В.В. Братищенко

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В БИЗНЕС-АНАЛИТИКЕ

Учебное пособие

Министерство науки и высшего образования Российской Федерации
Байкальский государственный университет

В.В. Братищенко

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В БИЗНЕС-АНАЛИТИКЕ

Учебное пособие

Иркутск
Издательство БГУ
2019

УДК 681.3
ББК 32.973
Б87

Печатается по решению редакционно-издательского совета
Байкальского государственного университета

Рецензенты канд. физ.-мат. наук, доц. В.В. Ступин
канд. техн. наук А.В. Родионов

Братищенко В.В.
Б87 Информационные технологии в бизнес-аналитике [Электронный ресурс] : учеб. пособие / В.В. Братищенко. — Иркутск : Изд-во БГУ, 2019. — 128 с. — Режим доступа: http://lib-catalog.bgu.ru.

В пособии описаны наиболее распространенные технологии и инструменты анализа данных. Прежде всего, это многомерный анализ данных OLAP — On-Line Analytical Processing. Изучение изменчивости показателей вдоль выбранных осей позволяет выявить и визуально оценить характер зависимостей показателей. Такой разведочный анализ является основой для формулировки предположений о наличии различных зависимостей. В пособии представлены наиболее распространенные задачи исследования зависимостей: классификация, кластеризация, регрессия, построение моделей временных рядов, выявление ассоциаций и последовательностей. Каждая из перечисленных задач решается с помощью набора моделей. Практическое решение задач проиллюстрировано с помощью аналитических служб SQL сервера Микрософт и соответствующей надстройки табличного процессора MS Excel.

Предназначено для студентов, обучающихся по программе магистратуры по направлению «Прикладная информатика в экономике».

УДК 681.3
ББК 32.973

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

Современное состояние автоматизации управления экономическими процессами характеризуется практически полным компьютеризированным учетом. Описание всех значимых событий в экономическом объекте хранится в компьютерной системе, как правило, в соответствующих таблицах баз данных. Это позволяет оперативно формировать отчеты, характеризующие различные стороны протекающих процессов.

Управление на основе бумажных отчетов представляется достаточно архаичным. Например, после обнаружения в бумажном отчете увеличения продаж некоторого товара, может появиться необходимость анализа продаж этого товара по продавцам, по датам продаж или по торговым точкам. Конечно, это можно сделать, изучая новый отчет. Однако, во-первых, большая часть нового отчета о продажах других товаров окажется ненужной, во-вторых, нужный отчет может быть не реализован в программной системе. Все это привело к появлению технологии многомерного анализа (OLAP — On-Line Analytical Processing), которая позволяет оперативно на экране компьютера получать сведения о влиянии на показатели одного фактора или произвольных комбинаций факторов.

Функции многомерного анализа успешно заменяют множество отчетов: можно выбрать показатели и факторы, влияющие на показатели, и сразу получить данные о зависимостях в табличном и графическом вариантах. Можно выполнить детализацию показателей для набора значений некоторого фактора или, наоборот, получить агрегированные значения. В результате аналитик выделяет феномены неудач или успехов, а также факторы, влияющие на появление феномена. Все это позволяет принимать обоснованные управленческие решения.

Многомерный анализ часто называют разведочным, так как в итоге аналитик получает ряды и графики, демонстрирующие изменения показателей в зависимости от выбранных факторов. Характер же этих зависимостей, закономерности влияния факторов остаются не формализованными и требуют дальнейшего изучения. Тем не менее, ряды и графики, полученные с помощью многомерного анализа, позволяют выдвигать обоснованные гипотезы о видах зависимостей. Обнаруженные зависимости можно использовать для прогноза ситуации, для расчета управляющих воздействий и их последствий. Модели исследования закономерностей объединяют под общим названием Data mining.

Технологию Data Mining достаточно точно определил Григорий Пятецкий-Шапиро (Gregory Piatetsky-Shapiro) — один из основателей этого направления: «Data Mining — это процесс обнаружения в сырых данных ранее неизвестных, нетривиальных, практически полезных и доступных для интерпретации знаний, необходимых для принятия решений в различных сферах человеческой деятельности».

Кроме термина Data Mining используется несколько близких понятий и технологий [1]:

– Извлечение знаний из баз данных (Knowledge Discovery in Databases — KDD) — процесс получения из данных знаний в виде зависимостей, правил, моделей, обычно состоящий из таких этапов, как выборка данных, их очистка и трансформация, моделирование и интерпретация полученных результатов.

– Business Intelligence — программные средства, функционирующие в рамках предприятия и обеспечивающие функции анализа информации и доступа к аналитической информации.

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

– средства построения хранилищ данных (data warehousing);

– системы многомерной аналитической обработки (On-Line Analytical Processing, OLAP);

– информационно-аналитические системы (Enterprise Information Systems, EIS);

– средства интеллектуального анализа данных (Data Mining);

– инструменты для выполнения запросов и построения отчетов (Query and Reporting Tools).

Кроме упомянутых технологий, широкое применение находят достижения науки под названием «Искусственный интеллект», которая объединяет методы и технологии воссоздания с помощью вычислительных систем и иных искусственных устройств разумных рассуждений и действий.

Таким образом, палитра современных методов анализа широка и разнообразна и представлена большим количеством соответствующих компьютерных технологий.

1. ОБЩИЕ СВЕДЕНИЯ

1.1. Готовые аналитические решения

Готовые к использованию информационные технологии являются высоко технологичными продуктами с полностью разработанными алгоритмами настройки моделей, проверки их адекватности и применения моделей для решения аналитических задач. Основную трудность при разработке аналитических технологий представляет не разработка математических методов и алгоритмов, а правильный выбор задач и моделей и грамотная интерпретация результатов моделирования.

Ниже перечислены наиболее распространенные классы аналитических инструментов.

1. Статистические пакеты, позволяющие решать наиболее распространенные задачи статистического анализа. Почти все такие программы вместе с классическими моделями математической статистики реализуют модели Data Mining. Примером может служить широко известный пакет Statistica (http://www.statsoft.ru). Применение статистических пакетов позволяет быстро выполнить аналитическое исследование. Однако, систематическое и многопользовательское использование статистических пакетов затруднено в силу их ориентации на одного пользователя — статистика. В таких пакетах слабым звеном является интеграция в единую вычислительную систему процессов обработки и накопления данных, объединяющих усилия разных коллективов пользователей и использующих разных программные системы.

2. Настольные пакеты Data Mining удобны для поиска разовых решений, сосредоточены на реализации некоторых алгоритмов и предоставляют ограниченные возможности интеграции данных и создания многопользовательских технологий, необходимых для корпоративных решений.

3. Аналитические платформы предоставляют возможности реализации всех этапов аналитической обработки: сбор данных из разных источников, предварительная обработка и очистка данных, создание хранилищ и реализация функций OLAP, создание, настройка, проверка адекватности моделей Data Mining, предоставление коллективного доступа к результатам моделирования.

4. СУБД с возможностями Data Mining предоставляют все возможности аналитических платформ, и дополнительно обеспечивают более тесную интеграцию и более высокую оперативность и производительность для «своих» источников данных.

5. Облачные технологии, реализующие модели Data Mining (IBM Watson Analytics, аналитические решения Microsoft на платформе Azure (Azure Machine Learning Service Documentation https://docs.microsoft.com/en-us/azure/machine-learning/service), сайты, предоставляющие инструменты поиска зависимостей) обладают набором привлекательных характеристик. Прежде всего, это отсутствие затрат на развертывание и поддержку работоспособности платформы. Однако, все равно необходимо реализовать технологию периодической загрузки

данных и настройки моделей. Другое достоинство таких решений — высокий уровень доступности результатов анализа по сети Интернет.

Наличие значительного количества готовых продуктов создает хорошие условия для разработки аналитических технологий поддержки принятия решений с минимальными затратами. Главной задачей при этом является не разработка программного обеспечения, а применение готовых компонентов для создания информационно-аналитических систем. Для этого важно организовать совместную работу принимающих решения менеджеров и ИТ-специалистов. Менеджер (эксперт) определяет цели, а также, назначение и порядок применения аналитических технологий, программист (аналитик) создает технологии сбора, очистки и обработки данных. Совместно они должны оценить применимость и качество результатов аналитики. В случае положительной оценки наиболее полезные модели реализуются в информационной системе и внедряются в практику управления.

Обычно внедрение аналитических информационных технологий происходит от разовых, пробных применений таких технологий для решения частных задач к систематической обработке данных. В технологическом плане это соответствует первоначальному применению частных и локальных решений с последующим переходом к систематической обработке на основе сетевых технологий. Независимо от выбранной архитектуры аналитических информационных технологий разработка таких проектов требует глубокого понимания решаемых задач и получаемых результатов.

1.2. Разработка проекта информационной аналитической технологии

Рекомендации по разработке аналитических технологий [1–5] при всем внешнем различии описывают сходные методики. В частности, стандарт CRISP (Cross-Industry Standard Process for Data Mining) [4] включает следующие этапы:

- постановку задачи (Business Understanding);
- определение данных для решения задачи (Data Understanding);
- подготовка данных (Data Preparation);
- построение модели (Modeling);
- оценка полезности модели (Evaluation);
- развертывание информационной аналитической технологии (Deployment).

Постановка задачи должна начинаться с определения того, для чего нужно применять аналитические технологии. Для ответа на этот вопрос менеджер выделяет проблемы, которые нужно решить, цели, которые нужно достичь, задачи, решение которых позволяет достичь поставленные цели, а ИТ-специалист определяет целесообразно ли для этого применять информационные технологии и какой должна быть архитектура аналитического обеспечения управления. В результате получается концептуальное представление о структуре и содержании проекта.

Определение данных для решения задачи заключается в описании результата решения и исходных данных, необходимых для получения этого результата. Необходимо выяснить из каких источников могут быть получены исходные данные, насколько эти данные полны, точны и достоверны. При этом важно исследовать образцы исходных данных, выделить типы данных, возможные значения, изучить данные на предмет ошибок и пропусков, определить необходимые уровни детализации данных.

Как правило, для анализа требуется выполнить предварительную **подготовку данных**. Кроме собственно выбора данных, при этом могут потребоваться различные вспомогательные преобразования. Достаточно часто требуется выполнить агрегирование данных. Выбросы в данных (например, случайная сверхкрупная продажа), ошибки в данных, отсутствие некоторых значений могут сильно исказить результаты анализа, поэтому требуется очистка данных. Если данные собираются из нескольких источников, то часто требуется преобразование форматов и приведение данных к единой системе справочников и шкал.

Построение модели — наиболее сложный этап разработки проекта. Одни и те же задачи могут решаться разными методами, и каждый метод имеет свою область применения, достоинства и недостатки. Поэтому на начальном этапе выбирают не одну, а несколько возможных моделей. Для каждой модели выполняется подбор параметров, оценка адекватности, достоверности и других характеристик модели. Как правило, уже на этом этапе часть моделей отбраковывается.

Оценка полезности модели заключается в тестировании ее применения для решения задач управления. Например, технологии исследования потребительской корзины могут оказаться неактуальным, если «устойчивые наборы» товаров составляют незначительную долю в товарообороте.

В случае положительной оценки полезности модели приступают к **развертыванию информационной аналитической технологии**. Необходимо создать технологию обновления исходных данных для анализа с определенной периодичностью, включить в нее пересчет параметров и характеристик моделей (перенастройку моделей), определить пользователей и порядок доступа к результатам моделирования, установить процедуры мониторинга информационной аналитической технологии.

Современные представления о жизненном цикле информационного проекта характеризуются, во-первых, гибкостью — возможностью пересмотра и изменения решений, принятых на предыдущих этапах проектирования, во-вторых, спиральностью разработки — по итогам внедрения и эксплуатации инициируется новый проект, расширяющий или модифицирующий функции предыдущего. Это особенно актуально для аналитических технологий, в которых полученные закономерности открывают новые возможности для применения и развития.

Решение на основе современных серверных технологий получается более сложным и дорогостоящим, чем разовое решение задачи выявления зависимости. Однако, в результате получается комплекс информационных процедур, охватывающий и выполняющий автоматически все апробированные и настроенные в процессе проектирования функции по аналитической обработке данных.

При этом не требуется больших затрат на поддержание аналитических технологий в работоспособном и актуальном состоянии.

1.3. Данные для анализа

Формулировка проблемы или задачи, для решения которых нужны аналитические технологии, требует привлечения знаний из нескольких областей. Прежде всего, необходимо понимание экономики изучаемых бизнес-процессов. Кроме точного понимания организации процесса, совокупности событий и явлений, связанных с последовательностью операций по исполнению процесса и управления им, важную роль играют экономические показатели. Именно они служат индикаторами и измерителями успешной реализации планов, эффективности работы, полноты использования ресурсов и т.д. Каждый процесс характеризуется большим набором показателей и факторов, влияющих на их значения.

Показатели также могут быть связаны друг с другом. Связь может быть простой, функциональной, определяющей как значения одних показателей вычисляются по другим показателям. Например, стоимость продажи вычисляется как произведение цены на количество. В более сложном случае зависимости показателей имеют вероятностный характер, например, влияние стимулирующих надбавок на качество труда и на процент брака.

Достижение целей управления измеряется соответствующим набором показателей. Обычной ситуацией при этом является сложность и противоречивость системы целей. Например, повышение качества товаров и услуг чаще всего связано с увеличением затрат на проведение соответствующих мероприятий. Это вступает в противоречие с целью снижения издержек. Поэтому важно соблюдение баланса целей. Это составляет сущность известной методологии стратегического управления, названной авторами сбалансированной системой показателей [6].

С каждым показателем связано некоторое число признаков, характеризующих, с одной стороны, множество данных по которым определено значение показателя, с другой стороны, влияние на показатель определенных факторов. Например, доход фирмы может быть определен за определенный год и для некоторого вида деятельности, т.е. с доходом связали два признака — временной интервал и вид деятельности. Выбор вида деятельности определяет данные, по которым рассчитывается доход. В то же время изучая зависимость дохода от вида деятельности можно оценить влияние этого фактора. Изучение зависимости дохода от времени, позволяет обнаружить сезонные колебания. Изучение совместного влияния вида деятельности и времени позволяет представить изменение во времени вклада в доход разных видов деятельности.

С точки зрения обработки данных нужно в процессе их регистрации запоминать значения показателей и соответствующих признаков. Например, фиксируя факт получения дохода, необходимо связать с ним вид деятельности и время получения. На самом деле в базы данных обычно вводится более подробная информация. Не просто величина дохода, а количество и цена проданных товаров

или оказанных услуг, не просто вид деятельности, а точное указание по прейскуранту за что были уплачены деньги. Дополнительно вводятся и другие сведения о факте получения дохода. Поэтому с каждым показателем можно связать много признаков. Чем подробнее данные в компьютерной системе, тем больше возможностей для анализа, поскольку сложно сказать заранее, как влияют на выбранный показатель отдельные признаки или их комбинации.

Объем исходных данных может быть очень большим. Тем не менее основные процедуры обработки остаются теми же самыми. Фиксируя значение признака (комбинацию значений признаков), получаем исходные данные для определения соответствующего значения показателя с помощью функции агрегирования. Чаще всего такой функцией является суммирование. Иногда применяется подсчет количества, вычисление среднего и других статистических характеристик. Ряд показателей вычисляется на основании других показателей. Например, средняя цена может быть вычислена как суммарная стоимость, деленная на суммарное количество.

Для снижения объема исходных данных применяют агрегирование. Например, можно агрегировать доход по часам, дням или месяцам в зависимости от назначения анализа. Если в распоряжении исследователя будут величины дохода в целом по дням, то невозможно будет выявить наиболее доходные часы. С другой стороны, поминутный доход даже в крупном супермаркете будет в значительной степени случайным и потребует усреднения для определения устойчивых закономерностей.

Таким образом, данные для анализа можно представить в виде таблицы, колонки которой можно разделить на показатели и признаки. Одна строка такой таблицы описывает значения показателей для заданного набора признаков. Например, для супермаркета показателями могут быть стоимость продаж, суммарный вес, суммарное количество, средняя цена, а признаками — дата и время, товар, классификационные признаки товара, продавец, покупатель. Одна строка такой таблицы будет описывать продажу товара указанным продавцом покупателю на определенную дату и время. Чаще всего исследуются зависимости показателей от признаков, но иногда зависимости возникают и между показателями.

При описании обработки данных строку таблицы называют записью, а значение колонки в записи полем или атрибутом. Для эффективной обработки требуется, чтобы атрибут был «простым» значением элементарного типа данных:

- число (целое, или вещественное);
- логическое значение (истина или ложь);
- дата-время;
- строка знаков, описывающая значение признака.

Информационное описание экономического показателя включает атрибут-основание, содержащий значение показателя, и атрибуты-признаки, характеризующие это значение.

Кроме перечисленных элементарных типов данных для описания множества значений и свойств атрибута используется понятие шкалы. Используют следующие виды шкал:

– дихотомическая шкала содержит только два значения (две категории);

– номинальная шкала содержит конечное множество значений, порядок которых не имеет значения, например, наименование производителя;

– порядковая шкала также содержит конечное множество значений, но в отличии от номинальной для них введено отношение порядка, примером такой шкалы могут быть оценки знаний;

– метрическая шкала, для которой определено понятие расстояния или меры, такой шкалой измеряется, например, вес или стоимость. В порядковых шкалах часто используются числа, например, оценки 2, 3, 4, 5, тем не менее эти значения не являются метрическими. Неправильно утверждать, что расстояние между «двойкой» и «тройкой» и между «четверкой» и «пятеркой» одинаковы, или что 5 «двоек» эквивалентны 2 «пятеркам». Некорректно также использовать средние значения порядковых шкал. Вместо этого необходимо использовать распределения вероятностей (частот) значений.

Для шкал, которые включают много значений, дополнительно вводят классификации. Например, для продукции в России введен Общероссийский классификатор продукции по видам экономической деятельности (ОКПД).

В ОКПД использованы иерархический метод классификации и последовательный метод кодирования. Код состоит из 2–9 цифровых знаков, и его структура может быть представлена в следующем виде:

XX	класс
XX.X	подкласс
XX.XX	группа
XX.XX.X	подгруппа
XX.XX.XX	вид
XX.XX.XX.XX0	категория
XX.XX.XX.XXX	подкатегория

Для описания классификатора в информационных системах обычно используют несколько атрибутов — по одному на каждый уровень классификации. Это удобно тем, что, зафиксировав атрибуты, указывающие, например группу «61.20 Услуги телекоммуникационные беспроводные», можно получить агрегированные значения показателя для данной группы, а добавляя атрибуты, указывающие подгруппы, можно выполнять детализацию показателя.

Одна из первых задач построения аналитических информационных технологий — это понимание набора признаков и показателей, среди которых и будут выделяться зависимости. Следующий шаг — выделение исходных данных для получения таблицы признаков и показателей. Для этого необходимо хорошо представлять структуру исходных данных, которые чаще всего представлены реляционными схемами в базах данных.

Вопросы

1. Перечислите классы программных продуктов, решающих аналитические задачи. Укажите особенности их применения.

2. Назовите этапы разработки аналитических технологий по стандарту CRISP. Определите работы, выполняемые на каждом этапе.

3. Опишите информационную структуру показателя. Значения показателей и признаки. Правила вычисления показателей. Применение функций агрегирования.

4. Перечислите и определите шкалы, используемые для атрибутов. Укажите особенности каждой шкалы.

2. МНОГОМЕРНЫЙ АНАЛИЗ ДАННЫХ

2.1. Структура многомерных данных

Анализ показателей, изучение зависимости показателей от признаков (факторов), исследование взаимного влияния показателей является основой аналитической работы. Прежде чем формулировать зависимости в аналитическом виде, необходимо эти зависимости «увидеть». Для этого изучают таблицы, отчеты, графики, демонстрирующие зависимости показателей и признаков.

Компьютерные технологии позволяют оперативно выполнять такую обработку по запросу аналитика. Для этого была предложена технология многомерного анализа данных — On-Line Analytical Processing (OLAP). Технология OLAP использует специальные структуры — хранилища данных (Data warehouses) или многомерные базы данных для обеспечения хранения, агрегирования и извлечения данных. Эта технология качественно превосходит формирование и печать отчетов: результат выдается немедленно, в удобной форме с соответствующими графиками и диаграммами. Пользователь может выбрать любое направление (комбинацию признаков) для агрегирования или детализации. Результатом многомерного анализа является выявление некоторых феноменов: отклонений, выбросов или провалов, тенденций, зависимостей.

Данные в хранилище попадают из систем регистрации данных (Online Transaction Processing — OLTP), которые предназначены для автоматизации бизнес-процессов и решают, в частности, учетные задачи. В таких системах регистрируются все значимые для управления события: продажи, отгрузки, платежи и т.д. Хранилище может пополняться не только сведениями из баз данных, но и за счет внешних источников, например, различных статистических отчетов, биржевых котировок, курсов валют. Данные в разных источниках могут иметь разные структуры, разные кодировки и форматы. Поэтому в процессе загрузки данных в хранилище приходится решать задачи по преобразованию систем классификации и форматов данных.

Технология OLTP позволяет выполнять вычисления, определять значения показателей для разных комбинаций признаков. Тем не менее целесообразно применять OLAP технологию по следующим причинам:

1. Необходим специальный инструментарий вычисления значений показателей для произвольного набора признаков и выявления зависимостей.

2. Анализировать данные оперативных систем учета напрямую невозможно или очень затруднительно: данные хранятся разрозненно в разных подсистемах, в разных СУБД, в форматах, на разных серверах корпоративной сети. Аналитическую обработку затрудняют сложность и запутанность структур хранения данных, избыточность детальной информации.

3. Сложные аналитические запросы к оперативной информации тормозят текущую работу информационной системы, надолго блокируя таблицы и захватывая ресурсы сервера.

К хранилищам данных предъявляют требования, сведенные в тест FASMI — Fast Analysis of Shared Multidimensional Information, предложенный Найджелом Пендсом (Nigel Pendse):

1. Fast (быстрый) — анализ должен производиться одинаково быстро по всем аспектам информации. Приемлемое время отклика — 5 секунд или менее.

2. Analysis (аналитичный) — должна быть реализована возможность осуществлять основные типы числового и статистического анализа.

3. Shared (разделяемый) — много пользователей должны иметь доступ к данным, при этом необходимо контролировать доступ к конфиденциальной информации.

4. Multidimensional (многомерный) — показатели должны вычисляться для произвольного набора классификационных признаков.

5. Information (информативный) — приложение должно иметь возможность обращаться к любой нужной информации, независимо от ее объема и места хранения.

В OLAP пользователь получает естественную, интуитивно понятную модель данных, представленную в виде многомерных ***кубов*** (cubes). ***Измерениями*** (dimensions) многомерной системы координат куба служат признаки анализируемого бизнес-процесса. ***Измерение*** — набор значений для идентификации некоторого свойства бизнес-процесса. Значения, «откладываемые» вдоль измерений, называются ***метками*** (members). Например, для анализа продаж измерениями могут быть товар, регион, тип покупателя, дата. Метками измерения «Товар» будут все наименования товаров из конкретной предметной области.

Измерение может иметь иерархическую структуру — задавать некоторую классификацию значений. Измерение «Товар» может включать классификацию, например, с использованием общероссийского классификатора продукции по видам экономической деятельности (ОКПД). В этом случае измерение описывается не одним, а несколькими атрибутами — по числу уровней иерархии в классификации. Стандартную иерархию образует время, для которого вводится набор из общеизвестных уровней (…, час, день, неделя, месяц, квартал, год, …). Кроме этого, с каждой меткой некоторого измерения могут связаны дополнительные свойства. Например, для даты можно определить свойство — день недели (понедельник, вторник, …).

Фиксированные значения всех измерений задают ***ячейку куба*** (cell), с которой связан набор ***показателей*** или ***мер*** (measures), количественно характеризующих процесс. Это могут быть объемы продаж в штуках, стоимости продаж, остатки на складе, издержки и т.п.

В качестве показателей в кубе, изображенном на рис. 1, использованы стоимость (верхнее число на рисунке) и количество (нижнее число на рисунке) проданных единиц, а в качестве измерений — продавцы, товары и покупатели.

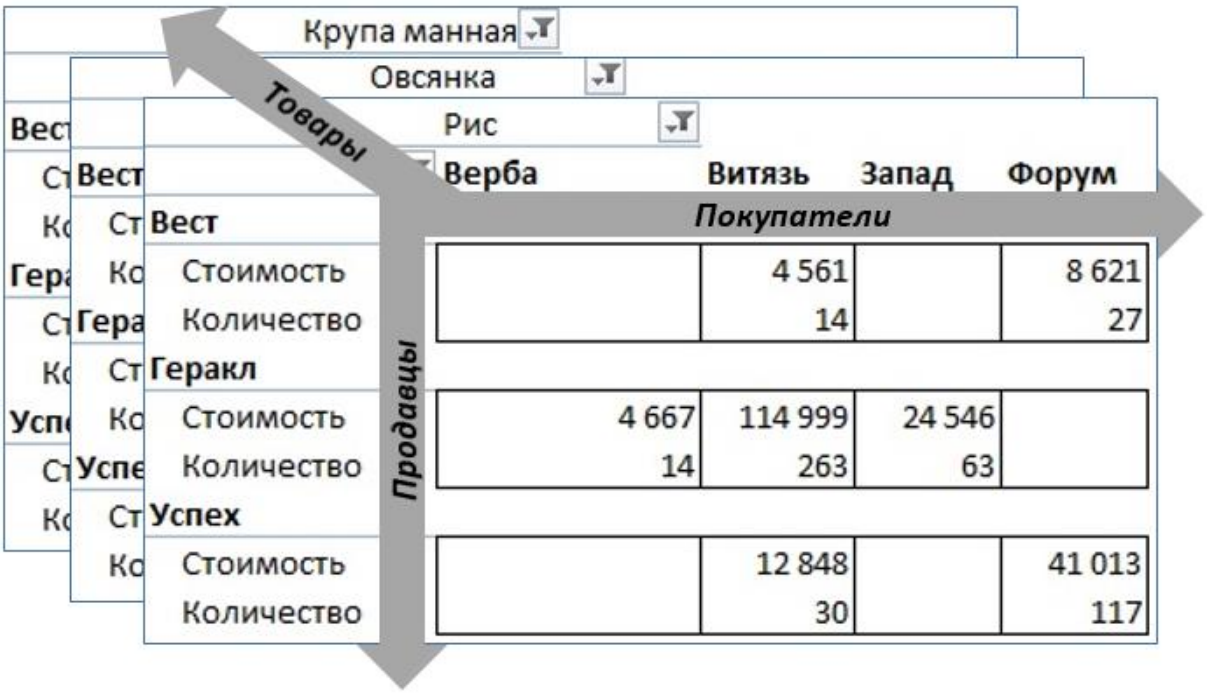

Рис. 1. Пример куба

Для каждого показателя определяется правило его вычисления. Фиксация меток измерений определяет множество данных, по которым вычисляется показатель. При этом применяются различные функции агрегирования. Наиболее распространенной функцией агрегирования является сумма. Кроме суммы применяются и другие функции агрегирования: количество (ячеек), минимум, максимум, среднее — и другие статистические характеристики.

Некоторые показатели вычисляются на основе других показателей. Например, средняя цена вычисляется как суммарная стоимость продаж, деленная на суммарное количество проданного товара. По способу вычисления показатели можно разделить на две категории: агрегированные показатели, определенные с помощью функций агрегирования по исходным данным, и вычисляемые показатели, определенные по другим показателям.

Набор показателей фактически рассматривается как одно дополнительное измерение куба: его метками являются названия показателей.

2.2. Операции над многомерными данными

Для многомерных данных вводят специальные операции исследования показателей в зависимости от выбранных комбинаций измерений. Операции над кубами снова дают куб. Одна ячейка куба — это тоже куб, только без измерений.

Формирование «Среза». ***Срез*** (Slice) — это часть куба, получившаяся в результате фиксации значений одного или нескольких измерений. Результатом среза снова является куб с меньшим количеством ячеек. Если выбирается метка, соответствующая некоторому уровню иерархии, то в результате уменьшается ко-

личество меток измерения и, соответственно, ячеек. Например, при выборе категории «Напитки» в измерении «Товары» остаются только метки различных напитков. При выборе метки конкретного товара измерение «Товары» свертывается до точки и количество измерений в срезе куба уменьшается.

Если фиксируются значения всех измерений, кроме двух, — получается обычная двумерная таблица. В этом случае горизонтальная ось (заголовки колонок таблицы) представлена метками одного измерения, вертикальная ось (заголовки строк) — метками другого, а в ячейках таблицы размещаются значения показателей. На рис. 2 изображен двумерный срез куба — зафиксировано значение «Крупа манная». Осталось два измерения — «Покупатели» и «Продавцы».

Товар = «Крупа манная»		**Покупатели**			
Продавцы		**Верба**	**Витязь**	**Запад**	**Форум**
Вест	Стоимость	5 244	33 941	5 167	
	Количество	17	67	20	
Геракл	Стоимость	53 683	35 294		
	Количество	191	69		
Успех	Стоимость				87 188
	Количество				181

Рис. 2. Двумерный срез куба для показателя «стоимость»

Операция агрегирования (Drill Up) — переход от детализированных данных к агрегированным. Агрегирование выполняется при уменьшении числа измерений или при переходе к обобщающим уровням иерархии. При этом для каждого показателя куба по множеству детальных значений нужно вычислить агрегированное значение показателя. Например, агрегирование по измерению «Товар» приведет к получению куба на рис. 3, аналогичного представленному на рис. 2. Разница будет в значениях показателей, которые для агрегирования будут вычисляться по всем товарам, а не только для товара «Крупа манная». Стоимость и количество для среза выбираются из ячеек, а для агрегирования вычисляются суммированием стоимостей и количеств проданных товаров для продавца и покупателя.

Товар = «Все»		**Покупатель**			
Продавцы	**Значения**	**Верба**	**Витязь**	**Запад**	**Форум**
Вест	Стоимость	108 038	221 158	145 868	220 280
	Количество	609	1 014	495	1 514
Геракл	Стоимость	220 082	172 823	41 417	
	Количество	1 269	479	167	
Успех	Стоимость	56 047	84 158		195 465
	Количество	369	660		782

Рис. 3. Агрегирование по товарам

Операция агрегирования приводит к уменьшению размерности, если она выполняется для всего измерения. В примере на рис. 3 в результате агрегирования исчезает измерение «Товары». Если агрегирование связано с переходом к обобщающим уровням иерархии, то соответствующее измерение не исчезает — в измерении уменьшается количество меток. При агрегировании данных по месяцам будут получены обобщающие значения показателей в каждом месяце, однако, временное измерение останется. В нем вместо трех уровней иерархии: год, месяц, день — останется два: год и месяц.

Операция детализации (Drill Down) заключается в переходе от агрегированных к детализированным данным. Детализация появляется при добавлении ранее свернутого измерения — при этом уровень агрегирования понижается: каждая ячейка куба заменяется набором ячеек по количеству меток в добавляемом измерении, а каждое значение показателя рассчитывается для новой комбинации координат измерений. На рис. 4 представлена замена агрегированных значений показателей детальными.

Детализация может быть связана с появлением свернутого измерения (на рис. 4 появляется измерение «Товары») и увеличением размерности. Детализация также может быть вызвана переходом на более детальный уровень иерархии измерения без изменения количества измерений.

Продавцы	**Товар**	**Значения**	**Покупатель**			
			Верба	**Витязь**	**Запад**	**Форум**
Вест	**Крупа манная**	Стоимость	5 244	33 941	5 167	
		Количество	17	67	20	
	Овсянка	Стоимость	3 840	50 565	2 778	69 950
		Количество	8	119	6	196
	Рис	Стоимость		4 561		8 621
		Количество		14		27

Рис. 4. Детализация агрегированных значений показателей

Работа пользователя с кубом обычно включает все перечисленные операции. К ним также относят вращение куба — перестановки осей в процессе визуализации. Первоначально менеджера интересуют агрегированные значения показателей, скажем сумма всех продаж за последний период. Затем можно выполнить детализацию этого показателя по товарам для определения товара, имеющих максимальную стоимость продаж. Возможно выполнить сечение по предыдущему периоду для сравнения с показателями текущего периода. Невозможно заранее предугадать направление анализа, тем не менее перечисленные операции позволяют реализовать любое из них.

2.3. Архитектура OLAP средств

Функции обработки многомерных данных в OLAP-приложениях могут быть разделены на три уровня:

1. Многомерное представление данных — средства конечного пользователя, обеспечивающие многомерную визуализацию, задание команд многомерного анализа (срез, агрегирование, детализация) и управление отображением получаемых данных.

2. Многомерная обработка — средство (язык) формулирования многомерных запросов (традиционный реляционный язык SQL здесь оказывается непригодным) и сервер, умеющий выполнять такие запросы.

3. Многомерное хранение — средства физической организации данных, обеспечивающие эффективное выполнение многомерных запросов и управление полномочиями доступа к многомерным данным.

Конкретные OLAP-продукты, как правило, представляют собой либо средство многомерного представления данных — OLAP-клиент (например, сводные таблицы в Excel или ProClarity фирмы Knosys), либо многомерную серверную СУБД — OLAP-сервер (например, Oracle Express Сервер или Microsoft Analysis Services).

Слой многомерной обработки обычно бывает встроен в OLAP-клиент и/или в OLAP-сервер, но может быть выделен в чистом виде, как, например, компонент Pivot Table Service фирмы Microsoft.

В решениях фирмы Microsoft основным компонентом аналитических служб является Analysis Services входящий в состав MS SQL server. Этот компонент предназначен для создания OLAP-кубов на основе реляционных хранилищ данных, а также для предоставления доступа к ним из клиентских приложений.

OLAP-куб, созданный с помощью аналитических служб Microsoft, может содержать данные, хранимые в ячейках кубов и полученные из разных источников, а также агрегированные значения, которые соответствуют уровням иерархии измерений и различным комбинациям меток измерений. Аналитические службы могут производить динамическое обновление куба, если в исходные таблицы были добавлены новые записи.

Аналитические службы сохраняют агрегированные данные только для функций агрегирования (сумма, количество экземпляров, максимум, минимум и другие). Однако, в случае необходимости можно создавать так называемые вычисляемые поазатели (calculated members) для получения других типов агрегированных значений (средних, средневзвешенных, дисперсий и т.д.). При этом, помимо применения встроенных средств создания агрегированных данных, Analysis Services позволяют использовать для вычисления агрегированных данных функции VBA или Excel, а также создавать собственные средства.

Наконец, аналитические службы Microsoft позволяют создавать, так называемые, виртуальные кубы (virtual cubes), которые в определенной степени являются аналогами представлений (view) реляционных СУБД. Виртуальные кубы не содержат данных, но позволяют представить в виде единого куба данные из нескольких кубов, имеющих хотя бы одно общее измерение.

Decision Support Objects (DSO) — это набор библиотек, содержащих COM-объекты, позволяющие создавать и модифицировать многомерные базы данных и содержащиеся в них объекты (кубы, коллективные измерения и т.д.). Эти библиотеки можно использовать для разработки собственных приложений, в которых осуществляется создание или модификация многомерных баз данных, в том числе и для реализации действий, не предусмотренных аналитическими службами.

Приложения в операционной системе MS Windows, предназначенные для чтения OLAP-данных, при взаимодействии с аналитическими службами обязательно используют PivotTable Service — библиотеки, загружаемые в адресное пространство клиентского приложения. Эти библиотеки автоматически устанавливаются вместе с аналитическими службами (независимо от того, какая именно их часть установлена — клиентская или серверная), а также вместе с Microsoft Office. В состав Microsoft SQL Server входит также инсталляционное приложение для установки PivotTable Service на компьютер, на котором не установлены эти службы.

Для взаимодействия с PivotTable Service клиентское приложение может использовать OLE DB for OLAP — расширение универсального механизма доступа к данным OLE DB, позволяющее обращаться к многомерным данным, а также ADO MD — библиотеки, представляющие собой надстройку над OLE DB for OLAP и являющиеся COM-серверами для доступа к многомерным данным, удобными для применения в клиентских приложениях.

Отметим, что спецификация OLE DB for OLAP является открытой. Это означает, что можно создавать и другие OLAP-серверы, поддерживающие OLE DB for OLAP (либо разрабатывать OLE DB-провайдеры к уже имеющимся OLAP-средствам), а также создавать клиентские приложения, обращающиеся к любым таким источникам данных с помощью PivotTable Service, OLE DB for OLAP и ADO MD.

Из других клиентских приложений, не входящих в состав аналитических служб, но часто используемых для просмотра OLAP-кубов, следует назвать приложения Microsoft Office, в частности Microsoft Excel. С помощью Excel можно обращаться к серверным OLAP-кубам, получая их двух- и трехмерные сечения или проекции на листах рабочих книг Excel в виде сводных таблиц, а также создавать локальные OLAP-кубы в виде файлов на основе реляционных данных, доступных с помощью OLE DB.

Кроме того, в состав Microsoft Office Web Components входит элемент управления ActiveX PivotTable List, позволяющий реализовать сходную функциональность как в обычном Windows-приложении, так и на HTML-странице, предназначенной для применения внутри корпоративной сети.

2.4. Создание многомерных баз данных в MS SQL Server Management Studio

Программа Management Studio позволяет выполнять все функции по администрированию многомерных баз данных. Для создания хранилища следует подключиться к аналитическим службам MS SQL сервера, выбрав команду «Соединить» \ «Службы Analysis Services» окне обозревателя объектов и указав имя сервера. После этого (при наличии полномочий) можно выполнять все необходимые команды для администрирования и доступа к данным. Реализация OLAP-технологии начинается с создания многомерной базы данных. Это можно сделать, выбрав команду «Создать базу данных» в контекстном меню списка многомерных баз данных. Для многомерной базы определяется имя базы данных и другие ее параметры. Далее в самой базе определяются источники данных для кубов, измерения и сами кубы. Для этого используются инструменты SQL Server Data Tools (SSDT), включенные в среду разработки Microsoft Visual Studio.

В среде Microsoft Visual Studio создается проект командой «Создать» \ «Проект» с последующим выбором среди шаблонов Business Intelligence «Проект многомерных данных и интеллектуального анализа» с заданием имени и расположения файлов проекта (рис. 5).

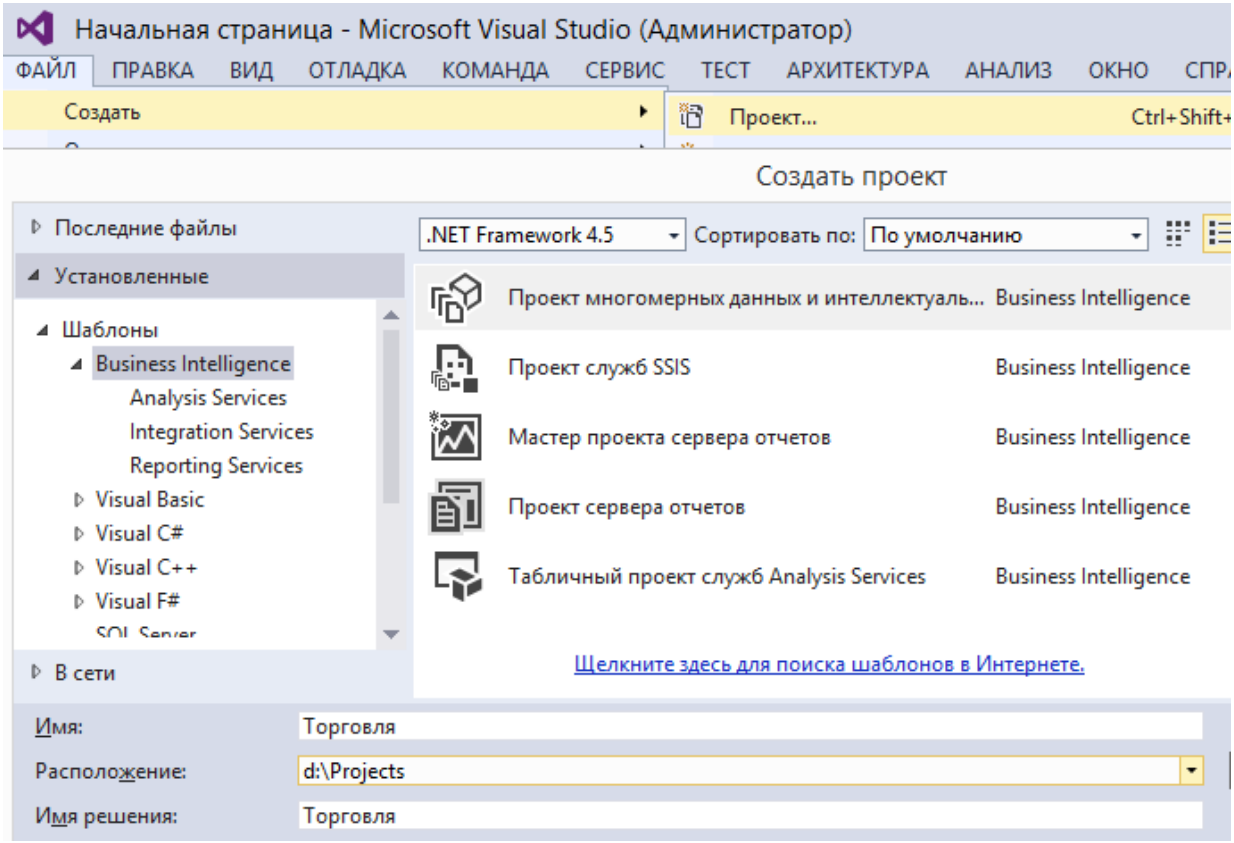

Рис. 5. Создание проекта многомерной базы данных

Далее для определения компонентов будет использоваться обозреватель решений (рис. 6) и команды контекстного меню.

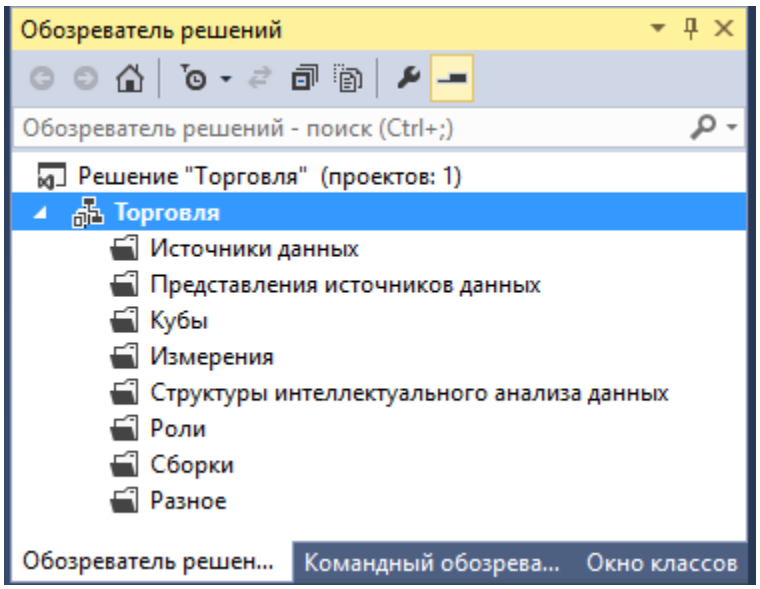

Рис. 6. Обозреватель решений проектов

Одно из основных свойств проекта (свойства проекта открывает одноименная команда контекстного меню) — это параметры развертывания (рис. 7): имя сервера и многомерная база данных. После определения проекта все компоненты его будут развернуты в указанной многомерной базе данных. Отделение проекта и структур многомерной базы данных позволяет неоднократно развертывать многомерные базы данных в различных серверах. Для этого достаточно изменить сервер в параметрах развертывания и выполнить развертывание заново.

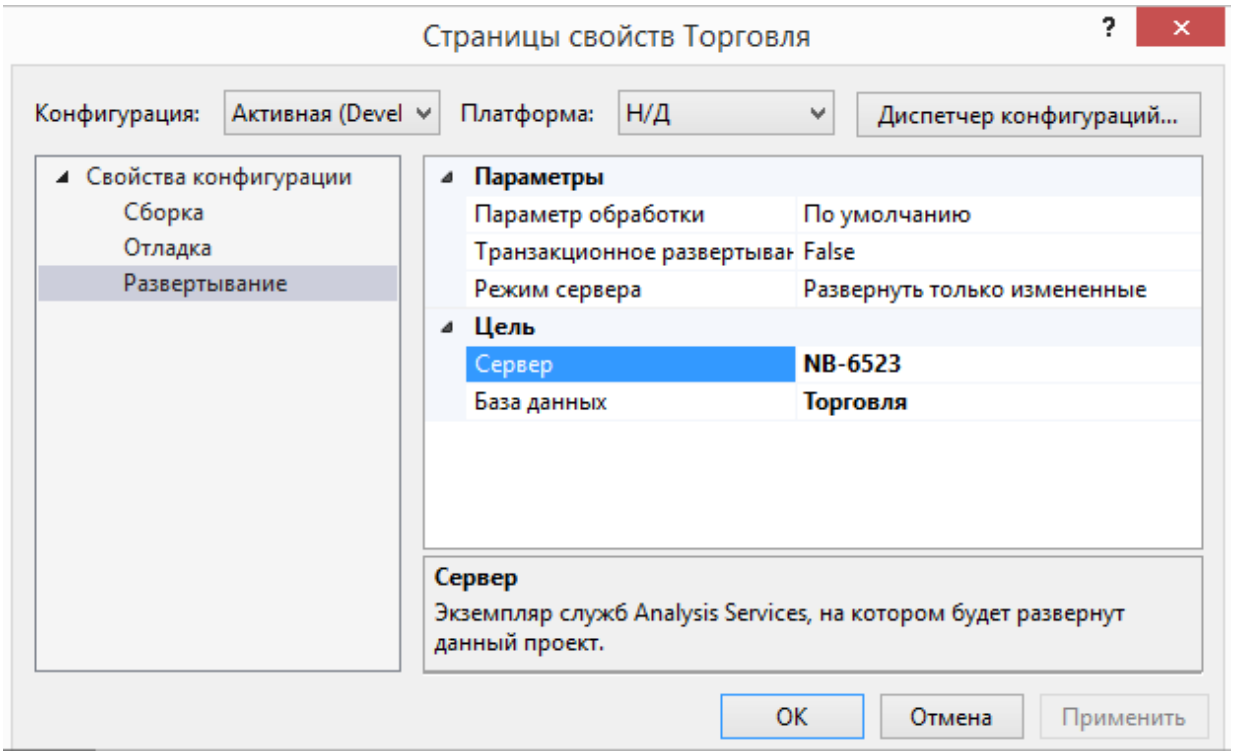

Рис. 7. Параметры развертывания проекта

2.5. Источники и представления исходных данных

Определение кубов начинается с описания источников исходных данных для них. Для описания источника данных, размещенного на MS SQL сервере, нужно выбрать из контекстного меню элемента «Источники данных» команду «Создать источник данных» и с помощью мастера (Рис. 8) определить поставщика данных (на Рис. 8 это — «Собственный поставщик OLE DB\SQL Server Native Client 11.0») и в случае сервера баз данных указать имя сервера (NB-6523) и базы (Торговля).

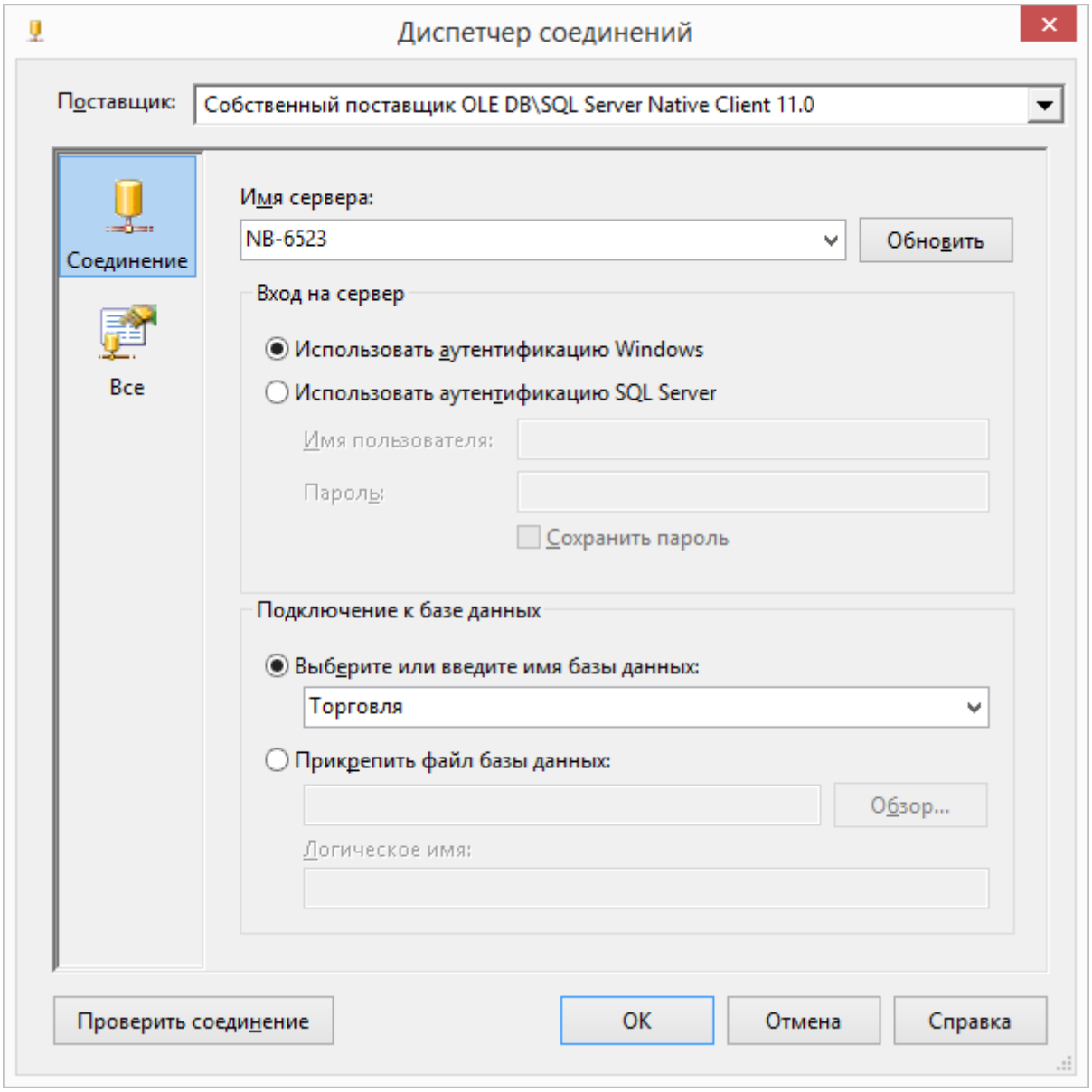

Рис. 8. Определение источника данных

Далее необходимо задать представление источника данных (или несколько представлений). Команда «Создать представление источника данных…» контекстного меню запускает соответствующий мастер, позволяющий выбрать таблицы со значениями показателей и метками измерений. Представление (рис. 9)

имеет вид подсхемы реляционной базы данных с указанием ограничений ссылочной целостности. Для примера с рис. 9 таблица «ТоварыВдоговорах» содержит описание продаж, по которому можно вычислить основные показатели «Количество», «Вес», «Стоимость» проданного товара. Другие таблицы можно использовать для построения измерений «Товары», «Продавцы», «Покупатели», «Договоры».

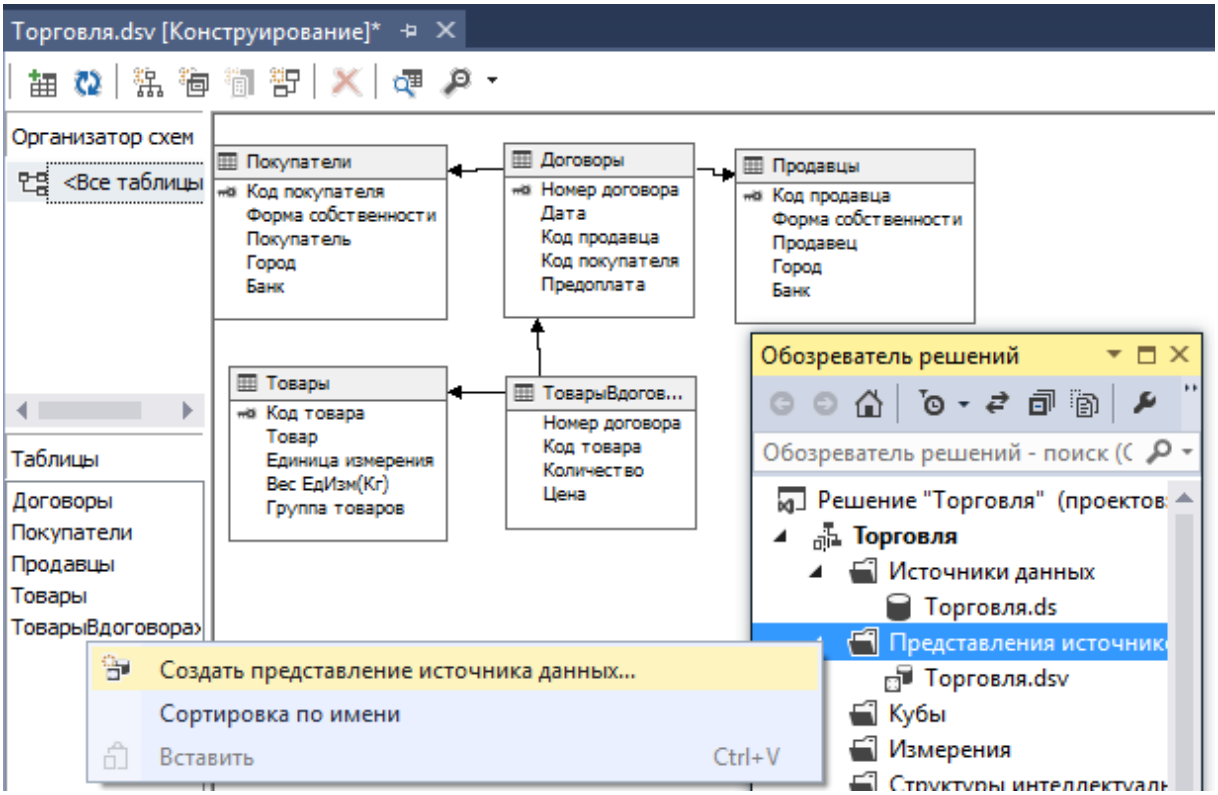

Рис. 9. Представление источника данных

Обычно таблицы реляционной БД не содержат всех показателей — их нужно вычислить. Для вычисления стоимости нужно создать (рис. 10) в представлении таблицы «ТоварыВдоговорах» именованное вычисление соответствующей контекстной командой.

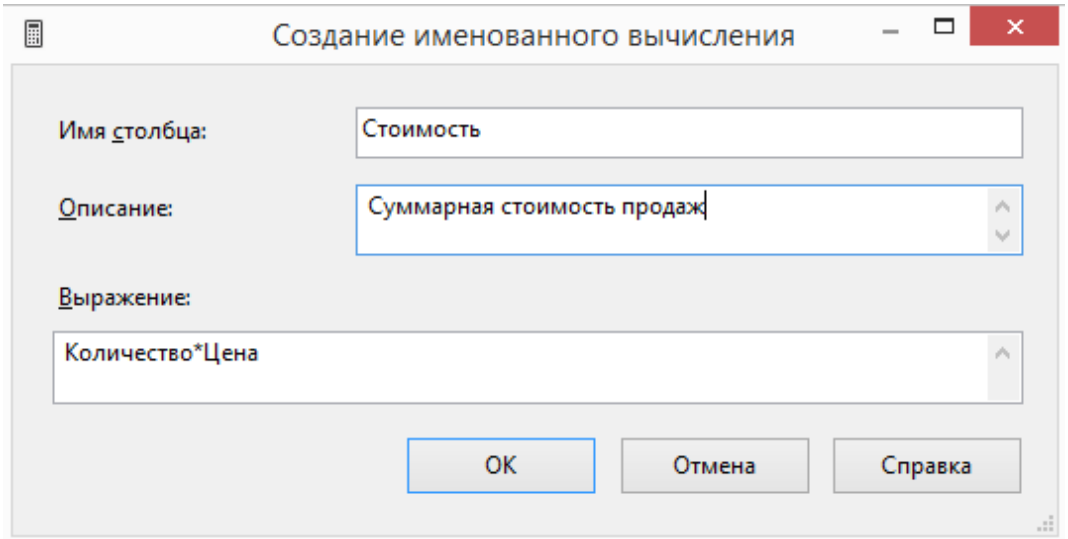

Рис. 10. Создание именованного вычисления

Аналогично создадим вычисляемое поле «Вес», определяемое выражением

«Количество(SELECT [Товары].[Вес ЕдИзм(Кг)] FROM [Товары] WHERE [dbo].[Товары].[Код товара] = [dbo].[ТоварыВдоговорах].[Код товара])».*

В таблице «Договоры» определим вычисление полей «Год», «Месяц», «День» выражениями «Year(Дата)», «Month(Дата)», «Day(Дата)». Любую таблицу можно просмотреть командой «Просмотр данных» контекстного меню таблицы и проконтролировать правильность вычислений.

В представлении можно выделить таблицу фактов (Fact Table), которая содержит значения показателей (или данные для их вычисления) для комбинации меток измерений, обычно представленных кодами. В примере с рис. 9 такой таблицей является «ТоварыВдоговорах». Данные для каждого измерения берутся из соответствующего справочника (Dimension Table), в котором значению кода соответствует метка измерения и другие атрибуты. Определение кубов начинается с создания измерений.

2.6. Создание измерений

Измерения создаются мастером, который запускается командой «Создать измерение» для вкладки «Измерения». В мастере выбирается метод создания — на основе существующей таблицы. Затем выбирается таблица, ключевые столбцы, столбец с именем метки (рис. 11) и атрибуты метки (рис. 12). В завершение работы мастера определяется имя измерения.

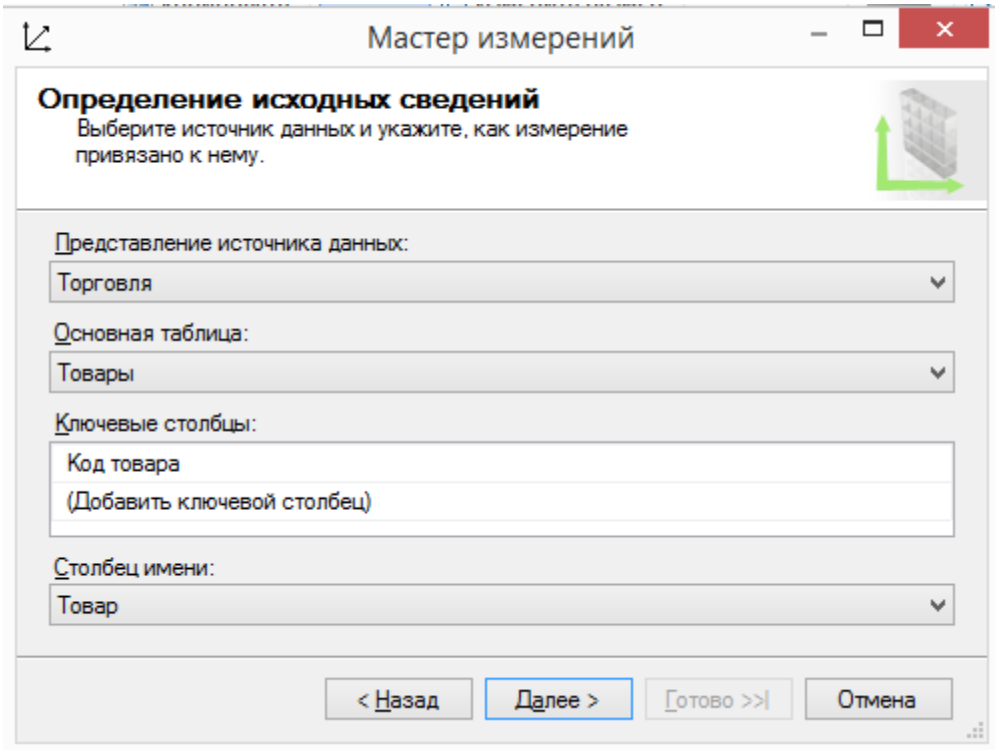

Рис. 11. Определение ключа и имени метки измерения

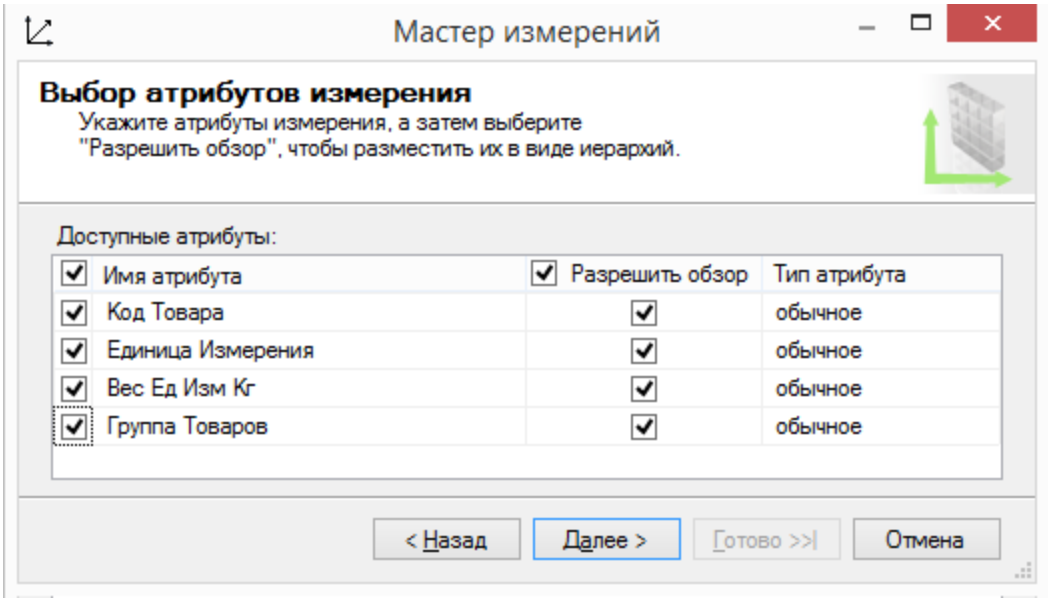

Рис. 12. Выбор атрибутов метки измерения

Для измерения можно определить одну или несколько иерархий. Например, для измерения «Товары» можно ввести двухуровневую иерархию «Группы товаров — товары». Это выполняется перетаскиванием полей в область определения иерархии (рис. 13).

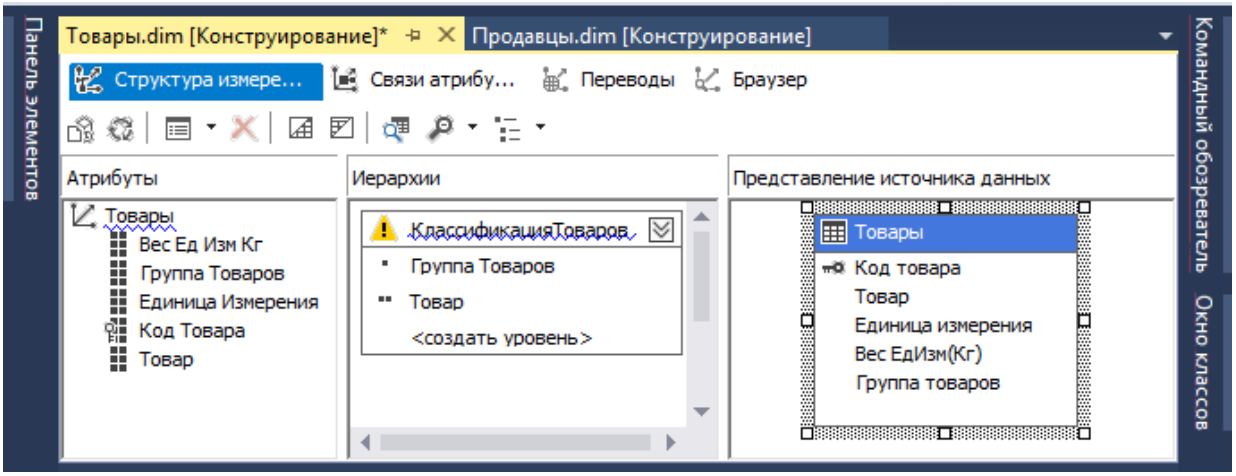

Рис. 13. Создание иерархии измерения

Возможно создание несбалансированных иерархий — типа «родитель-потомок» (parent-child). Такие иерархии нередко основаны на таблицах, где первичный ключ является одновременно и внешним ключом (например, для каждого работника указывается код его начальника).

Для появления измерения в многомерной БД необходимо выполнить развертывание (структуры из проекта переносятся в БД) и обработку (данные из источника копируются в БД) с помощью одноименных команд. После этого можно просматривать измерение (рис. 14).

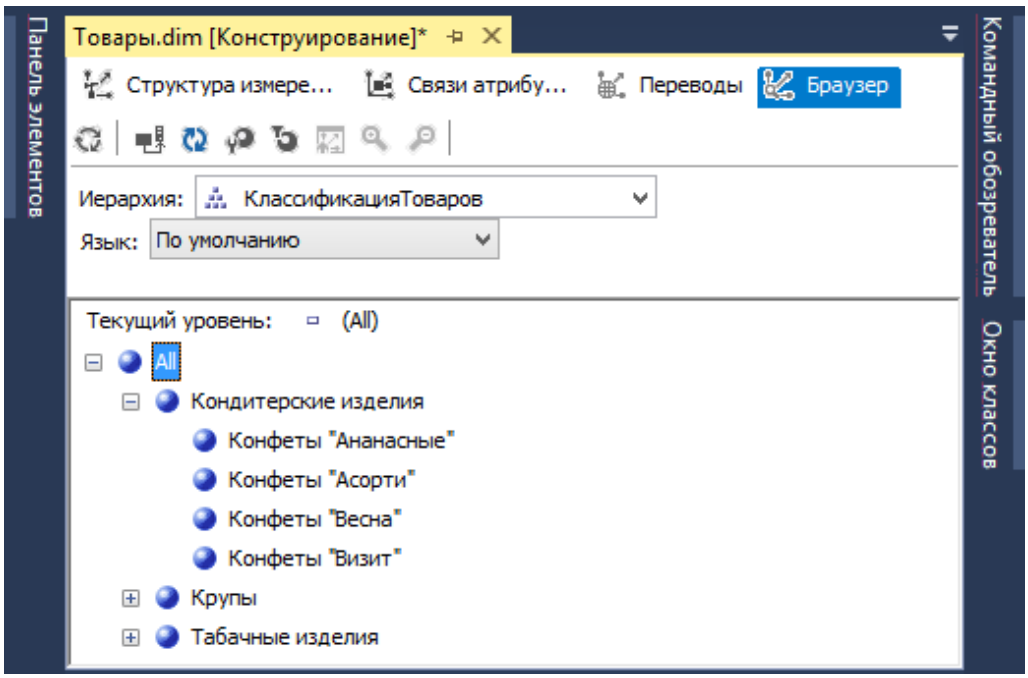

Рис. 14. Просмотр измерения

Аналогично создаются другие измерения «Продавцы», «Покупатели», «Договоры».

2.7. Определение OLAP-кубов

Куб заполняется значениями показателей на основании таблицы фактов, содержащей показатели (меры) и связанной с таблицами измерений.

Мастер создания куба (Cube wizard) включает выбор таблицы фактов (рис. 15). Для рассматриваемого примера — это таблица «ТоварыВдоговорах».

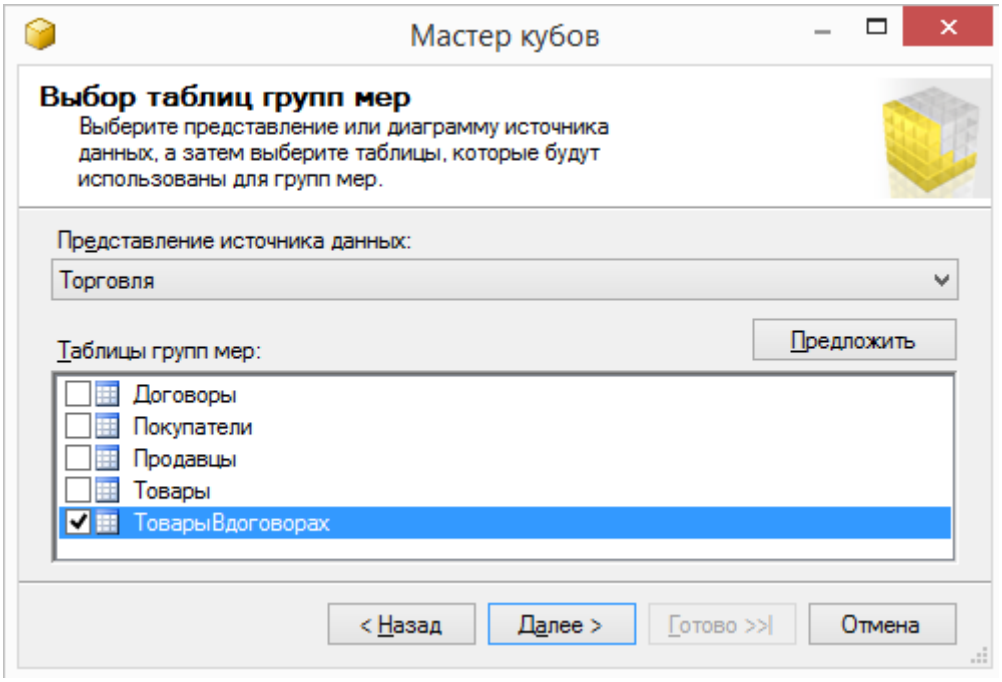

Рис. 15. Выбор таблицы фактов

Далее среди числовых полей таблицы фактов выбираются (рис. 16) показатели (меры). Среди них система предлагает дополнительно число записей — «Число ТоварыВдоговорах».

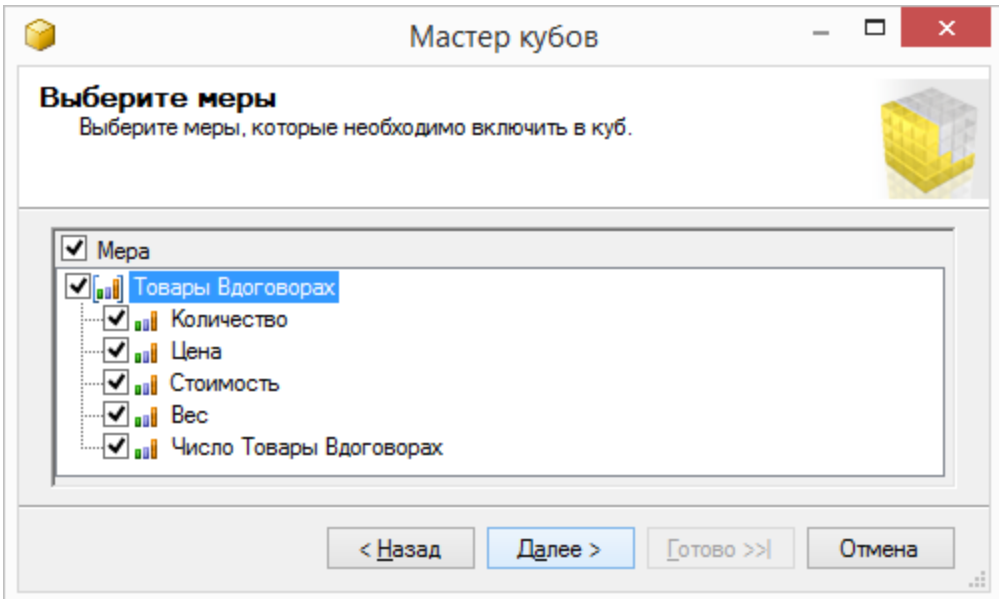

Рис. 16. Выбор показателей (мер)

После выбора показателей мастер предлагает выбрать измерения (рис. 17).

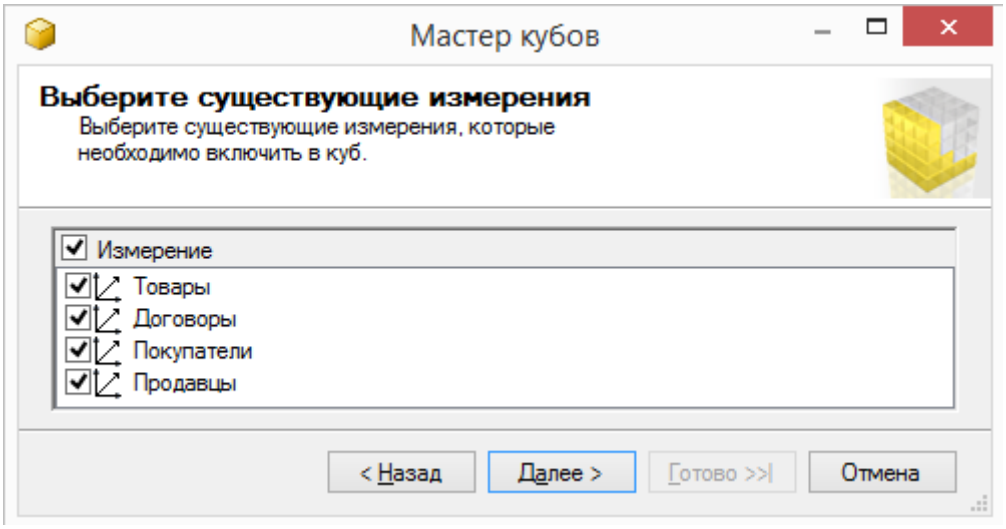

Рис. 17. Выбор измерений

Заканчивается работа мастера определением имени куба. В результате будет создано определение куба, на вкладке «Структура куба» будут представлены основные компоненты определения (рис. 18). Для каждого показателя (кроме числа записей) будет задана функция агрегирования Sum — для формирования агрегированных значений показателя, вычисляемого по множеству ячеек. Это не всегда правильно. В рассматриваемом примере сумма цен не будет суммарной стоимостью, а следовательно, будет вводить пользователей в заблуждение. Для

показателя «Цена» можно было бы задать усреднение в качестве функции агрегирования, однако и это будет некорректно. Правильно определить среднюю цену как суммарную стоимость, деленную на суммарное количество.

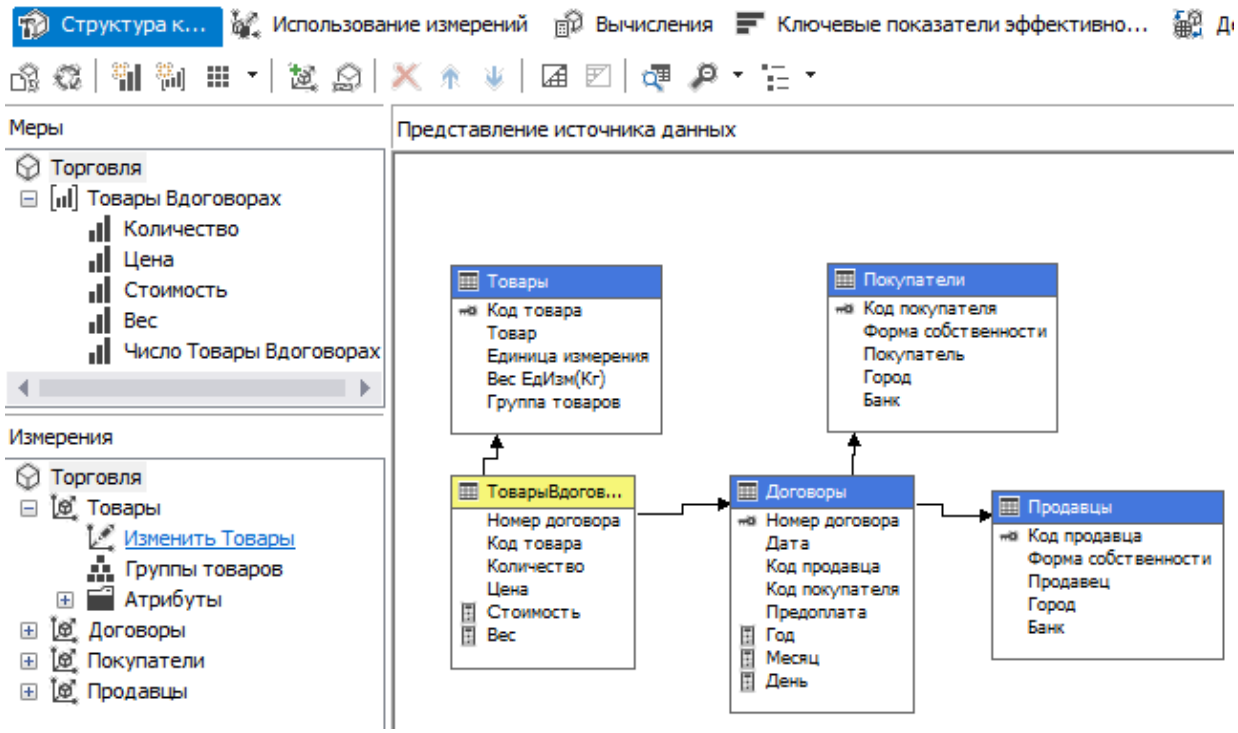

Рис. 18. Структура куба

Для задания другой функции агрегирования достаточно открыть свойства показателя. На рис. 19 показано изменение в окне свойств функции агрегирования с «Sum» на «Min» и имени показателя «Цена» на «МинЦена».

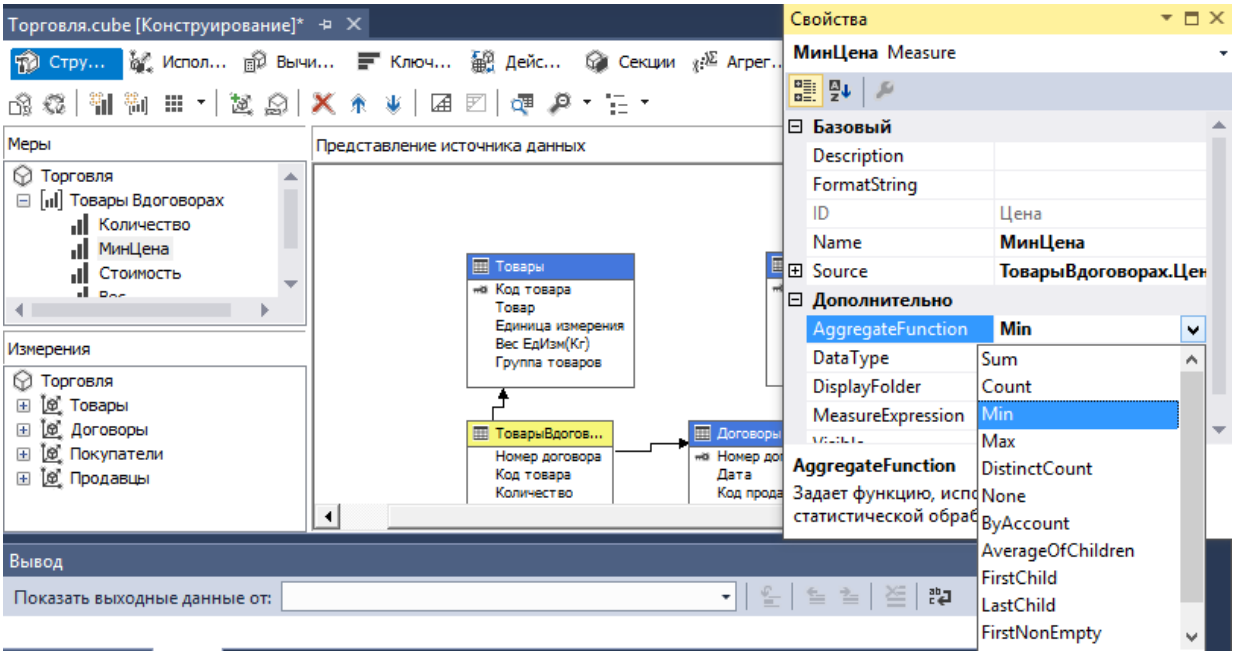

Рис. 19. Изменение свойств показателя

Для создания показателя, вычисляемого по другим показателям, нужно перейти на вкладку «Вычисления» и выполнить команду контекстного меню «Создать вычисляемый элемент». На рис. 20 показано определение средней цены в виде частного от деления суммарной стоимости на суммарное количество. Обратите внимание на идентификацию показателей в выражении. В процессе определения куба формируется специальное измерение — [Measures], которое содержит в виде меток имена показателей. Именно это именование и нужно использовать в формулах, это можно выполнить перетаскиванием имени показателя в поле «Выражение». Вычисляемые показатели не хранятся в кубе, а вычисляются по мере необходимости.

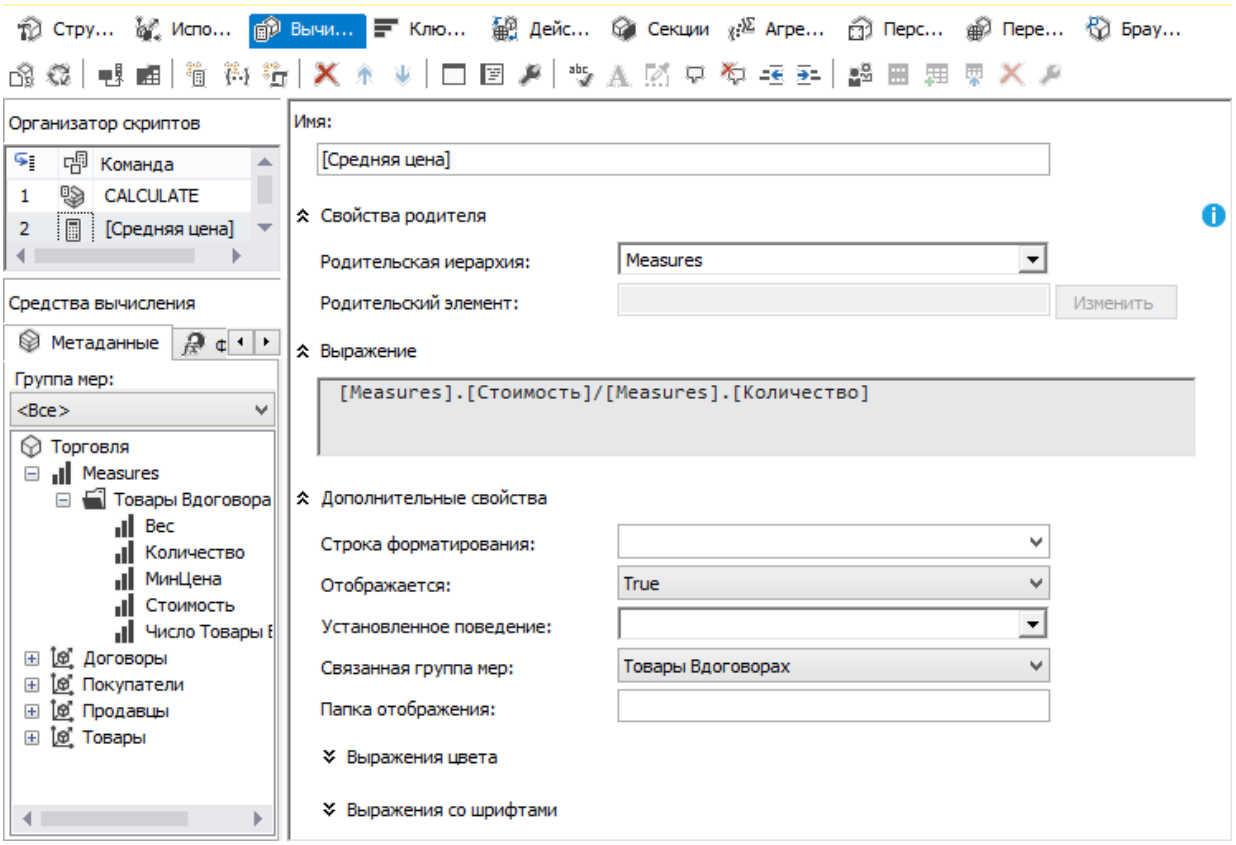

Рис. 20. Создание вычисляемого показателя

Мастер создания кубов создает простую структуру куба, в которой каждое измерение связано непосредственно с таблицей фактов. Такую схему называют «звезда». Измерения «Покупатели» и «Продавцы» связаны с таблицей фактов через измерение «Договоры». Такую схему называют «снежинка» и она не генерируется мастером создания куба. Такие связи нужно определять явно на вкладке «Использование измерений» (рис. 21). Для рассматриваемого примера на этой вкладке будут только измерения «Товары» и «Договоры», связанные с таблицей фактов по схеме «Звезда»

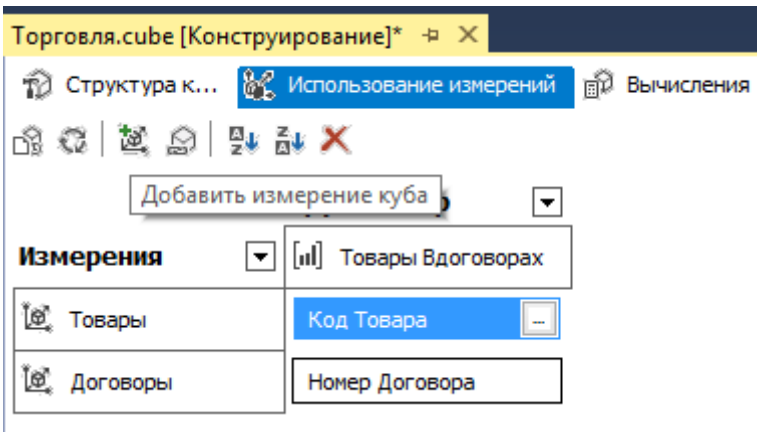

Рис. 21. Страница «Использование измерений»

Для добавления измерения можно воспользоваться одноименной кнопкой или командой контекстного меню. После выбора измерения нужно задать параметры связи (рис. 22): тип связи — «ссылочный» (соответствующий схеме «снежинка»), промежуточное измерение — «Договоры», атрибуты связи — «Код Покупателя» и для выбранного измерения, и для промежуточного.

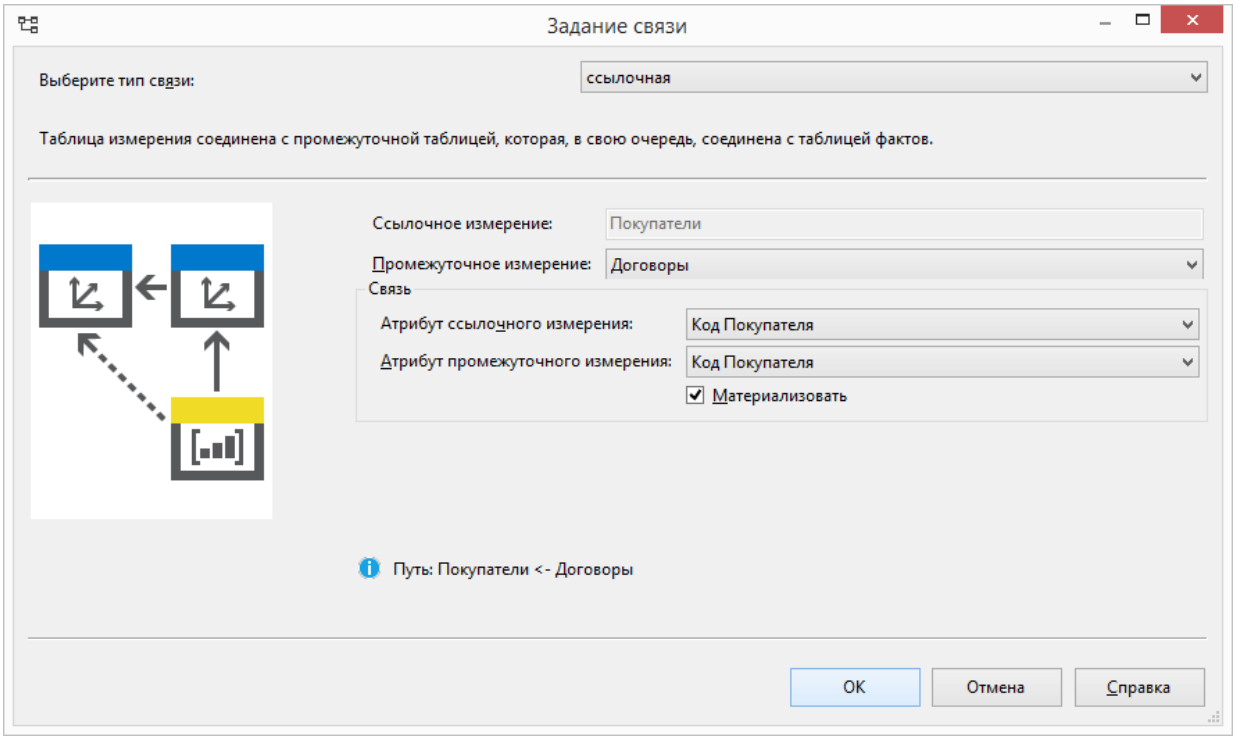

Рис. 22. Определение связи измерения и таблицы фактов

Аналогично производится добавление измерения «Продавцы». В результате на странице «Использование измерений» должны появиться все измерения куба (рис. 23).

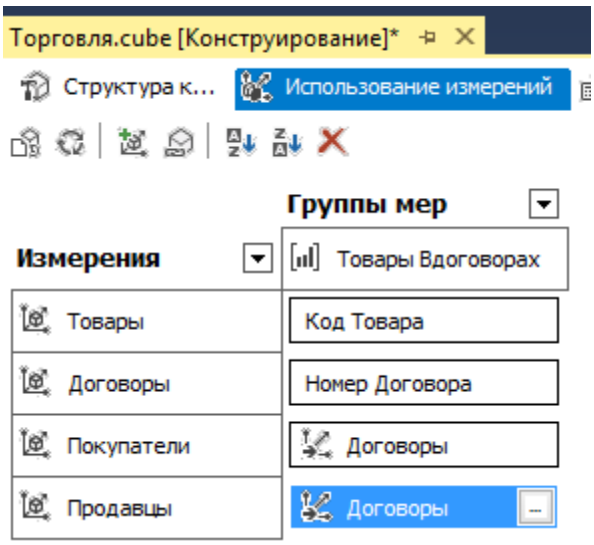

Рис. 23. Связи измерений и таблицы фактов

Чтобы использовать многомерную базу данных OLAP, сначала необходимо обработать ее кубы. Обработка куба— это его заполнение реальными данными из источника данных, организация вычислений и хранения агрегированных значений (агрегатов, обобщений).

Вычисления могут выполняться либо во время обработки куба, либо когда клиентское приложение запросит эти данные. Если создать агрегаты, то вычисления будут выполняться во время обработки куба и итоговые данные будут сохранены в базе данных OLAP.

Когда клиентское приложение запросит данные из куба, сначала будут просмотрены все сохраненные агрегаты, чтобы выяснить, существуют ли необходимые данные. Обнаружив запрошенные данные, сервер просто вернет их клиенту, не делая никаких дополнительных вычислений. Если же запрошенных данных нет ни в одном из сохраненных обобщений, то сервер должен вычислить данные «на лету». Используя агрегаты, можно существенно повысить производительность.

Однако агрегаты не лишены недостатков. Они занимают значительное место на сервере и могут потребовать много времени для обработки. Поэтому, создавая для куба обобщения, нужно найти компромисс между производительностью и памятью.

Для хранения куба используют секции. Секции представляют собой физическое место хранения исходных и агрегатов куба. При создании куба аналитические службы автоматически создают одну секцию. После создания куба можно вернуться к этом процессу и создать новые секции. Секции используются для физического сегментирования данных из куба.

Преимущество секций заключается в том, что у каждой из них может быть свой режим хранения и уникальный набор агрегатов. Таким образом, один логи-

ческий куб можно разделить на несколько источников физических данных, выбрав для каждого из них режим хранения и набор агрегатов. Управление секциями куба выполняется в конструкторе куба на вкладке «Секции» (рис. 24).

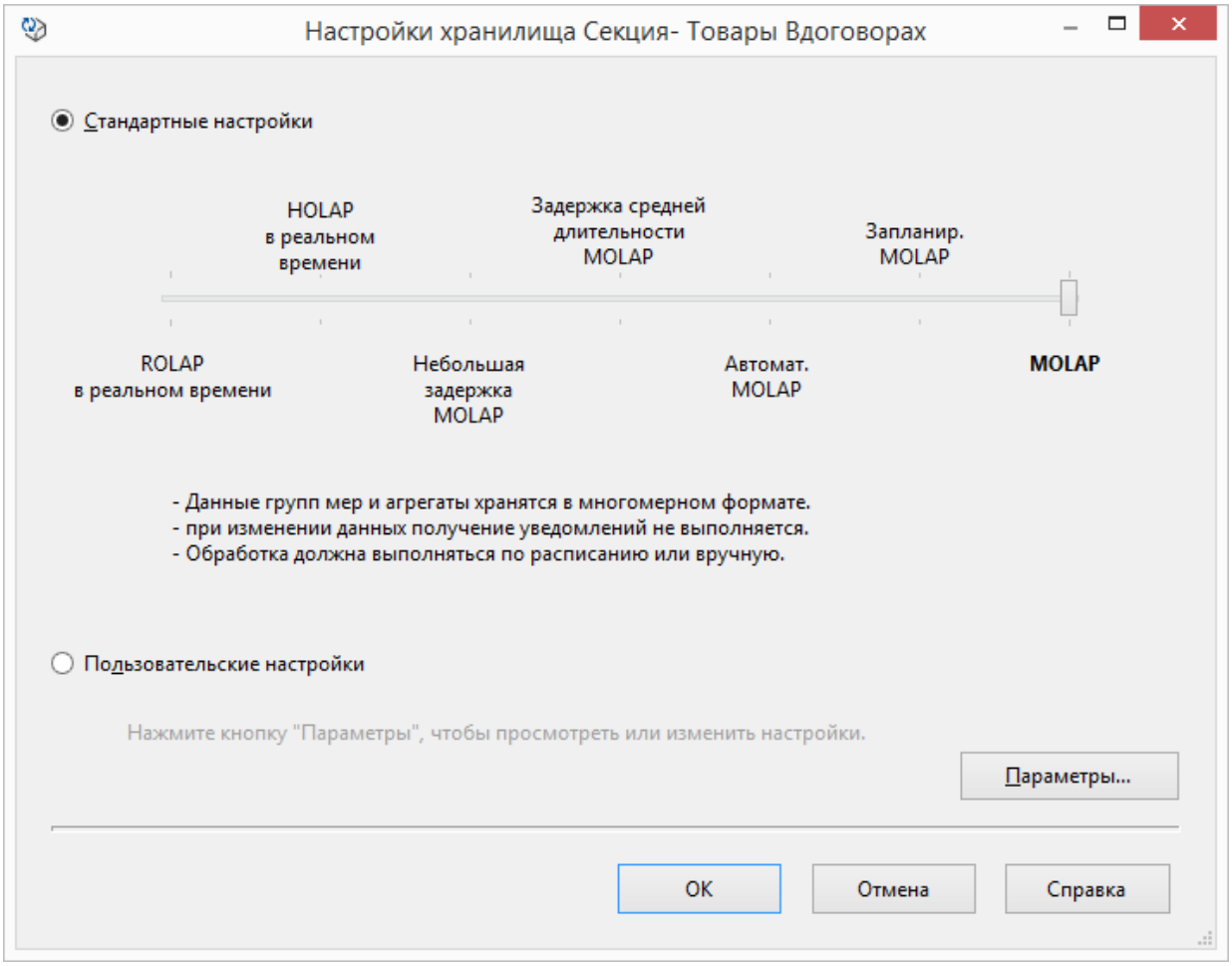

Рис. 24. Параметры хранения данных и агрегатов куба

Предусмотрены следующие режимы хранения:

– ROLAP (Relational OLAP) реального времени. Исходные данные и агрегаты хранятся в реляционном формате. Сервер осуществляет прослушивание уведомлений изменения данных, и все запросы отражают текущее состояние данных (нулевая задержка). Однако, этот метод может давать самое большое время отклика;

– HOLAP (Hibrid OLAP) реального времени. Исходные данные хранятся в реляционном формате, в то время как агрегаты хранятся в многомерном формате. Сервер осуществляет прослушивание уведомлений изменения данных и обновляет агрегаты по необходимости. При обновлении источника данных сервер переключается на реляционный OLAP (ROLAP) реального времени до обновления агрегатов. Все запросы отражают текущее состояние данных (нулевая задержка). Этот метод обычно обеспечивает лучшую общую производительность по сравнению с хранилищем ROLAP;

– MOLAP (Multidimensional OLAP) с малой задержкой. Исходные данные и агрегаты хранятся в многомерном формате. Сервер осуществляет прослушивание уведомлений изменения данных и переключается в режим ROLAP реального времени на время, пока объекты MOLAP повторно обрабатываются. Перед обновлением требуется интервал бездействия не менее 10 секунд. Если интервал бездействия не соблюдается, то активируется 10-минутный интервал прерывания. Обработка осуществляется автоматически при изменениях данных с целевой задержкой, равной 30 минутам после первого изменения. Эта настройка обычно будет использоваться для источника данных с частыми обновлениями, для которого производительность запросов является более важной, чем постоянное предоставление самых последних данных. Эта настройка автоматически обрабатывает объекты MOLAP при необходимости, после интервала задержки. Во время повторной обработки объектов MOLAP снижается производительность;

– MOLAP со средней задержкой. Технология хранения соответствует MOLAP с малой задержкой, за исключением задержки, равной четырем часам;

– автоматический MOLAP. Исходные данные и агрегаты хранятся в многомерном формате. Сервер осуществляет прослушивание уведомлений, но сохраняет самый последний кэш MOLAP при построении нового. Сервер никогда не переключается в режим OLAP реального времени, и запросы могут выдавать устаревшие данные при построении нового кэша. Обработка осуществляется автоматически при изменениях данных с целевой задержкой, равной двум часам. Эта настройка обычно используется для источника данных, для которого производительность запросов является ключевым фактором. Запросы не возвращают самые последние данные во время построения и обработки кэша;

– запланированный MOLAP. Исходные данные и агрегаты хранятся в многомерном формате. Сервер не получает уведомлений об изменении данных. Обработка осуществляется автоматически каждые 24 часа. Эта настройка обычно используется для источника данных, в котором необходимы только ежедневные обновления. Запросы всегда осуществляются в отношении данных в кэше MOLAP, который не очищается до тех пор, пока не будет построен новый кэш и его объекты не будут обработаны;

– MOLAP. Упреждающее кэширование не включено. Исходные данные и агрегаты хранятся в многомерном формате. Сервер не получает уведомлений об изменении данных. Обработка должна либо быть запланирована, либо осуществляться вручную. Эта настройка обычно используется для источника данных, в котором периодические обновления не важны для клиентских приложений, но для которого высокая производительность является критической.

Режим хранения выбирается, исходя из требований к оперативности и производительности многомерного анализа. Высокая производительность достигается за счет частичной потери оперативности.

2.8. Обеспечение безопасности данных OLAP

Аналитические службы предоставляют пользователям доступ к данным, предусматривающий проверку полномочий. Полномочия определяются для ролей (role), членами которых могут быть учетные записи (login) пользователей компьютерной системы. Пользователь получает полномочия тех ролей, в которые он включен.

После установки аналитических служб создается локальная группа OLAP Administrators. По умолчанию учетная запись пользователя, от имени которой устанавливались аналитические службы, становится членом этой группы. Все пользователи этой группы могут выполнять административные функции, в частности создавать новые роли, включать в них пользователей и предоставлять полномочия.

Роли определяются на уровне базы данных и отображаются в обозревателе. Добавить роль и пользователей можно с помощью контекстного меню. На рис. 25 для базы данных «Торговля» определена роль «Отдел сбыта» и на вкладке «Членство» в нее включены пользователи с логинами «user» и «seleznev».

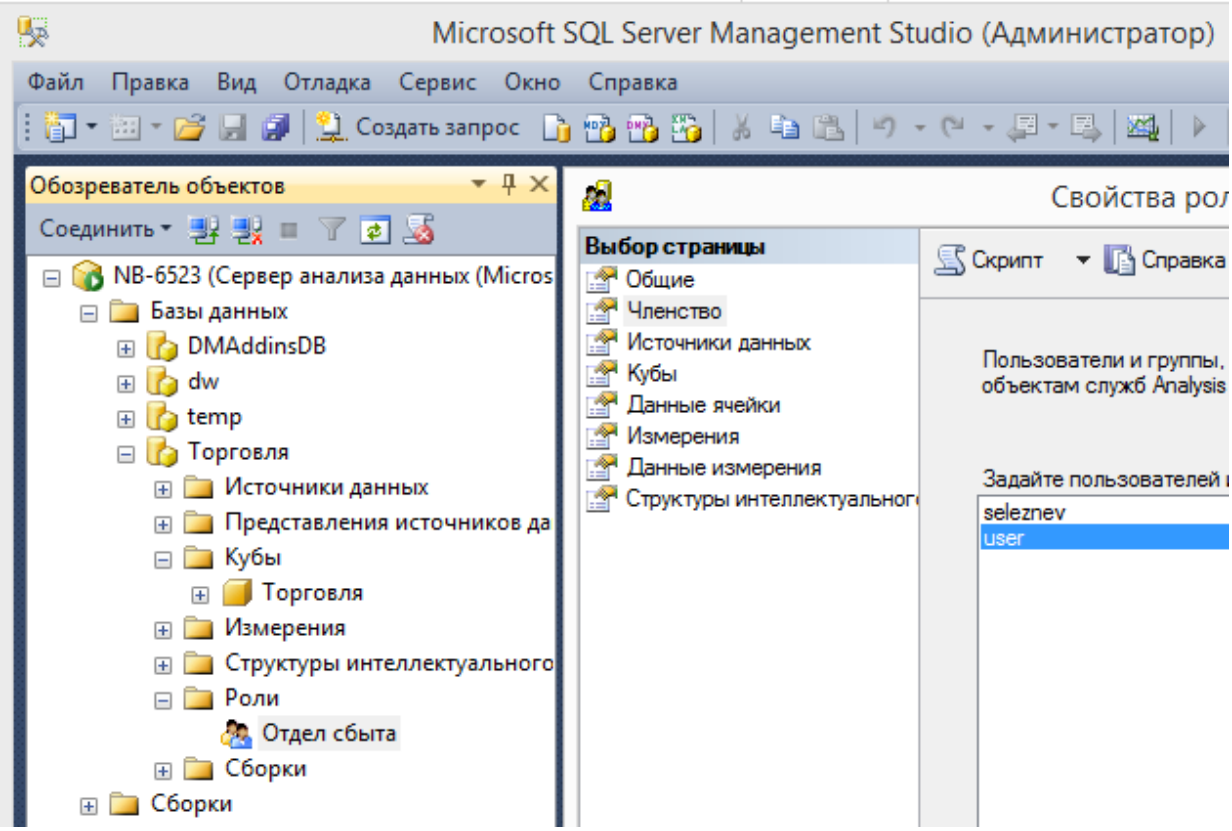

Рис. 25. Роль и ее состав

Основные полномочия определяются для доступа к данным куба. Для каждой роли на вкладке «Кубы» (рис. 26) можно определить для каких кубов пользователи, включенные в роль, могут выполнять чтение данных.

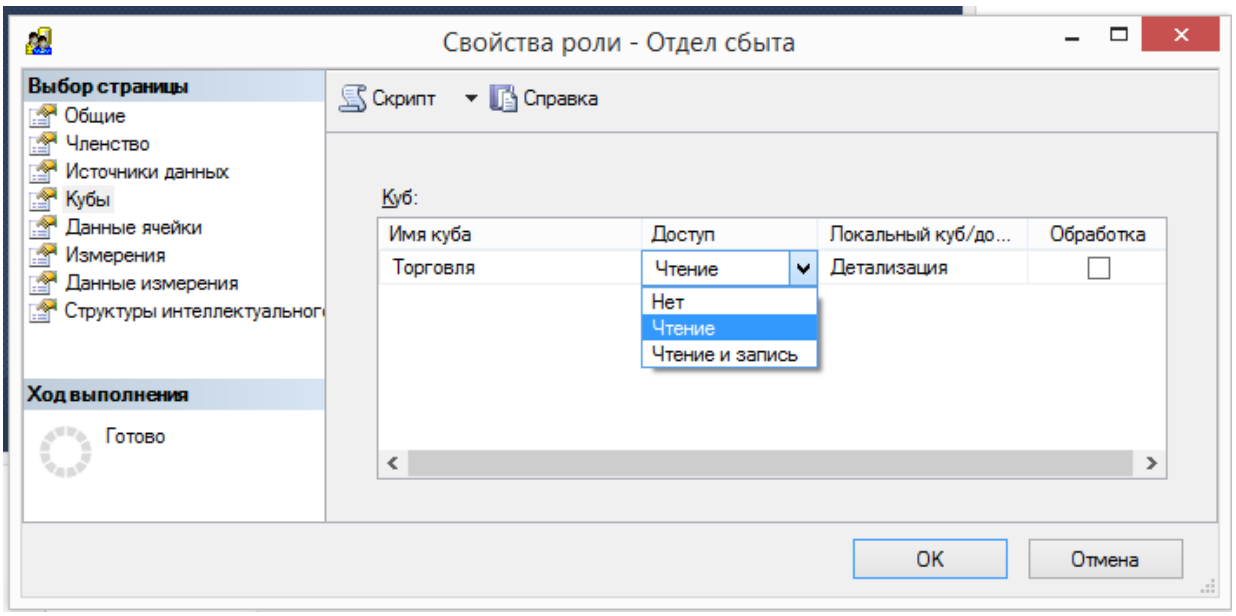

Рис. 26. Определение доступа членов роли «Отдел сбыта» к кубу

Указание куба означает доступ ко всем данным куба. Эти полномочия можно ограничить отдельными измерениями на вкладке «Измерения» (рис. 27).

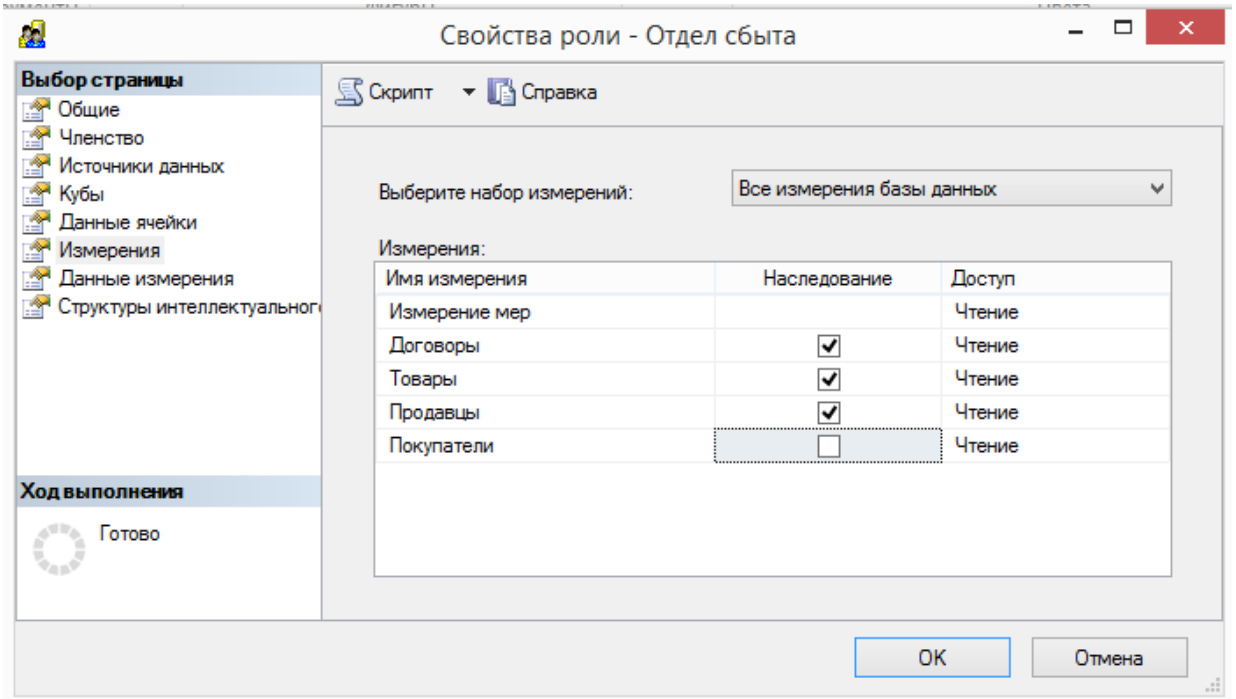

Рис. 27. Указание доступа к отдельным измерениям

Еще одна возможность — ограничение доступа данными, которые соответствуют отдельным меткам некоторого измерения — предоставляется на вкладке «Данные измерения». На рис. 28 продемонстрировано как можно выбрать для доступа определенные показатели (меры). Все показатели включаются в специфическое измерение «Measures» мер, создаваемое автоматически для каждого куба.

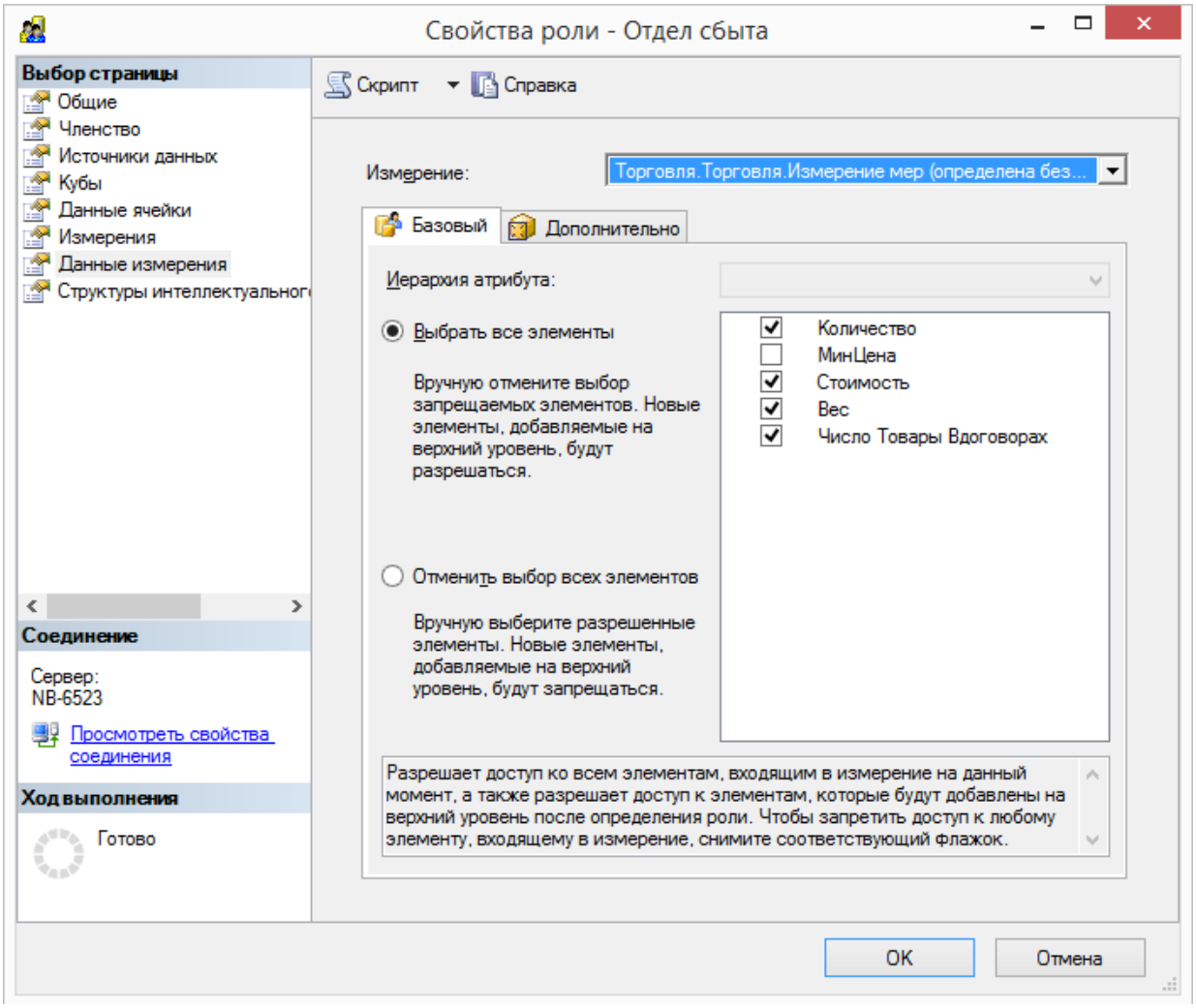

Рис. 28. Ограничение данных метками измерения

2.9. Клиенты OLAP-данных

Систематический сбор и накопление данных позволяют собрать огромный по объему материал для аналитической обработки. Доступ к этим данным может быть обеспечен на уровне специализированных программ, или в среде офисных приложений, или через Интернет. Наиболее трудоемким является разработка специализированных программ доступа к хранилищам данных, и одновременно программы могут обеспечить наибольший уровень сервиса для пользователей. Тем не менее, и другие варианты доступа являются достаточно удобными.

Доступ к OLAP-данным в Microsoft Excel

MS Excel сам по себе обладает OLAP функциональностью в виде сводных таблиц, являющихся аналогом кубов. Поэтому именно эта часть MS Office обеспечивает не только доступ к данным куба, но может применяться для анализа данных с помощью операций среза, агрегирования и детализации. Все эти операции выполняются в привычном интерфейсе для пользователей, умеющих работать со сводными таблицами. Они могут работать с серверными кубами как с

локальными сводными таблицами, даже не подозревая, что все вычисления выполняет не MS Excel, а аналитические службы на сервере.

Для построения сводной таблицы по данным хранилища в Microsoft Excel можно воспользоваться мастером сводных таблиц (команда «Данные», «Сводная таблица»):

1. Для доступа к данным из хранилища следует выбрать вариант «во внешнем источнике данных».

2. Для получения данных следует нажать кнопку «Выбрать подключение». Если нужного подключения нет в списке, то следует выбрать кнопку «Найти другие» и создать новое подключение соответствующим мастером:

2.1. Выбрать тип источника «Microsoft SQL Server Analysis Services».

2.2. Ввести имя сервера и определить способ аутентификации.

2.3. Выбрать базу данных и куб.

2.4. Определить имя подключения и сохранить файл подключения.

3. Далее выполняются обычные действия по определению макета сводной таблицы.

В результате пользователь получает готовый Excel-файл (рис. 29), обеспечивающий работу с кубом, расположенном на сервере и регулярно обновляемом.

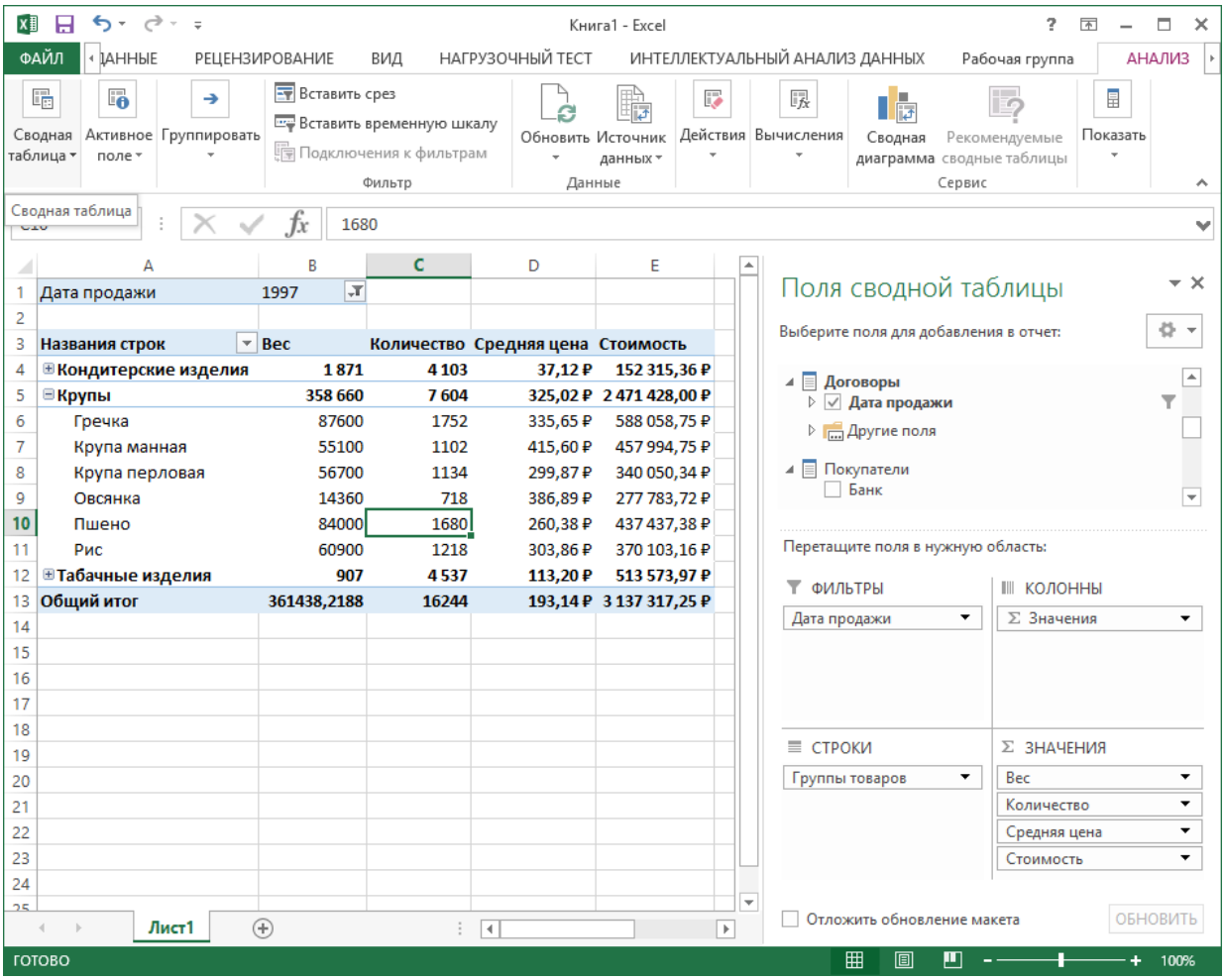

Рис. 29. Доступ к кубу на сервере с помощью MS Excel

Публикация сводных таблиц на Web-страницах

Самый простой способ построить Web-страницу с OLAP функциональностью заключается в использовании компонента PivotTable List. Для этого нужно сохранить сводную таблицу Microsoft Excel с доступом к кубу (см. выше) как Web-страницу: выберем в Microsoft Excel пункт меню «Файл», «Сохранить как веб-страницу», в появившейся диалоговой панели выберем переключатель «Добавить интерактивность», нажмем кнопку «Опубликовать», в диалоговой панели выберем из выпадающего списка «Элементы «Лист 1» и добавим «Работу со сводными таблицами».

Далее следует изменить заголовок, который появится на будущей Web-странице, и сохранить ее. Если открыть эту страницу в Microsoft Internet Explorer версии 4.01 или выше, мы увидим, что она содержит PivotTable List — элемент управления, предназначенный для просмотра OLAP-данных и сводных таблиц на Web-страницах.

Сразу же заметим, что этот элемент управления можно применять только в локальных сетях на компьютерах, для которых приобретена лицензия на Microsoft Office; другие способы его применения, например на Web-страницах, доступных в Интернете, запрещены лицензионным соглашением.

Пользователь, манипулирующий PivotTable List в браузере или в Windows-приложении, может, как и в сводной таблице Excel, перемещать данные в область строк, столбцов и страниц (в Microsoft Office Web Components приняты термины Row Area, Column Area и Filter Area) с диалоговой панели, напоминающей панель «Список полей сводной таблицы» из Excel.

Пользователь может также выполнять операцию детализации (drill-down), щелкая мышью на значках «+». Компонент PivotTable List позволяет сортировать и фильтровать данные. Во-первых, фильтрация данных может быть осуществлена с помощью отображения только выбранных членов измерений, которые могут быть отмечены в выпадающем списке, сходном с соответствующим списком Excel. Во-вторых, с помощью диалоговой панели «Команды и параметры» можно выбрать способы фильтрации и группировки данных.

Помимо этого, пользователь может изменять атрибуты отображения данных — цвет и шрифт текста, цвет фона, выравнивание текста, отображение и т.д. Для этого достаточно поместить курсор на один из элементов данных, атрибуты которого нужно изменить (например, на наименование члена измерения, на ячейку с суммарными данными или с итоговыми значениями), и выбрать новые атрибуты отображения данных этого типа в той же диалоговой панели «Команды и параметры».

Помимо этого, компонент PivotTable List позволяет на основе агрегированных данных вычислять доли или проценты общей суммы или суммы, соответствующей родительскому члену измерения (например, процент от годовой прибыли, полученный в данном квартале), — соответствующие опции можно найти в контекстных меню элементов данных.

Пользователю также доступен специально предназначенный для него файл справки (на русском языке, если используются Web-компоненты из комплекта поставки русской версии Microsoft Office XP). Однако пользователь не может изменить источник данных и отобразить на Web-странице другой OLAP-куб, поскольку право сделать это есть только у разработчика Web-страницы.

Отметим, что подобную Web-страницу можно создать и с помощью Microsoft FrontPage.

2.10. Язык запросов к многомерным данным MDX

Язык запросов к многомерным данным MDX (MultiDimensional eXpressions) был впервые введен в рамках спецификации OLE DB for OLAP для работы с многомерными кубами. Будучи открытым стандартом, MDX является основным инструментом программирования для аналитических служб Microsoft SQL Server.

Для выполнения запросов на языке MDX можно использовать окно MDX-запросов в MS SQL Server Management Studio. Для создания окна можно указать многомерную базу данных и воспользоваться контекстным меню. На рис. 30 для многомерной БД «Торговля» с помощью контекстного меню создается запрос для ввода и обработки многомерного выражения на MDX.

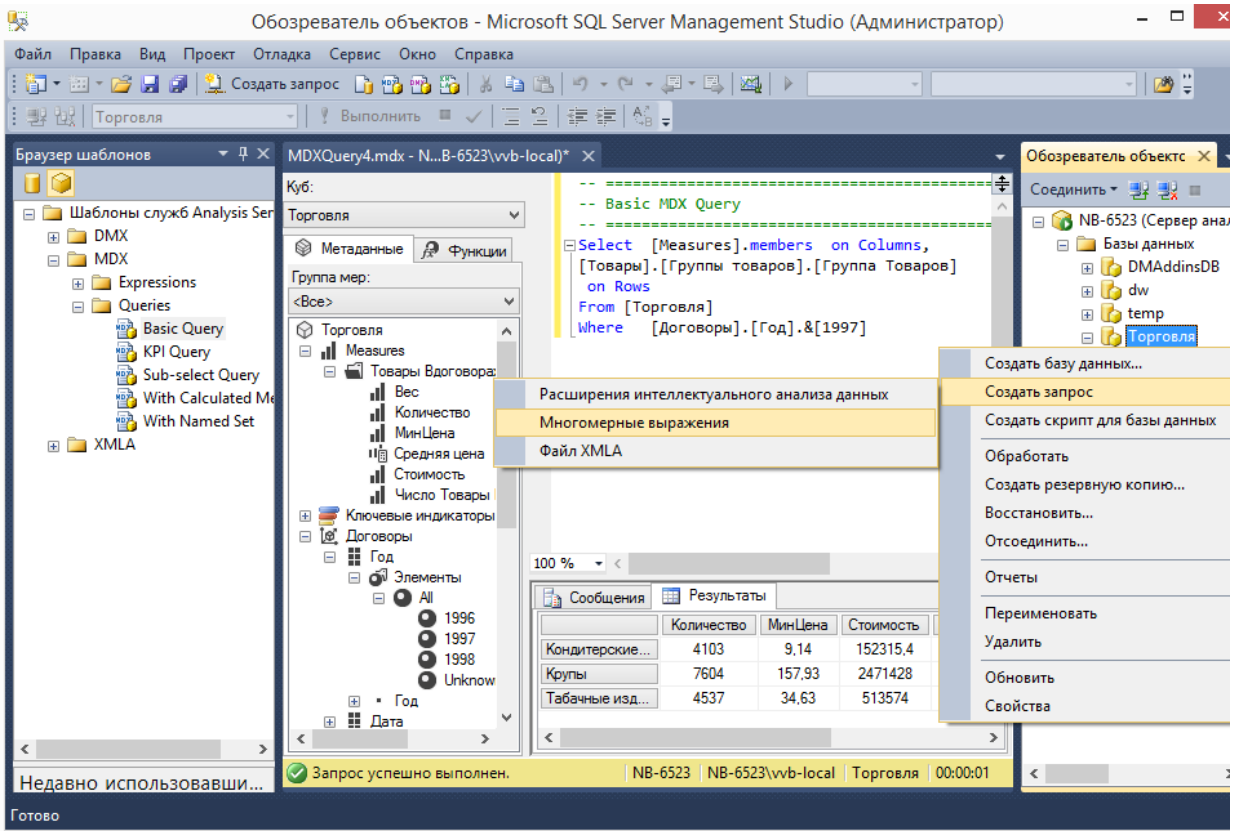

Рис. 30. Окно MDX-запросов в MS SQL Server Management Studio

Можно вводить MDX-команды непосредственно в панели запросов (3-я панель справа на рис. 30) или конструировать запрос, перетаскивая измерения и меры куба в панель запросов из области (2-я панель) метаданные куба. Помимо этого, можно использовать примеры функций из панели «Браузер шаблонов». Выполнить запрос можно нажав кнопку «Выполнить» на панели инструментов. Результат выполнения запроса отображается в нижней части экрана

MDX использует следующие понятия:

– ***компонент*** или ***метка*** (Member) — значение измерения на любом уровне иерархии. Определяется перечислением всех узлов, расположенных на пути от выбранной метки к вершине. Примеры:

[Товары].[КлассификацияТоваров]

[Товары].[КлассификацияТоваров].[Группа Товаров].&[Крупы]

[Товары].[КлассификацияТоваров].[Группа Товаров].&[Крупы].&[Рис]]

Для исключения двусмысленности имена заключают в квадратные скобки, если они содержат специальные знаки, такие как пробел. Символ амперсанда (&) перед именем метки указывает на ссылку по ключу. На каждую метку измерения можно ссылаться как по имени, так и по ключу, хотя рекомендуется все же последний вариант;

– ***кортеж*** (Tuple) — коллекция ячеек куба (срез), соответствующих некоторой комбинации значений измерений. Определяется как набор меток в круглых скобках. Размерность кортежа — множество измерений меток, вошедших в кортеж. Например, кортеж

([Measures].[Стоимость],

[Товары].[КлассификацияТоваров].[Группа Товаров].&[Крупы].&[Рис], [Договоры].[Год].&[1997])

соответствует измерениям «Measures», «Товары», «Договоры».

– ***множество (Set)*** — набор кортежей. Определяется как перечисление значений (кортежей) в фигурных скобках. Размерности всех кортежей множества должны совпадать. Например, {[Продавцы].[Продавец].&[Вест], [Продавцы].[Продавец].&[Фокус]}.

Запрос на языке MDX представляет собой команду, которая выглядит следующим образом:

SELECT [<ось> [, <ось>...]]

FROM [<куб>]

[WHERE [<срез>]],

где *<ось>* — описание измерения куба-результата в виде

<множество> ON <имя оси>,

<имя оси> может быть следующим:

– *COLUMNS* — колонки,

– *ROWS* — строки,

– *PAGES* — страницы,

– *SECTIONS* — разделы,

– *CHAPTERS* — главы,

– *AXIS(<номер>)* — ось с указанным номером (оси нумеруются с нуля).

Срез куба задается кортежем. Одно измерение не может употребляться одновременно для задания двух осей или оси и среза. Для указания множества меток иерархии (или ее уровня) применяется метод members. Кортеж заключается в круглые скобки, в то время как множество — в фигурные. Порядок перечисления измерений и мер в кортеже не имеет значения.

Результатом запроса всегда является куб — значения одного или нескольких показателей, соответствующих значениям измерений (см. рис. 31). Измерения располагаются по одной или нескольким осям, определенным в запросе. Например, выборка с рис. 31 создается следующим запросом

```
Select[Договоры].[Дата продажи].[Год] on Columns,
[Продавцы].[Продавец].[Продавец] on rows
From [Торговля]
Where [Measures].[Стоимость]
```

Сообщения | Результаты

	1996	1997	1998	Unknown
Вест	490371,3	273834,5	145867,5	(null)
Геракл	616712,4	358835,3	142352,9	(null)
Гермес	537810,8	267097,7	(null)	(null)
Единство	149437,1	197379	(null)	(null)
Норд	95125,13	483208	43964,01	(null)
Плюс	164509,2	105589,4	(null)	(null)
Ромашка	240157,6	267136,8	9080,44	(null)
СибАтом	106461,7	376934,3	12278,67	(null)
Тор	294353,5	256856,4	73893,12	(null)
Успех	263373,8	369430,3	5736,27	(null)
Федоров	(null)	(null)	(null)	(null)
Фокус	226937	181015,8	47303,55	(null)
Unknown	(null)	(null)	(null)	(null)

Рис. 31. Результат запроса MDX

Набором меток каждой оси можно управлять. Прежде всего, можно образовывать метки перечислением. Например,

```
Select {[Договоры].[Дата продажи].[Год].&[1996],
[Договоры].[Дата продажи].[Год].&[1998]} on Columns,
{[Продавцы].[Продавец].&[Норд],
[Продавцы].[Продавец].&[Успех]} on rows
From [Торговля]
Where [Measures].[Стоимость]
```

В результате в каждом измерении будет две метки (рис. 32)

	1996	1998
Норд	95125,13	43964,01
Успех	263373,8	5736,27

Рис. 32. Результат запроса с перечислением меток

Кроме этого, множество может быть задано интервалом *<1-я метка> : <последняя метка>*, например

```
Select {[Договоры].[Дата продажи].[Год].&[1996]:
[Договоры].[Дата продажи].[Год].&[1998]} on Columns,
{[Продавцы].[Продавец].&[Норд]:
[Продавцы].[Продавец].&[Успех]} on rows
From [Торговля]
Where [Measures].[Стоимость]
```

	1996	1997	1998
Норд	95125,13	483208	43964,01
Плюс	164509,2	105589,4	(null)
Ромашка	240157,6	267136,8	9080,44
СибАтом	106461,7	376934,3	12278,67
Тор	294353,5	256856,4	73893,12
Успех	263373,8	369430,3	5736,27

Рис. 33. Результат запроса с заданием множества меток интервалом

Для определения осей можно использовать функцию *Crossjoin(<Множество>, <Множество>)*, которая позволяет комбинировать несколько измерений в одно измерение куба-результата. Запрос

```
Select {[Measures].[Количество], [Measures].[Стоимость], [Measures].[Средняя
цена]} on Columns,
CrossJoin({[Договоры].[Дата продажи].[Год].&[1996],
[Договоры].[Дата продажи].[Год].&[1998]},
{[Продавцы].[Продавец].&[СибАтом]:
[Продавцы].[Продавец].&[Успех]}) on rows
From [Торговля]
```

формирует двухуровневую структуру (год продажи, продавцы) заголовков столбцов.

		Количество	Стоимость	Средняя цена
1996	СибАтом	571	106461,7	186,447843695271
1996	Тор	1567	294353,5	187,845265634971
1996	Успех	1469	263373,8	179,287780803268
1998	СибАтом	99	12278,67	124,026968907828
1998	Тор	300	73893,12	246,310390625
1998	Успех	48	5736,27	119,505625406901

Рис. 34. Результат запроса с функцией *CrossJoin*

Filter((<Множество>, <Условия>) позволяет выполнять фильтрацию меток измерения (оси) — выбираются метки, удовлетворяющие условию.

Запрос

```
Select {[Measures].[Количество], [Measures].[Стоимость], [Measures].[Средняя
цена]} on Columns,
Filter([Продавцы].[Продавец].members, [Measures].[Стоимость]>500000) on rows
```

From [Торговля]

выберет продавцов, у которых стоимость продаж больше 500000.

Order(<Множество>, <Выражение> [, ASC | DESC | BASC | BDESC]) сортирует метки измерения с сохранением иерархии *ASC | DESC* или без нее *BASC | BDESC*.

Ниже приведен запрос с сортировкой товаров по убыванию стоимости

Select {[Measures].[Количество],[Measures].[Стоимость]} on Columns,
ORDER([Товары].[КлассификацияТоваров].members, [Measures].[Стоимость],
basc) on rows
From [Торговля]

В результате (рис. 35) получится список всех меток измерения «Товары», отсортированных независимо от уровня иерархии метки. Применение ключа «*ASC*» приведет к сортировке сначала групп, а внутри каждой группы товаров.

	Количество	Стоимость
Конфеты "Весна"	2927	54437,46
Конфеты "Ананасные"	1840	71624,13
Конфеты "Асорти"	2489	77432,13
Конфеты "Визит"	2548	151652,8
Сигареты Петр 1	2360	174827,8
Сигареты LM	2485	300945,2
Сигареты Fine Line	2453	322485,7
Кондитерские изделия	9804	355146,5
Сигареты MarlBoro	2622	398807,4
Крупа перловая	2623	707153,3
Крупа манная	2040	848554,9
Овсянка	2337	887262,8
Пшено	3295	905662,6
Рис	2736	931824,3
Гречка	2801	970373,1
Табачные изделия	9920	1197066
Крупы	15832	5250831
All	35556	6803042

Рис. 35. Результат запроса с сортировкой меток

TopCount (<Множество>, n, <Выражение>) выделяет из множества первые *n* компонент с наибольшими значениями выражения, например запрос

Select {[Measures].[Количество],[Measures].[Стоимость]} on Columns,
TopCount([Продавцы].[Продавец].[Продавец].members, 5,
[Measures].[Стоимость]) on rows
From [Торговля]

выберет 5 первых продавцов с наибольшей стоимостью продаж.

Аналогичными функциями являются TopPercent, BottomCount, BottomPercent. Выражение

TopPercent([Продавцы].[Продавец].[Продавец].members, 80, [Measures].[Стоимость])

выбирает продавцов, суммарная стоимость продаж которых не меньше 80 %.

Для указания иерархии измерения используются следующие методы и функции:

.children, .FirstChild, . LastChild — «потомки» метки иерархии;

.Parent — «родительская» метка

.Siblings , .FirstSibling, .LastSibling — «соседи» по уровню.

Перечисленные методы указываются для определенной метки:

[Договоры].[Дата продажи].[Год].&[1996].&[10].parent — родитель для октября 1996 г. — 1996 г.,

[Договоры].[Дата продажи].[Год].&[1996].&[10].FirstChild — первая дата продаж в октябре 1996,

[Договоры].[Дата продажи].[Год].&[1996].&[10].LastChild — последняя дата продаж в октябре 1996,

[Договоры].[Дата продажи].[Год].&[1996].&[10].FirstSibling — первый месяц 1996 г.

[Договоры].[Дата продажи].[Год].&[1996].&[10].LastSibling — последний (двенадцатый) месяц 1996 г.

Есть специальные функции для формирования множества меток, расположенных выше или ниже выбранного узла по уровню иерархии:

Ascendants(<метка>) — функция возвращает родительскую метку;

Descendants(<метка> [, «Level»[, «Desc_flags»]]) — функция, возвращающая метки, расположенные ниже по уровню иерархии метки — аргумента, например, запрос

```
Select    {[Measures].[Количество],[Measures].[Стоимость]} on Columns,
   {
   Ascendants([Договоры].[Дата продажи].[Год].&[1996].&[10]),
   Descendants([Договоры].[Дата продажи].[Год].&[1996].&[10])
   }  on rows
From     [Торговля]
```

возвращает (рис. 36) для октября 1996 метки: 10, 1996, All — расположенные выше по иерархии и метки: 3, 11, 19, 27 — расположенные ниже (даты продаж в октябре 1996)

	Количество	Стоимость
10	952	138791,1
1996	16920	3185250
All	35556	6803042
10	952	138791,1
3	203	31773,73
11	346	38490,69
19	193	38861,48
27	210	29665,23

Рис. 36. Результат запроса со списком меток выше и ниже по иерархии

Определение *NON EMPTY <ось>* исключает строки, у которых все клетки пустые (например, не было продаж).

Во всех выражениях языка MDX значения из куба указываются при помощи кортежей. В примере

```
Select {[Measures].[Количество],[Measures].[Стоимость]} on Columns,
    Filter([Товары].[КлассификацияТоваров].[Товар].members,
    ([Measures].[Количество], [Договоры].[Год].&[1996]) >
    ([Measures].[Количество], [Договоры].[Год].&[1997]))  on rows
From [Торговля]
WHERE [Договоры].[Год].&[1997]
```

кортеж используется внутри функции фильтрации, чтобы для каждой метки измерения «Товар» вычислить показатель «Количество» за определенный год.

В запросах можно определять вычисления при помощи следующей конструкции

WITH <формула> [, <формула>] <Запрос>

Наиболее часто применяется формулы вычисления уровней измерений

MEMBER <имя уровня> AS '<выражение>'
[,<свойство>=<значение>...]

< имя уровня > — полностью квалифицированное имя с указанием измерения и уровня иерархии, к которому будет отнесено вычисляемое значение.

< выражение > — выражение, вычисляющее значение,

В качестве свойств ячейки можно указывать шрифты и другие особенности форматирования, например, FORMAT_STRING = '# ##0' означает вывод чисел с отбрасыванием дробной части и разделением групп разрядов.

В следующем запросе для каждого периода времени вычисляется суммарная стоимость продаж в сопоставлении с предыдущим периодом (рис. 37).

```
WITH MEMBER [Measures].[Стоимость товаров за прошлый период]
  AS '([Measures].[Стоимость], [Договоры].[Дата продажи].PrevMember)'
  MEMBER [Measures].[Увеличение стоимости товаров]
  AS '[Measures].[Стоимость]– [Measures].[Стоимость товаров за прошлый период]'
select {[Договоры].[Дата продажи].[Год] .Members} ON ROWS,
{[Measures].[Стоимость],
[Measures].[Стоимость товаров за прошлый период], [Measures].[Увеличение стоимости товаров] } ON COLUMNS
from [Торговля]
```

	Стоимость	Стоимость товаров за прошлый период	Увеличение стоимости товаров
1996	3185250	(null)	3185250
1997	3137318	3185250	-47932
1998	480476,5	3137318	-2656841

Рис. 37. Результат запроса с вычислением новых показателей

Можно вычислять не только новые уровни измерений, но и множества

WITH SET <set_name> AS '<set>'

Например, запрос:

```
WITH
  SET [Отсортированные товары] AS
    'ORDER( {[Товары].[КлассификацияТоваров].members} ,
                [Measures].[Стоимость], desc)'
  MEMBER [Measures].[All] AS
    '  Sum([Товары].[КлассификацияТоваров].[Товар].Members ,
          [Measures].[Стоимость]) '
  MEMBER [Measures].[Процент] AS
    '  [Measures].[Стоимость] / [Measures].[All] ',
      FORMAT_STRING = '# ##0.0 %'
select { [Measures].[Стоимость],
 [Measures].[Количество],[Measures].[Средняя цена] ,[Measures].[All],
 [Measures].[Процент]}  on columns,
[Отсортированные товары]  on rows
 from [Торговля]
```

возвращает данные, представленные на рис. 38.

	Стоимость	Количество	Средняя цена	All	Процент
All	6803042	35556	191,333164585443	6803043	100,0 %
Крупы	5250831	15832	331,659360788277	6803043,60546875	77,2 %
Гречка	970373,1	2801	346,438080149946	6803043,60546875	14,3 %
Рис	931824,3	2736	340,57903874269	6803043,60546875	13,7 %
Пшено	905662,6	3295	274,859673748103	6803043,60546875	13,3 %
Овсянка	887262,8	2337	379,658884253316	6803043,60546875	13,0 %
Крупа манная	848554,9	2040	415,958302696078	6803043,60546875	12,5 %
Крупа перловая	707153,3	2623	269,59712161647	6803043,60546875	10,4 %
Табачные изделия	1197066	9920	120,672001008065	6803043,60546875	17,6 %
Сигареты MarlBoro	398807,4	2622	152,100471967963	6803043,60546875	5,9 %
Сигареты Fine Line	322485,7	2453	131,465819914391	6803043,60546875	4,7 %
Сигареты LM	300945,2	2485	121,104715794769	6803043,60546875	4,4 %
Сигареты Петр 1	174827,8	2360	74,0795881885593	6803043,60546875	2,6 %
Кондитерские изделия	355146,5	9804	36,2246563902489	6803043,60546875	5,2 %
Конфеты "Визит"	151652,8	2548	59,5183722527473	6803043,60546875	2,2 %
Конфеты "Асорти"	77432,13	2489	31,1097328244275	6803043,60546875	1,1 %
Конфеты "Ананасные"	71624,13	1840	38,9261591372283	6803043,60546875	1,1 %
Конфеты "Весна"	54437,46	2927	18,5983795802016	6803043,60546875	0,8 %

Рис. 38. Результат запроса с вычислением множества

MDX-запросы можно использовать для извлечения данных из кубов аналогично SQL-запросам. Такое использование обладает большими аналитическими возможностями и высоким быстродействием.

Вопросы

1. Назначение аналитических баз данных. Отличие хранилищ аналитических баз от реляционных баз данных.
2. Причины построения и использования аналитических баз данных.
3. Требования, предъявляемые к аналитическим базам данных.
4. Опишите структуру данных аналитической базы.
5. Дайте определения понятий «куб», «измерение», «показатель», «ячейка куба», «метка измерения».
6. Перечислите и опишите основные операции в кубе.
7. Опишите технологию создания куба с помощью SQL Server Data Tools.
8. Что содержит таблица фактов?
9. Для чего используются обобщения (агрегаты)?
10. Перечислите и опишите режимы хранения данных в кубе.
11. Как обновляется содержание куба?
12. Опишите систему безопасности MS Analysis Services.
13. Какие средства можно использовать для доступа к данным налитической базы?
14. Для чего предназначен язык MDX? Каковы его основные возможности?
15. Как выглядит запрос на языке MDX? Каковы его основные компоненты?
16. Описание кортежей, множеств, меток в языке MDX.
17. Опишите применение функций сортировки, фильтрации, комбинирования измерений в языке MDX.
18. Опишите применение функций работы с иерархическими структурами в языке MDX.
19. Укажите возможности программирования вычислений в языке MDX.

3. ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ ДАННЫХ

Ключевым для понимания интеллектуального анализа данных является вероятностный характер зависимостей. С одной стороны, неопределенность может являться следствием неполноты наших знаний о наблюдаемых объектах и событиях, с другой стороны, случайность может быть неотъемлемой характеристикой наблюдаемых явлений, как в случае с подбрасыванием монетки. Так или иначе, выделенные закономерности являются «усредненными», т.е. они работают в основной доле случаев, но при этом возможны различные отклонения. Если расчеты свидетельствуют о кредитоспособности заемщика, то вероятность возврата кредита высока, но не стопроцентна. Существуют маловероятные варианты развития событий, при которых заемщик не сможет рассчитаться за кредит.

Именно вероятностная природа наблюдений, приводит к возможности применения разных моделей для решения одной и той же задачи. Поэтому очень важным является определение качества моделирования: соотношения доли явлений, объясненных моделью, с долей явлений, которые модель не объясняет и относит на долю случайных воздействий. И опять природная случайность не позволяет достичь точной оценки влияния случайности. Зачастую это и не требуется. Если применение модели увеличивает вероятность продажи, то это и приведет к увеличению объема продаж.

Предложены методы решения следующих задач выделения зависимостей и закономерностей:

– *Задача классификации*. Данные описывают объекты одинаковым набором атрибутов. Номенклатура классов определена заранее. Есть данные о классе каждого объекта. Необходимо установить правило определения класса объекта по набору атрибутов. Например, для оценки платежеспособности потенциального заемщика задано два класса: «платежеспособен» и «неплатежеспособен», которые определены для множества заемщиков по кредитной истории. Правило вычисления класса поможет снизить риски и увеличить доходность кредитования.

– *Задача кластеризации* — заключается в делении множества объектов на группы (кластеры) схожих по параметрам. Число кластеров и их характеристики могут быть заранее неизвестны и определяются в ходе построения кластеров, исходя из степени близости объединяемых объектов по совокупности параметров. Другое название этой задачи — сегментация. Например, выделение групп клиентов поможет сформировать более эффективную технологию продаж, учитывающую особенности каждой группы клиентов.

– *Задача регрессии*. В описании объектов выделяют исходные факторы — независимые переменные и результат их влияния — отклик или зависимую переменную. По исходным данным необходимо построить функцию, для вычисления усредненного отклика по значениям факторов. Кроме этого, поскольку зависимость имеет случайный характер, необходимо оценивать отклонения реальных наблюдений от значений функции регрессии.

– *Задача прогнозирования*. Определение следующих новых значений для продолжения имеющихся значений числовой последовательности (или нескольких последовательностей, между значениями в которых наблюдается зависимость). При этом могут учитываться имеющиеся тенденции (тренды), сезонность, другие факторы. Классическим примером является прогнозирование цен акций на бирже.

– *Задача определения взаимосвязей*, также называемая задачей поиска ассоциативных правил, заключается в определении часто встречающихся наборов объектов среди множества подобных наборов. Классическим примером является анализ потребительской корзины, который позволяет определить наборы товаров, чаще всего встречающиеся в одном заказе (в одном чеке).

– *Анализ последовательностей* или секвенциальный анализ одними авторами рассматривается как вариант предыдущей задачи, другими — выделяется отдельно. Целью, в данном случае, является обнаружение закономерностей в последовательностях событий. Подобная информация позволяет, например, предупредить сбой в работе информационной системы, получив сигнал о наступлении события, часто предшествующего сбою подобного типа. Другой пример применения — анализ последовательности переходов пользователей по страницам web-сайтов.

– *Анализ отклонений* позволяет отыскать среди множества событий те, которые существенно отличаются от нормы. Отклонение может сигнализировать о каком-то необычном событии (неожиданный результат эксперимента, мошенническая операция по банковской карте) или об ошибке ввода данных оператором.

Приведенный выше список является неполным. Например, активно развивается Text Mining — исследование зависимостей в некотором наборе документов или сообщений. Для практической реализации приведенные аналитические задачи следует сопоставить с целями и задачами бизнеса, чтобы выявить возможность и целесообразность применения соответствующих аналитических технологий. Ниже приведены некоторые примеры использования задач интеллектуального анализа данных.

Data Mining в розничной торговле:

– *анализ покупательской корзины* (анализ сходства) предназначен для выявления товаров, которые покупатели стремятся приобретать вместе. Знание покупательской корзины необходимо для улучшения рекламы, выработки стратегии создания запасов товаров и способов их раскладки в торговых залах,

– *исследование временных шаблонов (анализ последовательностей*) помогает торговым предприятиям принимать решения о создании товарных запасов,

– *создание прогнозирующих моделей* дает возможность разработки точно направленных экономичных мероприятий по продвижению товаров.

Data Mining в телекоммуникациях:

– *анализ последовательностей вызовов* — выявление стереотипов использования услуг и разработка привлекательных наборов услуг,

– *выявление лояльности клиентов (кластеризация и классификация)* — определение характеристик клиентов, которые, один раз воспользовавшись услугами данной компании, с большой долей вероятности останутся ей верными.

Data Mining в страховании

– *выявление мошенничества (кластеризация и классификация)* — выявление стереотипов мошенничества,

– *анализ риска (регрессия)* — точное знание рисков позволяет снижать цены и потери

Data Mining в банках

– *выявление мошенничества с кредитными карточками (кластеризация и классификация)* знание стереотипов такого мошенничества позволяет спланировать более эффективные меры безопасности

– *сегментация клиентов (кластеризация и классификация)* — более целенаправленная маркетинговая политика — различные виды услуг предлагаются разным группам клиентов.

– *прогнозирование изменений клиентуры* — прогнозные модели ценности своих клиентов, и соответствующее обслуживание каждой категории.

Приведенные примеры демонстрируют широкие возможности применения аналитических технологий.

3.1. Модели аналитической обработки

Практически в каждой аналитической модели происходит построение алгоритма $y = A(x)$ по имеющимся данным $(x_1, y_1), \ldots, (x_n, y_n)$. Аргументом x_i в общем случае является вектор $x_i = (x_{i1}, \ldots, x_{im})$ значений атрибутов. Например, для задачи классификации алгоритм вычисляет наиболее вероятный класс объекта по вектору атрибутов объекта. Для оценки качества работы алгоритма вводят функцию потерь $G(A,x)$, которая характеризует величину ошибки алгоритма. Например, квадрат отклонения $G(A,x_i) = (y_i - A(x_i))^2$. При $G(A,x) = 0$ ошибка отсутствует. В общем случае, ошибка является некоторой случайной величиной. Единичное наблюдение характеризует ошибку алгоритма в одном случае и не определяет среднее значение ошибки по всем данным. Для оценки адекватности алгоритма применяют функцию качества $Q(A,x_1, \ldots, x_n) = (G(A,x_1)+\ldots+ G(A,x_n))/n$, которая вычисляет среднее значение функции потерь по выборке.

При построении алгоритма добиваются минимального значения средней ошибки — функции качества. Как правило, для этого применяют не все имеющиеся данные, их некоторую часть — так называемую обучающую выборку. Оставшиеся данные применяют для контроля качества построения модели. Когда качество работы алгоритма на новых объектах, не вошедших в состав обучения, оказывается существенно хуже, чем на обучающей выборке, говорят об эффекте переобучения (overtraining) или переподгонки (overfitting). При решении

практических задач с этим явлением приходится сталкиваться очень часто. Причиной переобучения может быть попытка построить функцию для случайных колебаний обучающей выборки.

Как и многие другие производители, корпорация Microsoft предложила свои решения в области исследования зависимостей. Модели Data Mining реализованы в аналитических службах MS SQL сервера (Analysis services) в виде следующих алгоритмов [7]:

- упрощенный алгоритм Байеса — MicrosoftNaiveBayes;
- алгоритм дерева принятия решений — MicrosoftDecisionTrees;
- алгоритм временных рядов — MicrosoftTimeSeries;
- алгоритм кластеризации — MicrosoftClustering;
- алгоритм кластеризации последовательностей — MicrosoftSequenceClustering;
- алгоритм взаимосвязей– MicrosoftAssociationRules;
- алгоритм нейронной сети — MicrosoftNeuralNetwork;
- алгоритм линейной регрессии — MicrosoftLinearRegression;
- алгоритм логистической регрессии — MicrosoftLogisticRegression.

Перечисленные алгоритмы можно применять для решения различных задач исследования зависимостей.

Для настройки алгоритмов можно применять разные инструменты. Стандартный способ заключается в создании интеллектуального решения в среде SQL Server Data Tools (см. 2.4) — последовательно определяются следующие компоненты решения:

1) проект, в свойствах «Deployment» которого указывается в какой аналитической БД какого сервера будет происходить развертывание сервера;
2) источники данных (Data Sources), из которых будет извлекаться информация для настойки моделей;
3) представления данных (Data Source Views) для выбора данных из источников;
4) структуры (Рис. 39) для выявления зависимостей (Mining Structures);
5) модели (Mining Models), параметры которых настраиваются по данным из структур.

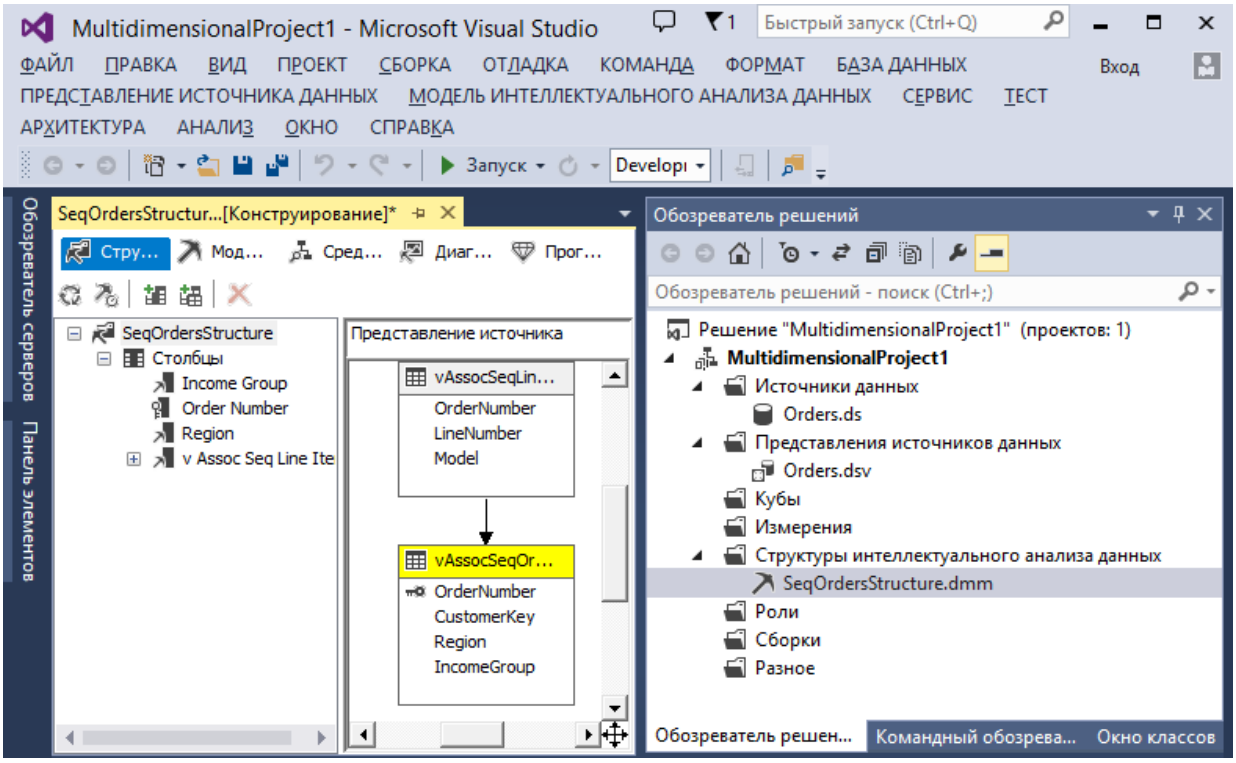

Рис. 39. Создание структуры в проекте в SQL Server Data Tools

Еще один способ работы с моделями Data Mining предоставляют надстройки MS Excel интеллектуального анализа данных. Этот инструмент позволяет обратиться к аналитическим службам Microsoft SQL Server прямо из Excel: определить параметры модели, передать данные, получить результаты обработки. Скачать надстройки (SQLServer2008_DMAddin.msi) можно с сайта Microsoft по ссылке http://www.microsoft.com/ru-ru/download/confirmation.aspx?id=20350. После установки надстроек в меню кнопки «Пуск» операционной системы добавляются команды настройки подключения к аналитическим службам сервера (Рис. 40). Основным в этой настройке является указание сервера, к которому будет происходить обращение из Excel, и определение аналитической базы данных, в которой будут размещаться модели. В меню Excel появляются две вкладки «Интеллектуальный анализ данных» (Рис. 41) и «Анализировать» (Рис. 42).

На вкладке «Интеллектуальный анализ данных» есть группа команд «Соединение», с помощью которых можно просмотреть или изменить параметры соединения с аналитическими службами сервера. На Рис. 43 показаны параметры соединения (аналитическая БД «DMAddinsDB» на сервере «nb-6523») и окно для определения соединения. После установки соединения Надстройки MS Excel готовы к применению моделей Data Mining для обработки данных электронных таблиц.

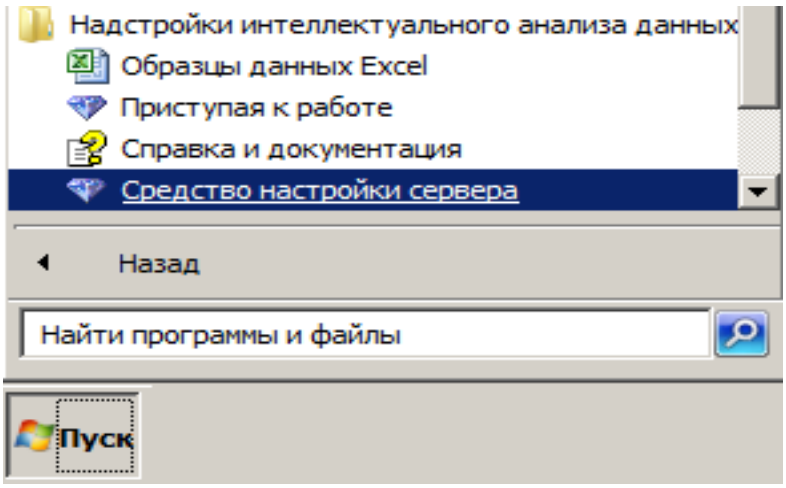

Рис. 40. Меню «Надстройки интеллектуального анализа данных»

Рис. 41. Меню «Интеллектуальный анализ данных»

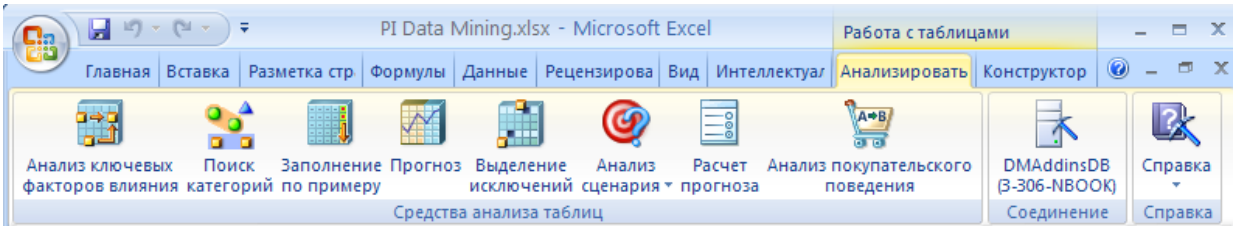

Рис. 42. Меню «Анализировать»

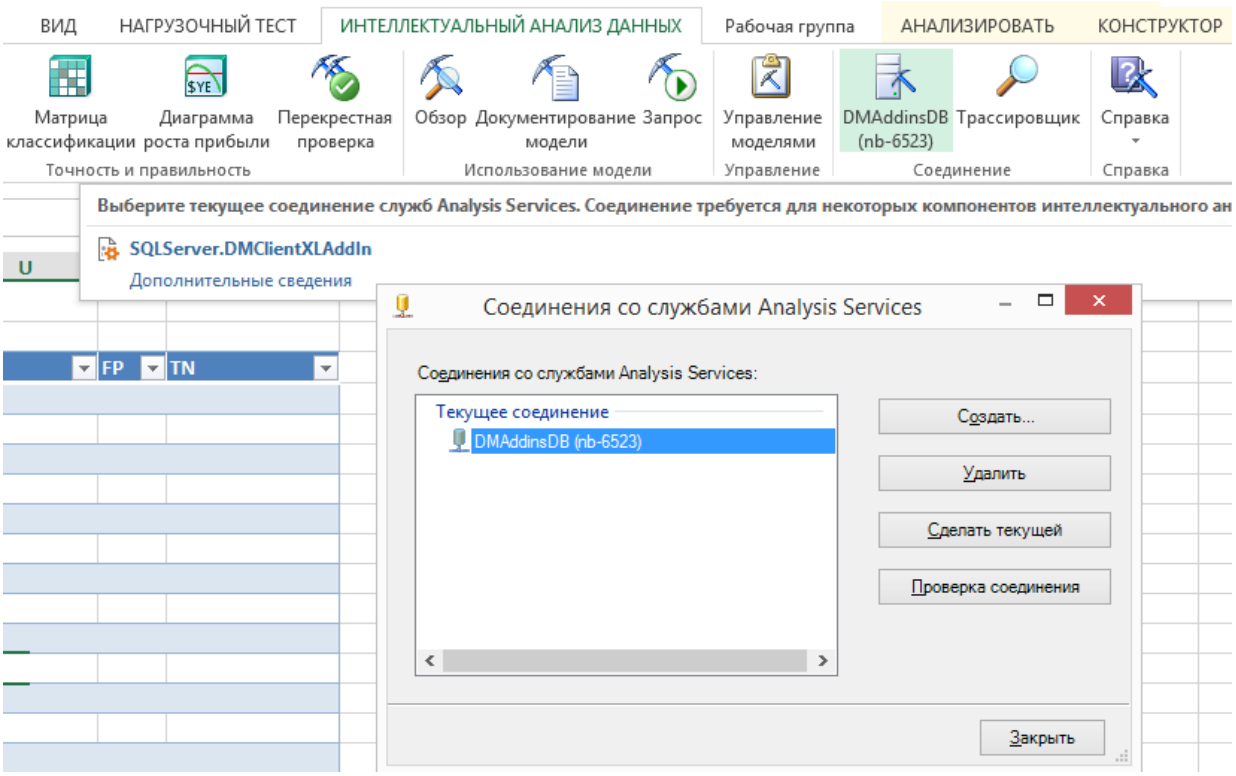

Рис. 43. Параметры соединения с сервером

3.2. Предварительная обработка данных

Предварительная обработка данных имеет особое значение для исследования зависимостей. Ошибки и неточности данных могут серьезно исказить если не характер, то параметры зависимостей. Предварительную обработку традиционно обозначают ETL по первым буквам трех операций: Extraction — извлечение, Transformation — преобразование, Loading — загрузка — последовательно выполняемых при выполнении предварительной обработки.

Извлечение данных

Извлечение — это построение таблицы наблюдений, которая содержит исходные (входные) значения переменных $x_1, \dots, x_m$ и значения одной или нескольких выходных переменных $y_1, \dots, y_k$, для каждой из которых нужно построить алгоритм $y_i = A_i(x_1, \dots, x_m)$ по наблюдениям. Одна строка содержит одно наблюдение входных и выходных переменных.

Таблица наблюдений обычно формируется по данным базы данных, которая содержит детальные данные по одному или нескольким бизнес-процессам. В базах данных информация по одному событию разнесена по нескольким таблицам. Рассмотрим пример с продажей лекарств. В таблице «Продажи» (см. В свою очередь таблица «Товары» содержит ссылку «Код товарной группы» на классификационные признаки товара в таблице «Группы товаров» (Таблица 4).

Таблица 1 отсутствуют данные о товаре и отделе, которые есть в таблицах «Товары» (см. Таблица 2) и «Отделы» (см. Таблица 3), на которые в таблице «Продажи» есть ссылки: «Код товара» и «Код отдела».

В свою очередь таблица «Товары» содержит ссылку «Код товарной группы» на классификационные признаки товара в таблице «Группы товаров» (Таблица 4).

Таблица 1

Таблица базы данных «Продажи» (фрагмент)

Дата продажи	Код товара	Код отдела	Час покупки	Количество	Сумма
01.12.2008	477	1	11	1	4,45
01.12.2008	485	1	11	1	4,56

Таблица 2

Таблица базы данных «Товары» (фрагмент)

Код товара	Наименование товара	Код товарной группы
35	Адреналина гидрохлорида раствор 0,1% р-р д/ин. 0,1 % амп. 1 мл [с нож.амп.] уп.контурн.яч. 5 Московс	159
354	Андрокур табл. 50 мг фл. 20 кор. 1 Schering-Plough Labo N.V.	178
477	Аспаркам табл. уп.контурн.б/яч. 10 Медисорб	108
485	Аспаркам табл. уп.контурн.б/яч. 10 Усолье-Сибирский ХФК	108

Таблица 3

Таблица базы данных «Отделы» (фрагмент)

Код отдела	Наименование отдела
1	Аптека № 1
2	Аптека № 3
3	Аптека № 2

Таблица 4

Таблица базы данных «Группы товаров» (фрагмент)

Код товарной группы	Товарная группа
108	Макро- и микроэлементы
159	Адрено- и симпатомиметики (альфа-, бета-)
177	Ингибиторы фибринолиза
178	Андрогены, антиандрогены

Обычный способ извлечения данных — это построение SQL-запроса, соединяющего в одну таблицу входные и выходные переменные:

SELECT [Дата продажи], [Час покупки], [Наименование товара], [Товарная группа], [Наименование отдела], [Количество], [Сумма]
FROM [Продажи] JOIN [Товары] ON [Продажи]. [Код товара] = [Товары]. [Код товара]
JOIN [Отделы] ON [Продажи]. [Код отдела] = [Отделы]. [Код отдела]
JOIN [Группы товаров] ON [Группы товаров]. [Код товарной группы] = [Товары]. [Код товарной группы]

В результате выполнения запроса получается таблица наблюдений, в которой первые пять колонок будут входными переменными, а последние две — выходными. Такой запрос становится частью аналитической системы. Его можно использовать тогда, когда возникает необходимость настроить модель с учетом новых данных.

Современные средства (например, Microsoft PowerBi) позволяют объединить в одном запросе таблицы из разных источников: можно было бы по с сайта центробанка по дате получить курс доллара для определения суммы в долларах США.

Преобразование данных

Преобразование данных определяется целями и сложностью исследований. Можно выделить несколько типовых преобразований:

Преобразование структуры данных — наиболее распространенное преобразование формирует одну таблицу из множества источников (таблиц). Представленный выше SQL-запрос является характерным примером такого преобразования.

Агрегирование — получение сводных значений по имеющимся детальным. Целью агрегирования может быть уменьшение объема исходных данных и/или усреднение случайных колебаний показателей. Например, целесообразно получить сводные данные о продажах лекарств по часам с помощью запроса:

```
SELECT [Дата продажи], [Час покупки], [Наименование товара], [Товарная группа], [Наименование отдела], SUM([Количество]) AS [Количество], SUM([Сумма]) AS [Сумма]
FROM [Продажи] JOIN [Товары] ON [Продажи]. [Код товара] = [Товары]. [Код товара]
JOIN [Отделы] ON [Продажи]. [Код отдела] = [Отделы]. [Код отдела]
JOIN [Группы товаров] ON [Группы товаров]. [Код товарной группы] = [Товары]. [Код товарной группы]
GROUP BY [Дата продажи], [Час покупки], [Наименование товара], [Товарная группа], [Наименование отдела]
```

Запрос аналогичен предыдущему и отличается группировкой по входным переменным с вычислением по каждой группе суммарных стоимости и количества.

Перевод значений выполняется если в разных источниках одно и тоже имеет разные обозначения. Самый простой вариант — это использование таблиц перевода содержащих пары «старое значение» — «новое значение». Иногда перевод значений можно выполнить по формуле, например, если разные единицы измерения приводятся к одной системе.

Создание новых данных возникает достаточно часто, когда нужно вычислять новые классификационные признаки (например, определить по дате день недели) или значения показателей (среднюю цену можно вычислять как суммарная стоимость деленая на суммарное количество).

Нормализация — приведение различных величин к однотипным интервалам. Один из приемов группировки объектов — это вычисление расстояния

между ними используя для этого разницы одинаковых атрибутов. При этом, разницы могут принадлежать разным интервалам. Например, для описания клиентов разница полов может быть 0 — одинаковый пол или 1 — разный пол, разница возрастов может быть целым числом от 0 до 100, разница доходов может исчисляться десятками тысяч. Ясно, что большие разницы будут давать б*о*льший вклад в расстояние между объектами. Это может существенно искажать картину, так как атрибуты имеют разные единицы измерения. Широко применяются различные приемы выравнивания области значений разных атрибутов. Во-первых, можно значения каждого атрибута пересчитывать в один интервал (например, [0,1]) с помощью линейных или нелинейных преобразований. Во-вторых, предполагая, что каждый атрибут является случайной величиной, пересчитывая его в случайную величину с нулевым математическим ожиданием и единичной дисперсией.

Обогащение данных — добавление к данным новой информацией для повышения ценности. Выделяют внешнее обогащение — когда данные берутся из внешних источников (например, применение динамики курсов валют, для получения динамики валютной стоимости). Внутреннее обогащение — это добавление данных из внутренних источников (например, к описанию каждого сотрудника добавляется новый атрибут — персональный рейтинг сотрудника).

Очистка данных — это одна из наиболее важных функций предобработки, потому что она связана с устранением ошибок и неточностей в данных, что напрямую влияет на точность выявления зависимостей. К ошибкам и неточностям в данных можно отнести следующие ситуации:

– орфографические ошибки;

– пропуски — отсутствие значений некоторых атрибутов;

– фиктивные значения, которые вынужденно подставляются в некоторых системах регистрации;

– логические несоответствия, например, дата окончание меньше даты начала;

– дубликаты записей, когда один и тот же объект описан неоднократно, с разными кодами;

– противоречия — разные описания одного объекта, например, две фамилии одного человека, возникшие при смене фамилии.

– нарушения целостности — появление ссылок на отсутствующие объекты;

– различные шкалы, когда в одно поле вводят величины разных шкал измерения, например, расстояние то в метрах, то в сантиметрах;

– закодированные значения, когда вместо значения атрибута вводят связанный с ним код;

– составные значения, когда в одно поле вводят дополнительную информацию, например, к фамилии добавили ученое звание.

Список таких ситуаций можно пополнять и пополнять. В каждой системе пользователь может совершать ошибки регистрации, которые сложно обнаружить.

Есть некоторые приемы, которые позволяют выделить если не ошибки, то сомнительные данные. Это конечно визуальное наблюдение и оценивание. Практически каждый внимательный пользователь может обнаружить орфографические ошибки. Полезным является применение различных словарей и классификаторов. Простая сортировка данных позволит сделать визуальный контроль более эффективным. Проверка первых и последних по порядку данных позволяет найти нетипичные компоненты такие как числа среди слов. Полезным оказывается изучение простых статистических характеристик: минимумов, максимумов, средних, дисперсий, частот отдельных значений, гистограмм. Наконец, визуальную проверку делает на порядок более эффективным построение графиков. Выделение типичных ошибок вручную позволяет построить алгоритмы обнаружения ошибок.

Отдельный случай составляет выделение выбросов: появление маловероятных значений или их комбинаций. По наблюдениям можно оценить среднее и дисперсию наблюдаемых значений и, задавая доверительную вероятность, вычислить границы интервалов, выход за которые свидетельствует о маловероятных событиях. Обычно такие экстремальные значения очень сильно влияют на изучаемые зависимости. Достаточно одного сильного выброса, чтобы исказить усредненные зависимости.

В каждом случае неполных, неточных или ошибочных данных нужно принимать решения в отношении их использования для исследования зависимостей. Возможны следующие варианты очистки:

1) Исключение таких данных из выборки. Так как исключается вся строка, то наблюдений становится меньше, и достоверность выводов снижается.

2) Замена данных на некоторые значения: средние, наиболее вероятные, минимальные или максимальные.

Выбор варианта зависит от анализируемой ситуации и возможных искажений исследуемых зависимостей.

3.3. Задача классификации

Задача классификации может быть сформулирована следующим образом. Задано множество объектов $X = \{x_1, \ldots, x_n\}$ и множество классов (названий) Y. Каждый объект $\boldsymbol{x_i}$ характеризуется набором атрибутов (x_{i1}, $x_{i2}, \ldots x_{im}$). Требуется построить алгоритм классификации $X \to Y$ по данным наблюдений $(x_1(x_{11},\ldots,x_{1m}), y_1),\ldots,(x_n(x_{n1},\ldots,x_{nm}), y_n)$.

Например, в банке для каждого заемщика устанавливают набор атрибутов таких, как доход, семейное положение, владение недвижимостью и т.д. В результате кредитных взаимоотношений по так называемой кредитной истории каждый заемщик получает класс «платежеспособный» или «неплатежеспособный». Класс определяется закономерностями и случайными обстоятельствами (даже самый платежеспособный заемщик может не вернуть кредит под воздействием комбинации неблагоприятных факторов). Таким образом, класс может быть определен с определенной долей вероятности. Причем, эта вероятность зависит

от атрибутов заемщика. В задаче классификации необходимо не только определить принадлежность объекта классу по набору атрибутов, но и оценить вероятность такой классификации.

Предположим, что по данным наблюдений, доля «неплатежеспособных» заемщиков составляет 10 %. Значение некоторого атрибута или комбинация значений атрибутов может существенно повлиять на эту долю, увеличивает или уменьшает вероятность отнесения заемщика к тому или иному классу. Знание таких закономерностей позволяет более успешно решать многие задачи. Если такого нет влияния атрибутов на платежеспособность, то решение задачи классификации не будет лучше выбора наугад.

Для решения задачи классификации применяют следующие модели и алгоритмы:

- упрощенный алгоритм Байеса (Naive Bayes),
- дерево решений (Decision Tree),
- логистическая регрессия (Logistic Regression),
- нейронные сети (Neural Network).

3.3.1. Упрощенный алгоритм Байеса

Алгоритм получил название по формуле Байеса для вычисления условных вероятностей. Рассмотрим следующий пример. Пусть вероятности классов совпадают с частотами и равны $P(H_0) = 82\ \%$ для платежеспособного клиента (событие H_0 : выбранный наугад клиент — платежеспособен) и $P(H_1) = 18\ \%$ для неплатежеспособного клиента (событие H_1: выбранный наугад клиент — неплатежеспособен). Рассмотрим влияние образования на класс клиента. По выборке доля клиентов с высшим образованием составляет $P(A)=65\ \%$. Доля платежеспособных клиентов с высшим образованием составляет $P(A\ H_0) = 59\ \%$. Если клиент имеет высшее образование, то вероятность «платежеспособности» по формуле Байеса составит

$$P(H_0|A) = P(A\ H_0) / P(A) = 0{,}59 / 0{,}65=0{,}908.$$

Это существенно увеличивает шансы правильной классификации. Таким образом, можно выделить значения атрибутов (или комбинации значений), которые влияют на принадлежность классу.

По наблюдениям находятся вероятности (частоты) классов $P(H_k)$, вероятности значений атрибутов $P(x_i)$ и вероятности значений атрибутов $P(x_i|H_k)$ при условии принадлежности классу H_k. Для объекта A с атрибутами $a_1, \ldots, a_n$ находят условные вероятности $P(H_k|A)$ и класс объекта определяют по наибольшей вероятности. Алгоритм чувствителен к редким значениям. Если вероятность какого-либо значения незначительна, то соответствующая частота может оказаться нулевой и это приведет к нулевому значению условной вероятности, хотя если исключить редкие значения атрибутов, то картина может существенно измениться.

Упрощение заключается в предположении независимости распределения признаков

$$P(x_1, \ldots, x_n) = P(x_1) \times \ldots \times P(x_n).$$

На практике данное предположение выполняется достаточно редко. Тем не менее метод обладает высоким быстродействием и дает приемлемые по точности результаты в условиях общей неопределенности. Кроме этого, достоинством данного алгоритма является низкая вычислительная сложность.

Для иллюстрации использования алгоритмов интеллектуального анализа будем использовать данные, предоставляемые Microsoft вместе с надстройками. На странице «Средства анализа таблиц» (Рис. 44) приведены атрибуты покупателей с указанием приобретения велосипеда. Значения этой колонки и содержат указание класса клиента. Все остальные атрибуты рассматриваются как факторы, влияющие на приобретение велосипеда.

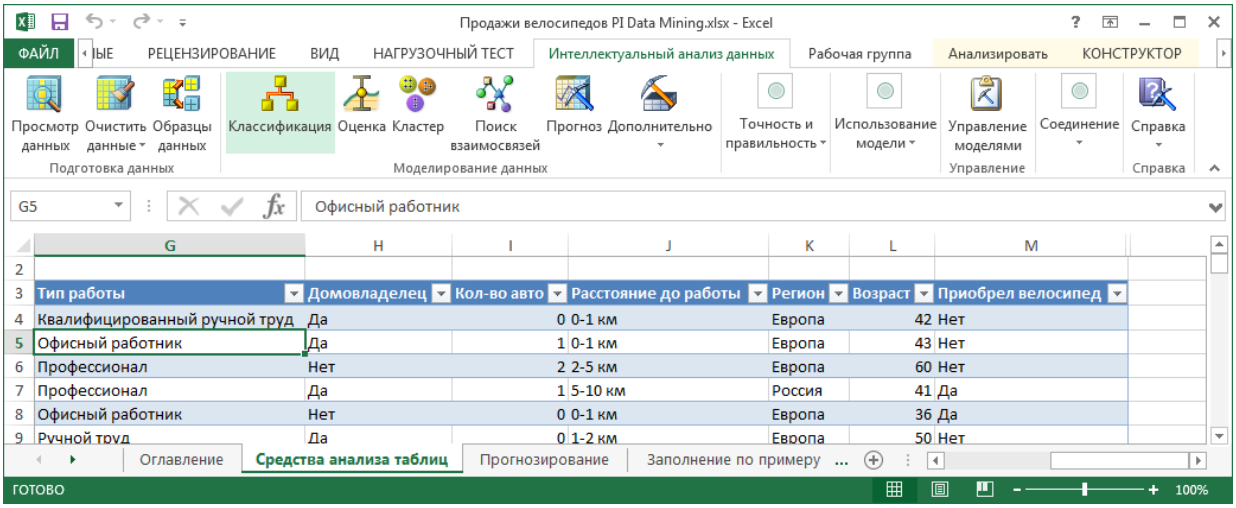

Рис. 44. Данные о покупателях велосипедов

Запускается процесс классификации одноименной кнопкой в меню «Интеллектуальный анализ данных». В диалоговых окнах мастера классификации нужно определить (Рис. 45) анализируемый столбец, который содержит обозначения классов (в данном примере — это «Приобрел Велосипед») и входные столбцы (все остальные за исключением «Приобрел Велосипед» и «Id» — столбца с номером клиента в системе).

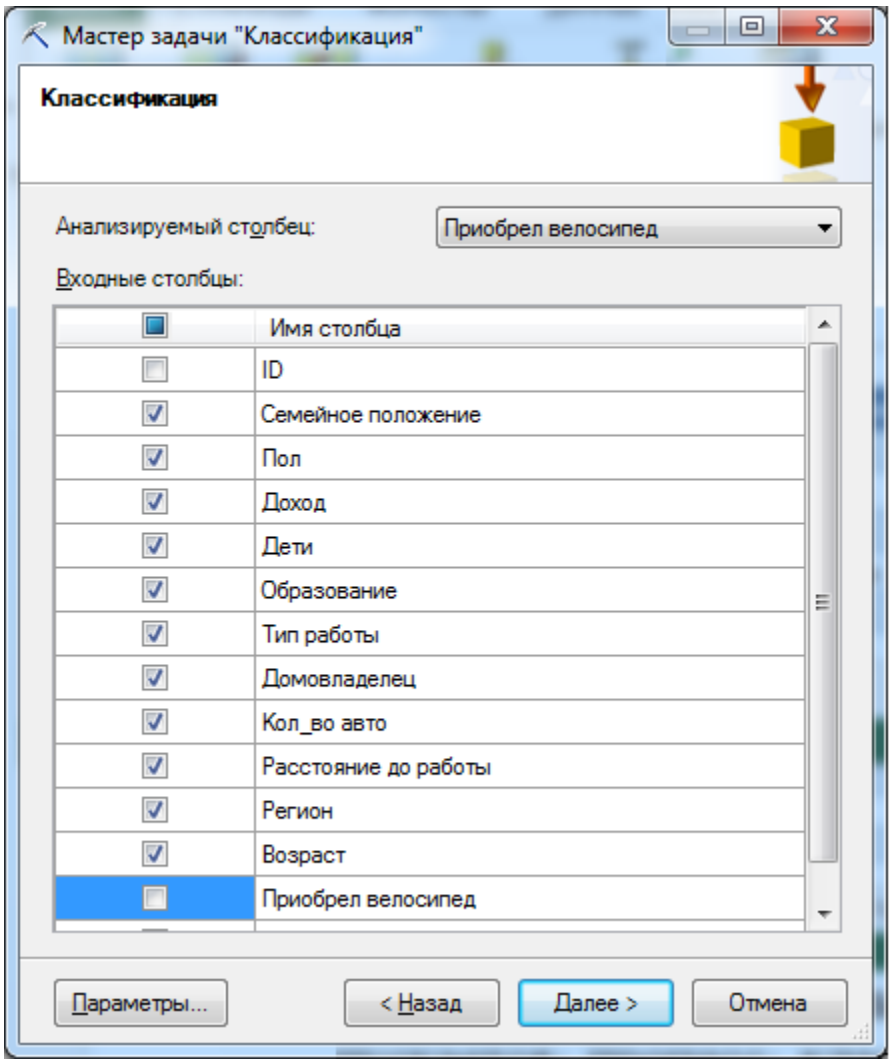

Рис. 45. Определение целевой колонки с указанием класса и входных колонок

Кнопка «Параметры» открывает окно (Рис. 46), в котором можно выбрать алгоритм классификации и параметры алгоритма. Для алгоритма Байеса нужно задать минимум вероятности. Приведенный пример содержит тысячу наблюдений, поэтому 0,001 соответствует минимальному количеству наблюдений.

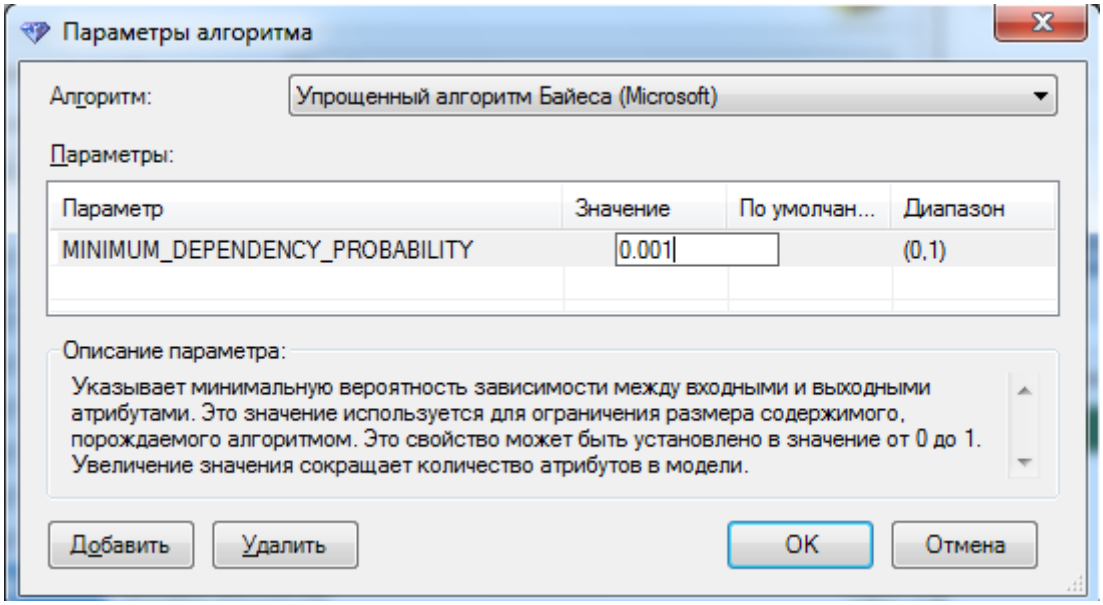

Рис. 46. Окно задания параметров классификации

Далее задается размер тестирующей выборки (остальная часть выборки будет использоваться для обучения), имя структуры («Классификация Приобрел велосипед») и имя модели («Алгоритм Байеса»). После завершения диалога аналитические службы сервера создают структуру интеллектуального анализа и модель в структуре, получают данные из Excel, выполняют обучение — настройку модели и открывают окно просмотра модели. В окне на вкладке «Сеть зависимостей» (Рис. 47) отображаются влияющие факторы. Степень влияния можно устанавливать бегунком слева.

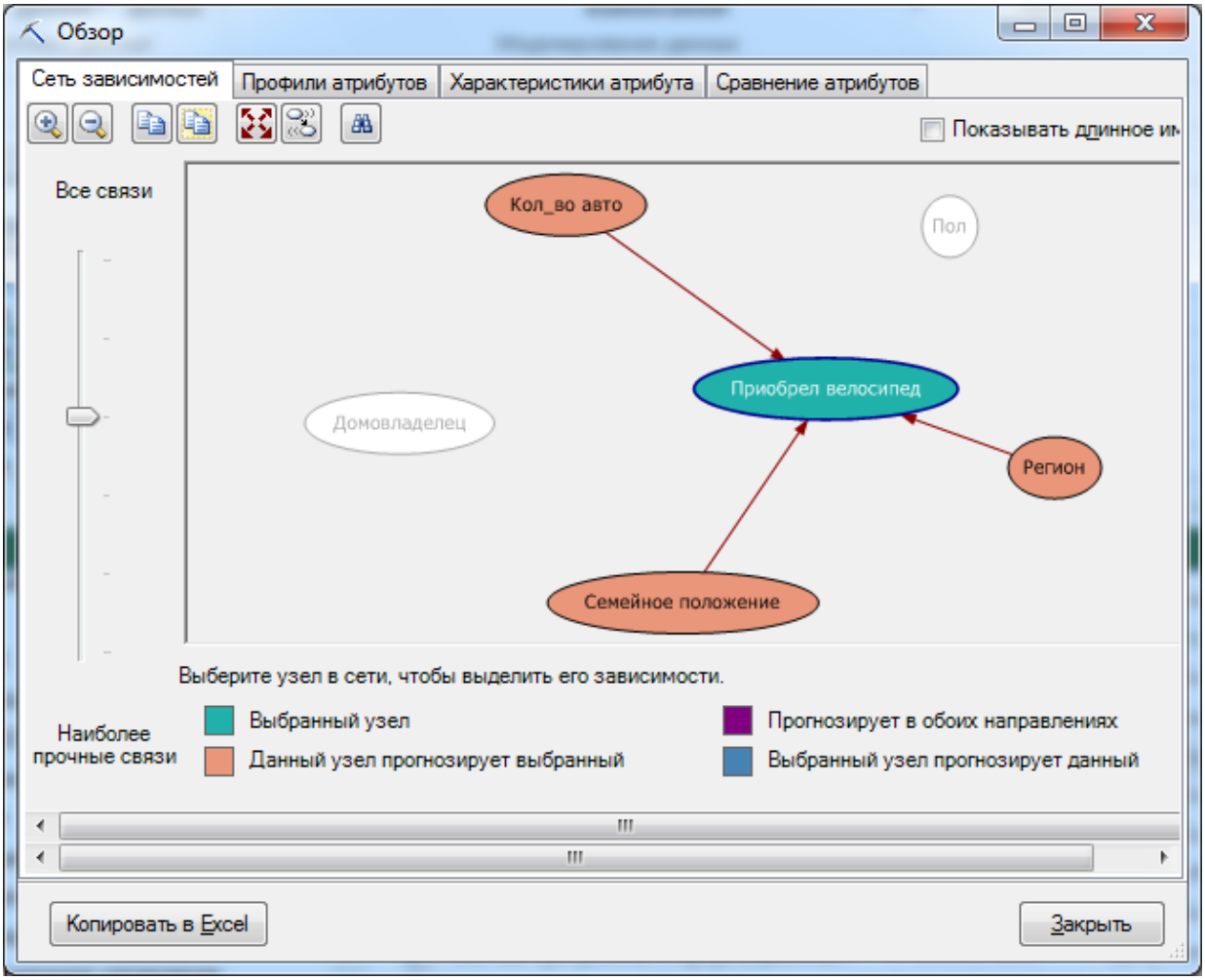

Рис. 47. Вкладка «Сеть зависимостей» окна обзора модели

На вкладке «Профили атрибутов» (Рис. 48) можно отследить как меняются частоты. Например, доля семейных клиентов составляет 55,3 %, а для покупателей велосипедов — 49,1 %. Это говорит о незначительном отрицательном влиянии семейного положения на решение купить велосипед: семейные клиенты меньше склонны к покупке велосипедов. Отсутствие автомобиля наоборот увеличивает шансы продать велосипед клиенту. Сравнивая исходные значения частот с частотами для отдельных классов можно выделить атрибуты, частоты которых изменяются значительно. Такие атрибуты и оказывают влияние на классификацию. Аналогичная информация представлена и на других вкладках (Рис. 49).

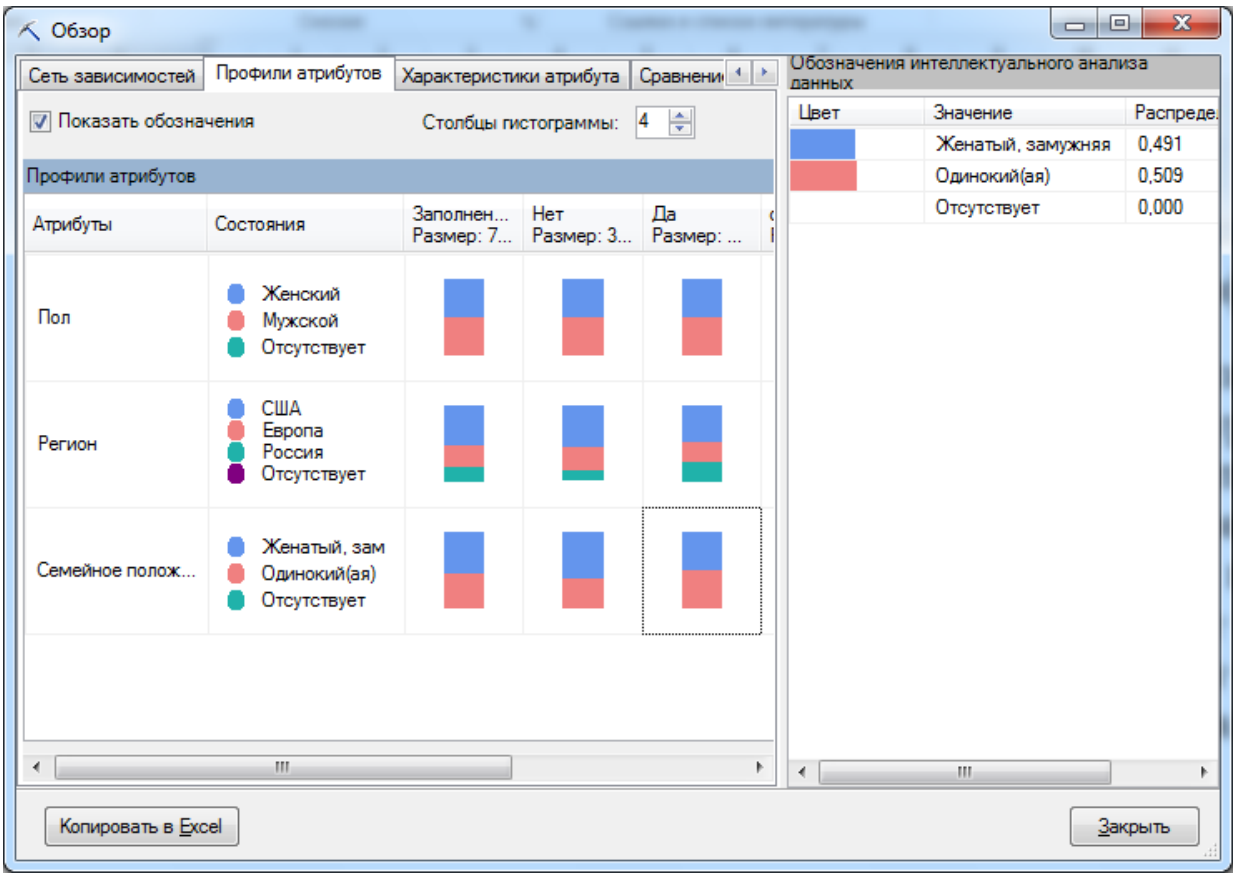

Рис. 48. Вкладка «Профили атрибутов» окна обзора модели

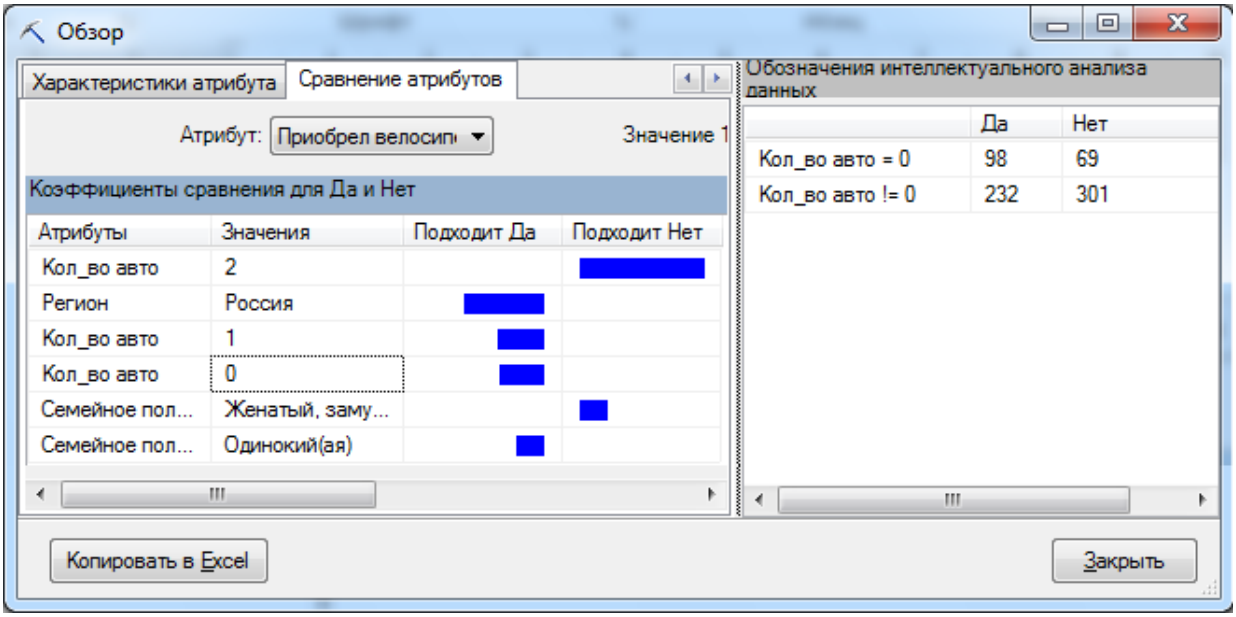

Рис. 49. Вкладка «Сравнение атрибутов» окна обзора модели

3.3.2. Деревья решений

Дерево решений — это способ записи алгоритма в виде иерархического дерева. В основе деревьев решений лежат решающие правила вида «если... то...». В каждом узле размещается одно правило и каждая дуга из вершины к потомку — дочерней вершине содержит условие перехода, а дочерняя вершина содержит вывод (терминальная вершина) или поддерево решений.

Узлу соответствует множество объектов, удовлетворяющих условиям, записанным в дугах на пути к родительской вершине. Каждый признак может быть использован для деления на несколько ветвей (поддеревьев).

На Рис. 50 представлено дерево решений для анализа заемщиков. Из рисунка видно, что система не дает однозначного решения, а дает вероятностные оценки. Причем, чем больше распределение частот отличается от корневого, тем полезнее деление. В идеале частоты должны равняться 1 или 0. Это соответствует детерминированной классификации. Однако очень редкие моделируемые ситуации полностью избавлены от случайных возмущений. Можно добиться детерминированной классификации для обучающего множества, однако это скорее всего приведет к переобучению — плохим результатам для тестового множества.

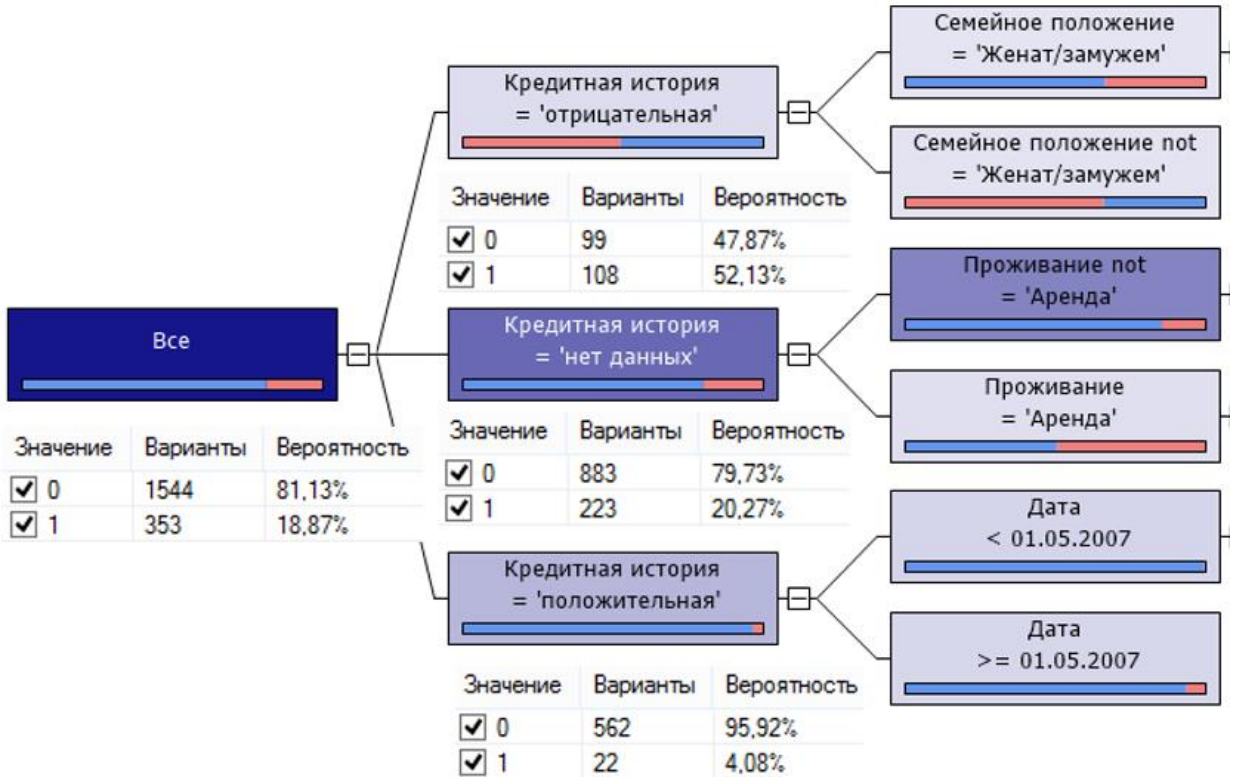

Рис. 50. Фрагмент дерева решений

Поиск решения в дереве начинается с корневой вершины. Для примера с Рис. 50 сначала проверяется кредитная история. Если кредитная история отсутствует, то проверяется атрибут «проживание» и т.д. Поиск решения заканчивается в терминальной вершине, которая содержит частоты для каждого возможного класса.

Выбор решающего правила должен приводить к предпочтению в выборе класса. Если частота перехода близка к долям классов, то такое деление не делает выбор более определенным, чем выбор наугад. Для дискретного значения атрибута достаточно рассчитать, как каждое значение меняет частоты (Рис. 50), чтобы оценить ценность правила.

Если атрибут является вещественным числом, то разбиение осуществляется в точке нелинейности, а каждый узел содержит регрессионную формулу (Рис. 51).

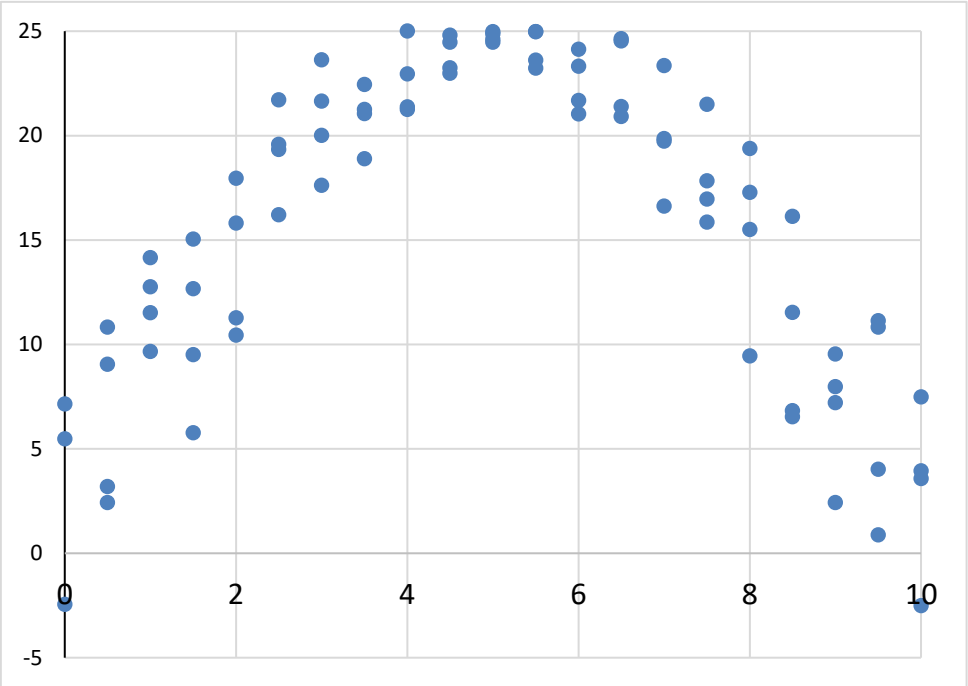

Рис. 51. Разбиение в случае вещественных атрибута и выходной переменной

На Рис. 52 представлено ветвление для зависимости непрерывных переменных.

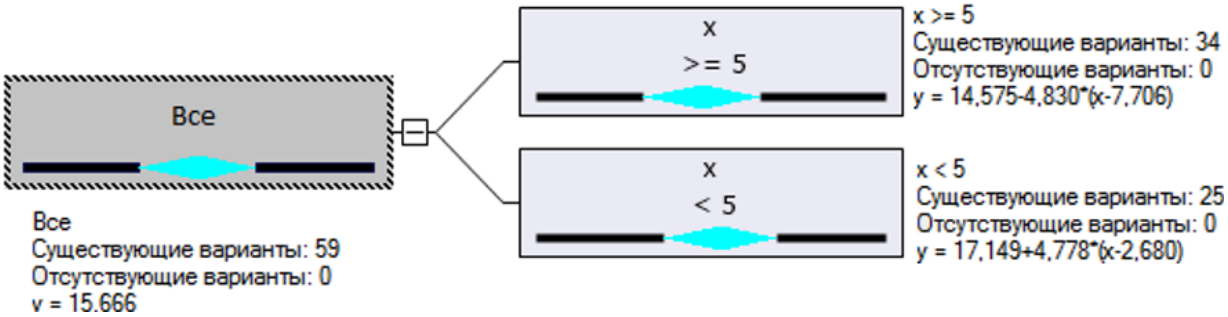

Рис. 52. Дерево решений в случае вещественных атрибута и выходной переменной

Каждое решающее правило связано с разбиением множества наблюдений на подмножества — каждое подмножество определяется условиями ветвления.

Построение дерева решений не является однозначным процессом — в качестве атрибута для разбиения может быть выбран любой входной атрибут. Варианты разбиения сравниваются с помощью разных критериев эффективности разбиения:

- индекс Джини;
- уменьшение дисперсии;
- информационный критерий;
- критерий хи-квадрат;
- *F* тест.

Мера Джини для узла представляет собой простую сумму квадратов долей классов в узле. Минимум меры — 0,5 = $(0,5)^2 + (0,5)^2$ — достигается, если доли двух классов равновероятны, максимум — 1 = $1^2 + 0^2$ — достигается, когда доля одного класса равна 1. Таким образом, чем ближе правило к детерминированному, тем выше мера Джини.

Дисперсия является мерой разброса. Нулевая дисперсия соответствует детерминированному правилу. Следовательно, разбиение с меньшей дисперсией более эффективно.

Энтропия — мера неопределенности разбиения вычисляется по формуле $I = p_0 log_2(p_0) + p_1 log_2(p_1) + \ldots$, где p_k — вероятность (доля) k-го класса в разбиении. Уменьшение энтропии свидетельствует о более определенном (эффективном) выборе.

Тест хи-квадрат позволяет оценить различие частот. Если в результате разбиения частоты потомков сильно отличаются от частот родителя, такое разбиение более эффективно.

F-критерий позволяет сравнить разбиение на классы: чем меньше отклонения внутри классов и больше отклонения классов друг от друга, тем эффективней разбиения.

Очевидно, что все критерии стремятся выбрать более детерминированный вариант классификации.

Как уже упоминалось, в процессе построения дерева важно не допустить переобучения — получение детерминированной классификации на обучающей выборке. В случае наличия случайных отклонений в наблюдениях это приведет к попытке построения правил, принимающих эти отклонения за детерминированную закономерность. Это, в свою очередь, приведет к плохим показателям (высокая доля неправильно классифицированных наблюдений) для тестового множества. Другая крайность — недообучение — возникает, если дерево получается слишком простым и ряд закономерностей относит к случайным отклонениям. Итак, основной признак и недообучения, и переобучения — это большая ошибка на тестовом множестве.

3.3.3. Добавление модели классификации «Дерево решений» к существующей структуре в MS Excel

В параграфе «Использование упрощенного алгоритма Байеса в MS Excel» приводился пример создания структуры и модели классификации. Для сравнения алгоритмов, решающих одинаковую задачу, удобно определять их в одной структуре. Для этого в меню «Интеллектуальный анализ данных» предусмотрена команда «Дополнительно» \ «Добавить модель к структуре» (Рис. 53). Далее предлагается выбрать алгоритм (Рис. 54) и параметры алгоритма (Рис. 55). На следующем шаге мастера нужно определить использование «Только прогноз» для столбца «Приобрел велосипед» (Рис. *56*).

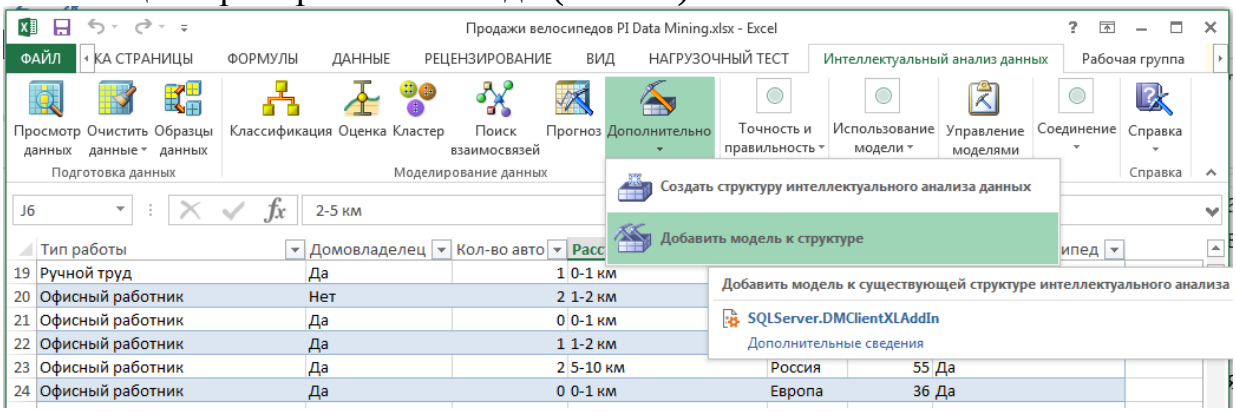

Рис. 53. Команда добавления модели к структуре

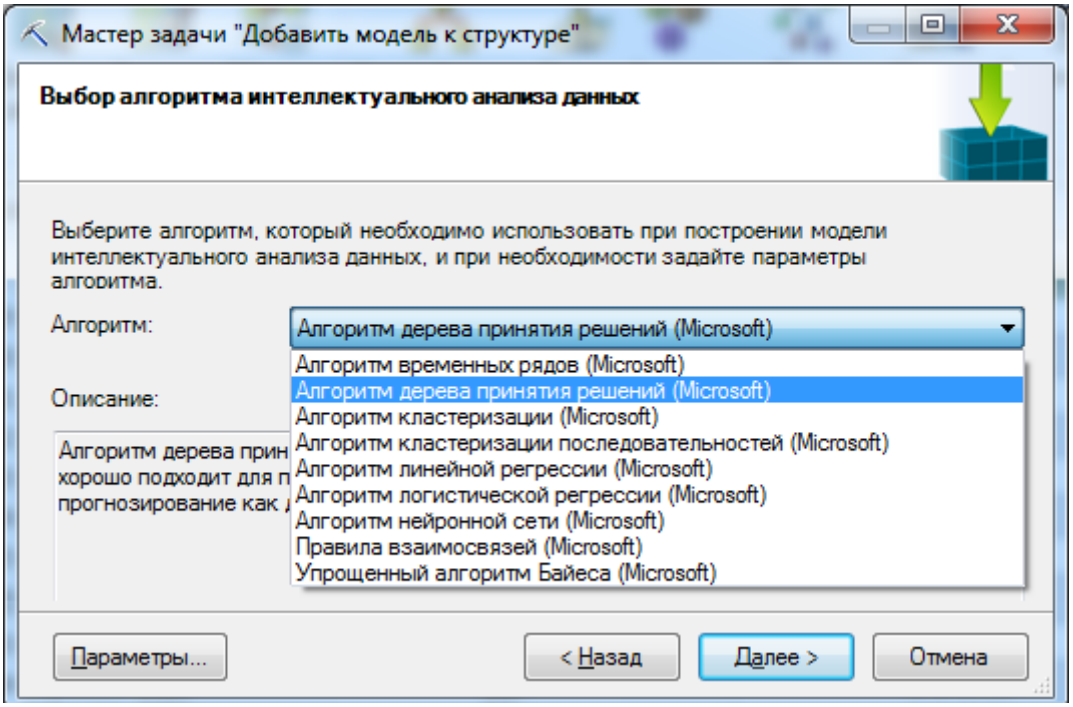

Рис. 54. Выбор алгоритма

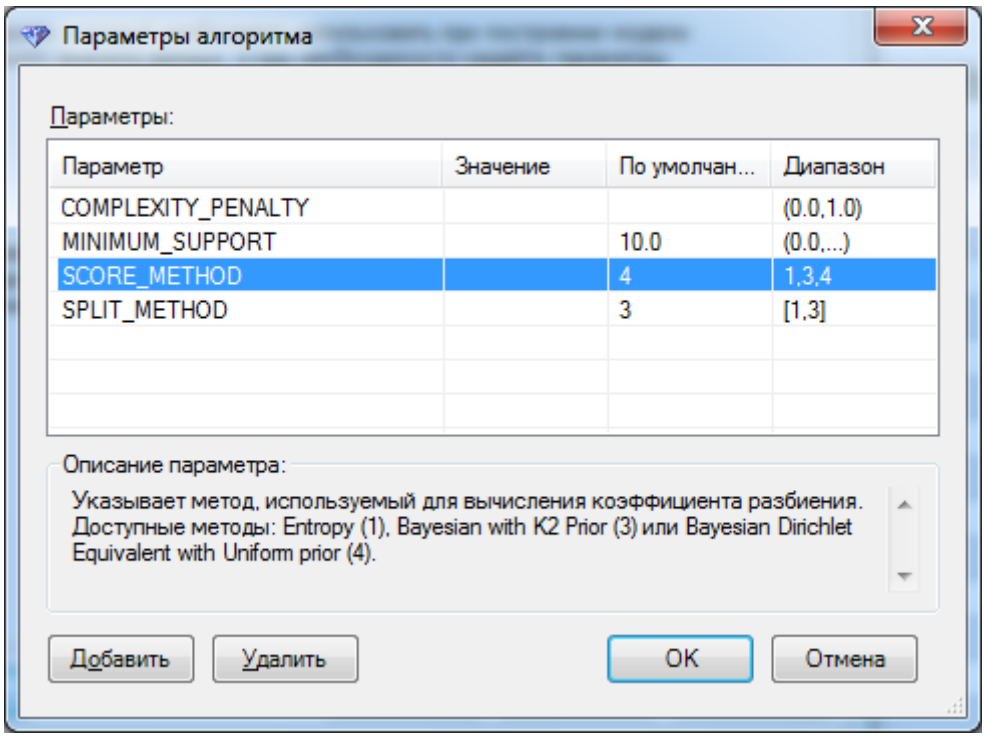

Рис. 55. Определение параметров алгоритма «дерево решений»

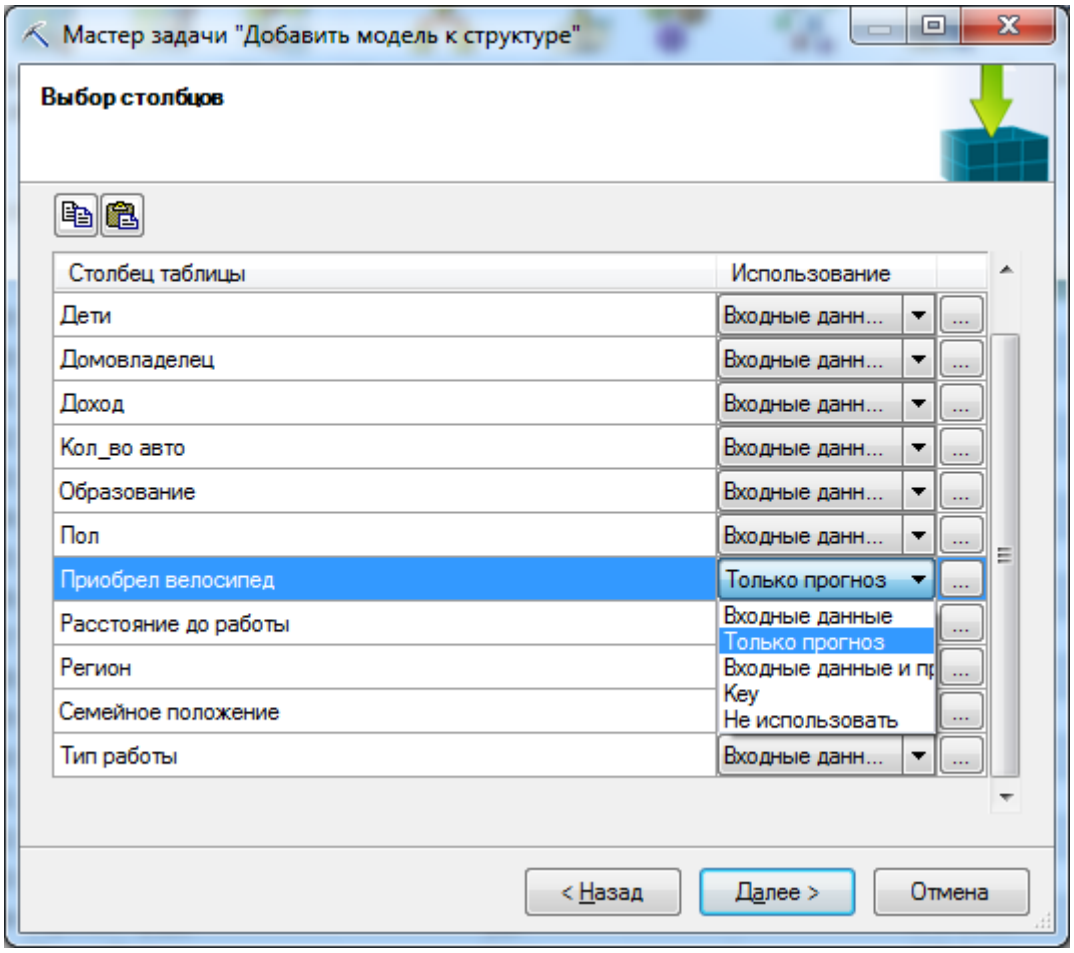

Рис. 56. Указание типа столбца таблицы

После обучения включается окно просмотра модели, которое в данном случае будет отображать дерево решений (Рис. 57) и зависимости атрибутов (Рис. 47).

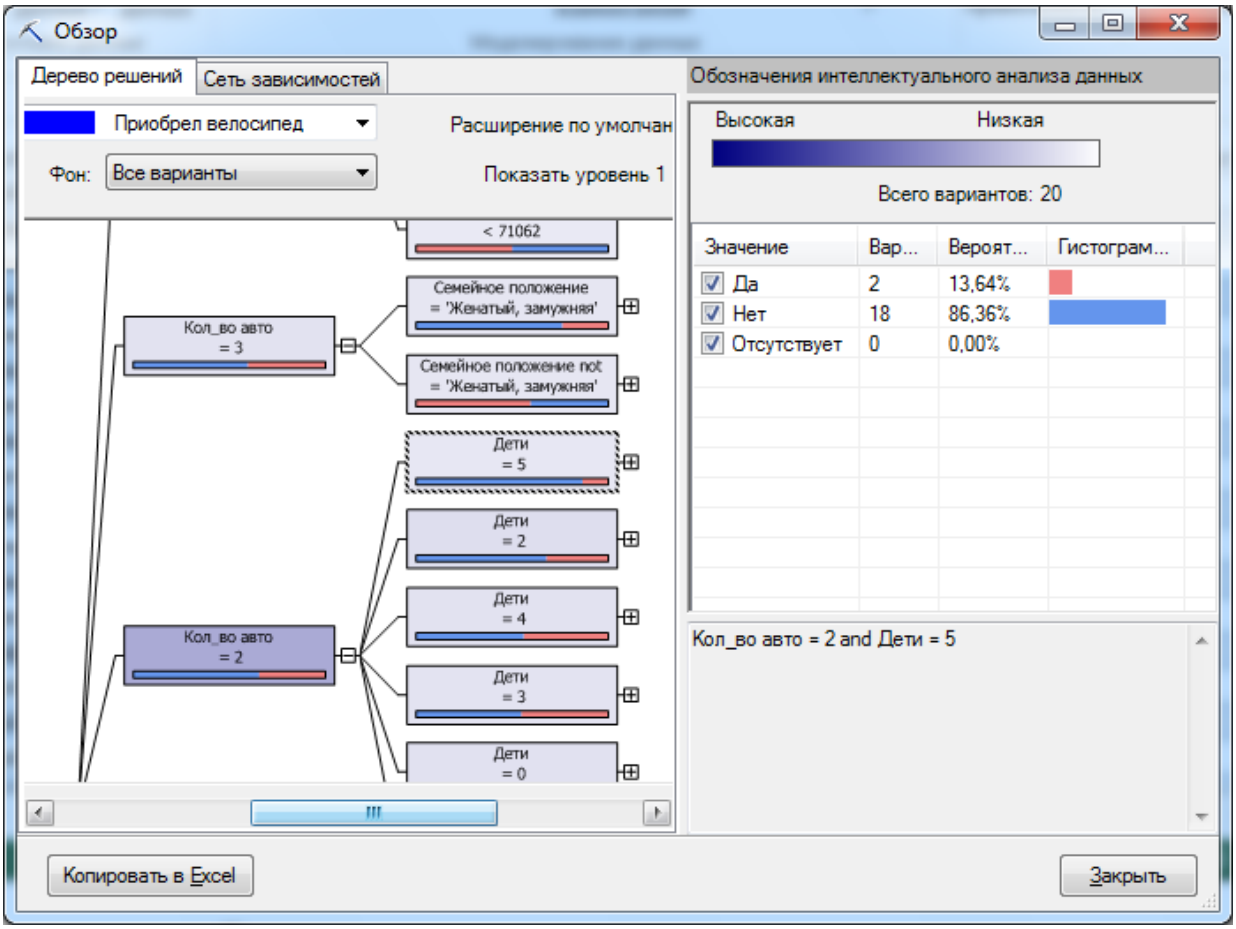

Рис. 57. Вкладка «Дерево решений» в окне просмотра модели

3.3.4. Логистическая регрессия

Логистическая регрессия является известным статистическим методом для определения влияния нескольких факторов на дихотомический результат. В частности, логистическую регрессию можно применять для решения задачи классификации с двумя классами.

Уравнение регрессии можно рассматривать как условное математическое ожидание $\boldsymbol{y = y(x) = M[y|X=x]}$. Если y принимает два значения 0 или 1, то математическое ожидание $\boldsymbol{y}$ есть вероятность $\boldsymbol{p(x)}$ единичного значения при условии $\boldsymbol{X = x}$. Линейную функцию используют для аппроксимации логарифма отношения вероятностей единицы и нуля

$$\ln\left(\frac{p(x)}{1-p(x)}\right) = x_0 + b_1 x_1 + \ldots + b_n x_n .$$

В результате $p(x)$ описывается логистической функцией (сигмоидой)

$$p(x) = \frac{\exp(x_0 + b_1 x_1 + ... + b_n x_n)}{1 + \exp(x_0 + b_1 x_1 + ... + b_n x_n)}$$

На рис. 57 представлены результаты обработки наблюдений — эмпирических частот для различных вариантов долей кредита в доходе заемщика (ось ***x*** от 0 до 100 %). На рисунке приведены полученная логистическая кривая для вероятности и логарифм отношения вероятностей (на вспомогательной правой оси). Применение логистической регрессии для решения задачи классификации аналогично применению других моделей классификации. Оно заключается в определении по атрибутам объекта вероятности его принадлежности к положительному или отрицательному классу. Отличие заключается в том, что для логистической регрессии можно не выделять тестовое множество и определять параметры по всем наблюдениям.

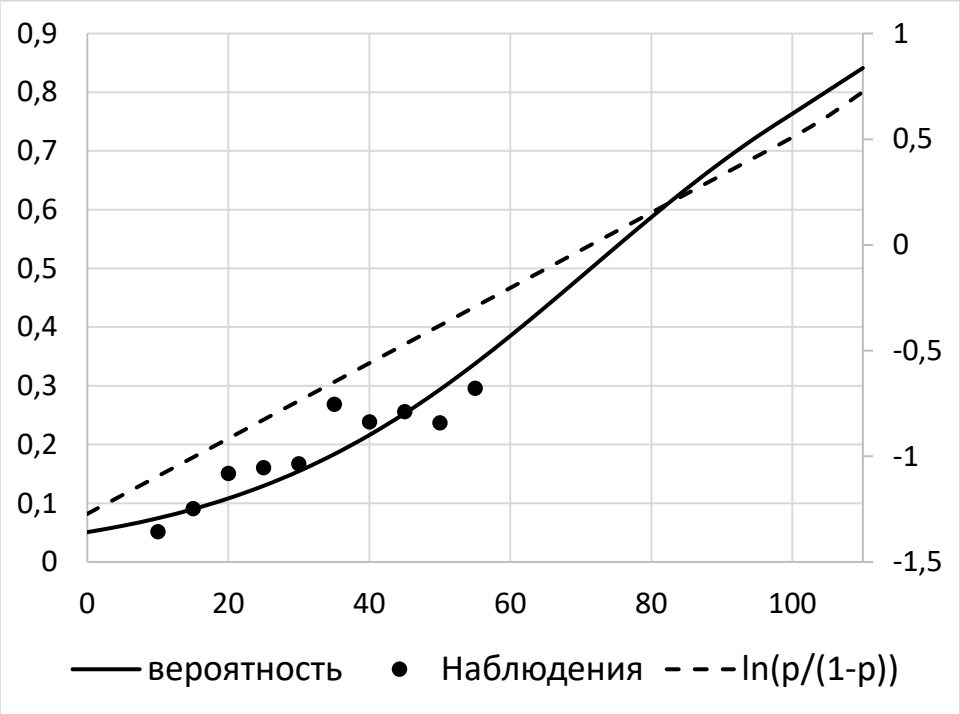

Рис. 58. Логистическая регрессия

Для добавления модели классификации «Алгоритм логистической регрессии» к существующей структуре в MS Excel в меню «Интеллектуальный анализ данных» нужно выбрать команду «Дополнительно» \ «Добавить модель к структуре» (Рис. 53). Далее следует выбрать алгоритм «Алгоритм логистической регрессии». На следующем шаге мастера нужно установить характеристику использования «Только прогноз» для столбца «Приобрел велосипед». В обзоре модели логистической регрессии (Рис. 59) приводятся вероятностные оценки влияния атрибутов на класс объекта.

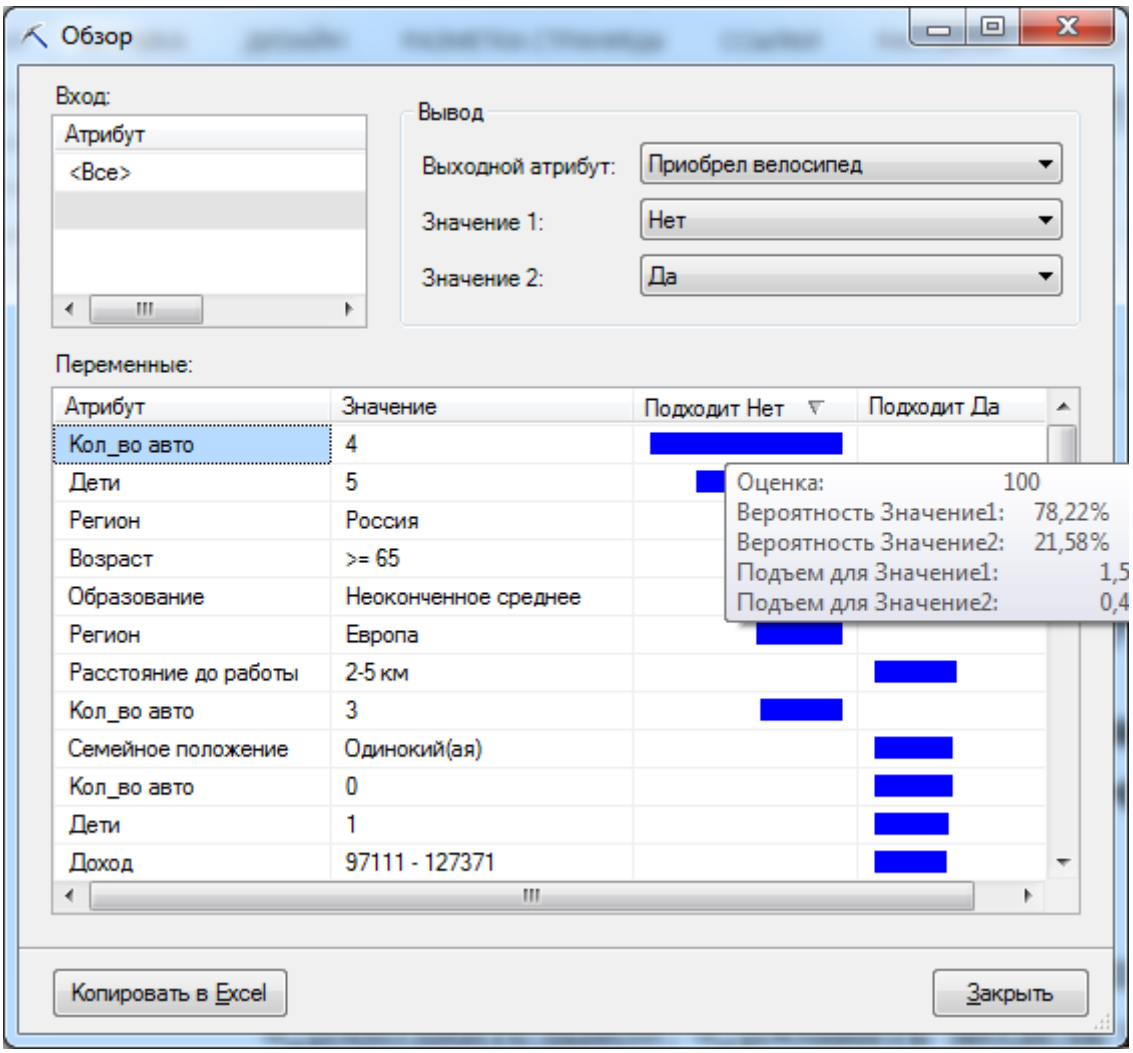

Рис. 59. Обзор модели логистической регрессии

3.3.5. Нейронные сети

Искусственные нейронные сети построены по аналогии с работой мозга. Нейронные сети могут иметь разные конфигурации. Нейронную сеть делят на слои. Входной слой соответствует множеству аргументов-признаков. Каждый нейрон имеет несколько входов и один выход. Выходы одного слоя передаются на входы следующего. Нейроны выходного слоя соответствуют выходным переменным.

В математическом смысле искусственный нейрон предусматривает следующую обработку входных для нейрона сигналов x_1, …, x_n. Каждое значение *xi* умножается на вес связи *wi* и суммируется со взвешенными значениями других сигналов. Полученная сумма

$$S = w_0 + w_1x_1 + \ldots + w_nx_n$$

подвергается нелинейному преобразованию, переводящему сумму к выходному значению нейрона, принадлежащего стандартному интервалу, чаще всего [0,1]. Наиболее популярным является логистическое преобразование (сигмоида), приведенное на Рис. 60.

$$y = 1/(1 + \exp(-aS)).$$

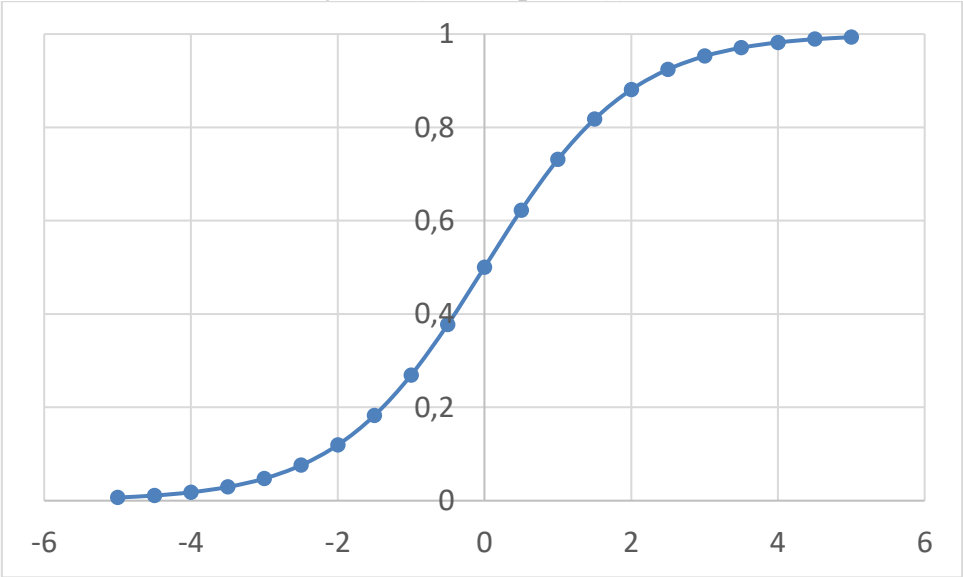

Рис. 60. Логистическое преобразование — сигмоида

Кроме параметров каждого нейрона нейронная сеть характеризуется архитектурой: количеством слоев, связями нейронов (см. Рис. 61). Искусственную нейронную сеть, в которой присутствует хотя бы один скрытый слой, называют многослойным персептроном (multilayer perceptron, MLP).

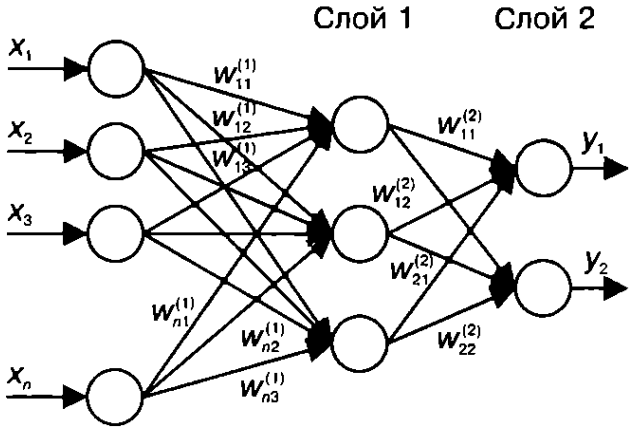

Рис. 61. Пример нейронной сети

Построение и обучение нейронной сети являются скорее искусством, чем наукой. Известны отдельные удачные конфигурации для решения некоторых частных задач. Например, для решения задачи кластеризации применяют сети Кохонена, которые являются двухслойными нейронными сетями. Известны только общие рекомендации [1]:

1. Число нейронов во входном и выходном слоях жестко определяется числом входных и выходных переменных модели соответственно.

2. Число нейронов в скрытых слоях и число скрытых слоев выбираются таким образом, чтобы количество образованных ими связей было меньше числа обучающих примеров как минимум в два-три раза.

Обучение нейронной сети сводится к подбору весов параметров. Эту задачу сводят к определению таких значений параметров, которые соответствуют минимуму ошибки.

1. На начальной стадии случайным образом присваиваются значения всем весам всех входов в сети. Значения обычно берутся из интервала (–1,1).

2. Для каждого обучающего варианта вычисляются выходы.

3. Вычисляются ошибки выходов. В качестве функции ошибки может использоваться квадрат остатка (квадрат разности между спрогнозированным и фактическим значением).

4. Шаги 2,3 повторяются для всех вариантов, используемых в качестве образцов. После этого веса в сети обновляются таким образом, чтобы минимизировать ошибки.

В процессе обучения может выполняться несколько итераций. После прекращения роста точности модели обучение завершается.

Для добавления модели классификации «Алгоритм нейронной сети» к существующей структуре в MS Excel в меню «Интеллектуальный анализ данных» нужно выполнить команду «Дополнительно» \ «Добавить модель к структуре» (Рис. 53). Далее выбрать «Алгоритм нейронной сети». На следующем шаге мастера следует выбрать характеристику использования «Только прогноз» для столбца «Приобрел велосипед». В результате обучения модели будут найдены параметры нейронной сети, минимизирующие ошибку классификации. В итоге демонстрируется оценка влияния атрибутов на класс клиента (

Рис. *62*).

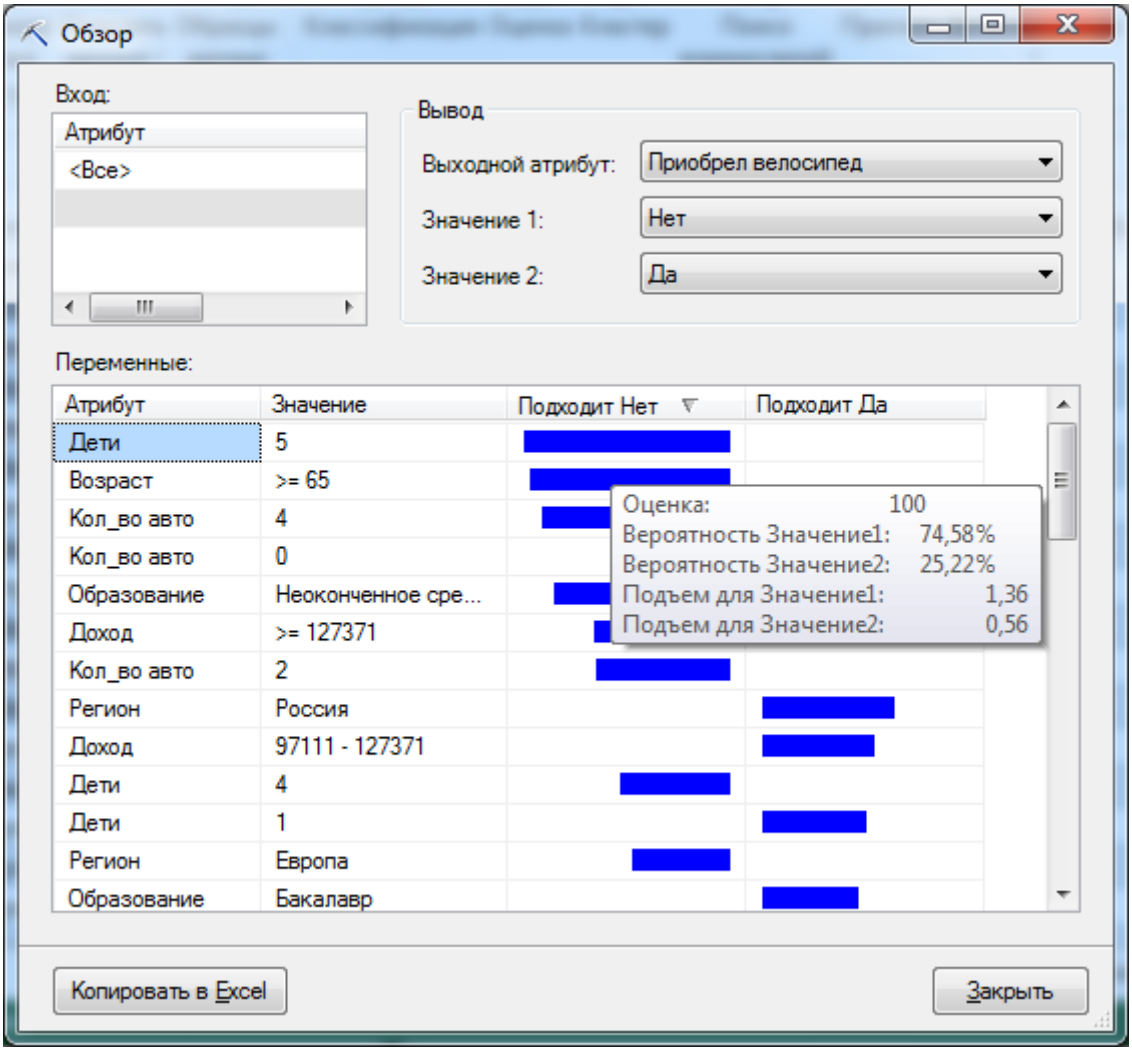

Рис. 62. Обзор модели нейронной сети

3.3.6. Просмотр структур и моделей интеллектуального анализа в SQL Server Management Studio

Все параметры и характеристики моделей могут быть просмотрены в Management Studio. Для этого в этой программе достаточно подключиться аналитическим службам соответствующего сервера (Рис. 63).

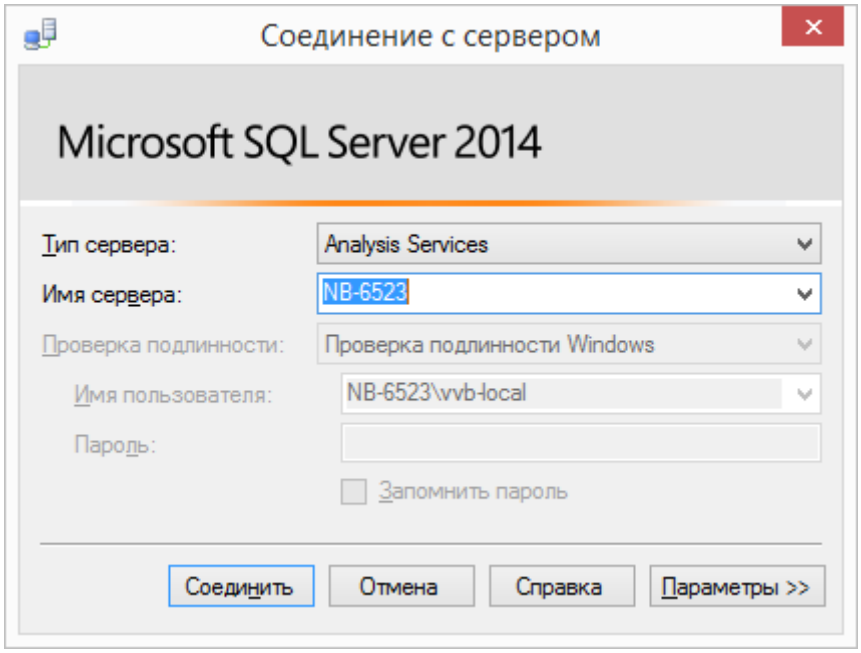

Рис. 63. Подключение к аналитическим службам сервера

В обозревателе Management Studio демонстрируются аналитические базы данных и созданные в них структуры. Обзор (browse) модели формирует такую же информацию, которая уже была рассмотрена выше. Обзор моделей также может быть запущен с помощью настройки Excel командой меню «Интеллектуальный анализ данных»\ «Использование модели» \ «Обзор» — Рис. 64.

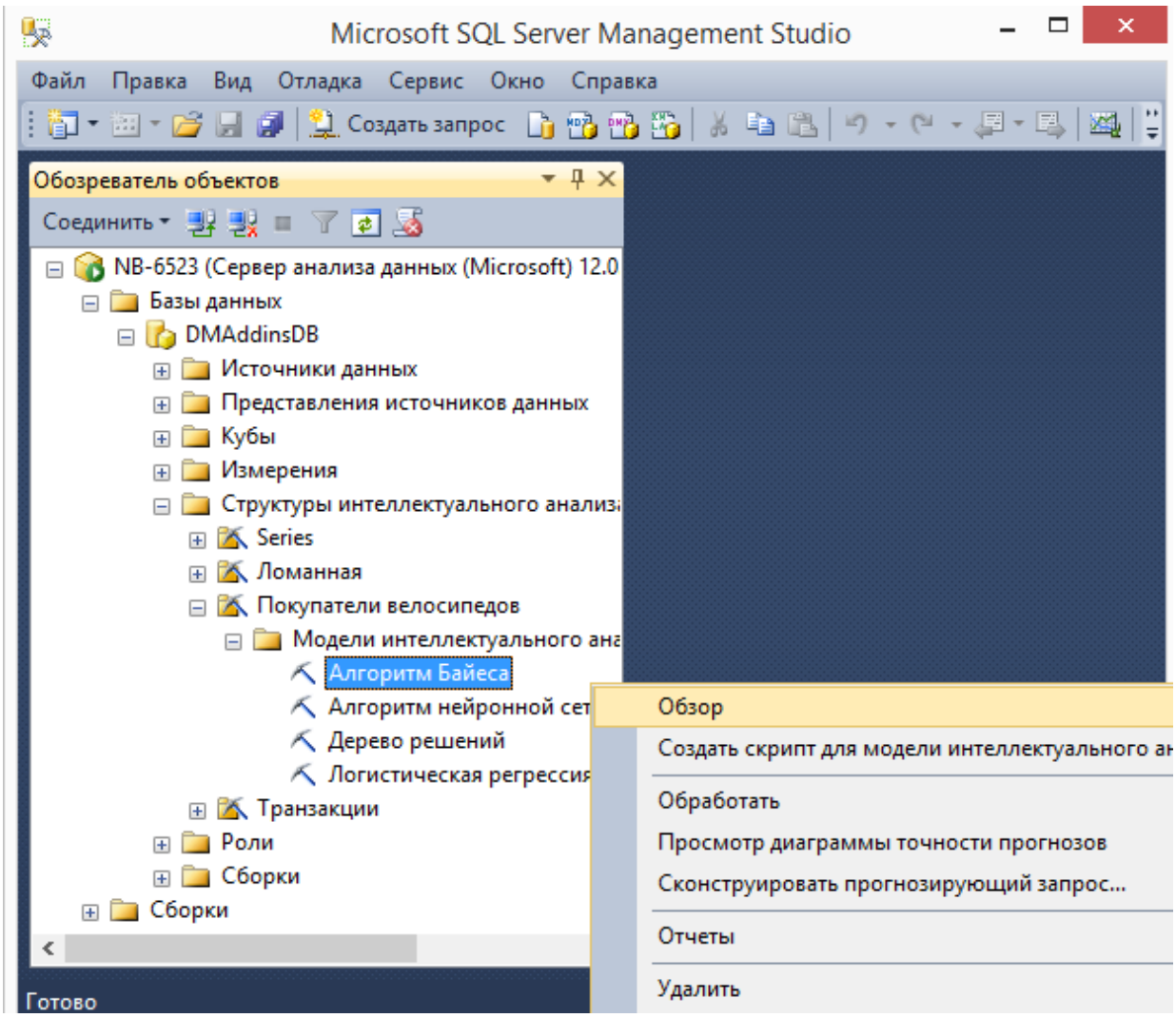

Рис. 64. Обзор моделей интеллектуального анализа в Management Studio

3.3.7. Точность и эффективность классификации

В силу случайности закономерностей, лежащих в основе классификации, результаты работы алгоритма могут оказаться ошибочными, с определенными ненулевыми вероятностями. В самом распространенном случае классификации все объекты делят всего на два класса. Один класс обычно означает положительное решение — «Да» (например, выдача кредита), другой — отрицательное — «Нет». Для этого случая исследование ошибок выглядит особенно наглядным и понятным. Действительно, может быть четыре результата классификации (см. табл. 1):

– истинноположительный (true positive, TP) — положительный класс модели совпадает с наблюдаемым;

– истинноотрицательный (true negative, TN) — отрицательный класс модели совпадает с наблюдаемым;

– ложноположительный (false positive, FP) — положительный класс модели не совпадает с наблюдаемым;

– ложноотрицательный (false negative, FN) — отрицательный класс модели не совпадает с наблюдаемым.

Таблица 5

Варианты бинарной классификации

Результаты определения класса алгоритмом	Результаты наблюдений	
	Да	Нет
Да	TP	FP
Нет	FN	TN

Для характеристики точности применяют несколько показателей. Прежде всего, это **общий показатель успеха** (overall success rate, ***OSR***), или просто **точность (accuracy)**

$$OSR = (TP + TN) / (TP + TN + FP + FN).$$

Точность показывает общую долю правильно классифицированных объектов. Не следует игнорировать невысокую точность — низкую вероятность правильной классификации. Если целевая аудитория для рекламного объявления составляет 5 %, то выбор адресатов без учета закономерности приведет к тому, что 95 % объявлений будет разослано зря. Если за счет решения задачи классификации удастся правильно выбрать 10 % адресатов, то эффективность рассылки увеличится в два раза.

Как правило, особый интерес представляет частота правильных классификаций положительных наблюдений. Этот показатель назвали **чувствительность**

$$Se = TP / (TP+FN).$$

Другой показатель — **специфичность**

$$Sp = TN / (TN + FP).$$

определяет частоту правильных классификаций отрицательных наблюдений.

Для данных из табл. 2 показатели будут иметь следующие значения:

$$OSR = (84+100)/(84+100+56+60) = 0{,}613333$$

$$Se = 84 /(84+56) = 0{,}6$$

$$Sp = 100/(60+100) = 0{,}625$$

Таблица 6

Примеры частот бинарной классификации

Результаты определения класса алгоритмом	Результаты наблюдений	
	Да	Нет
Да	84	60
Нет	56	100

Объект относят к классу, вероятность которого максимальна. Для бинарной классификации — это означает выбор класса, вероятность которого больше 0,5. Понятно, что такой выбор может быть очень надежным, если вероятность равна 0,99 или очень ненадежным, если вероятность равна 0,5. Если риски ведут к большим финансовым потерям (например, предоставление кредита ненадежному заемщику), то применяют точку отсечения ***p*** — пороговую вероятность положительного решения (принадлежности объекта к положительному классу).

Если вычисленная алгоритмом вероятность не меньше точки отсечения, то принимается положительное решение. Точка отсечения определяется по экономическим соображениям — с каждым вариантом классификации связывается доход или потери (табл. 7).

Изменяя p, можно получать разные соотношения средних доходов и расходов. Значение точки отсечения можно определить, добиваясь соответствия точки отсечения максимуму среднего дохода.

Таблица 7

Доходы и потери бинарной классификации заемщиков

Результаты определения класса алгоритмом	Результаты наблюдений	
	Да	Нет
Да	Доход от правильного положительного решения (проценты по кредиту)	Убытки от неправильного положительного решения (невозврат кредита)
Нет	Убытки от неправильного отрицательного решения (упущенная выгода)	Доход от правильного отрицательного решения (сохранение денежных средств для предоставления кредита надежным заемщикам)

Если установить точку отсечения, близкую к 1, то в положительном классе этой вероятности будет соответствовать небольшому количеству объектов, классифицируемых практически без ошибок. Однако в этом случае много наблюдений положительного класса будет ошибочно отнесено к отрицательному классу, и чувствительность будет близка 0 (Рис. 65). При этом в отрицательном классе практически не будет наблюдений, ошибочно отнесенных к положительному классу, и почти все наблюдения этого класса будут правильно классифицированы алгоритмом. Поэтому специфичность будет близка к 1. Если точка отсечения будет близка к 0 (почти все наблюдения относятся алгоритмом к положительному классу), то ситуация будет противоположной.

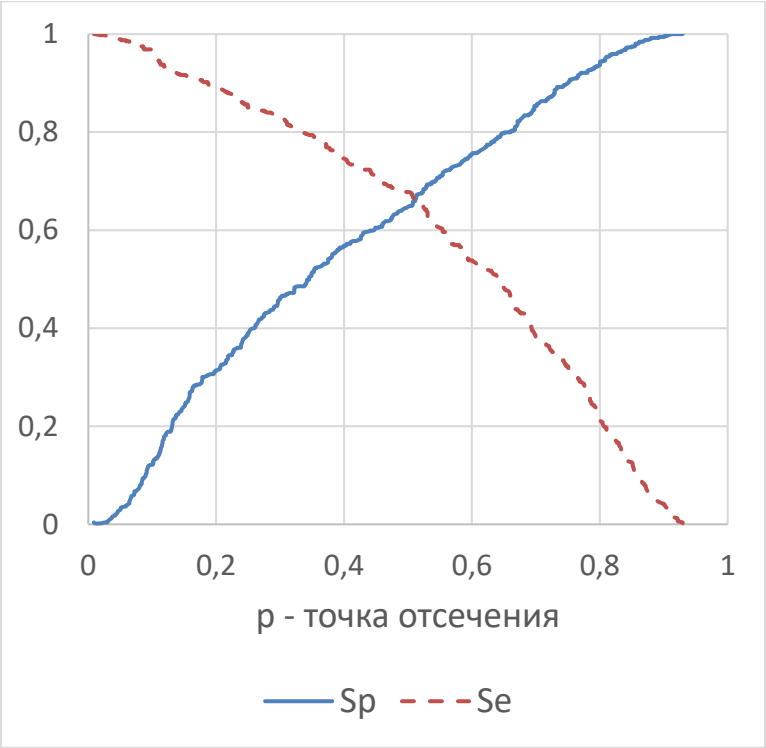

Рис. 65. Зависимость чувствительности ***Se*** и специфичности ***Sp*** от точки отсечения ***p***

Иногда используют график ROC-кривой (Рис. 66), на котором отображают зависимость чувствительности от (1-Se). Такая кривая позволяет определить наилучшее соотношение чувствительности и специфичности.

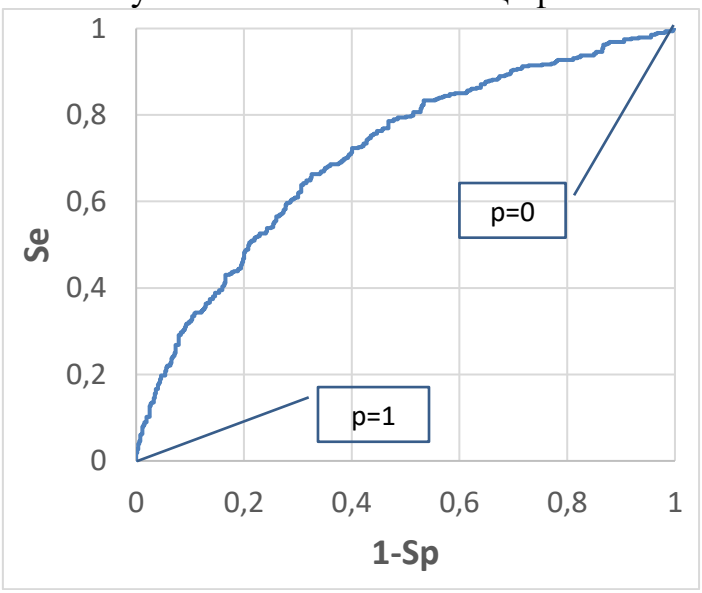

Рис. 66. ROC-кривая

Одной из характеристик эффективности алгоритма классификации является лифт (Lift)

$$L = (k / K) / (n / N),$$

где N — количество объектов;

n — количество объектов в положительном классе;

K — количество «перспективных» объектов, выбранных с помощью соответствующей точки отсечения;

k — количество верно классифицированных объектов среди ***K*** выбранных.

Лифт показывает во сколько раз алгоритм классификации эффективнее выбора наугад. График лифта приведен на

Рис. *67*. По графику видно, что при небольшом количестве наиболее перспективных объектов лифт максимален, а когда выбранные объекты соответствуют всей выборке, то лифт равен своему минимальному значению — 1. Чаще применяют лифт-диаграмму (Рис. 68), в которой для выбранной доли наиболее перспективных объектов определяют долю истинно положительных объектов. Для сравнения приводят идеальный классификатор и выбор наугад (лифт в среднем равен 1). Чем ближе классификатор к идеальному, тем он эффективнее.

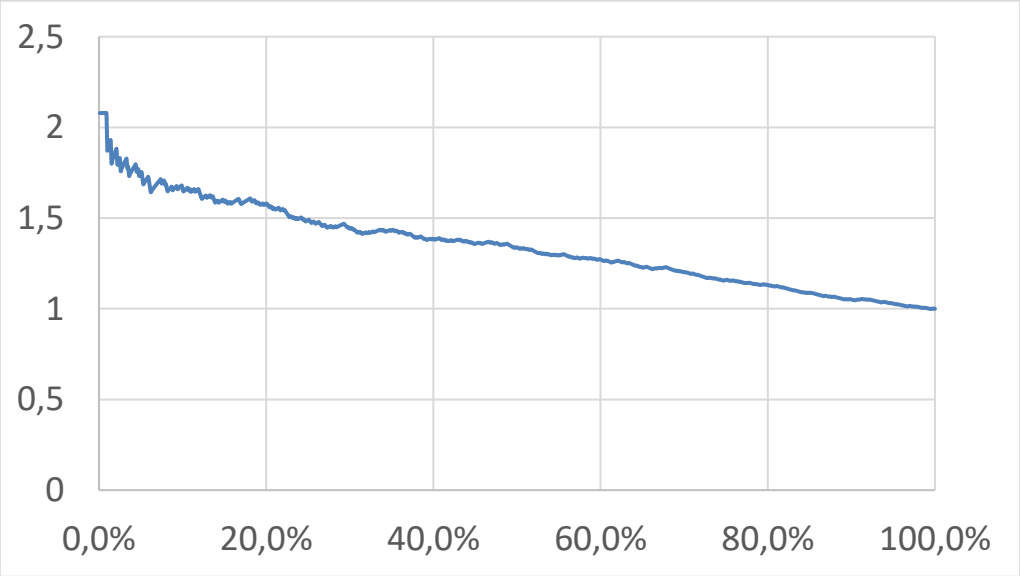

Рис. 67. График лифта в зависимости доли *K / N* выбранных объектов

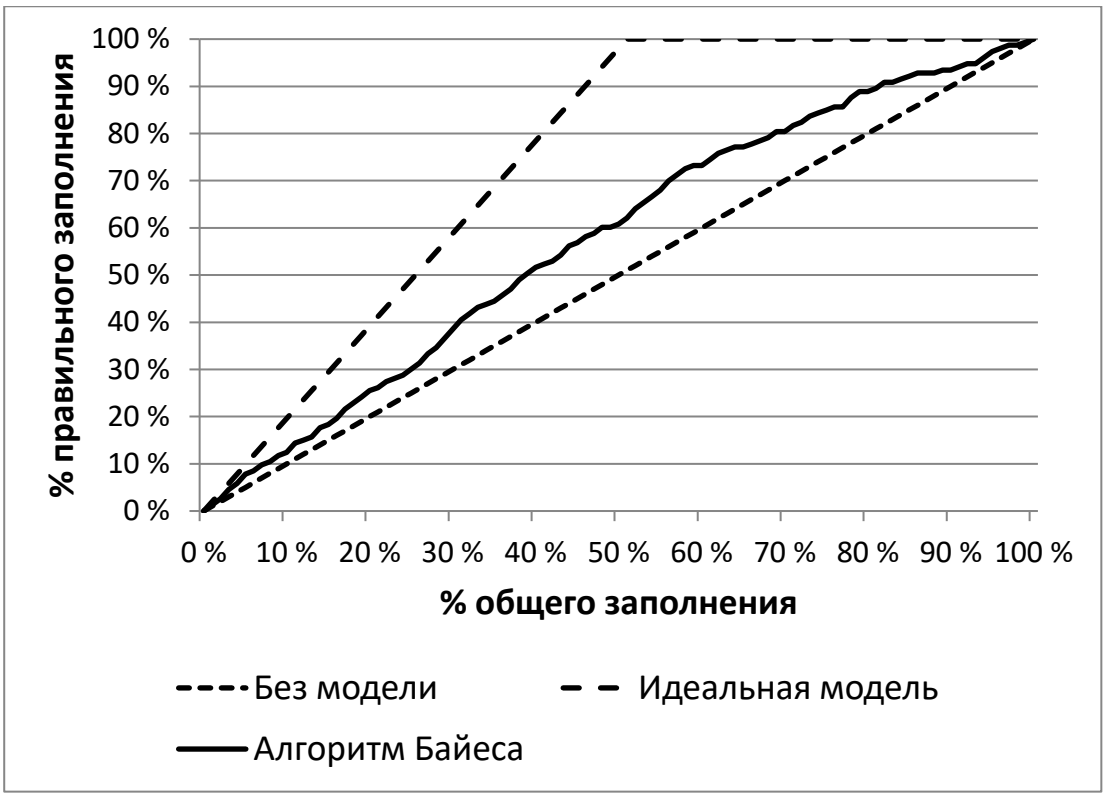

Рис. 68. Лифт-диаграмма

Приведенные характеристики позволяют не только оценивать эффективность алгоритма классификации, но и сравнивать эффективности разных алгоритмов, построенные для одних данных.

Приведенные графики демонстрируют общую особенность измерений — все характеристики являются оценками случайных величин, а, следовательно, сами являются случайными, хотя и с небольшой мерой разброса.

Оценка точности и эффективность классификации в Excel собраны в следующих командах меню «Интеллектуальный анализ данных»\ «Точность и правильность»:

– «Лифт-диаграмма» — построение одноименной диаграммы (диаграммы точности прогноза класса);

– «Матрица классификации» — вычисление показателей общего показателя успеха, чувствительности и специфичности в табличном виде в процентах и количестве наблюдений.

Специальный пункт «Диаграмма роста прибыли» позволяет построить кривую роста прибыли в зависимости от точки отсечения. В качестве параметров (Рис. 69) задают

– фиксированные затраты;

– заполнение — количество объектов (клиентов, продаж);

– индивидуальные затраты диаграммы роста прибыли (на одного клиента);

– доход на единицу.

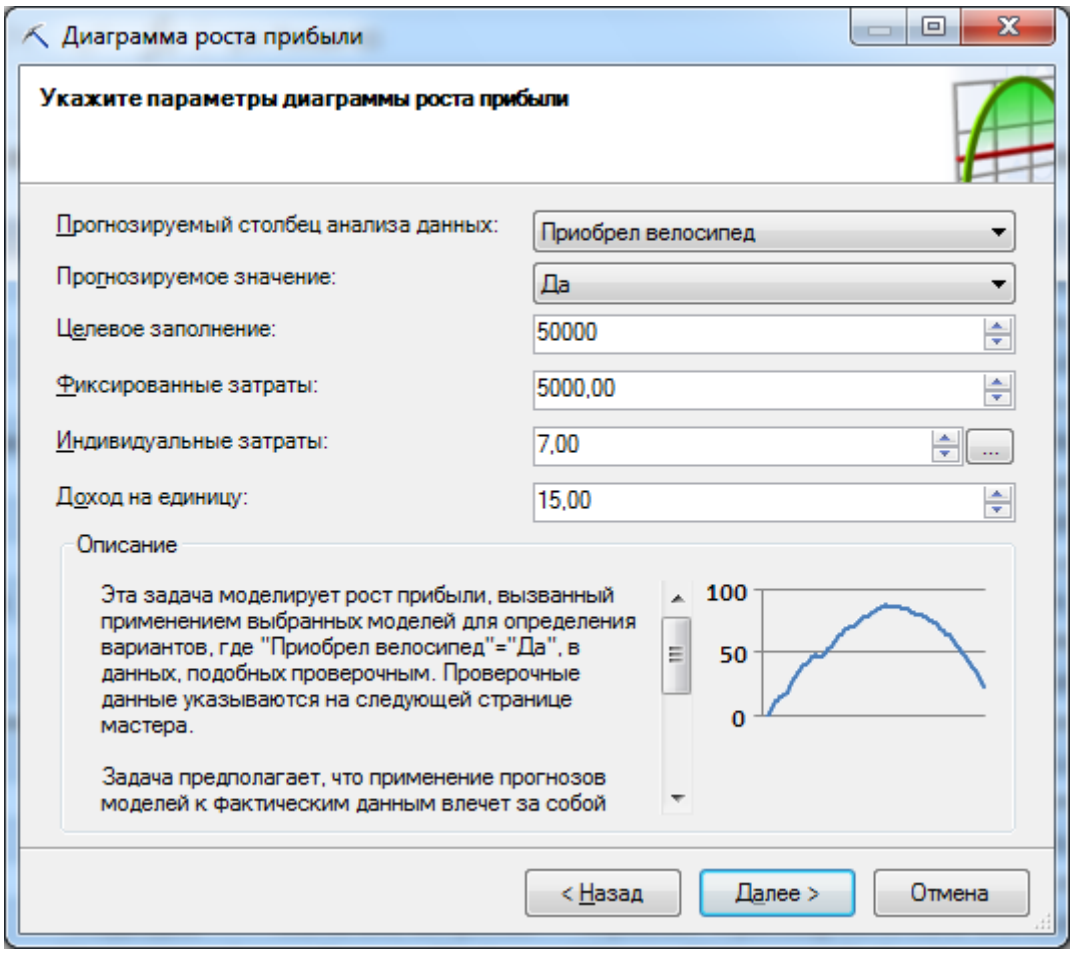

Рис. 69. Параметры расчета роста прибыли

Данная команда последовательно рассчитывает прибыть для 1 %, 2 %, … наиболее перспективных клиентов, строит соответствующий график и находит максимум. В результате получается таблица с максимальными значениями прибыли (табл. 5).

Таблица 8

Максимальные значения прибыли

Показатель	Алгоритм Байеса	Алгоритм нейронной сети	Дерево решений	Логистическая регрессия
Максимальная прибыль	71 500	65 500	60 000	74 500
Порог вероятности, %	46,25	43,54	50,00	37,96

Кроме этого, строится диаграмма роста прибыли (Рис. 70), наглядно демонстрирующая сначала рост прибыли за счет увеличения количества перспективных клиентов, затем уменьшения прибыли за счет увеличения затрат на клиентов, которые не принесут доход.

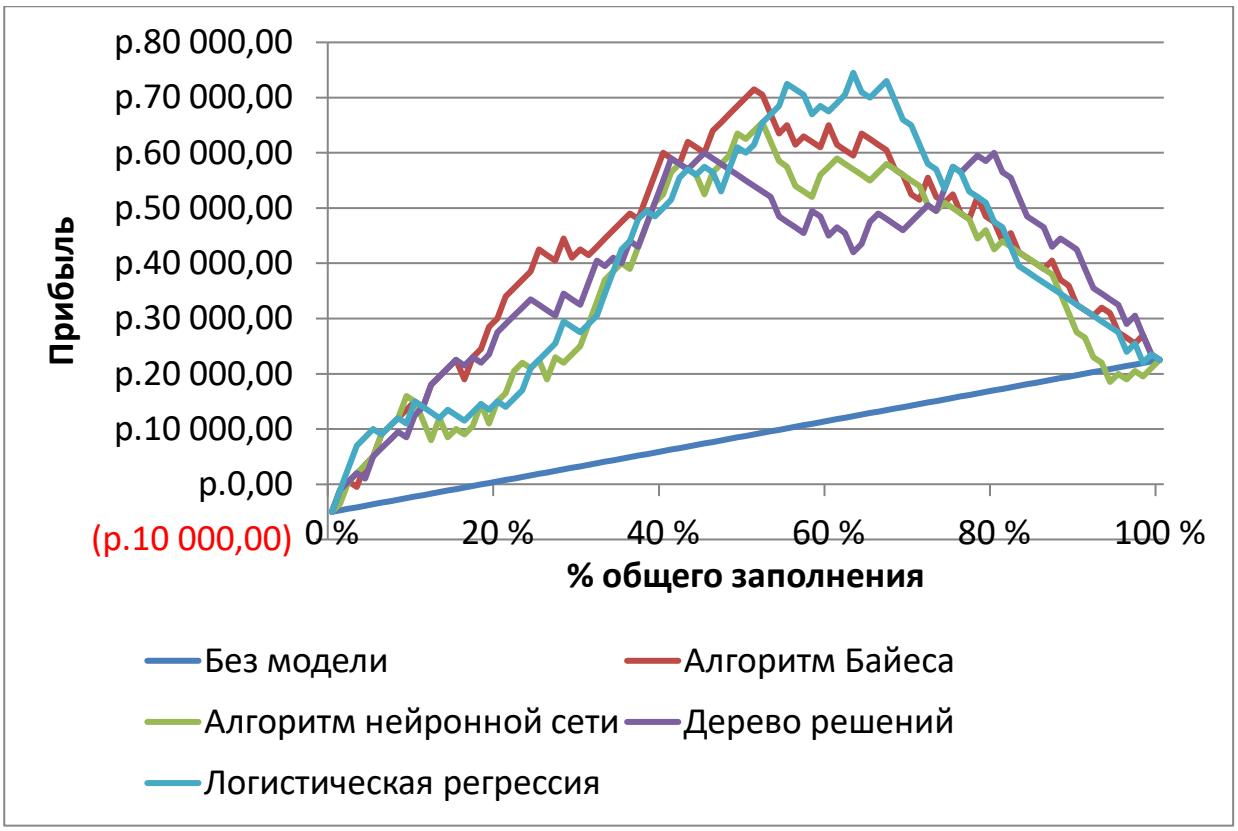

Рис. 70. Диаграмма роста прибыли

3.3.8. Применение моделей классификации

Настроенная модель может быть применена для решения стандартных задач. Для задачи классификации — это вычисление класса. Класс вычисляется по максимальной вероятности принадлежности объекта. Это решение может быть достаточно неопределенным. Поэтому, кроме класса, важна вероятность принадлежности этому классу. В надстройке Excel для применения моделей — вычисления параметров объектов с помощью построенной модели предусмотрена команда «Применение модели» / «Запрос» в меню «Интеллектуальный анализ данных».

Мастер определения параметров запроса требует сначала выбрать модель, затем задать для нее исходные данные в виде таблицы (например, таблицу с характеристиками новых клиентов). Для каждой строки таблицы (описания клиента) нужно установить соответствие атрибутов модели и таблицы (Рис. 71).

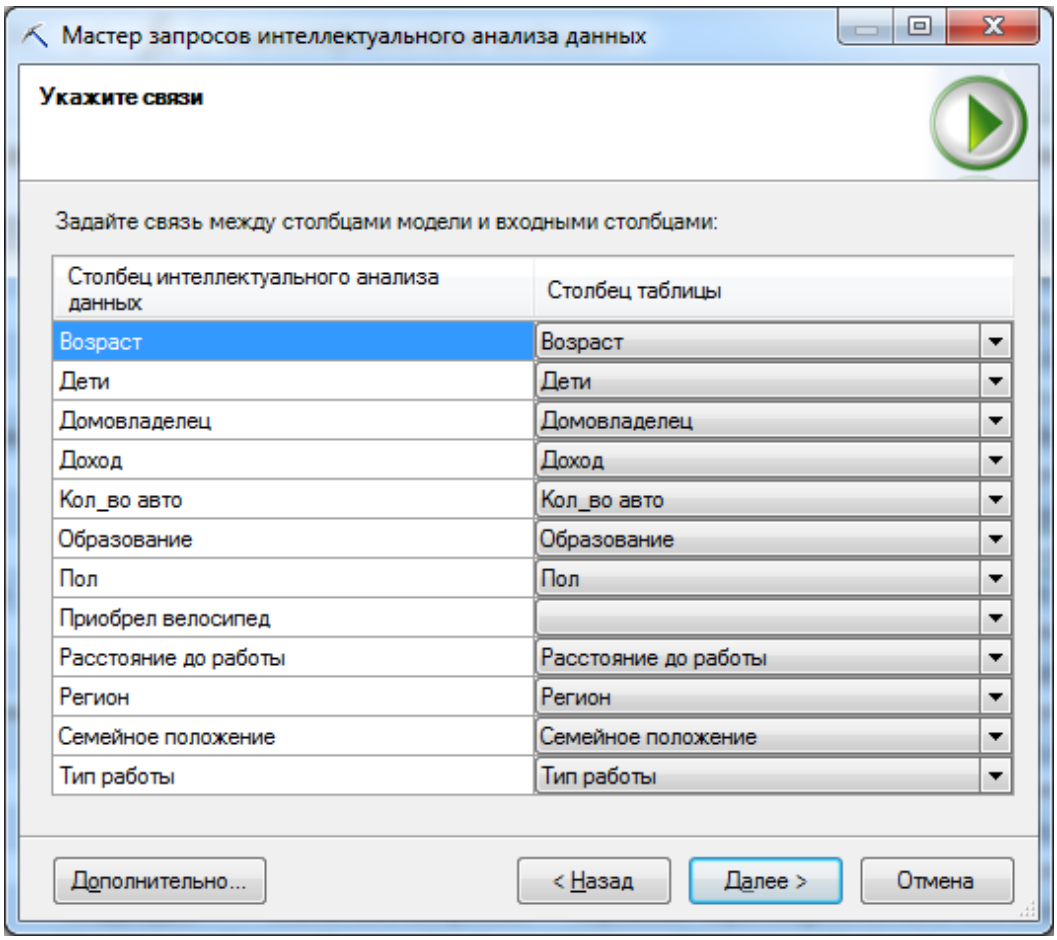

Рис. 71. Задание соответствия атрибутов модели и данных

Далее мастер предлагает выбрать выходные данные модели (Рис. 72). Кнопка «Добавить выходные данные» открывает окно (Рис. 73), в котором можно выбирать выходные данные модели из следующих: Predict — прогноз класса, Predict Probability — вероятность принадлежности объекта классу, Predict Support — поддержка (количество объектов с аналогичными характеристиками, принадлежащих прогнозируемому классу).

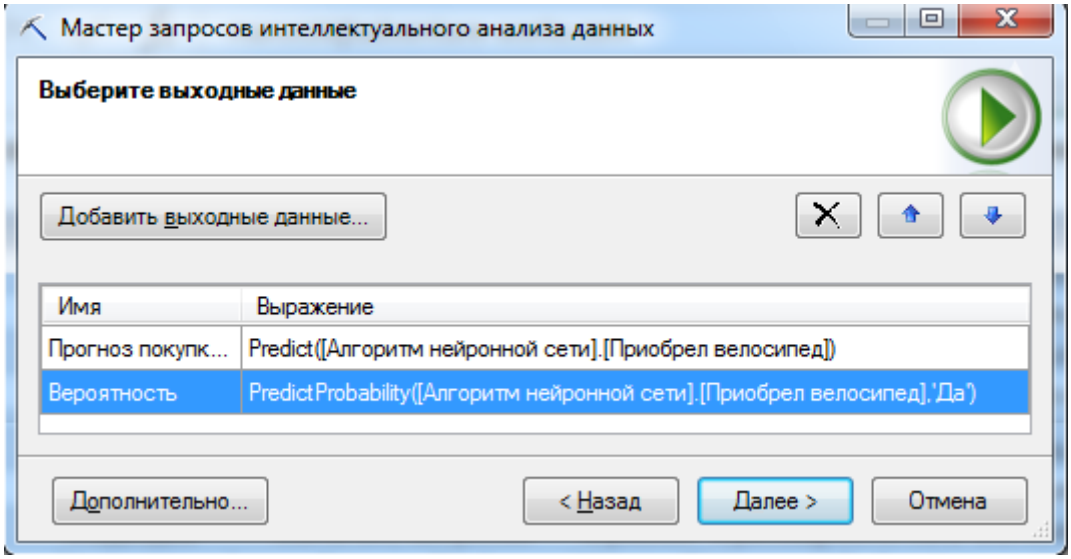

Рис. 72. Определение выходных данных модели классификации

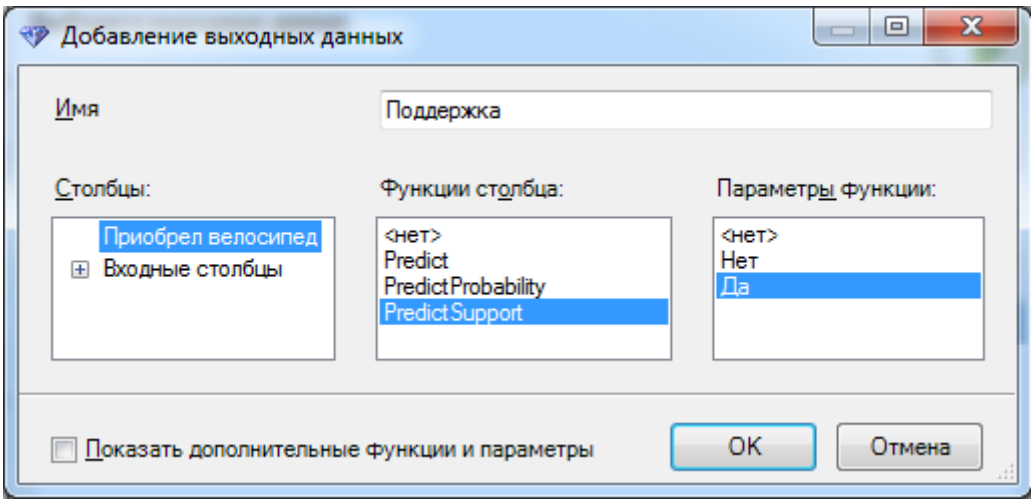

Рис. 73. Выбор возможных выходных данных модели классификации из списка

В результате исходные данные будут дополнены определенными выходными данными модели (табл. 5).

Таблица 9

Фрагмент таблицы, дополненной столбцами с модельными данными

Доход	Дети	Количество авто	Регион	Возраст	Прогноз покупки велосипеда	Вероятность	Поддержка
90000	0	0	Россия	40	Да	0,81	399
60000	3	1	Россия	41	Да	0,83	409
60000	3	1	Россия	41	Да	0,78	381
70000	0	1	Россия	38	Да	0,79	388
80000	5	4	Россия	38	Нет	0,41	201
70000	0	1	Россия	41	Да	0,79	386

Выходные данные модели можно использовать для прогнозирования класса, вероятностных расчетов доходности клиента, определения точки отсечения, прогноза среднего суммарного дохода и решения других задач.

3.4. Регрессионные модели

Модель регрессии заключается в подборе вектора a параметров так, чтобы функция $y = f(x,a)$ аппроксимировала зависимость y от x наилучшим образом. Для подбора параметров обычно применяют метод наименьших квадратов, т.е. находят параметры, минимизирующие сумму квадратов ошибок

$$a^* = \arg\min \sum_{i=1}^{m} \left(f(x_i, a) - y_i \right)^2$$

Семейство параметрических кривых $y = f(x,a)$ определяют, исследуя облако рассеивания. Если оно вытянуто вдоль прямой, то применяют линейные уравнения регрессии. В этом случае коэффициент корреляции характеризует

рассеивание. Чем он ближе к 1 или –1, тем меньше разброс и тем больше точность приближения зависимости линейной функцией. Нулевое значение коэффициента корреляции свидетельствует либо о независимости функции от значений аргументов, либо о нелинейном характере зависимости.

Один из вариантов параметрических кривых — это ломаная линия (Рис. *74*). В аналитических службах MS SQL сервера реализован именно этот вариант. Для определения точек перелома применяется алгоритм ветвления дерева решений, а на полученных отрезках — алгоритм линейной регрессии. Для построения модели в MS Excel выбирается команда «Оценка» в меню «Интеллектуальный анализ данных». Далее выбираются анализируемый столбец и входные столбцы. Можно задать методы ветвления и параметры алгоритма дерева решений.

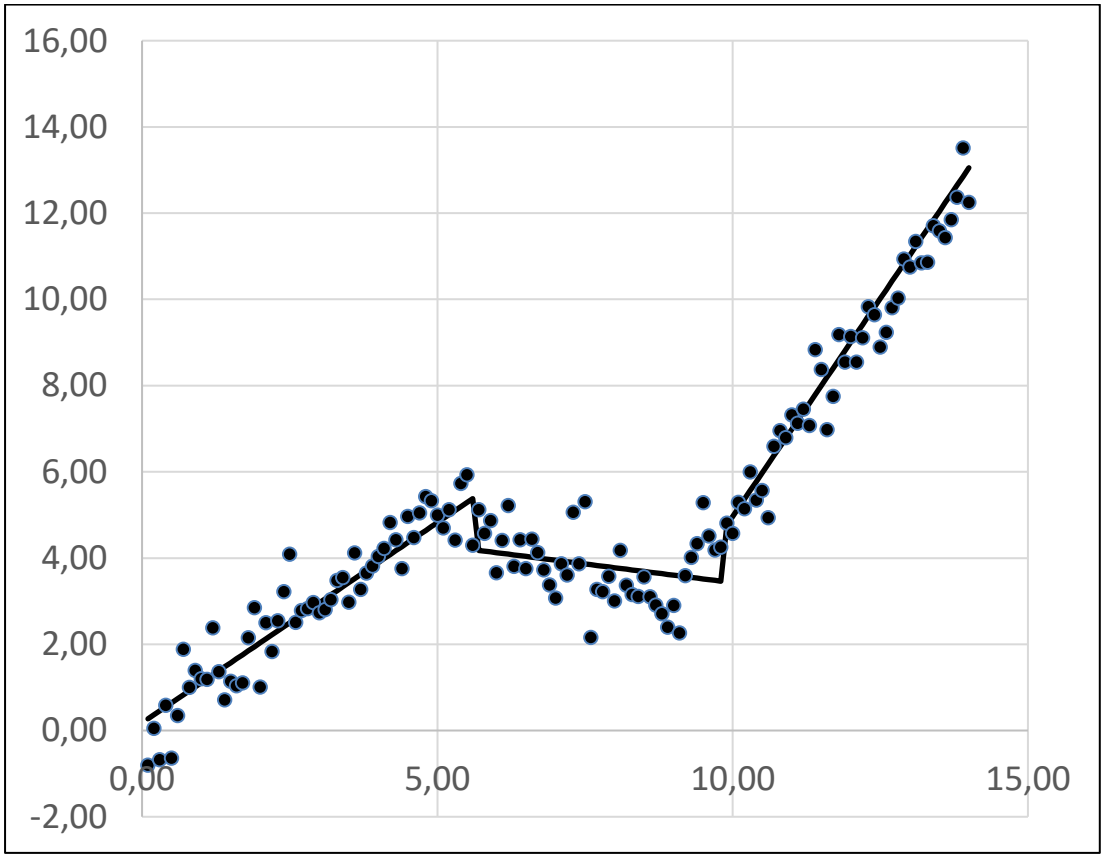

Рис. 74. Исходные данные и ломанная линия регрессии

В результате получается дерево решений (Рис. 75), каждый узел которого определяет формулу линейной регрессии. Применение модели (команда «Применение модели» / «Запрос» в меню «Интеллектуальный анализ данных») позволяет вычислить значение функции регрессии. На Рис. 74 представлены исходные данные и вычисленная регрессионная функция (сплошная линия). Скачки функции — это соединение линейных функций регрессии на разных участках.

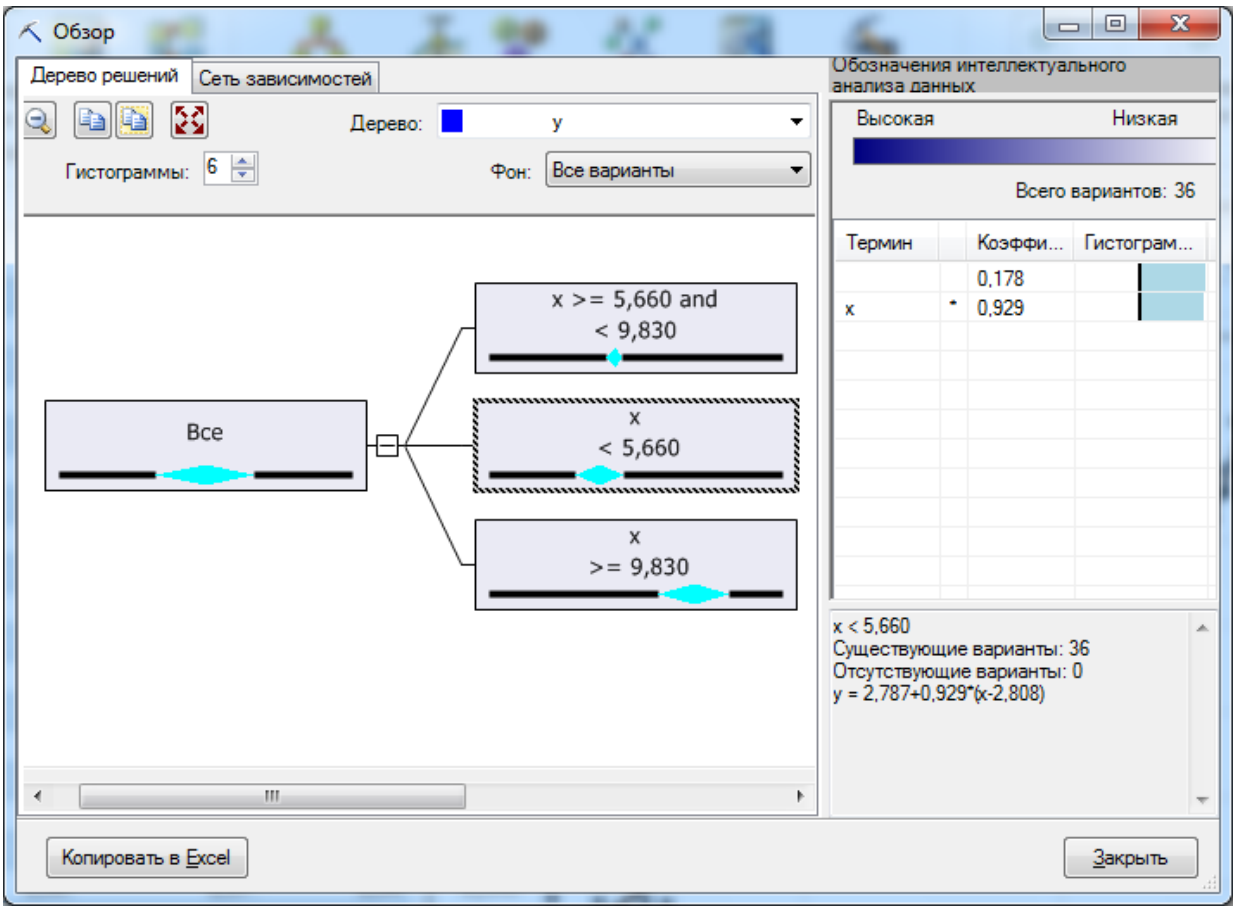

Рис. 75. Дерево решений построения кусочно-линейной регрессии

3.5. Задачи кластеризации

Кластеризация заключается в делении множества объектов, схожих по параметрам, на группы (кластеры). При этом, в отличие от классификации, число кластеров и их характеристики могут быть заранее неизвестны и определяться в ходе построения кластеров исходя из степени близости объединяемых объектов по совокупности параметров. Кроме этого, выбор меры «похожести» или близости свойств объектов между собой, как и критерия качества кластеризации, часто носит субъективный характер. Рассмотрим объекты, группируемые по двум атрибутам: возраст и зарплата. Разность возрастов будет измеряться единицами и десятками, а зарплат — тысячами и десятками тысяч. При этом «расстояние» между объектами будет определяться, в основном, зарплатами. Поэтому очень часто от исходных атрибутов переходят к нормируемым — пересчитанным к определенному интервалу, хотя и это не гарантирует объективности кластеризации.

С помощью кластеризации решают следующие задачи:

– Изучение данных. Разбиение множества объектов на группы помогает выявить внутренние закономерности, понять, насколько информативны свойства объектов.

– Облегчение анализа. При помощи кластеризации можно упростить дальнейшую обработку данных и построение моделей: каждый кластер обрабатывается индивидуально, и модель создается для каждого кластера в отдельности. В этом смысле кластеризация может рассматриваться как подготовительный этап.

– Сжатие данных. Кластеризация позволяет сократить объем хранимых данных, оставив по одному наиболее типичному представителю от каждого кластера.

– Прогнозирование. Поведение объекта можно прогнозировать на основе сходства с поведением других объектов кластера.

– Обнаружение аномалий. Кластеризация применяется для выделения нетипичных объектов. Эту задачу также называют обнаружением аномалий (outlier detection). Интерес здесь представляют кластеры (группы), в которые попадает крайне мало объектов (один-три).

Алгоритм К-средних (англ. k-means) реализует, так называемую, жесткую кластеризацию. Это значит, что вариант может принадлежать только одному кластеру. Идея алгоритма заключается в следующем:

1. Выбирается число кластеров *k*.

2. Из исходного множества данных случайным образом выбираются *k* записей, которые будут служить начальными *центрами кластеров*.

3. Для каждой записи исходной выборки определяется ближайший к ней *центр кластера*. При этом записи, «притянутые» определенным центром, образуют начальные кластеры.

4. Вычисляются центроиды — центры тяжести кластеров. Каждый центроид — это вектор, элементы которого представляют собой средние значения признаков, вычисленные по всем записям кластера. Затем *центр кластера* смещается в его центроид.

Шаги 3 и 4 итеративно повторяются, при этом может происходить изменение границ кластеров и смещение их центров. В результате минимизируется расстояние между элементами внутри кластеров. Остановка алгоритма производится тогда, когда границы кластеров и расположения центроидов не перестанут изменяться от итерации к итерации, т.е. на каждой итерации в каждом кластере будет оставаться один и тот же набор записей. Недостаток алгоритма заключается в том, что количество кластеров не определяется автоматически.

Алгоритм G-means. Распределение точек в кластере предполагается унимодальным и Гауссовским. Для каждого кластера определяются моды точки сгущения, если такая точка одна — распределение считают унимодальным, в противном случае — кластер разбивают на несколько.

Максимизация ожиданий (англ. Expectation-maximization, EM) относится к методам мягкой кластеризации, т.е. объект в этом случае может принадлежать нескольким кластерам и для всех возможных сочетаний вариантов с кластерами вычисляются вероятности.

При кластеризации методом EM алгоритм итеративно уточняет начальную модель кластеризации, подгоняя ее к данным, и определяет вероятность принад-

лежности точки данных кластеру. Этот алгоритм заканчивает работу, когда вероятностная модель соответствует данным. Функция, используемая для установления соответствия, — логарифм функции правдоподобия данных, вводимых в модель.

Если в процессе формируются пустые кластеры или количество элементов в одном или нескольких кластерах оказывается меньше заданного минимального значения, малочисленные кластеры заполняются повторно с помощью новых точек и алгоритм EM запускается снова. Результаты метода масштабируемой максимизации ожидания являются вероятностными: каждая точка данных принадлежит всем кластерам, но с разной вероятностью. Поскольку метод допускает перекрытие кластеров, сумма элементов всех кластеров может превышать число элементов обучающего набора.

В процессе применения кластеризации необходимо решать следующие проблемы:

1. Проблема выбора меры близости, которая обусловлена различной природой данных:

– разные меры расстояний:

• для номинальных шкал (неупорядоченные значения) расстояние равно 0 при совпадении признаков и 1 — при несовпадении;

• для порядковых шкал расстояние равно разнице порядковых номеров в шкале;

– метрические шкалы обладают разными единицами измерения;

– сложно свести разные метрики к одной. Один из приемов — нормирование каждого расстояния к одному интервалу. Все признаки будут одинаково значимыми.

2. Проблема выбора количества кластеров. Мало кластеров — грубая классификация. Много — возможная потеря однозначности разбиения на кластеры.

В литературе [1] приводятся следующие рекомендации по применению кластеризации:

– определить цель кластеризации:

• изучение данных;

• облегчение анализа;

• сжатие данных;

• прогнозирование;

• обнаружение аномалий;

– выбрать меру сходства и алгоритм;

– провести содержательный анализ результатов кластеризации.

Для выполнения кластеризации в надстройке интеллектуального анализа MS Excel выбирается команда «Кластер» в меню «Интеллектуальный анализ данных». С помощью мастера определения параметров модели указываются исходная таблица с данными, задается признак автоматического определения количества кластеров (сегментов) или количество кластеров, указываются входные столбцы. Кнопка «Параметры» открывает соответствующее окно (Рис. 76), в котором можно выбрать параметр кластеризации.

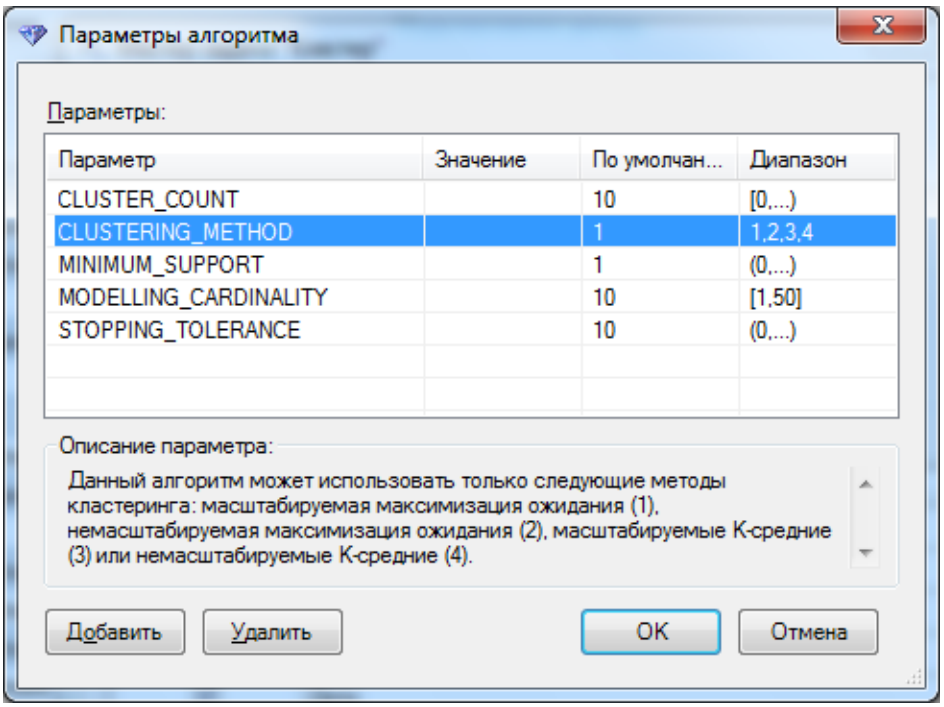

Рис. 76. Окно параметров модели кластеризации

В результате настройки модели определяется оптимальное количество кластеров, определяются центры кластеров. Обзор модели включает диаграмму кластеров (Рис. 77), на которой изображены кластеры и линии, характеризующие близость кластеров.

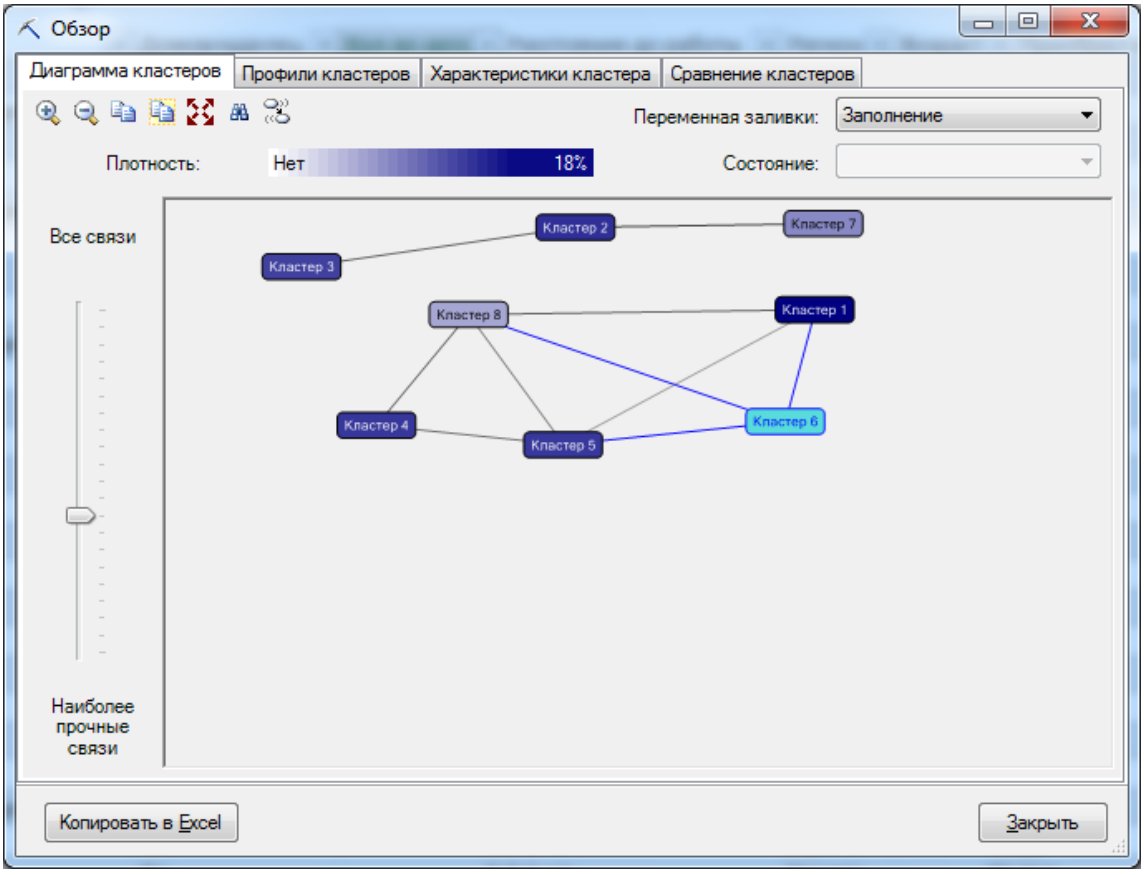

Рис. 77. Диаграмма кластеров

На вкладке профили кластеров (Рис. 78) можно понять по каким параметрам кластеры отличаются друг от друга. Например, для первого кластера характерны значения типа работы «Профессионал» и «Управление» с примерно одинаковыми долями.

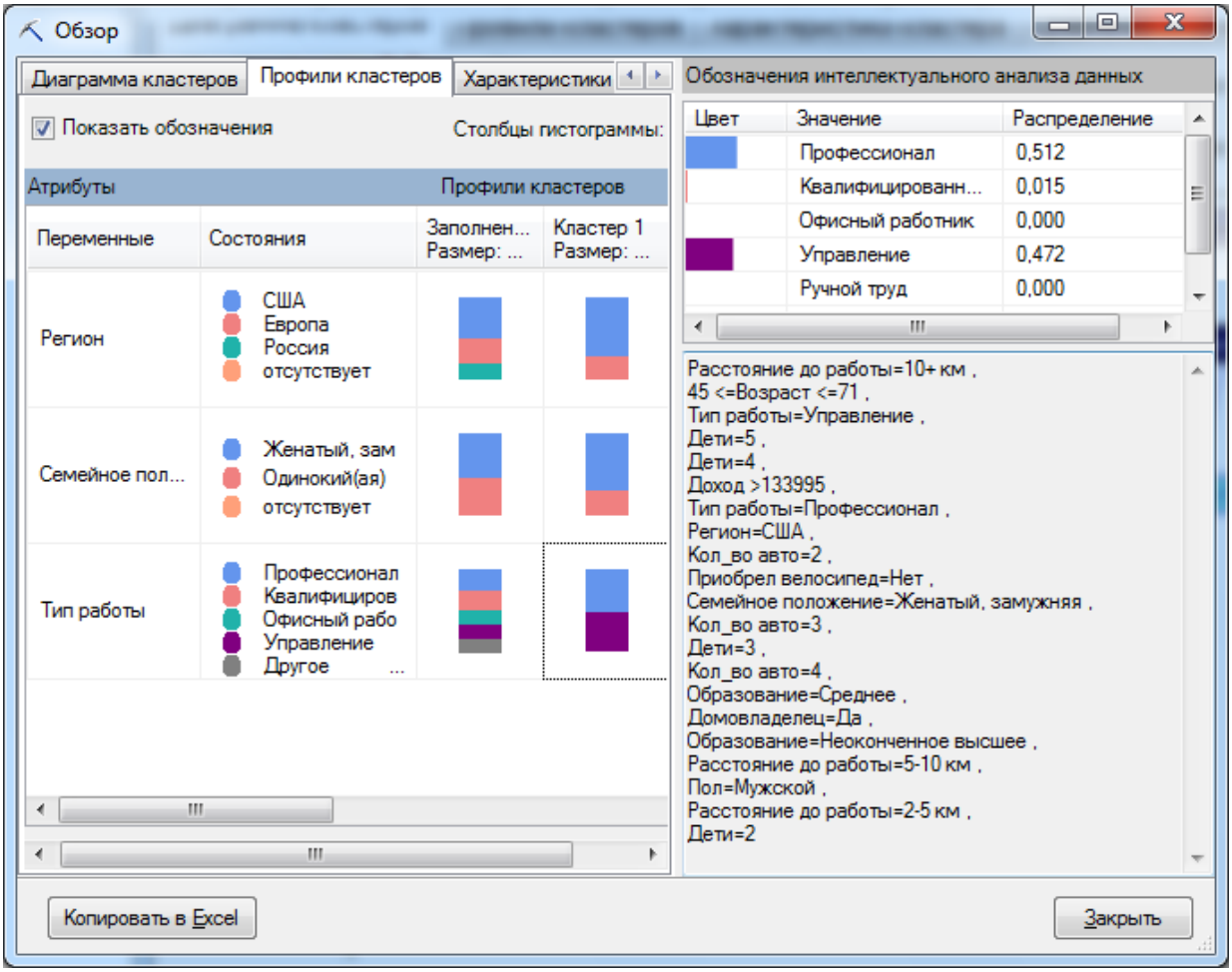

Рис. 78. Профили кластеров

На вкладке «Характеристики кластеров» (Рис. 79) приведены вероятности значений атрибутов для выбранного кластера. Можно найти кластер, для которого вероятность покупки велосипедов будет максимальна и использовать это обстоятельство в рекламной кампании.

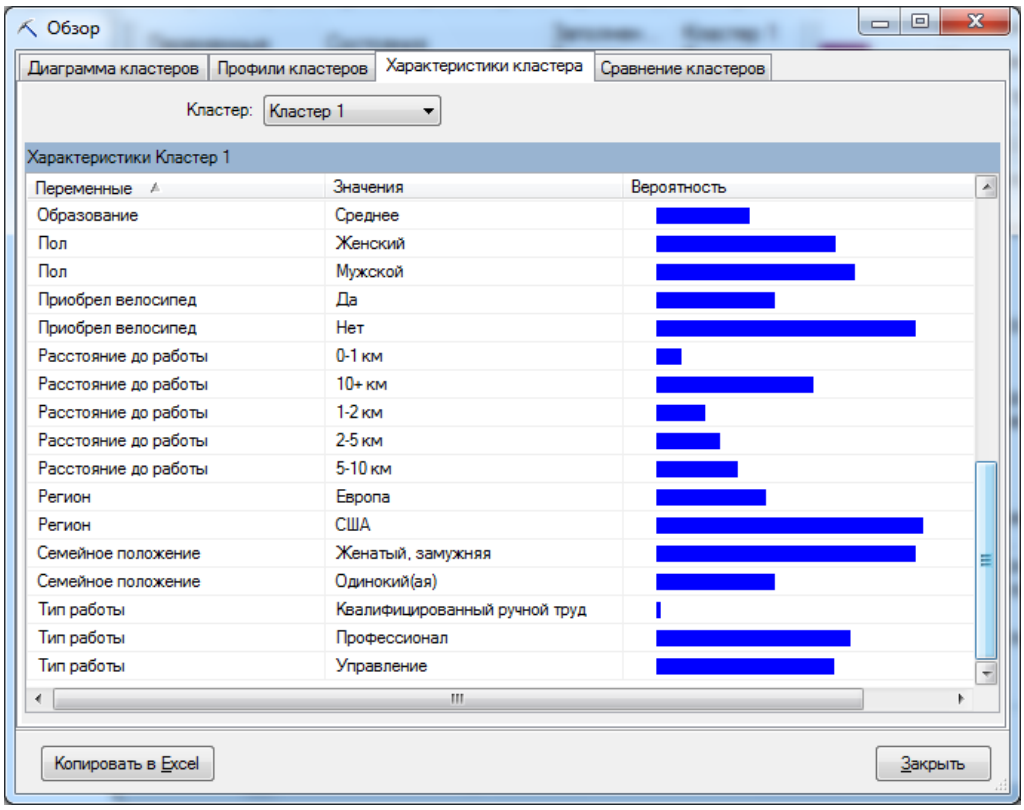

Рис. 79. Характеристики кластеров

На вкладке «Сравнение кластеров» (Рис. 80) можно выбирать и сравнивать кластеры. На Рис. 80 сравнивается кластер 2, для объектов которого вероятность купить велосипед максимальна и кластер 3 с минимальной вероятностью покупки велосипеда. Такое сравнение позволяет лучше понять влияние значений атрибутов на поведение покупателей.

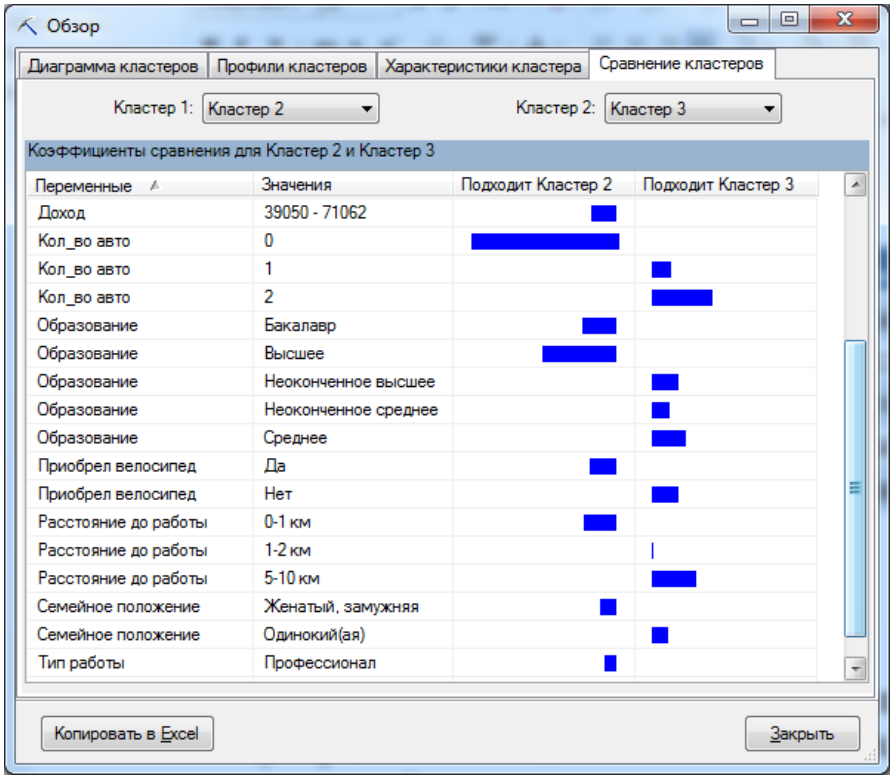

Рис. 80. Сравнение кластеров

Запрос к модели позволяет вычислять расстояние объекта до центра каждого кластера и вероятность принадлежности объекта каждому кластеру. В табл. 6 приведены клиенты, отсортированные по убыванию вероятности принадлежности первому кластеру, затем по убыванию вероятности принадлежности второму кластеру. Приведен фрагмент завершения клиентов первого кластера и начало списка второго. Таблица демонстрирует, что для последних по вероятности клиентов первого кластера вероятность принадлежности относительно невелика и выбор кластера не является достаточно определенным. На этом основано выделение исключений — выбор объектов, которых нельзя достоверно отнести к одному кластеру.

Таблица 10

Расстояния объектов до кластеров 1 и 2 и вероятности принадлежности

ID объекта	Категория	Расстояние 1	Вероятность 1	Расстояние 2	Вероятность 2
22174	Категория 1	0,4337	0,5663	1,0000	0,0000
22173	Категория 1	0,4822	0,5178	1,0000	0,0000
13981	Категория 1	0,6259	**0,3741**	1,0000	0,0000
14872	Категория 2	0,9392	0,0608	0,0610	0,9390
25598	Категория 2	1,0000	0,0000	0,0000	1,0000

В надстройке Excel в меню «Анализировать» реализована команда «Выделение исключений». В результате создается страница (Рис. 81), на которой можно установить порог (долю надежных данных). Данные, попавшие в исключения с заданным порогом, выделяются в исходной таблице цветом. Дополнительно выделяется исключительное значение атрибута.

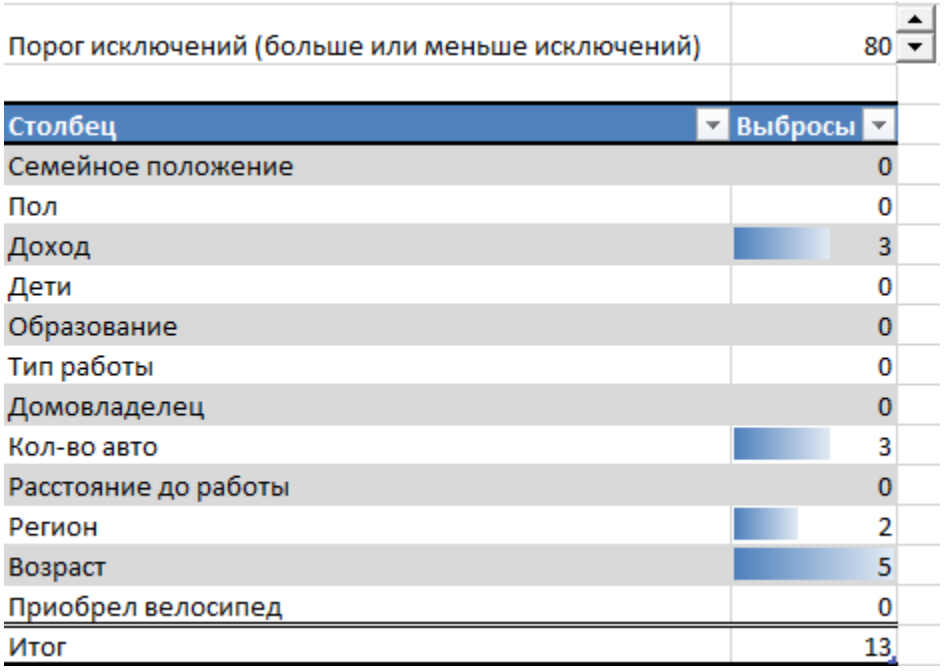

Рис. 81. Страница выделения исключений

3.6. Анализ временных рядов

Исходными данными для анализа временных рядов являются упорядоченные данные. В простейшем случае это может быть один ряд y1,…,yt,… наблюдений некоторой величины в разные моменты времени t. Для этих данных можно решать следующие задачи:

– Описание характеристик и закономерностей ряда.

– Моделирование — построение модели исследуемого процесса.

– Прогнозирование — предсказание будущих значений временного ряда.

– Управление. Зная свойства временных рядов, можно выработать методы воздействия на соответствующие бизнес-процессы для управления ими.

Формальные методы прогнозирования основаны на предположении, что все наблюдения сформировались под действием одних и тех же факторов, которые сохранятся такими же в прогнозируемом периоде. Ясно, что это перестает быть справедливым при изменении правил и нормативов, применяемых в экономике. В таких ситуациях применяют экспертные суждения. Объединение двух методик приводит к комплексному методу прогнозирования.

Один из подходов к изучению временного ряда заключается в разделении его на случайную и детерминированную составляющие. Чаще всего используют две такие модели:

$y_t = d_t + \varepsilon_t$ — аддитивная модель,

$y_t = d_t \times \varepsilon_t$ — мультипликативная модель.

Под случайной составляющей подразумевают стационарный случайный ряд, все компоненты которого распределены по одинаковому вероятностному закону. В самом простом случае случайную составляющую образуют независимые и одинаково распределенные случайные величины. В этом случае в аддитивной модели случайная составляющая интерпретируется как ошибка измерения. Независимые случайные величины образуют последовательность, в которой величины изменяются скачками. Если наблюдаются плавные изменения, то необходимо применять для случайной составляющей более сложные стационарные модели. Стационарные случайные последовательности определяют для каждого элемента последовательности одинаковое распределение вероятности, однако, между элементами может существовать статистическая зависимость, определяющая плавность изменения значений.

Детерминированную составляющую делят на тренд, сезонную составляющую и циклическую составляющую

$$d_t = r_t + s_t + c_t.$$

Тренд r_t — медленно меняющаяся составляющая ряда, которая описывает влияние на временной ряд долговременно действующих факторов, вызывающих плавные и длительные изменения ряда.

Сезонная составляющая s_t временного ряда описывает изменения его значений в пределах некоторого периода и представляет собой последовательность повторяющихся циклов одинаковой протяженности.

Циклическая составляющая c_t временного ряда состоит из интервалов подъема и спада, которые имеют различную протяженность, а также различную амплитуду расположенных в них значений.

Наличие в рядах данных циклических компонент связано с тем, что в пределах интервалов более глобальных изменений (чем, например, сезонных) могут наблюдаться не имеющие периодичности временные подъемы и спады, которые, в отличие от случайной компоненты, не вызваны действием случайных факторов, а являются особенностями бизнеса и обусловлены общеэкономической ситуацией.

Главной проблемой при построении моделей временных рядов является нестационарность ряда. Временной ряд называется стационарным, если его статистические свойства (математическое ожидание, дисперсия, зависимости элементов ряда) одинаковы на всем протяжении ряда. В противном случае ряд называется нестационарным. Применение к нестационарным рядам различных методов анализа, в том числе статистических, затруднено. Поэтому, прежде чем приступать к построению модели ряда, его стараются свести к стационарному.

Одним из инструментов анализа временных рядов является автокорреляционная функция — коэффициент корреляции ***r(t)*** значений ряда, смещенных на ***t*** относительно друг друга. Автокорреляционная функция стационарного ряда затухает при $t \to \infty$. На Рис. 82 ***r1(t)*** является автокорреляционной функцией стационарного ряда, а ***r2(t)*** — нестационарного ряда.

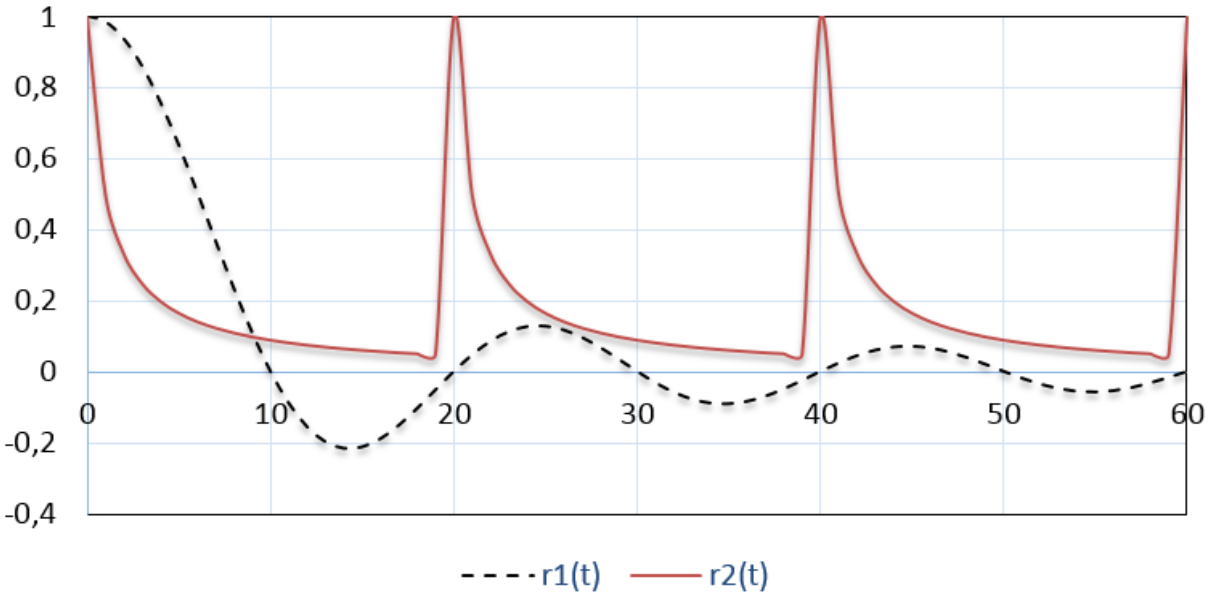

Рис. 82. Автокорреляционные функции стационарного ***r1*** и нестационарного ***r2*** рядов

Модель авторегрессии появилась как обобщение правила, в котором новое значение вычислялось как предыдущее, увеличенное на некоторую случайную величину ε_t. В общем случае учитывается ***p*** предыдущих значений с соответствующими весовыми коэффициентами

$$y_t = a_1 y_{t-1} + \ldots a_p y_{t-p} + \varepsilon_t.$$

Такую модель авторегрессии порядка ***p*** обозначают как ***AR(p)***.

Модель скользящего среднего является обобщением идеи сглаживания значений ряда усреднением q подряд идущих значений. В результате такого сглаживания случайные отклонения усредняются и полученный ряд точнее описывает неслучайные закономерности. Модель скользящего среднего ***MA(q)*** описывается уравнением

$$y_t = \varepsilon_t - b_1 \varepsilon_{t-1} - \ldots - b_q \varepsilon_{t-q}.$$

Объединение этих моделей называют моделью авторегрессии и скользящего среднего ***ARMA (p,q)***

$$y_t = a_1 y_{t-1} + \ldots a_p y_{t-p} + \varepsilon_t - b_1 \varepsilon_{t-1} - \ldots - b_q \varepsilon_{t-q}.$$

Эта модель при соблюдении определенных ограничений на коэффициенты описывает стационарные случайные процессы.

Для приведения нестационарного процесса к стационарному применяют вычисление разности Δ^d порядка d

$$\nabla^1(y_t) = y_t - y_{t-1},$$
$$\nabla^2(y_t) = \nabla^1\left(\nabla^1(y_t)\right) = y_t - 2y_{t-1} + y_{t-2},$$
$$\ldots$$

до тех пор, пока не получат стационарный процесс, который моделируют с помощью модели ARMA (p, q). Полученная модель получила название модель авторегрессии и интегрированного скользящего среднего **ARIMA(*p, d, q*)** — autoregressive integrated moving average.

Для коротких рядов наблюдений применяют дерево авторегрессии с перекрестным прогнозированием (autoregressive tree with cross prediction — ARTXP). Дерево решений делит весь временной интервал на отрезки и на каждом отрезке применяется модель авторегрессии.

По умолчанию в алгоритме временных рядов аналитических служб используется комбинация алгоритмов для анализа закономерностей и подготовки прогнозов. Алгоритм обучает две отдельные модели одних и тех же данных: в одной модели используется алгоритм ARTXP, а в другой — алгоритм ARIMA. Затем алгоритм объединяет результаты обеих моделей, чтобы сформировать наилучший прогноз для переменного числа временных срезов. Алгоритм ARTXP лучше подходит для краткосрочных прогнозов, поэтому, начиная ряд прогнозов, на него следует полагаться в большей степени. Но по мере того, как временные срезы, применяемые для прогнозирования, уходят все дальше в будущее, становится все более полезным алгоритм ARIMA.

Рассмотрим построение прогноза на примере (табл. 8), который содержит данные об объемах продаж по месяцам. Для выполнения прогнозирования в надстройке интеллектуального анализа MS Excel выбирается команда «Прогноз» в меню «Интеллектуальный анализ данных». В настройках модели нужно определить исходные данные: ряд значений времени и прогнозируемые ряды (Рис. 83). Можно также задать дополнительные параметры модели (кнопка «Параметры»), в частности выбрать алгоритм ARIMA, ARTXP или наилучшую из них.

Таблица 11

История продаж модели M200 для различных регионов

Год и месяц	Европа, р.	США, р.	Россия, р.
01.07.2001	20 324,94	20 324,94	64 424,81
01.08.2001	20 349,94	23 724,93	60 899,82
01.09.2001	16 949,95	16 974,95	10 174,97

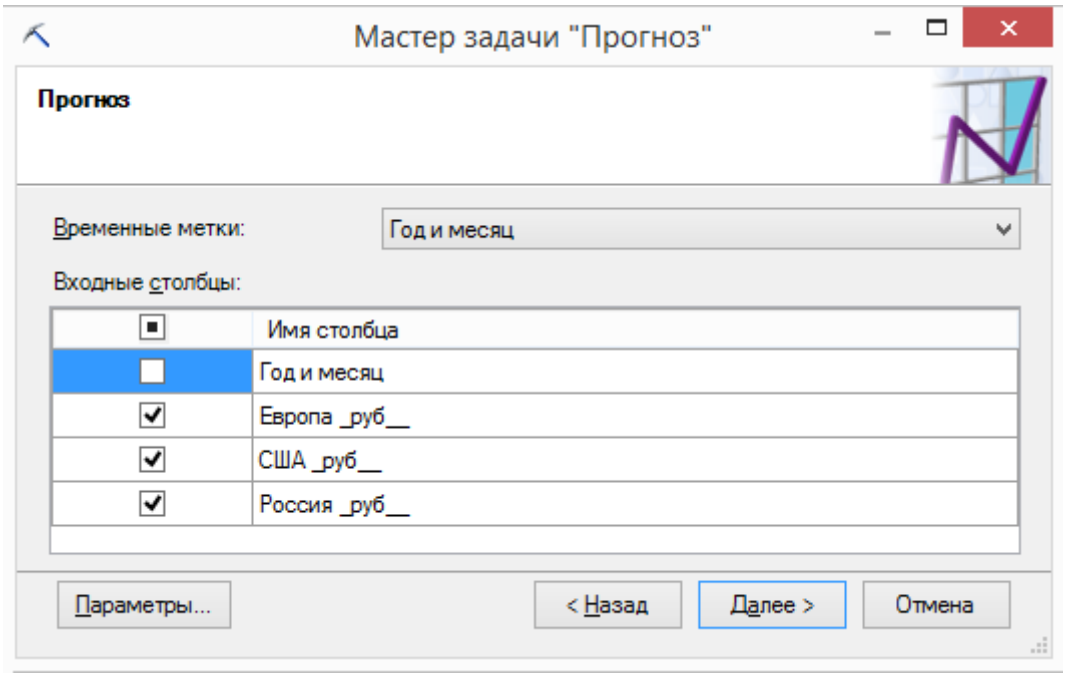

Рис. 83. Указание исходных данных для прогнозирования

После настройки модели открывается окно обзора модели. На вкладке «Диаграммы» (Рис. 84) демонстрируются графики прогнозируемых рядов, пунктиром показаны прогнозируемые значения. При включенном переключателе «Отображать отклонения» на графике демонстрируются интервалы возможных отклонений, характеризующие точность прогнозирования.

На вкладке «Модель» (Рис. 85) приводится описание модели в виде дерева решений и набора коэффициентов прогнозирующей формулы.

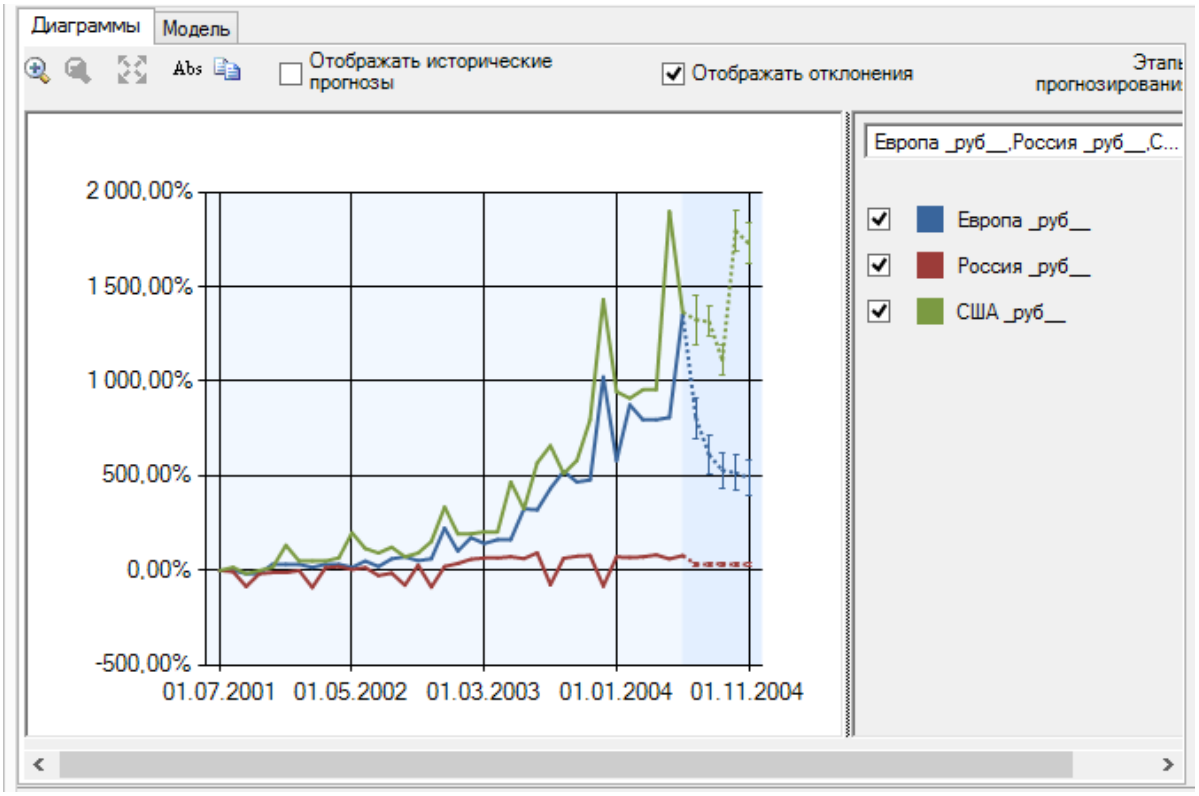

Рис. 84. Графики прогнозируемых рядов

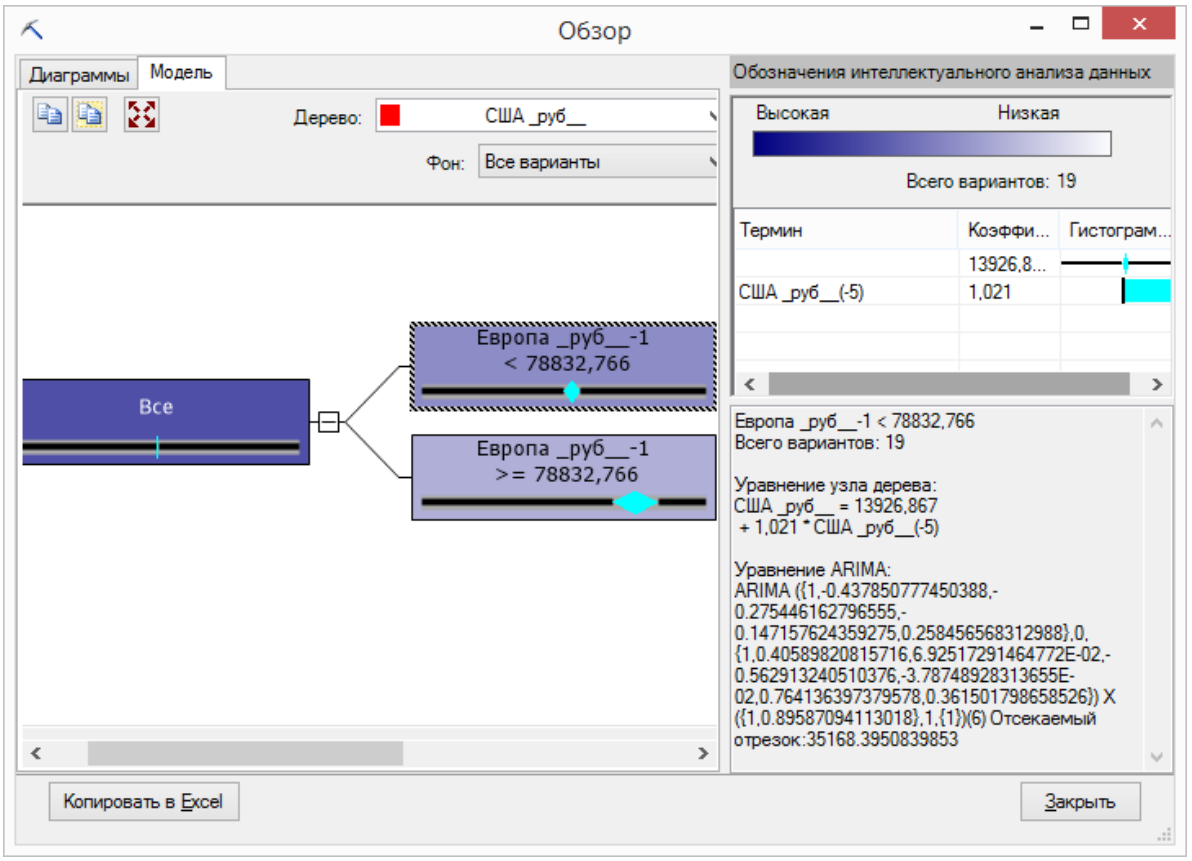

Рис. 85. Описание модели прогнозирования

3.7. Алгоритм взаимосвязей

Алгоритм взаимосвязей или ассоциативных правил (AssociationRules) позволяет выявить события, которые часто происходят совместно. Базовым понятием в теории ассоциативных правил является транзакция — некоторое множество событий, происходящих совместно. Типичная транзакция — приобретение клиентом товара в супермаркете (см. табл. 9), когда проводится поиск товаров, наиболее часто встречающихся в одном заказе (чеке, транзакции), после чего, на основе выявленных закономерностей, становится возможной выдача рекомендаций.

Таблица 12

Исходные данные алгоритма взаимосвязей — заказы в магазине

Номер заказа	Категория	Товар	Цена
SO61276	Трикотажные изделия	Куртка с короткими рукавами	539,99
SO61276	Кепки	Кепка велосипедная	8,99
SO61277	Горные велосипеды	Mountain-500	539,99
SO61277	Трикотажные изделия	Куртка с короткими рукавами	539,99
SO61277	Кепки	Кепка велосипедная	8,99
SO61278	Дорожные велосипеды	Road-350-W	2443,4
SO61278	Бутылки и крепления	Крепление для бутылки (дорожн. вел.)	8,99
SO61278	Бутылки и крепления	Бутылка для воды	4,99
SO61278	Трикотажные изделия	Куртка с короткими рукавами	539,99

Следующее важное понятие — предметный набор. Это непустое множество объектов, появившихся в одной транзакции. Примерами таких наборов для данных из табл. 8 будут следующие {Трикотажные изделия, Кепки}, {Горные велосипеды, Трикотажные изделия, Кепки}, {Дорожные велосипеды, Бутылки и крепления, Бутылки и крепления, Трикотажные изделия}.

Поиск ассоциаций — определение частоты (вероятности) предметного набора. Для практического применения нужны предметные наборы, имеющие значительную частоту.

Ассоциативное правило состоит из двух наборов предметов, называемых условием (antecedent) и следствием (consequent), записываемых в виде X → Y, что читается так: «Из X следует Y», Таким образом, ассоциативное правило формулируется в виде: «Если *условие*, то *следствие*». Фрагмент списка правил представлен на Рис. 86. Первое правило утверждает, что если покупатель приобрел велосипедные подставки, то с вероятностью 0,787 он купит шины и камеры.

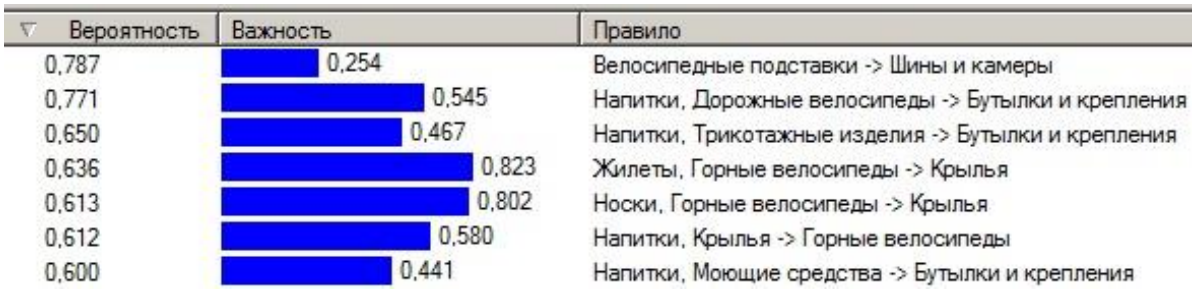

Рис. 86. Ассоциативные правила

Ассоциативные правила описывают связь между наборами предметов (Рис. 87), соответствующими условию (antecedent) и следствию (consequent). Эта связь характеризуется двумя показателями:

– ***S*** — поддержкой (support)

$$S(A \rightarrow B) = P\{A \cap B\} = Q(A \cap B) / Q(\Omega),$$

где $Q(A \cap B)$ — количество транзакций, содержащих A и B;

$Q(\Omega)$ — общее количество транзакций,

– ***C*** — достоверностью (confidence)

$$C(A \rightarrow B) = P\{B|A\} = Q(A \cap B) / Q(A),$$

где $Q(B)$ — количество транзакций, содержащих B.

Единичная достоверность свидетельствует о детерминированности правила: «Если в транзакции есть ***A***, то в транзакции присутствует ***B***». Если достоверность совпадает (или близка) с частотой следствия, то это свидетельствует о независимости следствия от условия.

Для оценки значимости правил используют лифт (lift) $L(A \rightarrow B) = C(A \rightarrow B) / S(B)$.

Рис. 87. Связи наборов

Лифт является обобщенной мерой связи двух предметных наборов: при значениях лифта больше 1 связь положительная, при 1 она отсутствует, а при значениях меньше 1 — отрицательная. Экстремальные значения, свидетельствующие о тесной зависимости, тем не менее, могут оказаться малозначимыми в силу незначительной поддержки (частота применения правила незначительна).

Другой мерой значимости правила, предложенной Г. Пятецким-Шапиро, является левередж:

$$T(A \rightarrow B) = S(A \rightarrow B) — S(A)S(B).$$

Левередж — это разность между наблюдаемой частотой, с которой условие и следствие появляются совместно (то есть поддержкой ассоциации), и произведением частот появления (поддержек) условия и следствия по отдельности.

Еще одну характеристику, представляющую собой меру полезности ассоциативного правила, называют улучшением (improvement) и вычисляют подобно левереджу, только берется не разность, а отношение наблюдаемой частоты и частот появления по отдельности:

$$I(A \rightarrow B) = S(A \rightarrow B) / (S(A)S(B)).$$

Улучшение показывает, полезнее ли правило случайного угадывания, Если $I(A \rightarrow B) > 1$, это значит, что вероятнее предсказать наличие набора B с помощью правила, чем угадать случайно. Такие меры, как лифт и левередж, могут использоваться для последующего ограничения набора рассматриваемых ассоциаций путем установки порога значимости, ниже которого ассоциации отбрасываются. На Рис. 86 важность для первого правила составляет 0,254, это говорит о небольшом положительном влиянии условия (велосипедные подставки).

Наличие правил позволяет увеличить объемы продаж за счет предложения товаров, попадающих в один набор с приобретенным.

Для поиска ассоциаций нужно указать таблицу с данными и в надстройке интеллектуального анализа MS Excel выбрать команду «Поиск взаимосвязей» в меню «Интеллектуальный анализ данных». В настройке поиска ассоциаций (Рис. 88) указывают идентификатор транзакции, обозначение элементов, минимальное количество повторений набора в транзакциях и минимальную частоту (вероятность) правила. В данном случае элементом набора выбрана категория товара для выделения ассоциаций между категориями, а не товар, потому что вероятность правил для ассоциаций категорий будет выше вероятностей ассоциаций товаров.

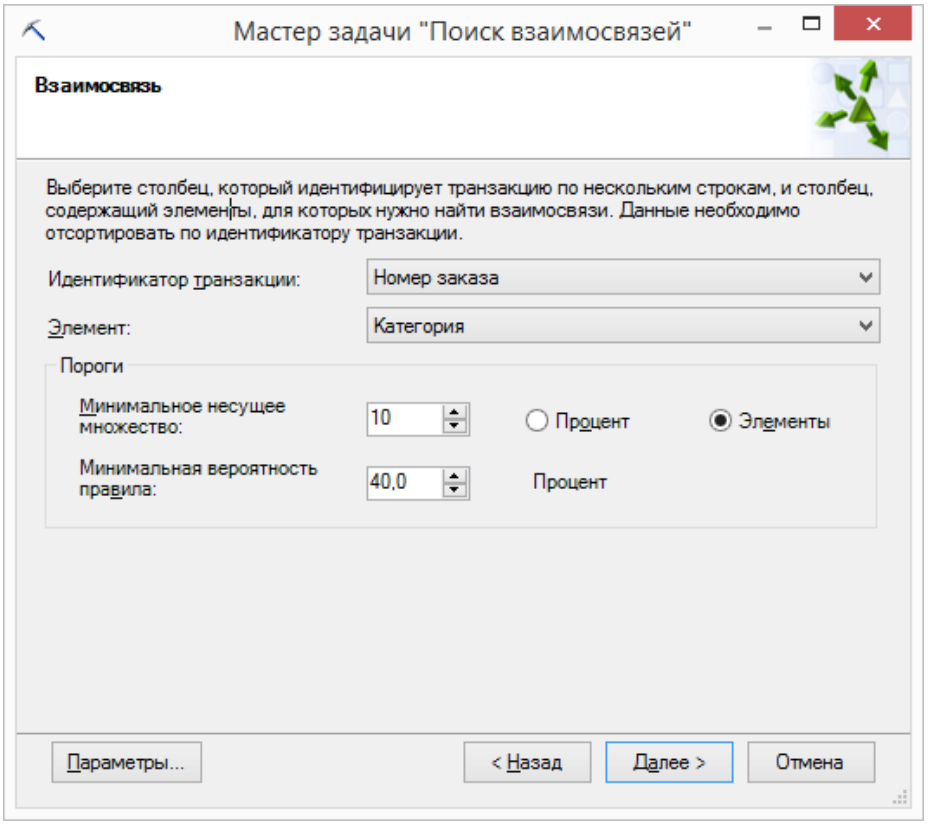

Рис. 88. Настройка поиска ассоциаций

Обзор модели включает список правил (Рис. 89). Компонент списка включает само правило, его вероятность и важность правила. Важность

$$\boldsymbol{I(A \rightarrow B) = \ln\left(\frac{P(B|A)}{P(B|\overline{A}}\right)}$$

характеризует связь компонентов набора:

$\boldsymbol{I(A \rightarrow B) = 0}$ — между A и B нет взаимосвязи,

$\boldsymbol{I(A \rightarrow B) > 0}$ — положительная связь,

$\boldsymbol{I(A \rightarrow B) < 0}$ — отрицательная связь.

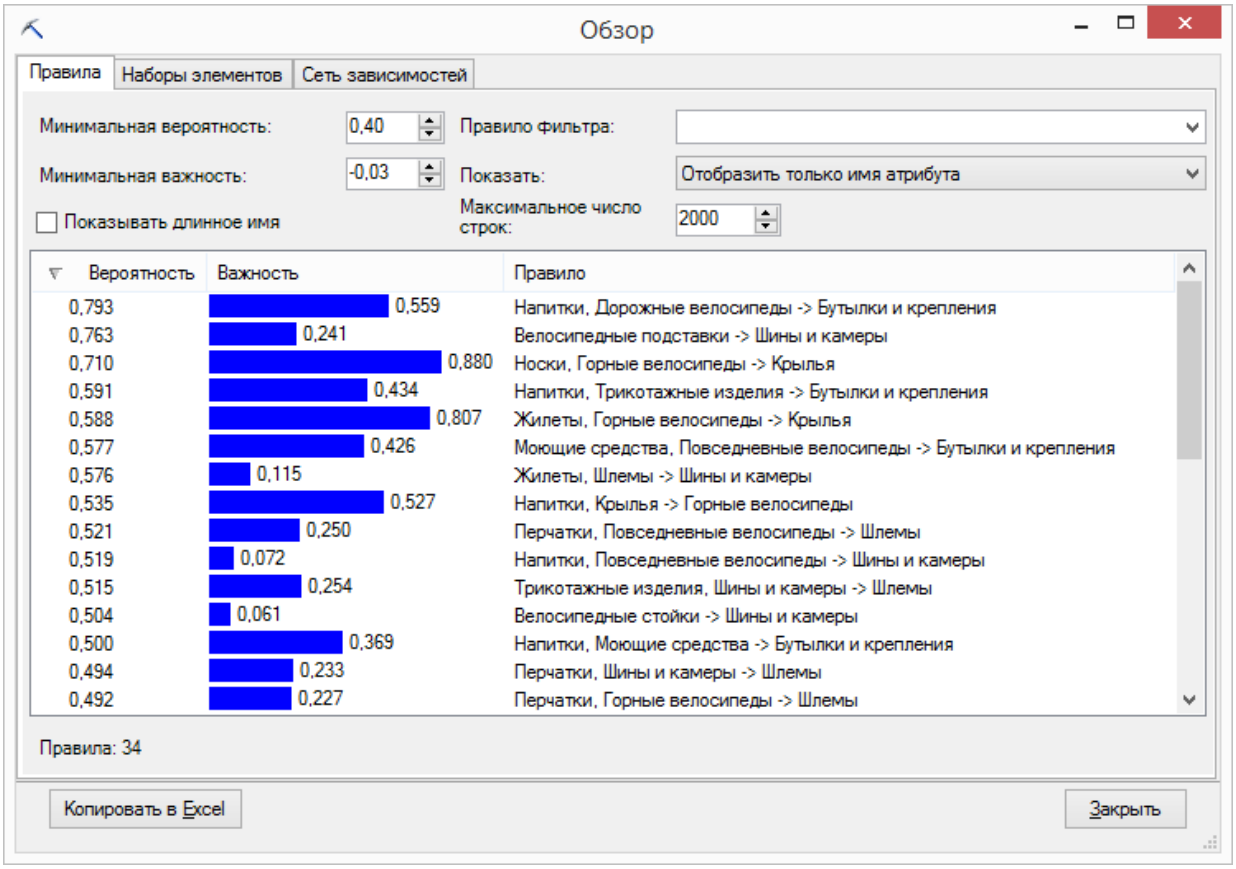

Рис. 89. Обзор правил ассоциаций

На следующей вкладке для каждого набора элементов приводится (Рис. 90) количество транзакций, содержащих набор (поддержку набора) и количество элементов набора (размер набора).

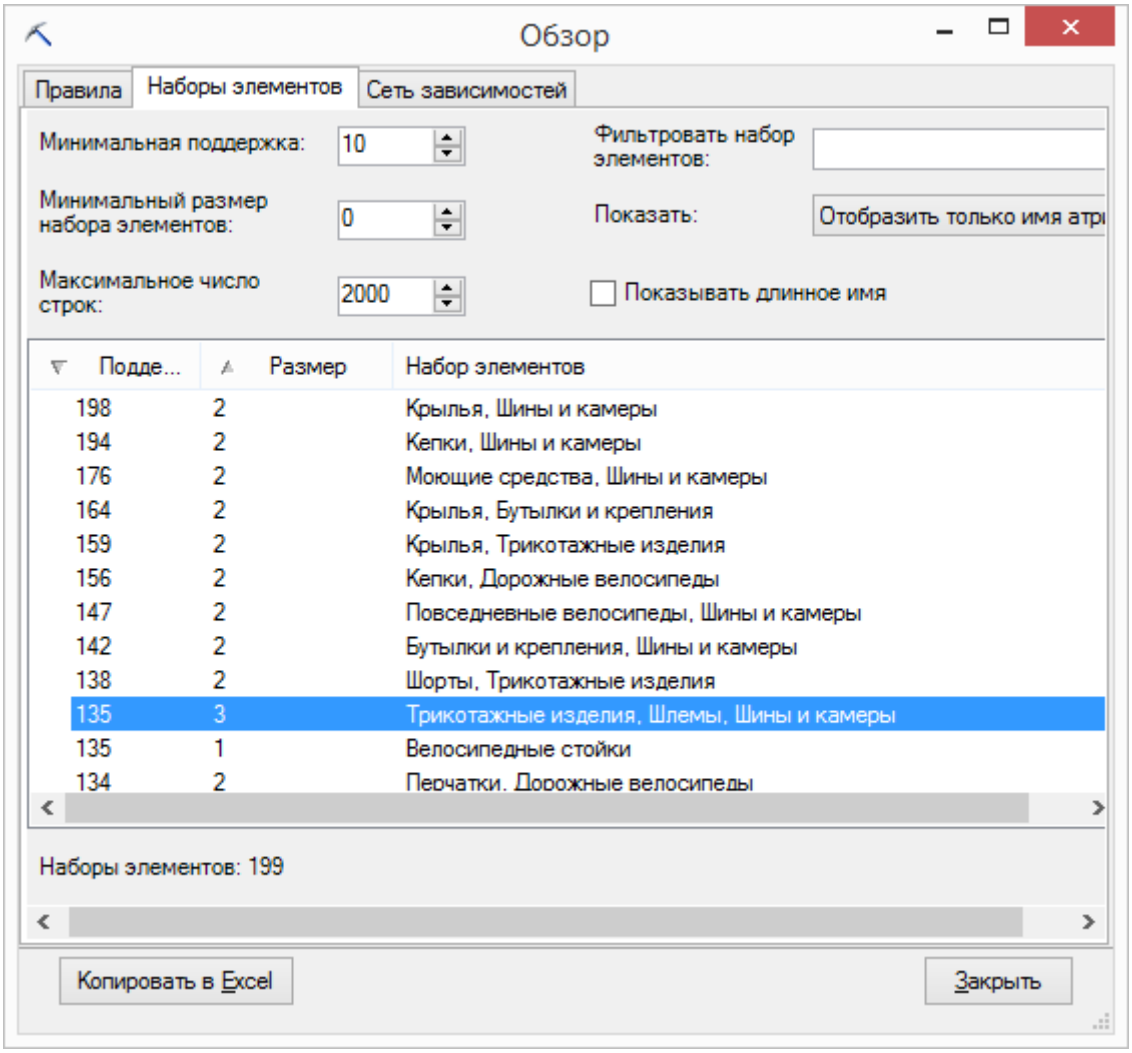

Рис. 90. Характеристики наборов

В сеть зависимостей (Рис. 91) включают зависимости по степени важности.

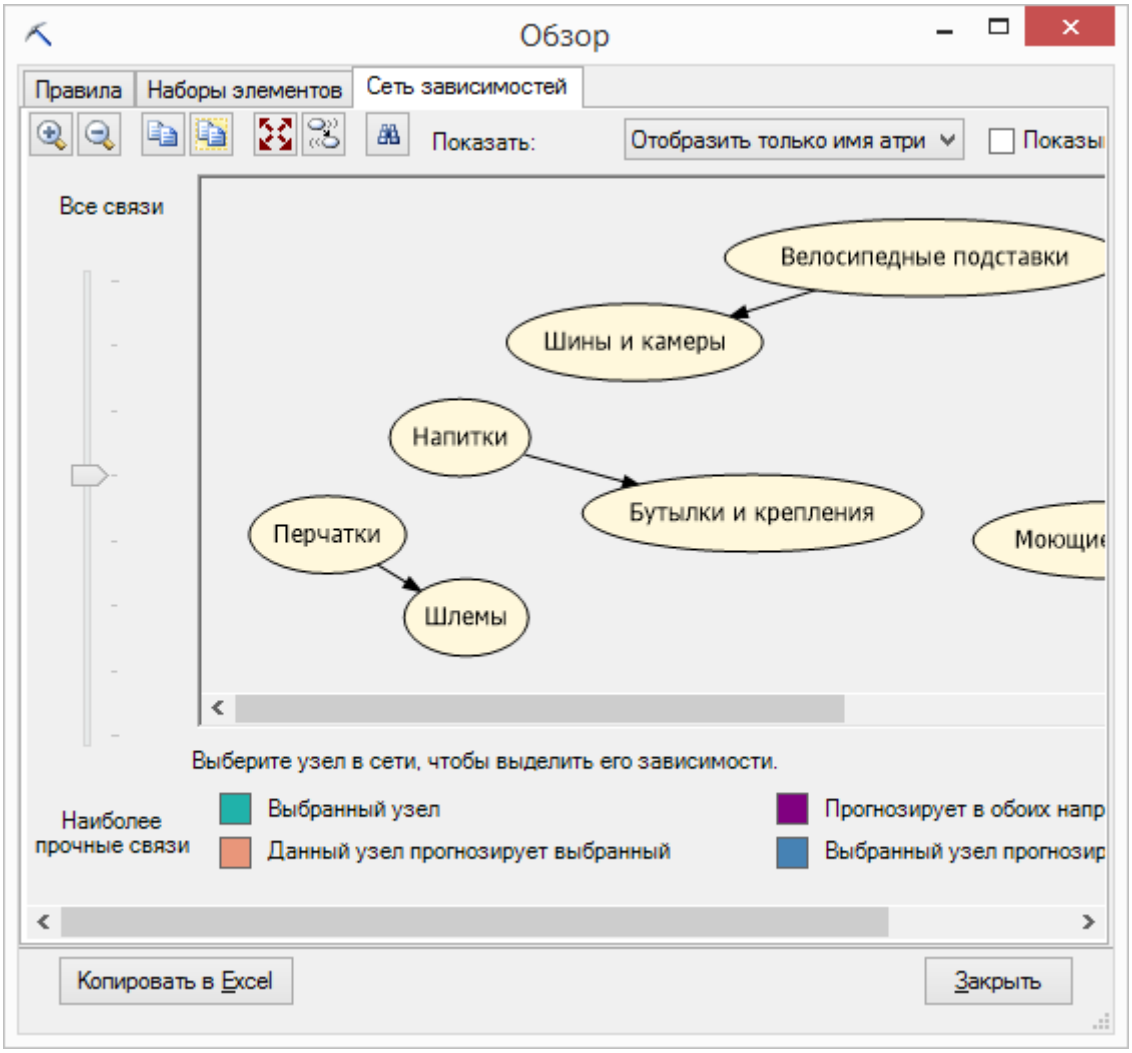

Рис. 91. Сеть зависимостей

Запрос к модели ассоциаций может выбрать товары (категории), которые с наибольшей вероятностью образуют набор с указанным товаром. Для этого на отдельном листе MS Excel введем в столбце заголовок «Категория» и ниже какое-нибудь значение категории, например, «Горные велосипеды». Для прогноза ассоциаций выберем команду «Запрос» в меню «Интеллектуальный анализ данных». С помощью запущенного мастера выберем модель ассоциаций и определим диапазон с введенными значениями с указанием наличия заголовков (Рис. 92). На следующем шаге устанавливается соответствие столбцов модели и данных для прогнозирования (Рис. 93). Далее кнопкой «Добавить выходные данные» задаем (Рис. 99) имя выходной таблицы («Прогноз»), количество прогнозируемых категорий (3), включение статистической информации (INCLUDE STATISTICS). После этого кнопкой «Дополнительно» задаем (Рис. 95) параметры запроса, заключенные в угловые скобки:

«Входные данные1» — исходные данные для прогнозирования,

<Столбец вариантов> — «Категория»,

<Вложенный столбец> — «Категория».

Результаты выполнения прогноза представлены на Рис. 96. Аналитические службы передали в MS Excel наиболее вероятные категории товаров, ассоциированные с горными велосипедами — это категории «Шины и камеры», «Шлемы», «Бутылки и крепления». Для каждой категории вычислена поддержка (SUPPORT), вероятность (PROBABILITY) и уточненная вероятность (ADJUSTED PROBABILITY), учитывающая частоту прогнозируемой категории). Такие прогнозы помогают подбирать наиболее вероятные наборы.

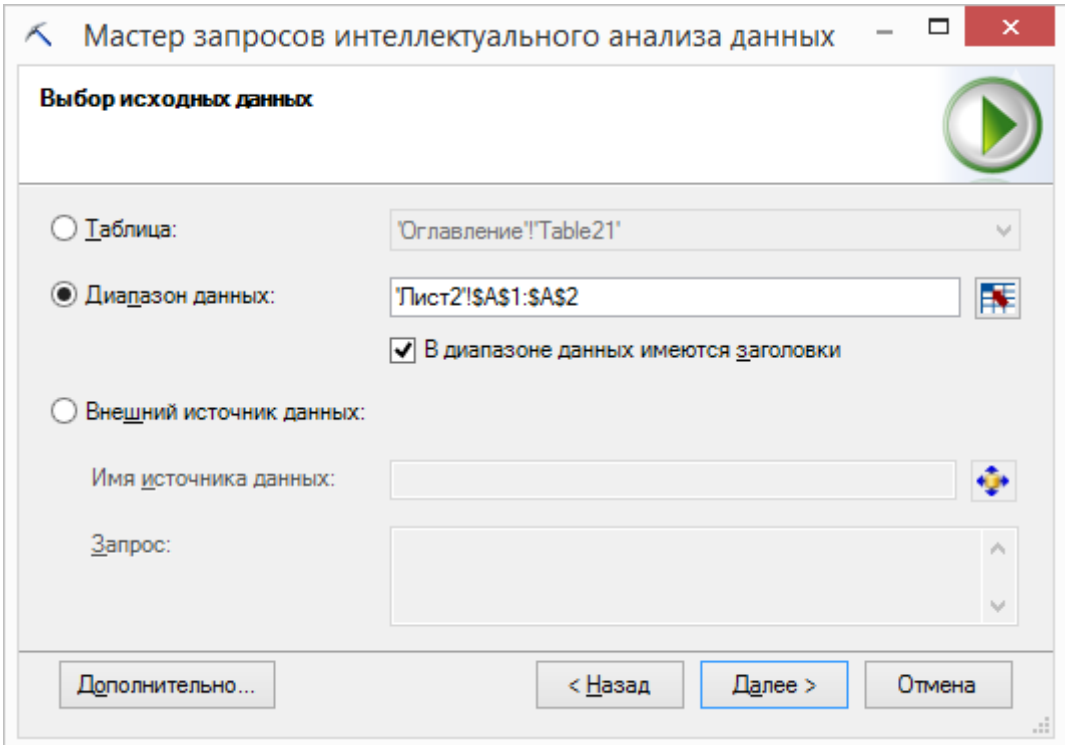

Рис. 92. Задание диапазона данных для прогноза

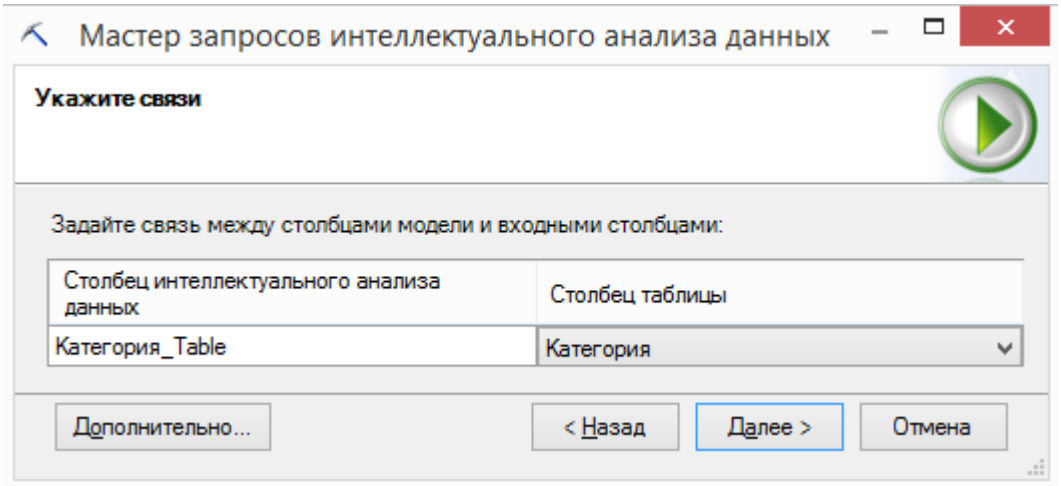

Рис. 93. Определение соответствия столбцов модели
и данных для прогнозирования

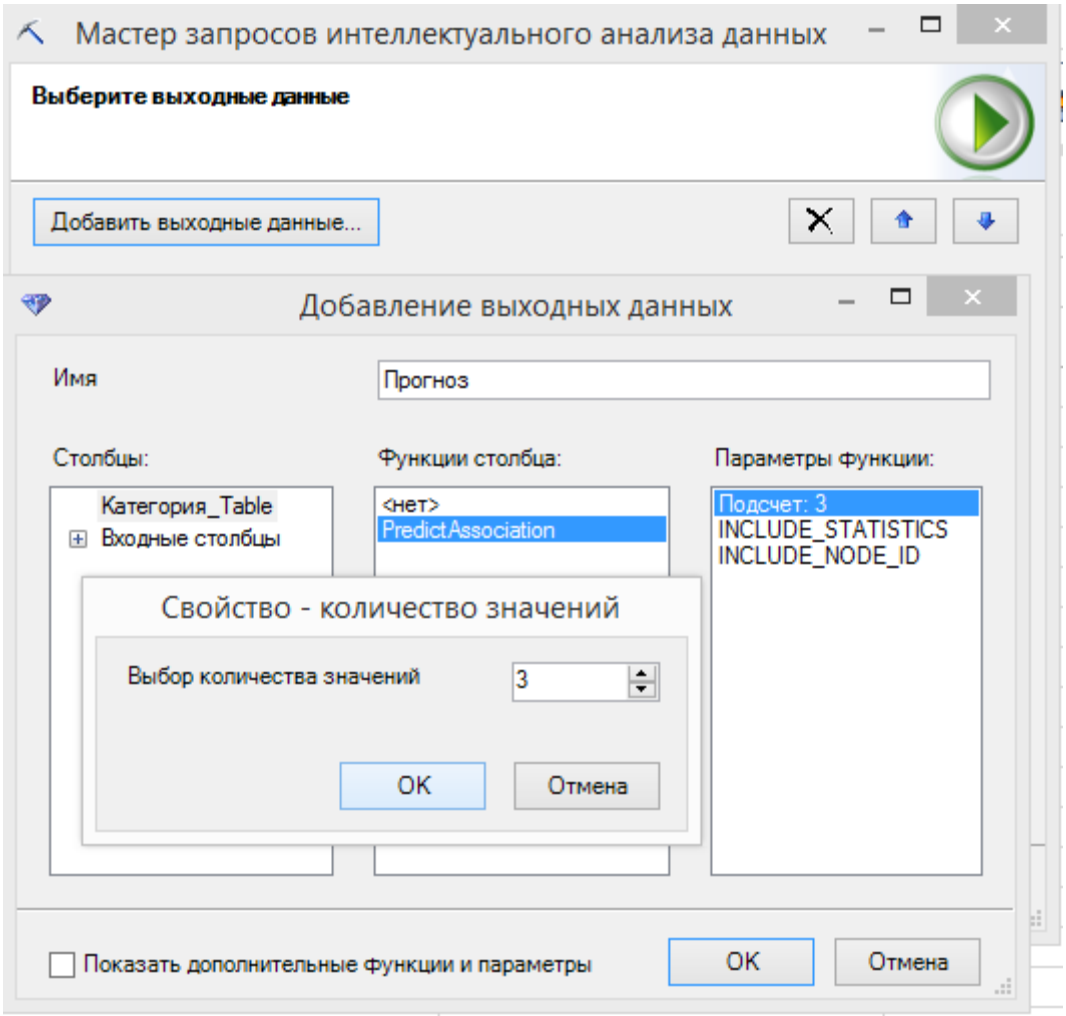

Рис. 94. Задание выходных данных для прогнозирования ассоциаций

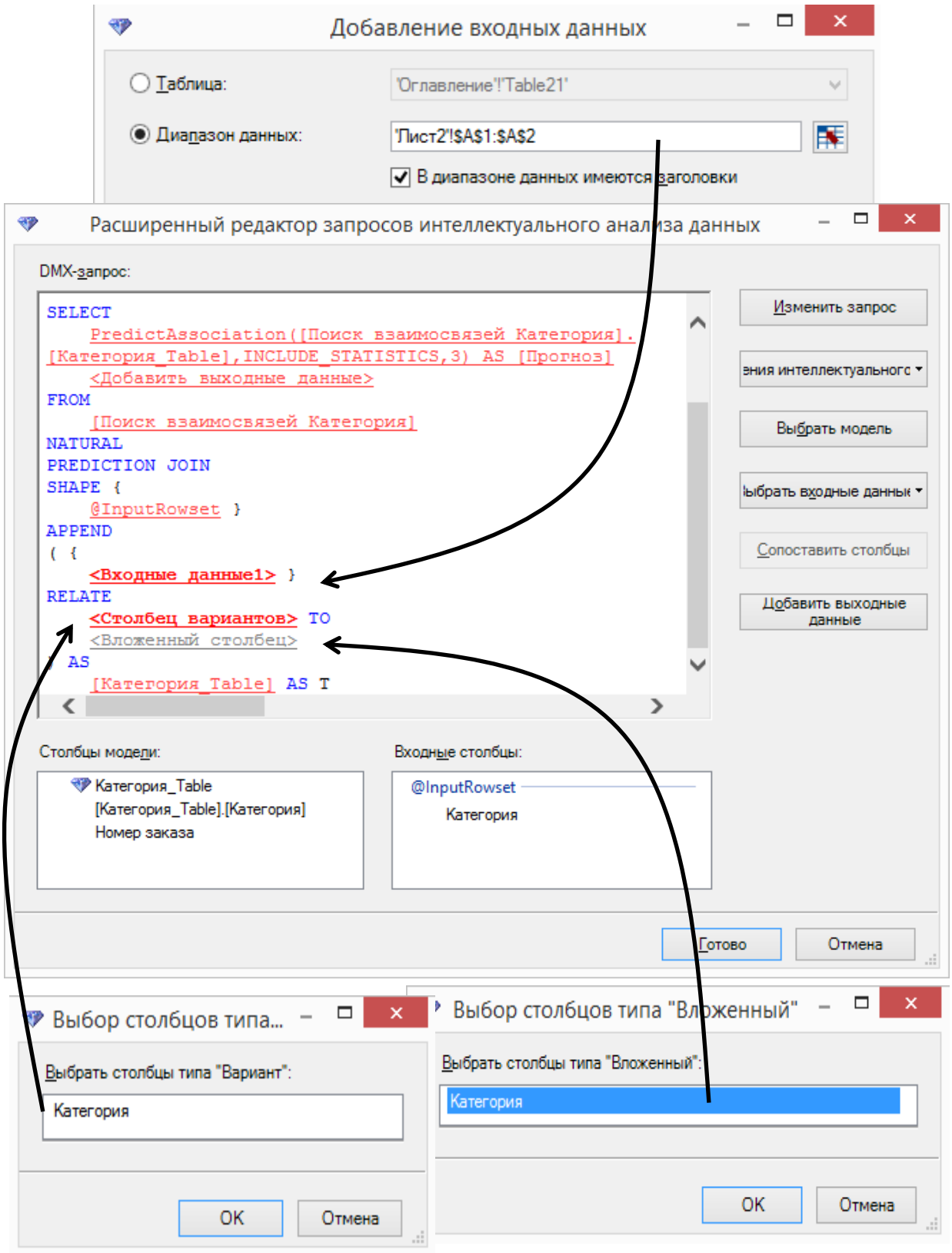

Рис. 95. Определение дополнительных параметров
для прогнозирования ассоциаций

	A	B	C	D
1	Категория			
2	Горные велосипеды			
3				
4	Прогноз.Категория	Прогноз.$SUPPORT	Прогноз.$PROBABILITY	Прогноз.$ADJUSTEDPROBABILITY
5	Шины и камеры	3990	0.436781609195402	0.353887478962945
6	**Шлемы**	**2676**	**0.292939244663383**	**0.253932533099606**
7	Бутылки и крепления	1994	0.218281335522715	0.19610075556475

Рис. 96. Результаты прогнозирования ассоциаций

3.8. Кластеризация последовательностей

При кластеризации последовательностей набор (множество), в котором порядок не имеет значения, заменяется на последовательность, для которой порядок следования определен. Можно привести следующие примеры последовательностей:

– последовательности веб-страниц, которые просматривают пользователи на сайте;

– журналы, в которых приведены списки событий, предшествовавших инциденту, такие как сбой жесткого диска или блокировки;

– записи транзакций, описывающие порядок, в котором клиент добавляет в корзину товары, выбранные в интернет-магазине;

– записи, следящие за взаимодействием с клиентом (или пациентом) во времени для прогнозирования отмены услуг или других нежелательных итогов.

Используемый аналитическими службами SQLServer 2008 алгоритм Micorosoft Sequence Clustering — это гибридный алгоритм, сочетающий методы кластеризации с анализом марковских цепей.

С помощью марковских моделей анализируется направленный граф, хранящий переходы между различными элементами последовательностей. Алгоритм Microsoft Sequence Clustering использует марковские цепи n-го порядка. Число n говорит о длине предыстории: если n = 1 — вероятность текущего состояния зависит только от предыдущего состояния, если n = 2 — вероятность текущего состояния зависит от двух предыдущих состояний, и так далее.

Вероятности перехода между состояниями образуют матрицу переходов. По мере увеличения порядка марковской цепи размер матрицы растет экспоненциально, соответственно растет и время обработки, что надо учитывать при решении практических задач.

Алгоритм пытается выделить возможные последовательностями, чтобы определить, какие последовательности лучше всего использовать в качестве входных данных для кластеризации. Созданный алгоритмом список последовательностей используется в качестве входных данных для применяемого по умолчанию EM-метода кластеризации (англ. Expectation Maximization, максимизации ожидания). Для каждого кластера определяется марковская цепь, описывающая вероятности последовательности состояний.

В надстройке интеллектуального анализа MS Excel отсутствуют средства построения модели кластеризации последовательностей. Построить эту модель (так же как и все остальные) можно с помощью SQL Server Data Tools. Исходными данными являются описания транзакций (Рис. 97), которые так же, как в алгоритме ассоциаций, идентифицируются определенным атрибутом (OrderNumber для примера). Дополнительно должен присутствовать атрибут (LineNumber), определяющий порядок следования.

OrderNumber	Region	IncomeGroup	LineNumber	Model
SO51176	Pacific	High	1	Road-250
SO51176	Pacific	High	2	Road Bottle Cage
SO51177	Pacific	Low	1	Touring-2000
SO51177	Pacific	Low	2	Sport-100
SO51178	Europe	High	1	Mountain-200
SO51178	Europe	High	2	Mountain Bottle Cage
SO51178	Europe	High	3	Water Bottle
SO51179	Europe	Moderate	1	Road-250
SO51179	Europe	Moderate	2	HL Road Tire

Рис. 97. Данные для кластеризации последовательностей

В SQL Server Data Tools создается проект интеллектуального анализа (Рис. 98).

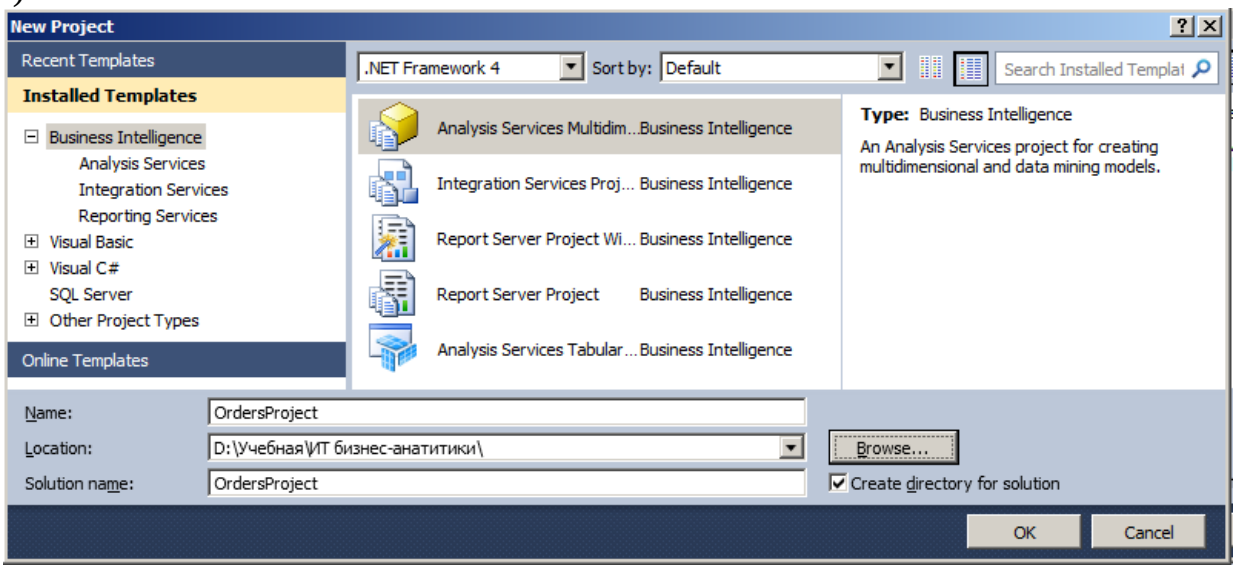

Рис. 98. Создание проекта интеллектуального анализа

В проекте командой «Проект» \ «Источник данных» определяется источник данных для модели. Основными параметрами источника данных являются (Рис. 99) поставщик — источник данных (на рис. — SQL сервер), имя сервера, база данных для настройки модели.

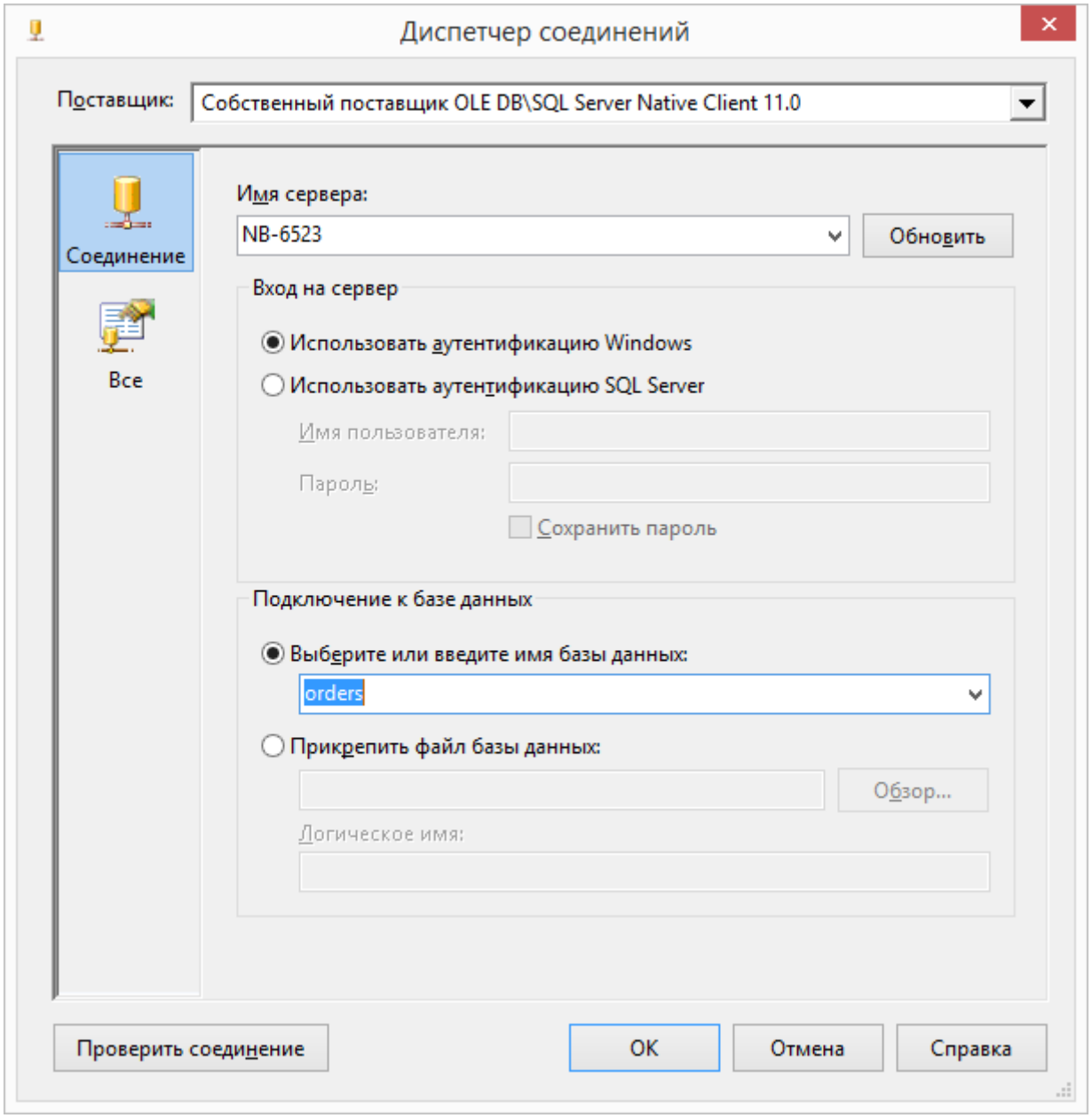

Рис. 99. Определение источника данных

Далее создается представление данных командой «Проект» \ «Создание представления источника данных». Соответствующий мастер позволяет определить какие именно данные будут использоваться для модели интеллектуального анализа (Рис. 100). Исходные данные (Рис. 97) сгруппированы в две исходные таблицы: одна описывает заказы, вторая — следование заказанных товаров.

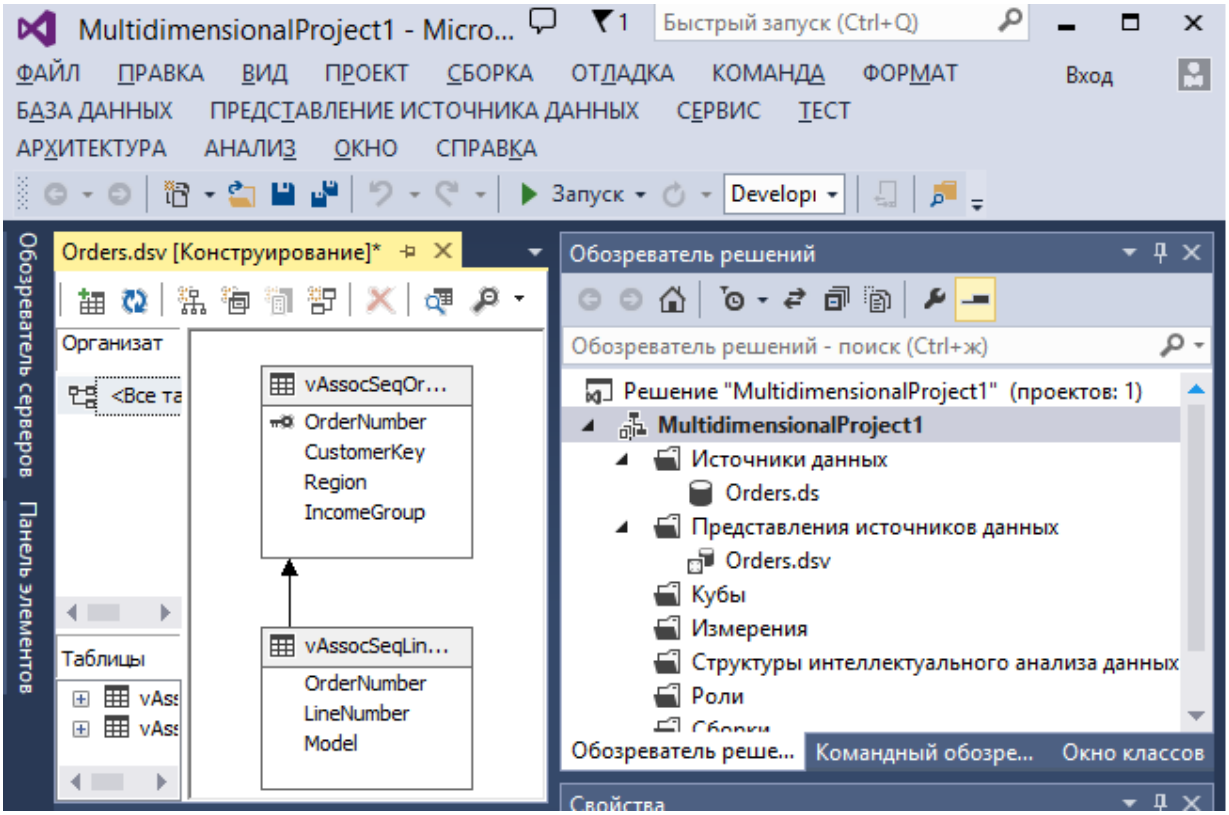

Рис. 100. Определение представления данных

Командой «Проект» \ «Создание структуры интеллектуального анализа» запускается мастер, в котором последовательно определяются:

1) выбор данных из реляционной базы;
2) выбор алгоритма кластеризации последовательностей;
3) выбор представления данных (vAssocSeqOrders);
4) выбор таблиц и их использования (на Рис. 101 — таблица vAssocSeqOrders определяет варианты последовательностей, таблица vAssocSeqLineItems является вложенной для каждого заказа и описывает последовательность выбора);
5) определение данных для модели (на Рис. 102 — выбираются входные данные и ключи, определяются выходные данные, которые надо прогнозировать — последовательно выбираемые модели);
6) определение типов данных (на Рис. 103 — определяются типы на основании типов данных в базе, которые можно переопределить исходя из их содержания);
7) определение объема проверочных данных — тестового множества, на котором будет проверяться качество моделирования (по умолчанию 30 %);
8) задание имен структуры и модели (Рис. 104).

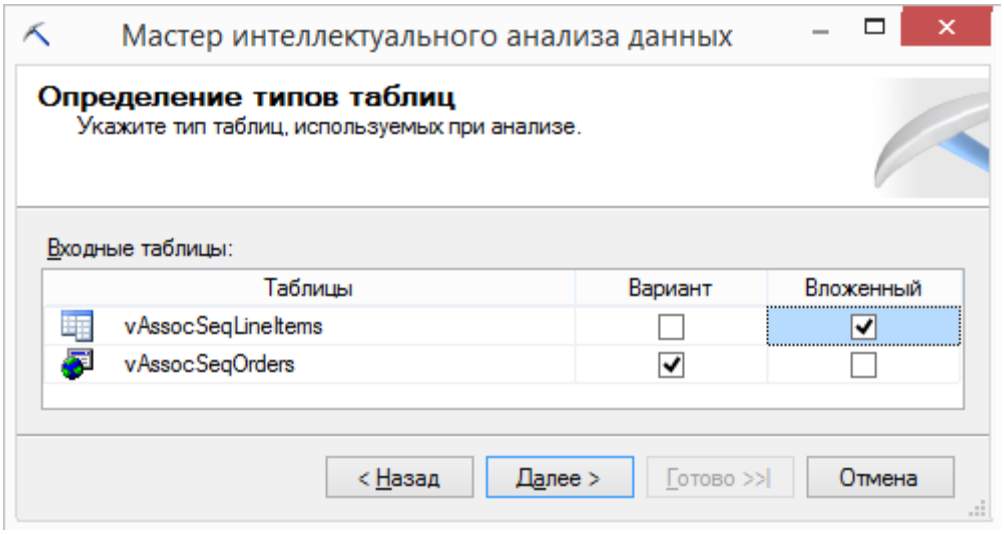

Рис. 101. Выбор таблиц

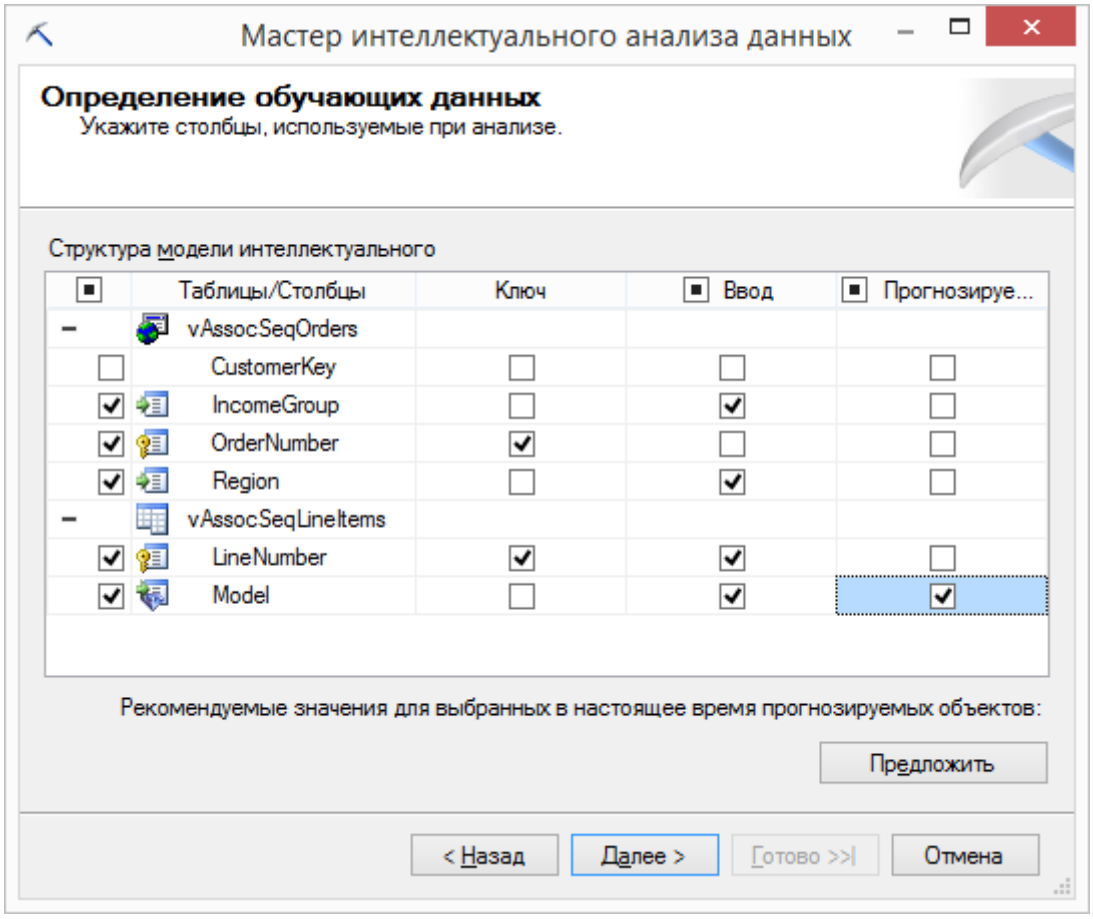

Рис. 102. Определение данных для модели

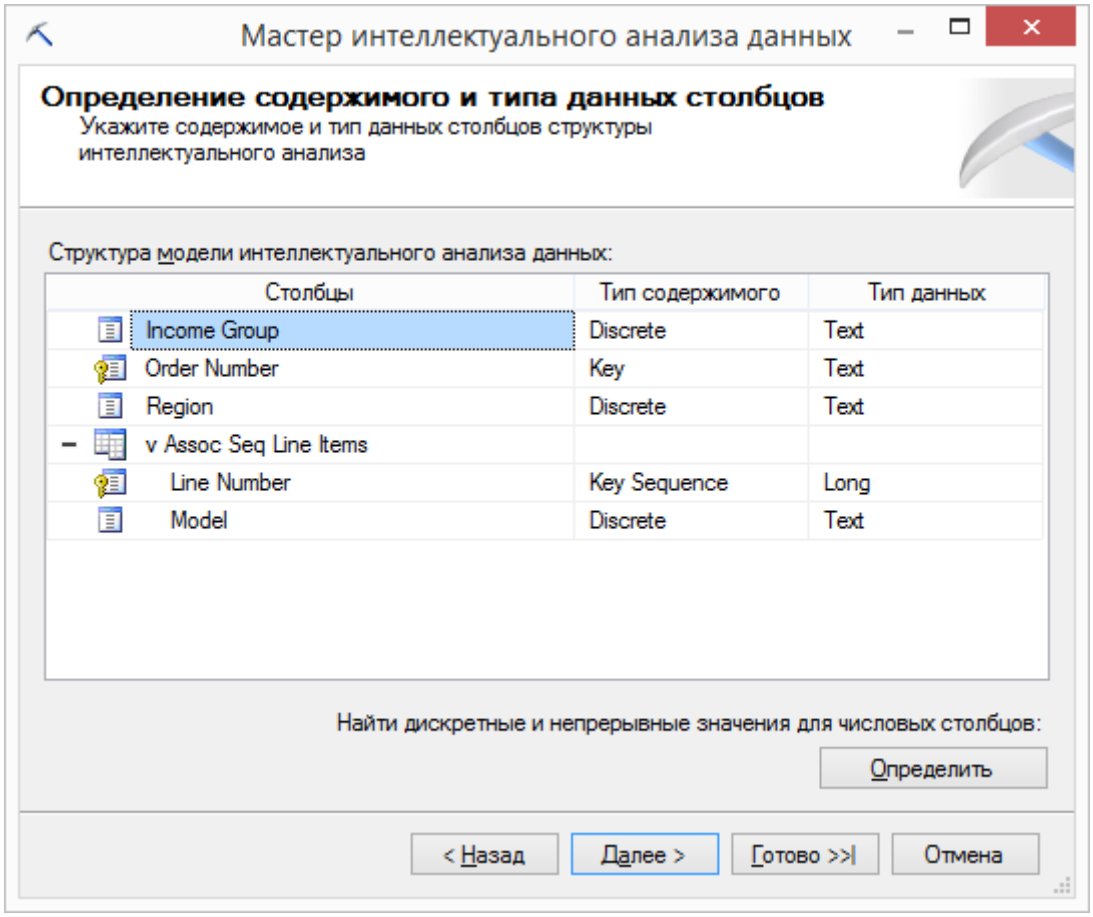

Рис. 103. Определение типов данных

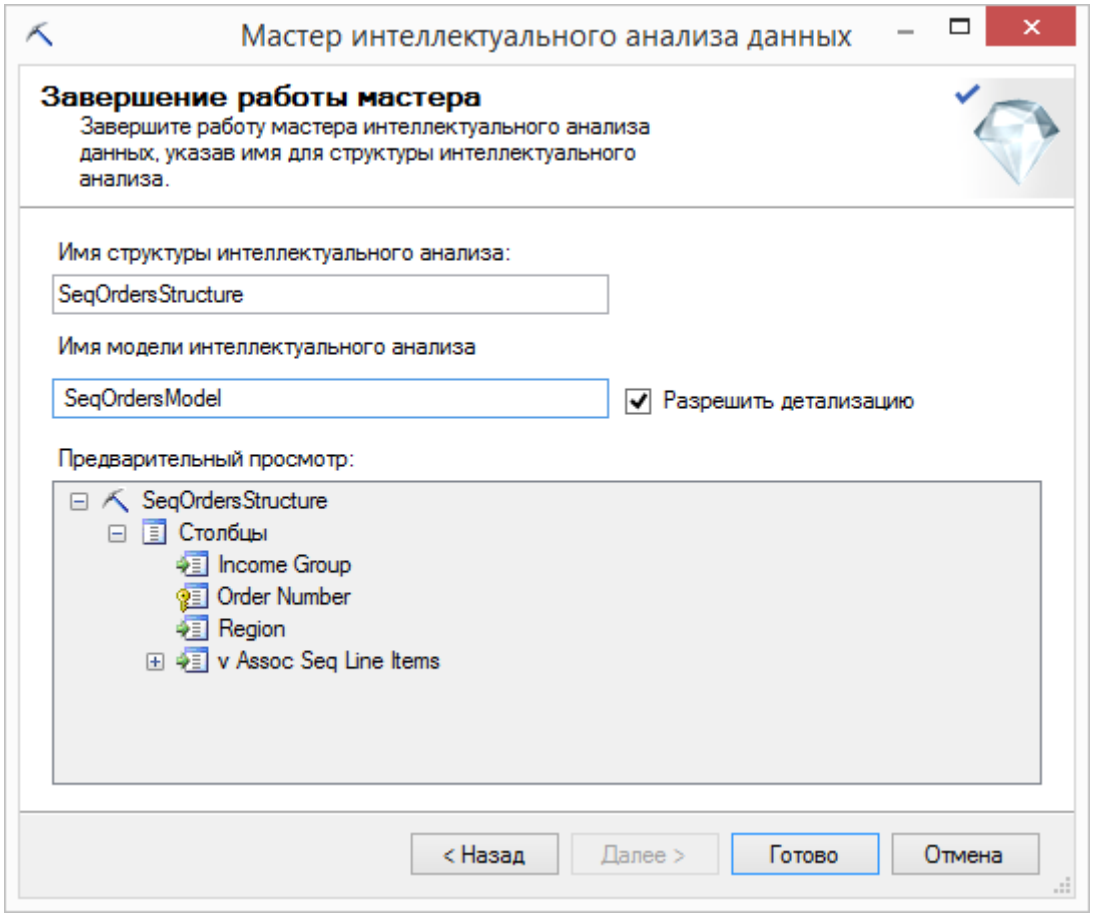

Рис. 104. Задание имен структуры и модели

Созданный проект необходимо развернуть на выбранном сервере. Параметры развертывания задаются в свойствах проекта (Рис. 105): выбирается сервер и база данных на сервере, которая будет хранить структуру и модель и предоставлять доступ к данным модели. Команда «Развернуть» приводит к переносу всех определений в указанную параметрами базу данных на выбранном сервере.

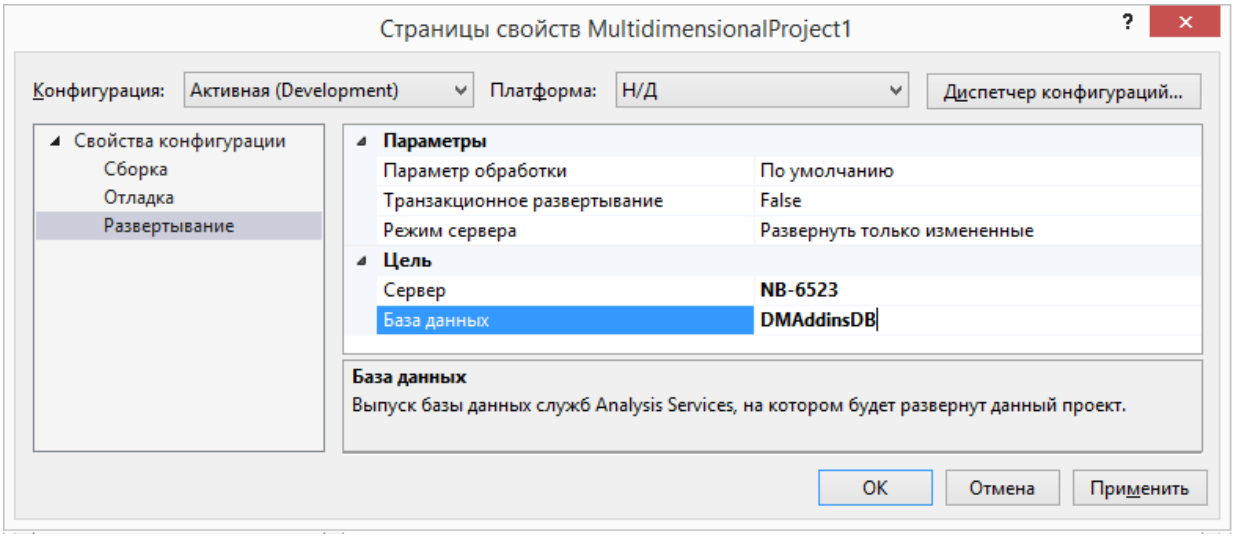

Рис. 105. Определение параметров развертывания

Развертывание создает структуры в базе данных на сервере. Для заполнения структур данными и настройки модели необходимо выполнить обработку.

Команда «Обработать» включает процесс передачи данных на сервер и определение параметров модели интеллектуального анализа. После обработки модель готова для применения.

Обзор модели кластеризации последовательностей представлен диаграммой кластеров (Рис. 106), демонстрирующей близость кластеров.

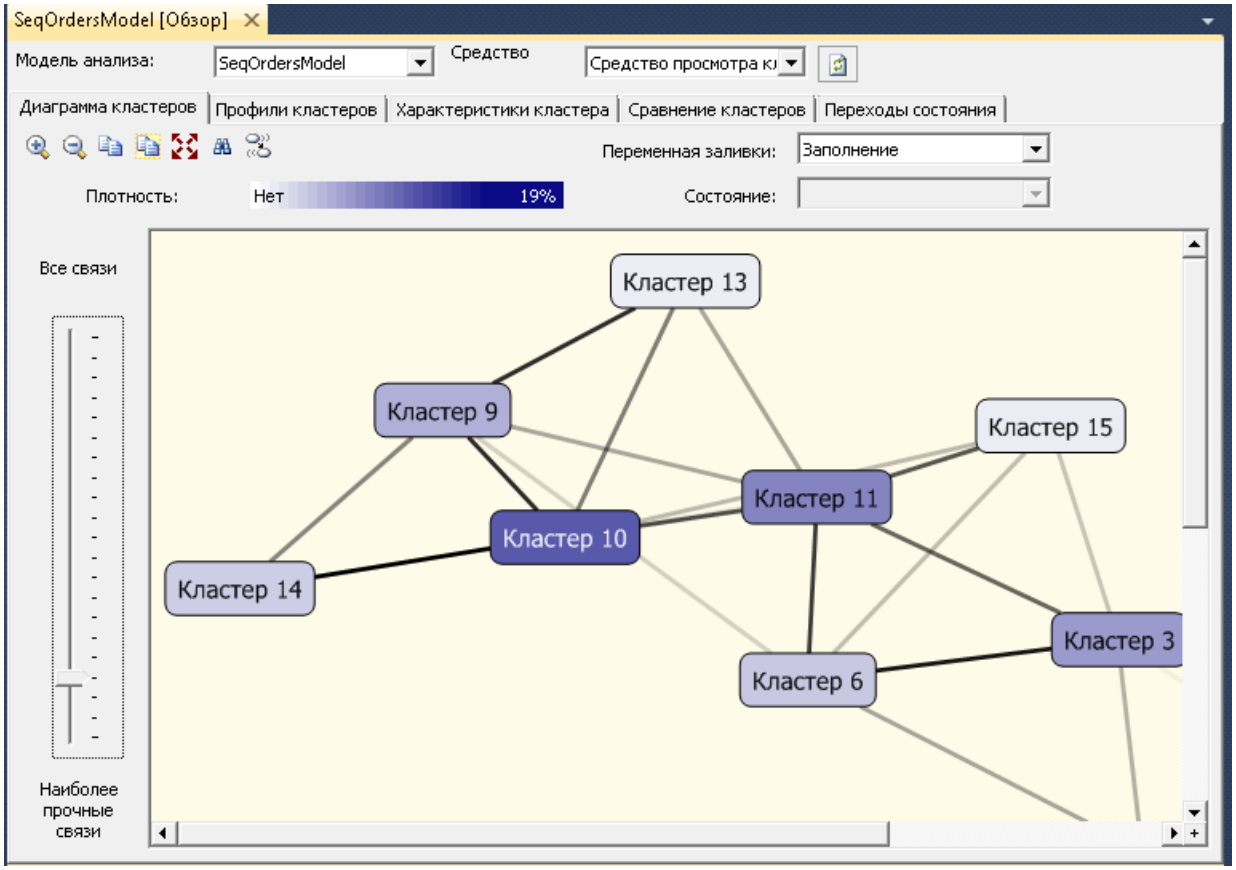

Рис. 106. Диаграмма кластеров

Профили кластеров (Рис. 107) позволяют понять разницу кластеров на основании частотных профилей атрибутов.

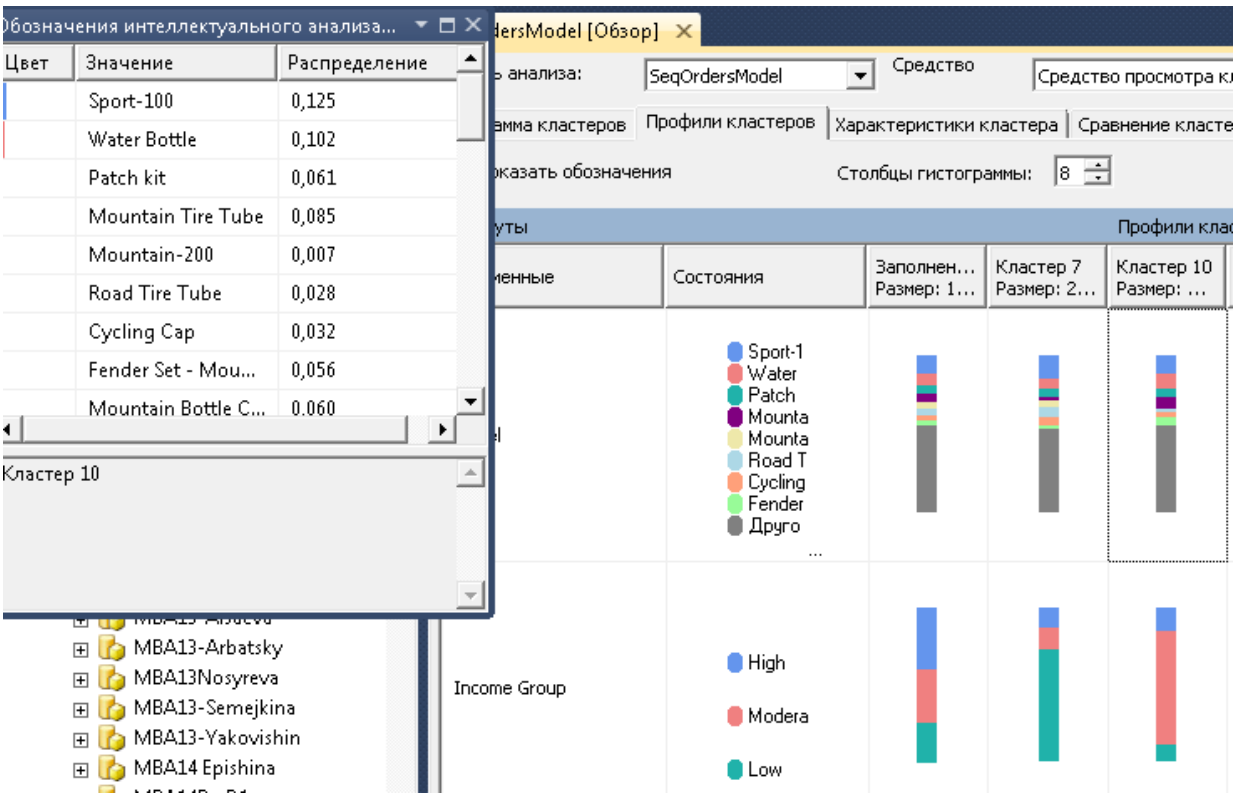

Рис. 107. Профили кластеров

Характеристики кластера (Рис. 108) содержат частоты различных атрибутов и переходов, отсортированных по убыванию частоты.

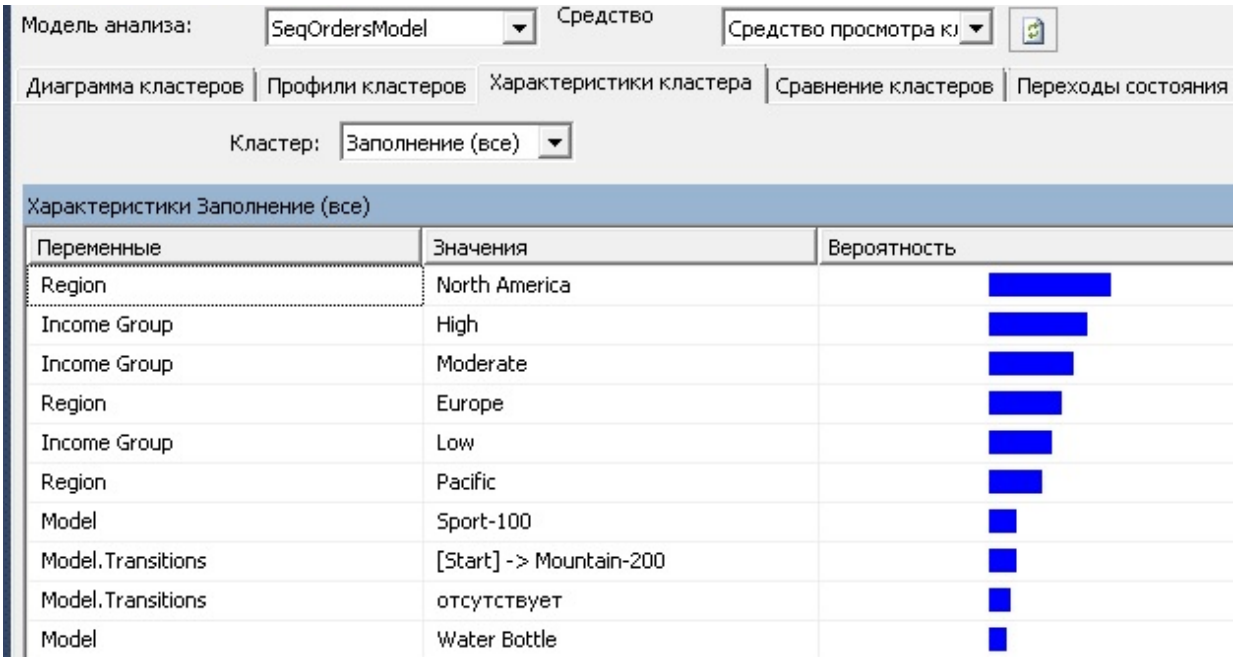

Рис. 108. Характеристики кластера

Сравнение кластеров (Рис. 109) демонстрирует различие выбранных кластеров по частотным характеристикам атрибутов.

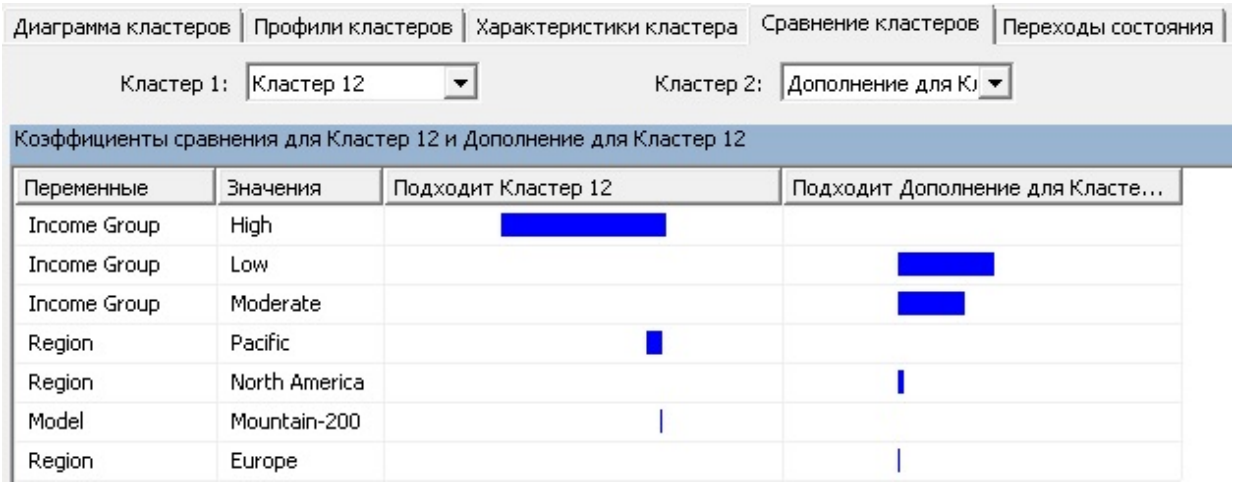

Рис. 109. Сравнение кластеров

Диаграмма переходов состояний (Рис. 110) демонстрирует выявленные закономерности следования событий. Для перехода на диаграмме показана вероятность такого события. Чем больше вероятность, тем сильнее связь событий и больше возможности предсказания следования.

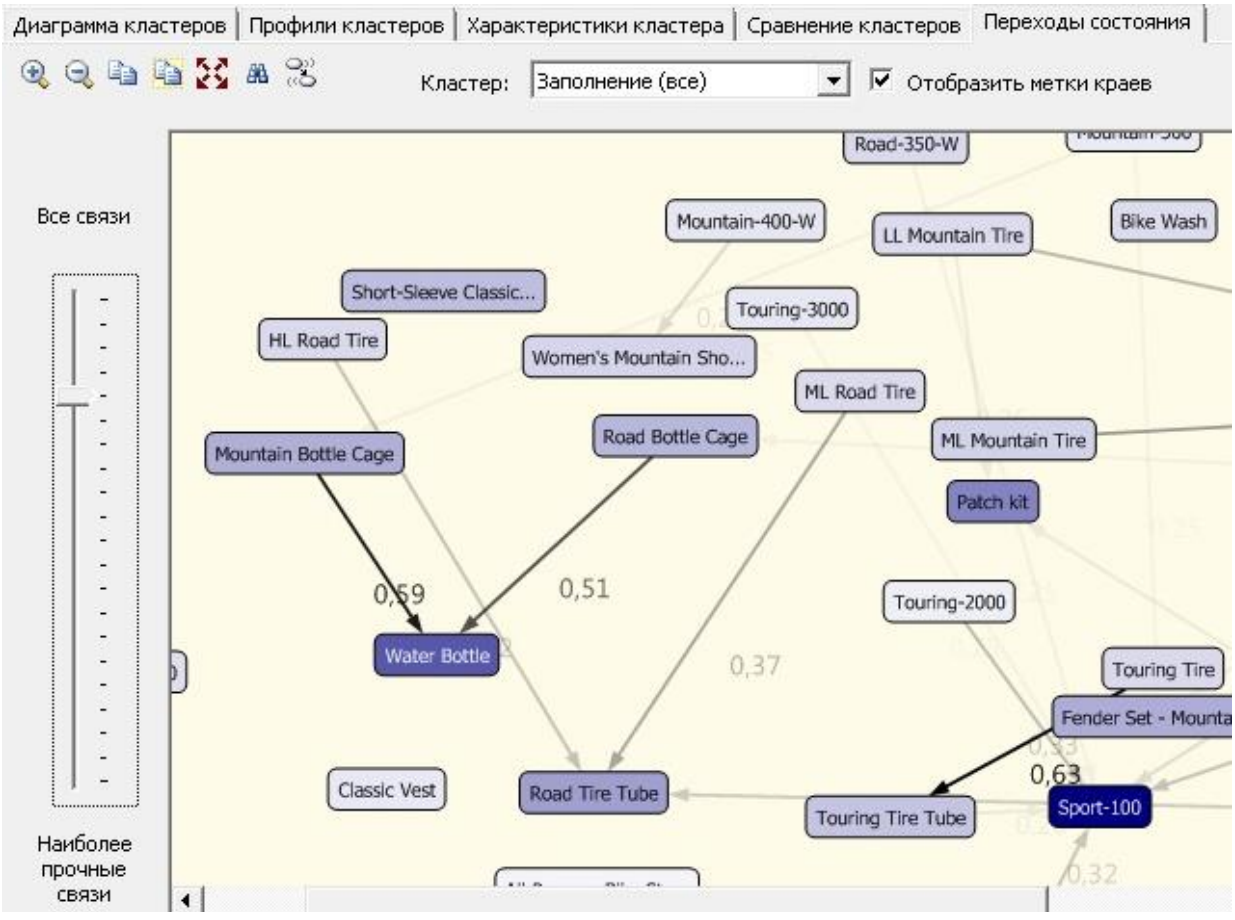

Рис. 110. Переходы из состояния в состояние

Запросы к моделям интеллектуального анализа

Графические средства определения моделей и построения запросов преобразуют в текстовые команды, передаваемые аналитическим службам. Для этого

применяется специальный язык Data Mining Extensions (DMX). Надстройка интеллектуального анализа содержит средства трассировки. Соответствующее окно (Рис. 111) включается командой «Соединение» \ «Трассировщик» на вкладке «Интеллектуальный анализ данных».

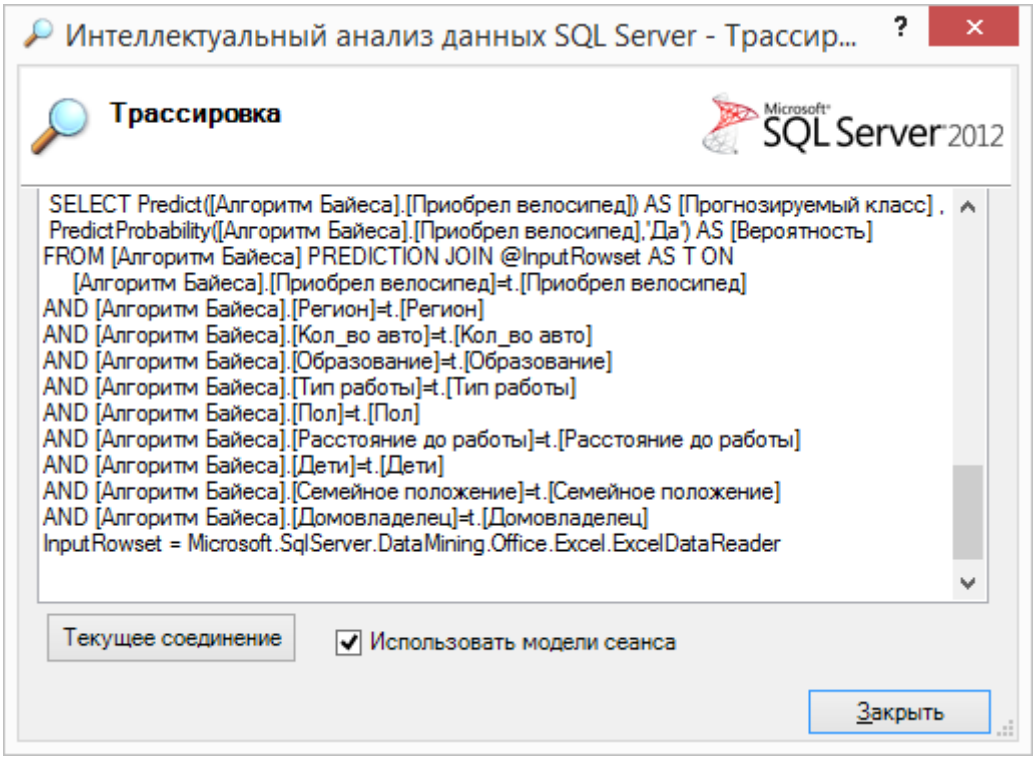

Рис. 111. Трассировка команд DMX в надстройке интеллектуального анализа

На рисунке представлена команда прогнозирования покупки велосипеда и определения вероятности такой покупки.

Для построения прогнозирующего запроса используется следующая команда:

SELECT <список прогнозируемых величин>
FROM <модель> PREDICTION JOIN <таблица с исходными для прогноза данными> ON <условие соответствия полей модели и данных>
WHERE <условие выбора>
ORDER BY <поля сортировки>

Список прогнозируемых величин позволяет задать столбцы и выражения, отображаемые в результирующем наборе, и может включать следующие данные:

– столбцы модели интеллектуального анализа данных Predict или PredictOnly;

– любой столбец входных данных, используемый при создании прогнозов;

– функции, возвращающие столбец данных;

В примере вычисляются

– прогнозируемый моделью класс с помощью функции Predict([Алгоритм Байеса].[Приобрел велосипед]);

– вероятность положительной классификации, используя функцию PredictProbability([Алгоритм Байеса].[Приобрел велосипед],'Да').

Для вычисления используется настроенная модель «Алгоритм Байеса». Исходные данные предоставляет @InputRowset, получающий данные из таблицы Excel:

InputRowset = Microsoft.SqlServer.DataMining.Office.Excel.ExcelDataReader

Конъюнкция условий в пункте WHERE устанавливает соответствие имен колонок модели и таблицы с исходными данными. В результате для каждой строки таблицы исходных данных вычисляется наиболее вероятный класс клиента и вероятность положительной классификации. При совпадении имен колонок модели и исходных данных прогнозирования можно употреблять оператор NATURAL PREDICTION JOIN без указания условий соответствия полей.

Запрос

```
SELECT
    PredictAssociation([Поиск взаимосвязей Категория].[Категория_Table],
INCLUDE_STATISTICS,5) AS [Выходные данные1]
    FROM [Поиск взаимосвязей Категория]
    NATURAL PREDICTION JOIN
    (SELECT
    (SELECT 'Горные велосипеды' AS [Категория]
    UNION SELECT 'Шины и камеры' AS [Категория])
    AS [Категория_Table])
    AS t
```

формирует при помощи функции PredictAssociation прогноз и статистические характеристики для набора {'Горные велосипеды', 'Шины и камеры'}. Вычисляются какие еще 5 товаров могут войти дополнительно в этот набор. Данные прогнозирования (Рис. 112) включают наиболее перспективные товары, отсортированные в порядке убывания вероятности.

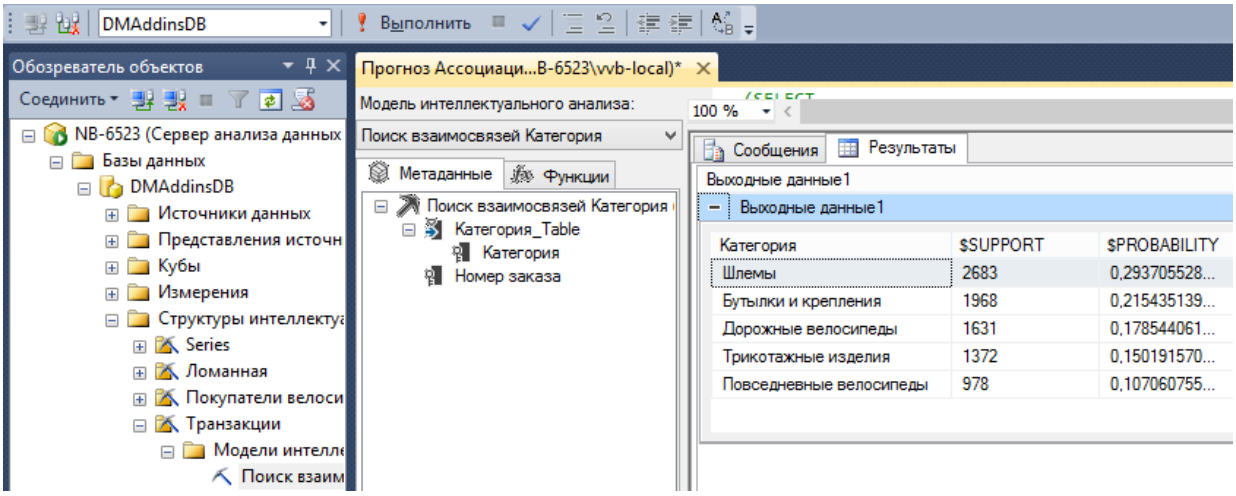

Рис. 112. Результаты прогнозирующего запроса для модели ассоциаций

Используется NATURAL PREDICTION JOIN для соединения модели и SELECT-конструктора таблицы «Категория_Table», включающей вложенную таблицу (вложенный SELECT) из двух категорий «Горные велосипеды» и «Шины и камеры».

Для каждой модели используются специальные функции для извлечения требуемой информации. Для кластеризации последовательностей необходимо предсказывать следующие компоненты последовательности. Для этого можно использовать функцию PredictSequence(<колонка, n>). Например, в качестве списка рекомендаций можно вернуть список из товаров, которые данный клиент приобретет с наибольшей вероятностью. Для модели из 3.2.13 запрос

```
SELECT FLATTENED PredictSequence([v Assoc Seq Line Items], 7)
FROM [SeqOrdersModel]
NATURAL PREDICTION JOIN
(SELECT  (SELECT 1 as [Line Number],
  'Road-250' as [Model]) AS [v Assoc Seq Line Items])
AS t
```

возвращает семь наиболее вероятных прогнозов (Таблица 13). Поскольку модель включает в себя вложенную таблицу, при прогнозах нужно использовать вложенную таблицу [v Assoc Seq Line Items] как ссылку на столбец. Кроме того, при передаче входных значений нужно сформировать соединение таблицы вариантов и столбцов вложенной таблицы, как показано во вложенной инструкции SELECT.

Таблица 13

Результаты предсказания следующего компонента
после выбора велосипеда «Road-250»

Expression.$SEQUENCE	Expression.Line Number	Expression.Model
1		Road Bottle Cage
2		Water Bottle
3		Sport-100
4		Half-Finger Gloves
5		Cycling Cap
6		Cycling Cap
7		Water Bottle

Столбец $sequence — это столбец, возвращаемый по умолчанию функцией PredictSequence для упорядочивания результатов прогноза. Столбец [Line Number] нужен для сопоставления ключей последовательностей в модели, но ключи не выводятся.

Интересно заметить, что в последовательности есть два компонента «Cycling Cap» («Велосипедная шапочка»). Это не ошибка. В зависимости от представления данных и их группировки при обучении модели появление таких последовательностей вполне вероятно. Например, покупатель мог приобрести одну велосипедную шапочку (красную), а затем другую (синюю), или приобрести две шапочки одного цвета одну за другой.

При достижении конца цепочки возможных переходов результаты прогнозирования не прекращаются, но значение, переданное в качестве входного, прибавляется к результатам. Например, если увеличить число прогнозов до 20, то в последних строках будет содержаться одно и то же значение, «All-Purpose Bike Stand» («Универсальная подставка для велосипеда»).

Вопросы

1. Перечислите задачи исследования зависимостей, укажите исходные данные, результаты решения каждой задачи.
2. Для каждой задачи исследования зависимостей приведите примеры применения решения.
3. Приведите общий вид модели аналитической обработки.
4. Как оценивается качество аналитической модели.
5. Применение надстройки MS Excel интеллектуального анализа данных.
6. Постановка задачи классификации.
7. Решение задачи классификации упрощенным алгоритмом Байеса.
8. Применение дерева решений для задачи классификации.
9. Применение логистической регрессии для решения задачи классификации.
10. Применение нейронной сети для решения задачи классификации.
11. Применение надстройки MS Excel интеллектуального анализа данных для решения задачи классификации.
12. Точность и эффективность классификации.
13. Зависимость прибыли от точки отсечения в задачах классификации.
14. Запросы к модели классификации в надстройке MS Excel интеллектуального анализа данных.
15. Регрессионная модель и ее решение в надстройке MS Excel интеллектуального анализа данных.
16. Задача кластеризации и ее применение.
17. Решение задачи кластеризации в надстройке MS Excel интеллектуального анализа данных.
18. Общие сведения по анализу временных рядов. Детерминированная и случайная компоненты временного ряда.
19. Стационарные временные ряда, модель авторегрессии и скользящего среднего стационарного ряда.
20. Модель авторегрессии и интегрированного скользящего среднего нестационарного случайного ряда.
21. Модель дерева авторегрессии с перекрестным прогнозированием (autoregressive tree with cross prediction — ARTXP).
22. Построение прогнозов в надстройке MS Excel интеллектуального анализа данных.
23. Алгоритм взаимосвязей или ассоциативных правил: исходные данные, характеристики ассоциативных правил.
24. Использование модели ассоциаций для прогнозирования.
25. Модель кластеризации последовательностей.

ЗАКЛЮЧЕНИЕ

Информационные системы накопили и продолжают накапливать огромные массивы первичных, «сырых» данных. Применение выявленных закономерностей позволяет решать многие задачи более эффективно и создает значительные конкурентные преимущества, что особенно ценно в условиях высокой конкуренции. Выявление закономерностей, которые содержатся в них –слишком сложная задача, если пытаться решать ее вручную.

Ответом на эти вызовы стало появление целого спектра аналитических информационных технологий, начиная с многомерного анализа, помогающего аналитику обнаружить закономерности и заканчивая методами исследования зависимостей, которые автоматически выявляют зависимости заданного вида в «сырых» данных.

Еще одной причиной распространения аналитических технологий является наличие значительного количества готовых программных средств для решения аналитических задач на разных уровнях. Возможно быстрое разовое решение поиска закономерностей в локальном варианте. Наиболее удачные решения можно переносить в компьютерной сети в среду аналитических серверов с предоставлением авторизованного доступа многим пользователям.

Количество решаемых аналитических задач все время возрастает. Это и задачи распознавания естественного языка, и распознавание образов в графической информации, изучение зависимостей в текстовых сообщениях (Text Mining), и распространение информации в компьютерных сетях. Широкое применение аналитических методов — это еще одно перспективное направление и технологии, и бизнеса.

СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

1. Паклин Н.Б. Бизнес-аналитика: от данных к знаниям : учеб. пособие / Н.Б. Паклин, В. Орешков. — 2-е изд., испр. — СПб. : Питер, 2013. — 701 с.

2. Технологии анализа данных: Data Mining, Visual Mining, Text Mining, OLAP : учеб. пособие / А.А. Барсегян [и др.]. — СПб. : БХВ-Петербург, 2008. — 375 с.

3. Чубукова И.А. Data Mining. Текст : учеб. пособие / И.А. Чубукова. — М. : БИНОМ : Лаборатория знаний, 2006. — 382 с.

4. The Cross Industrie Standard Process for Data Mining (CRISP). URL: www.crisp-dm.org.

5. Воронцов К.В. Машинное обучение (курс лекций) / К.В. Воронцов. — URL: http://www.MachineLearning.ru.

6. Каплан Р.С. Сбалансированная система показателей. От стратегии к действию / Р.С. Каплан, Д.П. Нортон. — М. : Олимп-Бизнес, 2004. — 320 с.

7. Алгоритмы интеллектуального анализа данных (службы Analysis Services — интеллектуальный анализ данных). URL: https://msdn.microsoft.com/ru-ru/library/bb522607(v=sql.120).aspx.

Учебное издание

Братищенко Владимир Владимирович

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В БИЗНЕС-АНАЛИТИКЕ

Учебное пособие

Издается в авторской редакции

ИД № 06318 от 26.11.01.

Подписано в пользование 01.04.19.

Издательство Байкальского государственного университета.
664003, г. Иркутск, ул. Ленина, 11.

http://bgu.ru.