НАРОДНАЯ УКРАИНСКАЯ АКАДЕМИЯ

С. Б. Данилевич
О. В. Дьячкова

СОВРЕМЕННЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ЭКОНОМИКЕ

БИЗНЕС-АНАЛИЗ ДАННЫХ СРЕДСТВАМИ АНАЛИТИЧЕСКОЙ ПЛАТФОРМЫ DEDUCTOR

Учебное пособие для студентов высших учебных заведений очного, заочно-дистанционного обучения и последипломного образования, обучающихся по направлениям подготовки 0305 – Экономика и предпринимательство

Харьков
Издательство НУА
2013

УДК 004.42(075.8)
ББК 32.973.26-018.2я73-1
Д18

Утверждено на заседании кафедры информационных технологий и математики Народной украинской академии. Протокол № 10 от 20. 05. 2013.

Рецензент: канд. техн. наук, доц. *К. С. Барашев*

У навчальному посібнику розглянуто актуальні завдання бізнес-аналізу даних, описано функціональні можливості аналітичної платформи Deductor, а також методи та засоби бізнес-моделювання за допомогою цієї програми. Окрім викладення теоретичного матеріалу посібник містить практичні завдання до кожної теми, що розкривають зміст курсу, питання для самоконтролю, предметний покажчик тощо.

Данилевич, Сергей Борисович.

Д 18 Современные информационные технологии в экономике. Бизнес-анализ данных средствами аналитической платформы Deductor : учеб. пособие для студентов вузов очного, заоч.-дистанц. обучения и последипломного образования, обучающихся по направлениям подготовки 0305 – Экономика и предпринимательство / С. Б. Данилевич, О. В. Дьячкова ; Нар. укр. акад., [каф. информ. технологий и математики]. – Харьков : Изд-во НУА, 2013. – 64 с.

В учебном пособии рассмотрены актуальные задачи бизнес-анализа данных, описаны функциональные возможности аналитической платформы Deductor, а также методы и средства бизнес-моделирования с помощью этой программы. Кроме изложения теоретического материала пособие содержит практические задания по каждой теме, раскрывающие содержание курса, вопросы для самоконтроля, предметный указатель и др.

УДК 004.42(075.8)
ББК 32.973.26-018.2я73-1

Содержание

Введение

На каждом предприятии одновременно реализуются множество процессов: производства, продажи, закупки, делопроизводства, управления и др. При этом требуется рационально использовать ресурсы, оптимизировать бюджет, максимально сократить расходы . Эти каждодневные операции описываются огромными объемами данных. Анализ этих данных с целью совершенствования, оптимизации процессов и системы управления в целом – весьма актуальная задача. Моделирование – эффективный способ решения этой задачи. Современному экономисту в практической деятельности необходимо уметь составлять модели реальных экономических объектов (процессов) и проводить всесторонний их анализ, выявить скрытые проблемы, выбрать и проверить пути их решения, без проведения реальных экспериментов.

Одним из средств анализа данных (в том числе поиска закономерностей в массивах данных), моделирования бизнес-задач, создания готовых аналитических решений являются аналитические платформы. Аналитическая платформа – комплексное программное обеспечение, содержащее инструменты, автоматизирующие все этапы анализа, от консолидации данных до эксплуатации моделей и интерпретации результатов.

Аналитическая платформа *Deductor* позволяет в единой программе получать законченные решения, связанные с обработкой и анализом структурированных данных с применением элементов искусственного интеллекта (*Data Mining*).

Освоение подобных средств студентами экономических специальностей позволит им в дальнейшей деятельности применять современные методы и средства для решения профессиональных задач.

Учебная версия Deductor Academic по сравнению с коммерческой полной версией обладает рядом функциональных ограничений. Так, в ней отсутствует пакетный запуск сценариев, импорт данных возможен только в текстовом формате, отсутствует ряд возможностей экспорта данных, нельзя одновременно работать с несколькими проектами и др. Однако для учебных целей эти ограничения не столь существенны. Например, преобразование в текстовый формат данных, созданных приложениями MS Office, как правило, не слишком затруднителен. Таким образом, на примере учебной бесплатной версии анали-

тической платформы Deductor можно освоить практически все возможности анализа данных полной версии.

Скачать бесплатный вариант последней версии аналитической платформы Deductor можно с сайта разработчика программы по адресу: http://www.basegroup.ru/download/deductor.

Данное пособие посвящено бизнес-анализу данных с помощью аналитической платформы Deductor. Оно предназначено для студентов, обучающихся по направлению «Экономика и предпринимательство» и изучающих дисциплину «Современные информационные технологии в экономике».

Каждый раздел пособия открывает теоретическая часть. В практической части подробно на конкретных примерах из практики рассмотрены средства программы Deductor для анализа бизнес-данных, даны задания для самостоятельной работы, контрольные вопросы для самопроверки. Необходимые термины, дополнительная информация для работы с программой вынесены в приложение.

1. СОЗДАНИЕ ПРОЕКТА DEDUCTOR STUDIO

Изучаемые понятия: аналитическая платформа, проект, источник данных, обработчик данных, визуализатор данных, сценарий, узел сценария, ветвь сценария, отчет.

Все версии аналитической платформы Deductor (коммерческие Enterprise и Professional и бесплатная Academic) содержат среди прочих 2 компонента – Warehouse и Studio.

Deductor Warehouse – это хранилище данных, консолидирующее информацию из разных источников.

Deductor Studio – это приложение для реализации всех этапов анализа данных, рабочее место аналитика. Эта программа включает все необходимые для анализа инструменты обработки: механизмы импорта данных из разнородных источников, методы очистки данных и предобработки, алгоритмы построения моделей и механизмы экспорта данных.

После стандартного запуска программы Deductor Studio появляется окно запуска, которое содержит информацию о версии программы, ограничениях, разработчике программы BaseGroup Labs (адреса сайта и электронной почты поддержки).

Затем открывается рабочее окно программы. Оно включает *Строку заголовка, Строку меню, Панель инструментов, Панель действий, Окно структуры объектов, Окно визуализации данных.*

Строка меню содержит группы команд:

- *Файл* – предназначена для работы с файлами проекта;
- *Правка* – предназначена для редактирования данных;
- *Вид* – включает команды подготовки сценариев, отчетов и выполнение подключений к источникам данных;
- *Сервис* – содержит команды настройки рабочей среды.

Меню *Файл* включает кроме стандартных команд (*Создать*, *Сохранить* и т.п.) вызов демопримеров анализа данных и работы с хранилищем данных.

Демопример анализа данных отображает примеры работы обработчиков данных. Демопример работы с хранилищем данных дает представление об импорте данных в хранилище.

При стандартной загрузке программы Deductor Studio на диске *C*: образуется папка *C:\Program Files\BaseGroup\Deductor\Manual*, которая содержит файлы *Руководство аналитика.pdf, Руководство администратора.pdf, Импорт и экспорт данных.pdf, Deductor.pdf.* Эти руко

водства описывают все возможности полной версии аналитической платформы Deductor.

Вся работа в Deductor Studio построена на создании сценариев обработки данных. *Сценарий* – это последовательность шагов, которую надо осуществить для получения необходимого результата. Шаги сценария называются его *узлами*. Цепочка последовательных узлов – *ветвь сценария*.

Сценарии создаются с помощью трех мастеров – импорта данных, экспорта данных и обработки данных.

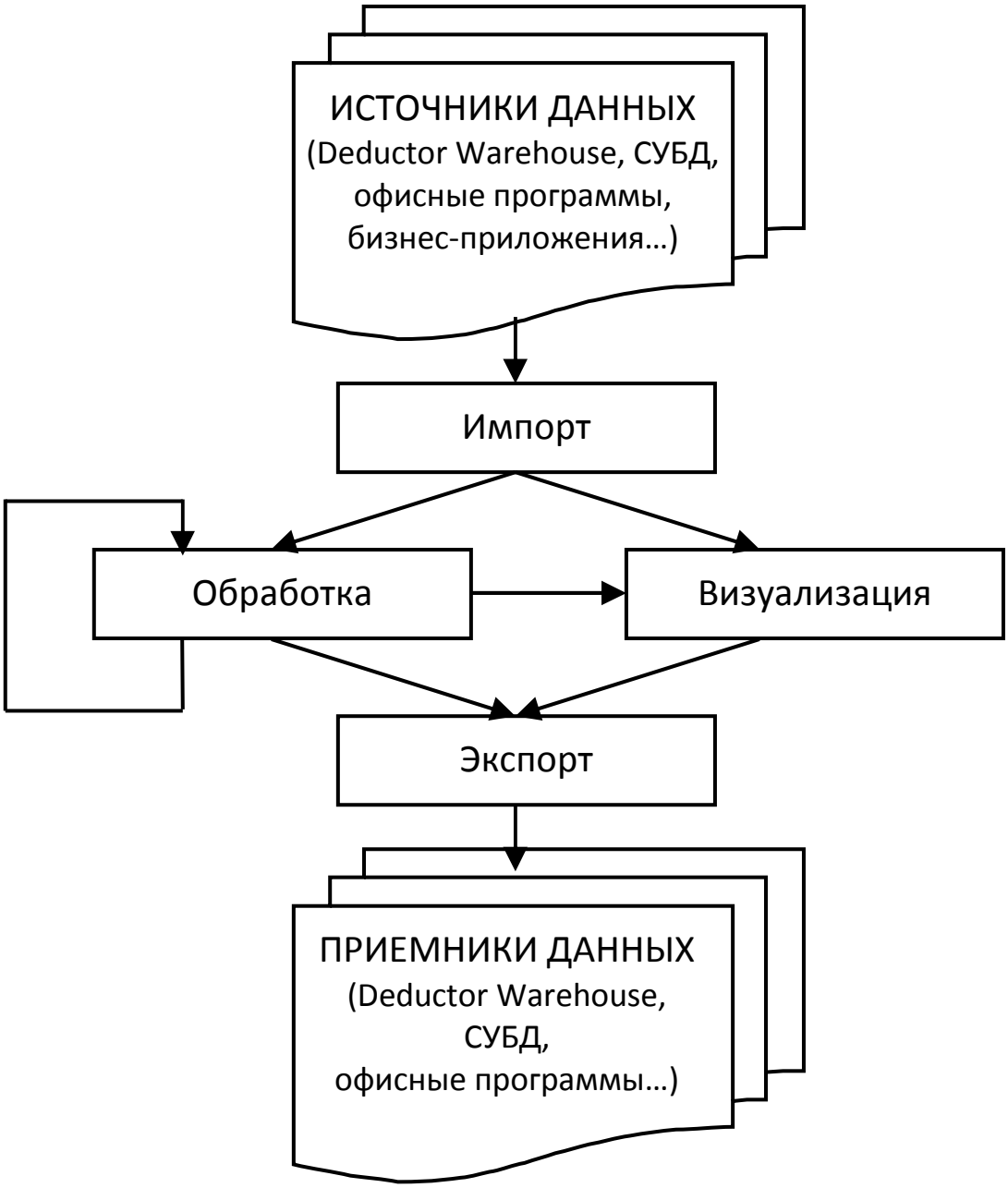

Рис. 1. Этапы анализа данных в Deductor Studio

Deductor Studio не имеет механизмов ввода данных вручную – все данные должны быть импортированы из внешних источников. Для этого предназначен *мастер импорта*. В версии Academic имеется возможность импортировать данные только из текстовых файлов. Соответственно, данные Excel, 1С, баз данных и т.п. потребуется сначала преобразовать в текстовый формат с разделителями.

Deductor Studio включает множество мастеров обработки данных (*обработчиков*). Они позволяют провести различные виды преобразований и анализа данных – например, их очистку или поиск закономерностей и т.д.

Результаты работы мастеров могут быть представлены различными способами – в табличном виде, с помощью диаграмм или более сложных *визуализаторов*. *Визуализацией* называют такой способ представления многомерных данных на двумерной плоскости (на экране), который качественно отражает основные закономерности в этих данных.

Созданные с помощью мастеров сценарии в совокупности образуют *проект*.

Для создания в процессе работы еще одного проекта Deductor Studio служит команда *Создать* в меню *Файл.*

Проект можно сохранить в *файле проекта*, который получает расширение *.ded*. Сохранение файлов стандартно (команды *Сохранить* и *Сохранить как* в меню *Файл* и кнопка *Сохранить* на панели инструментов).

Практическая работа 1. Экспорт, импорт и визуализация данных

Порядок выполнения работы:

1. Создайте папку *ПР1 Импорт и экспорт*, где будут храниться все файлы, связанные с проектом: исходные данные, аналитические сценарии, результаты. Скопируйте в нее файл исходных данных *Credit.txt*, входящий в поставку академической версии программы (папка демонстрационных примеров *C:\Program Files\BaseGroup\ Deductor\ Samples*).

2. Загрузите эти исходные данные в Deductor Studio с помощью *Мастера импорта*:

- Для запуска мастера используйте контекстное меню вкладки *Сценарии* или кнопку на ее панели инструментов либо клавишу F6.

- Строка *Имя файла* содержит абсолютный адрес импортируемого файла. В этом случае для повторного импорта потребуется, чтобы на компьютере были все папки, указанные в абсолютном адресе. Правильнее оставить относительный адрес. Тогда можно будет переносить эту папку на другой компьютер. Оставьте относительный адрес загружаемого файла *Credit.txt*.
- Выберите правильную кодировку (*DOS* или *Windows*).
- Проверьте, что первая строка является строкой заголовков столбцов. Можно начать импорт с нужной строки – не импортируя, например, заголовок таблицы.
- На следующем шаге Мастера импорта обратите внимание на разделитель целой и дробной части числа. Если в исходных данных дробная часть отделялась точкой (например, в данных программы 1С или в таблицах англоязычной версии Excel), то следует указать точку в качестве разделителя. Иначе при импорте числа не будут распознаны.
- Настройте параметры столбцов. Можно изменить метку столбца (заголовок). Например, уточнить название столбца *Количество*. Проверьте правильность автоматического определения типов данных для каждого поля из левой панели. Обратите внимание на значки в поле *Тип данных* и в левой панели окна.
- Запустите процесс импорта (кнопка *Пуск*).
- В качестве средства вывода укажите только *Таблица*.
- На последнем шаге укажите свою метку – название ветви сценария – например, *Импорт данных Credit.txt*.
- Проанализируйте данные, отобразившиеся в правой панели *Визуализация данных*.

3. Внесите изменения в узел сценария:
   - выберите в его контекстном меню команду *Настроить*;
   - на последнем шаге укажите вывод *Таблица* и *Статистика*;
   - ознакомьтесь с вкладкой *Статистика* в панели визуализации.
4. Вызовите настройку только визуализации данных для узла сценария:
   - выберите в его контекстном меню команду *Мастер визуализации*;
   - укажите вывод *Таблица, Статистика* и *Диаграмма*;
   - в ходе настройки диаграммы задайте построение точечной диаграммы по полю *Сумма кредита*, в качестве значений и подписей по оси X – поле *Срок кредита*;

- ознакомьтесь с вкладкой *Диаграмма* в панели визуализации. С помощью ее панели инструментов или контекстного меню измените тип диаграммы, отображаемые поля и пр. Отобразите панель детализации и выясните ее назначение.

5. Используя мастер визуализации, добавьте отображение кроме *Таблица, Статистика* и *Диаграмма*, средства *Гистограмма*. Настройте ее для вывода сумм кредитов.

6. Используя Мастер экспорта, выгрузите данные в виде текстового файла *Экспорт данных кредита.txt* в Вашу папку. Задайте экспорт не всех столбцов исходных данных (на Ваше усмотрение).

7. Сохраните в Вашей папке созданный Вами проект импорта и экспорта данных для возможного последующего повторения. Файл проекта назовите *Ваша_фамилия ПР1 Импорт и экспорт.* Закройте Deductor.

8. Проверьте содержимое Вашей папки *ПР1 Импорт и экспорт*, обратите внимание на форматы созданных файлов. Откройте снова созданный Вами файл проекта. Проанализируйте состояние ветви сценария и панели визуализации. Выполните повторно узлы сценария.

Заархивируйте папку *ПР1 Импорт и экспорт* в архивный файл *Ваша_фамилия ПР1 Импорт и экспорт.zip* и скопируйте его в папку $*control.*

Вопросы для самоконтроля

1. Какие компоненты содержит Deductor Studio?
2. Что такое проект в Deductor Studio?
3. Как создать новый проект?
4. Какие способы ввода данных имеются в Deductor Studio?
5. Какие мастера имеются в Deductor Studio?
6. Какие форматы данных возможны при импорте в Deductor Studio?
7. Какие шаги мастера импорта нужно пройти для импортирования данных из текстового файла?
8. Какое расширение имеет файл проекта?
9. Что сохраняет файл проекта? Сохраняются ли в нем импортированные данные?
10. В каком формате возможен экспорт данных из Deductor Studio?

2. ОЧИСТКА ДАННЫХ

Изучаемые понятия: парциальная обработка, аппроксимация, максимальное правдоподобие, аномалии, шумы, вейвлет-преобразование.

Данные – это отдельные факты, характеризующие объекты, процессы и явления предметной области, а также их свойства. Данные для анализа должны быть представительными, точными и достоверными. Однако они, как правило, поступают из разных источников и не всегда соответствуют требуемым критериям качества. Они могут содержать опечатки и орфографические ошибки, фиктивные и логически неверные значения, нарушать уникальность, стандарты на формат записи и т.п. Некачественные данные могут привести к неправильным результатам анализа.

К сожалению, Deductor Studio не имеет механизмов ввода и ручной правки данных и не все ошибки могут быть выявлены на начальной стадии. Тем не менее, *предварительная обработка данных* – необходимый шаг для обеспечения удовлетворительного результата анализа. Чем тщательнее подготовлены данные для анализа, тем в дальнейшем будет меньше проблем.

Предварительная обработка включает в себя *очистку данных* – набор методов повышения качества данных. К очистке данных относят заполнение пропусков, редактирование аномалий и ошибок, сглаживание данных, обнаружение дубликатов и противоречий и др.

Кроме очистки, могут потребоваться предварительные преобразования данных – например, снижение их размерности, устранение незначащих факторов и др.

В Deductor для решения задач очистки данных предназначен обработчик *Парциальная обработка*. Он включает несколько шагов. Каждое поле анализируемого набора обрабатывается независимо от остальных полей.

Заполнение пропусков (данные неизвестны либо их забыли внести и т.п.) должно быть первым шагом предварительной обработки данных. Для заполнения обработчик *Парциальная обработка* предлагает два способа: *Аппроксимация* и *Максимальное правдоподобие*.

Аппроксимация заполняет пропущенное значение, усредняя соседние. Используется только для упорядоченных данных. Максимальное правдоподобие подставляет значение, наиболее вероятное со статистической точки зрения. Рекомендуется использовать этот метод для неупорядоченных данных.

Редактирование аномалий. *Аномалии* (*аномальные значения*) – случайные либо редко происходящие события, которые не укладываются в общую картину процесса (резкие отклонения). Обработчик *Парциальная обработка* может отредактировать аномальные значения с разной степенью их подавления – малой, средней, большой.

Очистка от шумов. *Шумы* в данных обычно представляют собой быстрые случайные изменения значений. Шумы могут скрывать общую тенденцию, затрудняют построение модели прогноза.

Сглаживание данных необходимо, если ряд данных оказывается неравномерным, содержит большое количество мелких структур, что затрудняет поиск общих закономерностей.

Обработчик *Парциальная обработка* предлагает три механизма сглаживания и вычитания шума. Вариант *Сглаживание* позволяет указать величину полосы пропускаемых частот (т.е. отсекается все, что выше этого порога). Чем больше требуется сгладить данные, тем меньше должно быть значение полосы. Однако слишком узкая полоса может привести к потере полезной информации.

Вариант *Вычитание шума* может быть использован, если шум имеет нормальное распределение с малой дисперсией. Deductor позволяет задать степень вычитания шума: малую, среднюю и большую.

Сглаживание с помощью *вейвлет-преобразования* потребует задать глубину разложения и порядок вейвлета. *Глубина разложения* определяет масштаб отсеиваемых деталей. При увеличении глубины разложения модель отбрасывает шум все большего уровня. Однако слишком большие значения глубины разложения приводят к потере полезной информации. *Порядок вейвлета* определяет гладкость обработанного ряда данных: чем больше значение параметра, тем больше сглаживаются выбросы.

Дубликаты и противоречия в данных можно обнаружить с помощью обработчика Deductor *Дубликаты и противоречия*.

Практическая работа 2.
Очистка данных

Создайте папку *ПР2 Очистка данных* для сохранения всех файлов практической работы. Запустите Deductor, сохраните новый проект в этой папке в файле *Ваша_фамилия ПР2 Очистка данных.* Регулярно сохраняйте результаты работы.

Задание 1. Восстановление пропущенных данных.

Порядок выполнения работы:

1. Подготовьте в Excel таблицу значений функции sin(*x*), где аргумент *x* изменяется от 0 до 10 с шагом 0,05.

2. Удалите несколько значений функции (не аргумента), так чтобы были пропущенные данные.

3. Сохраните полученную таблицу в текстовом формате в Вашей папке.

4. Загрузите эти данные в Deductor. Назовите ветвь *Восстановление значений sin(x)*. Задайте отображение представлений *Таблица* и *Диаграмма*.

5. В открывшемся окне выберите пункт *Парциальная обработка* (в разделе *Очистка данных*). На втором ее шаге выберите переключатель *Аппроксимация*. На последующих шагах не задавайте никакой обработки данных. Отобразите представления *Таблица* и *Диаграмма*.

6. Сравните обработанные данные с необработанными.

Задание 2. Редактирование аномалий.

Порядок выполнения работы:

1. Скопируйте в свою папку демонстрационный файл *Anketa.txt*. Загрузите его данные в тот же проект с помощью *Мастера импорта.* Назовите ветвь *Редактирование аномалий Anketa*. Отобразите таблицу и точечную диаграмму для поля *Сумма кредита*.

2. Для созданной ветви проведите *Парциальную обработку*. При этом:

- в ее окне *Восстановление пропущенных данных* установите переключатель *Отключить*;
- в окне *Редактирование аномальных значений* выберите поле *Количество*, установите флажок *Редактирование аномальных значений*, в списке *Степень подавления* выберите пункт *Малая*.
- Сглаживание не задавайте. Выполните редактирование аномалий, нажав кнопки *Пуск* и *Далее*. Отобразите снова таблицу и точечную диаграмму для поля *Сумма кредита*.

3. Сравните обработанные и необработанные данные. Выведите в правой нижней части окна панель детализации. Выясните с ее помощью, на какие значения были заменены аномальные.

4. Для той же ветви *Редактирование аномалий Anketa* проведите еще одно редактирование аномалий, выбрав большую степень их подавления. Дайте узлам сценария понятные названия. Сравните новые величины для аномальных значений. Сделайте выводы.

Задание 3. Сглаживание и очистка от шумов.

1. Скопируйте в свою папку демонстрационный файл *dynamics_website.txt*. В том же проекте создайте еще одну ветвь сценария *Сглаживание шумов website*, импортировав данные этого файла. Отобразите при этом диаграмму (линии).

2. Проведите для этого узла трижды сглаживание и очистку от шумов всеми тремя способами (по очереди), которые предлагает обработчик *Парциальная обработка* (шаг *Спектральная обработка*). Для каждого из преобразований отобразите диаграмму (линии). Укажите настройки преобразований:

- для опции *Сглаживание* оставьте величину полосы пропускания равной 50;
- для *Вычитания шума* установите среднюю степень вычитания;
- для *Вейвлет-преобразования* оставьте значения глубины 3 и порядка 6.
- Дайте полученным трем узлам сценария осмысленные названия.
- Сравните с помощью диаграмм исходные данные и полученные после каждого из преобразований. Сделайте выводы.

Задание для самостоятельной работы.

1. Скопируйте в свою папку демонстрационный файл *Trade.txt*. Загрузите в тот же проект данные файла. Назовите ветвь *Очистка данных trade*. Отобразите диаграмму (линии).

2. Проведите парциальную обработку данных, применив к узлу сразу в один прием все три типа обработки: аппроксимацию данных, подавление аномалий большой степени и сглаживание данных с полосой пропуска 50. Отобразите диаграмму (линии). Сравните исходные и преобразованные данные.

3. Измените полученный узел. Замените сглаживание данных на вейвлет-преобразование с глубиной разложения 2 и порядком вейвлета 6. Сравните результаты.

Сохраните файл проекта. Заархивируйте папку *ПР2 Очистка данных* в архивный файл *Ваша_фамилия ПР2 Очистка данных.zip* и скопируйте его в папку *control.*

Вопросы для самоконтроля

1. Каково назначение предварительной обработки данных?
2. Каково назначение обработчика *Дубликаты и противоречия*?

3. В чем отличия предварительной обработки и очистки данных? Какие разделы обычно включает очистка данных?
4. Для чего предназначен обработчик Deductor *Парциальная обработка*?
5. Для чего необходимо восстанавливать пропущенные данные? Какие способы заполнения пропусков предлагает Deductor? Для каких данных подходит каждый из них?
6. Какие данные считаются зашумленными? Какие варианты сглаживания и вычитания шума предлагает Deductor?
7. Что такое аномалии и как они могут повлиять на результат анализа данных? Каким образом можно отредактировать аномальные значения в Deductor?

3. ХРАНИЛИЩЕ ДАННЫХ

Изучаемые понятия: хранилище данных, измерение, атрибут, факты, процесс, атрибут процесса.

Хранилище данных (ХД) – логически интегрированная система хранения данных, обеспечивающая максимально быстрый и удобный доступ к информации для проведения их анализа и поддержки принятия решений.

Аналитическая платформа Deductor включает в себя Deductor Warehouse – многомерное хранилище данных, которое консолидирует информацию из разных источников с целью аналитической обработки данных. В основе многомерного представления данных лежит их разделение на 2 группы – измерения и факты.

Измерения качественно описывают некоторый бизнес-процесс (например, наименования товаров, ФИО клиентов, названия городов и т.п.). Как правило, это текстовые данные (хотя могут быть и числовые – нпр., коды товаров, или даты). В любом случае это дискретные данные, т.е. принимающие значения из ограниченного набора.

Факты описывают бизнес-процесс количественно. Например, фактами могут служить цена товара, их количество, сумма кредита, площадь района, количество населения и т.п. Факты – величины непрерывные по своему характеру.

Кроме того, некоторые измерения могут требовать своего дополнительного, более полного описания. Для этого используются *атрибуты*. Атрибуты измерения как бы скрыты внутри него, характери-

зуют его. Например, для измерения «товар» атрибутами могут служить его цвет, вес, габариты и т.п. Можно рассматривать атрибуты как дополнительные столбцы, описывающие измерение.

Совокупность измерений, фактов и атрибутов образуют *процесс*. Процесс описывает определенное действие – продажу товаров, поступление денежных средств, производство изделий и т.д.

В Deductor Warehouse может одновременно храниться множество процессов с различным количеством измерений и фактов. В том числе процессы, имеющие общие измерения. Например, измерение «товар» может участвовать сразу в процессах «поставка», «продажа», «заказ», «отгрузка» и т.д.

Все загружаемые в Deductor Warehouse данные обязательно должны быть определены как измерение, атрибут либо факт.

Для работы с хранилищем данных из приложения Deductor Studio необходимо отобразить в его левой панели вкладку *Подключения*(команда в меню *Вид*).

Практическая работа 3. Хранилище данных

Задание. Создание и использование хранилища данных (ХД).

Создадим хранилище данных и воспользуемся им для первоначального анализа.

Порядок выполнения работы:

1. Создайте папку *ПР3 Хранилище данных.* Сохраняйте в ней все файлы практической работы.

2. Скопируйте в папку демонстрационный файл *export.txt*, данные которого будут помещены в ХД. Ознакомьтесь с его содержимым.

3. Запустите аналитическую платформу Deductor. Создайте ХД – для этого воспользуйтесь вкладкой *Подключения* и вызовите Мастер подключений. В Мастере введите:

- В поле *База данных* укажите созданную папку *ПР3 Хранилище данных* и название будущего файла *Ваша_фамилия Хранилище ПР3.* Остальные опции оставьте без изменений.
- На следующем шаге выберите последнюю (шестую) версию ХД.
- В следующем окне нажмите кнопку *Создать файл базы данных с необходимой структурой метаданных*. Должно появиться сообщение об успешном создании (пустой) базы (= файла с расширением *.gdb*).

- На последнем шаге выберите отображение сведений и метаданных.
- Присвойте ХД метку (название) *Хранилище данных Продажи*.

4. Для созданного ХД задайте его структуру (т.е. определите метаданные):
   - Запустите редактор метаданных (кнопкой на панели инструментов или контекстной командой *Редактор*).
   - В открывшемся окне нажмите кнопку *Разрешить редактировать*.

   4.1. Сначала определите все измерения. Для этого нажмите *Добавить* и пропишите название измерения (метка) – *Дата*; тип данных – *Дата/время*. Аналогично добавьте измерения *Товар, Продавец, Поставщик*.

   4.2. Создайте процесс продажи (метка *Продажа*).

   А) Развернув дерево этого процесса, добавьте для него измерения: установите курсор на его *Измерения* и выберите команду *Добавить*. Добавьте те измерения из всех доступных, по которым может потребоваться в будущем анализ процесса продаж (например, *Дата, Товар, Продавец*).

   Б) Добавьте для этого процесса факты. Установите курсор на его строку *Факты*, нажмите *Добавить* и задайте метку *Количество продаж*. Аналогично добавьте факт *Сумма продаж.*

   - Нажмите кнопку *Принять изменения*.

     Вы создали пустое хранилище данных, определили процесс, измерения и факты.

     Для размещения данных в ХД необходимы 2 шага: импорт данных из исходного файла в Deductor, а затем экспорт этих данных из Deductor в ХД.

5. Загрузите в Deductor с помощью Мастера импорта файл *export.txt*. Дайте узлу подходящее название.

6. Выгрузите эти данные в хранилище с помощью Мастера экспорта:
   - Сначала экспортируйте по очереди все измерения. При этом устанавливайте соответствия между измерением и нужным полем исходного файла данных. Каждый раз давайте узлам понятные названия (метки).
   - После этого экспортируйте процесс параллельно с фактами. При этом также укажите соответствие измерений и фактов с полями исходного файла данных. Для количества и суммы продаж задайте операцию суммирования.

Поставьте галочку *по Дате*. Таким образом, документы будут добавляться, если соответствующей даты нет, и заменяться, если такая дата есть в хранилище. Фактически будет происходить обновление по дате документа. Добавить новые данные можно очень быстро, если сохранена структура.

7. Сохраните файл проекта под названием *Ваша_фамилия ПР3 Хранилище данных.* Регулярно сохраняйте результаты работы.

Теперь можно использовать данные хранилища для анализа.

8. Выведите информацию о продажах только за определенный период:

- Сначала просмотрите узел исходных данных (таблицу) и выберите некоторый период.
- Импортируйте данные процесса *Продажи* из хранилища с помощью Мастера импорта. Выберите все измерения и факт *Сумма продаж.* На шаге определения срезов задайте желаемый фильтр по дате.
- Дайте содержательное название узлу импорта. Проанализируйте результат фильтрации данных.

9. Сохраните сценарий. Закройте и повторно откройте его и проверьте, что при повторном вызове будет выведена информация о продажах только за заданный период.

10. Добавьте еще одну ветвь сценария – подытожьте с помощью Мастера импорта суммарное и максимальное количество продаж всех товаров по каждому продавцу за каждую дату:

- Выберите для импорта из хранилища процесс *Продажи*, а для него – измерения *Дата и Продавец*. Для факта *Количество продаж* задайте операции суммы и максимума.
- Фильтрацию не устанавливайте. Дайте узлу подходящее название.
- После ознакомления с результатом перенастройте узел – добавьте еще и суммирование сумм продаж.

Сохраните и закройте файл проекта. Откройте его повторно и проверьте правильность работы.

Заархивируйте папку *ПР3 Хранилище данных* в архивный файл *Ваша_фамилия ПР3 Хранилище данных.zip* и скопируйте его в папку *control.*

Вопросы для самоконтроля

1. Что такое хранилище данных?
2. В чем различие базы данных и хранилища данных?
3. Что такое факты?
4. Что такое измерения?
5. Чем отличается атрибут процесса от измерения?
6. Могут ли в ХД одновременно сохраняться несколько процессов?
7. Могут ли несколько процессов иметь общие измерения?
8. В каком порядке требуется загружать факты измерения в ХД?

4. МНОГОМЕРНЫЙ АНАЛИЗ ДАННЫХ. OLAP-КУБ

Изучаемые понятия: OLAP-куб, кросс-таблица, кросс-диаграмма.

Для эффективного управления компанией требуется все более детализированная управленческая отчетность, чтобы один и тот же отчет мог предоставлять информацию в различных аналитических разрезах.

Для анализа удобными являются многомерные данные, которые содержат информацию о трех или более признаках для каждого объекта. Многомерные модели рассматривают данные либо как факты с соответствующими численными параметрами, либо как текстовые измерения, которые характеризуют эти факты. Для описания таких наборов данных используют понятие *многомерный куб*. По осям такого куба размещаются параметры, а в ячейках – зависящие от них данные.

Технология комплексного многомерного анализа данных и предоставления результатов этого анализа в удобной для использования форме получила название OLAP (OnLine Analytical Processing – оперативная аналитическая обработка данных).

OLAP-кубы являются инструментом для получения аналитической отчетности. Они представляют собой проекцию исходного многомерного куба данных на куб данных меньшей размерности. При этом значения ячеек объединяются. Данные в этом случае могут отображаться в виде кросс-таблиц или кросс-диаграмм.

Кросс-таблицы удобны тем, что данные можно сгруппировать произвольным образом, отфильтровать, отсортировать, переставить столбцы или строки и произвести множество других операций одним щелчком мыши. Deductor Studio позволяет при помощи этого механизма визуализации просмотреть не только исходную информацию, но и результаты обработки данных. Кросс-таблица отображает многомерные данные в виде двумерной таблицы, создавая по несколько заголовков строк и столбцов.

OLAP-технология уже сейчас стала обязательным элементом в современных и перспективных информационных системах. Особенно удобно применять ее при построении отчетов, в которых используется более одного справочника.

С помощью OLAP-куба можно получать несколько управленческих отчетов, являющиеся с точки зрения настроек в программном продукте одним отчетом, который можно просматривать разными способами.

Практическая работа 4. Визуализатор OLAP-куб

Рассмотрим работу с OLAP-кубом на примере отчета о продажах.

Порядок выполнения работы:

1. Создайте папку *ПР4 OLAP*. Скопируйте в нее файл *goods.txt*. Ознакомьтесь с содержимым файла, оцените типы данных его полей.

2. Запустите аналитическую платформу Deductor. Сохраните файл в созданной папке под названием *Ваша_фамилия ПР4 OLAP-куб.* Регулярно сохраняйте результаты работы.

3. Загрузите в Deductor файл *goods.txt*. При импорте присвойте полю *Количество* название (метку) *Количество товара*. Проверьте правильность определения типов данных полей. Дайте подходящее название узлу импорта.

Задание 1. Определение суммарных показателей продаж для каждого товара и каждого контрагента.

1. Воспользуйтесь визуализатором (Мастером визуализации) и активизируйте *OLAP анализ*:

- Обратите внимание на то, что строковые значения и даты автоматически определяются как измерения, а числовые – как факты.

- На четвертом шаге настройте OLAP-куб. Например, по строкам расположите поле *Наименование товара*, а по колонкам – поле *Поставщик* (перетащите поля мышью из левой панели в правую).
- Для фактов имеется большое количество настроек. Задайте суммирование для полей количества товара и дефектов.

В готовом кубе простым перетаскиванием можно изменять вид отчета, уточнять и детализировать информацию.

2. Отсортируйте все поля в исходном порядке.

3. Отобразите кросс-диаграмму (с помощью панели инструментов). Задайте точечное представление, выведите легенду.

4. Измените отображение фактов в кросс-таблице (для этого можно не вызывать повторно Мастер визуализации, а воспользоваться командами контекстного меню):

- Удалите суммирование количества дефектов.
- Для количества товара выведите сумму и в виде значения, и в виде процента от общих поставок товара.

5. Измените еще раз кросс-таблицу:

- Отобразите по строкам даты, а затем товары, по столбцам – поставщиков.
- Для фактов рассчитайте суммарное количество товара и среднюю цену. Пустые заголовки не отображайте.
- Разверните некоторые строки кросс-таблицы.
- Смените порядок заголовков строк: сначала даты, затем товары. Разверните некоторые строки кросс-таблицы.

6. Воспользуйтесь для исходного узла импорта Мастером обработки и попробуйте построить с его помощью кросс-таблицу, аналогичную последнему заданию. Сравните назначение, преимущества и недостатки обоих способов анализа. Не забудьте сохранить файл проекта.

Задание 2. Анализ ежемесячных продаж.

1. В исходных данных информация фиксировалась каждый день. Для анализа по месяцам воспользуемся аналитическим обработчиком *Дата–Время*. Вызовите его для узла импорта данных.

- В левой панели Мастера флажком отмечены поля, которые могут быть использованы в этом обработчике. Выберите поле даты.
- В правой панели окна Мастера выберите вариант *Год + месяц*.
- После нажатия кнопки *Готово* обратите внимание, что появился столбец *Год + месяц*, где все даты – первые числа месяца. Так, дата *21.08.2013* стала *01.08.2013*, что означает август 2013 года.

2. Проанализируйте помесячную активность поставщиков. Для этого в Мастере визуализации в OLAP-кубе задайте суммирование количества всех товаров для каждого поставщика по месяцам.

Чтобы определить, какие поставщики дают наибольшие поставки, отфильтруйте факты (контекстной командой *Селектор*). Выберите Измерение *Поставщик* и поставьте для его сумм условие *Доля от общего* 50%.

3. Сохраните сценарий, который Вы создали. В дальнейшем Вам не нужно будет повторять все операции для повторного анализа обновленных данных – достаточно воспользоваться сохраненным файлом проекта:

- Организуйте дальнейшее заполнение файла продаж – добавьте в исходный текстовый файл *goods.txt* Вашей папки (в любую позицию) 1-2 строки новых данных со свежими датами.
- Выполните для узла импорта продаж контекстную команду перечитать данные повторно. Проверьте появление новых данных в кросс-таблице этого узла. Обновите также (двойным щелком) узел преобразования даты. Проверьте появление свежих дат и в кросс-таблице этого узла.

Задание для самостоятельной работы.

1. Скопируйте в свою папку демонстрационный файл *export.txt.* Импортируйте в Deductor данные этого файла (т.е. создайте новую ветвь проекта). Визуализируйте данные в виде кросс-таблицы: подытожьте суммы продаж для каждого поставщика по каждой дате.

2. Добавьте в построенную кросс-таблицу вычисляемый факт (командой контекстного меню). Подсчитайте для каждого поставщика и каждой даты еще и число (количество) продавцов.

Сохраните файл проекта и скопируйте его в папку *control*. Заархивируйте Вашу папку *ПР4 OLAP* в архивный файл *Ваша_фамилия ПР4 OLAP.zip* и скопируйте его в папку *control.*

Вопросы для самоконтроля

1. Приведите пример многомерных данных.
2. Что такое OLAP-куб?
3. Как создать OLAP-куб в программе Deductor?
4. Какова назначение обработчика *Кросс таблица*?
5. Как построить кросс-диаграмму?

6. Как изменить настройки измерений в кросс-таблице? фактов в кросс-таблице?
7. В чем отличия обработчика и визуализатора кросс-таблиц?
8. Каким образом можно отсортировать измерения в кросс-таблице?
9. Как отфильтровать факты в кросс-таблице?

5.АВТОКОРРЕЛЯЦИОННЫЙ АНАЛИЗ

Изучаемые понятия: временной ряд, автокорреляция, коэффициенты корреляции, автокорреляционная функция**.**

Важным фактором для анализа временного ряда и прогноза является определение сезонности. В Deductor Studio инструментом, предназначенным для изучения сезонности, является *Автокорреляция*.

В процессе автокорреляционного анализа рассчитываются коэффициенты корреляции (мера взаимной зависимости) для двух значений выборки, отстоящих друг от друга на определенное количество отсчетов, называемые также *лагом*.

Совокупность коэффициентов корреляции по всем лагам представляет собой автокорреляционную функцию ряда (АКФ):

$$R(k) = corr(X(t), X(t+k)), \text{ где } k > 0 – \text{целое число (лаг).}$$

График автокорреляционной функции называется *коррелограммой*.

По поведению АКФ можно судить о характере анализируемой последовательности и наличии периодичности (например, сезонной).

Очевидно, что при $k = 0$ автокорреляционная функция будет максимальной и равной 1, т.е. значение последовательности полностью коррелированно само с собой, степень статистической взаимозависимости максимальна. Действительно, если факт появления данного значения имел место, то и соответствующая вероятность равна 1.

По мере увеличения числа лагов, т.е. увеличения расстояния между двумя значениями, для которых вычисляется коэффициент корреляции, значения АКФ будут убывать из-за уменьшения статистической взаимозависимости между этими значениями (вероятность появления одного из них все меньше влияет на вероятность появле-

ния другого). При этом, чем быстрее убывает АКФ, тем быстрее изменяется анализируемая последовательность. И наоборот, если АКФ убывает медленно, то и соответствующий процесс является относительно гладким.

Если в исходной выборке имеет место *тренд* (плавное увеличение или уменьшение значений ряда), то будет иметь место плавное изменение АКФ. При наличии сезонных колебаний в исходном наборе данных АКФ также будет иметь периодические всплески.

Для применения алгоритма автокорреляции в Deductor необходимо вызвать соответствующий обработчик, выбрать поле, для которого вычисляется АКФ. В поле *Количество отсчетов* следует указать количество отсчетов (лаг), для которых будут рассчитаны значения АКФ.

Практическая работа 5.
Расчет автокорреляции средствами Deductor

Задание. Определить наличие сезонности по месячному количеству продаж за определенный период времени.

Порядок выполнения работы:

1. Создайте папку *ПР5 Автокорреляция.* Запустите аналитическую платформу Deductor и сохраните файл проекта под названием *Ваша_фамилия ПР5 Автокорреляция.*
2. Скопируйте в Вашу папку текстовый файл *Trade.txt*. Импортируйте его данные в проект. Таблица с данными содержит столбцы *Период* – год и месяц продаж, *Количество* – количество продаж за этот месяц (внесите необходимые изменения в настройки по умолчанию параметров импорта).
3. Выберите в качестве визуализатора *Таблицу* и *Диаграмму* для просмотра исходной информации.
4. Откройте Мастер обработки и выберите в качестве обработки автокорреляцию.
5. На втором шаге Мастера настройте параметры столбцов. Укажите поле *Дата (Год + Месяц)* неиспользуемым, а поле *Количество* используемым.
6. Задайте *Количество отсчетов* равным, например, 15. Установите флажок *Включить поле отсчетов набор данных*.
7. Запустите процесс обработки.

8. Проанализируйте результаты – как в виде таблицы, так и в виде диаграммы по количеству продаж. Определите, есть ли сезонность, и если есть, то какая.

9. Вернитесь к диаграмме, построенной по исходным данным. Проверьте на ней динамику изменений продаж для найденной сезонности. Включите для этого отображение отметок – координат X и проанализируйте значения, например, для периодов с номерами 9, 21, 33 и т.д.

10. Вызовите перенастройку узла автокорреляции. Задайте количество отсчетов равным 40 периодам. Проверьте, сохраняется ли сезонность в более поздние периоды.

Заархивируйте Вашу папку *ПР5 Автокорреляция* в архивный файл *Ваша_фамилия ПР5 Автокорреляция.zip.* Скопируйте его в папку *control*.

Вопросы для самоконтроля

1. Какова цель автокорреляционного анализа?
2. В чем сходство и различие коэффициента корреляции в регрессионном анализе и коэффициента автокорреляции?
3. Что отражает коэффициент автокорреляции?
4. Каким образом производится автокорреляционный анализ в Deductor?
5. С какими целями проводится выявление сезонного эффекта?

6. ABCD-АНАЛИЗ. ТРАНСФОРМАЦИЯ ДАННЫХ

Изучаемые понятия: ABCD-анализ, принцип Парето, кумулятивная доля, кумулятивная сумма, трансформация данных, агрегация, квантование, уровень квантования, слияние данных, поля связи, внутреннее соединение, внешнее соединение (левое, правое), нормализация, кодирование.

ABCD-анализ

ABCD-анализ является одним из методов рационализации деятельности предприятия, который позволяет выделить наиболее существенные направления деятельности, направить деловую актив-

ность в сферу повышенной экономической значимости, одновременно снизить затраты в других сферах за счет устранения излишних функций и видов работ, повысить эффективность организационных и управленческих решений.

ABCD-анализ основан на принципе Парето, который гласит: 20% позиций вашего ассортимента делают 80% всех продаж компании.

Согласно ABCD-анализу все анализируемые объекты подразделяются по категориям:

- A – объекты, занимающие самые важные позиции (например, в торговом ассортименте дают 50% продаж);
- B – промежуточные позиции (30% продаж);
- C – не особо важные позиции (15% продаж).
- D (англ. *dead* – мертвый) – неликвидные позиции, которые вообще не продаются либо участвуют в оставшихся 5% продаж.

Алгоритм проведения ABCD-анализа:

1) подготовка данных для проведения анализа,
2) сортировка данных по убыванию,
3) суммирование всех данных,
4) определение доли каждой позиции в общей сумме,
5) расчет накопительного итога долей,
6) присвоение категории для каждой позиции.

Трансформация данных

Для решения большинства задач бизнес-анализа имеющиеся данные требуют предварительной подготовки не только с точки зрения их качества (очистка данных и т.п.), но и по своей организации и представлению.

Оптимизация представления и форматов данных для решения определенной задачи называют *трансформацией данных*. Виды трансформации зависят от решаемых задач и целей.

К трансформации данных относят ряд методов и алгоритмов.

Сортировка. Упорядочение данных иногда позволяет значительно упростить анализ, а то и решить некоторые задачи.

Группировка. Сначала необходимо определить, какие поля являются измерениями (представляют данные качественно), а какие – фактами (представляют данные количественно). В результате будут объединены все записи, имеющие одинаковые значения выбранного измерения(-ий), а соответствующие факты агрегированы. Функции агрегации – сумма, среднее, количество, максимум, минимум, медиана, первый, последний и др.

Настройка набора данных – изменение имен, меток, типов и назначения полей и т.п.;

Квантование – членение диапазона числовых значений на несколько интервалов. Каждому интервалу присваивается номер (иногда метка), называемый *уровнем квантования*, и значения поля исходной таблицы заменяются на значения уровня. Используется для преобразования непрерывных данных в дискретные, для снижения размерности данных.

Слияние – объединение двух таблиц или их частей в одну. Первая (*исходная*, или *входная*, к ней будут добавлены данные) и вторая (*связываемая*) таблицы должны иметь одно или несколько одинаковых полей (столбцов) – т. наз. *поля связи*. Виды слияния:

объединение – записи (строки) второй таблицы будут приписаны снизу к первой;

внутреннее соединение – будут оставлены только те записи, в которых совпадают значения указанных полей связи обеих таблиц;

внешнее соединение. *Левое внешнее соединение* – поля второй таблицы будут добавлены к первой, но заполнены будут не для всех записей, а только для тех, которые содержат одинаковые значения полей связи обеих таблиц. При *правом внешнем соединении* – поля из первой таблицы добавляются ко второй. *Полное внешнее соединение* создает итоговую таблицу, содержащую все строки и все поля обеих таблиц.

Нормализация – приведение значений данных к единому масштабу (масштабирование). Особенно важно перед применением алгоритмов Data Mining.

Кодирование – преобразование категориальных данных в числовые. Необходимо перед использованием нейронных сетей, деревьев решений, регрессии, которые требуют данные только в числовом виде.

Вычисление значений. Иногда требуются дополнительные значения (поля), которые могут быть рассчитаны на основании уже имеющихся. Для этого в Deductor имеется *Калькулятор*, который позволяет произвести манипуляции над данными (включая использование встроенных функций).

Иногда к методам трансформации относят фильтрацию данных, хотя обычно ее используют еще на стадии предварительной обработки данных.

Практическая работа 6.
ABCD-анализ средствами Deductor

Порядок выполнения работы:

1. Создайте папку *ПР6 ABCD-анализ.* Запустите аналитическую платформу Deductor и сохраните файл проекта под названием *Ваша_фамилия ПР6 ABCD-анализ.*

2. Скопируйте в Вашу папку текстовый файл *Продажи*. Импортируйте его данные в проект.

3. Добавьте (с помощью обработчика) поле *Год + месяц*.

4. Отсортируйте данные по полю *Год + месяц* по возрастанию (используйте соответствующий обработчик).

5. Представьте данные только для первого месяца из имеющихся данных, воспользуйтесь *Фильтром* в Мастере обработки.

6. Произведите группировку данных. Для этого потребуется указать, какие поля являются измерениями, а какие – фактами. По умолчанию Deductor предлагает сгруппировать данные по месяцам, контрагентам и товарам и рассчитать для них суммы продаж и количества. Оставьте эти настройки.

Проанализируйте результат. Сравните количество записей в таблице этого и предыдущего узла – сколько записей удалось сгруппировать?

7. Проведите для узла *Фильтр* другую группировку – сгруппируйте полученные данные теперь по всем товарам за выбранный месяц (поля дат и контрагентов – неиспользуемые; суммирование тех же фактов).

8. Проведите для узла *Фильтр* еще одну группировку – оставьте только дату (измерение) и сумму продаж (факт). Должна получиться только одна строка – общая сумма продаж за месяц.

9. Используем найденную общую сумму для расчета доли продаж каждого товара. Для этого добавим столбец с этой суммой за месяц в таблицу с отфильтрованными данными:

- Воспользуйтесь для узла *Фильтр* обработчиком *Слияние с узлом*. Укажите в качестве узла связи только что созданный узел группировки. Выберите *Левое внешнее соединение*.
- Выберите, по каким полям будет производиться связь (общее поле в связываемых таблицах одно – *Дата (Год + Месяц)*.
- Снимите флажки с тех полей, которые не нужны в итоговой таблице для расчета долей продаж товаров (поля дат, контрагентов).
- Назовите добавляемое расчетное поле *Сумма за месяц*.

10. Рассчитайте долю каждого товара с помощью обработчика *Калькулятор*. Назовите вычисляемое поле *Доля*. Она будет рассчитываться как поле *Сумма*, деленная на *Сумма за месяц*.

11. Отсортируйте данные поля *Доля* по убыванию, используя Мастер обработки, чтобы все действия записывались в сценарий.

12. Рассчитайте кумулятивную долю, то есть с накоплением (она вычисляется как сумма долей за все предшествующие периоды). В *Калькуляторе* есть специальная функция (в категории математических): *CumulativeSum*(""). В кавычки вставьте поле *Доля*. Назовите вычисляемое поле *Кумулятивная доля*.

13. Добавьте в таблицу еще одно вычисляемое поле – *Категория* – для проведения ABCD-анализа:

- Выберите строковый тип поля.
- Вставьте логическую функцию «если» (прочтите в описании, какая из двух подходит – IF или IFF). Задайте в ней условие: если кумулятивная доля до 50%, – категория A, следующие 30% – категория B, остальные товары – категория C.
- Очевидно, выражение будет иметь вид вроде:
  IF(EXPR_1<=0.50;"A"; IF(EXPR_1>=0.8;"C";"B")).

14. В сценарии найдите фильтр, с помощью которого отфильтровали именно этот месяц, и выберите другой. Проверьте работу последующих узлов сценария.

15. Проведите OLAP-анализ полученных результатов с помощью Мастера визуализации.

16. Экспортируйте полученный результат в текстовый файл в Вашей папке. Назовите его *Результат ABCD-анализа* и включите в него название товара, его кумулятивную долю и категорию.

Сохраните файл проекта. Заархивируйте Вашу папку *ПР6 ABCD-анализ* в архивный файл *Ваша_фамилия ПР6 ABCD-анализ.zip.* Скопируйте его в папку *control*.

Вопросы для самоконтроля

1. Какова цель ABCD-анализа?
2. В чем заключается принцип Парето?
3. Каковы критерии разделения объектов анализа по категориям?
4. Каковы этапы проведения ABCD-анализа?
5. В чем отличия функций IF и IFF?
6. Какие виды внешних соединений возможны в обработчике слияния данных? В чем их отличия?

7. XYZ-АНАЛИЗ

Изучаемые понятия: XYZ-анализ, среднее арифметическое, коэффициент вариации, среднеквадратическое отклонение.

XYZ-анализ позволяет классифицировать ресурсы (например, товарные запасы) в зависимости от стабильности их потребления / стабильности продаж. Применение XYZ-анализа показывает, насколько устойчив спрос на тот или иной товар.

Категория X — это группа товарных запасов, которая характеризуется стабильной величиной потребления / продаж и высокой точностью прогноза. В группу X попадают наиболее стабильно продаваемые товары (обычно не более 10% расхождения в динамике продаж).

К *категории Y* относятся товарные запасы, потребность в которых характеризуется известными тенденциями (например, сезонными колебаниями) и средними возможностями их прогнозирования.

К *категории Z* относятся товарные запасы, которые характеризуются нестабильным спросом / нерегулярными продажами и практически не поддаются прогнозированию.

Группа Z не поддается вообще никакому предсказанию (хаотичные продажи).

Как правило, используется следующее распределение товарных запасов по методу XYZ:

- *группа X* – объекты, коэффициент вариации по которым не превышает 10%,
- *группа Y* – объекты, коэффициент вариации по которым составляет 10–25%,
- *группа Z* – объекты, коэффициент вариации по которым превышает 25%,

хотя встречаются и другие.

Коэффициент вариации

Коэффициент вариации характеризует относительную меру отклонения измеренных значений от среднеарифметического:

Формула коэффициента вариации

$$V = \frac{\sigma}{a} \times 100\%,$$

где V – коэффициент вариации,

σ – среднеквадратическое отклонение,

a – среднее арифметическое.

Чем больше значение коэффициента вариации, тем относительно больший разброс и меньшая выравненность исследуемых значений.

Если коэффициент вариации меньше 10%, то изменчивость вариационного ряда принято считать незначительной, от 10% до 20% относится к средней, от 20% до 33% – к значительной, а если коэффициент вариации превышает 33%, то это говорит о неоднородности информации и необходимости исключения самых больших и самых маленьких значений.

Практическая работа 7.
XYZ-анализ средствами Deductor

Порядок выполнения работы:

1. Создайте папку под именем *ПР7 XYZ-анализ*.

2. Запустите аналитическую платформу Deductor. Создайте новый проект под именем *Ваша_фамилия ПР7 XYZ-анализ* и сохраните его в созданной папке. В эту же папку скопируйте текстовый файл *Продажи* и импортируйте в Deductor.

3. Добавьте в таблицу исходных данных поле *Год + месяц*.

4. Отсортируйте данные по полю *Год + месяц* по возрастанию.

5. Оставьте данные только для первого месяца из имеющихся, воспользовавшись *Фильтром* в Мастере обработки.

6. Сделайте две группировки:

- средние продажи за месяц каждого товара,
- стандартное отклонение продаж за месяц каждого товара.

7. Мастером обработки *Слияние с узлом* произведите внешнее левое соединение узла *Средние продажи за месяц* с узлом *Стандартное отклонение*. Дайте корректные названия результирующим столбцам таблицы.

8. С помощью *Калькулятора* вычислите коэффициент вариации в %, округлив до целого результат деления стандартного отклонения на среднее. Используйте функцию округления Round, например:

Round(Quantity_j / Quantity * 100; 0)

9. Мастером обработки *Настройка набора данных* установите для столбцов товара и стандартного отклонениея назначение *Используемое*, а для столбцов средних продаж за месяц и коэффициента вариации – *Информационное*.

10. С помощью Мастера обработки *Сортировка* упорядочьте поле коэффициента вариации по возрастанию.

11. Для проведения XYZ-группировки определите с помощью *Калькулятора* значение категории (группы): до 10% – категория X, свыше 25% – категория Z, остальные – категория Y. Формула будет иметь похожий вид:

IF (Variation<10; "X"; IF (Variation>25; "Z"; "Y"))

Не забудьте, что результат будет иметь строковый тип данных.

12. Мастером обработки *Настройка набора данных* установите для столбцов *Товар* и *Категория* назначение *Используемое*, а для остальных – *Неиспользуемое*.

13. В результате получим таблицу с двумя столбцами *Товар* и *Категория*, где каждый товар отнесен к соответствующей категории *X, Y* или *Z*.

14. Экспортируйте полученную таблицу в текстовый файл в папку *ПР7 XYZ-анализ*.

Сохраните файл проекта. Заархивируйте Вашу папку *ПР7 XYZ-анализ* в архивный файл *Ваша_фамилия ПР7 XYZ-анализ.zip.* Скопируйте его в папку *control*.

Вопросы для самоконтроля

1. Какова цель XYZ -анализа?
2. Каковы критерии разделения объектов анализа по категориям?
3. Каковы этапы проведения XYZ -анализа?
4. Чем характеризуются товары, попавшие в группу X?
5. Возможен ли XYZ-анализ средствами Excel?

8. ПРОГНОЗИРОВАНИЕ ВРЕМЕННЫХ РЯДОВ. НЕЙРОННЫЕ СЕТИ

Изучаемые понятия: Data Mining, самообучающиеся алгоритмы, машинное обучение; нейронная сеть, внутренний слой, обучение сети, эпоха обучения, обучающее множество, тестовое множество; период прогнозирования, горизонт прогнозирования, интервал прогнозирования; скользящее окно, глубина погружения.

Data Mining

Многие актуальные задачи бизнес-анализа относятся к методологии Data Mining (*англ*. добыча знаний, извлечение знаний, интеллектуальный анализ данных). Цель *Data Mining* – обнаружение в «сырых» данных ранее неизвестных, нетривиальных, практически полезных и доступных для интерпретации знаний, необходимых для принятия решений [2].

К Data Mining относят обычно задачи классификации, регрессии, кластеризации, поиска ассоциаций и последовательностей.

Классификация – это отнесение объектов к одному из заранее известных классов.

Регрессия – установление зависимости выходных непрерывных значений от исходных значений.

Кластеризация – разбиение объектов на группы (*кластеры*) похожих объектов.

Поиск ассоциаций – нахождение зависимости между событиями: что из одного события X следует событие Y.

Последовательные шаблоны – установление закономерностей между связанными во времени событиями.

Анализ отклонений – нахождение нетипичных, нехарактерных событий.

Для решения этих задач требуется найти закономерности в больших объемах данных. С этой целью часто используют методы статистического анализа, однако они не всегда применимы и не позволяют найти сложные закономерности.

Существует целый класс адаптивных алгоритмов, которые позволяют обнаружить скрытые сложные закономерности. Их основная черта – возможность обучения системы на исходных данных. При изменении ситуации достаточно для решения задачи переобучить заново систему на новом наборе данных. Поэтому такие алгоритмы называют *самообучающимися*, или *машинным обучением*.

К самообучающимся алгоритмам относятся нейронные сети, самоорганизующиеся карты, деревья решений и др.

Нейронные сети

Нейронные сети в искусственном интеллекте – это вычислительные структуры, которые имитируют в упрощенном виде некоторые аспекты процесса человеческого мышления. Таким образом, нейросети представляют собой упрощенные модели биологических

нейронных сетей. У нейронных сетей много важных свойств, но ключевое из них – это способность к обучению.

В основу построения нейросети заложен некий элементарный преобразователь, называемый *нейроном (искусственным нейроном)*.

Нейронная сеть имеет несколько слоев: 1) входной, 2) один или несколько скрытых внутренних и 3) выходной. Каждый слой может содержать несколько нейронов, связанных с другими нейронами (связи называются *весами*). Входной и выходной слои предназначены для связи с исходными и выходными данными соответственно.

Считается, что задачу любой сложности можно решить при помощи двухслойной нейросети (т.е. имеющей 2 внутренних слоя). Для решения многих задач вполне подойдет однослойная нейронная сеть.

Имитация некоторого процесса с помощью нейросети заключается в следующем. Любое изменение входов нейросети приводит к однозначно определенному изменению ее выходов – необходимо научить сеть находить такую зависимость выходных полей от входных. Для этого организуют предварительно процесс *обучения* нейросети.

Для обучения нейросети готовят таблицу данных с известными и входными, и выходными значениями (очищенными и нормализованными). (В Deductor реализована автоматическая нормализация данных перед построением нейросети.) Исходные данные разбивают на 2 части – *обучающую* и *тестовую*, На обучающем множестве данных будет идти поиск закономерностей, а на тестовом – проверка результата обучения сети.

Существует несколько алгоритмов обучения нейросети. Любой из них присваивает всем нейронам сети сначала случайные значения. В процессе обучения эти веса принимают определенные значения.

Обучение считают успешным, если процент распознанных примеров на обучающем и тестовом множествах близок к 100%.

Прогнозирование

Одной из часто встречающихся задач является прогнозирование результата на определенное время вперед на основании данных за прошедшее время. В Deductor для этого есть инструмент *Прогнозирование*, который появляется в Мастере обработки только после построения какой-либо модели прогноза: нейросети, линейной регрессии и т.д.

Параметры прогнозирования:

- период прогнозирования – это основная единица времени, на которую делается прогноз;
- горизонт прогнозирования – это число периодов в будущем, для которых составляется прогноз;
- интервал прогнозирования – частота, с которой делается новый прогноз.

Например, для прогноза, составленного на каждый из 10 последующих дней: период прогнозирования – сутки, а горизонт прогнозирования – 10 суток.

С увеличением горизонта прогнозирования точность прогноза обычно снижается.

Преобразование данных к скользящему окну

Когда требуется прогнозировать временной ряд, тем более если налицо его периодичность (сезонность), то лучшего результата можно добиться, учитывая значения факторов не только в данный момент времени, но и, например, за аналогичный период прошлого года. Такую возможность можно получить после трансформации данных к скользящему окну. *Скользящее окно* – метод отбора данных, при котором выделяется только некоторый непрерывный участок исходных данных, а затем многократно смещают этот участок на некоторый интервал.

Например, при сезонности продаж с периодом 12 месяцев, для прогнозирования продаж на месяц вперед можно в качестве входного фактора указать не только значение количества продаж за предыдущий месяц, но и за месяц ровно год назад. Обработка методом скользящего окна создает новые столбцы путем сдвига данных исходного столбца вниз и вверх (глубина погружения, горизонт прогноза).

Преобразование скользящего окна имеет два параметра: *глубина погружения* – количество «прошлых» отсчетов, попадающих в окно, и *горизонт прогнозирования* – количество «будущих» отсчетов.

Практическая работа 8.
Нейронные сети. Прогнозирование

Задание 1. Обучение нейронных сетей умножению.

Рассмотрим прогнозирование с помощью нейронных сетей на примере прогнозирования результата умножения двух чисел.

Порядок выполнения работы:

1. Создайте папку под именем *ПР8 Нейронные сети*. Создайте файл Excel, содержащий 3 столбца: два – с произвольными числами (используйте функцию СЛЧИС и масштабирование), третий – их произведения. Транслируйте формулы на 150-200 строк. Дайте названия столбцам: *Произведение* – результат умножения множителей *Аргумент1*, *Аргумент2.* Сохраните данные в файле *multi.txt* в Вашей папке.

2. Запустите аналитическую платформу Deductor. Создайте новый проект под именем *Ваша_фамилия ПР8 Нейронные сети* и сохраните его в созданной папке. Импортируйте данные файла *multi.txt* в проект.

3. Выберите для узла импорта обработчик *Нейросеть*.

- Укажите назначение полей: *Аргумент1*, *Аргумент2* – входные, а поле *Произведение* – выходное.
- На следующем шаге настройте разбиение исходного множества данных на обучающее, тестовое и валидационное. Выберите способ разбиения исходного множества данных *Случайно*.
- На следующем шаге укажите количество нейронов в скрытом слое – 1, остальные значения можно оставить по умолчанию.
- Следующий шаг предлагает выбрать алгоритм обучения и его параметры. Оставьте предустановленные значения.
- На следующем шаге настройте условия остановки обучения. Укажите, что следует считать пример распознанным, если ошибка меньше 0,005, и также укажите условие остановки обучения при достижении эпохи 10000.
- Запустите процесс обучения и пронаблюдайте в его ходе величину ошибки, а также процент распознанных примеров. Параметр *Частота обновления* отвечает за то, через какое количество эпох обучения выводится данная информация.

5. После обучения сети выберите в качестве визуализаторов *Диаграмму, Диаграмму рассеяния, Граф нейросети, Что–если*. На обычной диаграмме (точечной) отобразите два поля – *Произведение* и *Произведение_OUT* (прогноз).

6. Сравните эталонные данные с прогнозируемыми.

7. Проведите эксперимент с помощью визуализатора *Что–если*, введя любые значения множителей *Аргумент1* и *Аргумент2* и рассчитав результат их произведения. Помните, что значения должны быть из того же диапазона, что и в обучающем множестве.

Задание 2. Прогноз продаж с помощью нейронных сетей.

Имеются данные о ежемесячном количестве проданного товара за несколько лет. Необходимо на основании этих данных спрогнозировать, какое количество товара будет продано через неделю, через две недели.

Порядок выполнения работы:

1. Скопируйте в свою папку демонстрационный текстовый файл *Trade.txt* и импортируйте его данные в Deductor (укажите правильный разделитель дробной и целой части).

2. Отобразите диаграмму количества продаж. На ней видно, что данные содержат аномалии (выбросы) и шумы, за которыми трудно разглядеть тенденцию.

3. Выполните удаление аномалий и сглаживание. Выявите тенденцию.

4. Анализ исходных данных (диаграммы) позволяет предположить ежегодную периодичность продаж. Сформируйте соответствующее скользящее окно данных:

- Для этого запустите Мастер обработки, выберите в качестве обработчика *Скользящее окно*.
- Назначьте поле *Количество* используемым. Выберите глубину погружения 12 (т.е. для каждого месяца будут отображены данные предыдущих 12 месяцев). Горизонт прогнозирования – 0 (прогноз не строим).

5. Откройте Мастер обработки и выберите в нем *Нейронную сеть*:

- В качестве входных факторов используйте *Количество – 12*, *Количество – 11* (количество товара 12 и 11 месяцев назад относительно прогнозируемого месяца), а также *Количество – 2* и *Количество - 1* – данные за два предыдущих месяца. В качестве выходного поля укажите столбец *Количество*. Остальные поля сделайте информационными.
- На следующем шаге укажите разбиение тестового и обучающего множеств. Количество слоев и нейронов в нейросети, а также другие параметры настройки оставьте без изменения.
- Для визуализации выберите диаграмму и настройте отображение полей *Количество* и *Количество_OUT* – реального и рассчитанного значений.

6. Откройте Мастер обработки и выберите появившийся обработчик *Прогнозирование*.

- На втором шаге Мастера предлагается настроить связи столбцов для прогнозирования временного ряда: откуда брать данные для столбца при очередном шаге прогноза.
- Укажите горизонт прогноза равный трем.
- Выберите отображение *Таблицы* и *Диаграммы прогноза*. В Мастере настройки диаграммы укажите в качестве отображаемого столбец *Количество*, а в качестве подписей по оси X – столбец *Шаг прогноза*.

7. Предположим, прогноз решено делать на основании только данных о последних 3 месяцах продаж.

- Скопируйте узел (ветвь) *Скользящее окно*. Перенастройте копию (и переименуйте!) на глубину погружения 3 (*Количество* [*–3:0*]).
- Перенастройте (и переименуйте) следующий за ним узел нейросети.
- Сравните результат прогноза с предыдущим. Измените тип обеих диаграмм прогноза на точечную, проанализируйте числовые значения прогноза. Сделайте выводы.

Задание 3. Прогноз посещаемости сайта.

Построим прогноз посещаемости сайта.

Порядок выполнения работы:

1. Скопируйте в свою папку файл *dynamics_website.txt* и выполните импорт его данных в Deductor.

2. Отсортируйте данные. Отобразите точечную диаграмму. Выявите тенденцию.

3. Проведите необходимую парциальную обработку данных (для сглаживания выберите полосу пропускания 20-30).

4. Настройте скользящее окно за 2 последних месяца.

5. Настройте нейронную сеть. После ее обучения установите способы отображения: *Граф нейросети, Что–если, Диаграмма рассеяния, Таблица, Диаграмма*. Проанализируйте полученные данные.

6. В Мастере обработке выберите *Прогнозирование*. Установите горизонт прогнозирования на 2 месяца вперед.

7. Проанализируйте полученные данные.

Сохраните файл проекта. Заархивируйте Вашу папку *ПР8 Нейронные сети* в архивный файл *Ваша_фамилия ПР8 Нейронные сети.zip.* Скопируйте его в папку *control*.

Вопросы для самоконтроля

1. Что означает термин Data Mining?
2. Какие задачи относят к Data Mining?
3. Какие алгоритмы называют самообучающимися (машинным обучением)?
4. Что такое нейронная сеть? Каково ее назначение?
5. Что означает обучение нейронной сети?
6. Какие слои имеются в любой нейронной сети? Сколько внутренних слоев нейронной сети требуется для решения большинства задач?
7. Что такое обучающее и тестовое множество? Каким образом исходные данные могут быть разбиты на тестовое и обучающее множества?
8. Когда появляется инструмент *Прогнозирование* в программе Deductor?
9. Что показывает период прогнозирования?
10. Что показывает горизонт прогнозирования?
11. Что показывает интервал прогнозирования?
12. Что такое глубина погружения?
13. Каково назначение обработчика *Преобразование данных к скользящему окну*?

9. ПРОГНОЗИРОВАНИЕ С ПОМОЩЬЮ ЛИНЕЙНОЙ РЕГРЕССИИ

Изучаемые понятия: регрессия, линейная регрессия, диаграмма рассеяния.

Регрессия – это установление зависимости непрерывных выходных данных от входных. Имея построенную регрессионную модель, можно с ее помощью также решать задачи прогнозирования.

Линейная регрессия используется, когда предполагается, что зависимость между входными факторами и результатом линейная. Достоинством ее можно назвать быстроту обработки входных данных и простоту интерпретации полученных результатов.

В Deductor имеется обработчик *Линейная регрессия*, в результате работы которого строится линейная модель данных. Подготовка и настройка обучающей выборки, настройка назначений полей и их

нормализация для этого компонента производятся так же, как и для нейронных сетей.

Для визуальной оценки качества построенной регрессионной модели используют *диаграмму рассеяния*. Она отображает разброс между эталонными значениями выходного поля и значениями, рассчитанными моделью.

Нужно помнить, что линейная регрессия предназначена для поиска линейных зависимостей в данных. Если же зависимости нелинейные, то модель будет плохого качества. Это будет сразу видно на диаграмме рассеяния – прогнозные значения величины будут сильно разбросаны относительно действительных значений. В этом случае нужно использовать более мощные алгоритмы, например нейронные сети.

Необходимо также учитывать, что регрессия находит зависимость для того диапазона данных, который содержится в исходной таблице, – при выходе за его пределы результаты могут оказаться непредсказуемыми.

Практическая работа 9.
Линейная регрессия. Прогнозирование

Задание. Прогноз продаж с помощью линейной регрессии.

Спрогнозируйте объем продаж в ближайшие 3 месяца, построив линейную регрессию для имеющихся данных.

Порядок выполнения работы:

1. Создайте папку *ПР9 Регрессия.* Скопируйте в нее и импортируйте в Deductor данные из демонстрационного файла *Trade.txt*. Сохраните проект в файле *Ваша_фамилия ПР9 Регрессия*.

2. Удалите аномалии и произведите сглаживание. Для обоих узлов (импорта и предобработки) отобразите диаграммы. Сравните их.

3. Настройте скользящее окно, если известно, что для прогноза будут использованы данные трех предыдущих месяцев.

4. Запустите Мастер обработки и выберите в качестве обработки данных *Линейную регрессию*.

- Задайте назначение исходных столбцов. В предположении, что на прогноз влияет информация трех предыдущих месяцев, укажите входными столбцами поля: *Количество – 3, Количество – 2* и *Количество – 1*, а выходным полем – столбец *Количество*.

- На втором шаге Мастера можно настроить обучающее и тестовое множества и способ разделения исходного множества данных. На третьем шаге – ограничить диапазон входных данных.
- Отобразите диаграмму рассеяния и диаграмму полей *Количество* и *Количество_OUT* – реального и рассчитанного значений. Проанализируйте их.

5. После выполнения процесса обучения для построения прогноза запустите обработчик *Прогнозирование*.

- На первом шаге Мастера укажите связь между столбцами и горизонт прогноза 3.
- Далее выберите отображение *Таблицы* и *Диаграммы прогноза* (для столбца *Количество*, подписи по оси X – *Шаг прогноза*).

6. Проанализируйте результаты.

7. Сравните аналогичный результат прогноза по данным последних трех месяцев в предыдущей работе *ПР8 Нейронные сети*. Сделайте выводы.

8. Самостоятельно проведите аналогичное построение линейной регрессии и прогнозирование на основании данных за 12 и 11, 2 и 1 месяц назад относительно прогнозируемого месяца. Можно для этого скопировать и перенастроить узел скользящего окна, удалить все последующие узлы и создать необходимые.

9. Сравните результат с аналогичным прогнозом в предыдущей работе *ПР8 Нейронные сети*.

Сделайте выводы.

Сохраните файл проекта. Заархивируйте Вашу папку *ПР9 Регрессия* в архивный файл *Ваша_фамилия ПР9 Регрессия.zip.* Скопируйте его в папку *control*.

Вопросы для самоконтроля

1. Что такое регрессия?
2. В каких случаях используется линейная регрессия?
3. Каковы достоинства и недостатки метода линейной регрессии?
4. Для чего предназначена диаграмма рассеяния?
5. Каковы требования к данным, для которых будет использована построенная модель линейной регрессии?

10. ПРОГНОЗИРОВАНИЕ С ПОМОЩЬЮ ПОСТРОЕНИЯ ПОЛЬЗОВАТЕЛЬСКИХ МОДЕЛЕЙ

Изучаемые понятия: обработчик *Пользовательские модели*, таблица сопряженности.

В ситуации, когда сложные модели, основанные на линейной регрессии и нейронных сетях, не могут описать предметную область, есть возможность строить модели, основанные на экспертных оценках. Для этого в Deductor включен обработчик *Пользовательские модели*, который позволяет на основании исходных данных и пользовательских формул строить собственную модель.

Пользовательские формулы модели задаются в окне *Калькулятора*. Есть возможность использовать стандартные модели, встроенные в Deductor, такие как скользящее среднее, авторегрессия, ARMA, ARIMA и др. Для них достаточно указать необходимые коэффициенты. Чтобы ввести формулу стандартной модели, нужно с помощью кнопки *Функция* вызвать окно выбора функции, в разделе *Модели* выбрать стандартную модель и добавить нужные коэффициенты.

При построении пользовательской модели можно использовать стандартные математические, статистические, строковые и др. функции. Для выходных полей доступны визуализаторы группы Data Mining, предназначенные для оценки качества модели – *диаграмма рассеяния* и *таблица сопряженности*. Для пользовательских вычисляемых полей визуализаторы группы Data Mining недоступны.

Создание модели с использованием выходных полей аналогично обучению нейросети с учителем. Диаграмма рассеяния и таблица сопряженности позволяют оценить степень близости эталонных и рассчитанных выходов, а следовательно, качество модели. При добавлении же новых полей эталонных выходов нет, есть только рассчитанные моделью, и оценка качества модели с помощью подобных инструментов невозможна.

Для пользовательских моделей доступны визуализаторы из группы Data Mining: *Диаграмма рассеяния*, *«Что – если»* и *Таблица сопряженности*.

Первый из них позволяет оценить качество построенной модели по тому, насколько точно она описывает имеющиеся данные, если выходное поле является непрерывным.

Второй дает возможность анализировать модель по принципу «что будет, если» и позволяет исследовать ее поведение при подаче на вход тех или иных данных.

С помощью таблицы сопряженности можно сравнить значения дискретных выходных полей, рассчитанные моделью, с выходными значениями полей исходной выборки и определить, насколько точно выходы модели соответствуют эталонным значениям.

Практическая работа 10. Прогнозирование с помощью пользовательских моделей

Задание. Построение пользовательской модели.

Постройте прогноз продаж на последующие 3 периода, если каждый месяц наблюдается постоянный прирост объема продаж на 160000 ед. и спад продаж на 12% от аналогичного периода прошлого года, а также прирост в 2% по сравнению с предыдущим месяцем:

Прогноз = ОбъемПредыдущегоМесяца * 1.02 + 160000 – ОбъемМесяцаГодНазад * 0.12

Порядок выполнения работы:

1. Создайте папку *ПР10 Пользовательская модель.* Скопируйте в нее и импортируйте в Deductor данные из демонстрационного файла *Trade.txt*. Сохраните проект в файле *Ваша_фамилия ПР10 Пользовательская модель*.
2. Удалите аномалии и произведите сглаживание.
3. Настройте скользящее окно.
4. Запустите обработчик данных *Пользовательская модель*.

- Настройте поля исходных данных.
- Напишите формулу получения прогноза похожего вида:

  160000 – 0.12 * COL2B12 + 1.02 * COL2B1

  (где COL2B12 и COL2B1 – соответственно имена полей *Количество – 12* и *Количество - 1*).
- Отобразите диаграмму рассеяния и таблицу сопряженности. Проанализируйте их.

5. На основании полученной модели проведите прогнозирование на последующие 3 периода.

Сохраните файл проекта. Заархивируйте Вашу папку *ПР10 Пользовательская модель* в архивный файл *Ваша_фамилия ПР10 Пользовательская модель.zip.* Скопируйте его в папку *control*.

Вопросы для самоконтроля

1. В каких случаях применяется пользовательская модель?
2. Как в программе Deductor реализуется создание пользовательской модели?
3. Какие визуализаторы доступны для пользовательских моделей?
4. Как с помощью диаграммы рассеяния оценить качество модели?
5. Каким образом провести оценку качества модели с помощью таблицы сопряженности?
6. Какие возможности предоставляет визуализатор «Что – если» для оценки качества пользовательской модели?

11. КЛАССИФИКАЦИЯ С ПОМОЩЬЮ ДЕРЕВЬЕВ РЕШЕНИЙ

Изучаемые понятия: дерево решений, классификация, правило, узел дерева решений, лист дерева решений, поддержка, достоверность.

Деревья решений (decision trees) – это способ представления правил в иерархической, последовательной структуре, где каждому объекту соответствует единственный узел, дающий решение. Под правилом понимается логическая конструкция, представленная в виде «если … то …».

Дерево решений состоит из *узлов*, содержащих правила проверки условий, и *листьев* – конечных элементов дерева, указывающих на класс (т. наз. *узлов решения*).

Метод деревьев решений основан не на статистическом подходе, а на машинном обучении и является одним из самых популярных и мощных инструментов Data Mining.

В Deductor деревья решений применяются только для решения задачи классификации.

Классификация (лат. klassik – разряд, группа) – это отнесение объектов (предметов, явлений, процессов, понятий) к одному из заранее известных классов в соответствии с определенным признаком.

Иными словами, классификация устанавливает зависимость выходных данных от исходных – как и регрессия, – но, в отличие от регрессии, выходные данные должны быть дискретными.

Для построения дерева решений готовится обучающая выборка – так же, как для нейросети, однако выходное поле для дерева решений может быть только одно и дискретного типа.

Для обучения дерева решений требуется задать значения параметров:

- *минимальное количество примеров* (в узле) – этот параметр предотвращает создание листьев деревьев, содержащих всего несколько записей или даже одну (т. наз. эффект переобучения). Чем больше этот параметр, тем менее ветвистым получается дерево;
- *строить дерево с более достоверными правилами в ущерб компактности дерева* – при этом дерево получается более ветвистым (более сложной структуры);
- *уровень доверия, используемый при отсечении узлов дерева* (в процентах от 0 до 100) – чем меньше уровень доверия, тем больше узлов будет отсечено при построении дерева.

Найденные правила дерева решений применяются для тестового множества – той части исходных данных, которая не участвовала в построении дерева. Для него подсчитывается число полученных верных и ошибочных результатов.

Качество построенного дерева можно оценить по числу распознанных примеров в обучающем и тестовом наборах данных. Количество узлов в качественно построенном дереве не должно быть слишком велико – это означало бы, что найденные зависимости слабы. Кроме того, сложные деревья трудны для восприятия.

Для оценки найденных правил определяют показатели *достоверности* и *поддержки*. Deductor отображает эти значения для каждого узла.

Достоверность – количество примеров, правильно классифицированных данным узлом дерева.

Поддержка – общее количество примеров, классифицированных данным узлом.

Практическая работа 11.
Классификация с помощью деревьев решений

Задание 1. Классификация результатов голосования.

Найдите правила отнесения политиков к одной из партий на основании их поддержки серии законопроектов.

Порядок выполнения работы:

1. Создайте папку *ПР11 Деревья решений.* Скопируйте в нее и импортируйте в Deductor данные из демонстрационного файла *Vote.txt*. Сохраните проект в файле *Ваша_фамилия ПР11 Деревья решений*.

2. Запустите Мастер обработки и выберите обработчик *Дерево решений*.

- Укажите поле *Код* – информационным столбцом, *Класс* – выходным, а остальные – входными. Проверьте, как Deductor предлагает преобразовывать типы исходных данных (*Настройка нормализации*).
- Далее задайте случайный способ разбиения.
- Настройте параметры процесса обучения: минимальное количество примеров, при котором будет создан новый узел, – 2; включите опции *Строить дерево с более достоверными данными* и *Отсекать узлы дерева*. *Уровень доверия* установите 20%.
- Включите в разделе *Data Mining* визуализаторы *Таблицу сопряженности, Дерево решений*, *Правила.*

3. Проанализируйте *Таблицу сопряженности* – выясните количество ошибок работы алгоритма. Отобразите ее в виде процентов по вертикали. Выясните, какие доли партийцев были отнесены не к своему классу в обучающем множестве, в тестовом множестве данных.

4. Сравните способы представления правил в визуализаторах *Правила* и *Дерево решений.* Определите правила, по которым можно отнести депутата к той или иной партии. Выберите правила, вызывающие наибольшее доверие.

5. Добавьте к визуализации данных *Значимость атрибутов.* Определите самый значимый фактор, по которому можно судить о принадлежности депутата к той или иной партии.

Задание 2. Скоринговый анализ.

На основании имеющихся данных о заемщиках банка построить дерево решений отнесения заемщика к классу надежных или ненадежных.

Порядок выполнения работы:

1. Скопируйте в Вашу папку и импортируйте в Deductor данные из демонстрационного файла *Credit.txt*. В ходе импорта определите для поля *Давать кредит (число)* (содержащего только значения 0 или 1) тип данных – целое дискретное. Именно это поле потребуется для разделения заемщиков на два класса, а обработчик *Дерево решений* работает только с дискретными выходными полями.

2. Проанализируйте импортированные данные – какие, на Ваш взгляд, поля наиболее значимы для позитивного решения о кредитовании?

3. Вызовите обработчик *Дерево решений*.

- Укажите выходным поле *Давать кредит (число)*, а входными – все поля, кроме *Даты кредитования, Количества* и *Давать кредит.*
- Выберите случайное выделение обучающего множества, минимальное число примеров в узле – 4, остальные параметры по умолчанию.
- Включите в разделе *Data Mining* все визуализаторы, кроме *Обучающего набора.*

4. Проанализируйте результат классификации:

- Ознакомьтесь с найденными правилами (визуализатор *Правила*). Найдите правила с самой слабой поддержкой (отсортируйте таблицу по поддержке). Откройте детализацию, выясните информацию об этих клиентах.
- Разверните все ветви в *Дереве решений.* Продумайте, для каких целей удобнее представление правил в *Правилах* или *Дереве решений*.
- Сравните свои эмпирические представления о значимости данных с расчетными значениями (*Значимость атрибутов*).
- Воспользуйтесь визуализатором *Что-если*, введите свои (или виртуальные) данные о заемщике и определите класс его кредитоспособности. Не забудьте о необходимости ввода данных из тех же диапазонов, что и исходные. Для удобства такого ввода отобразите статистику.
- Выясните количество ошибок работы алгоритма (*Таблица сопряженности*). Отобразите таблицу в виде процентов по горизонтали. Выясните для тестового множества, какая доля неблагонадежных заемщиков была одобрена как положительные (вместо значения 0 было предложено значение 1).

5. Скопируйте узел дерева решений и перенастройте копию – ограничьте минимальное число правил в узле равным 3. (Не забудьте переименовать новый узел.)

- Насколько увеличилось число правил?
- Сравните значимые поля в этом и предыдущем дереве.

Сохраните файл проекта. Заархивируйте Вашу папку *ПР11 Деревья решений* в архивный файл *Ваша_фамилия ПР11 Деревья решений.zip.* Скопируйте его в папку *control*.

Вопросы для самоконтроля

1. Что такое дерево решений?
2. Из чего состоит дерево решений?
3. Каковы требования к исходным данным для построения дерева решений?
4. Для решения каких задач используется построение дерева решений в Deductor?
5. В чем отличие классификации от регрессии? Какие требования к выходным данным при проведении классификации? регрессии?
6. Какие параметры настройки дерева решений предлагает соответствующий обработчик Deductor?
7. Какими параметрами характеризуется построенное дерево решений?
8. В чем отличия показателей достоверности и поддержки правил?
9. Как зависит качество построенного дерева решений от количества распознанных примеров в обучающем и тестовом наборах данных?
10. Как зависит качество построенного дерева решений от количества узлов в дереве?

12. КЛАСТЕРИЗАЦИЯ ДАННЫХ

Изучаемые понятия: кластер, кластеризация, интерпретация, профили кластеров, поддержка кластера, мощность кластера, значимость поля, топографическая диаграмма, диаграмма размещения.

Группировка объектов по различным признакам (критериям) способствует более успешному их анализу. Если классы объектов заранее известны, используют классификацию. Однако может оказаться значительное число объектов, которые не попадают в заготовленные классы или относятся к ним лишь частично. В этих случаях может использоваться кластеризация (от англ. *cluster* (скопление) – *кластер*, группа объектов с общими признаками).

Кластеризация – это разбиение объектов на группы так, чтобы объекты внутри кластера были максимально похожи друг на друга и максимально отличались от объектов из других кластеров. Чем больше похожи объекты внутри кластера и чем больше отличий между кластерами, тем точнее кластеризация.

Кластеризация (= *сегментирование*, *группировка*) может выступать этапом анализа данных. При этом выделяют группы схожих объектов, изучают их особенности и для каждой группы строят отдельную модель. Так, в маркетинге можно выделить группы клиентов, покупателей, товаров, исходя не из общих соображений, а для данной конкретной ситуации, и разработать для каждой из них отдельную стратегию.

Обычно при кластеризации количество кластеров неизвестно и сложно выбрать меру «похожести» объектов. В связи с этим существует огромное число алгоритмов кластеризации. Однако это не все сложности данного метода. Сформированным группам объектов (кластерам) надо дать содержательную интерпретацию – только тогда можно рассчитывать на успешный анализ данных.

Для *интерпретации* результатов кластеризации надо исследовать каждый полученный кластер, его статистические характеристики, распределение значений каждого признака (параметра) в кластере, оценить *мощность кластера* (*поддержку кластера*) – число объектов, попавших в него.

Среди обработчиков Deductor Studio имеется инструмент *Кластеризация*. Он позволяет кластеризовать данные двумя способами:

а) указав заранее количество групп (кластеров),

б) разрешив определить количество кластеров автоматически.

Важно помнить, что оба метода предполагают наличие в исходных данных некоторых раздельных групп, сгустков объектов – иначе найденные кластеры будут малоинформативны.

Результаты группировки можно отобразить с помощью таблицы *Профили кластеров.* Она позволяет упорядочить данные в соответствии с поддержкой кластеров. Можно отсортировать данные по *значимости* полей данных – степени зависимости результата от данного поля. Значимость данных отображается в числовом и графическом виде. Кроме того, можно вывести статистические характеристики – средние значения, доверительный интервал, стандартные ошибку и отклонение и т.п.

Удобна для анализа *многомерная диаграмма*, с помощью которой можно отобразить в пространстве значения параметров для каждого кластера. Эта диаграмма допускает свободное вращение.

Одна из разновидностей многомерной диаграммы – *топографическая диаграмма*, которая использует цвет для отображения исследуемого параметра.

Диаграмма размещения позволяет отобразить больше данных, чем многомерная, – по сравнению с ней можно добавить еще три параметра, придав точкам диаграммы разные форму, цвет и размер.

Практическая работа 12. Кластеризация данных

Задание. Кластеризация с помощью одноименного обработчика Deductor.

Требуется распределить регионы на кластеры по демографическим данным.

Порядок выполнения работы:

1. Создайте папку *ПР12 Кластеризация.* Скопируйте в нее и импортируйте в Deductor данные из демонстрационного файла *Population.txt*. Сохраните проект в файле *Ваша_фамилия ПР12 Кластеризация*.

2. Проведите редактирование аномалий и удаление шумов для полей *Удельный вес городского населения, Изменение численности населения, Среднегодовая численность населения занятых в экономике, Среднедушевой денежный доход*.

3. Выберите Мастер обработки *Кластеризация*.

4. На втором шаге Мастера укажите поля *Численность населения* и *Регион* информационными столбцами, а 4 обработанных (сглаженных) поля – входными.

5. На следующем шаге определите все множество данных как обучающее.

6. Далее задайте фиксированное количество кластеров, равное пяти.

7. Выберите из списка визуализаторов способы отображения данных *Профили кластеров* и *Куб*.

8. Для настройки визуализатора *Куб* выберите рассматриваемые свойства как факты, а номер кластера и регион – как измерения. Затем укажите отображение среднего значения по всем фактам рассматриваемой группы.

9. Проанализируйте полученные результаты:

- Рассмотрите таблицу *Профили кластеров*. Задайте сортировку кластеров по убыванию *поддержки* (количества записей, попавших в кластер). Поля упорядочьте по убыванию *значимости* (степени зависимости результата от данного поля) для столбца *Итого* (т.е. для всех данных).

  Для каждого поля отобразите только три параметра – *Значимость, Доверительный интервал* и *Среднее значение поля*.

  Определите кластеры, где самым значимым параметром является среднедушевой доход (определите его среднюю величину). Проанализируйте другие значимые параметры для этих кластеров. Выясните, какие регионы попали в эти кластеры, отобразив *данные по кластеру*.

  Найдите кластеры регионов с положительным и отрицательным значимым приростом населения. Выясните другие значимые их характеристики.
- Добавьте визуализатор *Многомерная диаграмма:* задайте для него отображение регионов, номеров кластеров и (по очереди) какого-либо из исследуемых параметров – например, среднедушевого дохода и т.д. Отобразите легенду.
- Возможно, более удобным для анализа может оказаться визуализатор *Диаграмма размещения* – он позволяет вывести больше параметров за счет вариаций цвета, размера и формы точек диаграммы.
- Используйте для анализа данных вкладку *Куб*, выведя на ней кросс-диаграмму.

10. Проведите для тех же исходных данных кластеризацию на 4 кластера. Отобразите необходимые визуализаторы и проведите аналогичный анализ получившихся кластеров.

11. Выберите тот вариант кластеризации, который допускает более удачную интерпретацию. Переименуйте для него каждый кластер в зависимости от его особенностей.

12. Скопируйте узел удачной (на Ваш взгляд) кластеризации и переместите его непосредственно к импортированным данным (без их сглаживания). Проведите эту кластеризацию на несглаженных данных и сравните результаты группировки в обоих случаях.

Сохраните файл проекта. Заархивируйте Вашу папку *ПР12 Кластеризация* в архивный файл *Ваша_фамилия ПР12 Кластеризация.zip.* Скопируйте его в папку *control*.

Вопросы для самоконтроля

1. Что такое кластер?
2. Чем кластеризация отличается от классификации?
3. Как реализована кластеризация в программе Deductor?
4. Каким образом может быть задано число кластеров при кластеризации в Deductor?
5. Что такое мощность кластера (поддержка), значимость полей? Как эти показатели могут быть использованы в интерпретации результатов кластеризации?
6. Какие способы визуализации доступны для представления результатов кластеризации?
7. Как устроена таблица профилей, какие данные она позволяет отобразить?
8. Какие способы сортировки и фильтрации данных допускает таблица профилей?
9. В чем отличие представления многомерной диаграммы в виде поверхности или топографической карты?
10. Какие дополнительные средства по сравнению с многомерной диаграммой использует диаграмма размещения для отображения большего числа параметров?

13. САМООРГАНИЗУЮЩИЕСЯ КАРТЫ КОХОНЕНА

Изучаемые понятия: карта Кохонена, ячейка карты Кохонена, матрица расстояний, обучение без учителя.

Самоорганизующиеся *карты Кохонена* (Kohonen Self-Organizing Map) применяются для отображения результатов кластеризации. Карты Кохонена позволяют отобразить многомерное пространство в двумерном.

Карта Кохонена состоит из прямоугольных или шестиугольных ячеек. Объекты, признаки которых близки, попадают в одну ячейку или в ячейки, расположенные вблизи. В общем случае в ячейку попадает несколько объектов.

Ячейки карты раскрашиваются в разные цвета (или оттенки серого цвета) в зависимости от значений параметров, соответствующих каждой ячейке. Для этого сначала выделяются диапазоны значений параметра. Каждому диапазону ставится в соответствие цвет (или оттенок серого), и ячейки карты «раскрашиваются» соответствующими цветами.

Количество карт равно количеству анализируемых параметров. Каждая карта соответствует одному параметру объекта (т.е. одному входному полю данных).

Можно отобразить в виде карты *Матрицу расстояний*, показывающую, насколько близко расположены друг к другу соседи по кластеру. Если используются оттенки серого цвета, то чем дальше соседи, тем светлее цвет узла.

Отличие самоорганизующихся карт Кохонена в том, что могут вовсе отсутствовать выходные поля в обучающей выборке. Поскольку самообучающиеся карты – это алгоритм обучения без учителя, то у них вообще нет выходных полей в том виде, как в задачах регрессии и классификации. Даже если задать выходные поля, они не будут участвовать в обучении нейросети, а только лишь в отображении карт. Собственно, указание выходных полей удобно использовать в картах Кохонена для поиска зависимостей.

В остальном нормализация полей и настройка обучающей выборки проводится так же, как и для нейросетей.

Число кластеров обычно выбирают в пределах от 4 до 9 – 2-3 кластера малоинформативны, а больше десятка кластеров трудно обрабатывать.

Следует помнить о том, что карты Кохонена не решают задачу кластеризации, а лишь предлагают гипотезы о кластерной структуре

данных. Эти гипотезы могут оказаться ложными, поэтому их необходимо подтверждать другими методами.

Кроме того, обычно начальные значения построения карт Кохонена задаются случайными числами, так что если проводить обучение несколько раз, результаты могут быть непохожими.

Для каждого кластера необходимо провести его содержательную интерпретацию, понимая при этом, что любая кластеризация субъективна и не существует единого универсального алгоритма кластеризации.

Практическая работа 13. Кластеризация с помощью карт Кохонена

Задание 1. Кластеризация банков.

Имеется база данных коммерческих банков с показателями деятельности за текущий период. Необходимо провести их кластеризацию, т.е. выделить однородные группы банков.

Порядок выполнения работы:

1. Создайте папку *ПР13 Карты Кохонена.* Скопируйте в нее и импортируйте в Deductor данные из файла *Banks.txt*. Сохраните проект в файле *Ваша_фамилия ПР13 Карты Кохонена*.
2. Запустите Мастер обработки и выберите обработчик *Карта Кохонена*.

- На втором шаге Мастера укажите выходным столбцом поле *Прибыль*, а поля *Филиалы*, Сум*ма активов, Собственные активы, Банковские активы, Средства в банке* – входными.
- Оставьте значения по умолчанию для обучающего и тестового множеств, настроек параметров карты Кохонена, за исключением числа кластеров: задайте фиксированное число кластеров 4.

3. После завершения процесса обучения в списке визуализаторов выберите *Карту Кохонена* и *Что–если*.

- В Мастере настройки карты Кохонена укажите поля для отображения: все входные и выходные столбцы и кластеры. Способ раскраски – цветная палитра. Задайте отображение границ кластеров.

4. На полученной карте *Кластеры* все данные будут отображены в виде четырех областей разного цвета.

- Выберите кластер меньшего размера. Проанализируйте аналогичные по расположению области других карт: выясните размеры различных активов для банков этого кластера. (Цветовая шкала числовых величин расположена под каждой картой: синие цвета соответствуют наименьшим значениям, красные оттенки – наибольшим.)
- Откройте панель данных внизу окна и включите фильтрацию по кластеру. Выясните, какие банки попали в рассматриваемый кластер.
- Затем выберите самый большой кластер и проанализируйте финансовые показатели его банков на остальных картах. Аналогично – для остальных кластеров.

5. Изучите распределение кластеров на одной из карт с финансовыми показателями – например, *Сумма активов*. Добавьте визуализатор *Многомерная диаграмма* для этого же показателя (оси: номер кластера, банк, сумма активов). Сравните оба вида визуализации.

6. С помощью карты *Филиалы* проверьте, что число филиалов не сказалось на результатах кластеризации.

- Добавьте визуализацию *Профилей кластеров*. Оставьте в этой таблице параметры *Значимость, Доверительный интервал* и *Среднее*. Проверьте значимость поля *Филиалы* для всех кластеров – соответствует ли она карте Кохонена?

7. С помощью карты *Прибыль* проверьте, имеется ли связь прибыльности банков и их группировки.

- Создайте еще один узел сценария – аналогичную *Карту Кохонена* для тех же данных, указав вместо входного параметра *Филиалы* входной параметр *Прибыль*. Дайте различные названия обоим узлам сценария.
- Визуализируйте таблицу *Профили кластеров.* Проверьте значимость поля *Прибыль* для разных кластеров .
- Сравните оба результата кластеризации на картах Кохонена.

Сохраните файл проекта. Заархивируйте Вашу папку *ПР13 Карты Кохонена* в архивный файл *Ваша_фамилия ПР13 Карты Кохонена.zip.* Скопируйте его в папку *control*.

Вопросы для самоконтроля

1. Что такое карта Кохонена?
2. Для решения каких задач используются самоорганизующиеся карты Кохонена?

3. В чем отличие самоорганизующихся карт Кохонена от других средств отображения кластеризации?
4. Зачем производится раскраска ячеек?
5. Как определить число кластеров в обработчике *Карта Кохонена*?
6. Что такое обучение без учителя?
7. Что отображает карта *Матрица расстояний*?

14. АССОЦИАТИВНЫЕ ПРАВИЛА

Изучаемые понятия: ассоциативные правила, поддержка правила, достоверность, мощность множеств, количество множеств, лифт.

Ассоциация – выявление закономерностей между связанными событиями. Правила, основанные на ассоциации, называются *ассоциативными правилами*.

Транзакция – это множество событий, произошедших одновременно (групповая операция). События, произошедшие одновременно, называют *элементами транзакций*.

При поиске ассоциативных правил целью является нахождение частных зависимостей между объектами и событиями. Найденные зависимости представляются в виде правил и могут быть использованы как для лучшего понимания природы анализируемых данных, так и для предсказания появления событий. С помощью ассоциативных правил можно выявить закономерности между связанными событиями.

Параметры построения ассоциативных правил:

- *Минимальная и максимальная поддержка* в % – ограничивают пространство поиска часто встречающихся предметных наборов. Эти границы определяют множество популярных наборов, из которых и будут создаваться ассоциативные правила.
- *Минимальная и максимальная достоверность* в % – в результирующий набор попадут только те ассоциативные правила, которые удовлетворяют условиям минимальной и максимальной достоверности.
- *Максимальная мощность искомых часто встречающихся множеств* – параметр ограничивает длину k-предметного набора. Например, при установке значения 4 шаг генерации популярных наборов будет остановлен после получения множества

4-предметных наборов. В конечном итоге это позволяет избежать появления длинных ассоциативных правил, которые трудно интерпретируются.

- *Количество множеств* – число популярных наборов, удовлетворяющих заданным условиям минимальной поддержки и достоверности.
- *Количество правил* – число сгенерированных ассоциативных правил.
- *Лифт* – отношение частоты появления условия в транзакциях, которые также содержат и следствие, к частоте появления следствия в целом.

Практическая работа 14. Поиск ассоциативных правил

Задание 1. Поиск ассоциативных правил для розничной торговли.

Требуется решить задачу анализа потребительской корзины с целью последующего применения результатов для стимулирования продаж.

Порядок выполнения работы:

1. Создайте папку *ПР14 Ассоциации.* Скопируйте в нее и импортируйте в Deductor данные из файла *Supermarket.txt*, содержащего набор необходимых транзакций. Сохраните проект в файле *Ваша_фамилия ПР14 Ассоциации*.

2. Запустите Мастер обработки и выберите тип обработки *Ассоциативные правила*.

- Укажите столбец *Чек* идентификатором транзакции, а столбец *Товар* – элементом транзакции. Оба поля (идентификатор и элемент транзакции) должны быть дискретного вида.
- Установите в параметрах алгоритма % максимальной поддержки не более 35%. Очевидно, в этом случае не будет выявлено никаких ассоциативных правил – это видно уже на следующем, 3-м, шаге алгоритма. Вернувшись на шаг назад, увеличьте % максимальной поддержки до 40%. Убедитесь в существовании двух правил в этом случае. В конце концов установите 45% максимальной поддержки.

3. Запустите процесс поиска ассоциативных правил. Выберите визуализаторы *Популярные наборы, Правила, Дерево правил, Что–если*.

4. Проанализируйте полученные результаты.

5. Проведите еще раз настройку ассоциативных правил, настроив параметры построения ассоциативных правил: границы поддержки – 13% и 80% и достоверности – 60% и 90%. Проанализируйте предпочтения клиентов.

6. Проведите сегментирование покупателей по поведению при покупках.

Задание 2. Поиск ассоциативных правил для оптовой торговли.

Требуется решить задачу анализа оптовой торговли с помощью поиска ассоциаций.

Порядок выполнения работы:

1. Скопируйте в свою папку файл *conf.txt*, содержащий записи об оптовой торговле кондитерскими изделиями. Импортируйте данные файла в тот же проект .

2. Определите ассоциативные правила покупок и отобразите их с помощью визуализаторов *Правила, Дерево правил, Что–если* и *Популярные наборы*.

3. Проварьируйте величины минимальной и максимальной поддержки правил, установив в итоге минимальную поддержку не менее 5%, максимальную от 50% до 70%, минимальную и максимальную достоверность – 60% и 90% соответственно.

4. Проверьте ожидаемые ассоциации с помощью механизма *Что–если* (кнопка *Вычислить правила* либо комбинация клавиш Ctrl+Enter) при одновременной покупке двух каких-либо товаров.

Задание 3 для самостоятельной работы.

Скопируйте в свою папку файл *farm.txt* с данными об оптовой торговле фармацевтическими препаратами. Определите ассоциативные правила для этих данных. Установите минимальную и максимальную поддержку, минимальную и максимальную достоверность – 1%, 30%, 20%, 70% соответственно.

Сохраните файл проекта. Заархивируйте Вашу папку *ПР14 Ассоциации* в архивный файл *Ваша_фамилия ПР14 Ассоциации.zip.* Скопируйте его в папку *control*.

Вопросы для самоконтроля

1. Что такое ассоциативные правила?
2. Для решения каких задач могут быть использованы найденные ассоциации?
3. Как в программе Deductor осуществляется поиск ассоциативных правил?
4. Каковы требования к данным для использования обработчика поиска ассоциаций?
5. Каковы параметры построения ассоциативных правил?
6. Каким образом выбирать значения границ поддержки и достоверности при поиске ассоциаций? Как влияют эти значения на полученные результаты?

Основные комбинации клавиш Deductor Studio

Комбинация клавиш	Действие
Панель сценариев	
Ctrl + Enter	Открыть окно представления для текущего узла
Alt + Enter	Изменить параметры узла
Shift + Enter	Активизировать узел
F5	Вызвать Мастер визуализации
F6	Вызвать Мастер импорта
F7	Вызвать Мастер обработки
F8	Вызвать Мастер экспорта
Ctrl + Del	Удалить текущий узел
F2	Переименовать текущий узел
Ctrl + ↑	Переместить текущий узел на одну позицию вверх по дереву
Ctrl + ↓	Переместить текущий узел на одну позицию вниз по дереву
Таблица	
Ctrl + Home	Перейти к началу таблицы
Ctrl + End	Перейти в конец таблицы
Куб	
Enter	Показать окно детализации текущей ячейки
F4	Настроить размещение и варианты агрегации фактов
F9	Фильтровать данные по измерениям и значениям фактов
Ctrl + T	Транспонировать (поменять местами строки и столбцы)
Статистика	
Ctrl + 0, **Ctrl + 1 - Ctrl + 8**	Отобразить / скрыть все (или отдельные) статистические показатели
Карта Кохонена	
F4	Показать/скрыть окно данных
Num +	Увеличить размер карт
Num –	Уменьшить размер карт
Что-если	
F9	Рассчитать выходы
Дерево решений	
F4	Показать источник данных
F12	Показать информацию по узлу

Предметный указатель

Список литературы

РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

1. Паклин Н.Б., Орешков В.И. Бизнес-аналитика: от данных к знаниям : учеб. пособие (+CD) / Н.Б. Паклин, В.И. Орешков. – СПб. : Питер, 2013. – 704 с.

2. Deductor. Аналитическая платформа для эффективных бизнес-решений [Электронный ресурс]. – Режим доступа: http://basegroup.ru.

3. Deductor. Руководство аналитика. Версия 5.2. – BaseGroup Labs, 2010.

4. Дюк В.А, Самойленко А.П. Data Mining: учебный курс (+CD). / В.А. Дюк, А.П. Самойленко. – СПб. : Питер, 2001. – 368 с.

5. Барсегян А.А. Методы и модели анализа данных: OLAP и Data Mining; 2-е изд. / А.А. Барсегян , М.С. Куприянов , В.В. Степаненко и др. – СПб. : БХВ – Петербург, 2008. -

6. Кацко И.А. Практикум по анализу данных на компьютере : учеб.-практ. пособие / И.А.Кацко, Н.Б.Паклин. – М. : КолосС, 2009. – 278 с., илл.

7. Зиновьев А.Ю. Визуализация многомерных данных / А.Ю. Зиновьев. – Красноярск : Изд-во КГТУ, 2000. – 180 с.

8. Плескач В.Л. Інформаційні системи і технології на підприємствах : підручник; затверджено МОН / В.Л. Плескач, Т.Г. Затонацька – К. : Знання, 2011. – 718 с.

ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА

1. Чубукова И.А. Data Mining : учеб. пособие. – М. : Интернет-ун-т информ. технологий; БИНОМ; Лаборатория знаний, 2006. – 382 с.

2. Осовский С. Нейронные сети для обработки информации / С. Осовский. – М. : Финансы и статистика, 2004. – 344 с.

3. KDnuggets – Data Mining Community's Top Resource [Электронный ресурс]. – Режим доступа: http://www.kdnuggets.com.

4. Ханк Д.Э., Уичерн Д.У., Райте А.Дж. Бизнес-прогнозирование; 7-е изд.; пер. с англ. – М. : Вильямс, 2003.

Содержание