ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ

**Федеральное государственное
образовательное бюджетное учреждение
высшего профессионального образования
«САНКТ-ПЕТЕРБУРГСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ
им. проф. М. А. БОНЧ-БРУЕВИЧА»**

М. Б. Вольфсон

АНАЛИЗ ДАННЫХ

УЧЕБНОЕ ПОСОБИЕ

СПб ГУТ)))

**САНКТ-ПЕТЕРБУРГ
2015**

УДК 004.6 (075.8)
ББК 32.973я73
В72

Рецензенты:
кандидат физико-математических наук,
доцент кафедры высшей математики СПбГУТ *А. Л. Алимов,*
кандидат технических наук, доцент кафедры
информационных технологий в экономике СПбГУТ *Ю. П. Левчук*

Утверждено редакционно-издательским советом СПбГУТ
в качестве учебного пособия

Вольфсон, М. Б.
В72 Анализ данных : учебное пособие / М. Б. Вольфсон ; СПбГУТ. – СПб., 2015. – 82 с.

Рассматриваются принципы организации и построения систем поддержки принятия управленческих решений на предприятии на основе хранилищ данных, а также теоретические основы, методы и средства анализа данных на базе методологий OLAP и Data Mining.

Предназначено для студентов, обучающихся по направлению 38.03.05 «Бизнес-информатика», а также магистров и аспирантов. Возможно использование при выполнении курсовых и дипломных работ.

УДК 004.6 (075.8)
ББК 32.973я73

СОДЕРЖАНИЕ

ПРЕДИСЛОВИЕ

Современные условия ведения бизнеса предъявляют повышенные требования к системам управления: в условиях динамичной внешней среды и ужесточения конкуренции все более значительную роль начинают играть методы и модели экономического анализа, позволяющие оперативно реагировать на возникающие проблемы и имеющиеся возможности.

Задачи бизнес-анализа очень непросты, но здесь на помощь руководителю приходят современные управленческие концепции и технологии. Сегодня практически любая компания, будь то крупная или совсем небольшая, частная или государственная, использует в своей деятельности информационные системы. Это означает, что большинство предприятий уже владеет определенным объемом накопленных данных, представляющих собой немалую ценность.

Ценность корпоративных данных заключается не только в совокупной стоимости отдельных записей, но и зачастую в намного превышающей ее стоимости набора данных как источника дополнительной информации, которую невозможно получить на основании одной или нескольких записей, – такой, как сведения о закономерностях, тенденциях или взаимозависимостях между какими-либо данными, позволяющие принимать управленческие решения.

Именно поэтому в состав современных средств управления предприятиями, банковских информационных систем, других бизнес-приложений обычно включаются не только средства ввода и редактирования данных, но и средства их аналитической обработки, позволяющие тем или иным способом выявлять и представлять закономерности и тенденции в данных. Средства эти сегодня весьма разнообразны. Они включают в себя инструменты для построения и использования хранилищ данных, средства аналитической обработки данных, инструменты интеллектуального анализа данных и проч.

В первом разделе данного учебного пособия рассматривается концепция хранилищ данных, показывается их место при разработке систем бизнес-аналитики, изучаются типовые архитектуры хранилищ данных и принципы их функционирования.

Второй раздел посвящен технологии оперативной аналитической обработки данных (OLAP) на основе многомерных хранилищ данных.

В третьем разделе подробно рассматриваются методы и сферы применения технологии Data Mining для процесса поддержки принятия решений. Описание каждого метода сопровождается конкретным примером его использования.

1. ХРАНИЛИЩА ДАННЫХ

1.1. Задачи хранения и анализа данных

С каждым днем информация играет все более важную роль в нашей жизни. Ежедневно мы выходим в Интернет для поиска информации, общения в социальных сетях, для отправки и получения электронной почты, обмена фотографиями и видео. Информация, созданная частными лицами, приобретает ценность, когда ею обмениваются с другими людьми. В момент создания эта информация обычно размещается на ноутбкуах, мобильных телефонах, видеокамерах и проч. Для обмена ее нужно загружать через сеть в центры хранения данных.

Значимость, взаимосвязанность и объем информации в мире бизнеса тоже продолжают расти стремительными темпами. Успех в бизнесе зависит от быстрого и надежного доступа к соответствующей информации.

Авторы седьмого ежегодного исследования IDC «Цифровая вселенная» (IDC Digital Universe)[1], проведенного по заказу EMC, ожидают, что к 2020 г. произойдет 10-кратный рост мирового объема цифровой информации, при этом 10 % этого объема будет создаваться датчиками. Ученые прогнозируют, что с 2013 по 2020 гг. количество данных увеличится с 4,4 до 44 зеттабайт. Объем информации удваивается каждые 2 года, основной вклад в этот рост вносит интернет вещей.

По оценкам IDC, количество устройств и предметов, которые можно подключить к интернету в мире, приближается к 200 млрд, из которых 14 млрд, или 7 %, уже подключены и активно передают данные. На сегодняшний день данные от таких устройств составляют 2 % от мирового объема информации. Согласно прогнозам IDC, к 2020 г. уже 32 млрд подключенных устройств будут генерировать 10 % общего объема данных во всем мире.

Развитие интернета вещей также увеличит долю пригодных для анализа данных. На сегодняшний день только 22 % информации может быть полезным и только 5 % фактически анализируется. Остальные массивы авторы исследования называют «космическим мусором». Предполагается, что к 2020 г. благодаря развитию интернета вещей более 35 % информации будут считаться полезными.

Чтобы проиллюстрировать увеличение объемов информации в мире, исследователи приводят несколько наглядных примеров. Так, если записать в iPad Air (толщиной в 29 дюймов с 128 Гб памяти) весь объем информации, то понадобится батарея планшетов длиной 253 704 км, что составляет две трети расстояния до Луны, а к 2020 г. не хватит и шести таких

[1] http://russia.emc.com/collateral/analyst-reports/idc-digital-universe-2014.pdf

батарей. Другой пример: если представить один байт данных как кастрюлю с водой емкостью 3,7 л, то данных, создаваемых в наши дни за 10 с, будет достаточно, чтобы полностью залить дом средних размеров. В 2020 г. на это понадобится всего 2 с.

По прогнозам, объемы данных в России будут расти немного медленнее, чем в мире. Объем информации в России аналитики оценили в 155 экзабайт (2,4 % от мирового объема) и прогнозируют его рост до 980 экзабайт (2,2 % от мирового объема) к 2020 г. Увеличению объема информации в России будут способствовать рост числа пользователей Интернета, социальных сетей и смартфонов, а также миграция с аналогового телевидения на цифровое. Причину нестабильного развития систем хранения в России аналитики видят в высокой стоимости инфраструктуры и связанном с этим ограниченном финансировании.

Аналитики IDC делают тревожный вывод о том, что рост объемов данных существенно опережает рост емкости систем хранения. В 2013 г. совокупная доступная емкость систем хранения соответствовала 33 % объема цифровой информации. К 2020 г. ее будет достаточно для хранения менее чем 15 %. В 2013 г. менее 20 % данных размещалось в облаке, к 2020 г. эта величина удвоится и составит 40 %.

Во всем мире организации накапливают или уже накопили в процессе своей административно-хозяйственной деятельности большие объемы данных, в том числе и в электронном виде. Эти коллекции данных хранят в себе большие потенциальные возможности по извлечению новой аналитической информации, на основе которой можно и необходимо строить стратегию организации, выявлять тенденции развития рынка, находить новые решения, обусловливающие успешное развитие в условиях конкурентной борьбы.

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. Они 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.

До начала компьютерной эпохи данные создавались и передавались преимущественно на бумаге и пленке. Сегодня данные можно создавать посредством компьютера и хранить в цифровом виде.

Коммерческие предприятия генерируют огромное количество цифровых данных, а затем извлекают значимую информацию из этих данных с целью экономической выгоды. Таким образом, предприятиям необходимо заниматься сохранением данных и обеспечивать возможность доступа к ним на протяжении длительного периода.

В зависимости от способа управления и хранения данные можно классифицировать на структурированные и неструктурированные. **Структурированные данные** организуют в ряды и колонки строго определенного формата, чтобы приложения могли извлекать данные и эффективно обрабатывать их. Обычно хранятся с применением СУБД.

К **неструктурированным данным** можно отнести: офисную документацию, графические данные, чертежи, веб-страницы, сообщения электронной почты и ICQ, видео- и аудиофайлы и другие мультимедийные активы.

Предприятия заинтересованы главным образом в управлении неструктурированными данными, поскольку более 80 % данных предприятий не структурированы и требуют много места для их хранения, а также больших усилий для управления ими.

Компании анализируют исходные данные для выявления значимых тенденций. Основываясь на этих тенденциях, компания может спланировать или изменить свой подход. Эффективный анализ данных не только приносит прибыль существующим предприятиям, но и создает потенциал для новых деловых возможностей при плодотворном использовании информации.

1.2. Системы оперативной обработки информации

К середине 80-х гг. XX в. практически полностью завершился первый этап оснащения бизнеса и государственных структур средствами вычислительной техники и начался период бурного развития информационных систем для организации сбора и хранения больших массивов различного рода деловой и служебной информации. В основном это были корпоративные системы, предназначенные для оперативной обработки информации, которые обслуживали бухгалтерию, информационные архивы, телефонные сети, регистрацию документов, банковские операции и т. д. С появлением персональных компьютеров такие системы стали доступными для множества мелких и средних фирм, предприятий и организаций. **Системы оперативной обработки информации** получили название **OLTP** (On-Line Transaction Processing – оперативная, т. е. в режиме реального времени, обработка транзакций).

Транзакция – некоторый набор операций над базой данных, который рассматривается как единое завершенное, с точки зрения пользователя, действие над некоторой информацией, обычно связанное с обращением к базе данных [1].

Приложение OLTP часто интерактивно, т. е. обновление данных происходит в процессе оперативных транзакций. Для легкого и быстрого доступа к оперативной информации сегодня ее обычно хранят в реляционной базе данных.

Главное требование к OLTP-системам – быстрое обслуживание относительно простых запросов большого числа пользователей, при этом время ожидания выполнения типового запроса не должно превышать несколько секунд.

Обобщенная структура системы OLTP представлена на рис. 1.

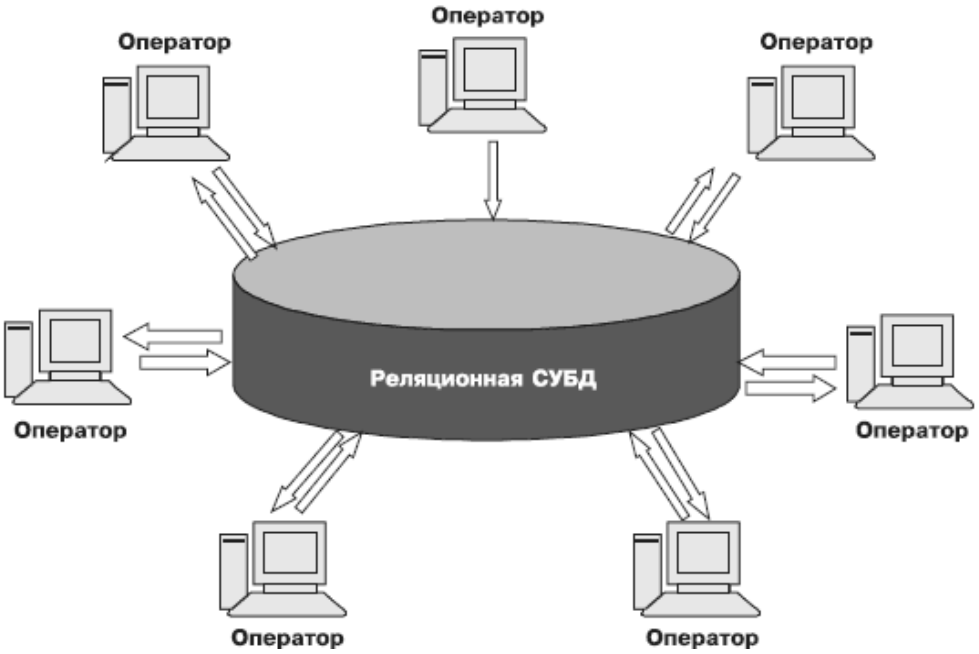


Рис. 1. OLTP-система

Типичным примером применения OLTP-систем является массовое обслуживание клиентов, например бронирование авиабилетов или оплата услуг телефонных компаний. Обе эти ситуации имеют два общих свойства: очень большое число клиентов и непрерывное поступление информации.

Со временем в таких системах начали аккумулироваться большие объемы данных – документы, сведения о банковских операциях, информация о клиентах, заключенных сделках, оказанных услугах и т. д.

Любая транзакционная система, как правило, содержит два типа таблиц. Один из них отвечает за быстрые транзакции.

Например, при продаже билетов необходимо обеспечить работу большого числа кассиров, которые обмениваются с системой короткими сообщениями. Действительно, вводимая и распечатываемая информация, касающаяся фамилии пассажира, даты вылета, рейса, места, пункта назначения, может быть оценена в 1000 байт. Таким образом, для обслуживания пассажиров необходима быстрая обработка коротких записей.

Другой тип таблиц содержит итоговые данные о продажах за указанный срок, по направлениям, по категориям пассажиров. Эти таблицы используются аналитиками и финансовыми специалистами раз в месяц, или в конце года, когда необходимо подвести итоги деятельности компании.

И если количество аналитиков в десятки раз меньше числа кассиров, то объемы данных, необходимых для анализа, превышают размер средней транзакции на несколько порядков величины.

Естественно, что во время выполнения аналитических работ время отклика системы на запрос о наличии билета увеличивается. Создание систем с резервом вычислительной мощности может сгладить негативное воздействие аналитической нагрузки на транзакционную активность, но приводит к значительному удорожанию комплекса, притом, что избыточная мощность большую часть времени остается невостребованной. Вторым фактором, приведшим к разделению аналитических и транзакционных систем, являются разные требования, которые предъявляют аналитические и транзакционные системы к вычислительным комплексам.

1.3. Системы поддержки принятия решений

Компьютерная революция позволила легко собирать и дешево хранить большие объемы оперативных данных. Многие компании хранят свои данные за многие годы. Собранная информация может оказаться весьма полезной в процессе управления организацией, поиска путей совершенствования деятельности и получения посредством этого конкурентных преимуществ.

Традиционно анализ данных в поисках необходимой информации выполнялся вручную. Но в последние несколько лет стало очевидно, что автоматизация анализа данных не только значительно облегчает этот процесс, но и экономически выгодна. Для этого нужны системы, которые позволяли бы выполнять не только простейшие действия над данными: подсчитывать суммы, средние, максимальные и минимальные значения. Появилась потребность в информационных системах, которые позволяли бы проводить глубокую аналитическую обработку, для чего необходимо решать такие задачи, как поиск скрытых структур и закономерностей в массивах данных, вывод из них правил, которым подчиняется данная предметная область, стратегическое и оперативное планирование, формирование нерегламентированных запросов, принятие решений и прогнозирование их последствий.

В 1996 г. агентство Gartner Group, занимающееся анализом рынков информационных технологий, уточнило введенный ею ранее термин «Business Intelligence» (BI). **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)**. СППР ориентированы на аналитическую обработку данных с целью получения знаний, необходимых для разработки решений в области управления. Дополнительным стимулом совершенствования этих систем стали такие факторы, как снижение стоимости высокопроизводительных компьютеров и расходов на хранение больших объемов информации, появление возможности обработки больших массивов данных и развитие соответствующих математических методов.

Рассмотрим основные отличия систем OLTP и СППР (табл. 1).

Таблица 1

Отличия СППР и OLTP-систем

Свойство	OLTP-система	СППР
Цели использования данных	Быстрый поиск, простейшие алгоритмы обработки	Аналитическая обработка с целью поиска скрытых закономерностей, построения прогнозов и моделей и т. д.
Уровень обобщения (детализации) данных	Детализированные	Как детализированные, так и обобщенные (агрегированные)
Требования к качеству данных	Возможны некорректные данные (ошибки регистрации, ввода и т. д.)	Ошибки в данных не допускаются, поскольку могут привести к некорректной работе аналитических алгоритмов
Формат хранения данных	Данные могут храниться в различных форматах в зависимости от приложения, в котором они были созданы	Данные хранятся и обрабатываются в едином формате
Время хранения данных	Как правило, не более года (в пределах отчетного периода)	Годы, десятилетия
Изменение данных	Данные могут добавляться, изменяться и удаляться	Допускается только пополнение; ранее добавленные данные изменяться не должны, что позволяет обеспечить их хронологию
Периодичность обновления	Часто, но в небольших объемах	Редко, но в больших объемах
Доступ к данным	Должен быть обеспечен доступ ко всем текущим (оперативным) данным	Должен быть обеспечен доступ к историческим (т. е. накопленным за достаточно длительный период времени) данным

Окончание табл. 1

Свойство	OLTP-система	СППР
Характер выполняемых запросов	Стандартные, настроенные заранее	Нерегламентированные, формируемые аналитиком «на лету» в зависимости от требуемого анализа
Время выполнения запроса	Несколько секунд	До нескольких минут (важно, но не критично)
Число пользователей	Поддержкой большого числа пользователей	Небольшое число пользователей (аналитики)

В зависимости от данных, с которыми работают СППР, выделяют два основных их типа: **статические** (информационные системы руководителя, Executive Information Systems – EIS) и **динамические DSS**.

EIS являются оперативными и предназначены для немедленного реагирования на текущую ситуацию. В большинстве они ориентированы на неподготовленного пользователя, потому имеют упрощенный интерфейс, содержат базовый набор предлагаемых возможностей, фиксированные формы представления информации и перечень решаемых задач и неспособны ответить на все вопросы, которые могут возникнуть при принятии решений. Такие системы основаны на типичных запросах, число которых относительно невелико; отчеты, полученные в результате таких запросов, представляются в максимально удобном виде. Результатом работы такой системы, как правило, являются многостраничные отчеты, которые нельзя изменить без привлечения программиста.

К динамическим DSS относят многофункциональные системы анализа и исследования данных. Они предполагают глубокую проработку данных, которую можно использовать в процессе принятий решений, и ориентированы на обработку неожиданных (ad hoc) запросов.

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.

Обобщенная структурная схема информационной СППР представлена на рис. 2.

Поддержка принятия решений на основе накопленных данных может выполняться в трех базовых сферах.

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).

Определяет задачи поиска закономерностей в накопленной информации, построение моделей и правил, которые объясняют найденные аномалии и/или прогнозируют развитие некоторых процессов.

Рис. 2. Структура информационной СППР

1.4. Хранилище данных

Информационная технология складирования данных (data warehousing) родилась в недрах компании IBM и была окончательно сформулирована Б. Инмоном и Р. Кимбаллом в 90-х гг. прошлого столетия [2–3] как метод решения информационно-аналитических задач в области принятия и поддержки решений. Возникнув на стыке технологии баз данных, систем поддержки принятия решений и компьютерного анализа данных, в дальнейшем концепция складирования данных претерпела эволюцию, поскольку оказалась пригодной для широкого круга приложений в бизнесе, науке и технологии.

Хранилище данных – это предметно-ориентированное, привязанное ко времени и неизменяемое собрание данных для поддержки процесса принятия управляющих решений [4].

Основные характеристики хранилищ данных:

− ориентация на предметную область. Учитывает специфику предметной области (клиенты, товары, продажи), а не прикладных областей деятельности (выписка счетов, контроль запасов, продажа товаров);

− интегрированность и внутренняя непротиворечивость. Поскольку данные в хранилище поступают из разных источников, необходимо привести их к единому формату;

− наличие исторических данных с привязкой ко времени (учет хронологии);

− неизменяемость. Данные не обновляются в оперативном режиме, а лишь регулярно пополняются;

− поддержка высокой скорости получения данных из хранилища;

− ориентация на проведение анализа и принятие стратегических решений;

− полнота (хранит как подробные сведения, так и частично и полностью обобщенные данные) и достоверность хранимых данных;

− поддержка качественного процесса пополнения данных;

− обслуживание относительно малого количества работников руководящего звена и аналитиков.

Хранилище данных (ХД) является местом складирования собираемых в системе данных и информационным источником для решения задач анализа данных и принятия решений. Как правило, объем информации в ХД является достаточно большим. Упрощенно можно сказать, что хранилище данных управляет данными, которые были собраны как из OLTP-систем, так и из внешних источников данных, и которые длительный период времени хранятся в системе.

Одной из главных целей создания систем складирования данных является их ориентация на анализ накопленных данных, т. е. структуризация данных в ХД должна быть выполнена таким образом, чтобы данные эффективно использовались в аналитических приложениях.

Можно выделить следующие причины для разделения данных систем складирования данных и OLTP-систем:

− различие целевых требований к системам складирования данных и OLTP-системам;

− необходимость собирать данные в ХД из различных информационных источников, т. е. если данные генерируются в самой OLTP-системе, то для системы складирования данных в большинстве случаев данные генерируются вне ее;

− данные, попадая в ХД, остаются в большинстве случаев неизменными;

− данные в ХД сохраняются длительное время.

Устройства для хранения данных также называются хранилищами. Тип используемого хранилища зависит от типа данных и их применения:

портативные носители (карты памяти), DVD, HDD и SSD, внешние дисковые массивы и ленты, RAID-массивы, корпоративные устройства памяти (Direct Attached Storage, Storage Area Network, Network Attached Storage) и др. Более подробно они описаны в [5].

Данные в ХД хранятся как в детализированном, так и в агрегированном виде. **Детализированные** данные поступают непосредственно из источников данных и соответствуют элементарным событиям, регистрируемым OLTP-системами. Такими данными могут быть ежедневные продажи, количество произведенных изделий и т. д. Это неделимые значения, попытка дополнительно детализировать которые лишает их логического смысла.

Многие задачи анализа (например, прогнозирование) требуют использования данных определенной степени обобщения. Например, суммы продаж, взятые по дням, могут дать очень неравномерный ряд данных, что затруднит выявление характерных периодов, закономерностей или тенденций. Однако, если обобщить эти данные в пределах недели или месяца и взять сумму, среднее, максимальное и минимальное значения за соответствующий период, то полученный ряд может оказаться более информативным. Процесс обобщения детализированных данных называется агрегированием, а сами обобщенные данные – **агрегированными.**

Поскольку один и тот же набор детализированных данных может породить несколько наборов агрегированных данных с различной степенью обобщения, объем ХД возрастает, иногда существенно. Часто это приводит к «взрывному», неконтролируемому росту ХД и вызывает серьезные технические проблемы. Однако, если бы агрегированные данные не содержались в ХД, а вычислялись в процессе выполнения запросов, время выполнения запроса увеличилось бы в несколько раз.

Для управления хранилищем данных используются **метаданные.** Слово «метаданные» (от греч. *meta* и лат. *data*) буквально переводится как «данные о данных». Метаданные в широком смысле необходимы для описания значения и свойств информации с целью лучшего ее понимания, использования и управления ею.

Метаданные – высокоуровневые средства отражения информационной модели и описания структуры данных, используемой в ХД. Метаданные должны содержать описание структуры данных хранилища и структуры данных импортируемых источников. Метаданные хранятся отдельно от данных в репозитарии метаданных.

Существуют два уровня метаданных – **технические** и **бизнес-данные**.

Технический уровень содержит метаданные, необходимые для обеспечения функционирования хранилища (статистика работы приложений, описание модели данных, названия таблиц и т. д.).

Бизнес-метаданные представляют собой описание предметной области, для работы в которой создается аналитическая система или ХД. Бизнес-метаданные описывают сущности, информация о которых содержится в ХД, – атрибуты объектов и их возможные значения и т. д.

Бизнес-метаданные образуют так называемый семантический слой. Пользователь оперирует близкими ему терминами предметной области: товар, клиент, продажи, покупки и т. д., а семантический слой транслирует бизнес-термины в низкоуровневые запросы к данным в хранилище.

Процесс разработки ХД весьма трудоемок, некоторые организации затрачивают на него несколько месяцев и даже лет, а также вкладывают значительные финансовые средства. Основными задачами, которые требуется решить в процессе разработки ХД, являются:

− выбор структуры хранения данных, обеспечивающей высокую скорость выполнения запросов и минимизацию объема оперативной памяти;

− первоначальное заполнение и последующее пополнение хранилища;

− обеспечение единой методики работы с разнородными данными и создание удобного интерфейса пользователя.

Круг задач интеллектуального анализа данных весьма широк, а сами задачи существенно различаются по уровню сложности. Поэтому, в зависимости от специфики решаемых задач и уровня их сложности, архитектура ХД и модели данных, используемых для их построения, могут различаться.

Обобщенная концептуальная модель ХД представлена на рис. 3.

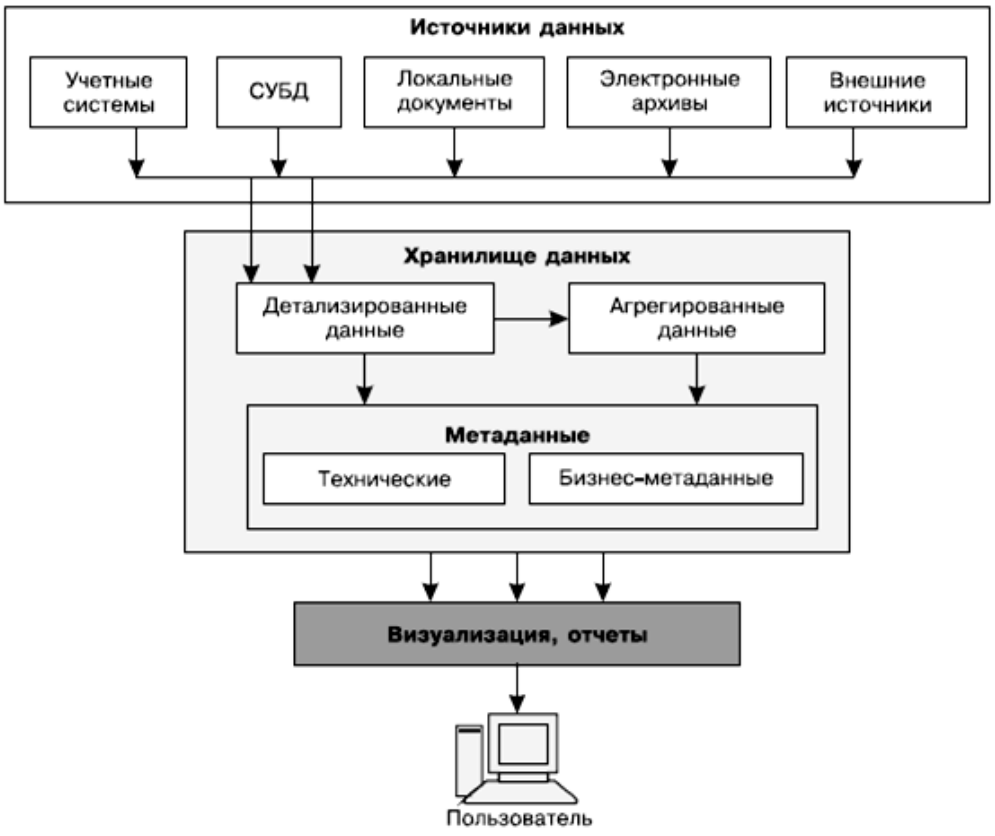


Рис. 3. Обобщенная концептуальная схема хранилища данных

Согласно рис. 3, данные извлекаются из различных источников и загружаются в ХД, которое содержит как собственно данные, представленные в соответствии с некоторой моделью, так и метаданные.

1.5. Извлечение данных (ETL)

Для того чтобы заставить ХД заработать, необходимо не просто обеспечить взаимодействие многих источников данных – важно тщательно спланировать это взаимодействие. Поэтому процессы извлечения, преобразования и загрузки данных играют важную роль в создании и эксплуатации ХД.

Под аббревиатурой **ETL** (*extraction, transformation, loading* – извлечение, преобразование и загрузка данных) понимается комплекс методов, реализующих процесс переноса исходных данных из различных источников в аналитическое приложение или поддерживающее его хранилище данных.

Процесс ETL реализуется либо путем разработки приложения ETL, либо путем создания комплекса встроенных программных процедур, либо – использования ETL-инструментария. Как правило, ETL-приложения используются при переносе данных внешних источников в ХД систем бизнес-аналитики. Поэтому организация процесса ETL является составной частью проекта разработки практически любого ХД.

Извлечение данных из разнотипных источников и перенос их в ХД с целью дальнейшей аналитической обработки связаны с рядом проблем:

− исходные данные расположены в источниках самых разнообразных типов и форматов, созданных в различных приложениях, и, кроме того, могут использовать различную кодировку. Для решения задач анализа данные должны быть преобразованы в единый универсальный формат, который поддерживается ХД и аналитическим приложением;

− данные в источниках обычно излишне детализированы, тогда как для решения задач анализа в большинстве случаев требуются обобщенные данные;

− исходные данные, как правило, являются «грязными» (отсутствующие, неточные или бесполезные данные), что мешает их корректному анализу.

Обобщенная структура процесса ETL представлена на рис. 4.

Извлечение данных

На этом этапе данные извлекаются из одного или нескольких источников и подготавливаются к преобразованию. Из источников должны извлекаться не только сами данные, но и информация, описывающая их структуру, из которой будут сформированы метаданные.

Процесс извлечения данных из источников данных можно разбить на следующие основные типы:

− извлечение данных при помощи приложений, основанных на выполнении SQL-команд;

− извлечение данных при помощи встроенных в СУБД механизмов импорта/экспорта данных (быстрее, чем через SQL);

− извлечение данных с помощью специально разработанных приложений.

Процесс извлечения данных может выполняться ежедневно, еженедельно или, в редких случаях, ежемесячно. Существует целый класс систем бизнес-аналитики, которые требуют извлечения данных в режиме реального времени: например, системы, анализирующие биржевые операции.

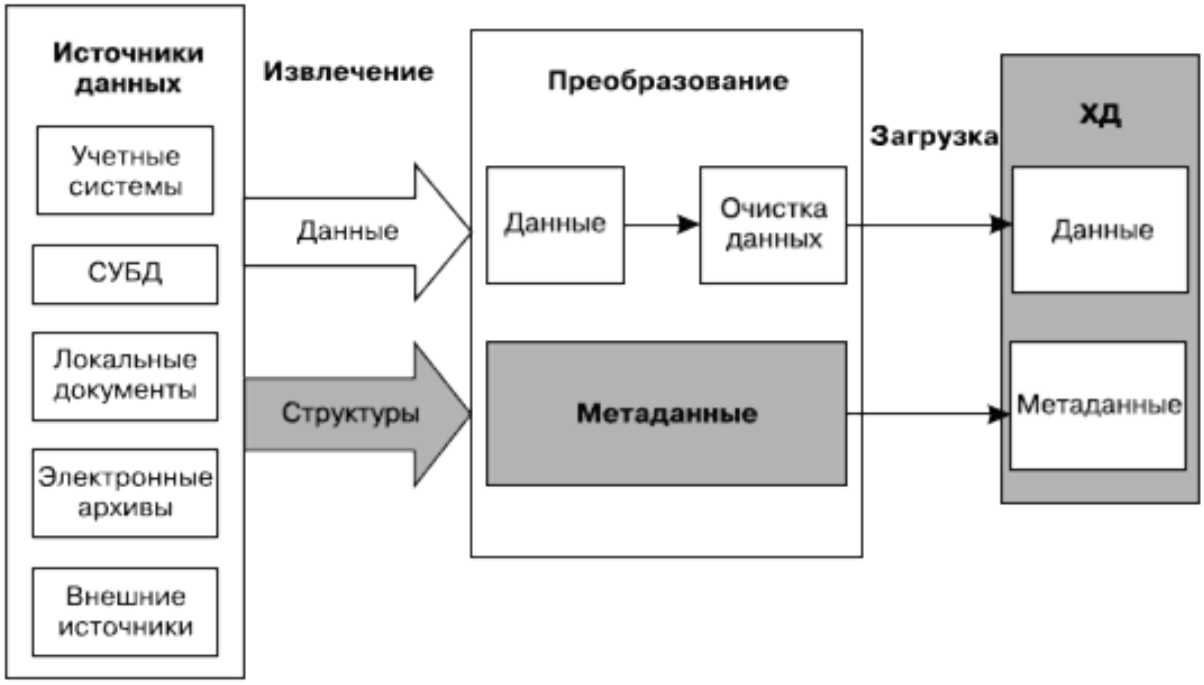


Рис. 4. Обобщенная структура процесса ETL

Преобразование данных

Процесс преобразования данных источников включает в себя следующие основные действия:

− преобразование типов данных;

− преобразования, связанные с нормализацией схемы данных;

− преобразования ключей;

− преобразования, связанные с обеспечением качества данных в ХД (очистка данных).

Очистка данных

Данные в ХД должны быть:

− точными – должны содержать правильные количественные значения метрик;

− полными – пользователи ХД должны знать, что имеют доступ ко всем релевантным данным;

− согласованными – агрегаты должны точно соответствовать подробным данным;

− уникальными – одни и те же объекты предметной области должны иметь одинаковые наименования и идентифицироваться в ХД одинаковыми ключами;

− актуальными – пользователи ХД должны знать, с какой частотой данные обновляются (т. е. на какую дату данные действительны).

Очистку данных можно разделить на следующие типы:

− конвертация и нормализация данных (согласование форматов данных, например, даты),

− обнаружение одинаковых имен атрибутов, с различными по смыслу значениями;

− стандартизация написания имен, представления адресов, устранение дубликатов;

− замещение кодов значениями (например, почтового индекса наименованием населенного пункта);

− исключение ненужных атрибутов (например, комментариев);

− стандартизация наименований таблиц, индексов и т. д.

Загрузка данных

Основная цель процесса загрузки данных состоит в быстрой загрузке данных в ХД.

Загрузка данных, основанная на использовании команд обновления SQL, является медленной, поэтому загрузка с помощью встроенных в СУБД средств импорта/экспорта является предпочтительной.

При загрузке данных должна быть гарантирована ссылочная целостность данных, а агрегаты должны быть построены и загружены одновременно с подробными данными.

1.6. Архитектура хранилищ данных

Под архитектурой ХД понимают совокупность программно-аппаратных компонент, совокупность технологических и организационных решений, предпринимаемых для создания, разработки и функционирования ХД, т. е. выбор аппаратного и программного обеспечения, выбор способов взаимодействия программно-аппаратных компонент, выбор способа решения проектной задачи по разработке и созданию ХД.

<table>
<tr><td rowspan="6">Документы</td><td rowspan="6">ETL</td><td>Ведение НСИ</td><td rowspan="6">SRD</td><td>Тематическая витрина данных</td><td>Сценарный анализ</td></tr>
<tr><td>Ведение метаданных</td><td>Региональная витрина данных</td><td>Статистический анализ</td></tr>
<tr><td rowspan="2">Транзакционные системы</td><td>Центральное хранилище данных</td><td>Витрина данных подразделения</td><td rowspan="2">Многомерный анализ</td></tr>
<tr><td rowspan="2">Оперативный склад данных</td><td rowspan="2">Прикладная витрина данных</td></tr>
<tr><td rowspan="2">Файлы</td><td rowspan="2">Отчетность</td></tr>
<tr><td>Архивы</td><td>Зоны временного хранения</td><td>Функциональная витрина данных</td><td>Планирование</td></tr>
<tr><td>Источники данных</td><td>Извлечение, преобразование, загрузка</td><td>Хранение данных</td><td>Распределение данных</td><td>Предоставление данных</td><td>Бизнес-приложения</td></tr>
</table>

Рис. 5. Шесть уровней архитектуры хранилища данных

Первый уровень представлен источниками данных, в качестве которых выступают транзакционные и унаследованные системы[2], архивы, разрозненные файлы известных форматов, документы (например, MS Office), а также любые иные источники структурированных данных.

На втором уровне размещается система ETL. Программно-аппаратный комплекс, на котором реализована система ETL, должен обладать значительной пропускной способностью, но еще важнее для него – это высокая вычислительная производительность. Поэтому лучшие из систем ETL способны обеспечивать высокую степень параллелизма вычислений, и даже работать с кластерами и вычислительными гридами[3].

Роль следующего уровня – надежное, защищенное от несанкционированного доступа, хранение данных. На этом уровне должны размещаться также системы ведения метаданных и **нормативно-справочной информации** (НСИ). В состав НСИ входят словари сокращений, справочники, классификаторы, нормативы, идентификаторы и кодификаторы. **Оперативный склад данных** (Operational Data Store) необходим тогда, когда требуется как можно более оперативный доступ к пусть неполным, не до конца согласованным данным, доступным с наименьшей возможной задержкой.

[2] Информационные системы, которые используются в организации уже значительное время и имеют ограничения по формату взаимодействия или доступа к данным.

[3] Грид-вычисления (англ. *grid* – решетка, сеть) – это форма распределенных вычислений, в которой «виртуальный суперкомпьютер» представлен в виде кластеров, соединенных с помощью сети, слабосвязанных гетерогенных компьютеров, работающих вместе для выполнения огромного количества операций.

Под **зонами временного хранения** (Staging area) понимаются области хранения данных, предназначенные для выполнения операций внешними пользователями или системами в соответствии с бизнес-требованиями обработки данных. Они нужны для реализации специфических бизнес-процессов (например, когда перед загрузкой данных контролер данных должен просмотреть их и дать разрешение на их загрузку в хранилище). Их выделение в отдельный компонент ХД необходим, так как для этих зон требуется создание дополнительных средств администрирования, мониторинга, обеспечения безопасности и аудита.

Информационные системы на уровне распределения данных выполняют задачи выборки, реструктуризации и доставки данных (**SRD** – Sample, Restructure, Deliver). В отличие от ETL, SRD выполняет выборку из единого хранилища данных, при этом система имеет дело с уже очищенными данными, структуры которых должны быть приведены в соответствие с требованиями различных приложений. SRD должно доставить данные в различные витрины в соответствии с правами доступа, графиком доставки и требованиями к составу информации.

Уровень предоставления данных предназначен для разделения функций хранения и функций обслуживания различных задач. **Витрины (киоски) данных** (data marts) – это срез ХД, представляющий собой массив тематической, узконаправленной информации, ориентированный, например, на пользователей одного департамента. Витрины данных должны иметь структуры данных, максимально отвечающие потребностям обслуживаемых задач. Поскольку не существует универсальных структур данных, оптимальных для любой задачи, витрины данных следует группировать по территориальным, тематическим, организационным, прикладным, функциональным и иным признакам.

Концепция витрин данных имеет ряд несомненных достоинств:

− аналитики видят и работают только с теми данными, которые им реально нужны;

− для реализации витрин данных не требуется мощная вычислительная техника;

− относительно небольшой объем хранимых данных, на организацию и поддержку которых не требуется значительных затрат;

− корпоративная информационная система может эффективно наращиваться за счет добавления новых витрин данных;

− использование витрин данных позволяет снизить нагрузку на централизованное ХД.

Уровень бизнес-приложений представлен сценарными расчетами и статистическим анализом, многомерным анализом, средствами планирования и подготовки отчетности и проч.

1.6.1. *Реляционные хранилища данных*

В начале 1970-х гг. англо-американский ученый Э. Кодд разработал реляционную модель организации хранимых данных, которая положила начало новому этапу эволюции СУБД. Применение реляционной модели при создании ХД в ряде случаев позволяет получить целый ряд преимуществ, особенно в части эффективности работы с большими массивами данных и использования памяти компьютера.

В отличие от OLTP систем, с которыми работают приложения, изменяющие данные, РХД проектируются таким образом, чтобы добиться минимального времени выполнения запросов на чтение (у оперативных же БД чаще всего минимизируется время выполнения запросов на изменение данных). Обычно данные копируются в хранилище из оперативных БД, согласно определенному расписанию.

В основе технологии РХД лежит принцип, в соответствии с которым измерения хранятся в плоских таблицах так же, как и в обычных реляционных СУБД, а факты (агрегируемые данные) – в отдельных специальных таблицах этой же базы данных. При этом таблица фактов является основой для связанных с ней таблиц измерений. Она содержит количественные характеристики объектов и событий, совокупность которых предполагается в дальнейшем анализировать.

Типичная структура РХД существенно отличается от структуры обычной реляционной БД. Как правило, эта структура денормализована (это повышает скорость выполнения запросов) и может допускать избыточность данных. Типичная структура РХД приведена на рис. 6.

Основные составляющие этой структуры – таблица фактов и таблицы измерений.

Факты – это данные, количественно описывающие бизнес-процесс, непрерывные по своему характеру, т. е. они могут принимать бесконечное множество значений. Примеры фактов — цена товара, их количество, сумма продаж, зарплата сотрудников и т. д.

Таблица фактов (в примере на рис. 6 она называется Sales_Fact) – это основная таблица ХД. Как правило, в нее входят сведения об объектах или событиях, совокупность которых будет в дальнейшем анализироваться. Помимо этого таблица фактов содержит одно или несколько числовых полей, на основании которых в процессе выполнения аналитических запросов получаются агрегатные данные.

Измерения – это категориальные атрибуты, наименования и свойства объектов, участвующих в некотором бизнес-процессе. Измерения могут быть и числовыми, если какой-либо категории (например, наименованию товара) соответствует числовой код, но в любом случае это данные дискретные, т. е. принимающие значения из ограниченного набора. Измерения качественно описывают исследуемый бизнес-процесс.

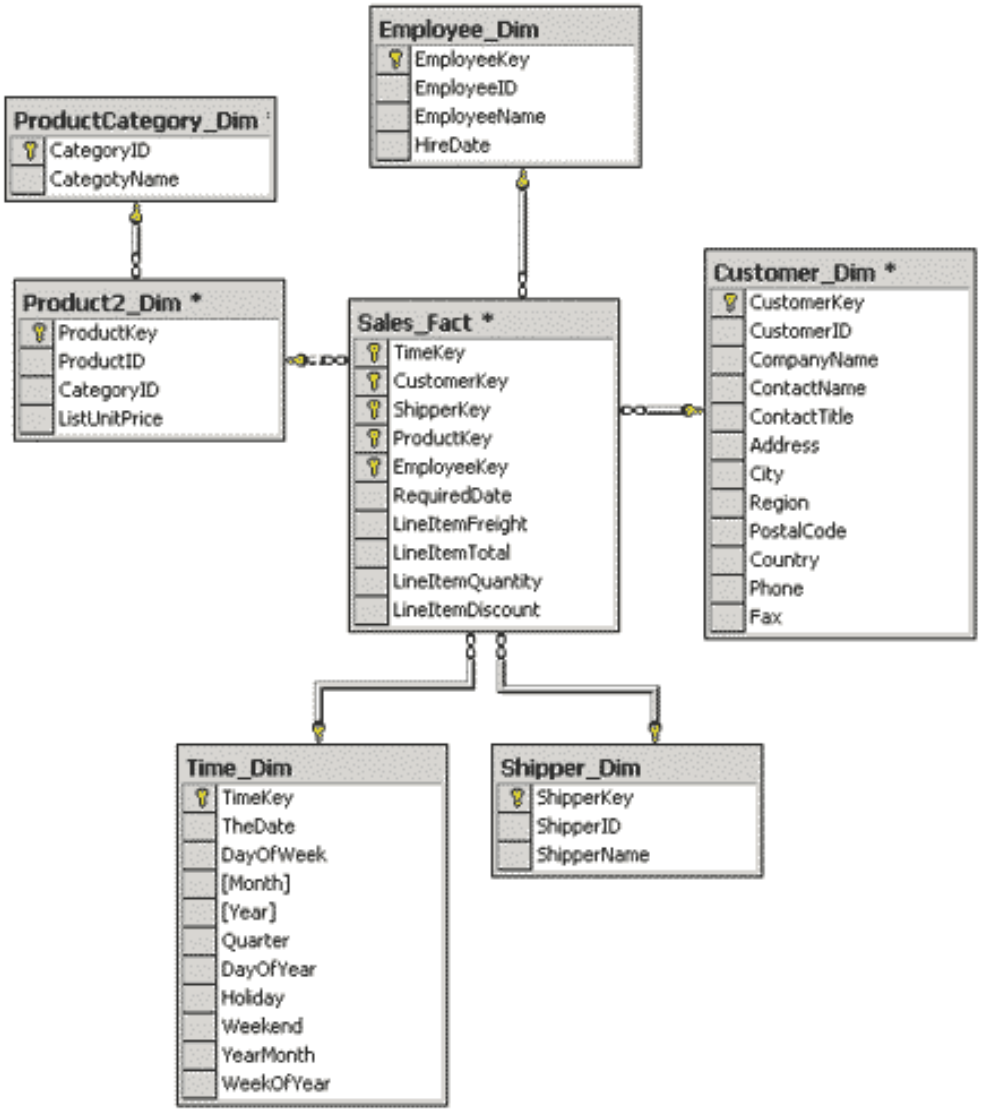


Рис. 6. Пример структуры хранилища данных.

Таблицы измерений содержат неизменяемые либо редко изменяемые данные. Каждая таблица измерений должна находиться в отношении «один ко многим» с таблицей фактов.

Отметим, что скорость роста таблиц измерений должна быть незначительной по сравнению со скоростью роста таблицы фактов; например, новая запись в таблицу измерений, характеризующую товары, добавляется только при появлении нового, не продававшегося ранее товара.

На логическом уровне различают две схемы построения РХД — «звезда» и «снежинка».

При использовании **схемы «звезда»** (рис. 7) центральной является таблица фактов, с которой связаны все таблицы измерений. Таким образом, информация о каждом измерении располагается в отдельной таблице, что упрощает их просмотр, а саму схему делает логически прозрачной и понятной пользователю.

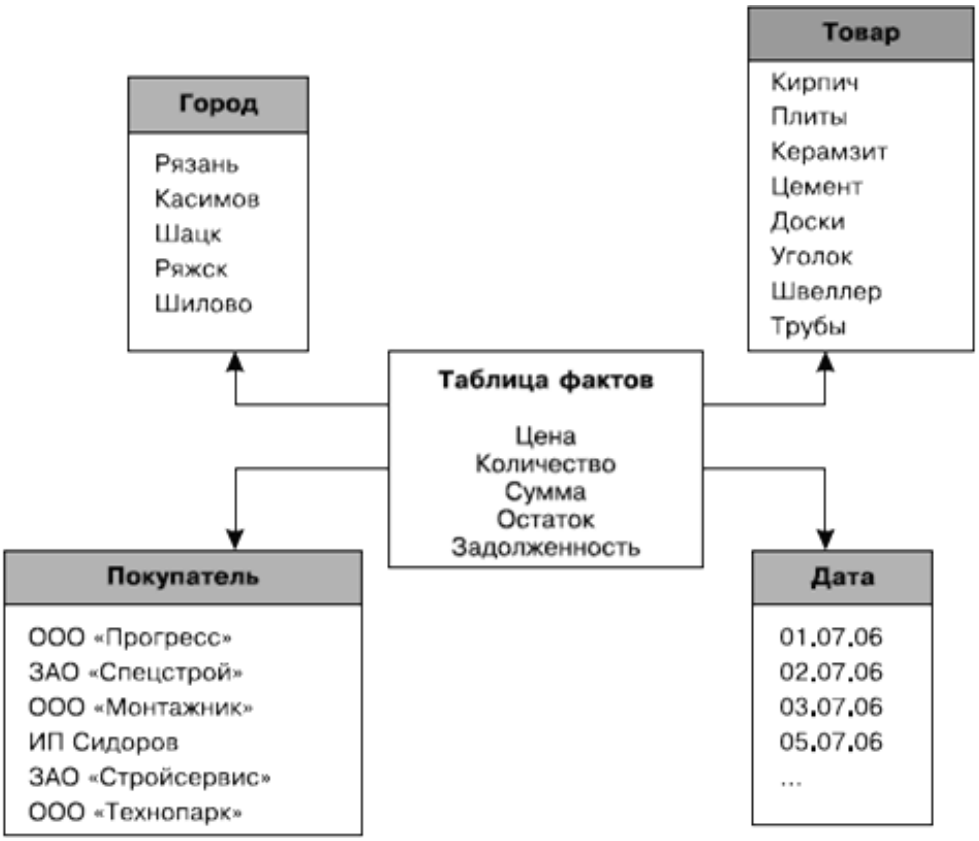


Рис. 7. Схема построения РХД «звезда»

Однако размещение всей информации об измерении в одной таблице оказывается не всегда оправданным. Например, если продаваемые товары объединены в группы (имеет место иерархия), то придется показать, к какой группе относится каждый товар, что приведет к многократному повторению названий групп. Это не только вызовет рост избыточности, но и повысит вероятность возникновения противоречий.

Для более эффективной работы с иерархическими измерениями была разработана модификация схемы «звезда», которая получила название «**снежинка**». Главной особенностью схемы «снежинка» является то, что информация об одном измерении может храниться в нескольких связанных таблицах (рис. 8). То есть, если хотя бы одна из таблиц измерений имеет одну или несколько связанных с ней других таблиц измерений, в этом случае будет применяться схема «снежинка».

Основное функциональное отличие схемы «снежинка» от схемы «звезда» – это возможность работы с иерархическими уровнями, определяющими степень детализации данных.

К преимуществам схемы «звезда» можно отнести:

− простоту и логическую прозрачность модели;

− более простую процедуру пополнения измерений, поскольку приходится работать только с одной таблицей.

Недостатками схемы «звезда» являются:

− медленная обработка измерений, поскольку одни и те же значения измерений могут встречаться несколько раз в одной и той же таблице;

− высокая вероятность возникновения несоответствий в данных (в частности, противоречий), например, из-за ошибок ввода.

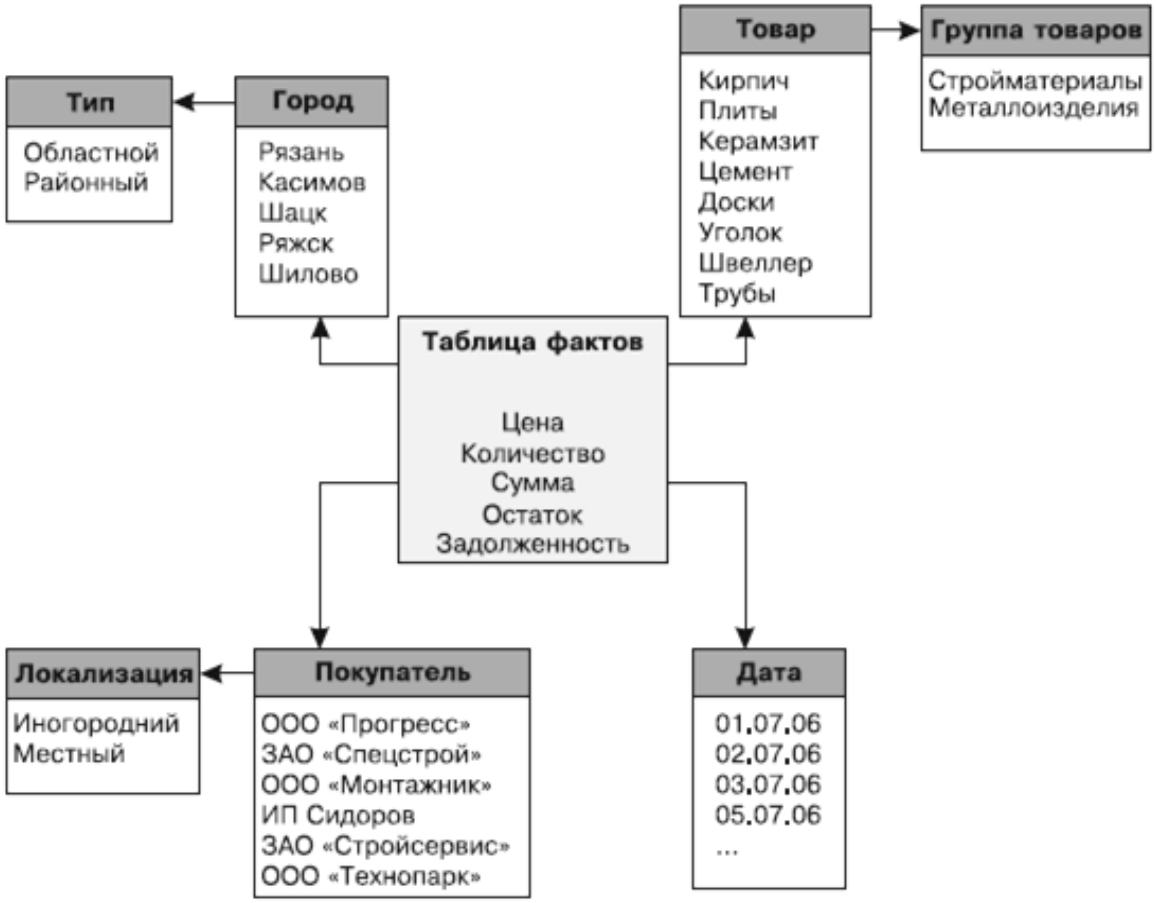


Рис. 8.Схема построения РХД «снежинка»

Преимуществами схемы «снежинка» являются:

− она ближе к представлению данных в многомерной модели;

− процедура загрузки из РХД в многомерные структуры более эффективна и проста, поскольку загрузка производится из отдельных таблиц;

− намного ниже вероятность появления ошибок несоответствия данных;

− большая, по сравнению со схемой «звезда», компактность представления данных, поскольку все значения измерений упоминаются только один раз.

Недостатки схемы «снежинка»:

− достаточно сложная для реализации и понимания структура данных;

− усложненная процедура добавления значений измерений.

Основные преимущества РХД следующие:

− объем хранимых данных практически неограничен;

− поскольку реляционные СУБД лежат в основе построения многих систем OLTP, которые обычно являются главными источниками данных для ХД, использование реляционной модели позволяет упростить процедуру загрузки и интеграции данных в хранилище;

− нет необходимости выполнять сложную физическую реорганизацию хранилища при добавлении новых измерений данных;

− обеспечиваются высокий уровень защиты данных и широкие возможности разграничения прав доступа.

Главный недостаток РХД – невысокая производительность, из-за большого числа таблиц агрегатов.

Таким образом, выбор реляционной модели при построении ХД целесообразен:

− если значителен объем хранимых данных;

− иерархия измерений несложная (не много агрегированных данных);

− требуется частое изменение размерности данных (можно ограничиться добавлением новых таблиц).

1.6.2. *Многомерные хранилища данных*

Большинство бизнес-процессов описывается множеством атрибутов. Если собрать всю информацию в таблицу, то она окажется сложной для визуального анализа. Более того, она может оказаться избыточной, так как в плоской таблице хранятся многомерные данные.

По Кодду **многомерное концептуальное представление** – множественная перспектива, состоящая из нескольких независимых измерений, вдоль которых могут быть проанализированы определенные совокупности данных. Одновременный анализ по нескольким измерениям определяется как **многомерный анализ.**

Основное назначение многомерных хранилищ данных (МХД) – поддержка систем, ориентированных на аналитическую обработку данных, поскольку такие хранилища лучше справляются с выполнением сложных нерегламентированных запросов.

В основе многомерного представления данных лежит их разделение на две группы – измерения и факты. **Измерение** – это множество объектов одного или нескольких типов, организованных в виде иерархической структуры и обеспечивающих информационный контекст числового показателя. Измерение принято визуализировать в виде ребра (оси) многомерного куба. В качестве одного из измерений используется время.

Факт – это числовая величина, которая располагается в ячейках гиперкуба. Они количественно характеризуют процесс.

Каждому набору значений измерений (например, «дата – товар – покупатель») будет соответствовать **ячейка** с фактами (cell), связанными с данным набором. Таким образом, между объектами бизнес-процесса и их

числовыми характеристиками будет установлена однозначная связь. Иногда вместо термина «ячейка» используется термин «**показатель**» (measure).

Принцип организации многомерного куба поясняется на рис. 9. В ячейке 1 будут располагаться факты, относящиеся к продаже цемента ООО «Спецстрой» 3 ноября, в ячейке 2 – к продаже плит ЗАО «Пирамида» 6 ноября, а в ячейке 3 – к продаже плит ООО «Спецстрой» 4 ноября.

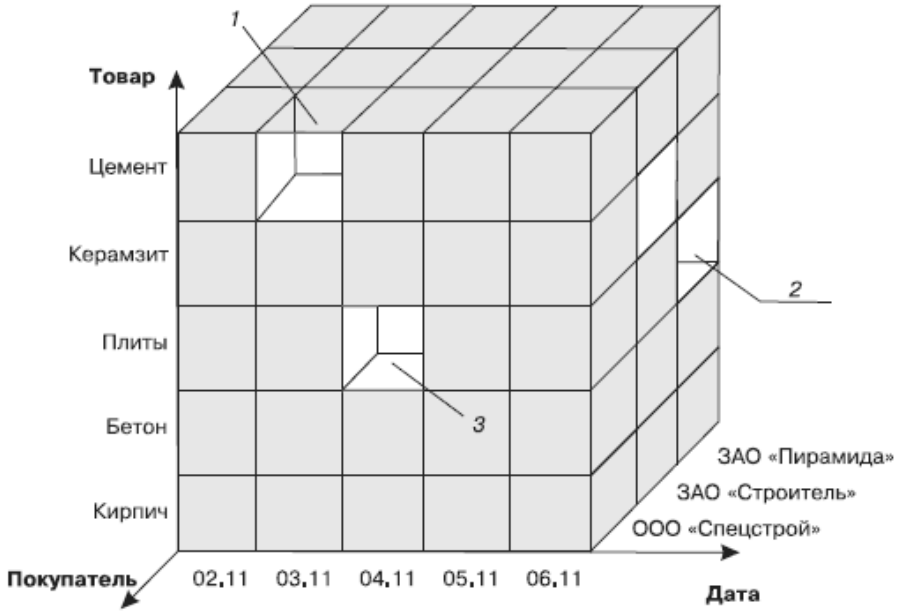


Рис. 9. Принцип организации многомерного куба

Многомерный взгляд на измерения Дата, Товар и Покупатель представлен на рис. 10. Фактами в данном случае могут быть Цена, Количество, Сумма. Тогда выделенный сегмент будет содержать информацию о том, сколько плит, на какую сумму и по какой цене приобрела фирма ЗАО «Строитель» 3 ноября.

Рис. 10. Измерения и факты в многомерном кубе

Таким образом, информация в многомерном хранилище данных является логически целостной. Это уже не просто наборы строковых и числовых значений, которые в случае реляционной модели нужно получать из различных таблиц, а целостные структуры типа «кому, что и в каком количестве было продано на данный момент времени».

Преимущества многомерного подхода:

− представление данных в виде многомерных кубов более наглядно, чем совокупность нормализованных таблиц реляционной модели, структуру которой представляет только администратор БД;

− более широкие возможности построения аналитических запросов к системе;

− в некоторых случаях использование многомерной модели позволяет значительно уменьшить продолжительность поиска, обеспечивая выполнение аналитических запросов практически в режиме реального времени. Это связано с тем, что агрегированные данные вычисляются предварительно и хранятся в многомерных кубах вместе с детализированными, поэтому тратить время на вычисление агрегатов при выполнении запроса уже не нужно.

Недостатки МХД:

− для реализации требуется больший объем памяти;

− многомерная структура труднее поддается модификации; при необходимости встроить еще одно измерение, требуется выполнить физическую перестройку всего многомерного куба.

Таким образом, применение МХД целесообразно только в тех случаях, когда объем используемых данных сравнительно невелик, а сама многомерная модель имеет стабильный набор измерений.

1.6.3. *Гибридные хранилища данных*

Гибридные хранилища (ГХД) сочетают высокую производительность, характерную для многомерной модели, и возможность хранить сколь угодно большие массивы данных, присущую реляционной модели.

Главным принципом построения ГХД является то, что детализированные данные хранятся в РХД, а агрегированные – в МХД.

Если данные, поступающие из OLTP-системы, имеют большой объем (несколько десятков тысяч записей в день и более) и высокую степень детализации, а для анализа используются в основном обобщенные данные, гибридная архитектура хранилища оказывается наиболее подходящей.

Преимущества ГХД:

− хранение данных в реляционной структуре делает их в большей степени системно независимыми, что особенно важно при использовании в управлении предприятием экономической информации;

− реляционная структура формирует устойчивые и непротиворечивые опорные точки для многомерного хранилища;

− поскольку РХД поддерживает актуальность и корректность данных, оно обеспечивает очень надежный транспортный уровень для доставки информации в многомерное хранилище;

− построение OLAP-куба выполняется по запросу. Такой подход позволяет избежать взрывного роста данных. При этом можно достичь оптимального времени исполнения клиентских запросов.

Недостатком гибридной модели является усложнение администрирования ХД из-за более сложного регламента его пополнения, поскольку при этом необходимо согласовывать изменения в реляционной и многомерной структурах.

1.7. Управление жизненным циклом информации

Жизненный цикл информации – это изменение ценности информации с течением времени. В момент создания данные могут обладать максимальной ценностью и часто использоваться. Со временем они используются все реже и становятся менее значимыми. Часть данных со временем превращается в ссылочные. Это происходит тогда, когда прекращается их модификация, но они все еще используются, например, для создания отчетов. Понимание жизненного цикла информации помогает установить соответствующую инфраструктуру хранения данных.

Например, в заказе на покупку ценность информации меняется с момента размещения заказа до истечения срока гарантии. В момент получения заказа и его обработки значимость информации максимальна. После выполнения заказа данные по заказу или клиенту перестают быть востребованными. Компания может перенаправить эти данные на более дешевое вспомогательное запоминающее устройство с более низким уровнем доступности до тех пор, пока не возникнет обращение к гарантийным обязательствам. По истечении срока гарантии компания может переместить информацию о заказе в архив.

По определению SNIA (Ассоциация производителей сетевых устройств хранения данных), **управление жизненным циклом информации** (Information Lifecycle Management – ILM) – это набор политик, процессов, практик, сервисов и инструментов, используемых для того, чтобы соотнести ценность информации с точки зрения бизнеса с наиболее подходящей и эффективной по стоимости инфраструктурой, начиная с момента создания информации и заканчивая ее размещением.

Таким образом, ILM – это упреждающая стратегия, позволяющая организации эффективно управлять данными на протяжении их жизненного

цикла в соответствии с заранее установленной политикой предприятия. ILM позволяет оптимизировать инфраструктуру хранения данных для максимальной отдачи вложений.

Актуальность данной задачи обусловлена целым рядом проблем, с которыми сегодня сталкивается предприятие:

− в настоящее время расходы на хранение составляют более 15 % ИТ-бюджетов;

− ежегодно объемы данных растут более чем на 50 %;

− в большинстве случаев дисковые устройства хранения используются менее чем на 50 %, 40 % из них являются избыточными;

− в мире существуют более 20 тыс. нормативных актов, включающих требования к хранению данных.

В рамках ILM предлагается классифицировать бизнес-информацию компании, прежде чем она попадет в инфраструктуру хранения. ILM предлагает уйти от управления данными и сфокусироваться на управлении информацией. С точки зрения данных, файл с текстом детективного романа, загруженный из электронной библиотеки, и контракт на многомиллионную сумму абсолютно равнозначны, но с точки зрения информации первый, что вполне естественно, не несет никакой ценности для компании.

На рис. 11 представлен график изменения ценности информации для бизнеса с течением времени по данным Enterprise Storage Group.

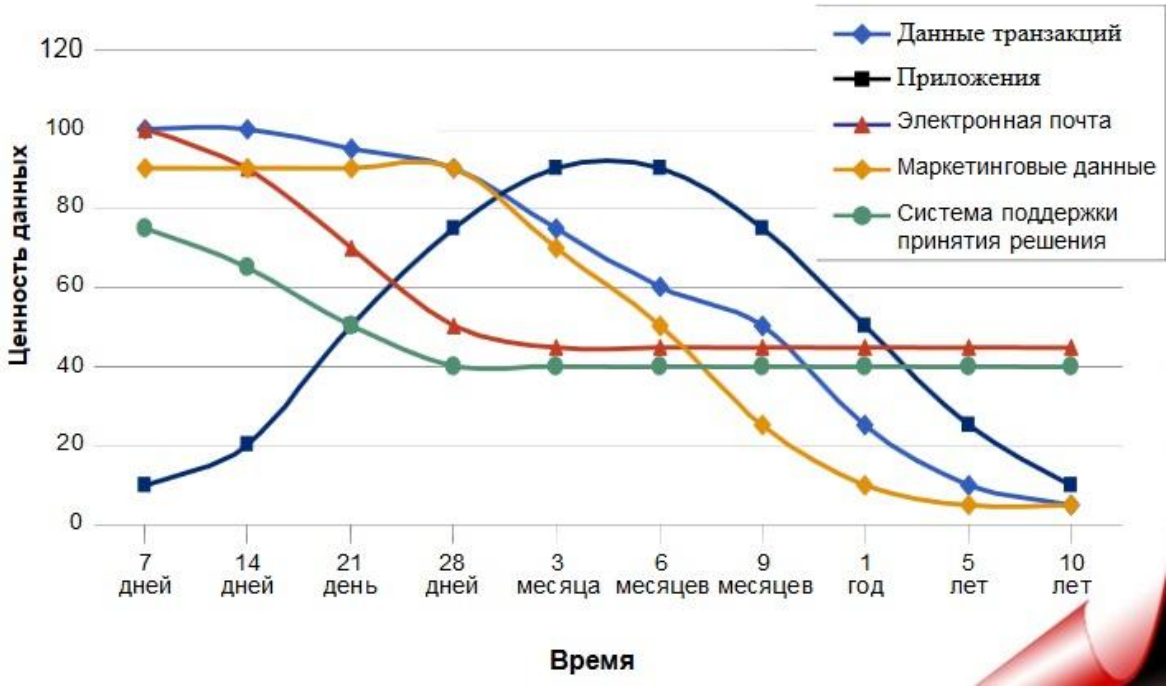


Рис. 11. Изменение ценности информации для бизнеса с течением времени

Стратегия ILM основана на многоуровневом хранении информации. **Многоуровневое хранение** – подход к определению различных уровней

хранения для снижения затрат на хранение. Каждый уровень имеет различные степени защиты, производительности, частоты доступа к данным и проч. Информация хранится и передается между уровнями, исходя из ее ценности с течением времени.

Процесс реализации ILM-стратегии состоит из четырех видов деятельности:

− классификация данных и приложений на базе правил и политики бизнеса;

− выполнение стратегии посредством механизмов управления информацией, начиная от создания данных и заканчивая их удалением;

− управление средой при использовании механизмов для снижения системной нагрузки;

− организация уровневого хранения ресурсов для их распределения по классам данных и хранения информации разных уровней ценности в соответствующем типе инфраструктуры.

Классическая схема классификации определяет четыре основных класса данных:

1) оперативные данные – оперативно используемые в бизнес-среде;

2) данные оперативного восстановления – данные, которые вышли из операционного использования, но могут быть восстановлены по требованию бизнеса;

3) архивные данные – окончательно вышли из использования бизнес-средой. Однако они не удаляются по причине регуляторных требований. Такие данные предоставляются по требованию контролирующих и государственных органов;

4) данные аварийного восстановления – необходимы для восстановления нормальной работоспособности бизнеса после серьезных аварий. Эти данные представляют собой минимальный объем критически важной информации.

В каждом конкретном случае количество типов данных может быть увеличено, если этого требует специфика бизнеса. В ходе проекта по классификации данных ставятся условия перехода данных из стадии в стадию, а также приемлемые сроки пребывания на каждой из стадий.

Для классов информации следует задать *уровень обслуживания* с точки зрения производительности (количество операций ввода/вывода в секунду IOPS), доступности (например, 99,999 %, ежедневный backup, ежечасное создание «снимков» – snapshot), катастрофоустойчивости или специальных требований, таких как WORM (Write Once Read Many – не стираемый архив). Также указывается политика жизненного цикла информации.

Пример ILM стратегии приведен в табл. 2.

В этой таблице учтена классификация по приложениям, файлам, а также определена ненужная информация (временные и дублированные файлы). Также указана политика по отношению к классам информации с точки зрения жизненного цикла: так, например, одни и те же файлы MS Excel созданные бухгалтерами, могут относиться к классу «критичных данных» или к классу «важных» в зависимости от того, был ли к ним доступ за последние 6 месяцев или нет.

Таблица 2

Пример ILM стратегии

Класс данных	Приложения/ файлы	Дата последнего доступа	Скорость доступа	Доступность	Политика ЖЦ
Критичные данные	Excel	< 6 мес.	40 мс	99,99 %	Перевести в класс важных, если не было доступа в течение 6 мес.
Важные	MS Outlook	> 30 дней	120 мс	99 %	Перевести в класс архив, если не было доступа в течение 30 дней и размер больше 20 Мб
Архив			3 мин	98 %	
Временные файлы	*.tmp	> 7 дней			Удалить
Дублированные файлы		> 30 дней			Удалить
Ненужные файлы	*.mp3				Удалить

Следующим этапом является инвентаризация имеющейся инфраструктуры, на основе которой будет построено решение, удовлетворяющее введенным классам информации. Инвентаризировать следует серверы, сеть хранения данных, системы хранения данных, ленточные библиотеки, а также наличие необходимых функций и решений, например, функции создания мгновенных «снимков» (snapshot) или синхронной репликации систем хранения данных.

Последним этапом является привязка классов информации к классам инфраструктуры. Важно отметить, что привязка является «живым» организмом в том смысле, что с течением времени информация меняет свою актуальность и в этой связи перемещается по классам инфраструктуры.

2. ОПЕРАТИВНЫЙ АНАЛИЗ ДАННЫХ

Технология комплексного многомерного анализа данных получила название OLAP (Online Analytical Processing). OLAP представляет собой методику оперативного извлечения нужной информации из больших массивов данных и формирования соответствующих отчетов. В основе концепции OLAP, лежит многомерное концептуальное представление данных.

Концепция OLAP была описана в 1993 г. Эдгаром Коддом, известным исследователем баз данных и автором реляционной модели данных [6]. Требования к OLAP-системам были разделены на четыре группы и названы «характеристиками»:

− **основные характеристики**: многомерность модели данных, интуитивные механизмы манипулирования данными, доступность данных, пакетное извлечение данных, архитектура «клиент-сервер», прозрачность, многопользовательская работа;

− **специальные характеристики**: обработка ненормализованных данных, хранение результатов отдельно от исходных данных, выделение отсутствующих данных, обработка отсутствующих значений;

− **характеристики построения отчетов**: гибкое построение отчетов, стабильная производительность при построении отчетов, автоматическое регулирование физического уровня;

− **управление размерностью**: общая функциональность, неограниченное число измерений и уровней агрегирования, неограниченные операции между данными различных измерений.

В 1995 г. на основе требований, изложенных Коддом, был сформулирован так называемый тест FASMI (Fast Analysis of Shared Multidimensional Information – быстрый анализ разделяемой многомерной информации).

Рассмотрим детально каждую из составляющих этой аббревиатуры [7].

Fast (быстрый). Это свойство означает, что OLAP-система должна обеспечивать ответ на запрос пользователя в среднем за пять секунд, при этом большинство запросов обрабатываются в пределах одной секунды, а самые сложные запросы должны обрабатываться в пределах двадцати секунд.

Analysis (аналитический). OLAP-система должна справляться с любым логическим и статистическим анализом, характерным для бизнес-приложений, и обеспечивать сохранение результатов в виде, доступном для конечного пользователя.

Средства анализа могут включать процедуры анализа временных рядов, распределения затрат, конверсии валют, моделирования изменений организационных структур и др.

Shared (разделяемый). Система должна предоставлять широкие возможности разграничения доступа к данным и одновременной работы многих пользователей.

Multidimensional (многомерный). Система должна обеспечивать концептуально многомерное представление данных, включая полную поддержку множественных иерархий.

Information (информация). Мощность различных программных продуктов характеризуется количеством обрабатываемых входных данных. Разные OLAP-системы имеют разную мощность: наиболее мощные из них могут оперировать, по крайней мере, в тысячу раз большим количеством данных по сравнению с самыми маломощными. При выборе OLAP-инструмента следует учитывать целый ряд факторов, включая дублирование данных, требуемую оперативную память, использование дискового пространства, эксплуатационные показатели, интеграцию с информационными хранилищами и т. п.

Главная идея OLAP 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 (рис. 12).

Рис. 12. Пример анализа данных гиперкуба

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. Следует отметить, что OLAP-функциональность

может быть реализована различными способами, начиная с простейших средств анализа данных в офисных приложениях и заканчивая распределенными аналитическими системами, основанными на серверных продуктах.

Существует три способа хранения данных в OLAP-системах или три архитектуры OLAP-серверов:

− MOLAP (Multidimensional OLAP) – исходные и агрегатные данные хранятся в многомерной базе данных;

− ROLAP (Relational OLAP) – исходные данные остаются в той же реляционной базе данных, где они изначально находились, агрегатные же данные помещают в специально созданные для их хранения служебные таблицы в той же базе данных;

− HOLAP (Hybrid OLAP) – исходные данные остаются в той же реляционной базе данных, где они изначально находились, а агрегатные хранятся в многомерной базе данных.

В широком смысле OLAP-системой называют любую DSS-систему, основанную на концепции хранилищ данных и обеспечивающую малое время выполнение (On-Line) аналитических запросов, независимо от того, используется ли многомерный анализ данных.

В процессе поиска и извлечения из гиперкуба нужной информации над его измерениями производится ряд действий, наиболее типичными из которых являются:

− сечение (срез);

− транспонирование;

− свертка;

− детализация.

Сечение

Для визуализации данных, хранящихся в кубе, применяются двумерные представления, имеющие сложные иерархические заголовки строк и столбцов. Пользователя редко интересуют все потенциально возможные комбинации значений измерений. Более того, он практически никогда не работает одновременно сразу со всем гиперкубом данных.

Подмножество гиперкуба, получившееся в результате фиксации значения одного или более измерений, называется срезом (Slice).

Сечение заключается в выделении подмножества ячеек гиперкуба при фиксировании значения одного или нескольких измерений. В результате сечения получается срез или несколько срезов, каждый из которых содержит информацию, связанную со значением измерения, по которому он был построен (рис. 13).

Манипулируя сечениями гиперкуба, пользователь всегда может получить информацию в нужном разрезе. Затем на основе построенных срезов может быть сформирована кросс-таблица (сводная таблица) и с ее помощью очень быстро получен необходимый отчет. Данная методика лежит в основе

технологии OLAP-анализа. Значения, «откладываемые» вдоль измерений, называются *метками* или членами измерений (members). Метки используются как для «разрезания» куба, так и для фильтрации выбираемых данных. Значения меток отображаются в двумерном представлении куба как заголовки строк и столбцов.

Исходный куб

	США	Канада	Мексика
Напитки	10 000	2000	1 000
Продукты питания	5000	500	250
Прочие товары	5000	500	250

Двумерный срез куба для одного факта

	США	Канада	Мексика
Январь	20 000	4000	2000
Февраль	30 000	6000	3000
Март	50 000	10 000	5000

Двумерный срез куба для нескольких фактов

	США	Канада	Мексика
Unit Sales	2000	400	200
Store Sales	30 000	6000	3000
Store Cost	10 000	2000	1000

Двумерный срез куба с несколькими измерениями на одной оси

	Январь			Февраль		
	США	Канада	Мексика	США	Канада	Мексика
Unit Sales	500	100	50	500	100	50
Store Sales	7500	1500	750	7500	1500	750
Store Cost	2500	500	250	2500	500	250

Рис. 13. Сечения гиперкуба

Транспонирование

Транспонирование (вращение) обычно применяется к плоским таблицам, полученным, например, в результате среза, и позволяет изменить порядок представления измерений таким образом, что измерения, отображавшиеся в столбцах, будут отображаться в строках, и наоборот. В ряде случаев транспонирование позволяет сделать таблицу более наглядной.

Свертка

Каждое измерение включает направления консолидации данных, состоящие из серии последовательных уровней обобщения (**уровней иерархии**), где каждый вышестоящий уровень соответствует большей степени агрегации данных по соответствующему измерению. В этом случае становится возможным произвольный выбор желаемого уровня детализации информации по каждому из измерений.

Например, метки измерения «Магазин» естественно объединяются в иерархию с уровнями: Мир, Страна, Штат, Город, Магазин.

В соответствии с уровнями иерархии вычисляются агрегатные значения, например, объем продаж для USA (уровень «Country») или для штата California (уровень «State»).

Детализация

Детализация – это процедура, обратная свертке; уровень обобщения данных уменьшается. При этом значения измерений более высокого иерархического уровня заменяются одним или несколькими значениями более низкого уровня, т. е. вместо наименований групп товаров отображаются наименования отдельных товаров.

3. ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ ДАННЫХ

3.1. Понятие Data Mining

Понятие Data Mining (добыча данных), появилось в 1978 г. **Data Mining** – это процесс поддержки принятия решений, основанный на поиске в «сырых» больших объемах данных скрытых (неочевидных), объективных и полезных на практике закономерностей, необходимых для принятия решений. Инструменты Data Mining могут находить такие закономерности самостоятельно и также самостоятельно строить гипотезы о взаимосвязях.

Неочевидные – найденные закономерности не обнаруживаются стандартными методами обработки информации или экспертным путем.

Объективные – найденные закономерности будут полностью соответствовать действительности, в отличие от экспертного мнения, которое всегда является субъективным.

Практически полезные – 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.

Традиционные методы анализа данных (статистические методы) и OLAP в основном ориентированы на проверку заранее сформулированных гипотез и на «грубый» разведочный анализ, составляющий основу 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.

Существует огромное количество областей применений Data Mining для решения задач в сфере бизнеса и науки.

Задача «Выдавать ли кредит клиенту?»

Без применения технологии Data Mining задача решается сотрудниками банка на основе их опыта, интуиции и субъективных представлений о том, какой клиент является благонадежным.

При помощи методов Data Mining задача 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

Классификация на «более выгодных» и «менее выгодных» клиентов. После определения наиболее выгодного сегмента есть смысл проводить более активную маркетинговую политику по привлечению клиентов именно среди найденной группы.

Задача прогнозирования остатка на счетах клиентов

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

Задача мерчендайзинга

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, выработки стратегии закупки товаров и их размещения на складах.

Задача медицинской диагностики

Традиционно для постановки медицинских диагнозов используются экспертные системы. С использованием Data Mining при помощи шаблонов можно разработать базу знаний для экспертной системы (симптомы пациента и его заболевание).

Web Mining

Web Mining применяет технологию Data Mining для анализа неструктурированной, неоднородной, распределенной и значительной по объему информации, содержащейся на Web-узлах.

Web Content Mining – автоматический поиск и извлечение информации из разнообразных источников Интернета, перегруженных «информационным шумом».

Web Usage Mining – обнаружение закономерностей в действиях пользователя Web-узла или группы пользователей. В результате сбора персонифицированных ретроспективных данных система накапливает определенные знания о клиенте и может рекомендовать ему определенные наборы товаров или услуг.

На основе информации обо всех посетителях сайта Web-система может выявить определенные группы посетителей и также рекомендовать им товары или же предлагать товары в рассылках.

Text Mining

Технология, разработанная на основе статистического и лингвистического анализа, предназначена для выполнения семантического (смыслового) анализа текстов, обеспечения навигации и поиска в неструктурированных текстах. Программы, реализующие эту задачу, должны оперировать естественным человеческим языком и при этом понимать семантику анализируемого текста.

Call Mining

Объединяет в себя распознавание речи, ее анализ и Data Mining. Цель – упрощение поиска в аудио-архивах, содержащих записи переговоров между операторами и клиентами. В технологии Call Mining разработано два подхода – на основе преобразования речи в текст и на базе фонетического анализа.

3.2. Основные задачи Data Mining

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

– обучение с учителем;

– обучение без учителя.

«Обучение с учителем» – один из способов машинного обучения, в ходе которого испытуемая система принудительно обучается с помощью примеров «стимул – реакция». Между входами и эталонными выходами может существовать некоторая зависимость, но она неизвестна. Известна только конечная совокупность прецедентов – обучающая выборка. Чтобы проверить способность модели к обобщению, всю обучающую выборку разделяют на два множества – обучающее и тестовое.

Обучающее множество включает данные, использующиеся для обучения (конструирования) модели. Оно содержит входные и выходные значения примеров.

На основе этих данных требуется восстановить зависимость (построить модель отношений «стимул – реакция»), т. е. построить алгоритм, способный для любого объекта выдать достаточно точный ответ.

Тестовое множество также содержит входные и выходные значения примеров. Выходные значения используются для проверки работоспособности модели.

Категория «обучение с учителем» представлена такими задачами Data Mining как классификация и прогнозирование.

«Обучение без учителя» – один из способов машинного обучения, при котором испытуемая система спонтанно обучается выполнять поставленную задачу без вмешательства со стороны экспериментатора. Как правило, это пригодно только для задач, в которых известны описания множества объектов (обучающего множества), и требуется обнаружить внутренние взаимосвязи, зависимости, закономерности, существующие между объектами.

Категория «обучение без учителя» представлена задачей кластеризации.

В процессе обучения модели можно выделить две ошибки: ошибку обучения и ошибку обобщения.

Ошибка обучения – это ошибка, допущенная моделью на обучающем множестве. На каждой итерации обучения для непрерывной входной переменной она рассчитывается как среднеквадратическая ошибка:

$$E = \frac{1}{N}\sum_{i=1}^{N}\left(y^{(i)} - y_d^{(i)}\right)^2,$$

где N – число обучающих примеров,

$y^{(i)}$ – значение на выходе для i-го примера,

$y_d^{(i)}$ – целевое значение.

Ошибка обобщения – это ошибка, полученная на тестовых примерах, т. е. вычисляемая по тем же формулам, но для тестового множества.

3.2.1. *Классификация данных*

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.

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).

Пример.

База данных о клиентах турагентства 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 (табл. 3). Необходимо определить, к какому классу принадлежит новый клиент, и какой из двух видов рекламных материалов ему стоит отсылать.

Таблица 3

Исходные данные для классификации

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

Для наглядности представим нашу базу данных в двухмерном измерении (возраст и доход), в виде множества объектов, принадлежащих классам 1 (черная метка) и 2 (серая метка).

Решение задачи будет состоять в том, чтобы определить, к какому классу относится новый клиент, на рис. 14 обозначенный белой меткой.

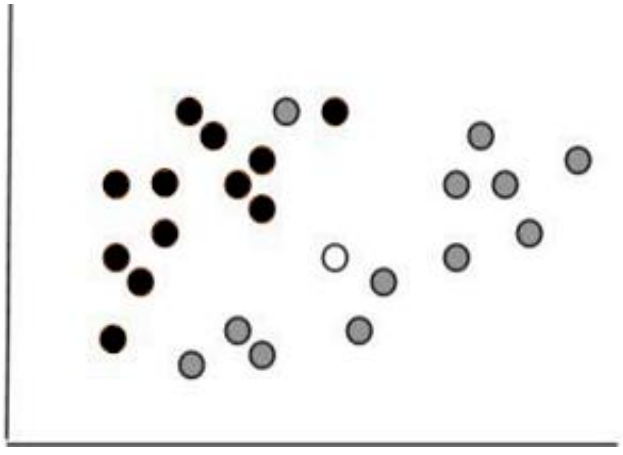


Рис. 14. Множество объектов базы данных
в двухмерном измерении

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.

Процесс классификации состоит из двух этапов: конструирования модели (описания множества предопределенных классов) и ее использования (классификации новых или неизвестных значений).

Основные методы классификации:

- 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 при помощи генетических алгоритмов.

Схематическое решение задачи классификации некоторыми методами приведено на рис. 15–17.

Оценка точности классификации может проводиться при помощи **кросс-проверки**, при которой точность классификации тестового множества сравнивается с точностью классификации обучающего множества.

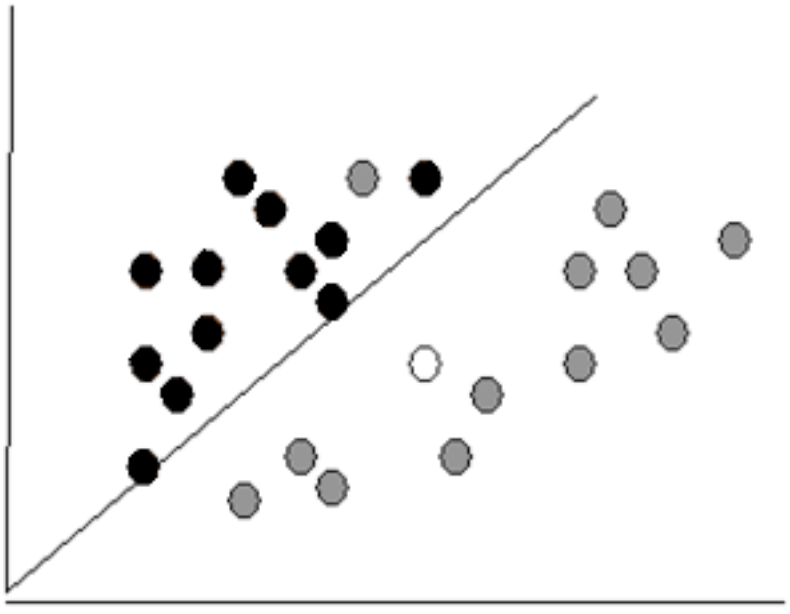


Рис. 15. Решение задачи классификации
методом линейной регрессии

```
ЕСЛИ X > 5 TO grey
ИНАЧЕ ЕСЛИ Y > 3 TO orange
        ИНАЧЕ ЕСЛИ X > 2 TO grey
                ИНАЧЕ orange
```

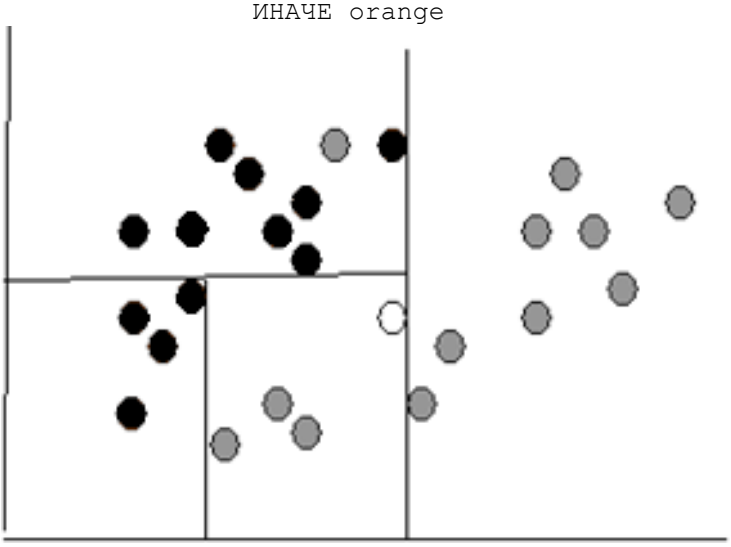


Рис. 16. Решение задачи классификации
методом деревьев решений

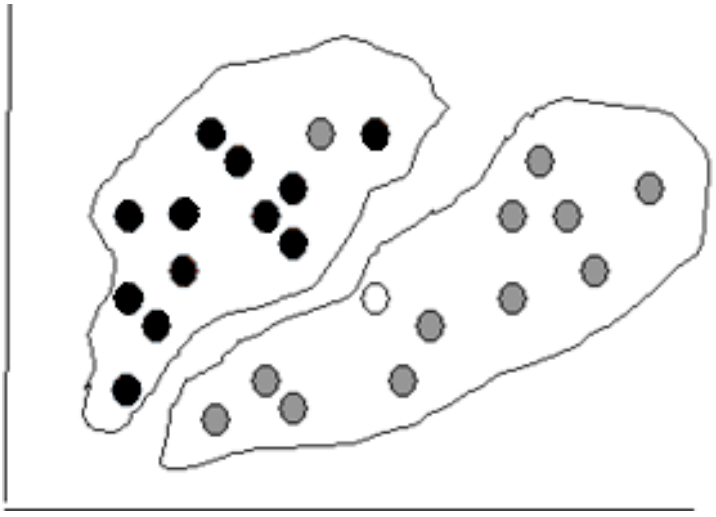


Рис. 17. Решение задачи классификации
методом нейронных сетей

3.2.2. *Кластеризация данных*

Кластеризация предназначена для разбиения совокупности объектов на однородные группы (кластеры), имеющие общие свойства. Если данные выборки представить как точки в признаковом пространстве, то задача кластеризации сводится к определению «сгущений точек».

Цель кластеризации – сегментация (объединение сходных событий в группы на основании сходных значений нескольких полей в наборе данных). В отличие от задачи классификации здесь классы объектов изначально не предопределены

Аналитику часто легче выделить группы схожих объектов, изучить их особенности и построить для каждой группы отдельную модель, чем создавать одну общую модель для всех данных. Таким приемом постоянно пользуются в маркетинге, выделяя группы клиентов, покупателей, товаров и разрабатывая для каждой из них отдельную стратегию

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.

Кластеры могут быть непересекающимися и пересекающимися (рис. 18).

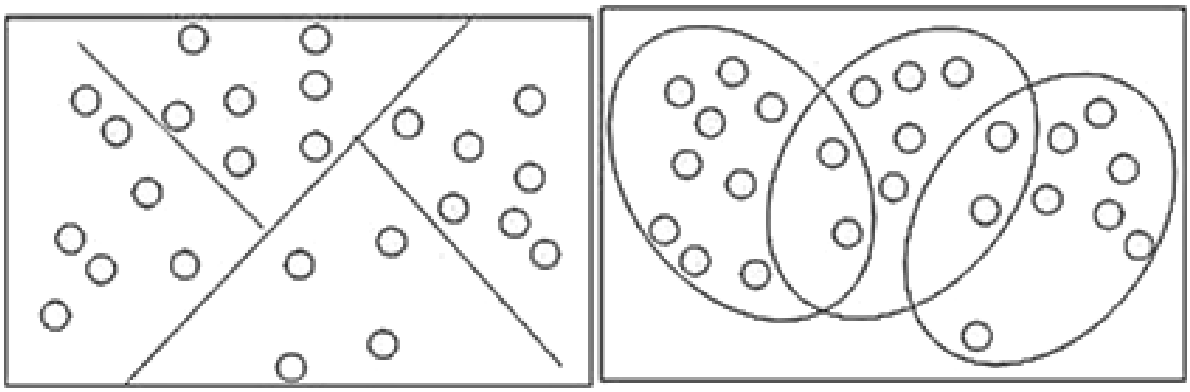


Рис. 18. Непересекающиеся и пересекающиеся кластеры

В случае перекрывающихся кластеров 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.

Критерием для определения схожести и различия кластеров является расстояние между точками на диаграмме рассеивания. Наиболее распространенный способ определения меры расстояния между кластерами (меры близости) – вычисление евклидова расстояния между двумя точками *i* и *j* на плоскости, когда известны их координаты *X* и *Y*.

$$D_{ij} = \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2}.$$

Кластеризация относится к стратегии «обучение без учителя».

Методы кластерного анализа можно разделить на две группы:

– *иерархические* (агломеративные и дивизимные). Суть иерархической кластеризации состоит в последовательном объединении меньших кластеров в большие или разделении больших кластеров на меньшие;

– *неиерархические*. 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.

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.

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

Пример.

Имеется база данных абонентов телекоммуникационной компании. Необходимо провести сегментацию абонентов и для каждого сегмента выработать свою маркетинговую стратегию. В результате можно, например, получить такие кластеры как «Бизнес-люди», «Тусовщики», «Работающие неактивные люди», «Неактивная молодежь», «Активная группа зрелого и пенсионного возраста», «Неактивная группа предпенсионного возраста», «Неактивная группа пенсионного возраста».

3.2.3. Прогнозирование

Прогнозирование в широком понимании этого слова определяется как опережающее отражение будущего. Целью прогнозирования является предсказание будущих событий.

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.

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 относящиеся к будущему).

Так, определение класса клиента является решением задачи классификации, а прогнозирование дохода, который принесет этот клиент в будущем году, будет решением задачи прогнозирования.

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

Примеры: прогноз движения денежных средств, прогнозирование урожайности агрокультуры, прогнозирование финансовой устойчивости предприятия, прогнозирование рынков и др. 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.

Основные методы прогнозирования:

– линейная регрессия;

– нейронные сети;

– деревья решений.

Основой для прогнозирования являются временные ряды. **Временной ряд** – последовательность наблюдаемых значений какого-либо признака, упорядоченных в неслучайные моменты времени. Например, данные биржевых торгов.

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.

В процессе определения структуры и закономерностей временного ряда предполагается обнаружение: шумов и выбросов, тренда, циклической (в том числе сезонной) компоненты.

Тренд – неслучайная функция, которая формируется под действием тенденций, влияющих на временной ряд (например, фактор роста исследуемого рынка).

Циклическая составляющая является периодически повторяющейся компонентой временного ряда. Свойство цикличности важно при определении количества ретроспективных данных, которые будут использоваться для прогнозирования.

Период прогнозирования – основная единица времени, на которую делается прогноз. Например, если необходимо узнать доход компании через месяц, то период прогнозирования – месяц.

Горизонт прогнозирования – это число периодов в будущем, которые покрывает прогноз. Например, если необходимо узнать прогноз на 12 месяцев вперед, с данными по каждому месяцу, то горизонт прогнозирования – 12 месяцев.

Горизонт прогнозирования должен быть не меньше, чем время, необходимое для реализации решения, принятого на основе этого прогноза. Только в этом случае прогнозирование будет иметь смысл. С увеличением горизонта точность прогноза снижается, а с уменьшением – повышается.

Интервал прогнозирования – частота, с которой делается новый прогноз. Может совпадать с периодом прогнозирования.

При длительном интервале возникает риск не идентифицировать изменения, произошедшие в процессе, при коротком – возрастают издержки на прогнозирование.

Точность прогноза характеризуется ошибкой прогноза. Наиболее распространенные виды ошибок:

– средняя ошибка. Вычисляется простым усреднением ошибок на каждом шаге. Недостаток этого вида ошибки – положительные и отрицательные ошибки аннулируют друг друга;

– среднеквадратическая ошибка. Вычисляется как сумма (или среднее) квадратов ошибок. Это наиболее часто используемая оценка точности прогноза;

– средняя абсолютная ошибка. 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.

3.2.4. Поиск ассоциативных правил

Ассоциация – высокая вероятность связи событий друг с другом. В ходе решения задачи поиска ассоциативных правил отыскиваются закономерности между связанными событиями в наборе данных. Здесь поиск закономерностей осуществляется не на основе свойств анализируемого объекта, а между несколькими событиями, которые происходят одновременно.

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.

Основные термины

Транзакция – множество событий, которые произошли одновременно.

Поддержка правила – количество или процент транзакций, содержащих определенный набор данных.

Достоверность правила показывает, какова вероятность того, что из события А следует событие В.

Например, есть правило: «Покупатель, приобретающий «хлеб», приобретет и «молоко» с вероятностью 75 %». В этом случае 75 % транзакций, содержащих хлеб, также содержат молоко. 3 % от общего числа всех транзакций содержат оба товара. 75 % – это достоверность правила, 3 % – это поддержка правила.

Если значение поддержки 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, 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.

3.2.5. Визуализация данных

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.

В результате визуализации создается графический образ анализируемых данных. Визуализация позволяет увидеть конечный результат вычислений, организовать управление вычислительным процессом. Применение визуализации помогает в процессе анализа данных увидеть аномалии, структуры, тренды.

Зачем же использовать визуализацию данных? Визуальная информация лучше воспринимается и позволяет быстро и эффективно донести

до зрителя собственные мысли и идеи. Физиологически, воспринятая визуально информация является основной для человека. Есть многочисленные исследования, подтверждающие, что:

− 90 % информации человек воспринимает через зрение;

− 70 % сенсорных рецепторов находятся в глазах;

− около половины нейронов головного мозга человека задействованы в обработке визуальной информации;

− на 19 % меньше при работе с визуальными данными используется когнитивная функция мозга, отвечающая за обработку и анализ информации;

− на 17 % выше производительность человека, работающего с визуальной информацией;

− на 4,5 % лучше воспоминаются подробные детали визуальной информации;

− 10 % информации человек запоминает из услышанного, 20 % – из прочитанного, и 80 % – из увиденного и сделанного;

− на 323 % лучше человек выполняет инструкцию, если она содержит иллюстрации.

Помимо легкого восприятия, визуализация данных имеет несколько преимуществ:

− акцентирование внимания на разных аспектах данных;

− анализ большого набора данных со сложной структурой;

− уменьшение информационной перегрузки человека и удержание его внимания;

− однозначность и ясность выводимых данных;

− выделение взаимосвязей и отношений, содержащихся в информации.

Задача визуализации данных состоит в том, чтобы преобразовать числовые массивы в геометрические образы или объекты. Если в двумерном изобразительном пространстве выбор форм представления сильно ограничен и они весьма абстрактны, то в пространстве 3D их можно в максимальной степени приблизить к моделируемой реальности. В этом аспекте визуализация данных развивается параллельно со смежными разделами машинной графики – геометрическим моделированием, анимацией и виртуальной реальностью.

Для решения задачи визуализации используются графические методы, показывающие наличие закономерностей в данных. 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.

Существует несколько типов визуализации:

− обычное визуальное представление количественной информации в схематической форме. К этой группе можно отнести круговые и линейные диаграммы, гистограммы и спектрограммы, таблицы и различные точечные графики;

− данные при визуализации могут быть преобразованы в форму, усиливающую восприятие и анализ этой информации. Например, карта и полярный график, временная линия и график с параллельными осями, диаграмма Эйлера;

− концептуальная визуализация позволяет разрабатывать сложные концепции, идеи и планы с помощью концептуальных карт, диаграмм Ганта, графов с минимальным путем и других подобных видов диаграмм;

− стратегическая визуализация переводит в визуальную форму различные данные об аспектах работы организаций. Это всевозможные диаграммы производительности, жизненного цикла и графики структур организаций;

− графически организовать структурную информацию с помощью пирамид, деревьев и карт данных поможет метафорическая визуализация, ярким примером которой является карта метро;

− комбинированная визуализация позволяет объединить несколько сложных графиков в одну схему, как в карте с прогнозом погоды.

3.3. Основные методы Data Mining

3.3.1. Корреляционно-регрессионный анализ

Для исследования интенсивности, вида и формы зависимостей применяется корреляционно-регрессионный анализ, который является методическим инструментарием при решении задач прогнозирования.

Корреляционный анализ – это количественный метод определения тесноты и направления взаимосвязи между выборочными переменными величинами.

Данный метод анализа применяется для оценки зависимости выходных полей данных от входных факторов и устранения незначащих факторов. Критерием принятия решения об исключении является **порог значимости**. Если корреляция между входным и выходным факторами меньше порога значимости, то соответствующий фактор отбрасывается как незначащий.

Коэффициент корреляции Пирсона, 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:

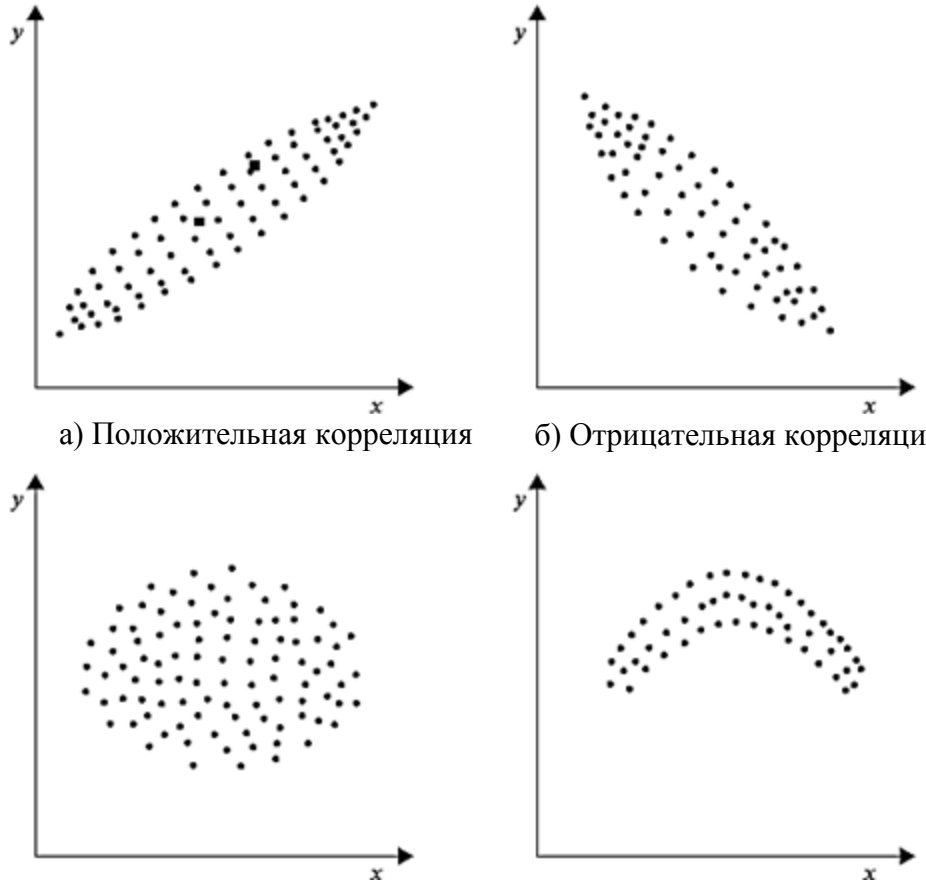
$$r_{xy} = \frac{n\sum xy - \sum x \sum y}{\sqrt{\left[n\sum x^2 - \left(\sum x\right)^2\right] \times \left[n\sum y^2 - \left(\sum y\right)^2\right]}},$$

где x – значение факторного признака;

y – значение результативного признака;

n – число пар данных.

Для графического представления связи двух переменных использована система координат с осями, соответствующими переменным x и y. Построенный график называется **диаграммой рассеивания**. Обычно значения независимого параметра откладывается по горизонтальной оси, а значения зависимого – по вертикальной (рис. 19).



а) Положительная корреляция б) Отрицательная корреляция

в) Отсутствие корреляции г) Нелинейная корреляция

Рис. 19. Диаграммы рассеивания

Регрессионный анализ – это количественный метод определения вида математической функции в причинно-следственной зависимости между переменными величинами.

Задачи регрессионного анализа

1. Установление формы зависимости.

Относительно формы зависимостей выделяют:

− линейная регрессия – выражается линейной функцией (например, связь между количеством тренировок на тренажере и количеством правильно решаемых задач в сессии);

− нелинейная регрессия – выражается нелинейной функцией (например, связь между уровнем мотивации и эффективностью выполнения задачи).

Относительно числа переменных выделяют:

− парная регрессия – регрессия между двумя переменными;

− множественная регрессия – регрессия между зависимой переменной и несколькими факторами.

В зависимости от характера регрессии различают:

− положительную регрессию. Она имеет место, если с увеличением (уменьшением) объясняющей переменной значения зависимой переменной также соответственно увеличиваются (уменьшаются);

− отрицательную регрессию. В этом случае с увеличением или уменьшением объясняющей переменной зависимая переменная уменьшается или увеличивается.

2. Определение функции регрессии.

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.

3. 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.**

Модель линейной регрессии является часто используемой и наиболее изученной. Уравнение линейной регрессии выглядит следующим образом:

$$Y = b_0 + b_1 * x_1 + b_2 * x_2 + \ldots + b_n * x_n,$$

где Y – выходная (зависимая) переменная модели;

$x_1, x_2, \ldots x_n$ – входные (независимые) переменные;

b_0 – константа (свободный член);

b_i – коэффициенты линейной регрессии (параметры модели).

Коэффициент регрессии показывает, на сколько (в абсолютном выражении) изменяется значение результативного признака при изменении факторного признака на единицу.

Задача линейной регрессии заключается в подборе коэффициентов b_i таким образом, чтобы на заданный входной вектор $X = (x_1, x_2 \ldots x_n)$ регрессионная модель формировала желаемое выходное значение Y.

Вычисляемая с помощью **метода наименьших квадратов** линия называется **линией регрессии**. Она характеризуется тем, что сумма квадратов расстояний от точек на диаграмме до этой линии минимальна (по сравнению со всеми возможными линиями).

В большинстве случаев наблюдается определенный разброс наблюдений относительно регрессионной прямой. **Остаток** – это отклонение отдельной точки (наблюдения) от линии регрессии (предсказанного значения).

В нелинейной регрессии часто встречаются полиномиальная, параболическая, гиперболическая, степенная, показательная и экспоненциальная зависимости.

Пример.

На рис. 20 представлен анализ зависимости между количеством работников и объемом производства.

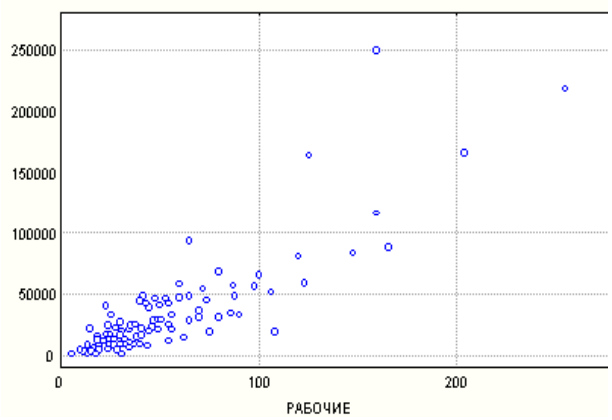
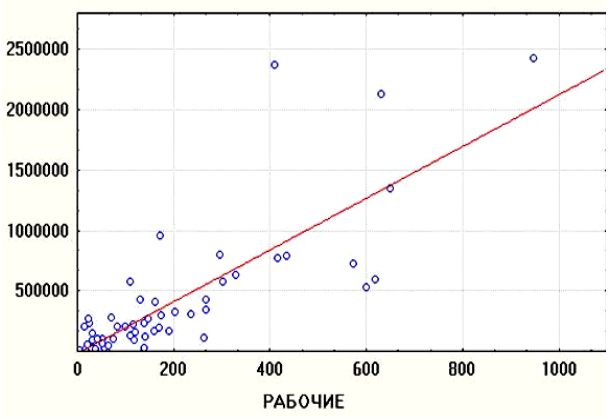


Диаграмма рассеяния



Линия регрессии

Рис. 20. Результаты корреляционно-регрессионного анализа

В случае если на объем производства влияют несколько факторов (например, количество работников и энерговооруженность), то результат множественной регрессии можно визуализировать через гиперплоскость (рис. 21).

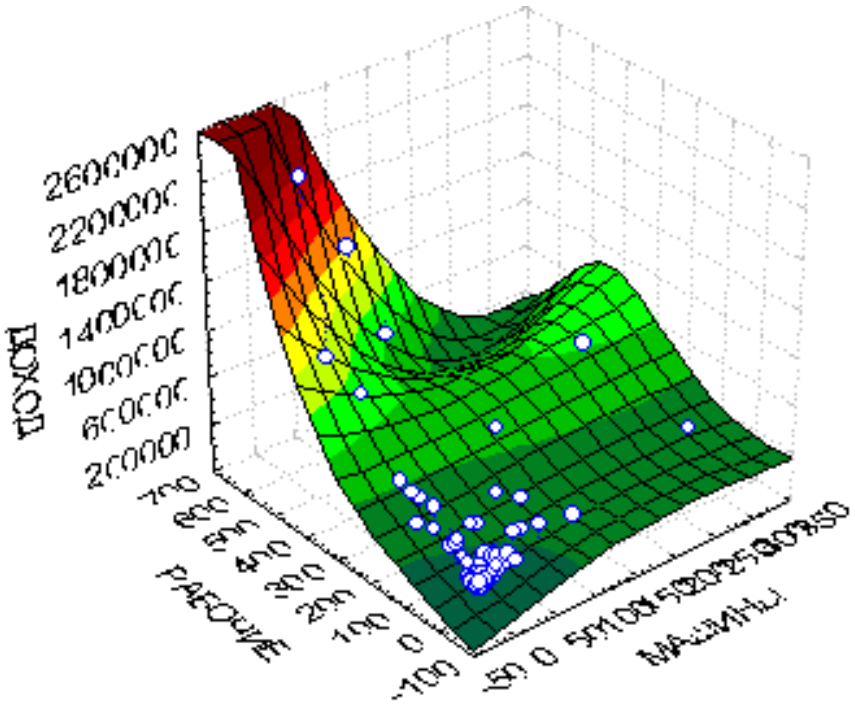


Рис. 21. Результаты
множественного регрессионного анализа

3.3.2. Деревья решений

Метод деревьев решений является одним из наиболее популярных методов решения задач классификации и прогнозирования.

Дерево решений представляет собой иерархическую структуру, базирующуюся на наборе вопросов, подразумевающих ответ «да» или «нет».

Если целевая переменная принимает дискретные значения, решается задача классификации. Если же зависимая переменная принимает непрерывные значения, то решается задача численного прогнозирования.

Пример.

«Играть ли в гольф?». Для решения требуется ответить на ряд вопросов, которые находятся в узлах дерева, начиная с его корня (рис. 22).

Основные понятия деревьев решений

Атрибуты – признаки, описывающие классифицируемые объекты (например, температура воздуха).

Узлы – содержат правила, с помощью которых производится проверка атрибутов. И множество объектов в узле разбивается на подмножества.

Листья – конечные узлы дерева, в которых содержатся подмножества, ассоциированные с классами («Играть», «Не играть»).

Корень дерева (корневой узел) – начальный (входной) узел дерева («Солнечно?»).

Внутренний узел (узел проверки): «Температура воздуха высокая?», «Идет ли дождь?».

Ветвь дерева (случаи ответа): «Да», «Нет».

Атрибут ветвления – атрибут, по которому будет производиться проверка правила.

Алгоритм построения дерева решений – метод, в соответствии с которым осуществляется выбор атрибута ветвления на каждом шаге.

Чистота – мера оценки разбиения узла. Наилучшим разбиением является то, которое дает наибольшее увеличение чистоты дочерних узлов относительно родительского.

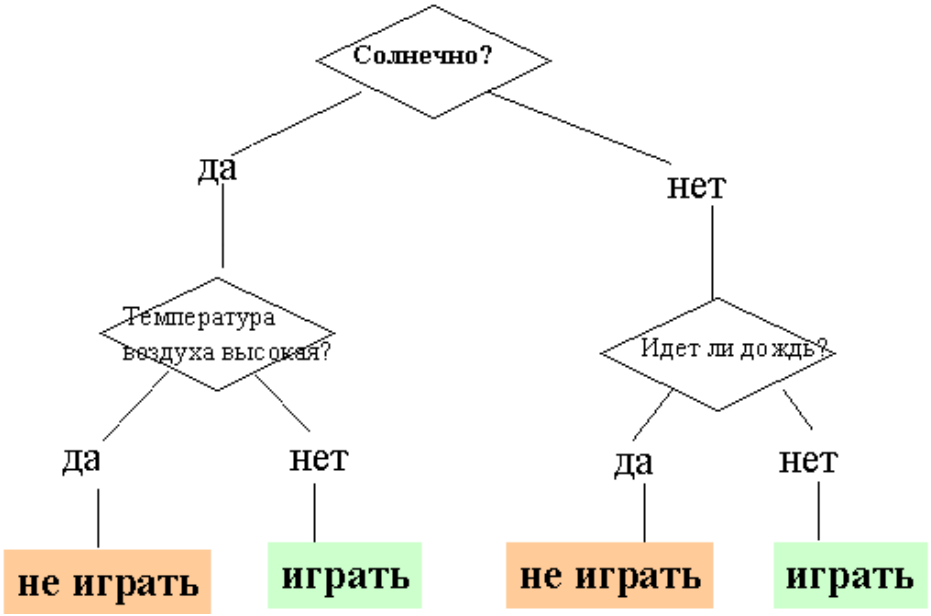


Рис. 22. Дерево решений «Играть ли в гольф?»

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.

Для выбора атрибута ветвления используются такие методы как: индекс Джини (метод разбиения выборки), энтропия (прирост информации), отношение прироста информации, тест хи-квадрат [9].

Достоинства деревьев решений:

− просты в понимании и интерпретации;

− не требуют подготовки данных;

− быстрый процесс обучения;

− используется модель «белого ящика». Если определенная ситуация наблюдается в модели, то ее можно объяснить при помощи булевой логики;

− метод является надежным;

− позволяют работать с большим объемом информации без специальных подготовительных процедур.

Хорошее разбиение должно создавать узлы примерно одинакового размера или как минимум не создавать узлы, содержащие всего несколько записей. Когда найти новые разбиения, повышающие его точность не удается и разбиение прекращается по всем ветвям, это значит, что построено **полное дерево.**

Однако в результате могут быть созданы слишком сложные конструкции (большая глубина дерева). Такие «ветвистые» деревья очень трудно понять. Ценность правила, справедливого для 2–3 объектов, крайне низка. Лучше иметь дерево, состоящее из малого количества узлов, которым бы соответствовало большое количество объектов из обучающей выборки.

Остановка – такой момент в процессе построения дерева, когда следует прекратить дальнейшие ветвления. Один из вариантов правил остановки – «ранняя остановка», она определяет целесообразность разбиения узла. Преимущество использования такого варианта – уменьшение времени на обучение модели. Однако здесь возникает риск снижения точности классификации.

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 станут чистыми или будут содержать не более чем заданное число объектов.

Решением проблемы слишком ветвистого дерева является его сокращение путем **отсечения** 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.

Для оценки качества классификации используют показатели поддержки и достоверности.

Поддержка определяется как отношение числа правильно классифицированных примеров в данном узле к общему числу попавших в него примеров.

Достоверность определяется как отношение числа правильно классифицированных примеров к числу ошибочно классифицированных.

На сегодняшний день существует большое число алгоритмов, реализующих деревья решений: CART, C4.5, CHAID, CN2, NewId, ITrule и другие [8].

3.3.3. Нейронные сети

Нейронные сети – это модели биологических нейронных сетей мозга, в которых нейроны имитируются относительно простыми, часто однотипными элементами (искусственными нейронами).

Идея нейронных сетей родилась в рамках теории искусственного интеллекта, в результате попыток имитировать способность биологических нервных систем обучаться и исправлять ошибки. Она основана на аналогии с функционированием нервной ткани и заключается в том, что исходные параметры рассматриваются как сигналы, преобразующиеся в соответствии с имеющимися связями между «нейронами», а в качестве ответа (результата анализа) рассматривается отклик всей сети на исходные данные.

Нейронная сеть может быть представлена направленным графом с взвешенными связями, в котором искусственные нейроны являются вершинами, а синаптические связи – дугами.

Среди задач Data Mining, решаемых с помощью нейронных сетей, можно выделить классификацию, прогнозирование и кластеризацию.

Примеры

Медицинская диагностика. Риск наступления осложнений может соответствовать сложной нелинейной комбинации наблюдаемых переменных, которая обнаруживается с помощью нейросетевого моделирования.

Прогнозирование объемов продаж. На основе ретроспективной информации о деятельности организации возможно определение объемов продаж на будущие периоды.

Предоставление кредита. Используя базу данных о клиентах банка, можно установить группу клиентов, которые относятся к группе потенциальных «неплательщиков».

Элементы нейронных сетей

Искусственный нейрон – элемент искусственных нейронных сетей, моделирующий некоторые функции биологического нейрона. Его главная функция – формировать выходной сигнал в зависимости от сигналов, поступающих на его входы (рис. 23).

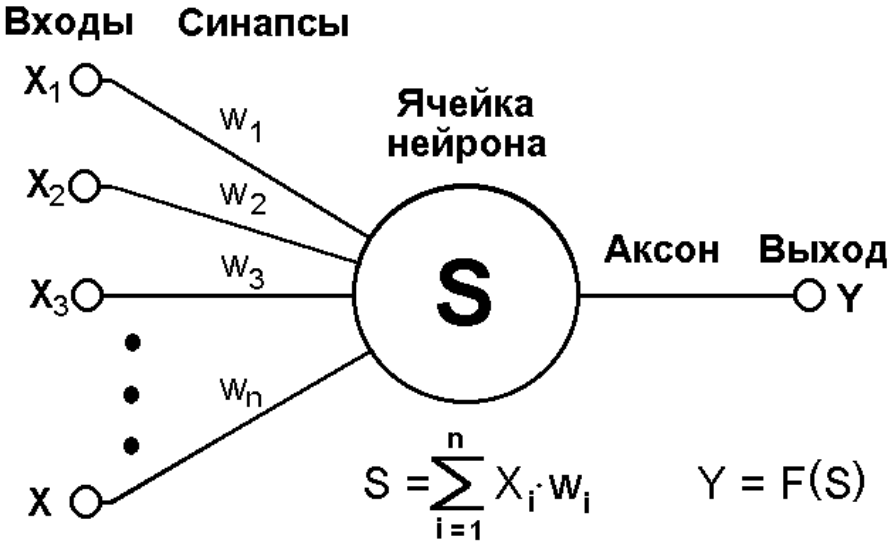


Рис. 23. Искусственный нейрон

Синапсы – однонаправленные входные связи, соединенные с выходами других нейронов.

Аксон – выходная связь данного нейрона, с которой сигнал поступает на синапсы следующих нейронов.

Каждый синапс характеризуется величиной *синаптической связи* (ее весом w_i). Текущее состояние нейрона определяется как взвешенная сумма его входов.

Активационная (характеристическая) функция – нелинейная функция, вычисляющая выходной сигнал нейрона.

Нелинейный преобразователь – элемент нейрона, преобразующий текущее состояние нейрона (выходной сигнал адаптивного сумматора) в выходной сигнал нейрона по некоторому нелинейному закону (активационной функции).

В качестве оператора нелинейного преобразования могут выступать различные функции (линейная, пороговая, сигмоидная). В силу монотонности и всюду дифференцируемости сигмоидная функция является наиболее распространенной.

Точка ветвления (выход) – это элемент 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.

Наиболее популярной архитектурой нейронных сетей являются слоистые сети.

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.

На рис. 24 представлена однослойная, а на рис. 25 двухслойная сеть.

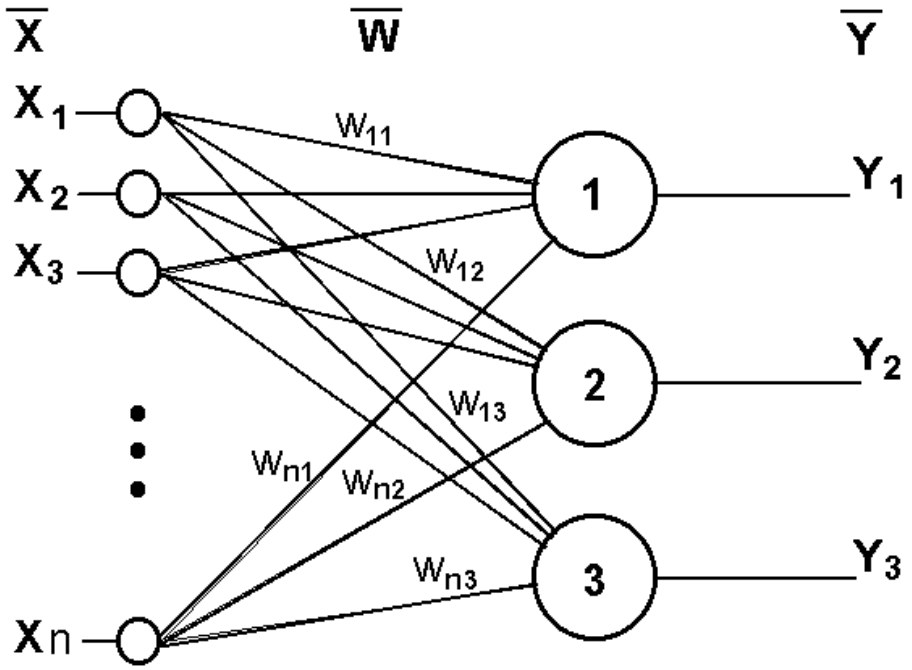


Рис. 24. Однослойная трехнейронная сеть

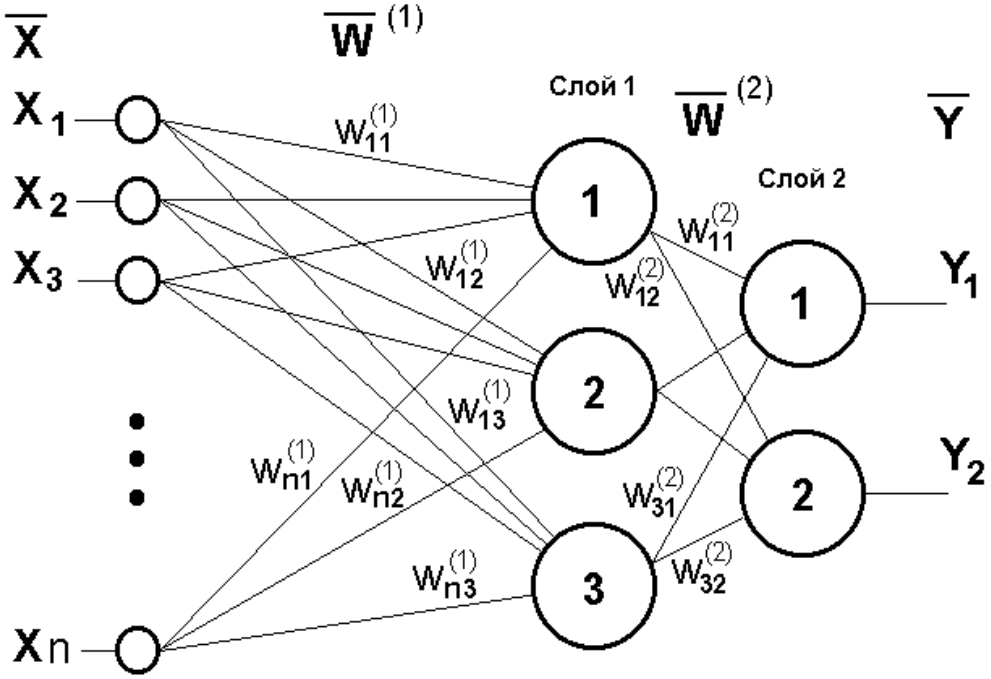


Рис. 25. Двухслойная сеть

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. 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 корреляция между образцами в наборе данных. Выходы нейронной сети формируются самостоятельно, а веса изменяются по алгоритму, учитывающему только входные и производные от них сигналы. В результате такого обучения объекты или примеры распределяются по категориям, сами категории и их количество могут быть заранее неизвестны.

Эпоха – одна итерация в процессе обучения, включающая предъявление всех примеров из обучающего множества и, возможно, проверку качества обучения на тестовом множестве.

Ошибка обучения для построенной нейронной сети вычисляется путем сравнения выходных и целевых (желаемых) значений. Из полученных разностей формируется **функция ошибок** – целевая функция, требующая минимизации в процессе управляемого обучения нейронной сети.

При подготовке данных для обучения нейронной сети необходимо обращать внимание на следующие существенные моменты:

1) чем больше размерность данных, тем больше времени потребуется для обучения сети;

2) следует определить наличие выбросов и оценить необходимость их присутствия в выборке;

3) обучающая выборка должна быть представительной (репрезентативной) и не должна содержать противоречий;

4) нейронная сеть работает только с числовыми входными данными;

5) на вход нейронной сети следует подавать значения из того диапазона, на котором она обучалась.

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.

При обработке временных рядов, например, с целью построения модели прогноза временного ряда, часто используется «**скользящее окно**». Данные необходимо преобразовать по специальной схеме. Сначала преобразуется исходный временный ряд в ряд приращений прогнозируемой величины. Затем выбирается глубина погружения, т. е. количество временных интервалов, по которым будет прогнозироваться следующий. В табл. 4 приведен пример для глубины погружения равной 4, т. е. прогнозирование величины на следующую итерацию осуществляется по результатам четырех предыдущих итераций.

Таблица 4

Скользящее окно

Hist1	Hist2	Hist3	Hist4	Hist0
D-1	D-2	D-3	D-4	D
D-2	D-3	D-4	D-5	D-1
D-3	D-4	D-5	D-6	D-2
…	…	…	…	…

Первые четыре колонки являются входами нейросети, последняя – выходом, т. е. на основе предыдущих значений изменения величины прогнозируется следующее значение ряда. В результате мы получаем «скользящее окно», в котором представлены данные за пять недель. Окно можно двигать по временной оси и изменять его ширину.

Таким образом, готовится обучающая выборка, и именно в таком виде предоставляются данные для последующего анализа.

В качестве недостатка нейронных сетей можно выделить длительное время их обучения.

3.3.4. Самоорганизующаяся карта Кохонена

Карты Кохонена – это соревновательная нейронная сеть с обучением без учителя. 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 – решение задачи кластеризации.

1. **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.

2. **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, не похожим ни на один из известных образцов, то она не сможет классифицировать такой набор и тем самым выявит его новизну.

Сеть Кохонена представляет собой два слоя: входной и выходной (рис. 26).

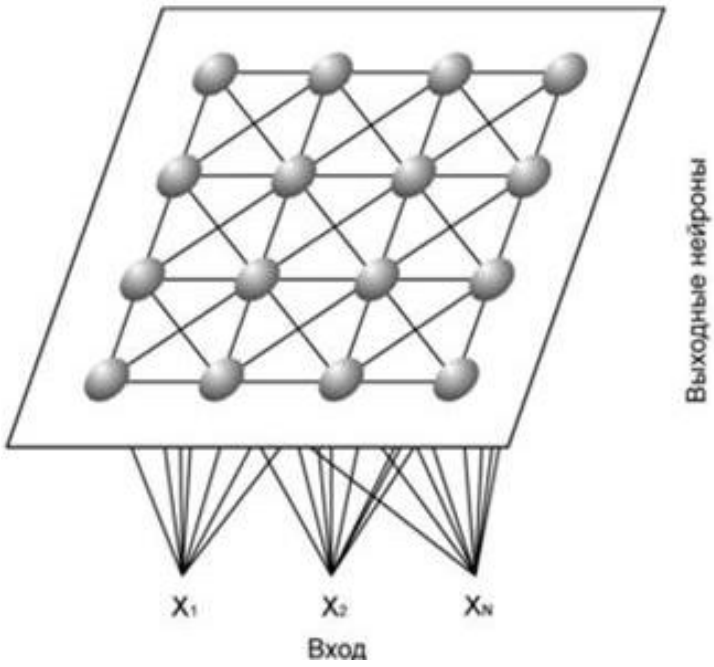


Рис. 26. Сеть Кохонена

Сеть Кохонена обучается методом последовательных приближений. В процессе обучения 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.

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.

Нейроны карты Кохонена располагают в виде двухмерной матрицы, раскрашивают эту матрицу в зависимости от анализируемых параметров нейронов. Для каждого входа рисуется своя карта, раскрашенная в соответствии со значением конкретного веса нейрона.

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.

Пример.

На рис. 27 изображена самоорганизующаяся карта Кохонена с шестиугольными ячейками, построенная по нескольким показателям качества жизни в разных странах. После построения гексагональная сетка не была окрашена, на нее были лишь нанесены метки стран, причем страны, оказавшиеся рядом на карте, обладают сходными показателями. Затем на карту Кохонена был наложен двумерный цветной спектр (на рисунке нет возможности показать цветную окраску), в результате чего каждый узел получил свой цвет, причем соседние узлы получили близкие по спектру цвета.

Цветовая разметка карты Кохонена, перенесенная на карту мира представлена на рис. 28.

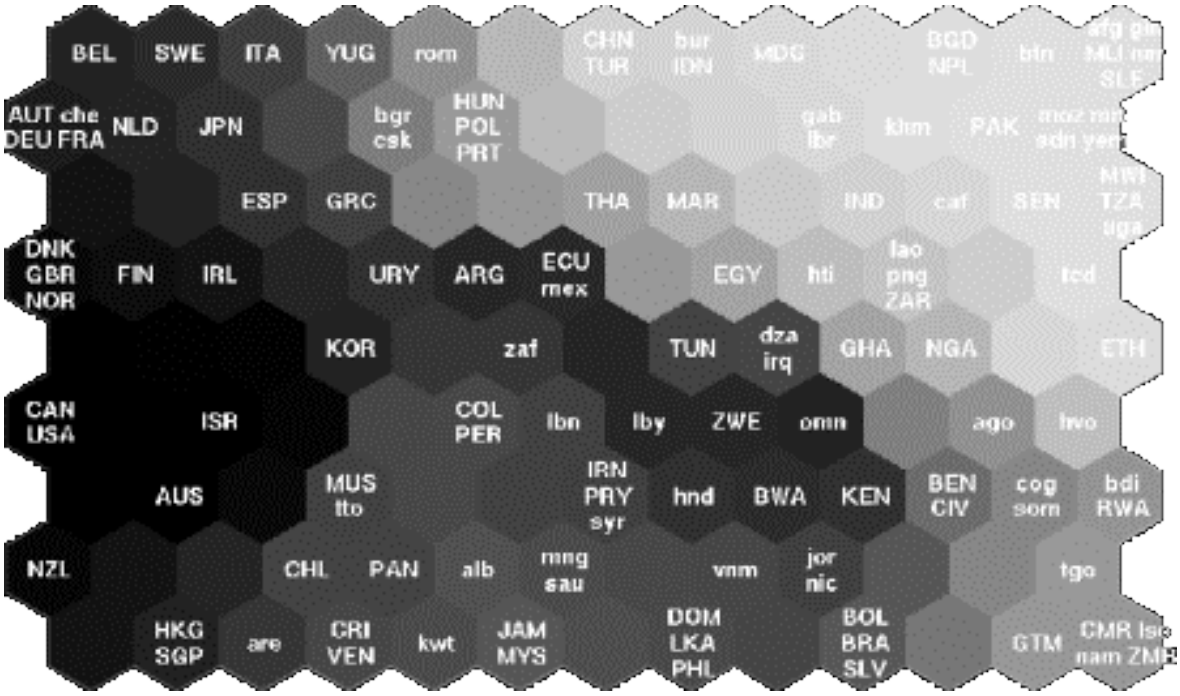


Рис. 27. Карта Кохонена

Рис. 28. Раскраска географической карты

Таким образом, карты Кохонена – мощное средство визуализации и разведочного анализа данных. Однако у них есть и недостаток, связанный с эвристическим характером данного метода. Задание начальных векторов весов нейронов является произвольным, при этом возможна потеря однозначности результата. Если обучать карту несколько раз, то всегда будут получаться непохожие итоги.

3.3.5. *Метод k-means (метод k-средних)*

Наиболее распространенным среди неиерархических методов кластеризации является алгоритм *k*-средних. 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.

Выбор числа *k* может базироваться на результатах предшествующих исследований, теоретических соображениях или интуиции. Если нет предположений относительно этого числа, рекомендуют создать 2 кластера, затем 3, 4, 5 и т. д., сравнивая полученные результаты.

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 центров 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 (центроиды), 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. 29 приведен пример работы алгоритма для *k*, равного двум.

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.

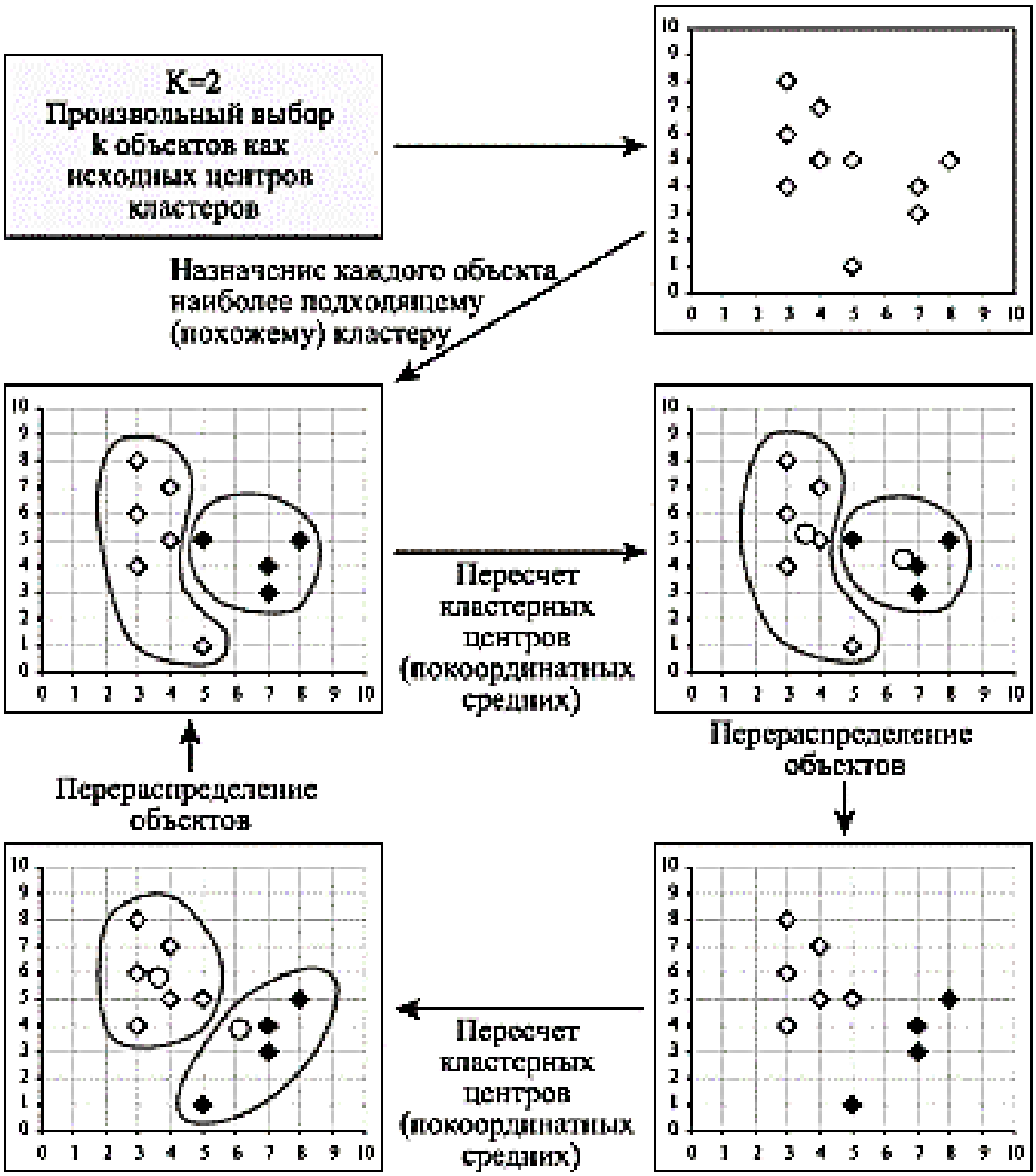


Рис. 29. Пример работы алгоритма *k*-средних (*k* = 2)

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.

3.3.6. Алгоритм Apriori

Наиболее популярным методом поиска ассоциативных правил является метод Apriori. Остальные методы подробно описаны в [9].

Работа данного алгоритма состоит из нескольких этапов, каждый из этапов состоит из следующих шагов:

- формирование кандидатов;
- подсчет кандидатов.

Формирование кандидатов – этап, на котором алгоритм, сканируя базу данных, создает множество *i*-элементных кандидатов (*i* – номер этапа). На этом этапе поддержка кандидатов не рассчитывается.

Подсчет кандидатов – этап, на котором вычисляется поддержка каждого *i*-элементного кандидата. Здесь же осуществляется отсечение кандидатов, поддержка которых меньше минимума, установленного пользователем.

Рассмотрим работу алгоритма Apriori на примере базы данных (табл. 5). Каждому товару для удобства присвоена переменная.

Таблица 5

Транзакционная база данных

ID	Приобретенные покупки	Присвоенные переменные
100	Хлеб, молоко, печенье	*a, b, c*
200	Молоко, сметана	*b, d*
300	Молоко, хлеб, сметана, печенье	*b, a, d, c*
400	Колбаса, печенье	*e, d*
500	Хлеб, молоко, печенье, сметана	*a, b, c, d*
600	Конфеты	*f*

На основе имеющейся базы данных необходимо найти закономерности между покупками. Иллюстрация работы алгоритма приведена на рис. 30. Минимальный уровень поддержки (min_sup) равен 3.

На первом этапе происходит формирование одноэлементных кандидатов. Далее алгоритм подсчитывает поддержку одноэлементных наборов. Наборы с уровнем поддержки меньше установленного, т. е. 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.

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.

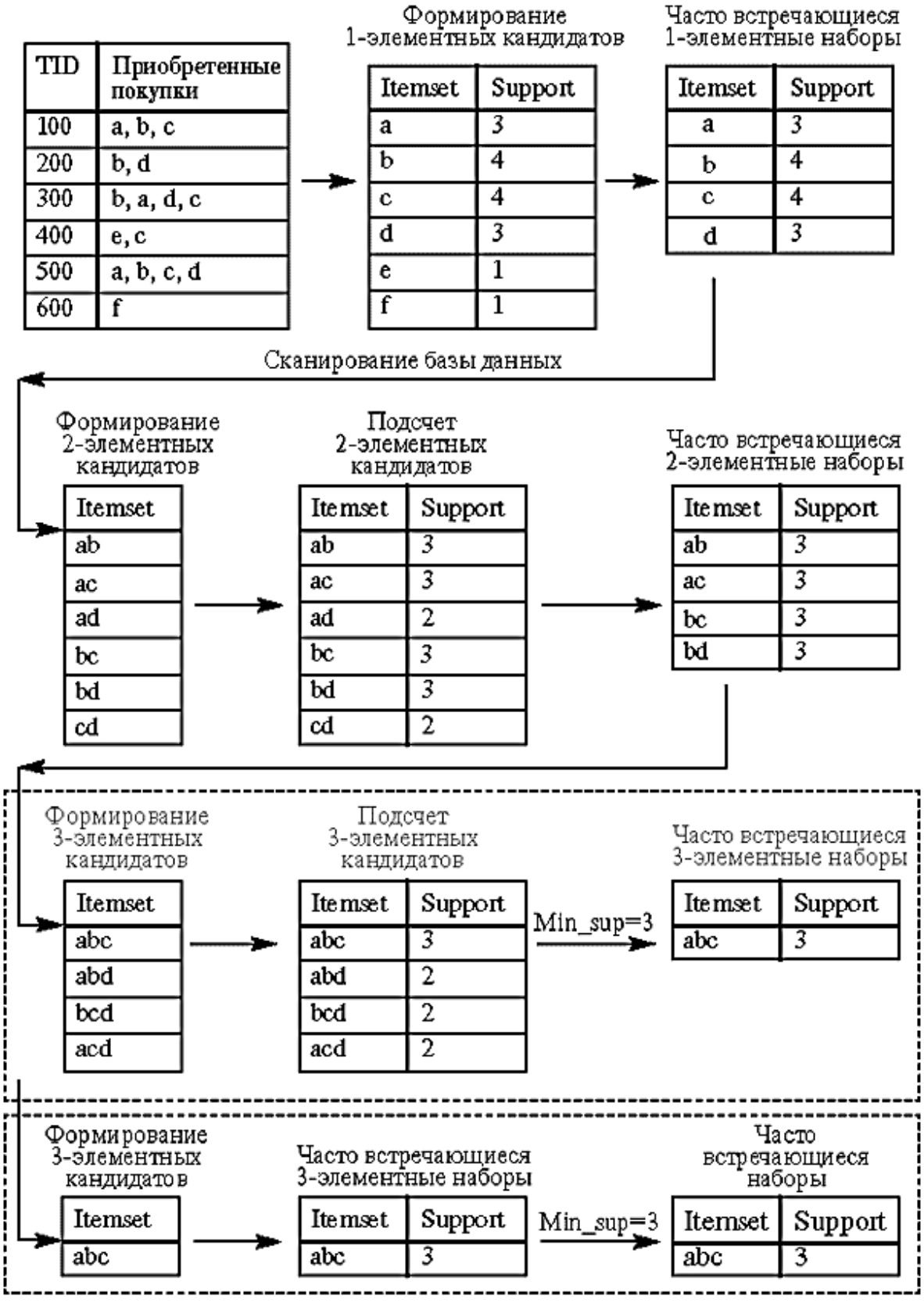


Рис. 30. Алгоритм Apriori

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 набор 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 (рис. 30)).

Алгоритм Apriori рассчитывает также поддержку наборов, которые не могут быть отсечены априори. Это так называемая негативная область, к ней принадлежат наборы-кандидаты, которые встречаются редко, их самих нельзя отнести к часто встречающимся, но все подмножества данных наборов являются часто встречающимися.

3.4. Способы и методы визуального представления данных

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.

Каждый из алгоритмов Data Mining использует определенный подход к визуализации.

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 визуализации данных могут выполнять одну из функций:

− являться иллюстрацией построения модели (например, представление графа нейронной сети);

− помогать интерпретировать полученный результат;

− являться средством оценки качества построенной модели;

− сочетать перечисленные выше функции (дерево решений, дендрограмма).

Примерами средств визуализации, при помощи которых можно оценить качество модели, являются диаграмма рассеивания, таблица сопряженности, график изменения величины ошибки.

Примерами средств визуализации, которые помогают интерпретировать результат, являются: линия тренда в линейной регрессии, карты Кохонена, диаграмма рассеивания в кластерном анализе.

Методы визуализации, в зависимости от количества используемых измерений, принято делить на две группы:

− представление данных в одном, двух и трех измерениях;

− представление данных в четырех и более измерениях.

Успех визуализации напрямую зависит от правильности ее применения, а именно от выбора типа графика, его верного использования и оформления.

График позволяет выразить идею, которую несут данные, наиболее полно и точно, поэтому очень важно выбрать подходящий тип диаграммы. Выбор можно осуществить по алгоритму: Определение целей визуализации данных → Определение типа данных → Выбор подходящего графика.

Цели визуализации – это реализация основной идеи информации, это то, ради чего нужно показать выбранные данные, какого эффекта нужно добиться – выявления отношений в информации, показа распределения данных, композиции или сравнения данных.

Отношения в данных – это то, как они зависят друг от друга, связь между ними. С помощью отношений можно выявить наличие или отсутствие

зависимостей между переменными. Если основная идея информации содержит фразы «относится к», «снижается/повышается при», то нужно стремиться показать именно отношения в данных.

Распределение данных – то, как они располагаются относительно чего-либо, сколько объектов попадает в определенные последовательные области числовых значений. Основная идея при этом будет содержать фразы «в диапазоне от *x* до *y*», «концентрация», «частотность», «распределение».

Композиция данных – объединение данных с целью анализа общей картины в целом, сравнения компонентов, составляющих процент от некоего целого. Ключевыми фразами для композиции являются «составило *x* %», «доля», «процент от целого».

Сравнение данных – объединение данных, с целью сравнения некоторых показателей, выявление того, как объекты соотносятся друг с другом. Также это сравнение компонентов, изменяющихся с течением времени. Ключевые фразы для идеи при сравнении – «больше/меньше чем», «равно», «изменяется», «повышается/понижается».

После определения цели визуализации требуется определить тип данных. Они могут по своему типу и структуре быть очень разнородными, но в самом простом случае выделяют непрерывные числовые и временные данные, дискретные данные, географические и логические данные.

Непрерывные числовые данные содержат в себе информацию зависимости одной числовой величины от другой, например графики функций. Непрерывные временные содержат в себе данные о событиях, происходящих на каком-либо промежутке времени, как график температуры, измеряемой каждый день. Дискретные данные могут содержать в себе зависимости категорийных величин, например график количества продаж товаров в разных магазинах. Географические данные содержат в себе различную информацию, связанную с местоположением, геологией и другими географическими показателями, яркий пример – обычная географическая карта. Логические данные показывают логическое расположение компонентов относительно друг друга, например генеалогическое древо семьи.

В зависимости от цели и данных можно выбрать наиболее подходящий им график. Лучше всего избегать разнообразия ради разнообразия, и выбирать по принципу «чем проще, тем лучше», использовать специфичные типы диаграмм только для специфичных данных, в остальных же случаях хорошо подойдут самые распространенные графики:

- линейный (line),
- с областями (area),
- колонки (столбцы) и гистограммы (bar),
- круговая диаграмма (pie, doughnut),
- полярный график (лепестковая диаграмма) (radar),

- точечная (пузырьковая) диаграмма (scatter, bubble),
- кольцевая диаграмма (ring),
- диаграмма разброса (span),
- карты (map),
- деревья (tree map, mental map),
- временные диаграммы (time line, gantt, waterfall).

Линейные диаграммы, графики с областями и гистограммы могут содержать в одном аргументе для одной категории несколько значений, которые могут быть как абсолютными (тогда к таким видам графикам прибавляется приставка stacked), так и относительными (full stacked).

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.

Столбиковые диаграммы и **гистограммы** 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. 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.

Пузырьковая диаграмма – это разновидность точечной диаграммы, в которой точки данных заменены пузырьками, причем их размер служит дополнительным (третьим) измерением данных.

Кольцевая диаграмма показывает процент от максимального количества, которое занимает одно из значений в наборе данных, в виде частично закрашенного кольца. Часто используется сразу несколько таких диаграмм, сравнивающих разные значения.

Диаграмма разброса показывает минимальную и максимальную величину значений внутри набора данных в виде урезанной столбиковой диаграммы. Начало столбика лежит не на горизонтальной оси, а в точке минимального значения по вертикали.

Лепестковая диаграмма сравнивает величины нескольких значений, каждая из которых соответствует точке на оси. Количество осей соответствует количеству значений, а точки объединены линями.

Среди деревьев и структурных диаграмм можно выделить **ментальную карту**. Она показывает состав и структуру явления в виде дерева, в котором каждый узел имеет один или несколько дочерних элементов. Это частный случай дерева, с той разницей, что ветви расходятся из узла, расположенного в центре изображения.

Диаграмма Венна – Эйлера показывает отношения между значениями набора данных в виде накладывающихся друг на друга кругов (чаще всего трех). Область, в которой пересекаются все круги, показывает общее между ними.

Более подробно с различными типами графиков можно ознакомиться в [10].

При выборе подходящего графика можно руководствоваться табл. 6.

Таблица 6

Выбор правильного типа графика

Цель визуализации/ Тип данных	Отношения в данных	Распределение данных	Сравнение данных	Композиция данных
Непрерывные числовые	line area scatter bubble	scatter bubble	line area radar	stacked line full stacked line stacked area full stacked area
Непрерывные временные	line area radar scatter bubble	time line gantt waterfall radar		gantt stacked line full stacked line stacked area full stacked area
Дискретные	bar scatter bubble		bar pie doughnut	pie doughnut stacked bar full stacked bar
Географические	map line area	map scatter	map bar	map stacked bar full stacked bar
Логические	tree mental tree		tree map	

Важно не только верно выбрать тип графика, но и правильно его использовать.

Не нужно нагружать график большим количеством информации. Оптимальное количество разных типов данных, категорий – не более 4–5, иначе целесообразнее разделить такую диаграмму на несколько.

Необходимо верно выбрать шкалу и ее масштаб для графика. Для гистограмм и графиков с областями предпочтительнее начинать шкалу значений с нуля. Не стоит использовать инвертированные шкалы – это очень часто вводит зрителя в заблуждение относительно данных.

Для круговых диаграмм и графиков, где показан процент от общей доли, сумма значений всегда должна составлять 100 %.

Для лучшего восприятия данных информацию на оси следует упорядочить – либо по значениям, либо по алфавиту, либо по логическому смыслу.

Правильное оформление графиков также немаловажно. К основным принципам оформления можно отнести:

− использовать палитры похожих, неярких цветов и постараться ограничиться набором из шести штук;

− вспомогательные и второстепенные линии должны быть простыми и не бросаться в глаза;

− там, где возможно, использовать только горизонтальные надписи на осях;

− для графиков с областями предпочтительнее использовать цвет с прозрачностью;

− для каждой категории на графике использовать свой цвет.

Представления информации в четырех и более измерениях недоступны для человеческого восприятия. Однако разработаны специальные методы для отображения и восприятия человеком такой информации.

Наиболее известные способы многомерного представления информации:

− параллельные координаты;

− «лица Чернова»;

− лепестковые диаграммы.

Рассмотрим более подробно визуализатор «**лица Чернова**». Это схема визуального представления многофакторных данных в виде человеческого лица. Каждая часть лица: нос, глаза, рот – представляет собой значение определенной переменной, назначенной для этой части. Анализ информации при помощи такого способа отображения основан на способности человека интуитивно находить сходства и различия в чертах лица.

Итак, каждое лицо – это массив элементов, каждый из которых принимает значение от 0 до 1. Значению соответствует внешний вид соответствующей части лица. Параметры исследуемых объектов приводятся к этим значениям. Экстремумы реальных данных будут приняты как 0 и 1, все остальное – как лежащее в этом промежутке. По полученному массиву конструируется лицо.

Добавление асимметрии, позволило увеличить вдвое количество переменных (рис. 31).

Рис. 31. «Лицо Чернова»

Асимметрия позволяет рассматривать объекты в прогрессе. На рис. 32 показаны различные параметры у пациентов, к которым применялось лечение. Левая сторона лица показывает значения параметров до, а правая – после лечения. Легко можно понять, кому и насколько стало лучше, даже не вникая в сущность исследуемых параметров [11].

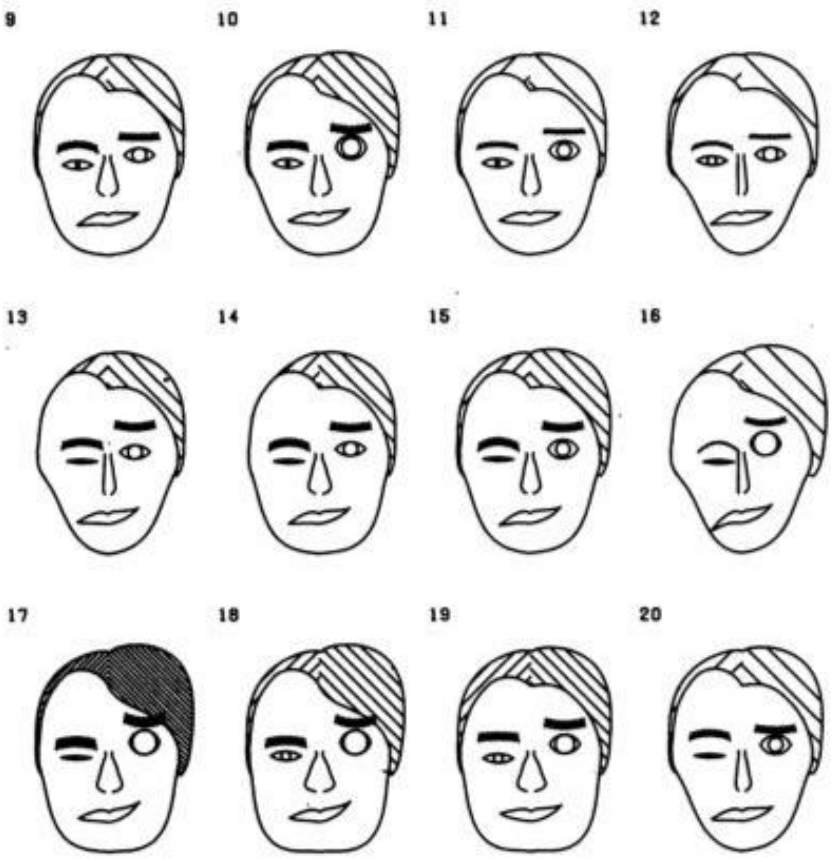


Рис. 32. Использование «лиц Чернова»

3.5. Этапы процесса Data Mining

Традиционный процесс Data Mining включает следующие этапы.

1. Анализ предметной области

Предметная область – это мысленно ограниченная область реальной действительности, подлежащая описанию или моделированию и исследованию. Она состоит из объектов, различаемых по свойствам и находящихся в определенных отношениях между собой или взаимодействующих каким-либо образом.

Исследователю необходимо уметь выделить существенную их часть. Например, при решении задачи «Выдавать ли кредит?» важными являются все данные про частную жизнь клиента. Для решения другой задачи банковской деятельности эти данные будут абсолютно неважны.

В процессе изучения предметной области должна быть создана ее модель. Знания из различных источников должны быть формализованы. Это могут быть текстовые описания предметной области или специализированные графические нотации. Модель предметной области описывает процессы, происходящие в предметной области, и данные, которые в этих процессах используются.

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.

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. 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.

Клиент покупает товар *A*. При появлении нового товара *B* клиент уже не покупает товар *A*, а покупает только товар *B*. Появление товара *B* изменило поведение клиента. Динамика поведения объектов часто описывается вместе со статикой.

3. Подготовка данных

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.

Данный этап включает в себя:

а) определение и анализ требований к данным.

На этом этапе осуществляется моделирование данных, т. е. определение и анализ требований к данным, которые необходимы для осуществления Data Mining. При этом изучаются вопросы распределения пользователей, вопросы доступа к данным, а также аналитические характеристики системы (измерения данных, основные виды выходных документов, последовательность преобразования информации и др.);

б) сбор данных.

Наличие в организации ХД делает анализ проще и эффективней. Однако далеко не все предприятия оснащены хранилищами данных. В этом случае источником для исходных данных являются существующие информационные системы, внешние источники, бумажные носители, а также знания экспертов или результаты опросов.

На этом этапе осуществляется кодирование некоторых данных. Например, уровень дохода может 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.

При определении необходимого количества данных следует учитывать, являются ли данные упорядоченными или нет.

Если данные упорядочены (временные ряды), желательно знать, включает ли такой набор данных сезонную/цикличную компоненту. Если данные не упорядочены, в ходе сбора данных следует соблюдать следующие правила:

− недостаточное количество записей в наборе данных может стать причиной построения некорректной модели. Возможно, некоторые данные являются устаревшими или описывают какую-то нетипичную ситуацию, и их нужно исключить из БД;

− 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;

в) предобработка данных.

Данные, полученные в результате сбора, должны соответствовать определенным критериям качества.

Качество данных – это критерий, определяющий полноту, точность, своевременность и возможность интерпретации данных.

Данные высокого качества – это полные, точные, своевременные данные, которые поддаются интерпретации. Данные низкого качества называют «грязными» данными.

«Грязные» данные – это отсутствующие, неточные или бесполезные данные с точки зрения практического применения. Они 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.

Первый этап очистки данных происходит еще в системе ETL. Однако 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

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).

С пропущенными значениями можно поступить следующим образом:

- 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

Дубликатами называются записи с одинаковыми значениями всех атрибутов. В большинстве случаев, продублированные данные являются результатом ошибок при подготовке данных.

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. Они 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 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.

Таким образом, наличие грязных 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.

4. Построение модели

Моделирование представляет собой построение модели и изучение ее свойств, которые подобны наиболее важным, с точки зрения аналитика, свойствам исследуемых объектов.

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.

Классификация типов моделей в зависимости от характерных свойств, присущих изучаемому объекту или системе:

− динамические (системы, изменяющиеся во времени) и статические;

− стохастические и детерминированные;

− непрерывные и дискретные;

− линейные и нелинейные;

− статистические; экспертные; модели, основанные на методах Data Mining;

− прогнозирующие (классификационные) и описательные (дескриптивные).

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.

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.

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.

Если модель достаточно сложна, а значит, требуется много времени на ее обучение и последующую оценку, то иногда можно построить и протестировать модель на небольшой части выборки. Однако этот вариант подходит только для однородных данных. 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. Если результаты полученной модели эксперт считает неудовлетворительными, следует вернуться на один из предыдущих шагов процесса Data Mining.

6. Применение модели.

На этом этапе выбранная модель используется применительно к новым данным с целью решения поставленных задач.

7. Коррекция и обновление модели.

По прошествии определенного промежутка времени с момента начала использования модели следует проанализировать полученные результаты, определить, действительно ли она эффективна.

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. 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.

Литература

1. *Вольфсон, М. Б.* Базы данных : учеб. пособие / М. Б. Вольфсон ; СПбГУТ. – СПб., 2008.

2. *Kimball, R.* The Data Warehouse Toolkit : Practical Techniques for Building Dimensional Data Warehouses / R. Kimball. – John Wiley & Sons, 1996.

3. *Kimball, R.* The Data Webhouse Toolkit: Building the Web-Enabled Data Warehouse / R. Kimball. – John Wiley & Sons, 2000.

4. *Inmon W. H.* Building the Data Warehouse, QED/Wiley, 1991, 312 p.

5. От хранения данных к управлению информацией / EMC. – СПб. : Питер, 2010.

6. *Codd, E. F.* Providing OLAP (on-line analytical processing) to user-analysts: An IT mandate / E. F. Codd, S. B. Codd, and C. T. Salley. – Technical report, 1993.

7. *Исаев, Д. В.* Аналитические информационные системы : учеб. пособие / Д. В. Исаев. – М. : ГУ–ВШЭ, 2008.

8. *Чубукова, И. А.* Data Mining [Электронный ресурс] / И. А. Чубукова. – НОУ ИНТУИТ, 2006.

9. *Паклин, Н. Б.* Бизнес-аналитика: от данных к знаниям (+CD) : учеб. пособие / Н. Б. Паклин, В. И. Орешков. – 2-е изд., испр. – СПб. : Питер, 2013.

10. *Желязны, Д.* Говори на языке диаграмм : пособие по визуальным коммуникациям для руководителей / Д. Желязны ; пер. с англ. – М. : Институт комплексных стратегических исследований, 2004.

11. *Flury, B.* Graphical Representation of Multivariate Data by Means of Asymmetrical Faces / B. Flury, H. Riedwyl // Journal of the American Statistical Association. – 1981.

Вольфсон Михаил Борисович

АНАЛИЗ ДАННЫХ

Учебное пособие

Редактор *Л. К. Паршина*

Компьютерная верстка *Н. А. Ефремовой*

План издания 2015 г., п. 112

Подписано к печати 04.08.2015
Объем 5,25 усл.-печ. л. Тираж 30 экз. Заказ 590

Редакционно-издательский отдел СПбГУТ
191186 СПб., наб. р. Мойки, 61

Отпечатано в СПбГУТ