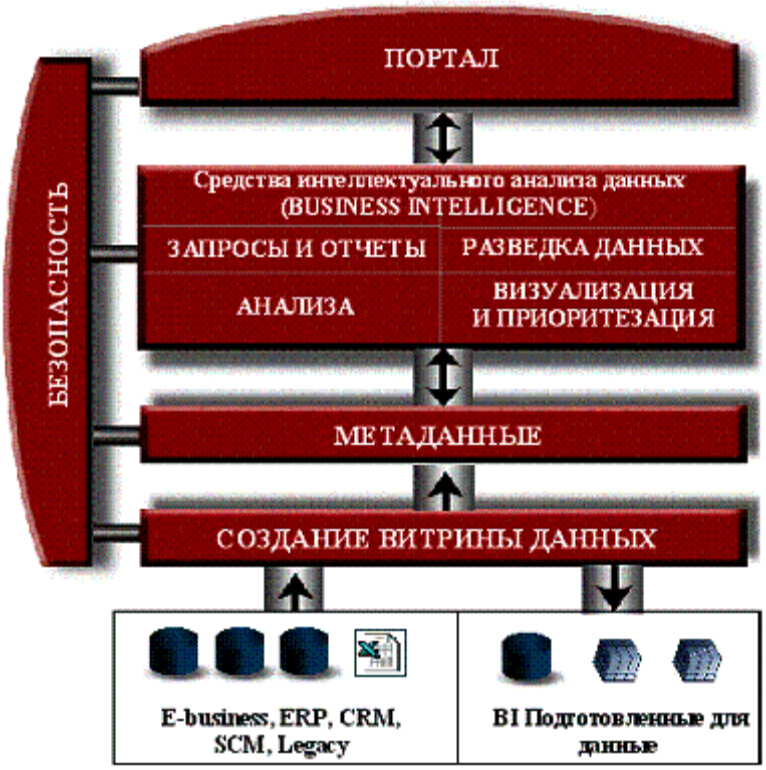

Рис. 25.1. Комплекс программных средств компании Cognos

Ниже перечислены основные программные продукты Cognos, которые относятся к проблемным областям, указанным на рисунке.

1. Работа с запросами и отчетами. Решения в области работы с отчетами ориентированы на различные типы пользователей. Продукты отличаются требованиями к уровню сложности отчетов и уровню навыков конечных пользователей:
    - Decision Stream - средство для создания витрин данных (data marts), оптимизированных на формирование запросов и построение отчетов;
    - Impromptu - средство для работы с запросами, а также со статическими и настраиваемыми отчетами;
    - PowerPlay - как средство построения многомерных отчетов;
    - Impromptu Web Reports - средства для работы со статическими отчетами через Web;
    - Cognos Query - средство для создания запросов, навигации и исследования данных в т.ч. через Web;

    - Visualizer - средство для работы с мощными визуальными отчетами.
2. Анализ данных. Средства анализа данных предназначены для анализа критической информации и выявления значимых факторов. Этот процесс охватывает полный набор аналитических задач и задач по построению отчетов, включая работу с отчетами бизнес-уровня, возможность перехода к данным нижнего уровня, создание и просмотр представлений с целью выявления приоритетов. Интеграция средств позволяет удобно переходить от исследования и анализа данных при помощи отчетов бизнес-уровня к исследованию и анализу данных по отчетам нижнего уровня (функция drill through):
    - PowerPlay - средство многомерного (OLAP) анализа и построения бизнес-отчетов;
    - Impromptu - средство для просмотра отчетов с детальной информацией нижнего уровня (для Windows);
    - Impromptu Web Reports - средство для просмотра отчетов с детальной информацией нижнего уровня (для Web);
    - Visualizer - средство визуального представления данных.
3. Визуализация и выявление приоритетов. К разделу визуализации информации и выявлению приоритетов можно отнести целый спектр продуктов. С их помощью пользователю становится доступна визуализированная информация, представленная в удобном виде для выявления критических факторов на больших массивах данных. В этих продуктах за основу принимается возможность анализа ключевых факторов, влияющих на рассматриваемую область знаний (бизнеса) при помощи широких возможностей по визуализации данных. Правильно выявленные приоритеты являются основой для принятия эффективных решений:
    - Visualizer - средство для представления информации в форме визуальных представлений с использованием визуальных элементов для выявления приоритетов;
    - PowerPlay как средство многомерного представления информации;
    - Impromptu как средство для работы с настраиваемыми отчетами;
    - Cognos Query - средство Web-пользователей для построения запросов.
4. Разведка данных (data mining). Средства разведки и добывания данных предлагают целый ряд возможностей по автоматизированному просмотру данных, позволяя вскрывать скрытые тенденции, выявлять приоритетные решения и действия путем отображения тех факторов, которые более других влияют на исследуемые показатели:
    - Scenario - средство сегментации и классификации;
    - 4Thought - средство прогнозирования;
    - Visualazer как средство визуализации.
5. Защита информации. Защита информации достигается за счет использования единого для всех приложений компонента, называемого Access Manager и позволяющего описывать классы пользователей и управлять ими для всех типов аналитических приложений Cognos. В дополнение к Access Manager, могут быть использованы также обычные возможности обеспечения безопасности на уровне базы данных и операционной системы. На практике возможно одновременное использование всех трех уровней защиты информации;
6. Описание метаданных. В качестве средства описания метаданных может быть использован единый для всех Cognos BI продуктов компонент, называемый Cognos Architect. Достоинство использования единого для всех средств модуля заключается в возможности единообразного представления бизнес-информации. Единожды сформулированные метаданные становятся доступными в любом аналитическом приложении Cognos.

Особенности методологии моделирования с применением Cognos 4Thought

Инструментальное средство Cognos 4Thought (рис. 25.2) входит в состав семейства современных программных средств обработки, анализа и прогнозирования данных, разработанного компанией Cognos.

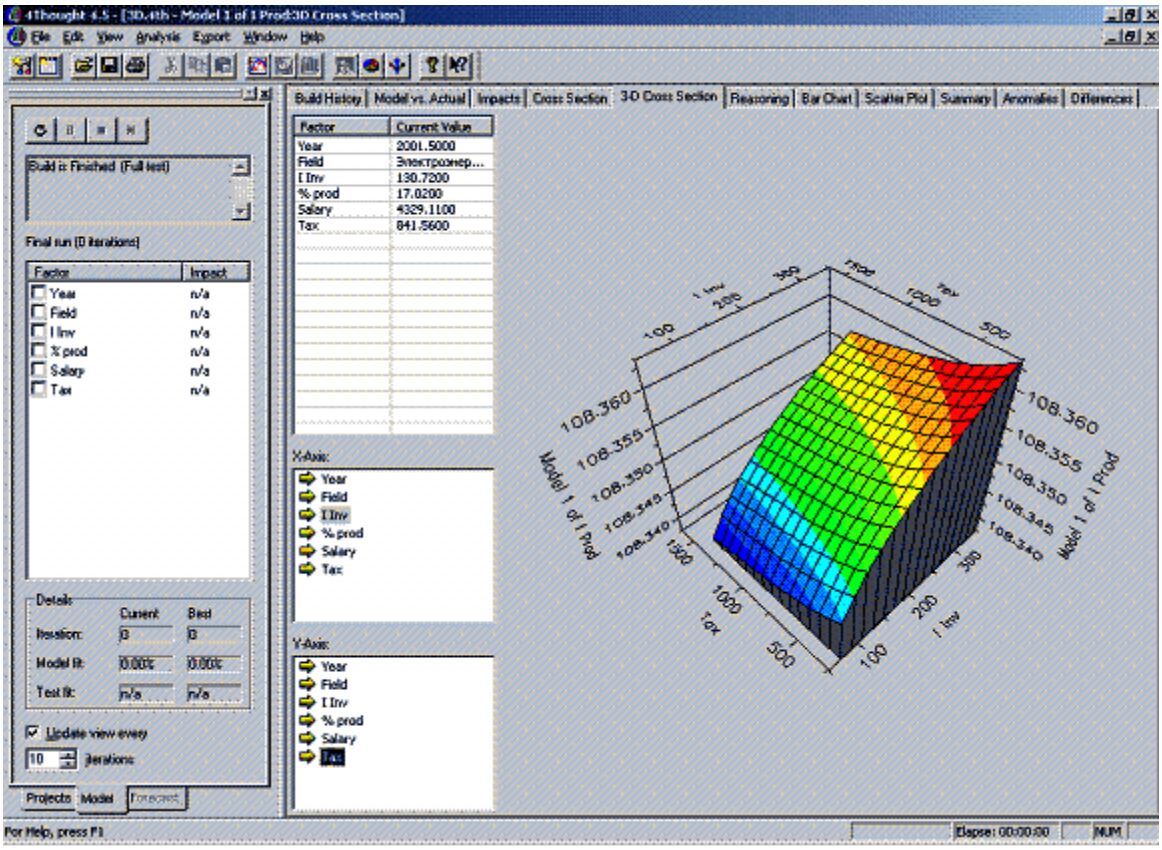

Рис. 25.2. Интерфейс инструментального средства Cognos 4Thought

В основу программного продукта Cognos 4Thought положена технология нейронных сетей. Использование нейронных сетей позволяет строить достаточно точные сложные нелинейные модели на основе неполной статистической выборки данных.

Cognos 4Thought предназначен для моделирования и прогнозирования. 4Thought может анализировать исторические данные во времени, затем продолжить эту временную линию в будущее, предсказывая тенденции.

На рис. 25.3 представлена типичная схема взаимодействия Cognos 4Thought с другими продуктами семейства, выполняющими подготовку данных для 4Thought.

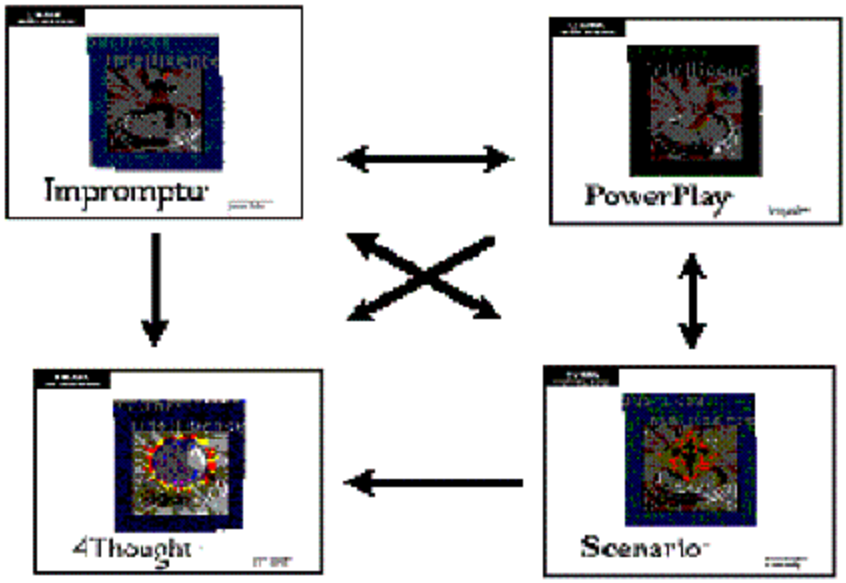

Рис. 25.3. Взаимодействие систем Impromptu, PowerPlay, Scenario и 4Thought

Системы Impromptu, PowerPlay, Scenario и 4Thought представляют собой взаимосвязанные и дополняющие друг друга инструментальные средства, поддерживающие наиболее эффективные технологии обработки данных и обеспечивающие решение широкого круга задач в бизнес-приложениях, от доступа к информации в распределенных базах данных до вычислительной обработки и интеллектуального анализа.

Cognos PowerPlay - это инструментальное средство для оперативного анализа данных и формирования отчетов по OLAP-технологии. Оно позволяет аналитикам исследовать данные под любым углом зрения, обеспечивая реальное многоуровневое видение текущего состояния организации. Главная особенность инструмента заключается в исключительной автоматизации процесса создания аналитического приложения, что позволяет за очень короткий срок создавать полномасштабные аналитические приложения, в основу которых положена технология OLAP.

Кроме того, инструмент отличается удобством применения: от пользователя требуются лишь навыки работы в среде Windows.

PowerPlay обеспечивает эффективный доступ ко всей имеющейся в организации информации, хранящейся в форме реляционных или не реляционных данных, таких как базы данных (Databases), склады данных (Data Warehouses), витрины данных (Data Marts) и электронные таблицы (Spreadsheets).

Созданный с помощью PowerPlay гиперкуб можно открыть в 4Thought. Гиперкуб представляет собой файл многомерных данных с расширением .mdc. Данные в таком файле организованы специальным образом для обеспечения быстрого доступа и детализации.

OLAP-кубы Cognos можно использовать как источники данных для модулей Data Mining (4Thought и Scenario), таким образом в продукции Cognos реализована интеграция технологий OLAP и Data Mining.

Cognos Impromptu - это инструмент фирмы Cognos для построения запросов любой сложности и отчетов произвольного формата пользователями, от которых не требуется навыков программирования. Отличительная черта этого средства - простота в использовании, которая достигается благодаря продуманному и интуитивно понятному интерфейсу.

Impromptu обеспечивает пользователей оперативной и детальной информацией, необходимой для принятия решений. Одним из основных достоинств Impromptu является возможность быстрого построения широкого спектра различных отчетов в зависимости от того, какие данные необходимы для принятия решения. Это означает, что пользователи могут формировать отчеты любой нужной структуры гораздо оперативнее и проще, чем при использовании других построителей отчетов.

Отчеты Impromptu также могут быть использованы в качестве входных данных для построения модели в Cognos 4Thought.

Cognos Scenario - это интеллектуальное инструментальное средство поиска (разведки) данных (Data Mining), которое позволяет руководителям (даже не знакомым с методиками статистического анализа) выявлять скрытые тенденции и модели бизнеса и "извлекать на поверхность" его ранее неизвестные закономерности и корреляционные связи.

Система Scenario спроектирована для построения моделей, описывающих особенности бизнеса по данным, которые при использовании традиционных методов анализа могли бы быть незамеченными. Удобный интерфейс этого приложения позволяет пользователям легко визуализировать имеющиеся сведения о бизнесе. Он автоматизирует обнаружение и ранжирование наиболее важных факторов, влияющих на бизнес, и выявление скрытых связей между этими факторами. Обладая подобным интерфейсом, Scenario делает процесс анализа данных, традиционно трудоемкий и дорогостоящий, простым и оперативным.

Результаты работы Scenario (ключевые показатели и факторы) могут быть переданы в 4Thought для выполнения прогнозирования.

Cognos 4Thought использует технологии математического моделирования, которые позволяют изучить взаимную связь факторов, влияющих на выбранную сферу деятельности. Это программное средство дает возможность плановикам создавать точную модель бизнеса, используемую для сравнения, прогнозирования, интерпретации результатов измерений.

4Thought поддерживает анализ на всех этапах:

1. **Сбор данных**. Данные вводятся непосредственно или получаются из внешних источников, например, MS Excel. Данные могут быть взяты у других программных средств семейства Cognos (Impromptu, ReportNet, PowerPlay и Scenario) или прямо из хранилища. Введенные данные отображаются в 4Thought в виде электронных таблиц, что позволяет достаточно просто их просматривать и анализировать;
2. **Преобразование данных**. Прежде чем попасть в модуль 4Thought, данные обычно очищаются в модуле Impromptu, который делает запросы к источникам данных (реляционным базам данных), позволяет накладывать фильтры на выборки данных (например, исключать строки, в которых значение показателя - целевой функции равно нулю, либо превращать одинаковые строки в одну строку, либо отсеивать строки если значение показателя является аномальным - выходит за пределы двух

среднеквадратичных отклонений вверх и трех вниз, и т.п., правила очистки данных можно произвольно настраивать). Отчеты Impromptu могут быть использованы в качестве входных данных для построения модели в 4Thought.

В модуле 4Thought также есть возможность просматривать данные и исключать аномалии (задавая допустимые интервалы, в которых может изменяться значение показателя), а также заменить пустые значения показателей на конкретные значения. При этом создаются новые поля: коэффициенты, пропорции, процентные соотношения, дающие более полную картину проблемы.

3. **Исследование данных**. Данные визуализируются для просмотра в виде электронных таблиц, графиков и диаграмм различного вида. Фактически, этот этап представляет собой предварительный просмотр данных перед построением модели в 4Thought (выявление аномалий, работа с дубликатами и пропусками).
4. **Создание модели**. 4Thought создает модель автоматически, но позволяет детальную интерактивную настройку параметров модели; пользователь контролирует ряд параметров, включая выбор факторов (например исключение несущественных факторов), отсеивание аномальных значений и т.д.
5. **Интерпретация**. После загрузки данных в модель 4Thought создает ряд отчетов и дает возможность работы с разнообразными графиками. Таким образом модель просматривается, проверяется достоверность полученных результатов, выявляются взаимозависимости факторов.
6. **Применение**. Реализованная модель используется для прогнозирования и определения наиболее существенных факторов, задающих изменения ключевых показателей.

4Thought позволяет выполнить обучение модели на репрезентативной выборке значений входных и выходных параметров нейронной сети. Для обучения может быть использована вся выборка либо ее часть - в таком случае оставшаяся часть выборки применяется для контроля точности (качества) обучения: отклонения значений выходов обученной нейронной сети от реальных значений. Обучение сети на одном наборе данных выполняется несколько раз (перед каждым обучением начальные значения весовых коэффициентов устанавливаются автоматически случайным образом), чтобы выбрать наилучшую точность обученной сети.

Cognos 4Thought позволяет, варьируя параметры сценарных условий, автоматически получать различные прогнозы на заданный период, отвечая на вопрос: "А что будет, если?" Результаты прогнозирования по всем отраслям региональной экономики можно получать в виде текстов, графиков, диаграмм, а также отчетных документов установленного образца, которые можно хранить в электронном виде или передавать потребителям по электронной почте. Такие возможности освобождают аналитиков от рутинной вычислительной и оформительской работы и позволяют сосредоточиться на вопросах стратегии и тактики регионального развития.

Cognos 4Thought отображает степень влияния факторов (входных переменных) на целевую переменную, что позволяет использовать его в качестве инструмента факторного анализа. То есть после настройки сети можно оценить, какие факторы вносят какой вклад в конечный результат.

4Thought может оперировать с временными рядами. Это позволяет обнаруживать и анализировать тренды в динамике экономических величин, а также строить прогноз значений показателей на несколько лет вперед. 4Thought поддерживает несколько

способов нормирования входных и выходных параметров, что дает возможность оперировать с экономическими величинами, влияние которых нелинейно.

При комплексном использовании продуктов семейства Cognos (рис. 25.3) в единой информационно-аналитической системе возникают дополнительные преимущества (синергетический эффект). Задачи по сбору и обработке информации в системе решаются на этапе формирования витрин данных с помощью инструмента PowerPlay Transformation Server.

Вопросы безопасности в системе (защиты от несанкционированного доступа) решаются с помощью инструмента Access Manager, входящего в состав пакета PowerPlay Transformation Server.

Инструменты PowerPlay и Impromptu используются для решения задач, связанных с мониторингом показателей, многомерным анализом информации, формированием отчетов, а инструменты 4Thought и Scenario - для прогнозирования показателей социально-экономического развития, а также для факторного анализа данных. Организация передачи данных между инструментами полностью автоматизирована. Простота интерфейса продуктов Cognos и ориентированность на пользователей-непрограммистов позволяет эффективно выполнять сложные задачи анализа. Публикация информации в интранет/экстранет-среде может осуществляться с помощью инструмента Upfront, входящего в состав пакета Cognos PowerPlay Enterprise Server.

Система STATISTICA Data Miner

Назначение. Система STATISTICA Data Miner (разработчик - компания StatSoft [109]) спроектирована и реализована как универсальное и всестороннее средство анализа данных - от взаимодействия с различными базами данных до создания готовых отчетов, реализующее так называемый графически-ориентированный подход [110, 111].

Система STATISTICA предлагает:

- Большой набор готовых решений;
- Удобный пользовательский интерфейс, полностью интегрированный с MS Office;
- Мощные средства разведочного анализа;
- Полностью оптимизированный пакет для работы с огромным объемом информации;
- Гибкий механизм управления;
- Многозадачность системы;
- Чрезвычайно быстрое и эффективное развертывание;
- Открытая COM-архитектура, неограниченные возможности автоматизации и поддержки пользовательских приложений (использование промышленного стандарта Visual Basic (является встроенным языком), Java, C/C++).

Сердцем STATISTICA Data Miner является браузер процедур Data Mining (рис. 25.4), который содержит более 300 основных процедур, специально оптимизированных под задачи Data Mining, средства логической связи между ними и управления потоками данных, что позволит Вам конструировать собственные аналитические методы.

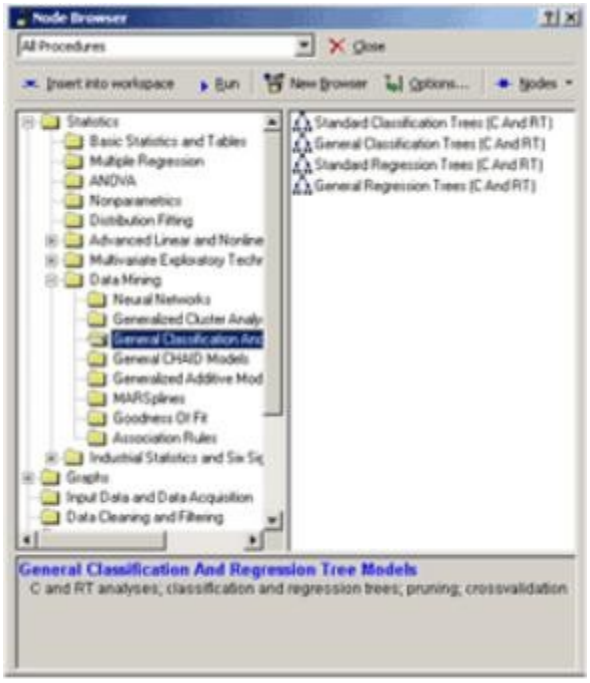

Рис. 25.4. Браузер процедур Data Mining

Рабочее пространство STATISTICA Data Miner состоит из четырех основных частей (рис. 25.5):

Рис. 25.5. Рабочее пространство STATISTICA Data Miner

1. Data Acquisition - сбор данных. В данной части пользователь идентифицирует источник данных для анализа, будь то файл данных или запрос из базы данных.
2. Data Preparation, Cleaning, Transformation - подготовка, преобразования и очистка данных. Здесь данные преобразуются, фильтруются, группируются и т.д.
3. Data Analysis, Modeling, Classification, Forecasting - анализ данных, моделирование, классификация, прогнозирование. Здесь пользователь может при помощи браузера или готовых моделей задать необходимые виды анализа данных, таких как прогнозирование, классификация, моделирование и т.д.

4. Reports - результаты. В данной части пользователь может просмотреть, задать вид и настроить результаты анализа (например, рабочая книга, отчет или электронная таблица).

Средства анализа STATISTICA Data Miner

Средства анализа STATISTICA Data Miner можно разделить на пять основных классов:

1. General Slicer/Dicer and Drill-Down Explorer - разметка/разбиение и углубленный анализ. Набор процедур, позволяющий разбивать, группировать переменные, вычислять описательные статистики, строить исследовательские графики и т.д.
2. General Classifier - классификация. STATISTICA Data Miner включает в себя полный пакет процедур классификации: обобщенные линейные модели, деревья классификации, регрессионные деревья, кластерный анализ и т.д.
3. General Modeler/Multivariate Explorer - обобщенные линейные, нелинейные и регрессионные модели. Данный элемент содержит линейные, нелинейные, обобщенные регрессионные модели и элементы анализа деревьев классификации.
4. General Forecaster - прогнозирование. Включает в себя модели АРПСС, сезонные модели АРПСС, экспоненциальное сглаживание, спектральный анализ Фурье, сезонная декомпозиция, прогнозирование при помощи нейронных сетей и т.д.
5. General Neural Networks Explorer - нейросетевой анализ. В данной части содержится наиболее полный пакет процедур нейросетевого анализа.

Приведенные выше элементы являются комбинацией модулей других продуктов StatSoft. Кроме них, STATISTICA Data Miner содержит набор специализированных процедур Data Mining, которые дополняют линейку инструментов Data Mining:

- Feature Selection and Variable Filtering (for very large data sets) - специальная выборка и фильтрация данных (для больших объемов данных). Данный модуль автоматически выбирает подмножества переменных из заданного файла данных для последующего анализа. Например, модуль может обработать около миллиона входных переменных с целью определения предикторов для регрессии или классификации.
- Association Rules - правила ассоциации. Модуль является реализацией так называемого априорного алгоритма обнаружения правил ассоциации. Например, результат работы этого алгоритма мог бы быть следующим: клиент после покупки продукт "A", в 95 случаях из 100 в течение следующих двух недель после этого заказывает продукт "B" или "C".
- Interactive Drill-Down Explorer - интерактивный углубленный анализ. Представляет собой набор средств для гибкого исследования больших наборов данных. На первом шаге вы задаете набор переменных для углубленного анализа данных, на каждом последующем шаге выбираете необходимую подгруппу данных для последующего анализа.
- Generalized EM & k-Means Cluster Analysis - обобщенный метод максимума среднего и кластеризация методом К средних. Данный модуль - это расширение методов кластерного анализа. Он предназначен для обработки больших наборов данных и позволяет кластеризовывать как непрерывные, так и категориальные переменные, обеспечивает все необходимые функциональные возможности для распознавания образов.
- Generalized Additive Models (GAM) - обобщенные аддитивные модели (GAM). Набор методов, разработанных и популяризованных Hastie и Tibshirani.
- General Classification and Regression Trees (GTrees) - обобщенные классификационные и регрессионные деревья (GTrees). Модуль является полной реализацией методов, разработанных Breiman, Friedman, Olshen и Stone (1984). Кроме этого, модуль содержит разного рода доработки и дополнения, такие как оптимизации алгоритмов для больших объемов данных и т.д. Модуль является набором методов обобщенной классификации и регрессионных деревьев.

- General CHAID (Chi-square Automatic Interaction Detection) Models - обобщенные CHAID-модели (Хи-квадрат автоматическое обнаружение взаимодействия). Подобно предыдущему элементу, этот модуль является оптимизацией данной математической модели для больших объемов данных.
- Interactive Classification and Regression Trees - интерактивная классификация и регрессионные деревья. В дополнение к модулям автоматического построения разного рода деревьев, STATISTICA Data Miner также включает средства для формирования таких деревьев в интерактивном режиме.
- Boosted Trees - расширяемые простые деревья. Последние исследования аналитических алгоритмов показывают, что для некоторых задач построения "сложных" оценок, прогнозов и классификаций использование последовательно увеличиваемых простых деревьев дает более точные результаты, чем нейронные сети или сложные цельные деревья. Данный модуль реализует алгоритм построения простых увеличиваемых (расширяемых) деревьев.
- Multivariate Adaptive Regression Splines (Mar Splines) - многомерные адаптивные регрессионные сплайны (Mar Splines). Данный модуль основан на реализации методики предложенной Friedman (1991; Multivariate Adaptive Regression Splines, Annals of Statistics, 19, 1-141); в STATISTICA Data Miner расширены опции MARSPLINES для того, чтобы приспособить задачи регрессии и классификации к непрерывным и категориальным предикторам.

Модуль MAP-сплайны предназначен для обработки как категориальных, так и непрерывных переменных вне зависимости от того, являются ли они предикторами или переменными отклика. В случае категориальных переменных отклика, модуль MAP-сплайны рассматривает текущую задачу как задачу классификации. Напротив, если зависимые переменные непрерывны, то задача расценивается как регрессионная. Модуль MAP-сплайны автоматически определяет тип задачи.

MAP-сплайны - непараметрическая процедура, в работе которой не используется никаких предположений об общем виде функциональных связей между зависимыми и независимыми переменными. Процедура устанавливает зависимости по набору коэффициентов и базисных функций, которые полностью определяются из исходных данных. В некотором смысле, метод основан на принципе "разделяй и властвуй", в соответствии с которым пространство значений входных переменных разбивается на области со своими собственными уравнениями регрессии или классификации. Это делает использование MAP-сплайнов особенно эффективным для задач с пространствами значений входных переменных высокой размерности.

Метод MAP-сплайнов нашел особенно много применений в области добычи данных по причине того, что он не опирается на предположения о типе и не накладывает ограничений на класс зависимостей (например, линейных, логистических и т.п.) между предикторными и зависимыми (выходными) переменными. Таким образом, метод позволяет получить содержательные модели (т.е. модели, дающие весьма точные предсказания) даже в тех случаях, когда связи между предикторными и зависимыми переменными имеют немонотонный характер и сложны для приближения параметрическими моделями.

- Goodness of Fit Computations - критерии согласия. Данный модуль производит вычисления различных статистических критериев согласия как для непрерывных переменных, так и для категориальных.
- Rapid Deployment of Predictive Models - быстрые прогнозирующие модели (для большого числа наблюдаемых значений). Модуль позволяет строить за короткое время

классификационные и прогнозирующие модели для большого объема данных. Полученные результаты могут быть непосредственно сохранены во внешней базе данных.

Несложно заметить, что система STATISTICA включает огромный набор различных аналитических процедур, и это делает его недоступным для обычных пользователей, которые слабо разбираются в методах анализа данных. Компанией StatSoft предложен вариант работы для обычных пользователей, обладающих небольшими опытом и знаниями в анализе данных и математической статистике.

Для этого, кроме общих методов анализа, были встроены готовые законченные (сконструированные) модули анализа данных, предназначенные для решения наиболее важных и популярных задач: прогнозирования, классификации, создания правил ассоциации и т.д.

Далее кратко описана схема работы в Data Miner.

Шаг 1. Работу в Data Miner начнем с подменю "Добыча данных" в меню "Анализ" (рис. 25.6). Выбрав пункт "Добытчик данных - Мои процедуры" или "Добытчик данных - Все процедуры", мы запустим рабочую среду STATISTICA Data Mining.

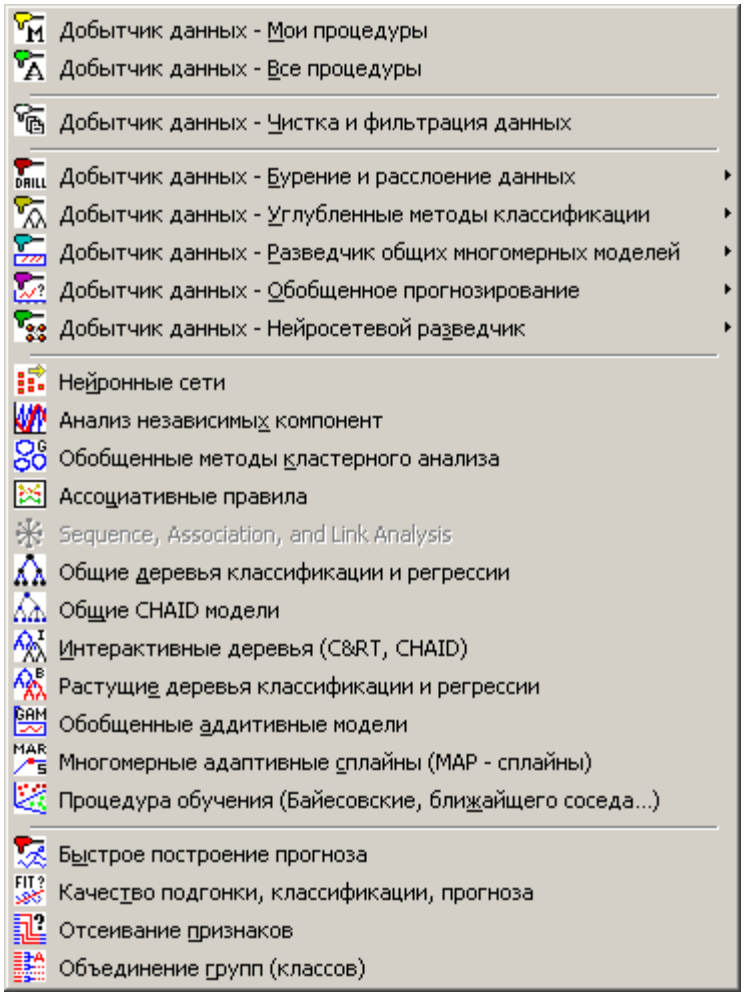

Рис. 25.6. Пункт "Добытчик данных"

Шаг 2. Для примера возьмем файл Boston2.sta из папки примеров STATISTICA. В следующем примере анализируются данные о жилищном строительстве в Бостоне. Цена участка под застройку классифицируется как Низкая - Low, Средняя - Medium или Высокая - High в зависимости от значения зависимой переменной Price. Имеется один категориальный предиктор - Cat1 и 12 порядковых предикторов - Ord1-Ord12. Весь набор данных, состоящий из 1012 наблюдений, содержится в файле примеров Boston2.sta. Выбор таблицы показан на рис. 25.7.

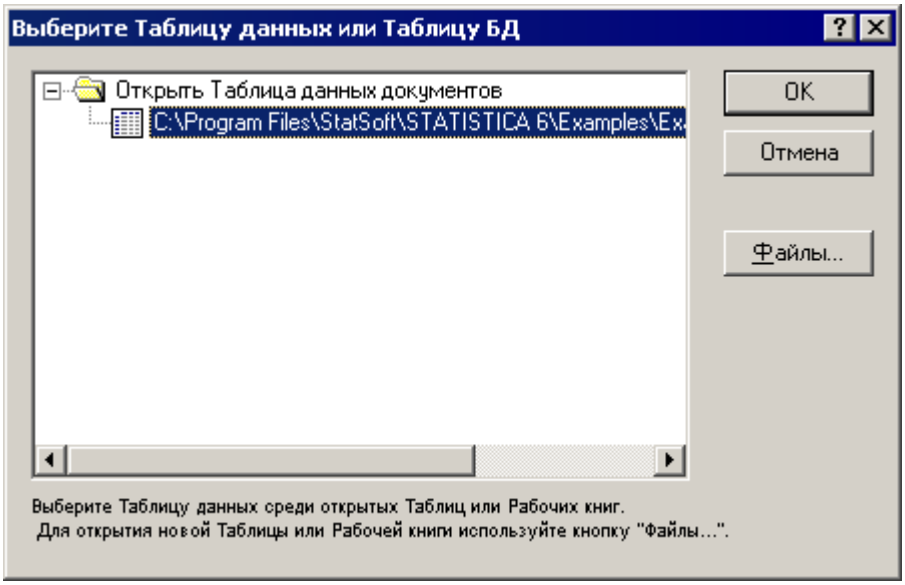

Рис. 25.7. Выбор таблицы для анализа

Шаг 3. После выбора файла появится окно диалога "Выберите зависимые переменные и предикторы", показанное на рис. 25.8.

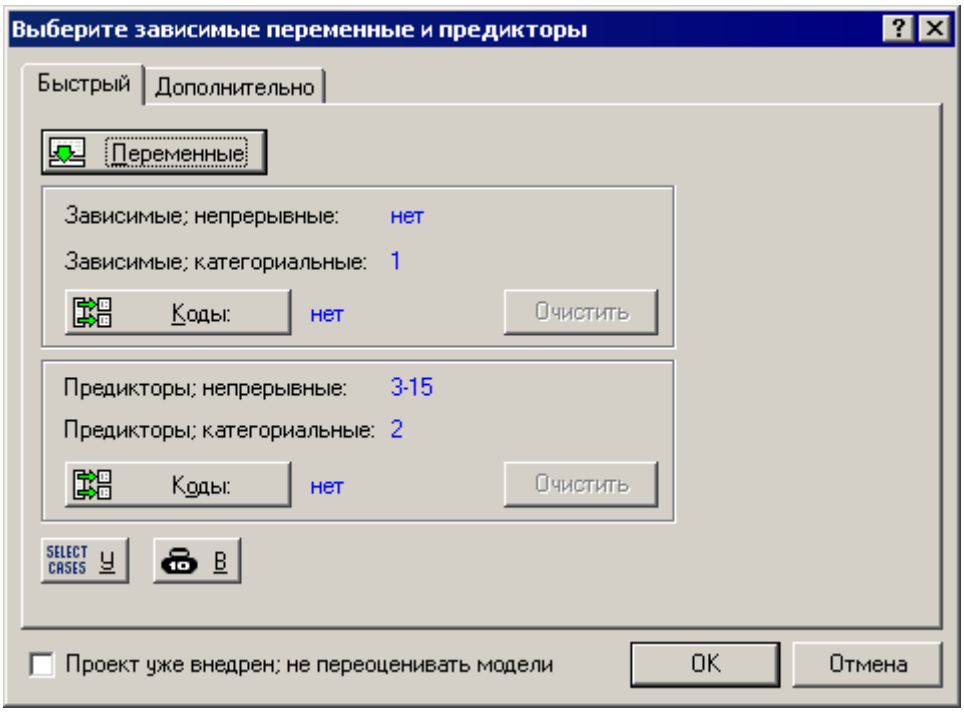

Рис. 25.8. Выбор зависимых переменных и предикторов

Выбираем зависимые переменные (непрерывные и категориальные) и предикторы (непрерывные и категориальные), исходя из знаний о структуре данных, описанной выше. Нажимаем OK.

Шаг 4. Запускаем "Диспетчер узлов" (нажимаем на кнопку

в окне Data Miner). В данном диалоге, показанном на рис. 25.9, мы можем выбрать вид анализа или задать операцию преобразования данных.

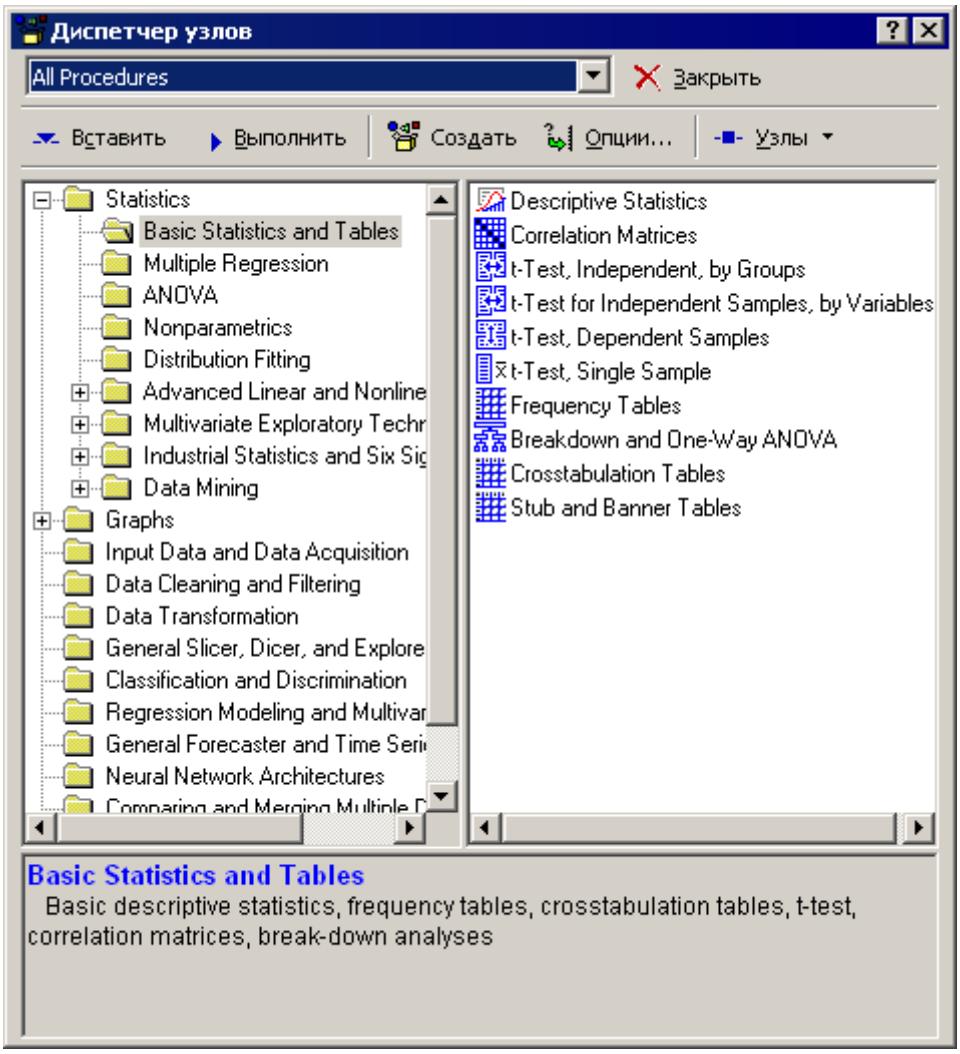

Рис. 25.9. "Диспетчер узлов"

Диспетчер узлов включает в себя все доступные процедуры для добычи данных. Всего доступно около 260 методов фильтрации и очистки данных, методов анализа. По

умолчанию, процедуры помещены в папки и отсортированы в соответствии с типом анализа, который они выполняют. Однако пользователь имеет возможность создать собственную конфигурацию сортировки методов.

Для того чтобы выбрать необходимый анализ, необходимо выделить его на правой панели и нажать кнопку "вставить". В нижней части диалога дается описание выбираемых методов.

Выберем, для примера, Descriptive Statistics и Standard Classification Trees with Deployment (C And RT) . Окно Data Miner выглядит следующим образом.

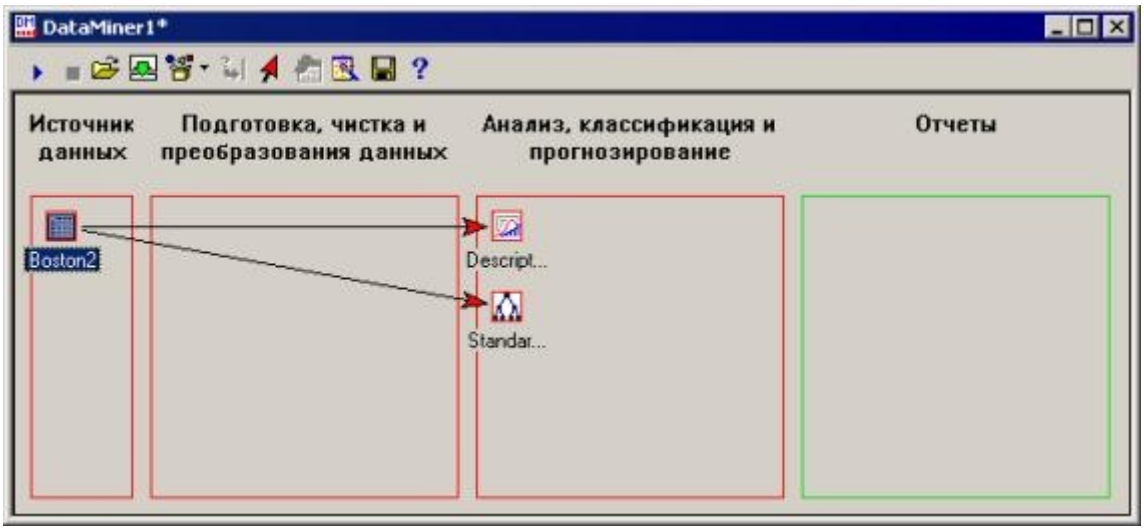

Рис. 25.10. Окно Data Miner с узлами выбранных анализов

Источник данных в рабочей области Data Miner автоматически будет соединен с узлами выбранных анализов. Операции создания/удаления связей можно производить и вручную.

Шаг 5. Теперь выполним проект. Все узлы, соединенные с источниками данных активными стрелками, будут проведены.

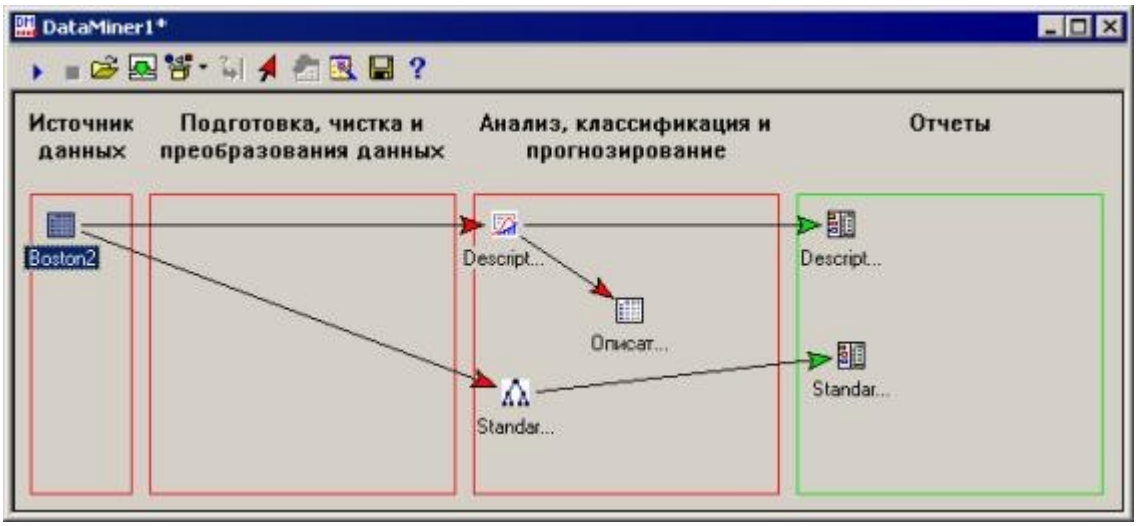

Рис. 25.11. Окно Data Miner после выполнения проекта

Далее можно просмотреть результаты (в столбце отчетов). Подробные отчеты создаются по умолчанию для каждого вида анализа. Для рабочих книг результатов доступна полная функциональность системы STATISTICA.

Шаг 6. На следующем шаге просматриваем результаты, редактируем параметры анализа.

Кроме того, в диспетчере узлов STATISTICA Data Miner содержатся разнообразные процедуры для классификации и Дискриминантного анализа, Регрессионных моделей и Многомерного анализа, а также Обобщенные временные ряды и прогнозирование. Все эти инструменты можно использовать для проведения сложного анализа в автоматическом режиме, а также для оценивания качества модели.

Инструменты Oracle Data Mining и Deductor

Oracle Data Mining

В марте 1998 компания Oracle [112] объявила о совместной деятельности с 7 партнерами - поставщиками инструментов Data Mining. Далее последовало включение в Oracle8i средств поддержки алгоритмов Data mining. В июне 1999 года Oracle приобретает Darwin (Thinking Machines Corp.). В 2000-2001 годах выходят новые версии Darwin, Oracle Data Mining Suite. В июне 2001 года выходит Oracle9i Data Mining.

Oracle Data Mining является опцией или модулем в Oracle Enterprise Edition (версия Oracle Database 10g). Опция Oracle Data Mining (ODM) предназначена для анализа данных методами, относящимися к технологии извлечения знаний, или Data Mining. В редакциях Personal Edition, Standard Edition, OneStandard Edition эта опция недоступна.

ODM поддерживает все этапы технологии извлечения знаний, включая постановку задачи, подготовку данных, автоматическое построение моделей, анализ и тестирование результатов, использование моделей в реальных приложениях [113].

Существенно, что модели строятся автоматически на основе анализа имеющихся данных об объектах, наблюдениях и ситуациях с помощью специальных алгоритмов. Основу опции ODM составляют процедуры, реализующие различные алгоритмы построения моделей классификации, регрессии, кластеризации.

На этапе подготовки данных обеспечивается доступ к любым реляционным базам, текстовым файлам, файлам формата SAS. Дополнительные средства преобразования и очистки данных позволяют изменять вид представления, проводить нормализацию значений, выявлять неопределенные или отсутствующие значения. На основе подготовленных данных специальные процедуры автоматически строят модели для дальнейшего прогнозирования, классификации новых ситуаций, выявления аналогий. ODM поддерживает построение пяти различных типов моделей. Графические средства предоставляют широкие возможности для анализа полученных результатов, верификации моделей на тестовых наборах данных, оценки точности и устойчивости результатов. Уточненные и проверенные модели можно включать в существующие приложения путем генерации их описаний на C, C++, Java, а также разрабатывать новые специализированные приложения с помощью входящего в состав среды ODM средства разработки Software Development Kit (SDK).

Важной особенностью системы ODM являются его технические характеристики: работа в архитектуре клиент-сервер, широкое использование техники параллельных вычислений, высокая степень масштабируемости при увеличении вычислительных ресурсов.

Характеристики Oracle Data Mining [114]:

- Встроенные в Oracle Database алгоритмы извлечения знаний (DataMining Server).
- DM-инфраструктура вместо готовой инструментальной среды.
- API для разработки.

Встроенные алгоритмы извлечения знаний позволяют упростить процесс извлечения знаний, устраняют необходимость дополнительного перемещения и хранения данных. Обладают производительностью и масштабируемостью.

Oracle Data Mining API. Использование Java API для разработки на Java основано на принципах JDM (стандарт для Data Mining).

Версия Data Mining 10g поддерживает спектр алгоритмов, которые приведены в таблице 26.1.

Таблица 26.1. Алгоритмы, реализованные в Oracle Data Mining

Классификационные модели	Na_ve Bayes, Adaptive Bayes Network
Классификации и регрессионные модели	Support Vector Machine
Поиск существенных атрибутов	Minimal Descriptor Length
Кластеризация	Enhanced K-means, O-cluster
Поиск ассоциаций	Apriory Algorithm
Выделение признаков	Non-Negative Matrix Factorization

Особенность алгоритмов, реализованных в Oracle Data Mining, состоит в том, что все они работают непосредственно с реляционными базами данных и не требуют выгрузки и сохранения данных в специальных форматах. Кроме собственно алгоритмов, в опцию ODM входят средства подготовки данных, оценки результатов, применения моделей к новым наборам данных. Использовать все эти возможности можно как на программном уровне с помощью Java API или PL/SQL API, так и с помощью графической среды ODM Client, которая ориентирована на работу аналитиков, решающих задачи прогнозирования, выявления тенденций, сегментации и другие.

Oracle Data Mining - функциональные возможности

Функции - Oracle Data Mining строит прогнозирующие и дескрипторные модели.

Прогнозирующие модели:

- классификация;
- регрессия;
- поиск существенных атрибутов.

Дескрипторные модели:

- кластеризация;
- поиск ассоциаций;
- выделение признаков.

Прогнозирующие модели

Краткая характеристика алгоритмов классификации

Алгоритмы Naive Bayes (NB):

- Работает быстрее, чем ABN (по времени построения модели).
- Этот алгоритм лучше использовать для числа атрибутов < 200.
- Точность алгоритма меньше, чем в ABN.

Adaptive Bayes Network (ABN):

- Этот алгоритм лучше для большого числа атрибутов.
- Наглядность модели (генерация правил).
- Более точные модели, чем в NB.
- Больше параметров настройки.

Support Vector Machine.

Регрессия

Регрессия применяется для прогнозирования непрерывных величин. Простейшим случаем является линейная регрессия. Используется также метод Support Vector Machine.

Поиск существенных атрибутов

Основная задача - выявление атрибутов, наиболее важних для прогнозирования целевых значений. Используется для ускорения процесса построения классификационной модели.

Используемый алгоритм - Minimum Descriptor Length (MDL).

Дескрипторные модели

Алгоритмы кластеризации

Алгоритм Enhanced k-means Clustering

В этом алгоритме число кластеров изначально задается пользователем. Кластеризация проводится только по числовым атрибутам, их число не должно быть слишком велико. Количество записей может быть каким угодно.

Алгоритм O-Cluster

Этот алгоритм, в отличие от предыдущего, автоматически определяет число кластеров. Он может работать как с числовыми, так и с категориальными атрибутами. Может работать с большим числом атрибутов, т.е. более 10, и с большим количеством записей, более 1000.

Аналитическая платформа Deductor

Состав и назначение аналитической платформы Deductor (разработчик - компания BaseGroup Labs [115]). Deductor состоит из двух компонентов: аналитического приложения Deductor Studio и многомерного хранилища данных Deductor Warehouse [48] .

Архитектура системы Deductor представлена на рис. 26.1.

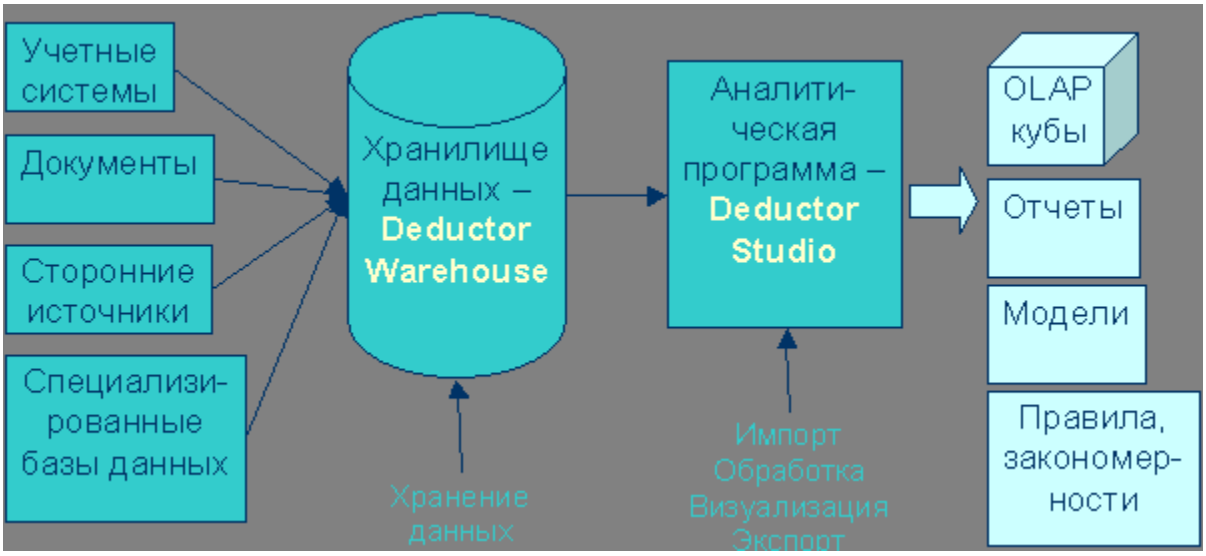

Рис. 26.1. Архитектура системы Deductor

Deductor Warehouse - многомерное хранилище данных, аккумулирующее всю необходимую для анализа предметной области информацию. Использование единого хранилища позволяет обеспечить непротиворечивость данных, их централизованное хранение и автоматически создает всю необходимую поддержку процесса анализа данных. Deductor Warehouse оптимизирован для решения именно аналитических задач, что положительно сказывается на скорости доступа к данным.

Deductor Studio - это программа, предназначенная для анализа информации из различных источников данных. Она реализует функции импорта, обработки, визуализации и экспорта данных. Deductor Studio может функционировать и без хранилища данных, получая информацию из любых других источников, но наиболее оптимальным является их совместное использование.

Поддержка процесса от разведочного анализа до отображения данных

Deductor Studio позволяет пройти все этапы анализа данных. Схема на рис. 26.2 отображает процесс извлечения знаний из данных.

Рис. 26.2. Процесс извлечения знаний из данных в Deductor Studio

Рассмотрим этот процесс более детально.

На начальном этапе в программу загружаются или импортируются данные из какого-либо произвольного источника. Хранилище данных Deductor Warehouse является одним из источников данных. Поддерживаются также другие, сторонние источники:

- текстовый файл с разделителями;
- Microsoft Excel;
- Microsoft Access;
- Dbase;
- CSV-файлы;
- ADO-источники - позволяют получить информацию из любого ODBC-источника (Oracle, MS SQL, Sybase и прочее).

Обычно в программу загружаются не все данные, а какая-то выборка, необходимая для дальнейшего анализа.

После получения выборки можно получить подробную статистику по ней, посмотреть, как выглядят данные на диаграммах и гистограммах.

После такого разведочного анализа можно принимать решения о необходимости предобработки данных. Например, если статистика показывает, что в выборке есть пустые значения (пропуски данных), можно применить фильтрацию для их устранения.

Предобработанные данные далее подвергаются трансформации. Например, нечисловые данные преобразуются в числовые, что необходимо для некоторых алгоритмов.

Непрерывные данные могут быть разбиты на интервалы, то есть производится их дискретизация.

К трансформированным данным применяются методы более глубокого анализа. На этом этапе выявляются скрытые зависимости и закономерности в данных, на основании которых строятся различные модели. Модель представляет собой шаблон, который содержит формализованные знания.

Последний этап - интерпретация - предназначен, чтобы из формализованных знаний получить знания на языке предметной области.

Архитектура Deductor Studio

Вся работа по анализу данных в Deductor Studio базируется на выполнении следующих действий:

- импорт данных;
- обработка данных;
- визуализация;
- экспорт данных.

На рис. 26.3 показана схема функционирования Deductor Studio. Отправной точкой для анализа всегда является процедура импорта данных. Полученный набор данных может быть обработан любым из доступных способов.

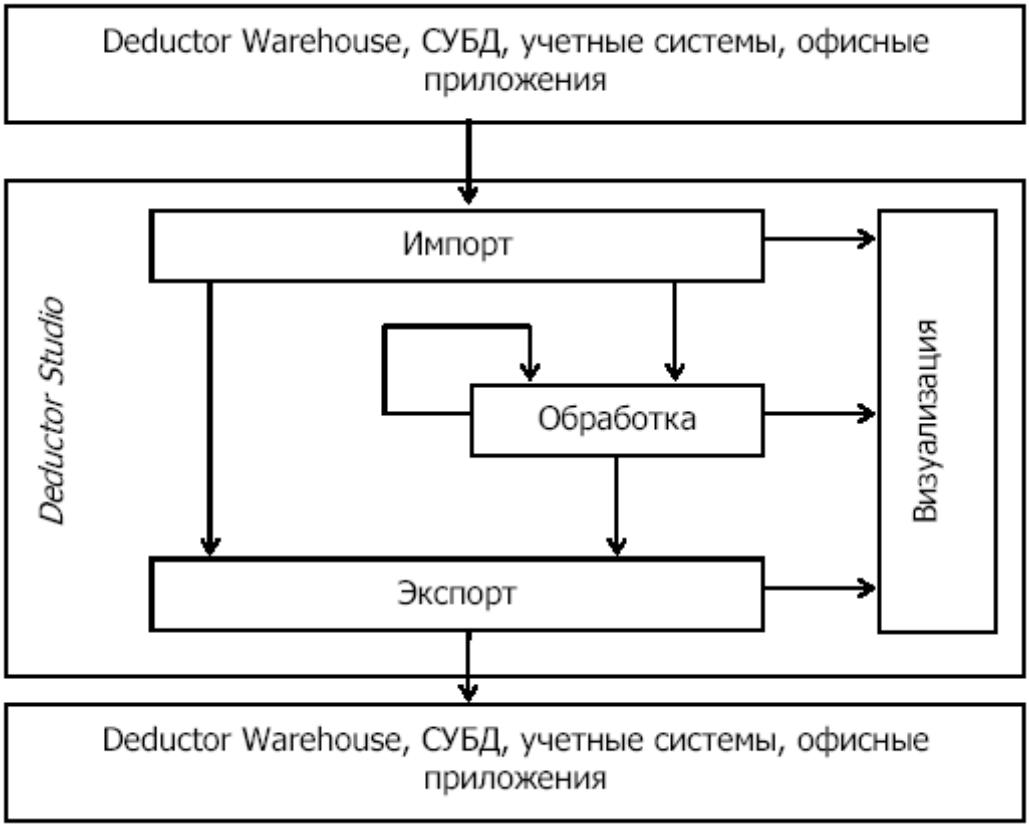

Рис. 26.3. Схема функционирования Deductor Studio

Результатом обработки также является набор данных, который, в свою очередь, опять может быть обработан. Импортированный набор данных, а также данные, полученные на каждом этапе обработки, могут быть **экспортированы** для последующего использования в других, например, в учетных системах. Поддерживаются следующие форматы:

- хранилище данных Deductor Warehouse ;
- Microsoft Excel;
- Microsoft Word;
- HTML;
- XML;
- Dbase;
- буфер обмена Windows;
- текстовой файл с разделителями.

Результаты каждого действия можно **отобразить** различными способами:

- OLAP-кубы (кросс-таблица, кросс-диаграмма);
- плоская таблица;
- диаграмма, гистограмма;
- статистика;
- анализ по принципу "что-если";
- граф нейросети;
- дерево - иерархическая система правил;
- прочее.

Способ возможных отображений зависит от выбранного метода обработки данных. Например, нейросеть содержит визуализатор "Граф нейросети", специфичный только для нее. Некоторые способы визуализации пригодны почти для всех методов обработки, например, в виде таблицы, диаграммы или гистограммы.

Последовательность действий, которые необходимо провести для анализа данных, называется сценарием.

Сценарий можно автоматически выполнять на любых данных. Типовой сценарий изображен на рис. 26.4.

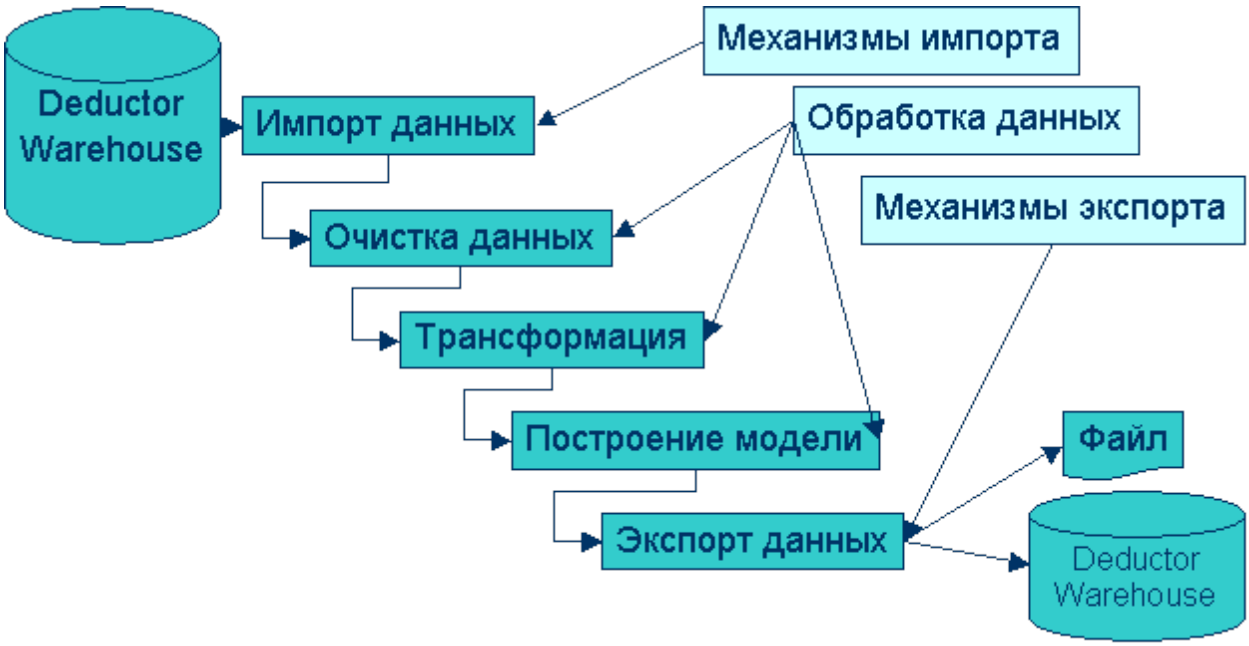

Рис. 26.4. Типовой сценарий Deductor Studio

Архитектура Deductor Warehouse

Deductor Warehouse - многомерное хранилище данных, аккумулирующее всю необходимую для анализа предметной области информацию. Вся информация в хранилище содержится в структурах типа "звезда", где в центре расположены таблицы фактов, а "лучами" являются измерения. Пример такой структуры представлен на рис. 26.5.

Рис. 26.5. Пример структуры типа "звезда"

Такая архитектура хранилища наиболее адекватна задачам анализа данных.

Каждая "звезда" называется процессом и описывает определенное действие.

В Deductor Warehouse может одновременно храниться множество процессов, имеющим общие измерения.

Что представляет собой хранилище Deductor Warehouse ? Физически - это реляционная база данных, которая содержит таблицы для хранения информации и таблицы связей, обеспечивающие целостное хранение сведений. Поверх реляционной базы данных реализован специальный слой, который преобразует реляционное представление к многомерному. Многомерное представление используется потому, что оно намного лучше реляционного соответствует идеологии анализа данных. Благодаря этому слою пользователь оперирует многомерными понятиями, такими как "измерение" или "факт", а система автоматически производит все необходимые манипуляции, необходимые для работы с реляционной СУБД.

Deductor Warehouse реализует универсальное многомерное хранение, т.е. может содержать множество процессов с различным количеством измерений и фактов. Настройка процессов, задание измерений, свойств и фактов задается при первой загрузке в хранилище данных. Вся работа с хранилищем осуществляется средствами Deductor Studio.

Описание аналитических алгоритмов

Кроме консолидации данных, работа по созданию законченного аналитического решения содержит несколько этапов.

Очистка данных. На этом этапе проводится редактирование аномалий, заполнение пропусков, сглаживание, очистка от шумов, обнаружение дубликатов и противоречий.

Трансформация данных. Производится замена пустых значений, квантование, табличная замена значений, преобразование к скользящему окну, изменение формата набора данных.

Data Mining. Строятся модели с использованием нейронных сетей, деревьев решений, самоорганизующихся карт, ассоциативных правил.

Интерпретация результатов.

На рис. 26.6 представлены алгоритмы, используемые в программе, сгруппированные по назначению.

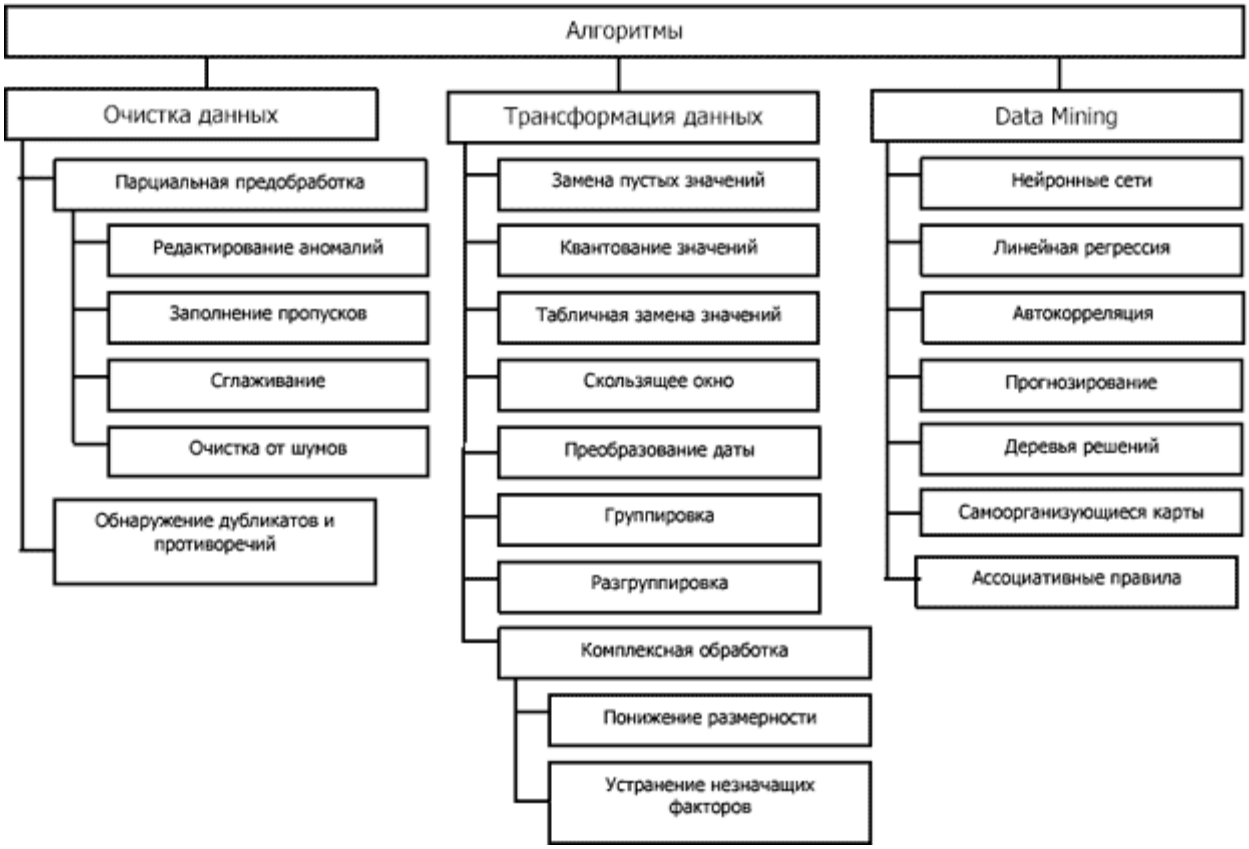

Рис. 26.6. Алгоритмы, используемые в Deductor

Группа 1. Очистка данных

Редактирование аномалий

Автоматическое редактирование аномальных значений осуществляется с применением методов робастной фильтрации, в основе которых лежит использование робастных статистических оценок, таких, например, как медиана. При этом можно задать эмпирически подобранный критерий того, что считать аномалией. Например, задание в качестве степени подавления аномальных данных значения "слабая" означает наиболее терпимое отношение к величине допустимых выбросов.

Заполнение пропусков

В программе предусмотрено два способа заполнения пропущенных данных.

- Аппроксимация - пропущенные данные восстанавливаются методом аппроксимации.
- Максимальное правдоподобие - алгоритм подставляет наиболее вероятные значения вместо пропущенных данных.

Метод аппроксимации рекомендуется использовать в рядах, где данные упорядочены. В этом методе применяется последовательный рекуррентный фильтр второго порядка (фильтр Калмана). Входные данные последовательно подаются на вход фильтра, и если очередное значение ряда отсутствует, оно заменяется значением, которое экстраполируется фильтром.

Метод максимального правдоподобия рекомендуется применять на неупорядоченных данных. При использовании этого метода строится плотность распределения вероятностей, и отсутствующие данные заменяются значением, соответствующим ее максимуму.

Сглаживание

Для сглаживания рядов данных в программе используются два алгоритма.

Первый способ сглаживания - это низкочастотная фильтрация с использованием быстрого преобразования Фурье. При этом задается верхнее значение полосы пропускаемых частот. При подавлении шумов на основе анализа распределения составляющих Фурье спектра на выход фильтра пропускаются спектральные составляющие, которые превышают некоторый порог, рассчитанный по эмпирическим формулам в соответствии с заданным критерием степени вычитания шума. Чем больше требуется сгладить данные, тем меньше должно быть значение полосы. Однако слишком узкая полоса может привести к потере полезной информации. Следует заметить, что этот алгоритм наиболее эффективен, если анализируемые данные есть сумма полезного сигнала и белого шума.

Второй способ сглаживания - это вейвлет-преобразование. Если выбран данный метод, то необходимо задать глубину разложения и порядок вейвлета. "Масштаб" отсеиваемых деталей зависит от глубины разложения: чем больше эта величина, тем более "крупные" детали в исходных данных будут отброшены. При достаточно больших значениях параметра (порядка 7-9) выполняется не только очистка данных от шума, но и их сглаживание ("обрезаются" резкие выбросы). Использование слишком больших значений глубины разложения может привести к потере полезной информации из-за слишком высокой степени "огрубления" данных. Порядок вейвлета определяет гладкость восстановленного ряда данных: чем меньше значение параметра, тем ярче будут выражены "выбросы", и наоборот - при больших значения параметра "выбросы" будут сглажены.

Очистка от шумов

При выборе режима очистки от шумов необходимо задать степень вычитания шума: малую, среднюю или большую. При использовании вычитания шума следует соблюдать осторожность, т.к. реализованный здесь эвристический алгоритм гарантирует удовлетворительные результаты лишь при выполнении двух условий:

1. дисперсия шума значительно меньше энергии полезного сигнала;
2. шум имеет нормальное распределение.

Обнаружение дубликатов и противоречий

Суть обработки состоит в том, что определяются входные и выходные поля. Алгоритм ищет во всем наборе записи, для которых одинаковым входным полям соответствуют одинаковые (дубликаты) или разные (противоречия) выходные поля. На основании этой информации создаются два дополнительных логических поля - "Дубликат" и "Противоречие", принимающие значения "правда" или "ложь".

Группа 2. Трансформация данных

Анализируемая информация, представленная в виде набора данных, имеет определенный формат. Для анализа различных аспектов информации может потребоваться изменение ее формата, или трансформация. Трансформация данных состоит из трех этапов, выполняемых в строгой последовательности (каждый из которых однако, может быть пропущен).

Квантование значений

При выполнении этой операции осуществляется разбиение диапазона числовых значений на указанное количество интервалов определенным методом и замена каждого обрабатываемого значения на число, связанное с интервалом, к которому оно относится, либо на метку интервала. Интервалы разбиения включают в себя нижнюю границу, но не включают верхнюю, кроме последнего интервала, который включает в себя обе границы. Результатом преобразования может быть: номер интервала (от нуля до значения, на единицу меньшего количества интервалов), значение нижней или верхней границы интервала разбиения, среднее значение интервала разбиения, метка интервала.

Квантование может быть осуществлено интервальным или квантильным методом.

Интервальное квантование подразумевает разбиение диапазона значений на указанное количество значений равной длины. Например, если значения в поле попадают в диапазон от 0 до 10, то при интервальном квантовании на 10 интервалов мы получим отрезки от 0 до 1, от 1 до 2 и т.д. При этом 0 будет относиться к первому интервалу, 1 - ко второму, а 9 и 10 - к десятому.

Квантильное квантование подразумевает разбиение диапазона значений на равновероятные интервалы, то есть на интервалы, содержащие равное (или, по крайней мере, примерно равное) количество значений. Нарушение равенства возможно только тогда, когда значения, попадающие на границу интервала, встречаются в наборе данных несколько раз. В этом случае все они относятся к одному определенному интервалу и могут вызвать "перевес" в его сторону.

Табличная замена значений

В результате выполнения этой операции производится замена значений по таблице подстановки, которая содержит пары, состоящие из исходного и выходного значения. Например, 0 - "красный", 1 - "зеленый", 2 - "синий". Или "зима" - "январь", "весна" - "апрель", "лето" - "июль", "осень" - "октябрь". Для каждого значения исходного набора данных ищется соответствие среди исходных значений таблицы подстановки. Если соответствие найдено, то значение меняется на соответствующее выходное значение из таблицы подстановки. Если значение не найдено в таблице, оно может быть либо заменено значением, указанным для замены "по умолчанию", либо оставлено без изменений (если такое значение не указано).

"Скользящее окно"

При решении некоторых задач, например, при прогнозировании временных рядов с помощью нейросети, требуется подавать на вход анализатора значения несколько смежных отсчетов из исходного набора данных. Такой метод отбора данных называется скользящим окном (окно - поскольку выделяется только некоторый непрерывный участок данных, скользящее - поскольку это окно "перемещается" по всему набору). При этом

эффективность реализации заметно повышается, если не выбирать данные каждый раз из нескольких последовательных записей, а последовательно расположить данные, относящиеся к конкретной позиции окна, в одной записи.

Преобразование даты

Разбиение даты необходимо для анализа всевозможных показателей за определенный период (день, неделя, месяц, квартал, год). Суть разбиения заключается в том, что на основе столбца с информацией о дате формируется другой столбец, в котором указывается, к какому заданному интервалу времени принадлежит строка данных. Тип интервала задается аналитиком, исходя из того, что он хочет получить, - данные за год, квартал, месяц, неделю, день или сразу по всем интервалам.

Группировка

Трудно делать какие-либо выводы по данным каждой записи в отдельности. Аналитику для принятия решения часто необходима сводная информация. Совокупные данные намного более информативны, тем более если их можно получить в разных разрезах. В Deductor Studio предусмотрен инструмент, реализующий сбор сводной информации, - "Группировка". Группировка позволяет объединять записи по полям-измерениям, агрегируя данные в полях-фактах для дальнейшего анализа.

Разгруппировка

Группировка используется для объединения фактов по каким-либо измерениям. При этом под объединением понимается применение некоторой функции агрегации. Если в исходном наборе данных присутствовали какие-либо другие измерения, то теряется информация о значениях фактов в разрезе этих измерений. Алгоритм разгруппировки позволяет восстановить эти факты, но их значения восстанавливаются не точно, а пропорционально вкладу в сгруппированные значения.

Комплексная предобработка

Термин "предобработка" можно трактовать шире, а именно, как процесс предварительного экспресс-анализа данных. Например, как оценить, является ли фактор значимым или нет, все ли факторы учтены для объяснения поведения результирующей величины и так далее. Для этих целей используются такие алгоритмы как корреляционный анализ, факторный анализ, метод главных компонент, регрессионный анализ. Подобный анализ в Deductor Studio называется комплексной предобработкой, в рамках которой осуществляется понижение размерности входных данных и/или устранение незначащих факторов.

Понижение размерности пространства факторов

Понижение размерности необходимо в случаях, когда входные факторы коррелированы друг с другом, т.е. взаимозависимы. Имеется возможность пересчитать их в другую систему координат, выделяя при этом главные компоненты. Понижение размерности получается путем отбрасывания компонент, в наименьшей степени объясняющих дисперсию результирующих значений (при этом предполагается, что исходные факторы полностью объясняют дисперсию результирующих факторов).

Требуется указать порог значимости, задающий дисперсию результата. Значение порога значимости может изменяться от 0 до 1.

Устранение незначащих факторов

Устранение незначащих факторов основано на поиске таких значений, которые в наименьшей степени коррелированы (взаимосвязаны) с выходным результатом. Такие факторы могут быть исключены из результирующего набора данных практически без потери полезной информации. Критерием принятия решения об исключении является порог значимости. Если корреляция (степень взаимозависимости) между входным и выходным факторами меньше порога значимости, то соответствующий фактор отбрасывается как незначащий.

Группа 3. Data Mining

Алгоритмы Data Mining в пакете Deductor представлены таким набором:

- нейронные сети;
- линейная регрессия;
- прогнозирование;
- автокорреляция;
- деревья решений;
- самоорганизующиеся карты;
- ассоциативные правила.

Использование нейронных сетей, самоорганизующихся карт и ассоциативных правил на примере пакета Deductor было рассмотрено нами во втором разделе курса лекций.

Инструмент KXEN

Мы продолжаем изучение ведущих мировых производителей программного обеспечения Data Mining. В этой лекции мы остановимся на программном обеспечении KXEN, которое является разработкой одноименной французско-американской компании [116], работающей на рынке с 1998 года. Аббревиатура KXEN означает "Knowledge eXtraction Engines" - "движки" для извлечения знаний.

Сразу следует сказать, что разработка KXEN имеет особый подход к анализу данных [117]. В KXEN нет деревьев решений, нейронных сетей и других популярных техник.

KXEN - это инструмент для моделирования, который позволяет говорить об эволюции Data Mining и реинжиниринге аналитического процесса в организации в целом.

В основе этих утверждений лежат достижения современной математики и принципиально иной подход к изучению явлений в бизнесе.

Следует отметить, что все происходящее внутри KXEN сильно отличается (по крайней мере, по своей философии) от того, что мы привыкли считать традиционным Data Mining.

Бизнес-моделирование KXEN - это анализ деятельности компании и ее окружения путем построения математических моделей. Он используется в тех случаях, когда необходимо понять взаимосвязь между различными событиями и выявить ключевые движущие силы и закономерности в поведении интересующих нас объектов или процессов.

KXEN охватывает четыре основных типа аналитических задач:

- Задачи регрессии/классификации (в т.ч. определение вкладов переменных);
- Задачи сегментации/кластеризации;
- Анализ временных рядов;
- Поиск ассоциативных правил (анализ потребительской корзины).

Построенная модель в результате становится механизмом анализа, т.е. частью бизнес-процесса организации. Главная идея здесь - на основе построенных моделей создать систему "сквозного" анализа происходящих процессов, позволяющую автоматически производить их оценку и строить прогнозы в режиме реального времени (по мере того, как те или иные операции фиксируются учетными системами организации).

Реинжиниринг аналитического процесса

Использование в качестве инструмента для моделирования программного обеспечения KXEN предлагает усовершенствовать аналитический процесс, устранив трудности, часто возникающие в процессе поиска закономерностей, среди которых: трудоемкость подготовки данных; сложность выбора переменных, включенных в модель; требования к квалификации аналитиков; сложность интерпретации полученных результатов; сложность построения моделей. Эти и другие проблемы были нами рассмотрены на протяжении курса лекций.

Особенность KXEN заключается в том, что заложенный в него математический аппарат (на основе Теории минимизации структурного риска Владимира Вапника) позволяет практически полностью **автоматизировать процесс построения моделей** и на порядок увеличить скорость проводимого анализа. Отличия традиционного процесса Data Mining и подхода KXEN приведены на рис. 27.1.

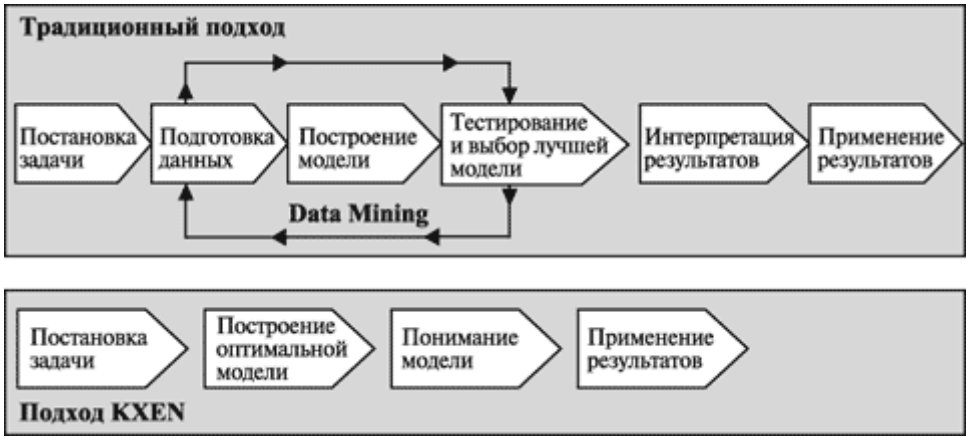

Рис. 27.1. Отличия традиционного процесса Data Mining и подхода KXEN

Таким образом, построение модели в KXEN из исследовательского проекта превращается в функцию предсказательного анализа в режиме on-line в формате "вопрос-ответ". Причем ответы даются в тех же терминах, в которых был сформулирован вопрос, и задача пользователя сводится к тому, чтобы задавать нужные вопросы и указывать данные для анализа.

Среди **преимуществ** KXEN можно назвать:

- Удобная и безопасная работа с данными: данные никуда не перегружаются, KXEN обрабатывает их строка за строкой (текстовые файлы или интеграция с DB2, Oracle и MS SQL Server, в т.ч. через ODBC);
- Наглядность результатов моделирования, легкость для понимания: графическое отображение моделей + score-карты;
- Широкие возможности применения моделей: автоматическая генерация кода моделей на языках C++, XML, PMML, HTML, AWK, SQL, JAVA, VB, SAS, при этом модель сможет работать автономно.

Технические характеристики продукта

KXEN Analytic Framework™ представляет собой набор описательных и предсказательных аналитических модулей, которые можно скомбинировать в зависимости от задачи заказчика. KXEN не является закрытым приложением, он встраивается в существующие системы организации, благодаря открытым программным интерфейсам. Поэтому форма представления результатов анализа, с которой будут работать сотрудники на местах, может определяться пожеланиями Заказчика и особенностями его бизнес-процесса.

Средства KXEN представляют собой приложения в архитектуре Клиент/сервер. Сервер KXEN осуществляет жизненный цикл модели - построение, обучение, корректировку, использование новых данных. С Клиентов осуществляется управление указанными

процессами. Могут быть использованы стандартные клиентские рабочие места, поставляемые KXEN, или разработаны новые под конкретные задачи. Клиентское программное обеспечение KXEN поставляется с исходными кодами и может быть модифицировано или взято в качестве основы для собственной разработки.

Цель дальнейшего материала - познакомить студента с логическими доводами и соображениями, которые легли в основу создания KXEN.

Этот материал будет, в первую очередь, полезен с точки зрения выбора инструментов и методов предсказательного анализа для решения бизнес-задач. Он поможет произвести оценку KXEN и сопоставить его с традиционными решениями в области Data Mining.

Следует отметить, что для работы с KXEN от пользователя не требуется специальной квалификации и знаний в области анализа и статистики. От него требуются данные, которые требуется проанализировать, и определение типа задачи, которую нужно решить. Имеются в виду задачи описательного или предсказательного анализа или, говоря техническим языком, задачи классификации, регрессии или кластеризации.

Предпосылки создания KXEN

В 1990-е годы были получены важные результаты в математике и машинном обучении. Инициатором исследований в этой области стал Владимир Вапник, опубликовавший свою Статистическую Теорию Обучения. Он был первым, кто приоткрыл дверь к новым путям декомпозиции ошибки, получаемой в процессе применения методов машинного обучения. Он обнаружил и описал структуру этой ошибки и на основе сделанных выводов отыскал способ структурировать методы моделирования.

Что же дает такая структура? Вместо того, чтобы случайным образом выбирать и опробовать все имеющиеся методы, она позволяет определить направление для поиска и сравнения методов между собой. Возникает резонный вопрос: "Так значит, все-таки нужно иметь все эти методы, чтобы сравнивать их между собой?" Ответ: "нет". И вот почему.

Поскольку подбор подходящего метода анализа стал осознанным, а не случайным, и в основе его лежат математические выводы, то появилась возможность извлечь мета-алгоритмы и осуществлять такой поиск автоматически. Этот подход используется на всех этапах обработки данных в KXEN. В действительности, идеологи KXEN использовали и проверяли указанные концепции в решении реальных проблем более десяти лет.

Здесь необходимо уточнить один важный момент. При всех достоинствах мета-алгоритмов KXEN, они не идеальны, и связано это с необходимостью компромисса. Пользователю нужна высокая скорость анализа и в то же время легкая интерпретация полученного результата.

Тот математический аппарат, который заложен в KXEN, в ходе анализа строит несколько конкурирующих моделей. Но этот процесс осуществляется не случайным образом (перебором разных методов моделирования), а путем изучения различных наборов моделей с опорой на **Теорию минимизации структурного риска В. Вапника** (Structured Risk Minimization). Создатели KXEN разработали механизм сравнения моделей, с тем чтобы добиться наилучшего соотношения между их точностью и надежностью, и уже эту оптимальную модель представить в качестве результата анализа пользователю.

В предыдущих лекциях мы установили, что одна из ключевых проблем в предсказательном анализе - приведение данных в соответствие с используемыми алгоритмами. Одни алгоритмы воспринимают только символы, другие - только числа. Очень часто эксперту приходится тратить много времени на предварительную подготовку данных и их кодирование (обработка пропусков в данных, обработка выбросов, кодирование данных в зависимости от выбранного алгоритма для анализа и т.д.) Также предполагается, что в распоряжении аналитика есть алгоритмы, которые позволяют получить хорошие и состоятельные результаты.

Каким же образом KXEN решает эту проблему? Разработчики KXEN интегрировали способы автоматической обработки отсутствующих и нетипичных значений и технологию предварительного кодирования. Подготовка данных в KXEN разделена на два этапа. На первом этапе, который называется **"преобразование данных"** (Data Manipulation), специалисты в предметной области выбирают в базе данных или самостоятельно создают атрибуты (переменные, столбцы), которые могут представлять интерес для их задачи. Например, ни одна автоматическая система не скажет, что последняя пятница месяца является хорошим индикатором для прогнозирования денежных потоков между банками. Второй этап подготовки данных включает в себя **оптимальное кодирование указанных атрибутов** для их наилучшего анализа в рамках выбранных алгоритмов. Задача KXEN заключается в том, чтобы, как только бизнес-пользователь проведет описание задачи, обеспечить автоматическое кодирование соответствующих данных и извлечь максимум сведений по поставленному вопросу.

Другим ключевым моментом является **интерпретируемость результатов**. Все компоненты KXEN сконструированы таким образом, чтобы представлять конечным пользователям содержательные результаты, т.е. содержательное наполнение, которое отображается в виде графиков, например, понятие вклада переменных, важности категорий, индикаторов качества и надежности.

Сами по себе методы описательного и предсказательного анализа бесполезны. Чтобы извлечь прибыль из модели прогнозирования оттока клиентов, необходимо внедрить эту модель в операционную среду компании и на основе прогнозной оценки предпринимать те или иные действия. Поскольку перед использованием моделей их необходимо натренировать (адаптировать к текущей ситуации), дескриптивный анализ и прогнозирование - это всего лишь часть процесса. Недостаточно обеспечить себя инструментом, необходимо обеспечить себя компонентами, которые будут интегрированы в операционную среду. В результате появляются следующие требования:

1. Четкий и лаконичный API.
2. Возможность интеграции в любой пользовательский интерфейс.
3. Отсутствие необходимости временного или постоянного копирования данных для анализа.

   Это требование отражает ограничение на архитектуру. Разработчики KXEN намеренно отказались от копирования данных во временное хранилище в процессе анализа. Тренировка моделей в KXEN осуществляется путем нескольких разверток на данных, т.е. "на лету" (строка за строкой). Правда, в этом случае от пользователя может потребоваться посмотреть на выборку несколько раз в зависимости от компонент, которые включаются в анализ.

4. Возможность внедрения моделей в операционную среду компании.

Выполнение этого требования дает возможность не только производить моделирование в режиме on-line, но и экспортировать построенные модели, используя другие программные языки, например Java, SQL, PMML и др. Встраивание модели KXEN в виде программного кода в рабочую базу данных позволяет производить анализ и получать прогнозную оценку в регулярном режиме.

У потенциального пользователя может возникнуть вопрос, почему KXEN не создает отдельное приложение. Ответ достаточно прост - в этой сфере работают уже очень много игроков; также известно, что издержки входа при создании подходящего приложения очень велики. Поэтому создатели KXEN выбрали путь партнерства с ведущими компаниями, которые уже работают на этих вертикальных рынках.

Примером такого партнерства является специальный модуль KXEN для Clementine, хорошо известного приложения Data Mining от SPSS, который интересен как с точки зрения самой интеграции приложений, так и сочетания KXEN с более традиционными техниками Data Mining.

И еще один вопрос, который часто задается потенциальными партнерами: "Зачем мне встраивать технологию KXEN вместо того, чтобы просто связать свое приложение с приложением одного из вендоров (продавцов) Data Mining?" Ответ на этот вопрос следующий: практический опыт показал, что использование дескриптивного анализа и прогнозирования не заканчивается построением модели. Данные меняются со временем, и необходимо периодически производить мониторинг эффективности моделей с целью принятия решения об их корректировке или выставления меток в операционной среде. Компания KXEN включила управление конфигурацией модели в API, тем самым обеспечив сигнализацию об автоматическом выявлении отклонений на входных распределениях или во взаимосвязях входов-выходов. Очевидно, в последнем случае необходимо использование надежных методов, потому что статистические отклонения в производительности модели не должны являться следствием техники моделирования, но должны идентифицировать различия в данных, которые требуется моделировать.

Средства KXEN специально построены на компонентной архитектуре для возможности встраивания в среды не только с целью мониторинга жизненного цикла модели, но и управления этим циклом. Это невозможно через простое соединение с популярным средством прогнозирования. KXEN будет генерировать осмысленные ответы на ситуации из реальной жизни автоматически, просто и действительно быстро. Таким образом, реальный смысл не в том, чтобы запускать внешний пользовательский интерфейс для построения моделей, а в том, чтобы иметь возможность:

- выявлять отклонения в операционной среде;
- запускать переобучение моделей;
- использовать эти модели в режиме реального времени или в процессе пакетной обработки;
- строить операционные пользовательские интерфейсы, которые будут использовать все возможности по построению моделей.

Структура KXEN Analytic Framework Version 3.0

KXEN Analytic Framework по своей сути не является монолитным приложением, а выполняет роль компонента, который встраивается в существующую программную среду.

Этот "движок" может быть подключен к DBMS-системам (например, Oracle или MS SQL-Server) через протоколы ODBS.

KXEN Analytic Framework представляет собой набор модулей для проведения описательного и предсказательного анализа. Учитывая специфику задач конкретной организации, конструируется оптимальный вариант программного обеспечения KXEN. Благодаря открытым программным интерфейсам, KXEN легко встраивается в существующие системы организации. Поэтому форма представления результатов анализа, с которой будут работать сотрудники на местах, может определяться пожеланиями Заказчика и особенностями его бизнес-процесса. На рис. 27.2 представлена структура KXEN Analytic Framework Version 3.0.

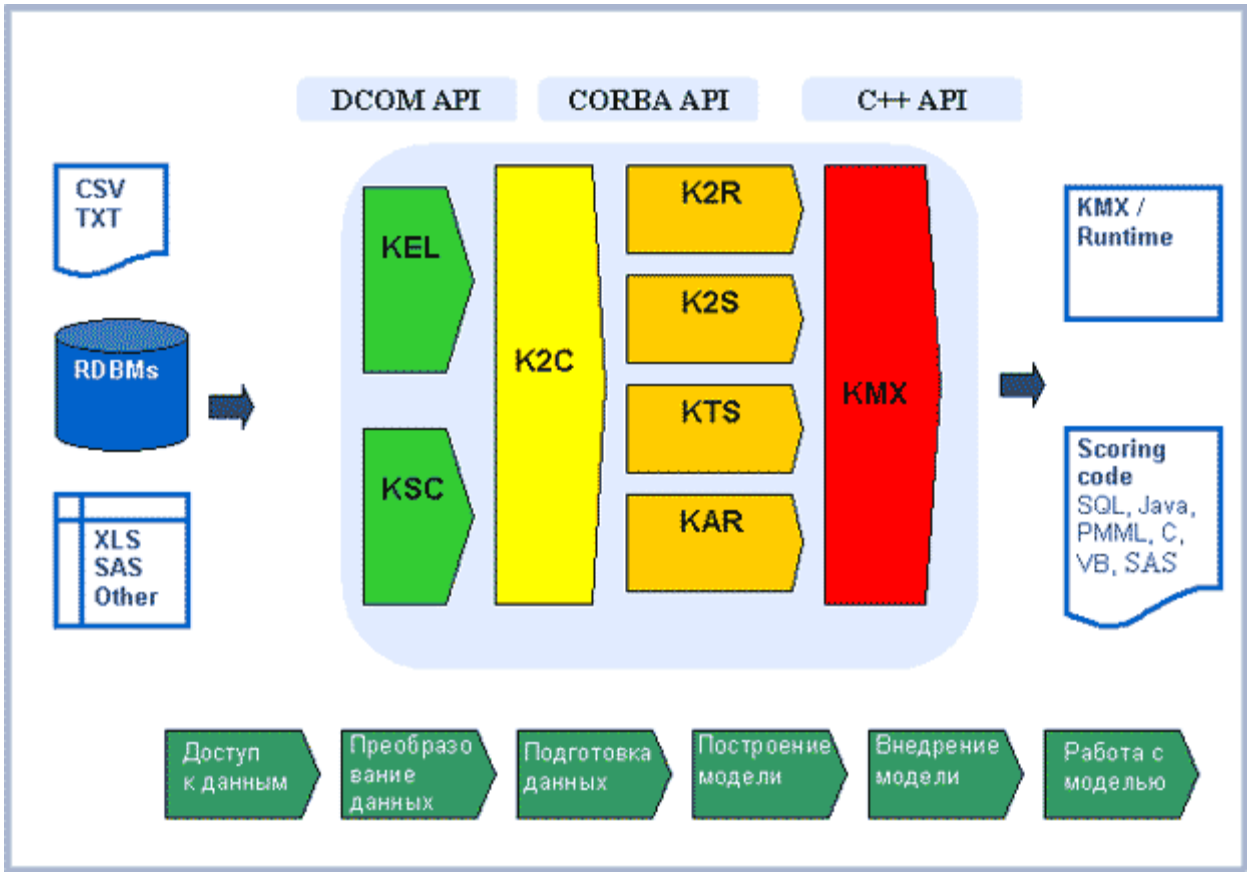

Рис. 27.2. Структура KXEN Analytic Framework Version 3.0

Рассмотрим ключевые компоненты системы KXEN.

Компонент Агрегирования Событий (KXEN Event Log - KEL) предназначен для агрегирования событий, произошедших за определенные периоды времени. Применение KEL позволяет соединить транзакционные данные с демографическими данными о клиенте. Компонент используется в случаях, когда "сырые" данные содержат

одновременно статическую информацию (например, возраст, пол или профессия индивида) и динамические переменные (например, шаблоны покупок или транзакции по кредитной карте). Данные автоматически агрегируются внутри определенных пользователем интервалов без программирования на SQL или внесения изменений в схему базы данных. Компонент KEL комбинирует и сжимает эти данные для того, чтобы сделать их доступными для других компонентов KXEN.

Преимуществом использования данного компонента является возможность интегрировать дополнительные источники информации "на лету" для того, чтобы улучшить качество модели.

Компонент Кодирования Последовательностей (KXEN Sequence Coder - KSC) позволяет агрегировать события в серии транзакций. Например, поток "кликов" клиента, фиксирующийся на Web-сайте, может трансформироваться в ряды данных для каждой сессии. Каждая колонка отражает конкретный переход с одной страницы на другую. Как и в случае с KEL, новые колонки данных могут добавляться к существующим данным о клиентах и доступны для обработки другими компонентами KXEN.

Преимуществом использования данного компонента является возможность применять незадействованные прежде источники информации для того, чтобы улучшить качество прогнозирующих моделей.

Компонент Согласованного Кодирования (KXEN Consistent Coder - K2C) позволяет автоматически подготовить данные и трансформировать их в формат, подходящий для использования аналитическими приложениями KXEN. Использование K2C позволяет трансформировать номинальные и порядковые переменные, автоматически заполнять отсутствующие значения и выявлять выбросы.

Преимуществом использования данного компонента является возможность автоматизации подготовки данных, которая позволяет освободить время для непосредственно исследований и моделирования.

Компонент Робастной Регрессии (KXEN Robust Regression - K2R) использует подходящий регрессионный алгоритм для того, чтобы построить модели, описывающие

существующие зависимости, и сгенерировать прогнозирующие модели. Эти модели могут затем применяться для скоринга, регрессии и классификации. В отличие от традиционных регрессионных алгоритмов, использование K2R позволяет безопасно справляться с большим количеством переменных (более 10 000). Модуль K2R строит индикаторы и графики, которые позволяют легко убедиться в качестве и надежности построенной модели.

Преимуществом использования данного компонента является автоматизация процесса интеллектуального анализа данных. Модели позволяют детализировать индивидуальные вклады переменных.

Компонент Интеллектуальной Сегментации (KXEN Smart Segmenter - K2S) позволяет выявить естественные группы (кластеры) в наборе данных. Модуль оптимизирован для того, чтобы находить кластеры, которые относятся к конкретной поставленной задаче. Он описывает свойства каждой группы и указывает на ее отличия от всей выборки. Как и в случае с другими модулями, этот модуль также строит индикаторы качества и надежности модели.

Преимуществом использования данного компонента является автоматическое выявление групп, значимых для той конкретной задачи, которую необходимо решить.

Машина Опорных Векторов KXEN (Support Vector Machine - KSVM) позволяет производить бинарную классификацию. Использование компонента подходит для решения задач, основанных на наборах данных с небольшим количеством наблюдений и большим количеством переменных. Это делает модуль идеальным для решения задач в областях с очень большим количеством размерностей, таких как медицина и биология.

Преимуществом использования данного компонента является возможность решения задач, которые прежде требовали написания специальных программ, с помощью промышленного программного обеспечения.

Компонент Анализа Временных Рядов (KXEN Time Series - KTS) позволяет прогнозировать значимые шаблоны и тренды во временных рядах. Используйте накопившиеся хронологические данные для того, чтобы спрогнозировать результаты

следующих периодов. Модуль KTS выявляет тренды, периодичность и сезонность для того, чтобы получить точные и достоверные прогнозы.

Преимущество: Появляется возможность подстроиться под повторяющиеся шаблоны Вашего бизнеса и предсказывать сокращения поставок до того как они произойдут.

Компонент Экспорта Моделей (KXEN Model Export - KMX) позволяет создавать коды различного типа: SQL, C, VB, SAS, PMML и многих других для встраивания в существующие приложения и бизнес-процессы. Построенная модель в виде кода может быть передана на другую машину для дальнейшего анализа данных в пакетном либо интерактивном режиме.

Преимущество: Использование данного модуля дает возможность производить анализ вновь поступающей информации с помощью модели автономно, вне самой системы моделирования. Это существенно ускоряет внедрение моделей в производственный процесс и не требует создания специальных условий и программ для анализа всей базы данных с помощью модели. Пользователь также может перевести модель на тот язык, который поддерживает его компьютер.

Технология IOLAP

И, в заключение, рассмотрим технологию IOLAP™ от KXEN - интеллектуальную оперативную аналитическую обработку, позволяющую извлечь из данных наиболее релевантную информацию.

Традиционные OLAP-инструменты предоставляют богатую функциональность для детализации, выделения срезов, движения по данным. Однако при значительных объемах информации возникают ограничения в использовании этой функциональности. Как, например, узнать, какие из 200 измерений в кубе имеют существенное отношение к интересующему пользователя вопросу?

Что позволяет IOLAP:

- Структурировать данные таким образом, что в первую очередь отображается наиболее актуальная для пользователя информация.
- Определить и отобразить переменные по степени их значимости по отношению к интересующему вопросу (вклад объясняющих переменных).
- Детализировать иерархию каждой переменной.
- Определить качество и степень достоверности полученных результатов на основе двух индикаторов.

Технология IOLAP™ использует алгоритмы, заложенные в аналитических приложениях KXEN, и доступна для работы через Microsoft Excel. Технология IOLAP™ легко интегрируется с другими OLAP-средствами и интерфейсами пользователей.

Data Mining консалтинг

В предыдущей лекции мы рассматривали инструменты Data Mining, которые можно приобрести на рынке готового программного обеспечения. Как мы уже упоминали ранее, существуют и другие варианты: заказ готового решения у фирмы-разработчика или адаптация программного обеспечения под конкретную задачу.

Различные варианты внедрения Data Mining имеют свои сильные и слабые стороны. Так, преимуществами готового программного обеспечения являются готовые алгоритмы, техническая поддержка производителя, полная конфиденциальность информации, а также не требуется дописывать программный код, существует возможность приобретения различных модулей и надстроек к используемому пакету, общение с другими пользователями пакета и др.

Однако, такое решение имеет и слабые стороны. В зависимости от инструмента, это может быть достаточно высокая стоимость лицензий на программное обеспечение, невозможность добавлять свои функции, сложность подготовки данных, практическое отсутствие в интерфейсе терминов предметной области и другие. Такое решение требует наличия высококвалифицированных кадров, которые смогут качественно подготовить данные к анализу, знают, какие алгоритмы следует применять для решения каких задач, сумеют проинтерпретировать полученные результаты в терминах решаемых бизнес-задач. Далеко не каждая компания может содержать штат таких специалистов, а зачастую их содержание даже неэффективно.

Представим ситуацию, когда менеджер сталкивается "один на один" с одним из продуктов, в котором реализованы методы технологии Data Mining (от самых простых, включающих 1-2 алгоритма, до полнофункциональных программных комплексов, предлагающих десятки различных алгоритмов). Перед ним стоит задача - выявить наиболее перспективных потенциальных клиентов, а он видит перед собой всего лишь набор математических алгоритмов: Это и есть "обратная сторона" использования готовых инструментов.

Таким образом, покупке готового инструмента должна предшествовать серьезная подготовка к внедрению Data Mining. Некоторые аспекты этой подготовки (ее организационные факторы) были описаны в предыдущем разделе курса.

Если описанные сложности не учтены при внедрении готового инструмента, компания может столкнуться с трудностями, не всегда преодолимыми, и, как результат - разочароваться в технологии Data Mining.

Далее мы рассмотрим другой вариант внедрения: воспользоваться Data Mining-консалтингом и/или так называемой адаптацией программного обеспечения под конкретную задачу.

Data Mining-услуги

По данным консалтинговой компании Meta Group, в мире не менее 85% рынка Data Mining занимают именно услуги, т.е. консультации по эффективному внедрению этой

технологии для решения актуальных бизнес-задач. На сайте KDnuggets можно найти перечень более ста известных компаний, занимающихся консалтингом в сфере Data Mining.

Одна из всемирно известных консалтинговых компаний в сфере Data Mining - компания Two Crows (www.twocrows.com). Она специализируется на публикации отчетности Data Mining, проводит образовательные семинары, консультирует пользователей и разработчиков Data Mining во всем мире. Одна из известных методологий Data Mining - методология, разработанная компанией Two Crows.

К консалтинговым Data Mining-компаниям относят и некоторых производителей готового программного обеспечения, продукцию некоторых мы рассмотрели в этом разделе курса:

- IBM Global Business Intelligence Solutions, www.ibm.com/bi;
- SAS Institute, www.sas.com/datamining;
- SPSS, www.spss.com;
- StatSoft, www.StatSoft.com.

Некоторые консалтинговые компании предоставляют свои услуги на определенных территориях. Это, например, компания Arvato Business Intelligence, www.arvatobi.fr, обеспечивающая Data Mining консультирование и моделирование во Франции, Германии, Испании и некоторых других европейских странах. Некоторые консалтинговые компании специализируются на предоставлении услуг в определенных предметных областях. Например, компания Blue Hawk LLC, www.bluehawk.biz, осуществляет Data Mining и предоставляет консультационные услуги в сферах Direct Marketing и CRM. Некоторые компании предоставляют услуги с использованием определенных методов Data Mining. Компания Bayesia, (www.bayesia.com), предоставляет консультирование и "настройку решения" под клиента на основе байесовской классификации. Компания Visual Analytics (www.visualanalytics.com) обеспечивает услуги по бизнес-консультированию для нахождения шаблонов с использованием визуального Data Mining.

Рассмотрим преимущества, которые имеет этот вариант внедрения Data Mining по сравнению с готовыми программными продуктами и их самостоятельным использованием.

Высококвалифицированные специалисты. Для эффективного применения технологии Data Mining требуются квалифицированные специалисты, которые сумеют качественно провести весь цикл анализа. Пока что таких грамотных специалистов на просторах СНГ очень немного, и потому они довольно дороги. Обучение же собственных, во-первых, достаточно рискованно (его с удовольствием переманит конкурент), во-вторых, выльется в немалые затраты (такие курсы стоят дорого). Клиенты, воспользовавшись услугами консалтинговой компании, получают доступ к высококлассным профессионалам компании, экономя при этом значительные средства на поиске или обучении собственных специалистов.

Адаптированность. Готовые продукты изначально предназначены для решения хотя и широкого, но все же стандартного и ограниченного круга задач - адаптация продукта к условиям конкретного бизнеса ложится на плечи сотрудников компании. Здесь перед заказчиком опять встанет упомянутая проблема квалифицированных специалистов. Консалтинговая компания предоставляет услуги, полностью адаптированные под бизнес заказчика и его задачи.

Гибкость инструмента - его возможность быстро подстроить программное обеспечение под нужды бизнеса:

- возможность выбора наиболее удобных понятий, в терминах которых должны быть сформулированы знания или термины предметной области; так, анализируя конкурентов, действующих на рынке, их можно поделить на "сильных" и "слабых" или же на "агрессивных", "спокойных" и "пассивных" - в зависимости от того, что интересует аналитика в определенный момент. Соответственно, знания будут сформулированы в выбранных заказчиком терминах, и в итоге он получает решение именно в тех терминах, которые ему интересны и понятны.
- получение осмысленных и понятных заказчику знаний в естественной форме. Использование адаптированного под конкретный бизнес программного обеспечения избавит пользователя от необходимости изучения формул или зависимостей в математической форме, а предоставит знания в наиболее интуитивном виде.

Итак, услуги по применению Data Mining становятся все более востребованными. Далее мы опишем практический опыт российской компании SnowCactus, предоставляющей услуги по применению Data Mining.

Компания SnowCactus разработала ряд решений на основе Data Mining разнообразных бизнес-задач [118], например:

- анализ клиентов, выявление наиболее доходных и перспективных покупателей;
- анализ и прогнозирование продаж продукции;
- оптимизация работы с поставщиками;
- оптимизация бюджета продвижения товаров;
- формирование ассортимента;
- оценка кредитоспособности заемщиков;
- повышение эффективности подбора персонала.

В процессе решения каждой конкретной бизнес-задачи специалисты компании изучают имеющиеся в наличии данные и подбирают те математические алгоритмы, которые наиболее подходят для ее решения в данных условиях.

Работа с клиентом

На примере российской компании SnowCactus рассмотрим процедуру работы консалтинговой компании с клиентом. Комплекс услуг этой компании включает в себя планирование, организацию и осуществление полного цикла использования технологии Data Mining для бизнеса.

Весь цикл представлен на рис. 28.1. Он, по своей сути, является методологией Data Mining. Как уже упоминалось в предыдущих лекциях, методология Data Mining может быть разработана внутри организации, в соответствии с последовательностью работ, выполняемых в рамках аналитического процесса.

Рис. 28.1. Цикл использования технологии Data Mining в SnowCactus

Цикл состоит из пяти этапов.

Этап 1. Постановка бизнес-задачи

На первом этапе компания вместе с заказчиком формулирует конкретные бизнес-задачи. При первом прохождении этого цикла задача может быть поставлена довольно широко: например, построить профили высокоприбыльных клиентов или определить группы нелояльных покупателей. Во время дальнейших проходов поставленные задачи можно уточнять, расширять и углублять. При формулировании задачи компания учитывает наличие данных, необходимых для ее решения. На этом этапе специалисты компании наравне со специалистами клиента принимают непосредственное участие в процессе формулирования задач, избавляя клиента от технической необходимости ставить задачу в терминах технологии Data Mining.

Этап 2. Первичное исследование данных

После того как бизнес-задача сформулирована, специалисты компании приступают к предварительному исследованию данных, необходимых для решения поставленной задачи. Этот этап компания также практически полностью берет на себя - со стороны заказчика здесь может потребоваться лишь минимальное участие для выяснения, например, смысла исследуемых данных или формулирования интересных для него понятий.

Этап 3. Подготовка данных

На третьем этапе специалисты компании подготавливают данные для их дальнейшего анализа. Для этого используется весь спектр методов подготовки данных, в каждом конкретном случае специалисты выбирают наиболее подходящие методы.

Этап 4. Анализ данных

Основной этап - четвертый - непосредственно анализ данных. Это полностью технический процесс, который специалисты компании проводят самостоятельно при помощи, в основном, собственных разработок. Спектр применяемых алгоритмов очень широк - от методов нечеткой кластеризации и деревьев решений до нейронных сетей и методов извлечения нечетких лингвистических правил.

Этап 5. Интерпретация результатов

На последнем этапе цикла специалисты компании вместе с клиентом занимаются интерпретацией полученных знаний. Это значит, что компания, во-первых, представляет найденные знания в удобной и понятной для заказчика форме, они вместе выясняют, какое значение результаты имеют для бизнеса клиента, а затем, при необходимости, отвечают на сопутствующие вопросы клиента и уточняют полученные знания.

После решения поставленной на первом этапе бизнес-задачи у клиента могут появиться новые вопросы, возникнуть новые бизнес-задачи. Например, он захочет уточнить и расширить полученные знания. В этом случае компания возвращается к первому этапу - постановке новой или уточнению решенной бизнес-задачи, и снова проходит по все этапам, таким образом предоставляя клиенту наиболее полные и качественные знания для развития его бизнеса.

Примеры решения

Возьмем два примера решения задач, один из них - оценка кредитоспособности заемщика банка. Задача "Выдавать ли кредит?" уже рассматривалась нами на протяжении курса. Рассмотрим реализацию этой задачи в системе dm-Score - адаптированного под конкретную бизнес-задачу программного обеспечения кредитного скоринга.

Пример 1. Система кредитного скоринга dm-Score (система, предназначенная для оценки кредитоспособности заемщиков - физических лиц банка).

Это задача внедрения системы кредитного скоринга dm-Score (dm - от Data Mining) в банке для анализа кредитных историй и выявления скрытых влияний параметров заемщиков на их кредитоспособность.

Такая система должна вписываться в информационное пространство банка, т.е. напрямую взаимодействовать с базами данных, где хранится информация о заемщиках и кредитах, с автоматизированной банковской системой (АБС), другим программным обеспечением, и работать с ними как единое целое.

В процессе внедрения специалисты знакомятся с используемыми в банке АБС системой автоматизации ритейла, базами данных и т.д., согласовывают с специалистами требования к системе скоринга - как функциональные, так и нефункциональные, а также изучают, какие данные накоплены банком и какие задачи они позволяют решать, осуществляют адаптацию системы в соответствии с ними.

Одним из важных преимуществ внедрения системы dm-Score является то, что в процессе внедрения учитываются все индивидуальные требования и пожелания к ней со стороны банка. Важно также отметить, что в этом случае происходит интеграция системы dm-Score в информационное пространство банка, а не наоборот, т.е. внедрение не потребует каких-либо изменений в существующих бизнес-процессах.

Таким образом, в результате внедрения Ваш банк получает систему скоринга, которая учитывает все специфические особенности и потребности банка-клиента.

Описываемая система dm-Score позволяет решать следующие задачи:

- оценка кредитоспособности заемщика (скоринг заемщика);
- принятие решения о выдаче кредита или отказе в нем. При этом система может объяснить специалисту банка, почему было принято именно такое решение;
- определение максимального размера кредита (лимита кредита по кредитной карте) на основе скоринга заемщика;
- вынесение профессионального суждения о кредитном риске по ссудам;
- выработка индивидуальных условий кредитования для каждого заемщика с учетом риска для банка;
- прогнозирование поведения заемщика, т.е. наличие и частоту просрочек конкретного заемщика, средний размер используемого кредита по кредитной карте и т.д.;
- оптимизация анкеты заемщика (исключение не значимых вопросов без ухудшения качества анкеты);
- проверка анкеты конкретного заемщика на полноту и внутреннюю непротиворечивость;
- решение других задач, специфичных для конкретного банка.

Техническое описание решения

Как уже отмечалось, система кредитного скоринга dm-Score является решением, полностью интегрированным с используемым в банке программным обеспечением: АБС, системой автоматизации ритейла, СУБД и др. В процессе внедрения она вписывается в информационное пространство банка, взаимодействует и работает с ним как единое целое. Такой подход позволяет избежать ненужного дублирования функций и, как следствие, приводит к более эффективному использованию имеющихся в банке ресурсов. Схематично устройство системы dm-Score изображено на рис. 28.2.

Рис. 28.2. Устройство системы dm-Score

Система dm-Score состоит из двух аналитических блоков - блока анализа данных и блока принятия решений.

Блок анализа (серверная часть). В блоке анализа системы dm-Score осуществляется анализ данных о заемщиках банка, о выданных кредитах и истории их погашения на основе аналитической технологии Data Mining. Благодаря интеграции с АБС банка, блок анализа может получать данные напрямую из нее.

Система dm-Score делает свои выводы на основе данных, уже накопленных банком в процессе работы на рынке розничного кредитования. При этом в процессе внедрения система настраивается именно на тот набор данных, на который ориентирован конкретный банк. Иными словами, система dm-Score готова работать с теми данными, которые есть в наличии, и не требует фиксирования на какой-либо конкретной жестко заданной анкете.

В процессе анализа данных о заемщиках и кредитах применяются различные математические методы, которые выявляют в них факторы и их комбинации, влияющие на кредитоспособность заемщиков, и силу их влияния. Обнаруженные зависимости составляют основу для принятия решений в соответствующем блоке. Блок анализа должен периодически использоваться для анализа новых данных банка (приходят новые заемщики, текущие производят выплаты), для обеспечения актуальности системы dm-Score и адекватности принимаемых ею решений.

Блок принятия решений (клиентская часть). Блок принятия решений используется непосредственно для получения заключения системы dm-Score о кредитоспособности заемщика, о возможности выдачи ему кредита, о максимально допустимом размере кредита и т.д. С этим блоком работает сотрудник банка, который либо вводит в него анкету нового заемщика, либо получает ее из торговой точки, где банк осуществляет программу потребительского кредитования.

Благодаря тесной интеграции системы dm-Score с информационным пространством банка, результаты работы этого блока передаются непосредственно в АБС и систему автоматизации ритейла, которые уже формируют все необходимые документы, ведут историю кредита и т.д. Таким образом, и система dm-Score, и все банковские системы работают как одно целое, повышая производительность труда сотрудников банка.

В результате решения рассмотренной выше задачи с использованием технологии Data Mining банк получает определенные преимущества, например, в сравнении с использованием экспертных методик.

Первая из них - это объективность. Data Mining, в отличие от экспертных методик, находит объективные закономерности между различными факторами, таким образом позволяя минимизировать влияние субъективного человеческого фактора на принятие решений.

Автоматизация. В отличие от экспертных методик, методика на основе Data Mining может быть эффективно автоматизирована и способна обрабатывать большие потоки заявок в режиме реального времени. На вход поступает анкета заемщика, система сразу же выдает решение - кредитный рейтинг, лимит кредита и т. д.

Точность. В отличие от статистических методов анализа данных, технология Data Mining осуществляет более глубокий анализ, выявляя зависимости, которые неочевидны. А это значит, что методика на основе Data Mining учитывает больше важных факторов и, следовательно, дает более точные рекомендации. В частности, это подтверждается успешным опытом применения технологии ведущими западными банками.

Адаптируемость. Со временем кредитная ситуация меняется, поэтому необходимо постоянно отслеживать изменения в поведении заемщиков. Методика, основанная на технологии Data Mining, учитывает все эти изменения, так как периодически производит анализ новых данных. Таким образом, она постоянно адаптируется под изменяющиеся условия. Это также позволяет принимать более обоснованные и точные кредитные решения.

Гибкость. Иногда возникает необходимость внести изменения в анкету заемщика, претендующего на кредит, - например, добавить дополнительные пункты, какие-то убрать, изменить варианты ответов на вопрос и т.д. Хорошая методика не должна при этом требовать привлечения квалифицированных экспертов для ее адаптации под новую структуру данных.

Объяснимость. Еще одна важная характеристика хорошей методики: возможность объяснить, почему данный заемщик получил определенный кредитный рейтинг (например, почему ему следует отказать в выдаче кредита) или почему ему следует установить именно такой лимит по карточке и т.д.

Пример 2.Анализ резюме: пример решения практической бизнес-задачи клиента.

Приведем пример решения конкретной бизнес-задачи одного из рекрутинговых агентств, в которой технология Data Mining применялась для анализа резюме. Это агентство специализируется на подборе персонала для IT-компаний, за время работы оно успело накопить базу из нескольких тысяч резюме кандидатов на различные вакансии.

Сначала была решена проблема разного формата всех резюме, при этом разработанный стандарт разметки позволил вновь присланные резюме размечать сразу же при поступлении. Таким образом, попутно была решена и другая задача компании - создание эффективного стандарта систематизации накопленной информации. Такая система нужна агентству не только для анализа резюме при помощи Data Mining, но и для более эффективного поиска по базе, для статистической обработки и т.д. После разметки резюме специалисты компании приступили к подготовке данных для анализа, важность этого этапа была описана в предыдущем разделе курса лекций. Следующий этап - проведение непосредственного анализа данных при помощи специально разработанного инструментария Data Mining.

В результате анализа удалось построить подробные профили (портреты) лояльных сотрудников и тех, кто склонен менять работу чаще, чем раз в год, были построены профили различных возрастных и иных социально-демографических групп, сотрудников различных отделов, выпускников различных вузов и многое другое. Например, выяснилось, что наиболее склонны к постоянной смене мест работы женщины 20-25 лет. Для сотрудников отдела маркетинга также характерна частая смена мест работы. Какие выводы из этого делает агентство при подборе персонала? Если для клиента важно найти лояльного и преданного сотрудника, который не уйдет с работы через полгода, агентство фокусирует свои поиски на мужчинах 35-45 лет, окончивших Московский Государственный Университет. Если же клиенту важно быстро найти человека на временную работу, агентство может предложить ему девушку 20-25 лет. Или другой пример: при помощи Data Mining компания выявила, что успешная работа на топ-менеджерских позициях в IT-компаниях наиболее характерна для выпускников того же Московского Государственного Университета. Каков вывод? Когда агентству нужно найти клиенту хорошего исполнительного директора, оно фокусирует поиски на этих выпускниках и делает свою работу быстро и более качественно.

Таким образом, благодаря технологии Data Mining агентство может заранее сузить круг поиска кандидатов и, следовательно, проводить подбор персонала более эффективно - быстрее и с меньшими издержками.

Выводы

Выбор инструментального средства Data Mining и способа его внедрения должен проводиться в соответствии с конкретными целями и задачами, учитывать уровень финансовых возможностей компании, квалификацию пользователей и целый ряд других факторов. Поскольку внедрение Data Mining почти всегда требует серьезных финансовых затрат. Также следует не только учитывать задачи, которые стоят перед компанией сегодня, но и рассчитывать на возможность возникновения новых задач в перспективе.