ПЕРВОЕ ВЫСШЕЕ ТЕХНИЧЕСКОЕ УЧЕБНОЕ ЗАВЕДЕНИЕ РОССИИ

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего образования
САНКТ-ПЕТЕРБУРГСКИЙ ГОРНЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ

Руководитель программы аспирантуры
доцент Ю.В. Ильюшин

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ПРОВЕДЕНИЯ ПРАКТИЧЕСКИХ ЗАНЯТИЙ ПО ДИСЦИПЛИНЕ

СИСТЕМЫ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Подготовка научных и научно-педагогических кадров в аспирантуре

Область науки:	2. Технические науки
Группа научных специальностей:	2.3. Информационные технологии и телекоммуникации
Научная специальность:	2.3.1. Системный анализ, управление и обработка информации, статистика
Отрасли науки:	Технические
Форма освоения программы аспирантуры:	Очная
Срок освоения программы аспирантуры:	3 года
Составитель:	к.т.н., доц. Мазаков Е.Б.

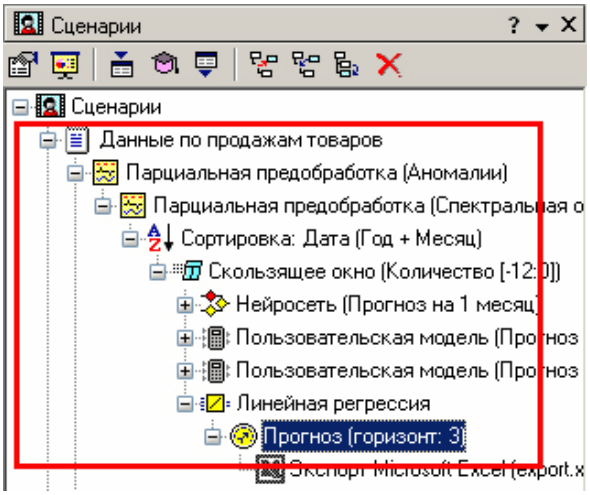
Санкт-Петербург

СОДЕРЖАНИЕ

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 1
ПРОГНОЗИРОВАНИЕ С ПОМОЩЬЮ ЛИНЕЙНОЙ РЕГРЕССИИ

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

Рассмотрим фрагмент проекта "Демопример анализа данных.ded".



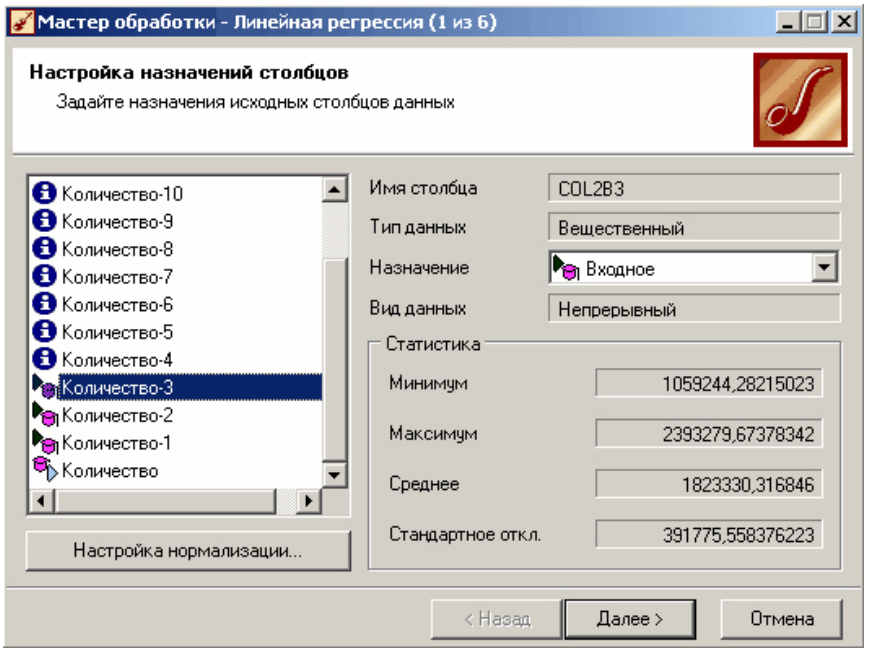
Исходные данные

Рассмотрим применение линейной регрессии на примере данных по продажам, находящихся в файле "Trade.txt". Будем строить прогноз с помощью линейной регрессии от ветки импорта "Данные по продажам товаров" сразу после обработчика "Скользящее окно".

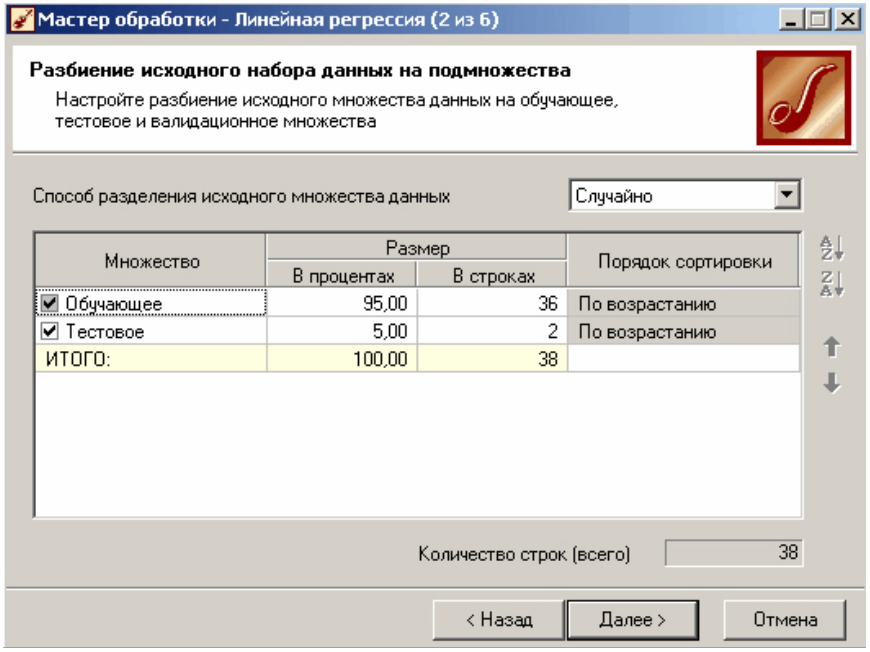
Обучение линейной регрессии

Для построения линейной регрессии необходимо запустить Мастер обработки и выбрать в качестве обработки данных Линейную регрессию.

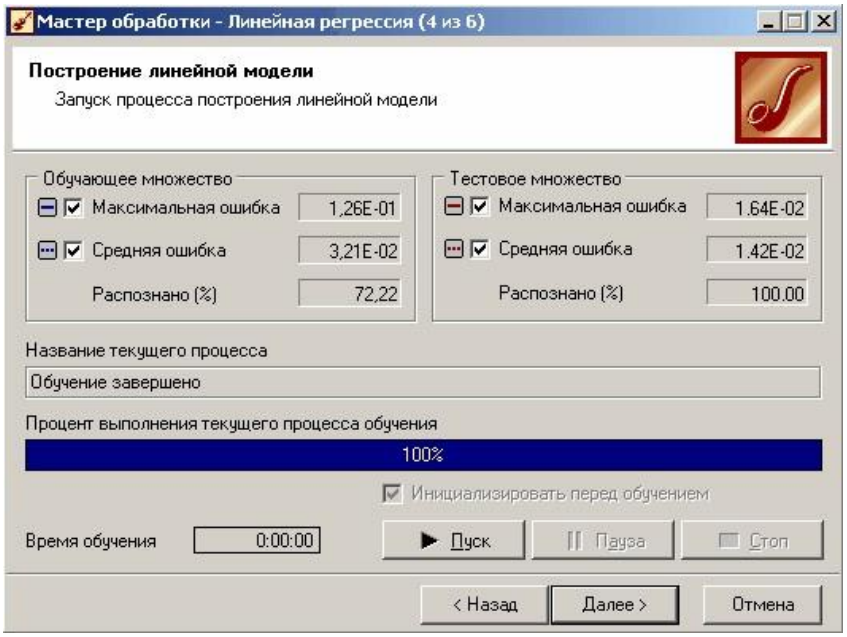
На первом шаге задаем назначение исходных столбцов. Предположим, что на прогноз влияет информация за 3 прошлых месяца, тогда укажем входными столбцами поля: "Количество – 3", "Количество – 2", и "Количество – 1". В качестве выходного поля укажем столбец "Количество".



На следующем шаге происходит настройка обучающего и тестового множеств, способ разложения исходного множества данных.

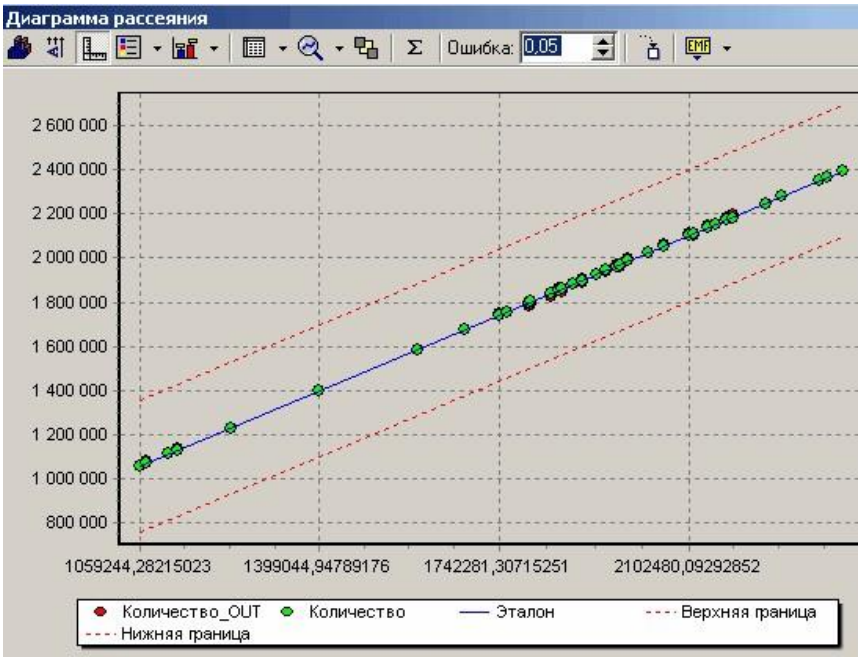


Третий шаг установки позволяет осуществить ограничение диапазона входных значений. Данный шаг оставим без изменений. При нажатии на кнопку "Далее" появляется окно запуска процесса обучения. В процессе выполнения видно, какая часть распознана на этапе обучения и теста.



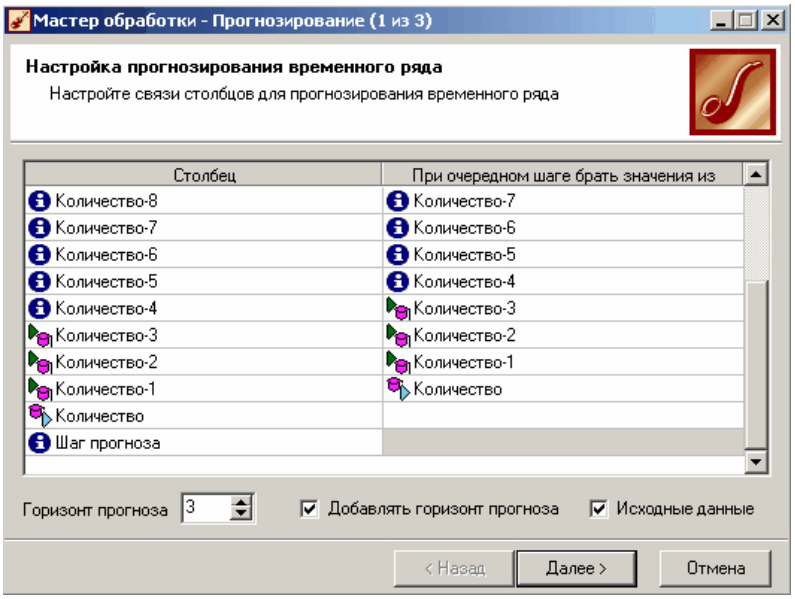
Результат

После выполнения процесса выберем в качестве способа отображения диаграмму рассеяния и отображение результатов в виде диаграммы. Как видно из диаграммы рассеяния, обучение прошло с хорошей точностью.



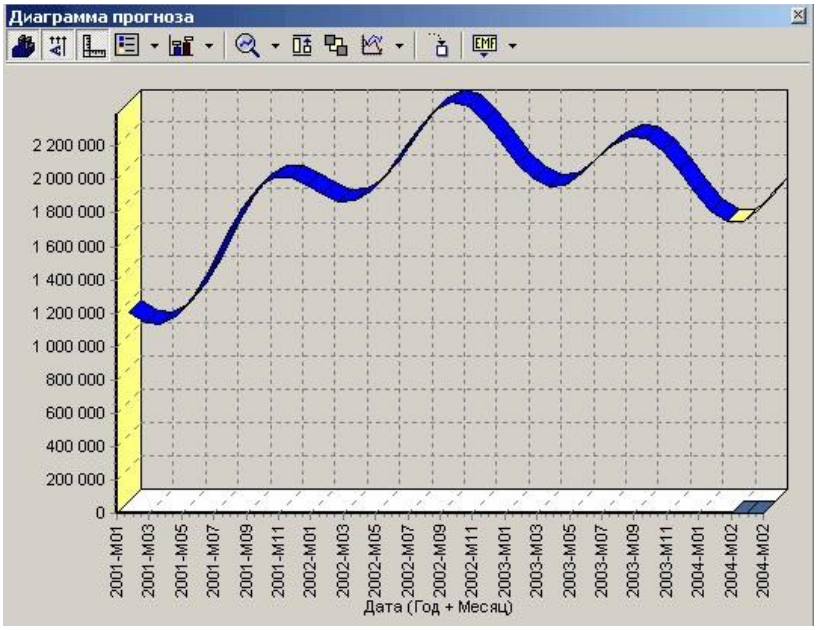
Прогнозирование

Теперь для построения прогноза запустим Мастера обработки, в котором выберем прогнозирование. На первом шаге обработчика происходит настройка связи столбцов для прогнозирования. Укажем связь между столбцами и горизонт прогноза равный 3.



Результат

На следующем шаге задаются параметры визуализации. Для данного примера выбираем отображение результатов в виде диаграммы прогноза. Теперь аналитик может дать прогноз о продажах, основываясь на модели, построенной с помощью линейной регрессией.



Выводы

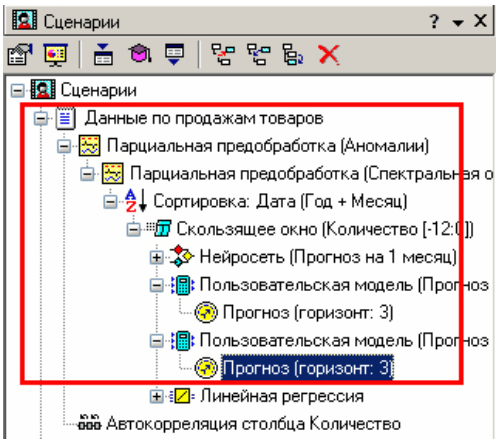
Данный пример показал целесообразность применения линейного регрессионного анализа для прогнозирования линейных зависимостей.

Простота настроек и быстрота построения модели иногда бывают необходимы. Аналитику достаточно указать входные столбцы – факторы, выходные — результат, указать способ разбиения данных на тестовое и обучающее множество и запустить процесс обучения. Причем после этого будут доступны все механизмы визуализации и анализа данных, позволяющие построить прогноз, провести эксперимент по принципу "Что– если", исследовать зависимость результата от значений входных факторов, оценить качество построенной модели по диаграмме рассеяния. Также по результатам работы этого алгоритма можно подтвердить или опровергнуть гипотезу о наличии линейной зависимости.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 2
ПРОГНОЗИРОВАНИЕ С ПОМОЩЬЮ ПОСТРОЕНИЯ ПОЛЬЗОВАТЕЛЬСКИХ МОДЕЛЕЙ

Пользовательская модель позволяет создавать аналитические модели на основании формул и экспертных оценок. Такая возможность требуется в тех случаях, когда объем исходной выборки мал, либо ее качество недостаточно для того, чтобы, например, обучить нейронную сеть. В этом случае можно воспользоваться хорошо известными простыми моделями, которые задаются с помощью формул.

Рассмотрим фрагмент проекта "Демопример анализа данных.ded".



Исходные данные

Рассмотрим применение пользовательской модели на примере данных по продажам, находящихся в файле "Trade.txt". Будем строить пользовательские модели на ветке "Данные по продажам товаров" сразу после обработчика "Скользящее окно".

Рассмотрим две пользовательские модели. Пусть аналитику известен характер продаж определенных товаров. Так, например, известно, что каждый месяц наблюдается постоянный прирост объема продаж, равный 160000, и спад продаж, равный 12% от аналогичного периода прошлого года, а также прирост в 2% по сравнению с текущим месяцем. Таким образом, аналитик может рассчитать прогноз по формуле:

Прогноз = ОбъемТекущегоМесяца * 1.02 + 160000 – ОбъемМесяцаГодНазад * 0.12 .

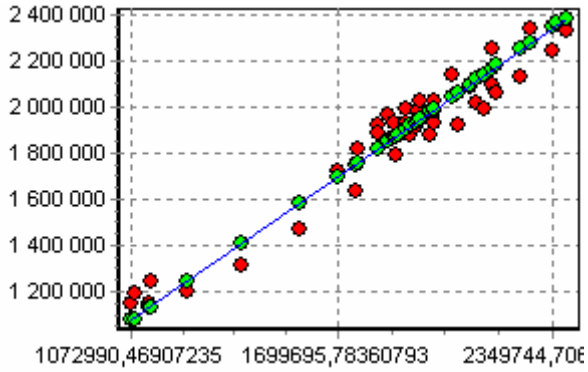
Также аналитик может воспользоваться реализованной моделью скользящего среднего, которая подразумевает, что объем продаж следующего месяца равен среднему объему продаж некоторого количества предшествующих месяцев. Рассмотрим их по очереди.

Прогнозирование с применением пользовательских моделей. Для построения пользовательской модели необходимо запустить Мастер обработки и выбрать в качестве обработки данных пользовательскую модель. На втором шаге Мастера настроим поля исходных данных.

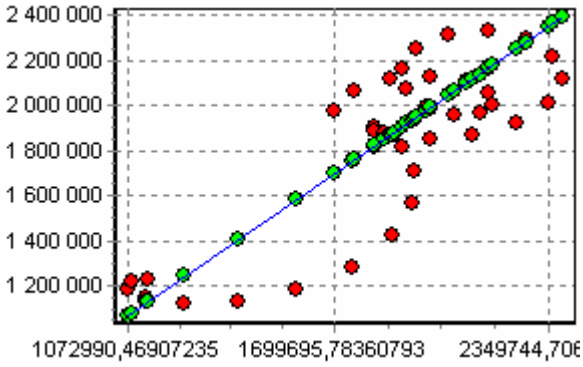
Для первой модели необходимо выбрать в качестве входных полей. "Количество – 12" и "Количество – 1", а выходным будет поле "Количество". При построении второй пользовательской модели необходимо на данном этапе в качестве входов указать поля "Количество – 5" … "Количество – 1". На следующем шаге Мастера необходимо написать формулу получения прогноза. В поле ввода выражения необходимо написать правую часть формулы, известную аналитику, а именно "160000 - 0.12 * COL2B12 + 1.02 * COL2B1" (COL2B12 и COL2B1 – соответственно имена полей "Количество – 12" и "Количество – 1").

При построении второйпользовательской модели выражение будет следующее: "MOVINGAVERAGE(COL2B1;COL2B2;COL2B3;COL2B4;COL2B5)" (здесь используется встроенная функция расчета среднего значения, в данном случае среднего объема продаж за пять предыдущих месяцев). Далее необходимо перейти на следующий шаг и

выбрать способ визуализации. Вот как, например, выглядят диаграммы рассеяния обеих пользовательских моделей:



а) Модель, полученная по формуле

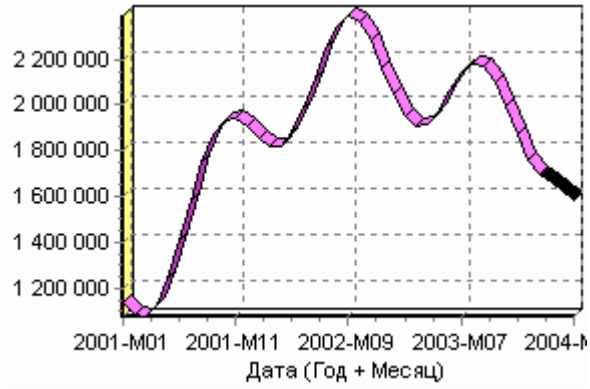


б) Модель скользящего среднего

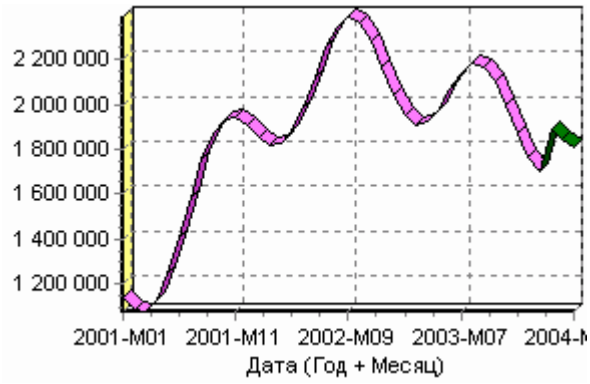
Далее также как и в примере построения прогноза объема продаж после обеих пользовательских моделей построим прогноз на 3 месяца вперед.

Результат

Вот как выглядят их диаграммы прогноза:



а) Модель, полученная по формуле



б) Модель скользящего среднего

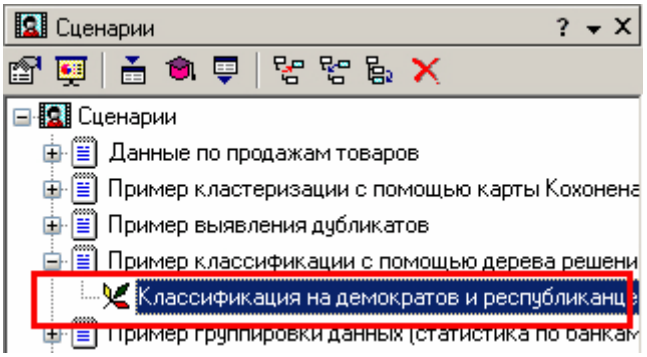
Выводы

Данный пример показал целесообразность применения пользовательских моделей для прогнозирования простых или до определенной степени известных зависимостей. Простота настроек и быстрота построения модели иногда бывают необходимы. Причем после этого будут доступны все механизмы визуализации и анализа данных, позволяющие построить прогноз, провести эксперимент по принципу "Что–если", исследовать зависимость результата от значений входных факторов, оценить качество построенной модели по таблице сопряженности или диаграмме рассеяния и возможно скорректировать расчетную формулу для более точного отражения зависимости.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 3
КЛАССИФИКАЦИЯ С ПОМОЩЬЮ ДЕРЕВЬЕВ РЕШЕНИЙ

Деревья решений применяются для решения задачи классификации. Дерево представляет собой иерархический набор условий (правил), согласно которым данные относятся к тому или иному классу. В построенном дереве присутствует информация о достоверности того или иного правила. Рассчитывается значимость каждого входного поля.

Рассмотрим фрагмент проекта "Демопример анализа данных.ded".



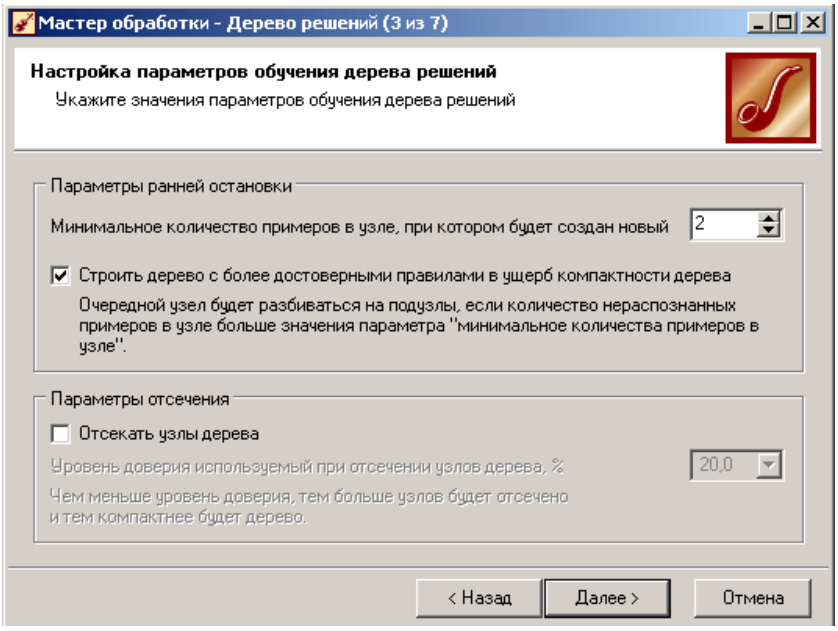
Исходные данные

Пусть аналитик имеет данные по тому, как голосуют депутаты конгресса США по различным законопроектам. Также известна партийная принадлежность каждого депутата — республиканец или демократ. Перед аналитиком поставлена задача: классифицировать депутатов на демократов и республиканцев в зависимости от того, как они голосуют. Данные по голосованию находятся в файле "Vote.txt". Таблица содержит следующие поля : "Код" — порядковый номер, "Класс" — класс голосующего (демократ или республиканец), остальные поля информируют о том, как голосовали депутаты за принятие различных законопроектов ("да" , "нет" , "воздержался").

Классификация на демократов и республиканцев

Для решения задачи запустим Мастер обработки. Выберем в качестве обработки дерево решений. В Мастере построения дерева решения на втором шаге настроим поле "Код" информационным, "Класс" выходным, остальные поля входными.

Далее предлагается настроить способ разбиения исходного множества данных на обучающее и тестовое. Зададим случайный способ разбиения, когда данные для тестового и обучающего множества берутся из исходного набора случайным образом. На следующем шаге Мастера предлагается настроить параметры процесса обучения, а именно минимальное количество примеров, при котором будет создан новый узел (пусть узел создается, если в него попали два и более примеров), а также предлагается возможность строить дерево с более достоверными правилами. Включим данные опции.



На следующем шаге Мастера выбираем процесс построения дерева решения в автоматическом режиме или интерактивном (полуавтоматическом). Выберем сначала автоматический режим построения и запустим. Далее можно увидеть информацию о количестве распознанных примеров.

После построения дерева можно увидеть, что почти все примеры и на обучающей и на тестовой выборке распознаны.

Перейдем на следующий шаг Мастера для выбора способа визуализации полученных результатов. Основной целью аналитика является отнесение депутата к той или иной партии. Механизм отнесения должен быть таким, чтобы депутат указал, как он будет голосовать за различные законопроекты, а дерево решений ответит на вопрос, кто он — демократ или республиканец. Такой механизм предлагает визуализатор "Что–если". Не менее важным является и просмотр самого дерева решений, на котором можно определить, какие факторы являются более важными (верхние узлы дерева), какие второстепенными, а какие вообще не оказывают влияния (входные факторы, вообще не присутствующие в дереве решений). Поэтому выберем также и визуализатор "Дерево решений". Формализованные правила классификации, выраженные в форме "Если <Условие>, тогда <Класс>", можно увидеть, выбрав визуализатор "Правила (дерево решений)". Часто аналитику бывает полезно узнать, сколько примеров было распознано неверно, какие именно примеры были отнесены к какому классу ошибочно. На этот вопрос дает ответ визуализатор "Таблица сопряженности". Очень важно знать, каким образом каждый фактор влияет на классификацию. Такую информацию предоставляет визуализатор "Значимость атрибутов".

Результат

Проанализируем данные при помощи имеющихся визуализаторов. Для начала посмотрим на таблицу сопряженности.

Дерево решений | Таблица сопряженности | Значимость атрибутов

Класс

	Классифицировано		
Фактически	демократ	республиканец	Итого
демократ	92		92
республиканец	4	54	58
Итого	96	54	150

По диагонали таблицы расположены примеры, которые были правильно распознаны, в остальных ячейках — те, которые были отнесены к другому классу. В данном случае дерево правильно классифицировало практически все примеры.

Перейдем к основному визуализатору для данного алгоритма — "Дерево решений". Как видно, дерево решений получилось не очень громоздкое, большая часть факторов (законопроектов) была отсечена, т.е. влияние их на принадлежность к партии минимальна или его вообще нет (по- видимому, по этим вопросам у партий нет принципиального противостояния).

Дерево решений | Правила | Значимость атрибутов | Таблица сопряженности

Правил: 9 из 9 | Фильтр: Без фильтрации

№	Номер	Условие			Следствие	Поддержка		Достоверность	
		Показатель	Знак	Значение	Класс	Кол-во	%	Кол-во	%
1	1	Закон о врачах	=	воздержался	демократ	4	2,82	3	75,00
2	2	Закон о врачах	=	да	республиканец	1	0,70	1	100,00
		Проект по Сальвадору	=	воздержался					
3	3	Закон о врачах	=	да	республиканец	4	2,82	4	100,00
		Проект по Сальвадору	=	да					
		Закон об образовании	=	воздержался					

Самым значимым фактором оказалась позиция, занимаемая депутатами по пакету законов, касающихся врачей, т. е. если депутат голосует против законопроекта о врачах, то он демократ (об это можно говорить с полной уверенностью, потому что в узел попало 83 примера). Достоверно судить о том, что депутат — республиканец, можно, если он голосовал за законопроект о врачах, а также за законопроект по Сальвадору, а также был против законопроекта об усыновлении. Данный визуализатор предоставляет возможность просмотра примеров, которые попали в тот или иной узел, а также информацию об узле.

Более удобно посмотреть значимость факторов или атрибутов в визуализаторе "Значимость атрибутов".

Целевой атрибут: Класс		
№	Атрибут	Значимость, %
4	Закон о врачах	92.207
16	Проект по экспорту	3.498
5	Проект по Сальвадору	2.455
3	Проект по усыновлению	1.840
12	Закон об образовании	0.000
11	Проект по альтернативным источникам топлива	0.000
13	Проект по фондам	0.000
15	Проект по таможенным пошлинам	0.000
14	Проект по преступности	0.000
10	Закон об иммигрантах	0.000
6	Закон о религиях	0.000
2	Проект по водным ресурсам	0.000
1	Проект по инвалидам	0.000
9	Проект по ракетам	0.000
8	Проект помощи Никарагуа	0.000
7	Антиспутниковый проект	0.000

С помощью данного визуализатора можно определить, насколько сильно выходное поле зависит от каждого из входных факторов. Чем больше значимость атрибута, тем больший вклад он вносит при классификации. В данном случае самый большой вклад вносит закон о врачах, как и было сказано выше.

На визуализаторе "Правила" представлен список всех правил, согласно которым можно отнести депутата к той или иной партии. Правила можно сортировать по поддержке, достоверности, фильтровать по выходному классу (к примеру, показать только те правила, согласно которым депутат является демократом с сортировкой по поддержке).

Дерево решений | Правила | Значимость атрибутов | Таблица сопряженности

Правил: 9 из 9 | Фильтр: Без фильтрации

№	Номер	Условие			Следствие	Поддержка		Достоверность	
		Показатель	Знак	Значение	Класс	Кол-во	%	Кол-во	%
1	1	Закон о врачах	=	воздержался	демократ	4	2,82	3	75,00
2	2	Закон о врачах	=	да	республиканец	1	0,70	1	100,00
		Проект по Сальвадору	=	воздержался					
3	3	Закон о врачах	=	да	республиканец	4	2,82	4	100,00
		Проект по Сальвадору	=	да					
		Закон об образовании	=	воздержался					

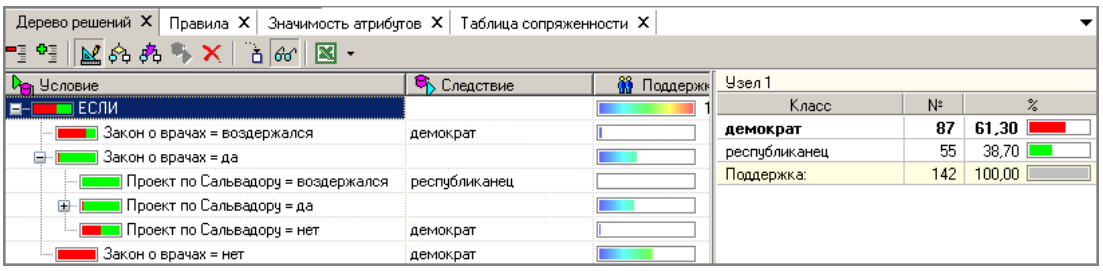
Данные представлены в виде таблицы. Полями этой таблицы являются:

- номер правила;
- условие, которое однозначно определяет принадлежность к партии;
- следствие — то, кем является депутат, голосовавший согласно этому условию;
- поддержка — количество и процент примеров из исходной выборки, которые отвечают этому условию;
- достоверность — процентное отношение количества верно распознанных примеров, отвечающих данному условию, к общему количеству примеров, отвечающих данному условию.

Исходя из данных этой таблицы, аналитик может сказать, что именно влияет на то, что депутат является демократом или республиканцем, какова цена этого влияния (поддержка) и какова достоверность правила. В данном случае совершенно очевидно, что из всего списка правил с достаточно большим доверием можно отнестись к двум : правилу №9 и правилу №7. Таким образом, получается, что демократы принципиально против законопроектов, касающихся врачей. Республиканцы же, наоборот, за принятие этих законопроектов и также за принятие законопроекта по Сальвадору, но категорически против законопроектов по усыновлению.

Теперь аналитик может точно сказать, кто есть кто.

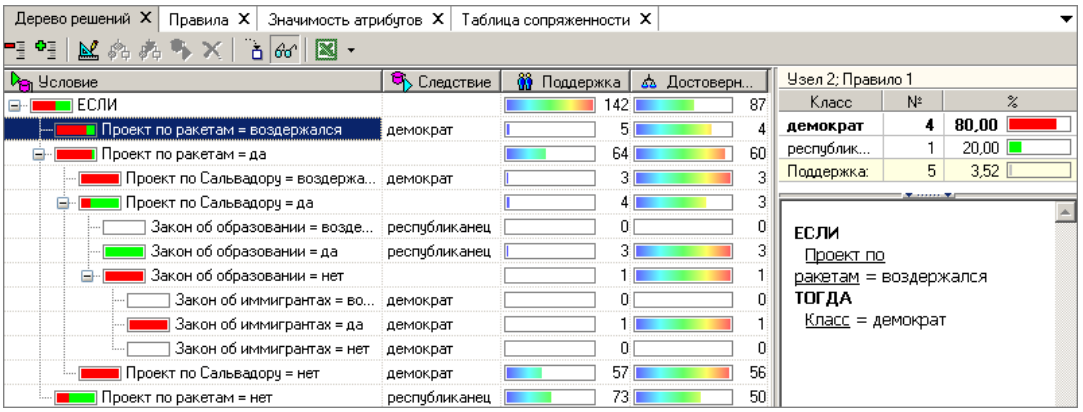
Но иногда аналитик считает правильным построить дерево решений исходя из своих соображений или внести некоторую корректировку, и тогда необходимо выбрать интерактивный режим построения, в результате чего получим следующее окно дерева решений.



Для внесения изменений в него используют следующие кнопки:

- включение/выключение интерактивного режима;
- разбить текущий узел на подузлы;
- построить дерево решений начиная с текущего узла.

Допустим, что аналитик думает, что основное правило которое надо учитывать в построение дерева решений есть проект о ракетах. Тогда для данного построения выберем корневой каталог в дереве решений и нажмем кнопку и в появившемся окне выберем проект по ракетам. В результате получим новое дерево решений с новыми правилами и законами.



Выводы

Пример показал простоту и удобство применения деревьев решений для классификации на республиканцев и демократов. Мастер предлагает широкие возможности по настройке процесса построения дерева решений. Это и настройка назначения столбцов, способов нормализации, настройка источника данных для учителя (тестовое и обучающее множества), настройка количества примеров в узле и настройка достоверности правил. После построения дерева стали видны его достоинства для анализа. Алгоритм сам отсек несущественные факторы, выявил степень влияния тех или иных факторов на результат, описал при помощи формальных правил способ классификации, а также выдал информацию о достоверности и поддержке того или иного правила. Также были продемонстрированы широкие возможности визуализации построенного дерева. Все это говорит о незаменимости деревьев решений для классификации.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 4
КЛАСТЕРИЗАЦИЯ С ПОМОЩЬЮ АЛГОРИТМА K-MEANS

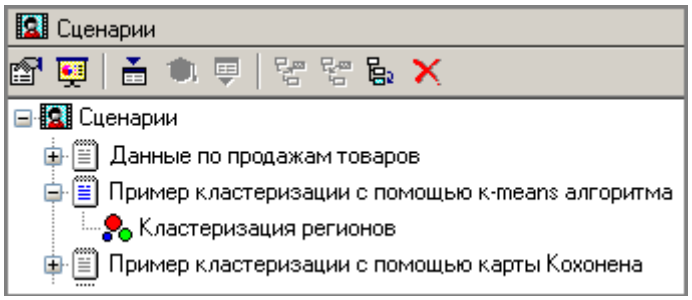
Задача кластеризации применяется для автоматического разбиения элементов некоторого множества на группы в зависимости от схожести их свойств. Данная задача выполняет предварительную подготовку данных для дальнейшего анализа каждой выявленной группы в отдельности. Задача кластеризации применяется для анализа общей структуры множества данных, упрощения анализа за счет рассматривания каждой группы в отдельности, а также сокращения объемов хранимых данных путем выбора наиболее индивидуальных представителей, выявления аномальных значений.

Сформированные подгруппы в задаче кластеризации применяются дальше в задачах классификации и прогнозирования.

Одним из наиболее распространенных и простых алгоритмов кластеризации является алгоритм k-means. Этот алгоритм основан на оптимизации суммы квадратов взвешенных отклонений координат объектов от центров искомых кластеров.

В Deductor Studio для автоматизации этого процесса есть соответствующий инструмент — "Кластеризация".

Рассмотрим фрагмент проекта "Демопример анализа данных.ded".



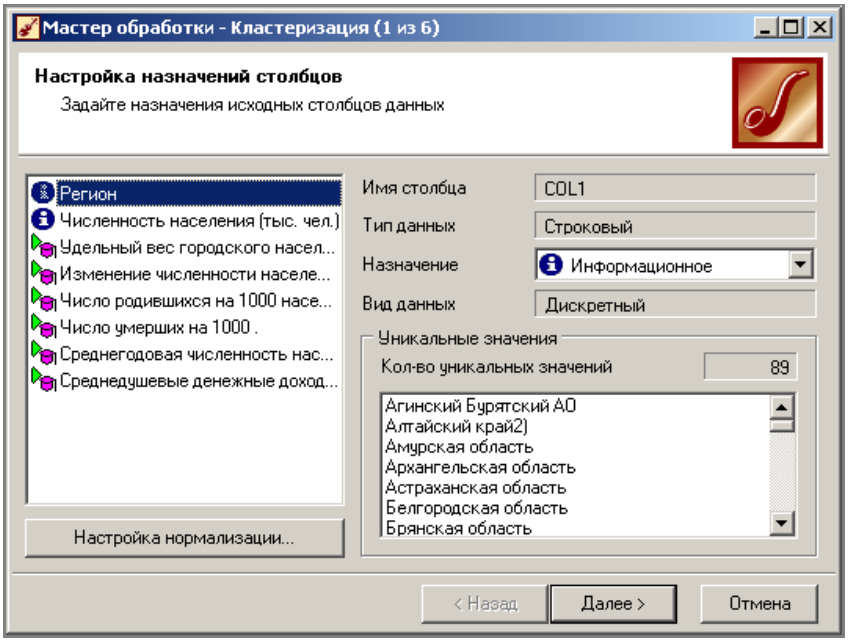
Исходные данные

Рассмотрим механизм кластеризации, реализованный на алгоритме k- means, основываясь на данных роста численности населения по регионам РФ за 2000 год. Исходная таблица находится в файле "Polution.txt". Задача состоит в распределении регионов на функциональные группы по демографической картине в них и выявлении скрытых закономерностей.

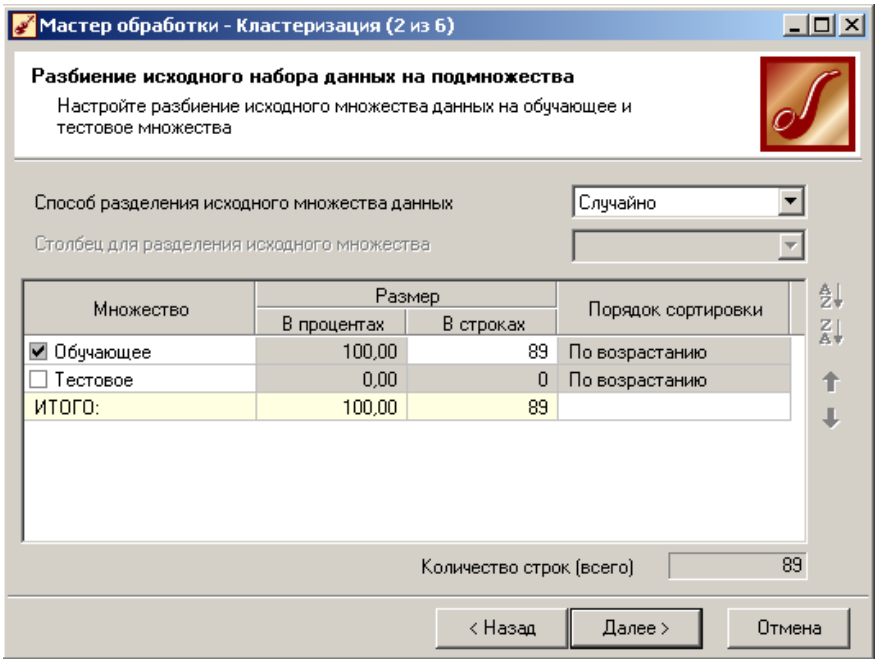
Кластеризация регионов

Вначале необходимо осуществить импорт рассматриваемых данных из файла.

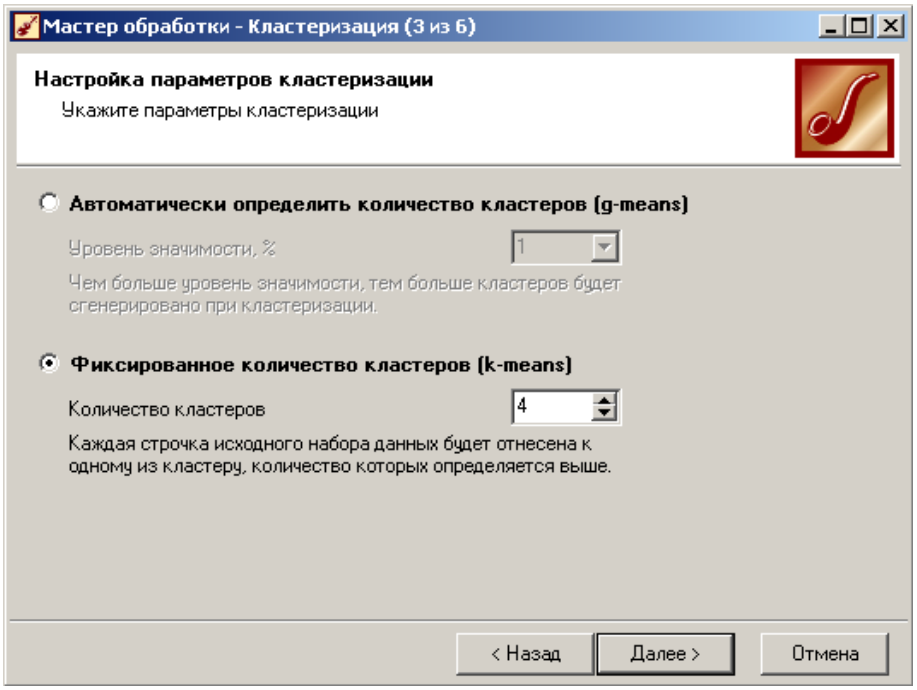
После этого выбираем и запускаем Мастер обработки "Кластеризация". При запуске Мастера необходимо настроить назначения столбцов, т.е. выбрать свойства, по которым будет происходить группировка объектов. Укажем столбцам "Численность населения " и "Регион" назначение "Информационное", а "Удельный вес городского населения", "Изменение численности населения", "Число родившихся на 1000", "Число умерших на 1000", "Среднегодовая численность населения занятых в экономике" , "Среднедушевой денежный доход" — "Входное".



На следующем шаге Мастера необходимо настроить способ разделения исходного множества данных на тестовое и обучающее, а также количество примеров в том и другом множестве. Укажем, что данные обоих множеств берутся случайным образом, и определим все множество как обучающее.

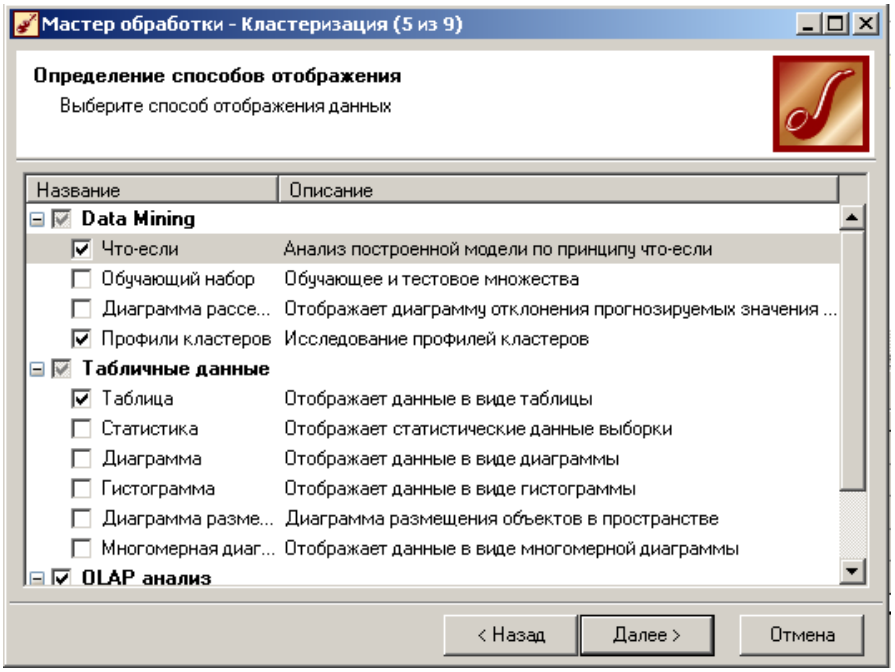


Следующий шаг предлагает настроить параметры кластеризации, определить на какое количество кластеров будет распределяться исходное множество. По мнению экспертов в стране наблюдается четыре тенденции развития регионов, поэтому выберем фиксированное количество кластеров равное четырем.

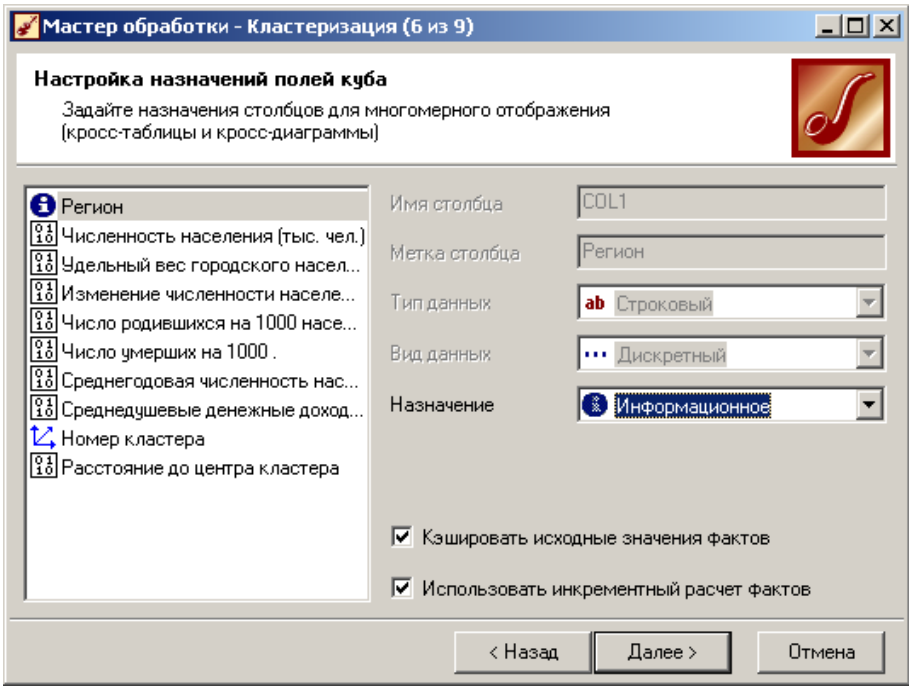


Результат

Для отображения полученных групп кластеров выберем в обработчике "Кластеризация" из списка визуализаторов способы отображения данных: "Что-если" для решения задачи классификации, отнесение региона к одному из кластеров, "Профили кластеров" для определения структуры формирования группы кластеров и "Куб" для наглядного просмотра полученных результатов.

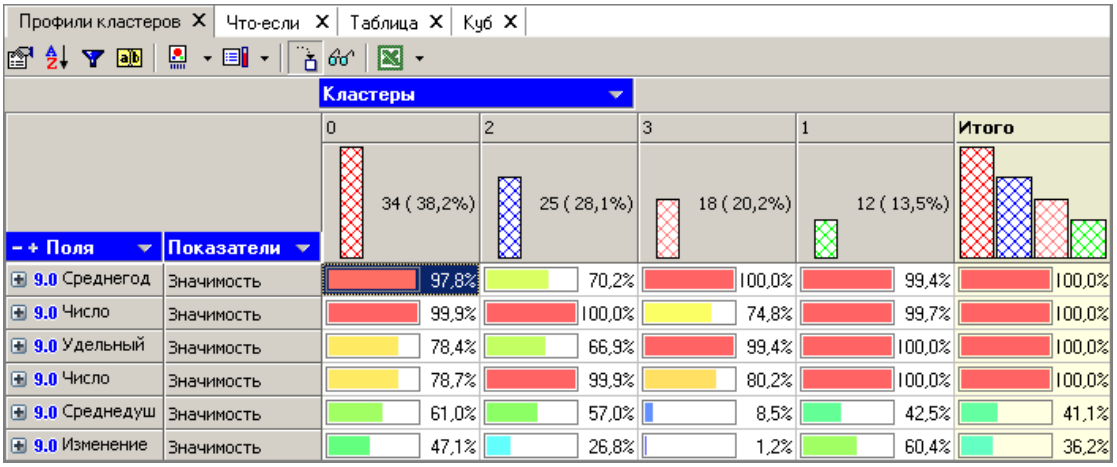


Для настройки визуализатора "Куб" необходимо выбрать рассматриваемые свойства как факты, а номер кластера и регионы как измерение. Наиболее правильно в дальнейших настройках задать отображение фактов как среднее по рассматриваемое группе.



Общую структуру сформированных алгоритмом кластеров можно просмотреть в визуализаторе "Профили кластеров". В нем представлены все рассматриваемые свойства вместе с характером влияния их на состав кластера. Основным определяющим состав кластера фактором является значимость свойств, выраженная в процентах. Общая значимость рассматриваемого поля определяется вариабельностью ее рассматриваемых параметров. Значимость для непрерывных и дискретных полей определяется по-разному.

Значимость для непрерывных полей устанавливается в зависимости от отклонения среднего значения рассматриваемой группы кластеров от общего среднего всей выборки, чем больше выражено данное отклонение, тем больше его значимость. Значимость для дискретных полей определяется наличием индивидуальных различий, между рассматриваемыми группами, чем больше выражены различия, тем больше значимость. Для каждого рассматриваемого свойства в кластере вычисляется: доверительный интервал, среднее, стандартное отклонение и стандартная ошибка.

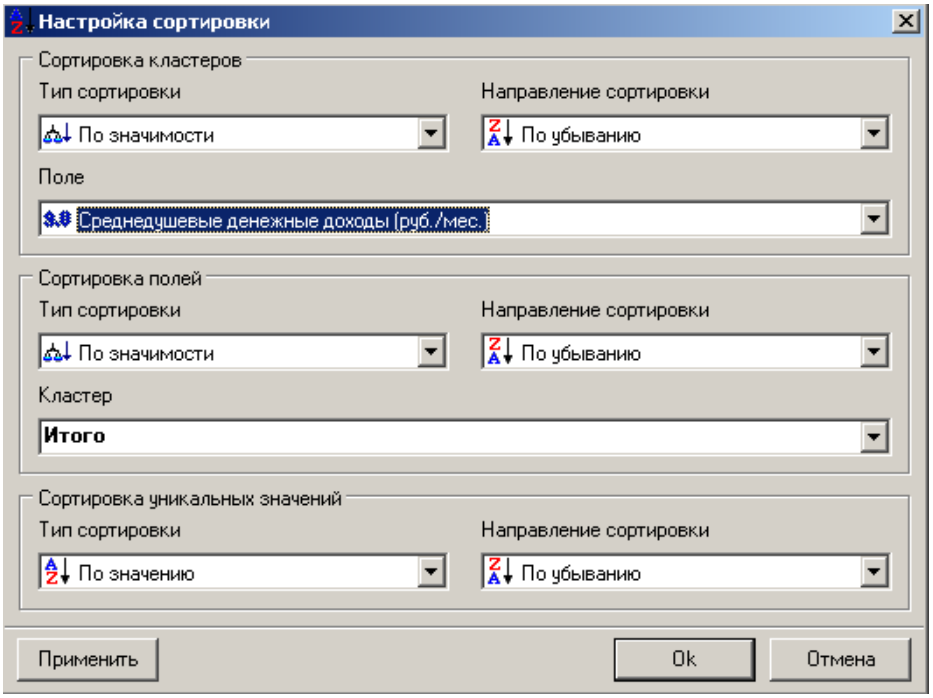


Алгоритм автоматически разбил регионы на четыре кластера с разной поддержкой и разными процентами значимости свойств. Первый кластер является показателем демографической обстановки страны, так как собрал в себя максимальное количество кластеров. Наиболее ярко выраженными кластерами по заданным свойствам является нулевой и третий кластер они мак-

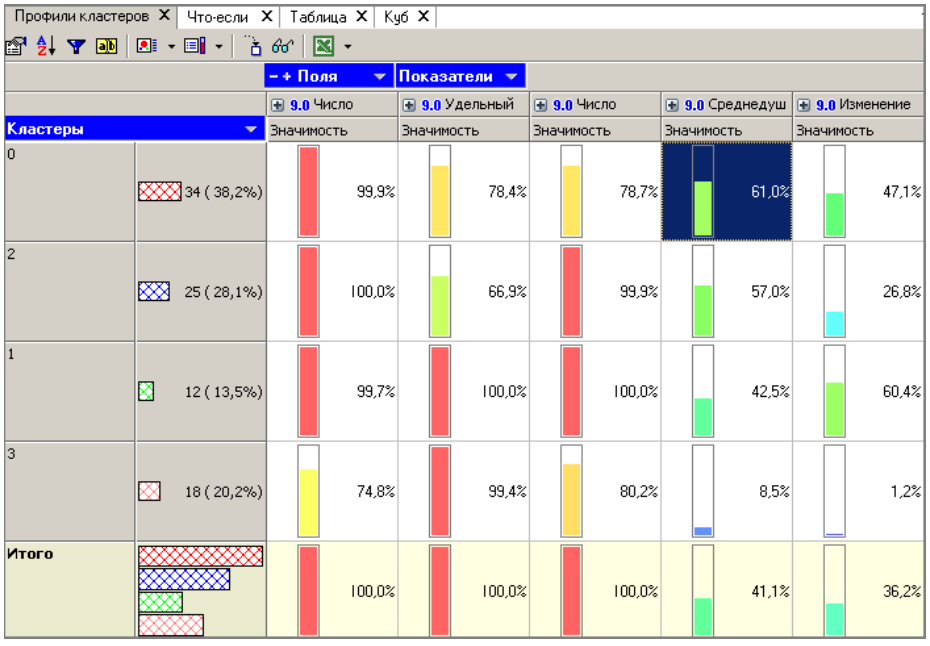
симально отличаются от остальных рассматриваемых групп значениями свойств, и минимальной поддержкой.

Малозначимым и почти не влияющим свойством на распределение является изменение численности населения по сравнению с предыдущим годом, при необходимости данным свойством можно пренебречь.

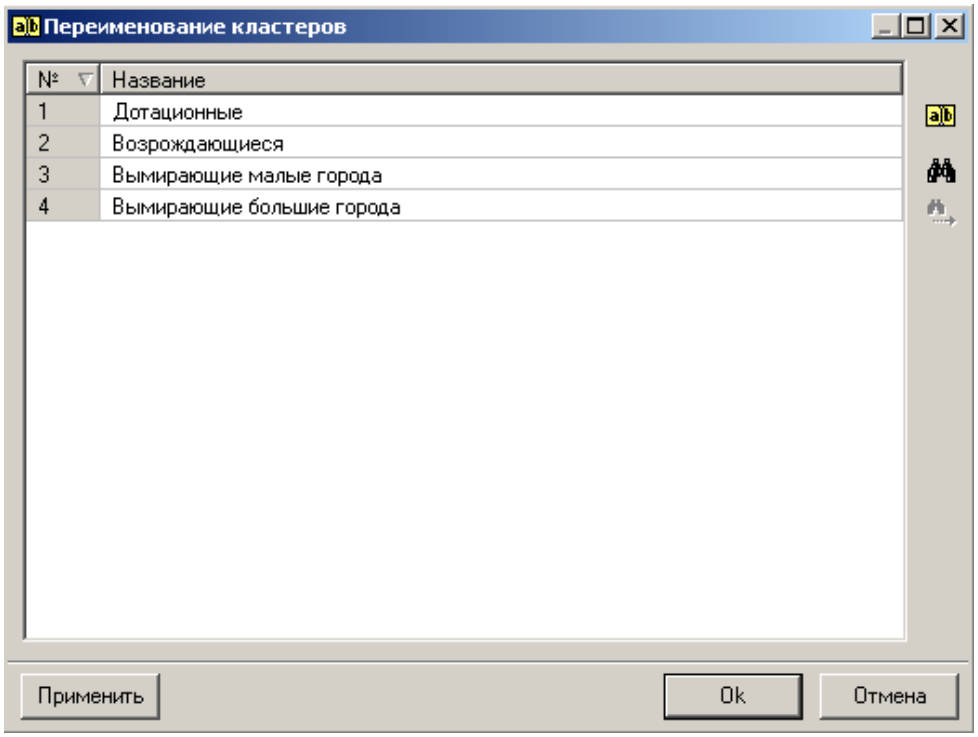
Определим кластеры, где самым значимым параметром является среднедушевой доход, для этого нажмем кнопку настройка сортировки на панели инструментов, и зададим параметры сортировки. Выберем тип сортировки по значимости, направление по убыванию и поле, по которому будем производить сортировку, остальное оставим без изменения.



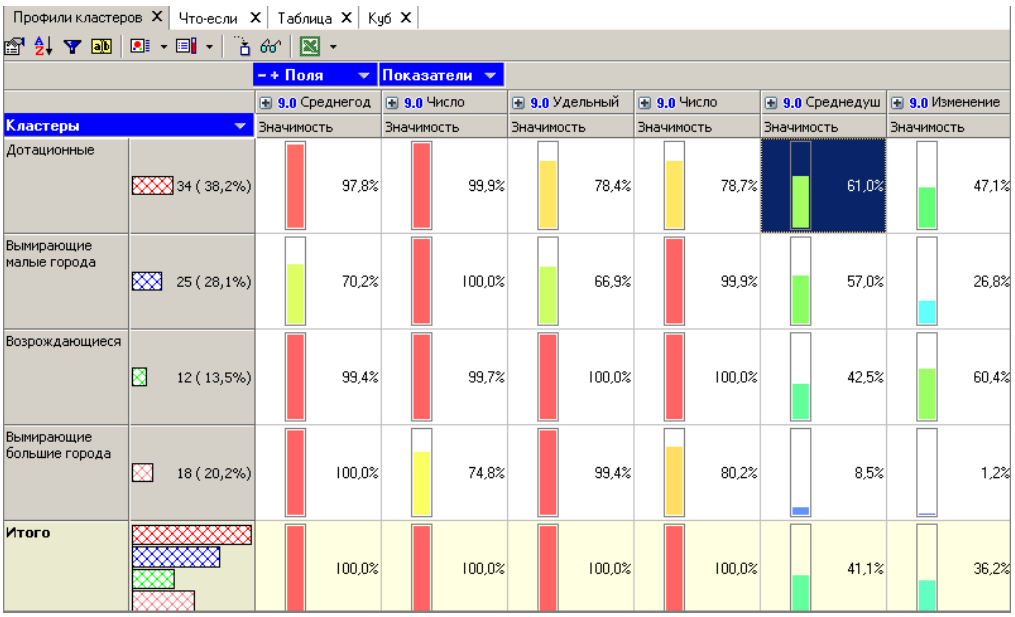
Кластеры поменялись местами в зависимости от значимости среднедушевого дохода в рассматриваемом наборе. Наиболее отличающиеся кластеры по среднедушевому годовому отчету будут иметь максимальную значимость.



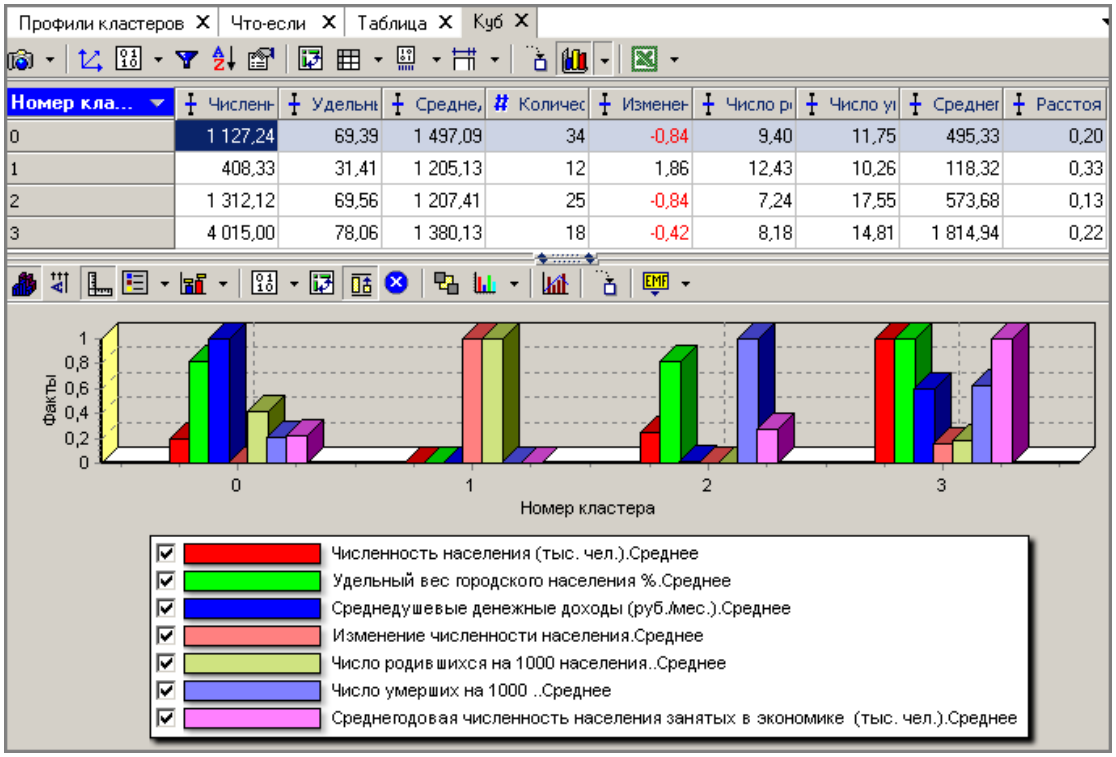
С помощью кнопки переименование кластеров можно им присвоить им рабочее название в профиле кластеров.



Вид профиля кластеров принимает вид



Результаты по сформированным кластерам наиболее удобно рассматриваются с помощью визуализатора "Куб", в котором встроена кросс- диаграмма, изображающая полученные кластеры в графическом виде, что существенно упрощает анализ.



Из полученной кросс-диаграммы видно, что все регионы разбились на четыре кластера, условно можно их назвать:

"Дотационные" — нулевой кластер; "Возрождающиеся" — первый кластер; "Вымирающие малые города" — второй кластер; "Вымирающие большие города" — третий кластер. В трех из четырех кластеров наблюдается картина того, что численность населения очень сильно падает, число умерших в несколько раз больше числа родившихся. Эти кластеры показывают демографическую обстановку в РФ, так как в их состав входит большая часть регионов страны. Имеется только один кластер, где положение дел более-менее хорошее, это первый кластер. На основе анализа демографической политики данного региона можно поднять рождаемость в стране.

Выводы

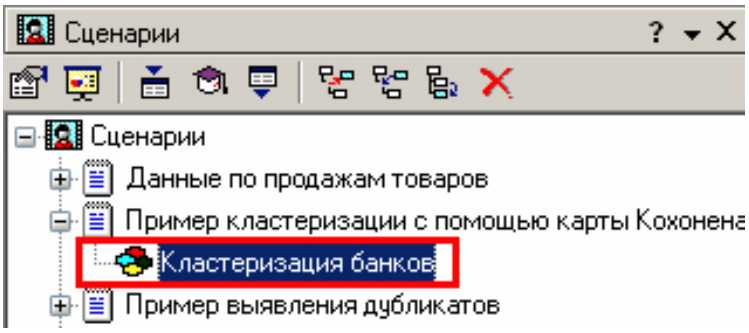
Рассмотренный пример проиллюстрировал, применение кластеризации для группового анализа данных. С помощью задачи кластеризации все регионы сгруппировались на кластеры по параметрам входных значений, интерпретация которых осуществляется с помощью кросс-диаграммы и куба. Но кажущаяся простота задачи кластеризации обманчива, она требует полной собранности аналитика при анализе полученных результатов и наличии чувства интуиции. Именно аналитик решает на сколько кластеров необходимо разбить исследуемый набор данных и какие свойства будут основными при построении кластера, т.е. аналитик закладывает фундамент решении задачи. Но это не все проблемы, связанные с задачей кластеризации одной из особенностей применения k-means алгоритма, а так же и многих других является, то что при повторном построении задачи кластеризации можно не получить одинакового результата, это связано с тем что данные очень разрозненные и алгоритм выбирает случайным образом центры кластеров.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 5
КЛАСТЕРИЗАЦИЯ С ПОМОЩЬЮ САМООРГАНИЗУЮЩЕЙСЯ КАРТЫ КОХОНЕНА

Самоорганизующаяся карта Кохонена является разновидностью нейронной сети. Она применяется, когда необходимо решить задачу кластеризации, т.е. распределить данные по нескольким кластерам. Алгоритм определяет расположение кластеров в многомерном пространстве факторов. Исходные данные будут относиться к какому-либо кластеру в зависимости от расстояния до него. Многомерное пространство трудно для представления в графическом виде. Механизм же построения карты Кохонена позволяет отобразить многомерное пространство в двумерном, которое более удобно и для визуализации, и для интерпретации результатов аналитиком.

Также с помощью построенной карты Кохонена можно решить и задачу прогнозирования. В этом случае результирующее поле (то, которое необходимо спрогнозировать) в построении карты не участвует. После кластеризации, используя диаграмму "Что-если", можно провести эксперимент. Алгоритм определяет точку пространства, где расположены введенные для прогноза данные и к какому кластеру принадлежит данная точка, и подсчитывает среднее по результирующему полю всех точек этого кластера, что и будет результатом прогноза (для дискретных данных результатом прогноза является значение, больше всего встречающееся в результирующем поле всех ячеек кластера).

Рассмотрим фрагмент проекта "Демопример анализа данных.ded".



Исходные данные

Рассмотрим механизм кластеризации путем построения самоорганизующейся карты, основываясь на информации по банкам.

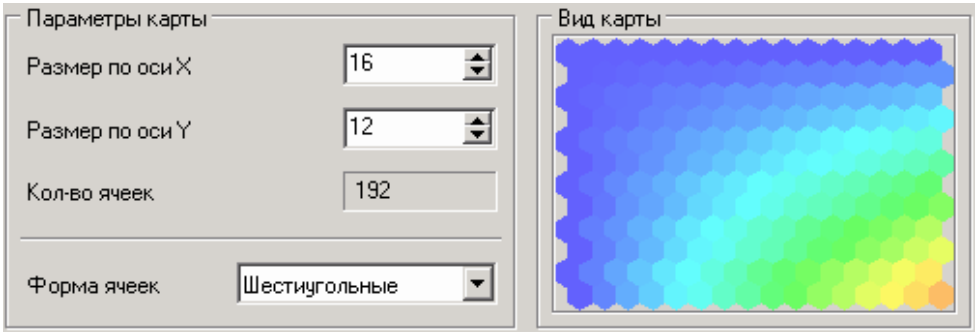
Исходная таблица находится в файле "Banks.txt". Задача состоит в том, чтобы определить по различным данным банка его прибыль и наличие скрытых закономерностей.

Кластеризация банков

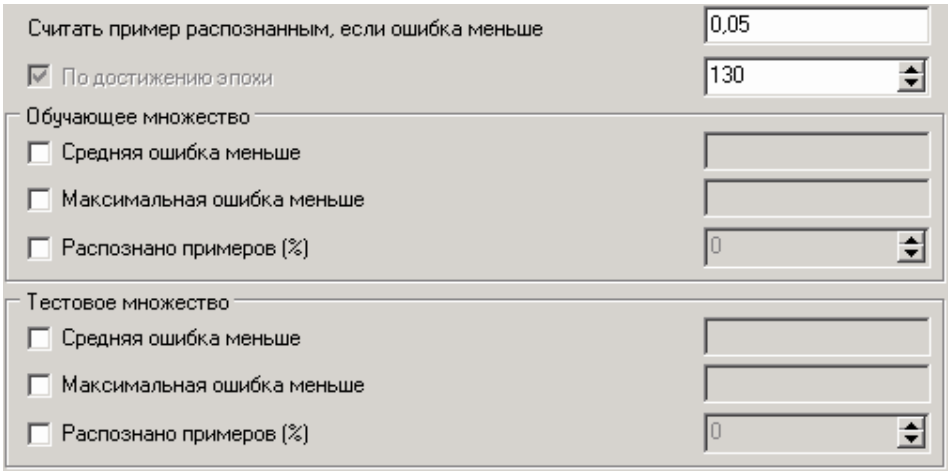
Для начала необходимо импортировать данные из файла. После этого запустим Мастер обработки и выберем из списка метод обработки "Карта Кохонена". На втором шаге Мастера настроим назначения столбцов. Укажем столбцу "Прибыль" назначение "Выходной", а "Филиалы", "Сумма активов", "Собственные активы", "Банковские активы", "Средства в банке" – "Входной", т. е. на основе данных о банке будем относить его к тому или иному классу.

На третьем шаге Мастера необходимо настроить способ разделения исходного множества данных на тестовое и обучающее, а также количество примеров в том и другом множестве. Укажем, что данные обоих множеств берутся случайным образом, а остальные значения оставим без изменений.

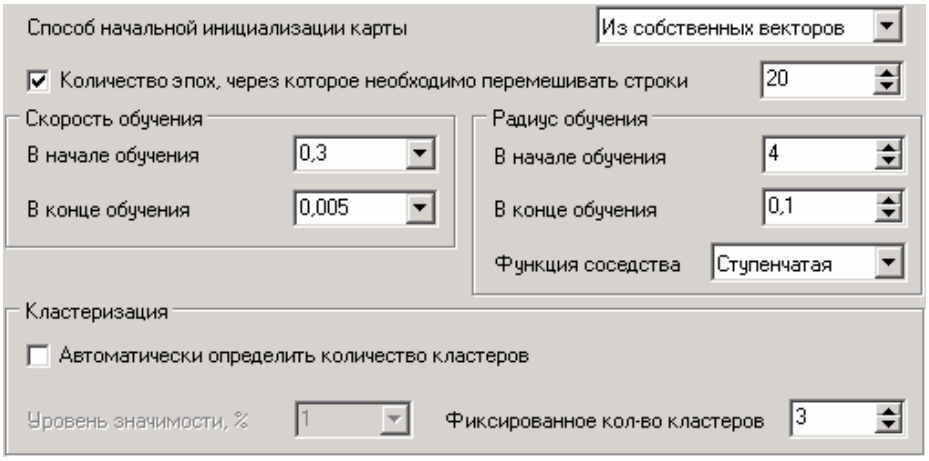
Следующий шаг предлагает настроить параметры карты (Количество ячеек по X и по Y, их форму) и параметры обучения (способ начальной инициализации, тип функции соседства, перемешивать ли строки обучающего множества и количество эпох, через которые необходимо перемешивание). Значения по умолчанию вполне подходят.



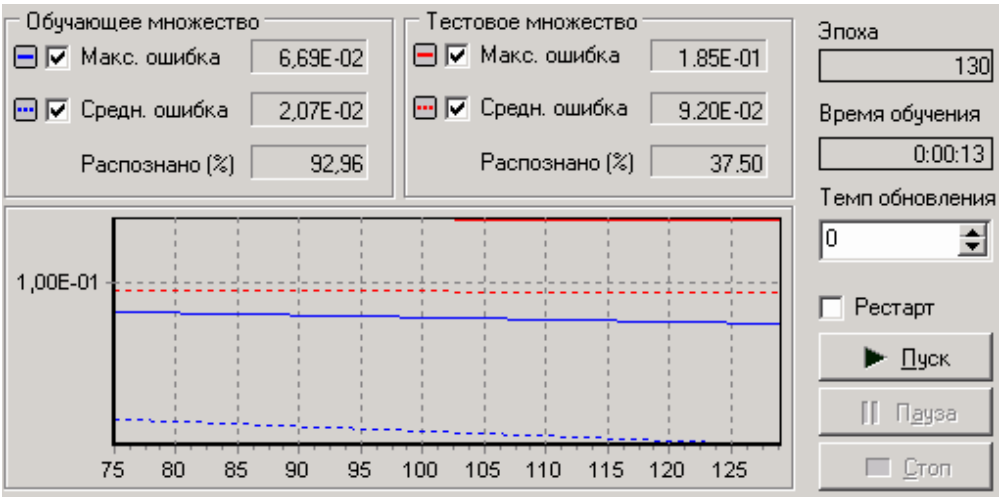
На пятом шаге Мастера следует настроить параметры остановки обучения. Оставим параметры по умолчанию.



На шестом шаге настраиваются остальные параметры обучения: способ начальной инициализации, тип функции соседства, а также параметры кластеризации — автоматическое определение числа кластеров с соответствующим уровнем значимости либо фиксированное количество кластеров.



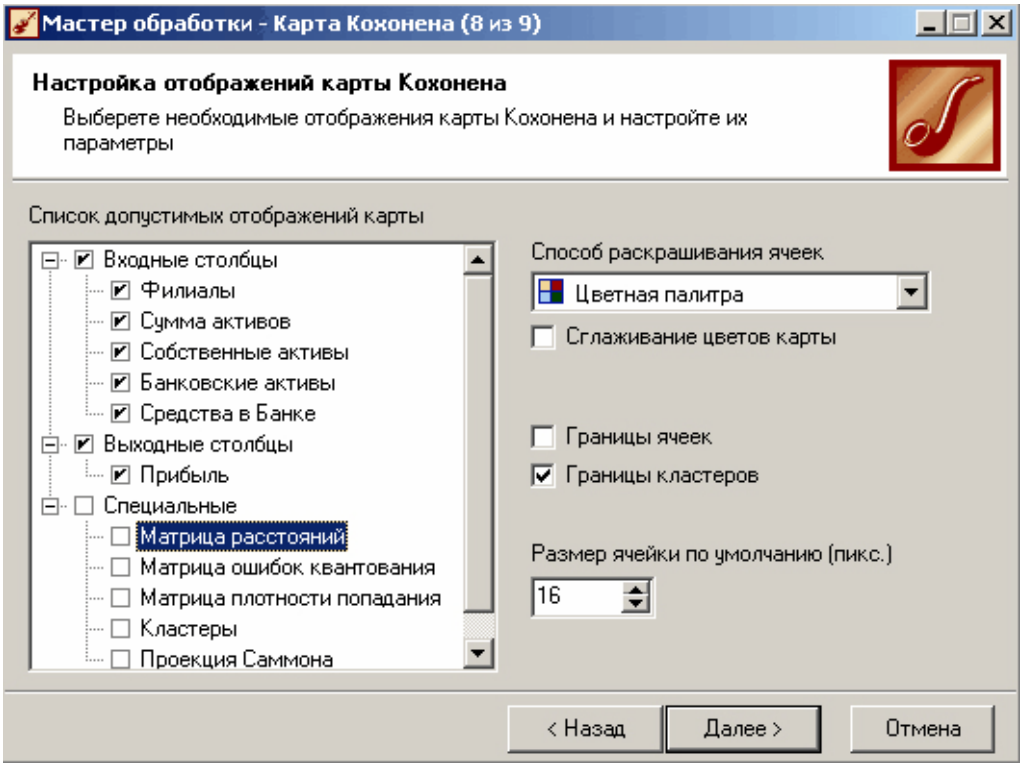
На седьмом шаге предлагается запустить сам процесс обучения. Во время обучения можно посмотреть количество распознанных примеров и текущие значения ошибок. Здесь нужно нажать на кнопку "Пуск" и дождаться завершения процесса обработки.



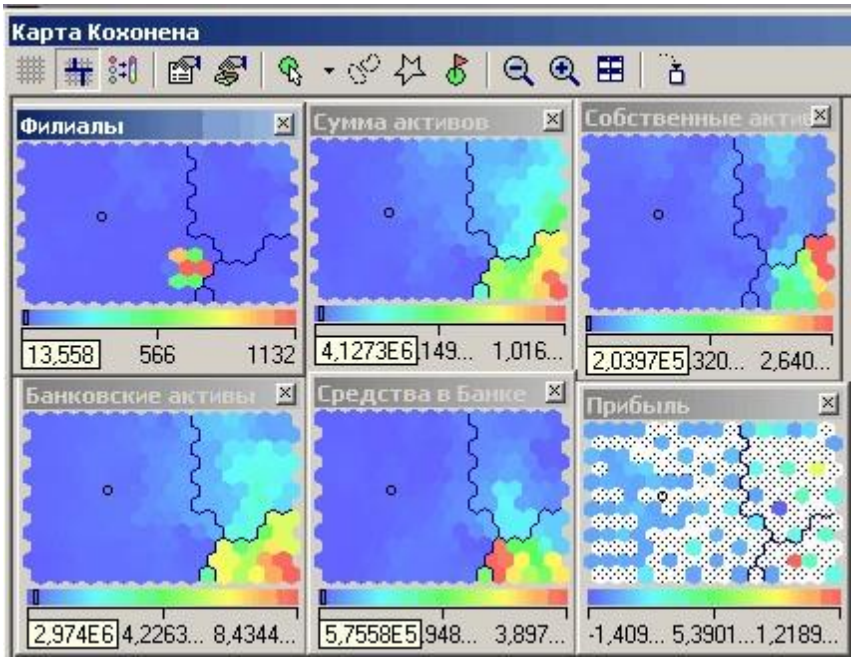
Результат

После этого требуется в списке визуализаторов выбрать появившуюся теперь Карту Кохонена для просмотра результатов кластеризации, а также визуализатор "Что-если" для прогнозирования прибыли банков.

Далее в Мастере настройки отображения карты Кохонена надлежит указать поля, которые необходимы для отображения.



В итоге получаем Карту Кохонена.



Можно видеть, что наиболее прибыльные банки попали в кластеры, что находятся в правой части карты. Для этих банков характерны большая сумма активов и средств в банке. Количество же филиалов не оказывает существенного влияния на прибыльность, т.к. банки с большим количеством филиалов разместились в левом не самом прибыльном кластере (см. проекцию "Филиалы").

Выводы

Данный пример показал область применения самоорганизующихся карт. Изначально имелось многомерное (четырехмерное) пространство входных факторов. Алгоритм представил его в двумерном виде, который удобнее анализировать. Основным визуализатором после построения является "Самоорганизующаяся карта".

Мастер предоставляет широкий набор настроек параметров обучения: настройка нормализации столбцов, настройка разбиения на тестовое и обучающее множество, настройка условий остановки обучения, настройка параметров карты и параметров обучения.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 6
ПОИСК АССОЦИАТИВНЫХ ПРАВИЛ

Ассоциативные правила позволяют находить закономерности между связанными событиями. Примером такого правила служит утверждение, что покупатель, приобретающий "Хлеб", приобретет и "Молоко". Впервые эта задача была предложена для поиска ассоциативных правил для нахождения типичных шаблонов покупок, совершаемых в супермаркетах, поэтому иногда ее еще называют анализом рыночной корзины (market basket analysis).

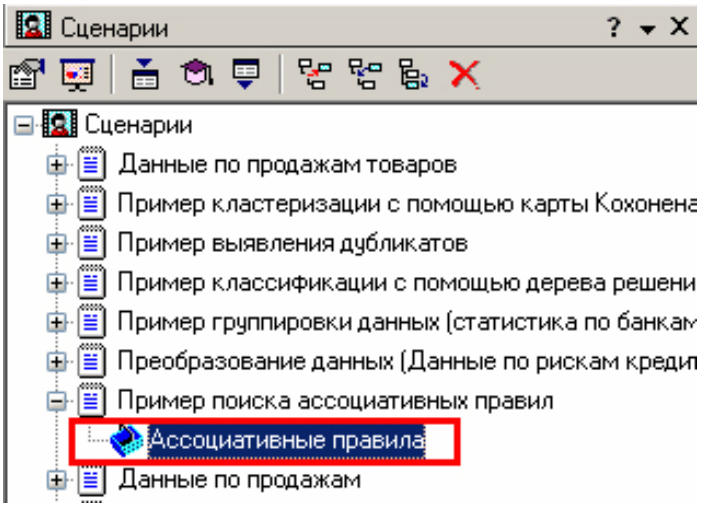
Пусть имеется база данных, состоящая из покупательских транзакций. Каждая транзакция — это набор товаров, купленных покупателем за один визит. Такую транзакцию еще называют рыночной корзиной. Целью анализа является установление следующих зависимостей: если в транзакции встретился некоторый набор элементов X, то на основании этого можно сделать вывод о том, что другой набор элементов Y также должен появиться в этой транзакции. Установление таких зависимостей дает нам возможность находить очень простые и интуитивно понятные правила.

Основными характеристиками таких правил являются поддержка и достоверность. Правило "Из X следует Y" имеет поддержку s, если s% транзакций из всего набора содержат наборы элементов X и Y. Достоверность правила показывает, какова вероятность того, что из X следует Y. Правило "Из X следует Y" справедливо с достоверностью c, если c% транзакций из всего множества, содержащих набор элементов X, также содержат набор элементов Y. Покажем на конкретном примере: пусть 75% транзакций, содержащих хлеб, также содержат молоко, а 3% от общего числа всех транзакций содержат оба товара. 75% — это достоверность правила, а 3% — это поддержка.

Алгоритмы поиска ассоциативных правил предназначены для нахождения всех правил вида "из X следует Y", причем поддержка и достоверность этих правил должны находиться в рамках некоторых наперед заданных границ, называемых соответственно минимальной и максимальной поддержкой и минимальной и максимальной достоверностью.

Границы значений параметров поддержки и достоверности выбираются таким образом, чтобы ограничить количество найденных правил. Если поддержка имеет большое значение, то алгоритмы будут находить правила, хорошо известные аналитикам или настолько очевидные, что нет никакого смысла проводить такой анализ. С другой стороны, низкое значение поддержки ведет к генерации огромного количества правил, что, конечно, требует существенных вычислительных ресурсов. Тем не менее, большинство интересных правил находится именно при низком значении порога поддержки, хотя слишком низкое значение поддержки ведет к генерации статистически необоснованных правил. Таким образом, необходимо найти компромисс, обеспечивающий, во-первых, интересность правил и, во-вторых, их статистическую обоснованность. Поэтому значения этих границ напрямую зависят от характера анализируемых данных и подбираются индивидуально. Еще одним параметром, ограничивающим количество найденных правил, является максимальная мощность часто встречающихся множеств. Если этот параметр указан, то при поиске правил будут рассматриваться только множества, количество элементов которых будет не больше данного параметра.

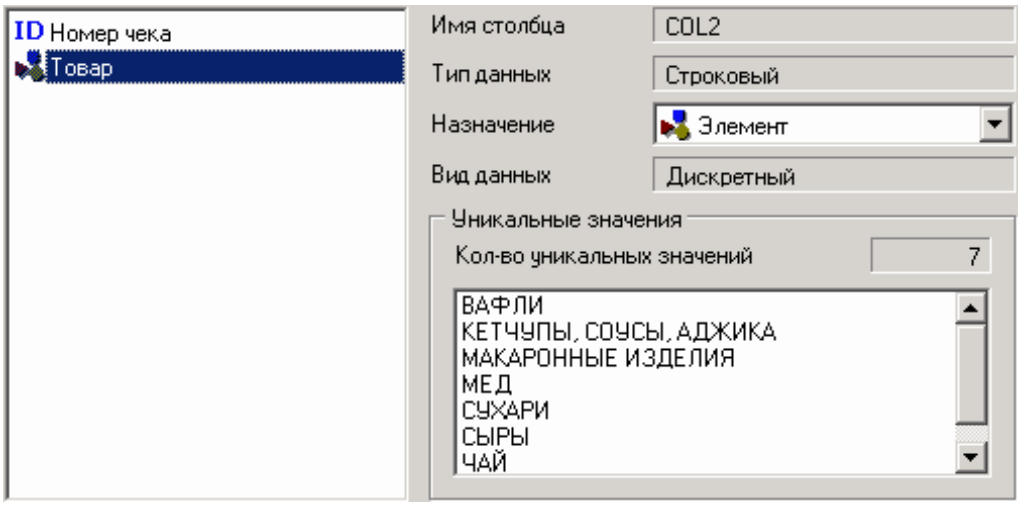
Рассмотрим фрагмент проекта "Демопример анализа данных.ded".



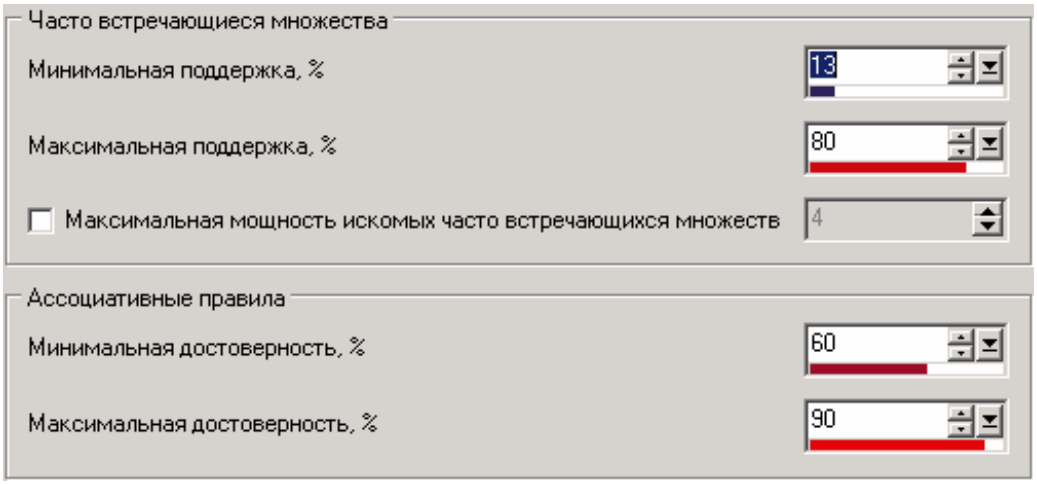
Исходные данные

Рассмотрим механизм поиска ассоциативных правил на примере данных о продажах товаров в некоторой торговой точке. Данные находятся в файле "Supermarket.txt". В таблице представлена информация по покупкам продуктов нескольких групп. Она имеет всего два поля "Номер чека" и "Товар". Необходимо решить задачу анализа потребительской корзины с целью последующего применения результатов для стимулирования продаж. Поиск ассоциативных правил

Для поиска ассоциативных правил запустим Мастер обработки. В нем выберем тип обработки "Ассоциативные правила". На втором шаге Мастера следует указать, какой столбец является идентификатором транзакции (чек), который должен быть дискретным, а какой элементом транзакции (товар).



Следующий шаг позволяет настроить параметры построения ассоциативных правил: минимальную и максимальную поддержку, минимальную и максимальную достоверность, а также максимальную мощность множества. Исходя из характера имеющихся данных, следует указать границы поддержки – 13% и 80% и достоверности 60% и 90%.



Следующий шаг позволяет запустить процесс поиска ассоциативных правил. На экране отображается информация о количестве множеств и найденных правил, а также числе часто встречающихся множеств.

После завершения процесса поиска полученные результаты можно посмотреть, используя появившиеся специальные визуализаторы "Популярные наборы", "Правила", "Дерево правил", "Что-если".

Популярные наборы — это множества, состоящие из одного и более элементов, которые наиболее часто встречаются в транзакциях одновременно. На сколько часто встречается множество в исходном наборе транзакций, можно судить по поддержке. Данный визуализатор отображает множества в виде списка.

№	Номе	Элементы	Поддержка		Мощность
			Кол-во	%	
1	1	ВАФЛИ	14	31,82	1
2	2	КЕТЧУПЫ, СОУСЫ, АДЖИКА	23	52,27	1
3	3	МАКАРОННЫЕ ИЗДЕЛИЯ	24	54,55	1
4	4	МЕД	22	50,00	1
5	5	СУХАРИ	14	31,82	1
6	6	СЫРЫ	19	43,18	1
7	7	ЧАЙ	33	75,00	1
8	8	ВАФЛИ МЕД	6	13,64	2
9	9	ВАФЛИ СУХАРИ	10	22,73	2
10	10	ВАФЛИ ЧАЙ	13	29,55	2

Само название визуализатора говорит о том, как применить данные результаты на практике. Получившиеся наборы товаров наиболее часто покупают в данной торговой точке, следовательно можно принимать решения о поставках товаров, их размещении и т.д.

Результат

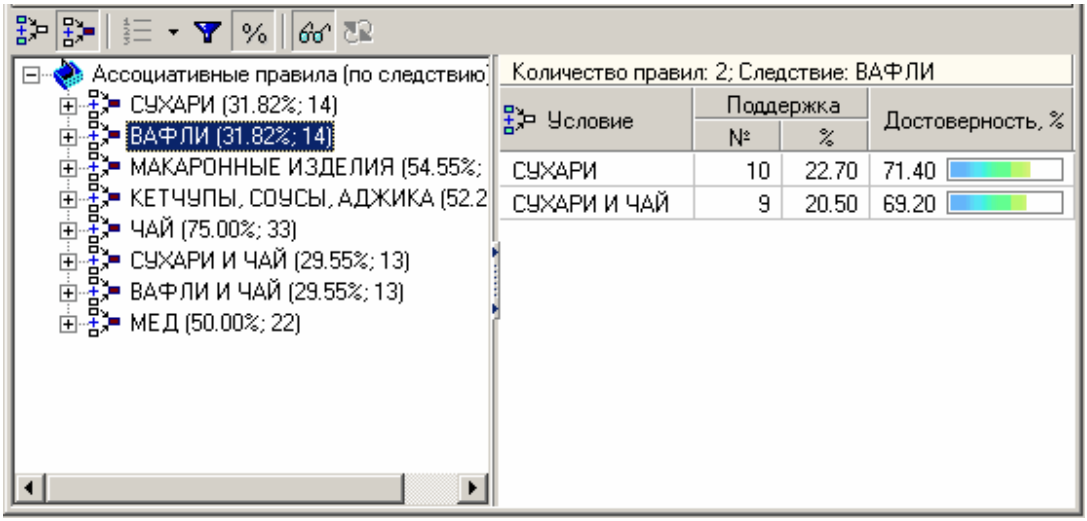
Визуализатор "Правила" отображает ассоциативные правила в виде списка правил. Этот список представлен таблицей со столбцами: "Номер правила", "Условие", "Следствие", "Поддержка, %", "Поддержка, Количество", "Достоверность", "Лифт".

№	Нс	Условие	Следствие	Поддержка		Достоверность	Лифт
				Кол-во	%		
1	1	ВАФЛИ	СУХАРИ	10	22,73	71,43	2,245
2	2	СУХАРИ	ВАФЛИ	10	22,73	71,43	2,245
3	3	КЕТЧУПЫ, СОУСЫ, АДЖИКА	МАКАРОННЫЕ ИЗДЕЛИЯ	20	45,45	86,96	1,594
4	4	МАКАРОННЫЕ ИЗДЕЛИЯ	КЕТЧУПЫ, СОУСЫ, АДЖИК	20	45,45	83,33	1,594
5	5	МЕД	ЧАЙ	18	40,91	81,82	1,091

Таким образом, эксперту предоставляется набор правил, которые описывают поведение покупателей. Например, если покупатель купил вафли, то он с вероятностью 71% также купит и сухари. Меру значимости данного ассоциативного правила показывает лифт, чем величина лифта выше, тем более значимо данное правило по сравнению с его аналогами.

Визуализатор "Дерево правил" — это всегда двухуровневое дерево. Оно может быть построено либо по условию, либо по следствию. При построении дерева правил по условию на первом (верхнем) уровне находятся узлы с условиями, а на втором уровне — узлы со следствием. Второй вариант дерева правил — дерево, построенное по следствию. Здесь на первом уровне располагаются узлы со следствием.

Справа от дерева находится список правил, построенный по выбранному узлу дерева. Для каждого правила отображаются поддержка и достоверность. Если дерево построено по условию, то вверху списка отображается условие правила, а список состоит из его следствий. Тогда правила отвечают на вопрос, что будет при таком условии. Если же дерево построено по следствию, то вверху списка отображается следствие правила, а список состоит из его условий. Эти правила отвечают на вопрос, что нужно, чтобы было заданное следствие. Данный визуализатор отображает те же самые правила, что и предыдущий, но в более удобной для анализа форме.



В данном случае правила отображены по условию. Тогда отображаемый в данный момент результат можно интерпретировать как 2 правила:

1. Если покупатель приобрел вафли, то он с вероятностью 71% также приобретет сухари.
2. Если покупатель приобрел вафли, то он с вероятностью 64% также приобретет сухари и чай.

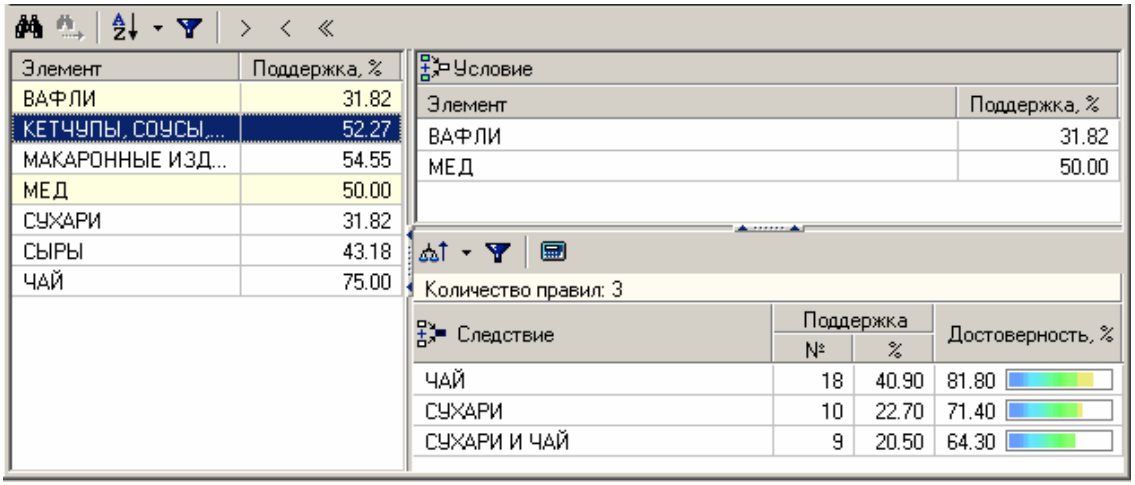
Аналогично интерпретируются и остальные правила.

Анализ "Что–если" в ассоциативных правилах позволяет ответить на вопрос, что получим в качестве следствия, если выберем данные условия? Например, какие товары приобретаются совместно с выбранными товарами. В окне слева расположен список всех элементов транзакций. Справа от каждого элемента указана поддержка: сколько раз данный элемент встречается в транзакциях.

В правом верхнем углу расположен список элементов, входящих в условие. Это, например, список товаров, которые приобрел покупатель. Для них нужно найти следствие. Например, товары, приобретаемые совместно с ними. Чтобы предложить человеку то, что он возможно забыл купить.

В правом нижнем углу расположен список следствий. Справа от элементов списка отображается поддержка и достоверность.

Пусть необходимо проанализировать, что, возможно, забыл покупатель приобрести, если он уже взял вафли и мед. Для этого следует добавить в список условий эти товары (например, с помощью двойного щелчка мыши) и затем нажать на кнопку "Вычислить правила". При этом в списке следствий появятся товары, совместно приобретаемые с данными. В данном случае появятся "Сухари", "Чай", "Сухари и чай", т. е., может быть, покупатель забыл приобрести сухари, чай или и то и другое.



Выводы

Как показал данный пример, результаты анализа можно применить и для сегментации покупателей по поведению при покупках, и для анализа предпочтений клиентов, и для планирования расположения товаров в супермаркетах, кросс-маркетинге. Предлагаемый набор визуализаторов позволяет эксперту найти интересные, необычные закономерности, понять, почему так происходит, и применить их на практике.

В данном примере найденные правила можно использовать для сегментации клиентов на два сегмента: клиенты, покупающие макаронные изделия и соусы к ним, и клиенты, покупающие все к чаю. В разрезе анализа предпочтений можно узнать, что наибольшей популярностью в данном магазине пользуются чай, мед, макаронные изделия, кетчупы, соусы и аджика. В разрезе размещения товаров в супермаркете можно применить результаты предыдущих двух анализов, т. е. располагать чай рядом с медом, а кетчупы, соусы и аджику рядом с макаронными изделиями и т.д.