

## **ЛАБОРАТОРНАЯ РАБОТА 9. ПРОГРАММА ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ WEKA**

### **ЦЕЛЬ И ЗАДАЧИ РАБОТЫ**

Целью лабораторной работы является формирование и закрепление навыков по работе с системой анализа данных WEKA.

Задачи:

1. Получить общие теоретические сведения о платформе WEKA.
2. Получить навыки по установке WEKA.
3. Ознакомиться с форматами входных данных.
4. Получить навыки решения задач регрессионного анализа, задач классификации и кластеризации.

## Платформа WEKA - общее описание

Платформа WEKA (Waikato Environment for Knowledge Analysis) была разработана в 1993 году в университете Уайкато (Новая Зеландия) и является широко известной, свободно распространяемой системой анализа данных, написанной на языке Java.

Целью проекта являлось создание современной среды разработки средств машинного обучения и применения их к реальным данным, создание доступных для повсеместного применения средств машинного обучения. Предполагается, что с помощью данной среды специалист в прикладной области сможет использовать методы машинного обучения для извлечения полезных знаний из данных, возможно, очень большого объёма. Пользователями WEKA являются исследователи в области машинного обучения и прикладных наук. WEKA также широко используется в учебных целях.

WEKA включает в себя библиотеку алгоритмов машинного обучения для решения задач интеллектуального анализа данных (Data Mining). Основными возможностями платформы являются: предобработка данных (Data preprocessing), классификация, кластеризация, выбор атрибутов данных, визуализация данных. Помимо этого WEKA поддерживает поиск ассоциативных правил, формирование наборов исходных данных, средства автоматической документации исходного кода и т.д.

По данным официального сайта WEKA на сегодняшний момент существует уже 50 проектов, расширяющих или использующих функциональность платформы (YALE, FastKMeans, FAENIM и др.).

## Практическое использование WEKA

Использование в медицине: ученые Г. Партибан, А. Раджеш и С. К. Криваца разработали систему прогнозирования болезней сердца для людей, страдающих сахарным диабетом. Система основана на методе NBC, в качестве входных данных были взяты следующие показатели: возраст, пол, наследственность, кровяное давление и уровень сахара в крови; система возвращала вероятность заболевания сердца. Исходные данные были собраны в одном из ведущих исследовательских институтов Ченнаи и содержали данные о 500 пациентах. Анализ данных производился с использованием программы WEKA. После обучения Байесовской модели, система смогла правильно классифицировать 74% пациентов. Результаты показывают, что предложенный метод хорошо работает по сравнению с аналогами, особенно учитывая тот факт, что атрибуты, взятые в качестве входных данных, не являются прямыми показателями болезни сердца.

Предсказание производительности работников: в настоящее время все крупные компании уделяют большое внимание качеству трудового капитала, работодатели заинтересованы в найме высококвалифицированного персонала, который, как ожидается, также будет высокоэффективным. Ученые К. Радайдех и И. Наги, используя WEKA, разработали универсальную модель прогнозирования производительности потенциального сотрудника, которая позволяет избежать риска, связанного с наймом недостаточно квалифицированного работника.

В последнее время появились различные виды финансового мошенничества, например: мошенничество с кредитными картами, отмывание денег, подделка отчетных документов и т.д. Далеко не все аудиторы во время ревизии могут обнаружить подобные аферы, поэтому возникла необходимость использования средств интеллектуального анализа данных. Ученые Х. С. Кохом и С. К. Лоу чтобы определить скрытые проблемы в финансовых отчетах, построили модель на основе дерева решений и использовали в ней следующие показатели: коэффициент текущей ликвидности, рыночная стоимость собственного капитала к совокупным активам, общая сумма

обязательств к совокупным активам, чистый доход и нераспределенная прибыль.

## Установка

Скачать [WEKA](http://www.cs.waikato.ac.nz/ml/weka/downloading.html) можно на официальном сайте проекта по адресу: <http://www.cs.waikato.ac.nz/ml/weka/downloading.html>.

Пролистайте страницу вниз до описания стабильных версий WEKA для разных ОС (см. Рис. 9.1). Так как WEKA написана на Java, ей нужно JRE (Java Runtime Environment - минимальная реализация виртуальной машины, необходимая для исполнения Java-приложений), поэтому если на вашем компьютере нет JRE, выберите для установки версию WEKA, включающую в себя JRE.

• **Stable version**

Weka 3.8 is the latest stable version of Weka. This branch of Weka receives bug fixes only, although new features may become available in packages. There are different options for downloading and installing it on your system:

- **Windows**
  - Click [here](#) to download a self-extracting executable for 64-bit Windows that includes Oracle's 64-bit Java VM 1.8 (weka-3-8-1-jre-x64.exe, 112.0 MB)
  - Click [here](#) to download a self-extracting executable for 64-bit Windows without a Java VM (weka-3-8-1-x64.exe, 50.6 MB)
  - Click [here](#) to download a self-extracting executable for 32-bit Windows that includes Oracle's 32-bit Java VM 1.8 (weka-3-8-1-jre.exe, 106.4 MB)
  - Click [here](#) to download a self-extracting executable for 32-bit Windows without a Java VM (weka-3-8-1.exe, 50.6 MB)
- **Mac OS X**
  - Click [here](#) to download a disk image for OS X that contains a Mac application including Oracle's Java 1.8 JVM (weka-3-8-1-oracle-jvm.dmg, 123.8 MB)
- **Other platforms (Linux, etc.)**
  - Click [here](#) to download a zip archive containing Weka (weka-3-8-1.zip, 50.9 MB)

Рис. 9.1 Стабильные версии WEKA

После скачивания запустите ехе-файл и начните установку (см. Рис. 5.2).



Рис. 9.2 Окно Setup Wizard

Согласитесь с условиями лицензии (см. Рис. 9.3).

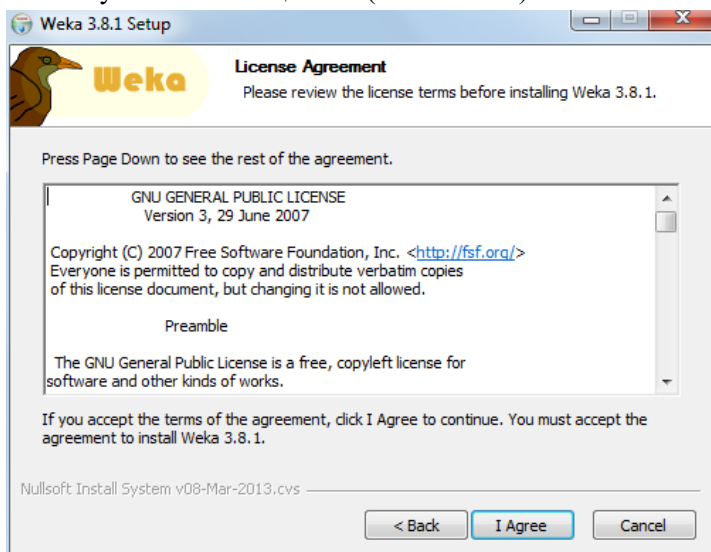


Рис. 9.3 Условия лицензии

Выберите компоненты для установки (см. Рис. 9.4).

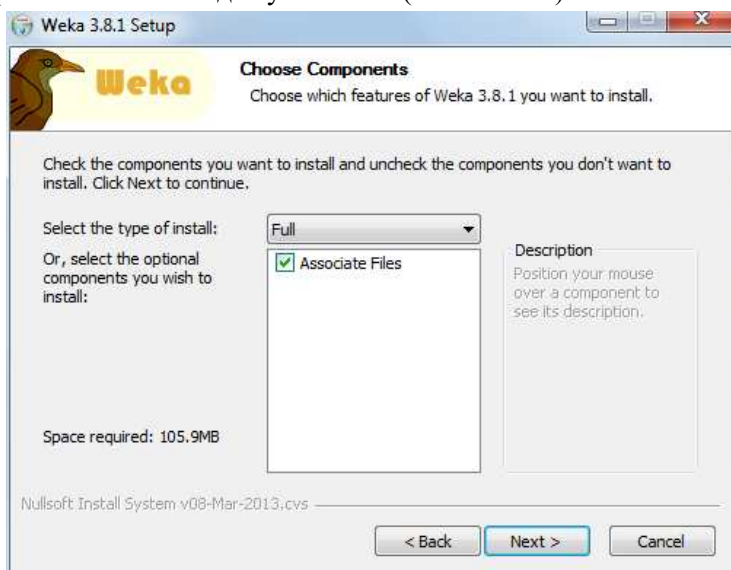


Рис. 9.4 Компоненты для установки

Выберите расположение (см. Рис. 9.5).

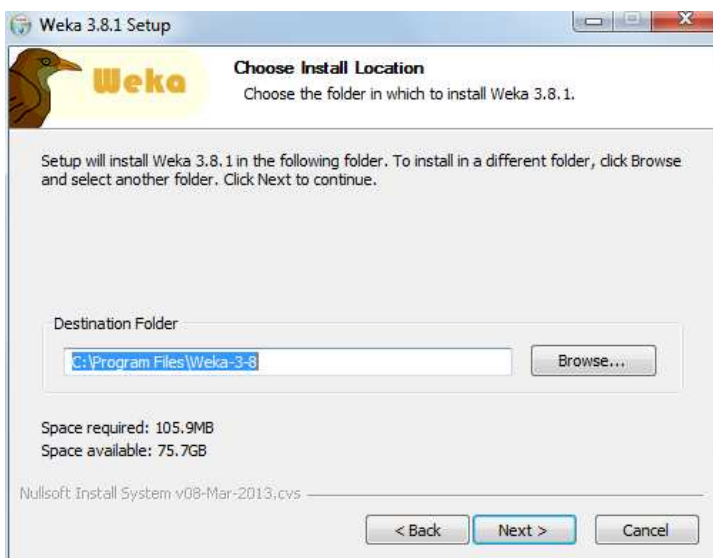


Рис. 9.5 Расположение пакета

После этого начнется установка (см. Рис. 9.6).

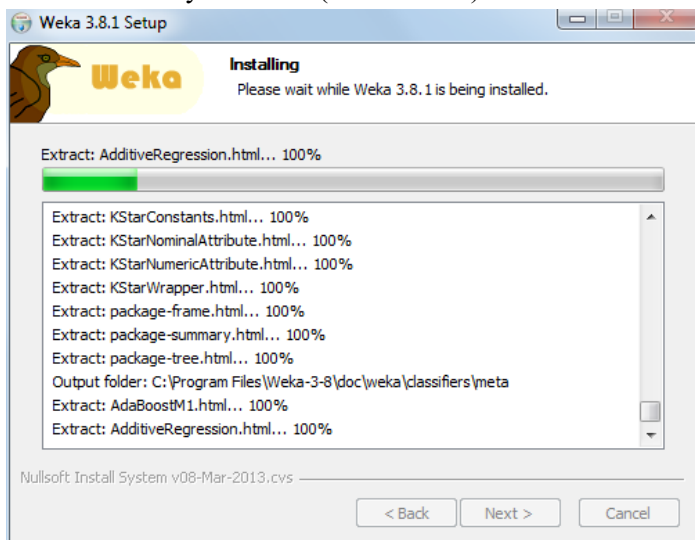


Рис. 9.6 Установка WEKA

## Формат входных данных

Программа WEKA может работать с входными данными определенного формата: ARFF (Attribute-Relation File Format), CSV (Comma-Separated Values) и JSON (JavaScript Object Notation).

Согласно официальной документации WEKA разработчики рекомендуют работать с форматом ARFF, он установлен по умолчанию и WEKA изначально создавалась именно для работы с ним – поэтому использование ARFF гарантирует стабильную работу всех алгоритмов, более точные выходные результаты и наименьшее время обработки данных. Например, если в качестве входных данных использовать файл формата CSV, то WEKA не сможет сразу обработать данные, т.к. CSV не описывает типы данных, соответственно WEKA сначала должна прочитать весь файл, определить тип данных для каждого значения и только потом приступить к обработке данных. Так же согласно отзывам пользователей, некоторые алгоритмы, заложенные в WEKA (например, алгоритмы на основе деревьев решений), неточно работают на файлах формата CSV.

ARFF файлы состоят из двух частей:

1) Заголовок – содержит общую информацию о БД (например: название БД, имя создателя ARFF-файла, дата последнего обновления и т.д.), далее идет служебное слово `@RELATION` \_имя\_БД, после чего идет список атрибутов и их типов: служебное слово `@ATTRIBUTE` \_имя\_атрибута \_тип (см. Рис. 9.7).

```
% 1. Title: Iris Plants Database
%
% 2. Sources:
%   (a) Creator: R.A. Fisher
%   (b) Donor: Michael Marshall (MARSHALL%PLU@io.arc.nasa.gov)
%   (c) Date: July, 1988
%
@RELATION iris

@ATTRIBUTE sepallength NUMERIC
@ATTRIBUTE sepalwidth NUMERIC
@ATTRIBUTE petallength NUMERIC
@ATTRIBUTE petalwidth NUMERIC
@ATTRIBUTE class {Iris-setosa,Iris-versicolor,Iris-virginica}
```

Рис. 9.7 Заголовок ARFF-файла

2) Данные – этот блок начинается со служебного слова `@DATA`, после которого через запятую перечислены сами данные в строгом соответствии с атрибутами, описанными в заголовке (см. Рис. 9.8).

```
@DATA
5.1,3.5,1.4,0.2,Iris-setosa
4.9,3.0,1.4,0.2,Iris-setosa
4.7,3.2,1.3,0.2,Iris-setosa
4.6,3.1,1.5,0.2,Iris-setosa
5.0,3.6,1.4,0.2,Iris-setosa
5.4,3.9,1.7,0.4,Iris-setosa
4.6,3.4,1.4,0.3,Iris-setosa
5.0,3.4,1.5,0.2,Iris-setosa
4.4,2.9,1.4,0.2,Iris-setosa
4.9,3.1,1.5,0.1,Iris-setosa
```

Рис. 9.8 Данные ARFF-файла

Если БД представлена в формате CSV или JSON, имеет большое число атрибутов и неудобно вручную переводить ее в формат ARFF, то можно сделать это в автоматическом режиме.

Например, возьмем файл CSV со следующим содержанием (см. Рис. 9.9):

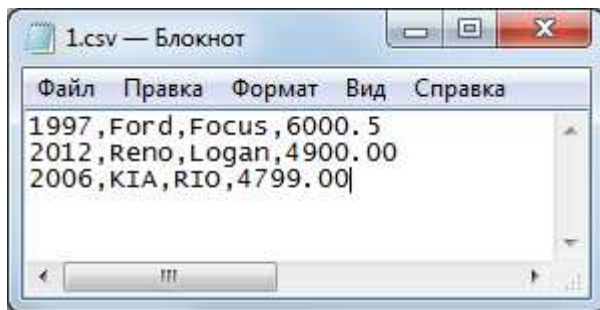


Рис. 9.9 CSV-файл

В первую строку файла нужно добавить следующий текст:  
year,mark,model,price

Далее нужно зайти в программу WEKA, открыть файл в формате .csv (Open File) и пересохранить его в формате .arff (Save). В результате этих действий будет сгенерирован новый файл со следующим содержанием (см. Рис. 9.10):



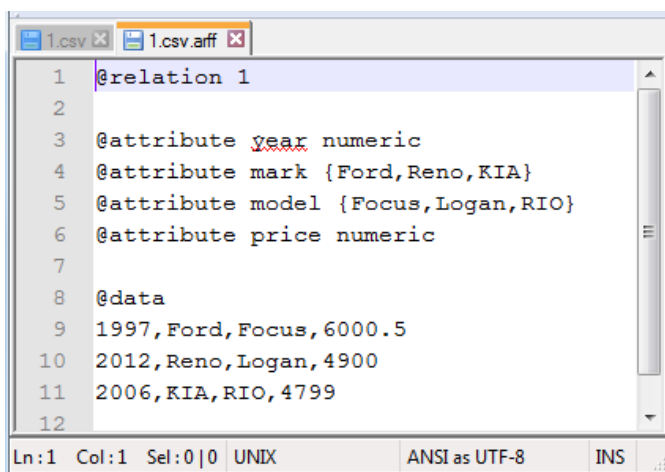


Рис. 9.10 Новый ARFF-файл

Посмотреть общую информацию о формате ARFF можно по следующему адресу:

[http://weka.wikispaces.com/ARFF+%28book+version%29?cm\\_mc\\_uid=33626507721114522502964&cm\\_mc\\_sid\\_50200000=1454610427](http://weka.wikispaces.com/ARFF+%28book+version%29?cm_mc_uid=33626507721114522502964&cm_mc_sid_50200000=1454610427).

### UC Irvine Machine Learning Repository

Так как WEKA работает с большими объемами данных, а целью данной лабораторной работы является знакомство с основными возможностями программы, БД для обучения следует брать из открытых источников, например из UC Irvine Machine Learning Repository - <http://archive.ics.uci.edu/ml/datasets.html>.

Рассмотрим пример работы с этим репозиторием: сначала нужно выбрать БД для работы, например, БД с данными о цветках ириса <http://archive.ics.uci.edu/ml/datasets/Iris> (см. Рис. 9.11).

## Iris Data Set

Download [Data Folder](#) [Data Set Description](#)

Abstract: Famous database; from Fisher, 1936



Data Set Characteristics:	Multivariate	Number of Instances:	150	Area:	Life
Attribute Characteristics:	Real	Number of Attributes:	4	Date Donated	1988-07-01
Associated Tasks:	Classification	Missing Values?	No	Number of Web Hits:	1431404

### Source:

Creator:

R.A. Fisher

Donor:

Michael Marshall (MARSHALL%PLU%@\*io.arc.nasa.gov)

Рис. 9.11 БД Iris

После нажатия на Data Folder можно увидеть файлы, доступные для скачивания (см. Рис. 9.12): файлы с данными имеют расширение «.data» - bezdekIris.data и iris.data; файл с подробным описанием БД имеет расширение «.names» – iris.names.

## Index of /ml/machine-learning-databases/iris

Name	Last modified	Size	Description
<a href="#">Parent Directory</a>	-	-	-
<a href="#">Index</a>	03-Dec-1996 04:01	105	
<a href="#">bezdekIris.data</a>	14-Dec-1999 12:12	4.4K	
<a href="#">iris.data</a>	08-Mar-1993 16:27	4.4K	
<a href="#">iris.names</a>	11-Jul-2000 21:30	2.9K	

Apache/2.2.15 (CentOS) Server at archive.ics.uci.edu Port 80

Рис. 9.12 Файлы для скачивания

Для формирования .arff файла выполните следующие действия:

- 1) Создайте пустой .arff – файл.
- 2) Откройте файл с описанием БД (iris.names), перейдите к седьмому пункту - Attribute Information (см. Рис. 9.13), скопируйте имена атрибутов в файл, созданный в пункте 1.

```

4. Relevant Information:
--- This is perhaps the best known database to be found in the pattern
    recognition literature. Fisher's paper is a classic in the field
    and is referenced frequently to this day. (See Duda & Hart, for
    example.) The data set contains 3 classes of 50 instances each,
    where each class refers to a type of iris plant. One class is
    linearly separable from the other 2; the latter are NOT linearly
    separable from each other.
--- Predicted attribute: class of iris plant.
--- This is an exceedingly simple domain.
--- This data differs from the data presented in Fishers article
    (identified by Steve Chadwick,  spchadwick@espeedaz.net )
    The 35th sample should be: 4.9,3.1,1.5,0.2,"Iris-setosa"
    where the error is in the fourth feature.
    The 38th sample: 4.9,3.6,1.4,0.1,"Iris-setosa"
    where the errors are in the second and third features.

5. Number of Instances: 150 (50 in each of three classes)

6. Number of Attributes: 4 numeric, predictive attributes and the class

7. Attribute Information:
1. sepal length in cm
2. sepal width in cm
3. petal length in cm
4. petal width in cm
5. class:
   -- Iris Setosa
   -- Iris Versicolour
   -- Iris Virginica

8. Missing Attribute Values: None

```

Рис. 9.13 Файл с описанием БД

3) Откройте файл с данными (iris.data) и скопируйте их в файл, созданный в пункте 1.

4) Приведите данные в файле к формату ARFF: создайте заголовок с ключевыми словами @RELATION, @ATTRIBUTE (см. Рис. 9.14).

```

@RELATION iris

@ATTRIBUTE sepallength NUMERIC
@ATTRIBUTE sepalwidth NUMERIC
@ATTRIBUTE petallength NUMERIC
@ATTRIBUTE petalwidth NUMERIC
@ATTRIBUTE class {Iris-setosa,Iris-versicolor,Iris-virginica}

```

Рис. 9.14 Заголовок ARFF-файла

5) Приведите данные в файле к формату ARFF: создайте блок с данными с ключевым словом @DATA (см. Рис. 9.15).

```
@DATA
5.1,3.5,1.4,0.2,Iris-setosa
4.9,3.0,1.4,0.2,Iris-setosa
4.7,3.2,1.3,0.2,Iris-setosa
4.6,3.1,1.5,0.2,Iris-setosa
5.0,3.6,1.4,0.2,Iris-setosa
5.4,3.9,1.7,0.4,Iris-setosa
4.6,3.4,1.4,0.3,Iris-setosa
5.0,3.4,1.5,0.2,Iris-setosa
4.4,2.9,1.4,0.2,Iris-setosa
4.9,3.1,1.5,0.1,Iris-setosa
```

Рис. 9.15 Данные ARFF-файла

Также некоторые файлы с данными в формате ARFF доступны на официальном сайте WEKA –

<http://www.cs.waikato.ac.nz/ml/weka/datasets.html> или на вашем диске после установки WEKA, например, по адресу C:\Program Files\Weka-3-8\data.

### Предварительная обработка данных в WEKA

При запуске WEKA предлагает пользователю выбрать тип графического интерфейса программы: Explorer, Experimenter, KnowledgeFlow, Workbench и Simple CLI (см. Рис. 9.16). В данной лабораторной работе используется Explorer.



Рис. 9.16 Окно выбора GUI

После выбора GUI откроется основное окно WEKA – закладка Preprocess (см. Рис. 9.17). Для загрузки данных нужно нажать Open File и выбрать файл с БД, например, noise.arff.

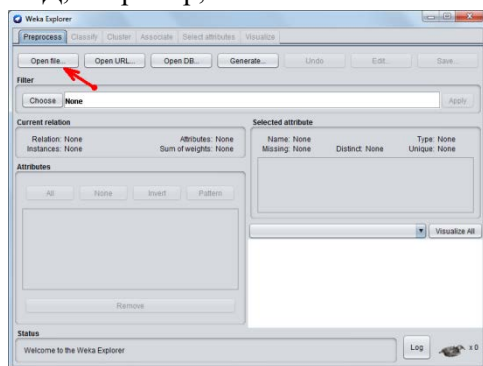


Рис. 9.17 Закладка Preprocess

Вид главного окна после загрузки данных представлен на Рисунке 9.18. В блоке под номером один находится общая информация о БД: имя, количество атрибутов и число строк (объектов) в БД. В блоке под номером два находится список атрибутов БД. В блоке под номером три находится информация о наборе данных в конкретном столбце: максимальное, минимальное и среднее значения в столбце и стандартное отклонение (статистический показатель рассеивания значений случайной величины). Во втором блоке можно кликнуть на интересующий атрибут и увидеть информацию по нему в третьем блоке.

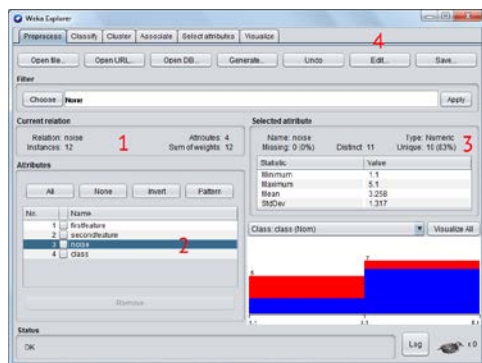


Рис. 9.18 Вкладка Preprocess после загрузки данных

Для редактирования данных напрямую из WEKA, нужно нажать на кнопку Edit (цифра 4, Рис.9.18), после чего откроется окно, представленное на Рисунке 9.19. Кроме редактирования в этом окне можно добавить данные - если нажать на кнопку Add instance. Все изменения никак не отразятся на исходном файле, который был загружен в WEKA; для сохранения изменений нужно нажать кнопку SAVE и создать новый файл.

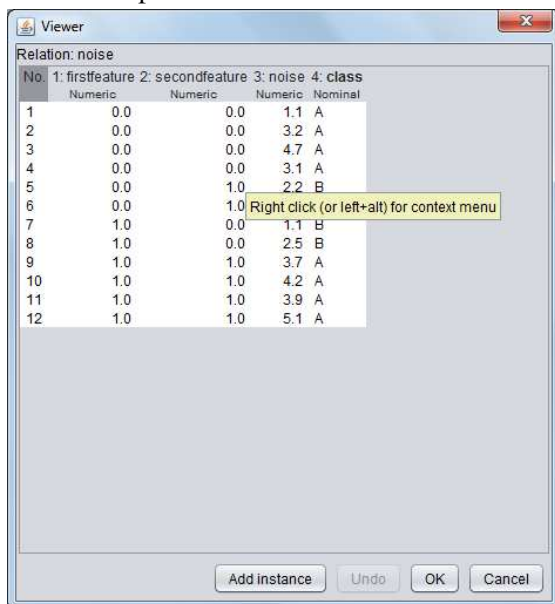


Рис. 9.19 Окно для редактирования и добавления данных

На верхней панели Filter можно выбрать преобразование данных. Например, нажатием кнопки Choose из иерархического списка выберем опцию Weka/filters/unsupervised/attribute/Normalize (см. Рис. 9.20). На панели отобразится название фильтра, щёлкнув по нему правой клавишей мыши, можно изменить его параметры. Нажатие кнопки Apply запускает фильтр – в данном случае происходит нормализация (см. Рис. 9.21). Для отмены изменений, внесенных фильтром, нужно нажать Undo.

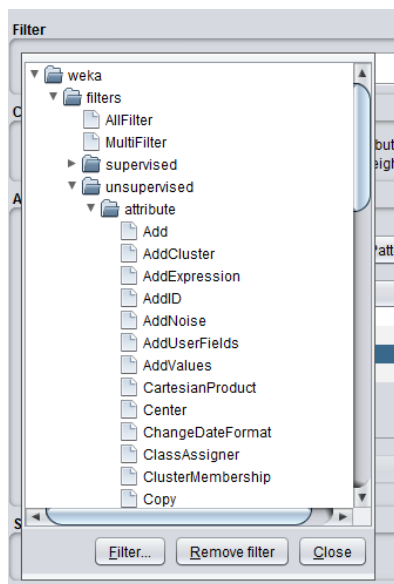


Рис. 9.20 Выбор алгоритма для предобработки исходных данных

No.	firstfeature	secondfeature	noise
1	0.0	0.0	0.0 A
2	0.0	0.0	0.52... A
3	0.0	0.0	0.90... A
4	0.0	0.0	0.50... A
5	0.0	1.0	0.27... B
6	0.0	1.0	0.8 B
7	1.0	0.0	0.0 B
8	1.0	0.0	0.35... B
9	1.0	1.0	0.65... A
10	1.0	1.0	0.77... A
11	1.0	1.0	0.70... A
12	1.0	1.0	1.0 A

Рис. 9.21 Нормализованные данные

Для того чтобы WEKA не учитывала значения некоторого столбца (например, firstfeature) в результирующей модели, их можно удалить: для этого нужно перейти на вкладку Preprocess, в блоке Attributes выбрать столбец для удаления и нажать кнопку Remove (см. Рис. 9.22).

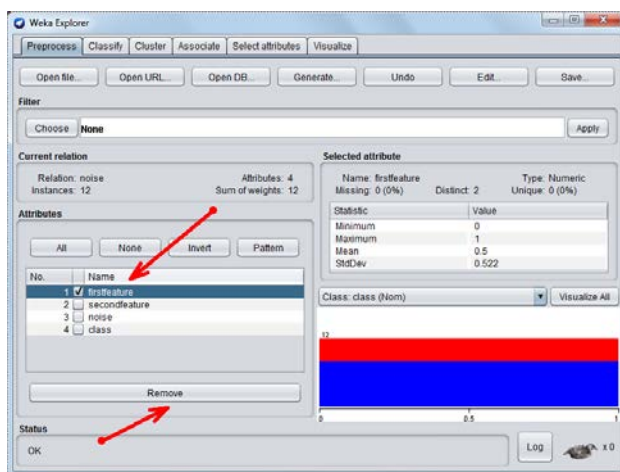


Рис. 9.22 Удаление столбца данных

## Регрессионный анализ

Модель регрессионного анализа используется для прогнозирования значения одной зависимой переменной, исходя из известных значений нескольких независимых параметров.

Рассмотрим следующую задачу: у некоторого риэлтерского агентства есть большая БД с данными о домах; на основании этих данных нужно определить приблизительную стоимость некоего абстрактного дома. Зависимой переменной является цена на дом, которая определяется исходя из нескольких независимых переменных, например: площади дома, наличия/отсутствия участка, площади участка, количества спален, дороговизны отделки, состояния бытовой техники и т.д. Формат БД агентства представлен на Рисунке 9.23.

Площадь дома (кв.футы)	Размер участка	Количество спален	Гранитная отделка на кухне	Современное сантехническое оборудование?	Продажная цена
3529	9191	6	0	0	\$205,000
3247	10061	5	1	1	\$224,900
4032	10150	5	0	1	\$197,900
2397	14156	4	1	0	\$189,900
2200	9600	4	0	1*	\$195,000
3536	19994	6	1	1	\$325,000
2983	9365	5	0	1	\$230,000
3198	9669	5	1	1	????

Рис. 9.23 Формат БД агентства



Вначале нужно привести БД к формату ARFF – это было сделано заранее, данные находятся в файле task\_1\_houses.arff, который нужно загрузить в WEKA.

Для создания регрессионной модели, нужно выполнить следующие действия:

1. Открыть закладку Classify (см. Рис. 9.24).

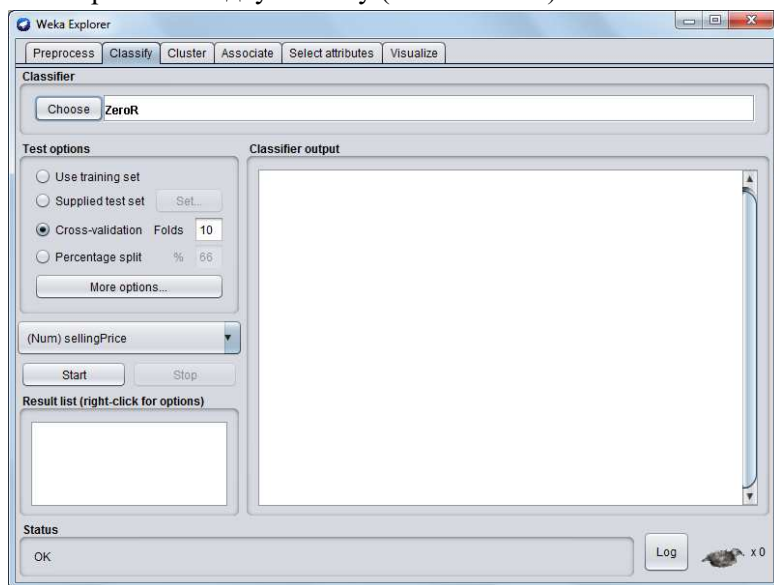


Рис. 9.24 Закладка Classify

2. Выбрать тип модели для анализа: кликнуть по кнопке Choose и в меню functions выбрать тип LinearRegression (см. Рис. 9.25). Следует отметить, что в меню есть два типа модели регрессионного анализа: LinearRegression и SimpleLinearRegression. В данном случае был выбран тип LinearRegression, а не SimpleLinearRegression, потому что SimpleLinearRegression определяет значение зависимой переменной по значениям только одного независимого параметра (в данном примере их шесть).

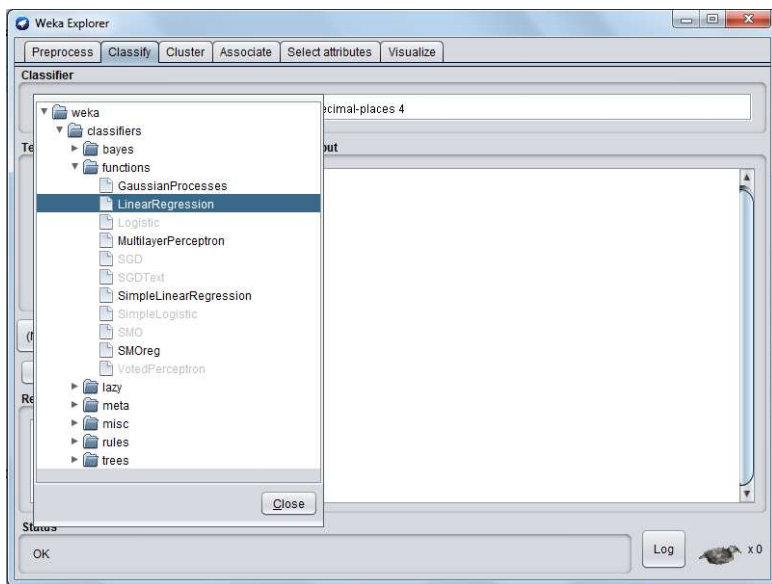


Рис.9.25 Выбор типа модели для анализа

3. Выбрать данные для создания модели: для проведения регрессионного анализа нужна опция Use training set – тогда модель будет создана на основе БД из загруженного ранее ARFF-файла (см. Рис. 9.26).

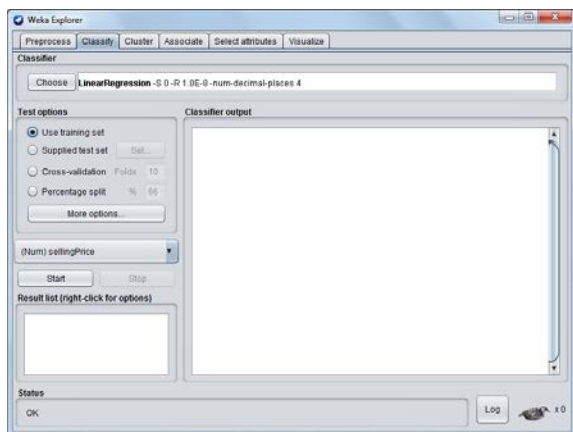


Рис. 9.26 Выбор данных для создания модели

4. Выбрать зависимую переменную – имя столбца, в котором находится неизвестное значение, расчет которого является основной

задачей. В данном случае это цена дома (sellingPrice). После блока Test options находится раскрывающийся список, в котором нужно выбрать зависимый параметр.

5. Нажать кнопку Start. Результаты работы представлены на Рисунке 9.27.

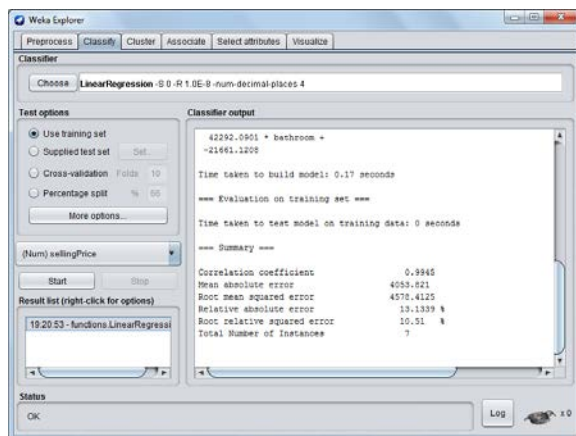


Рис. 9.27 Результаты работы WEKA

На левой нижней панели Result List выводится перечень всех запусков классификаций данных. Новый элемент в этом списке появляется после завершения настройки и тестирования классификатора (вызванных нажатием кнопки Start). В списке указывается время окончания классификации и название классификатора. Если по элементу списка щёлкнуть правой кнопкой мыши, то появится дополнительное меню, с помощью которого можно посмотреть отчёт в отдельном окне (View in separate window), загрузить и сохранить модель, т.е. тип и параметры классификатора (Load model, Save model), визуализировать ошибки классификатора (Visualize classifier errors, см. также работу с вкладкой Visualize), посмотреть различные кривые отступов и ошибок. Для деревьев классификации доступна опция Tree View.

Кликнем на область Classifier output и промотаем её вверх. В блоке Run information находится общая информация о БД (см. Рис. 9.28).

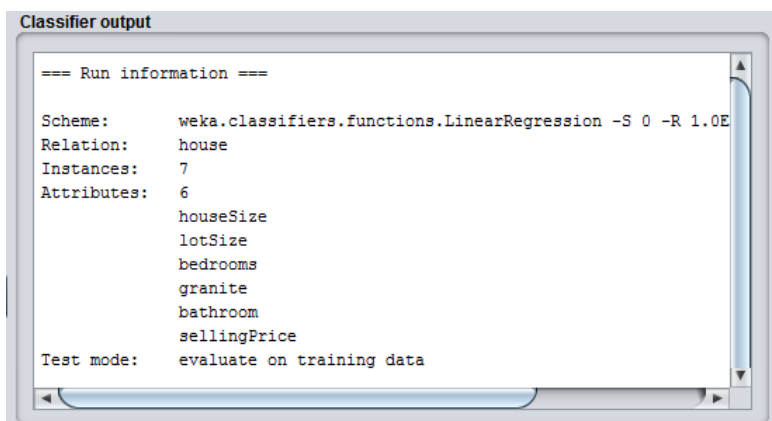


Рис. 9.28 Общая информация о БД

В блоке Classifier model указывается тип модели для анализа ([Linear Regression Model](#)) и сама итоговая модель (sellingPrice) (см. Рис. 9.29).

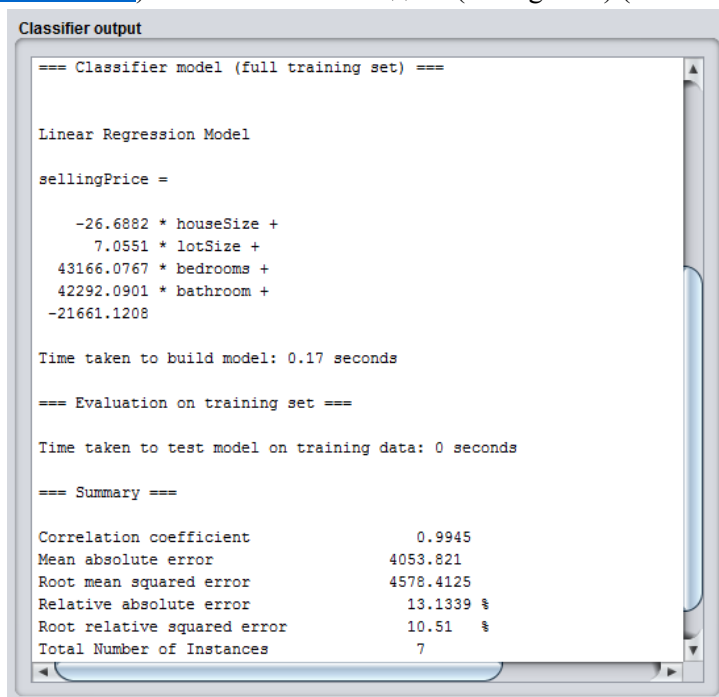


Рис. 9.29 Итоговая модель и ее тип

Для того чтобы рассчитать стоимость некоего нового дома, нужно подставить его характеристики в полученную модель:

```

sellingPrice = (-26.6882 * 3198) +
               (7.0551 * 9669) +
               (43166.0767 * 5) +
               (42292.0901 * 1)
               - 21661.1208

sellingPrice = 219,328

```

Рис. 9.30 Формула для расчета стоимости дома

Однако следует помнить, что основная задача ДМ состоит не в расчете какого-то конкретного значения, а в обнаружении скрытых зависимостей в больших наборах данных. Поэтому помимо расчетов стоимости дома необходимо рассмотреть зависимости между данными модели и выделить правила формирования цен на недвижимость.

Проанализировав модель можно сделать следующие выводы:

1) Так как в результирующей модели отсутствует параметр *granite*, следовательно, гранитные элементы в оформлении никак не влияют на цену дома.

2) Так как параметр *bathroom* в БД является бинарным и присутствует в результирующей модели, можно сделать вывод, что современное сантехническое оборудование сильно влияет на цену дома, а именно добавляет 42292 у.е. к его цене.

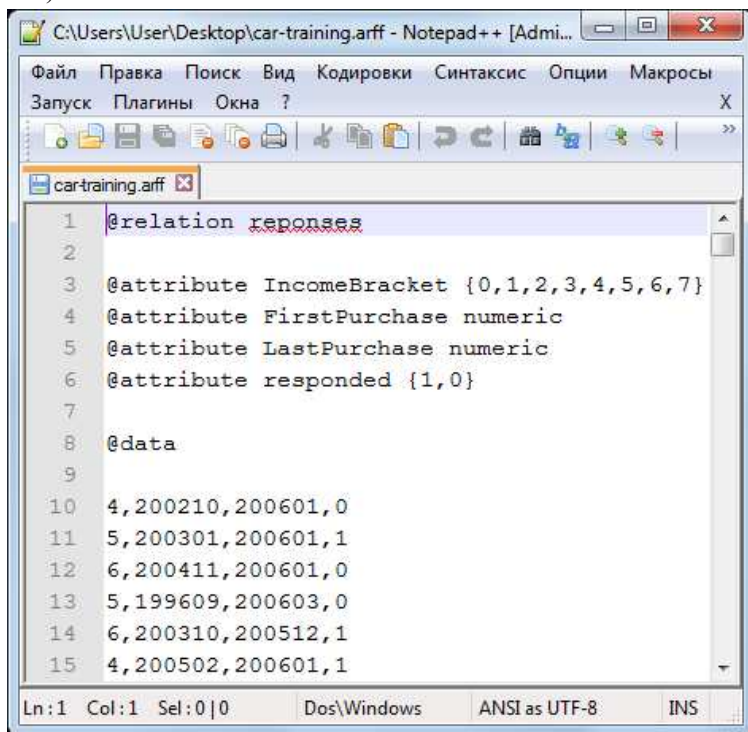
3) Коэффициент в модели перед параметром *houseSize* отрицательный, значит, большая площадь дома снижает его цену. Однако подобное утверждение кажется нелогичным – такую неточность можно объяснить тем, что на самом деле, размер дома не является независимой величиной. Этот параметр связан, например, с количеством спален (в больших домах и количество спален больше). Таким образом, приходим к выводу, что полученная модель неточна, но это можно поправить – удалить столбец *houseSize* и пересоздать модель заново.

### Классификационный анализ на основе деревьев

Рассмотрим задачу классификации: есть информация о продажах автомобилей от некого дилерского центра (распределение по доходам; год/месяц покупки первого и последнего автомобилей в центре; использование расширенной гарантии), нужно классифицировать

клиентов центра на несколько групп, чтобы для каждой группы создать персональное предложение по расширенной гарантии. Цель дилерского центра – повысить уровень продаж, опираясь на результаты интеллектуального анализа полученных данных.

Данных в формате ARFF представлены следующим образом (см. Рис. 9.31):



```
1 @relation reponses
2
3 @attribute IncomeBracket {0,1,2,3,4,5,6,7}
4 @attribute FirstPurchase numeric
5 @attribute LastPurchase numeric
6 @attribute responded {1,0}
7
8 @data
9
10 4,200210,200601,0
11 5,200301,200601,1
12 6,200411,200601,0
13 5,199609,200603,0
14 6,200310,200512,1
15 4,200502,200601,1
```

Рис. 9.31. Данных в формате ARFF

Изначально весь набор данных был разделен так, чтобы часть данных использовалась для создания модели, а часть для проверки ее точности (чтобы убедиться, что модель не является подогнанной под конкретный набор данных).

Для создания модели нужно выполнить следующие действия:

1. Загрузить файл car-training.arff в WEKA (см. Рис. 9.32).

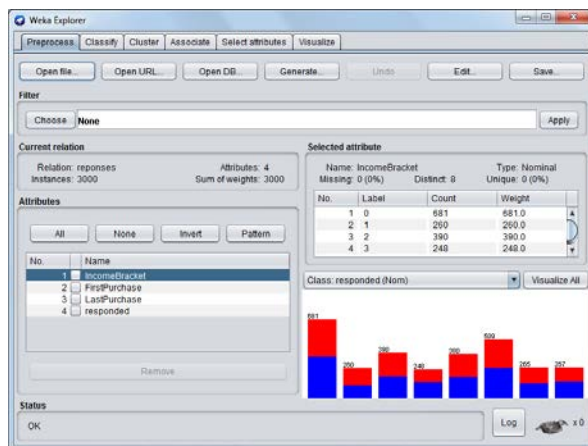


Рис. 9.32 Закладка Preprocess

2. Выбрать модель для классификации: открыть закладку Classify, выбрать опцию trees, а потом опцию J48 (J48 - реализация алгоритма C4.5 на языке Java; C4.5 - алгоритм для построения деревьев решений, разработанный Джоном Квинланом) (см. Рис. 9.33).

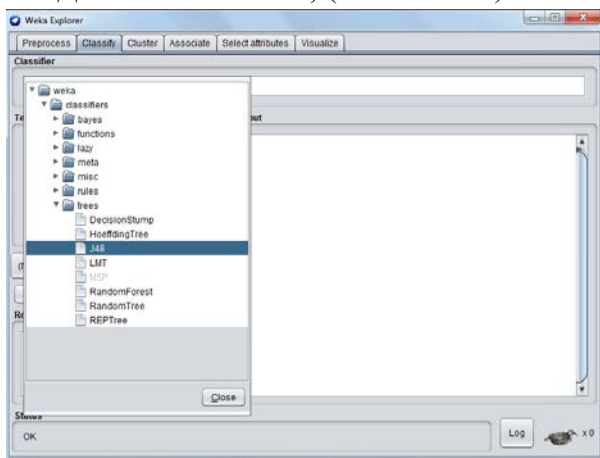


Рис. 9.33 Выбор модели для классификации

3. Включить опцию Use training set, чтобы пакет WEKA при создании модели использовал данные, которые мы только что загрузили в виде файла.

4. Создать модель: нажать кнопку Start. Результаты работы представлены на Рисунке 5.34. Наиболее существенные данные – это

показатели классификации «Correctly Classified Instances» (59.1%) - показатель точности модели и «Incorrectly Classified Instances» (40.9%). Таблица Confusion Matrix показывает количество ложноположительных (516) и ложноотрицательных (710) распознаваний.

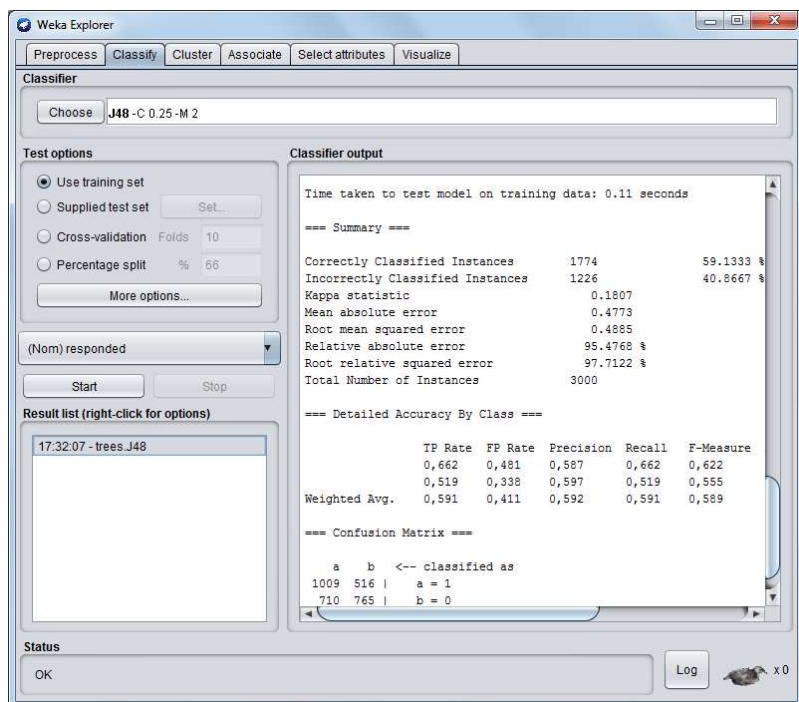


Рис. 9.34 Результаты работы WEKA

5. Чтобы увидеть [дерево классификации](#), нужно щелкнуть правой кнопкой мышки в панели Result list и в контекстном меню выбрать опцию Visualize tree (см. Рис.9.35). На экране отобразится визуальное представление классификационного дерева модели (см. Рис. 9.36). Еще один способ увидеть дерево модели – прокрутить вверх вывод в окне Classifier Output, там можно найти текстовое описание дерева с узлами и листьями (см. Рис. 9.37).



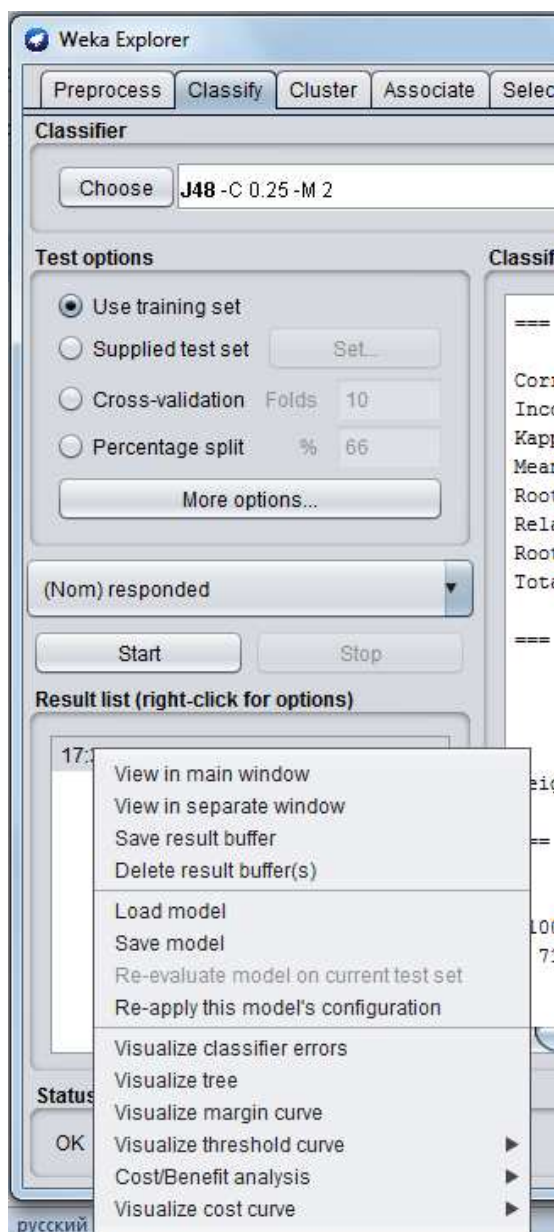


Рис. 9.35 Просмотр дерева классификации

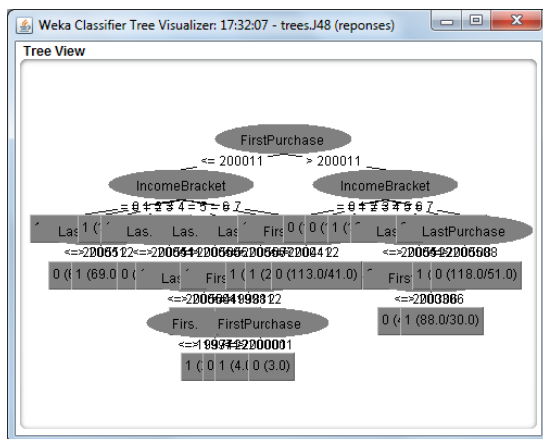


Рис. 9.36 Визуальное представление классификационного дерева модели

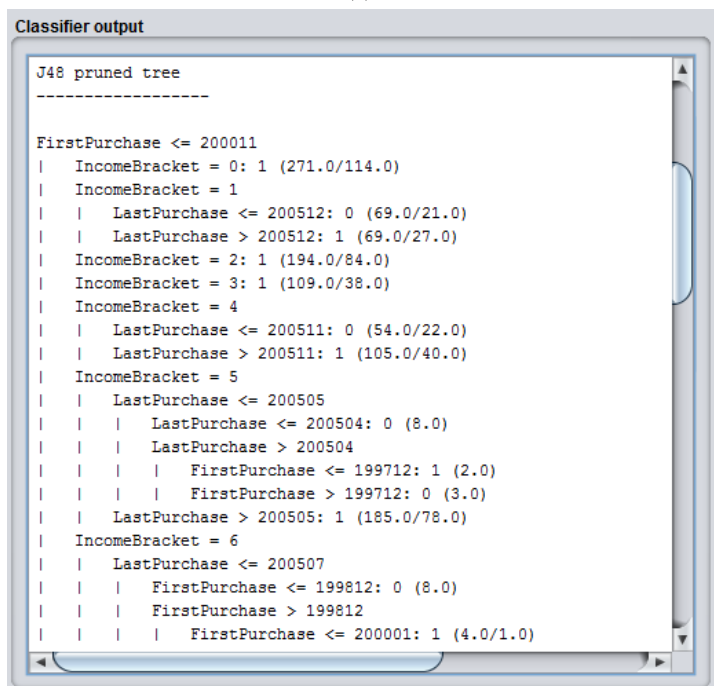


Рис. 9.37 Текстовое описание дерева с узлами и листьями

6. Остался последний этап проверки классификационного дерева: нужно пропустить оставшийся набор данных через полученную модель и проверить, насколько результаты классификации будут отличаться от

реальных данных. Для этого в секции Test options нужно выбрать опцию Supplied test set и нажать на кнопку Set, потом выбрать файл car\_test.arff (вторая часть БД), и нажать на кнопку Start. Результаты работы представлены на Рисунке 9.38.

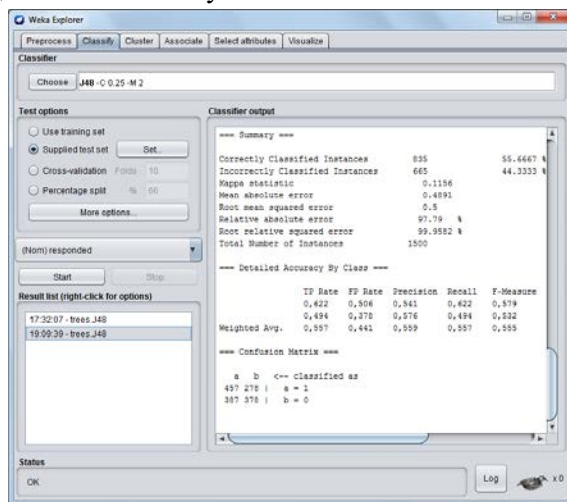


Рис. 9.38 Результаты работы

Сравнивая показатель Correctly Classified Instances для тестового набора (55,7%) с этим же показателем для обучающего набора (59,1%), можно сделать вывод, что точность модели для двух разных наборов данных примерно одинакова. Это значит, что новые данные, которые будут использоваться в этой модели в будущем, не снизят точность ее работы.

Однако, т.к. точность модели довольно низка, можно сделать вывод, что модель не подходит для практических исследований, т.к. работает с точностью чуть выше 50% (с таким же успехом можно просто пытаться угадать значение случайным образом). Это приводит к пониманию весьма важного факта: существуют случаи, когда использование алгоритмов интеллектуального анализа данных приводит к созданию неудачной аналитической модели. Полученная классификационная модель не подходит для анализа этих данных, т.к. не дает никаких полезных сведений, а ее использование может привести к принятию неверных решений и потенциальной потере денег.

Означает ли это, что исходные данные вообще не подлежат никакому анализу? Ответ демонстрирует еще одну важную особенность интеллектуального анализа данных: используя метод «ближайших соседей», можно создать другую модель на базе этого же набора данных, с точностью работы 88%. Итак, всегда необходимо помнить, что для того, чтобы извлечь полезную информацию из большого набора данных, нужно сначала выбрать подходящую модель.

### Классификационный анализ на основе метода Ближайших соседей

Для создания модели нужно выполнить следующие действия:

1. Загрузить файл car-training.arff в WEKA.
2. Открыть закладку Classify. В панели Chose выбрать опцию lazy, а затем IBk (Instance-Based – обучение на примерах, k – количество соседей). Количество ближайших соседей в модели можно изменять, для этого нужно щелкнуть правой кнопкой мышки на поле «IBk-K 1...» и выбрать опцию Show Properties (см. Рис. 9.39). Обычно, точность модели повышается по мере добавления соседей.

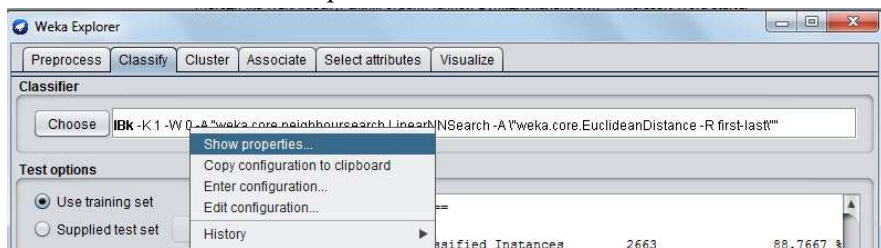


Рис. 9.39 Вызов подменю Properties

3. Щелкнуть опцию Use training set.
4. Нажать кнопку Start. Результаты работы представлены на Рисунке 9.40.

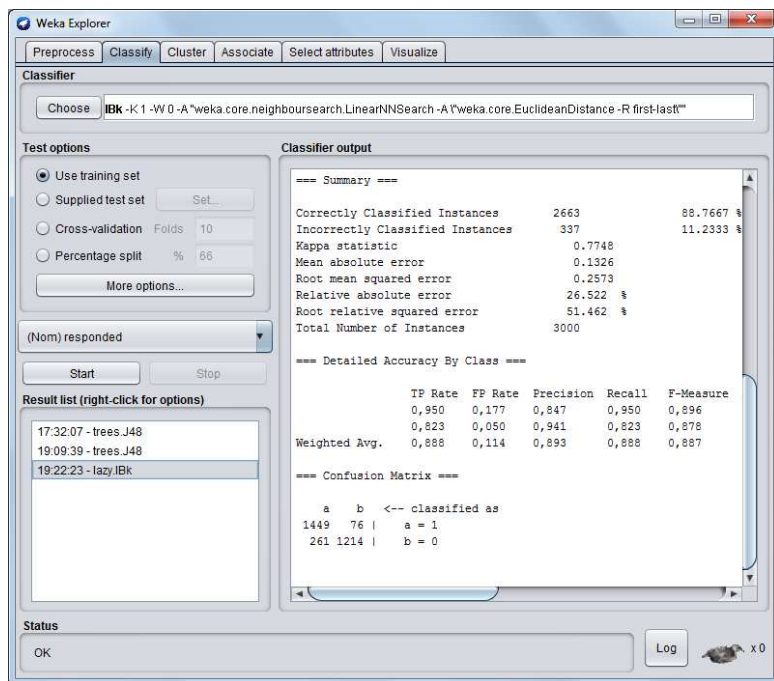


Рис. 9.40 Результаты работы WEKA

У модели, использующей метод ближайших соседей, показатель точности довольно высокий и равен 88.8%.

Результаты использования модели показывают, что есть 76 ложноположительных распознаваний (2.5%) и 261 ложноотрицательных распознаваний (8.7%). Ложноположительное распознавание означает, что модель считает, что данный покупатель приобретет расширенную гарантию, хотя на самом деле он отказался от покупки. Ложноотрицательное распознавание означает, что данный покупатель откажется от расширенной гарантии, а на самом деле он ее купил. Предположим, что стоимость каждой рекламной листовки, рассылаемой дилером, составляет 3 у.е., а покупка одной расширенной гарантии приносит ему 400 у.е. дохода. Таким образом, ошибки ложного распознавание в терминах расходов и доходов нашего дилера будут выглядеть следующим образом:  $400 - (2.5\% * 3) - (8.7\% * 400) = 365$  у.е. Следовательно, ложное распознавание ошибается в пользу дилера. Сравним этот показатель с данными модели классификации на

основе деревьев:  $400 - (17.2\% * 3) - (23.7\% * 400) = 304$  у.е. Как вы видите, использование более точной модели повышает потенциальный доход дилера на 20%.

Однако следует учитывать, что модель классификации на основе метода ближайших соседей становится практически бесполезной на небольших наборах данных.

### Классификационный анализ на основе многослойного персептрона

Для создания модели нужно выполнить следующие действия:

1. Загрузить файл noise.arff в WEKA.
2. Открыть закладку Cluster. В панели Chose выбрать опцию weka/classifiers/functions/MultilayerPerceptron (см. Рис. 9.41).
3. Настроить параметры алгоритма кластеризации: щелкнуть левой кнопкой на опции MultilayerPerceptron; если выбрать значение параметра GUI равным true, то появится возможность редактировать нейронную сеть (см. Рис. 9.42).

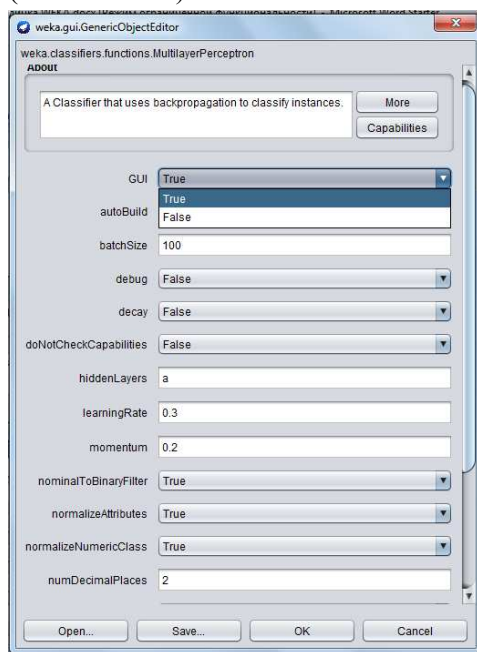


Рис. 9.41. Параметры алгоритма

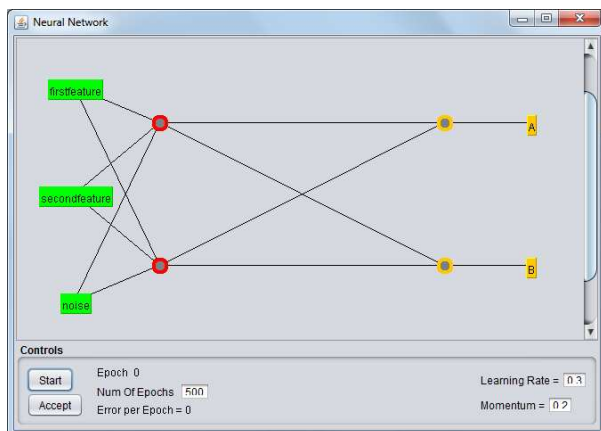


Рис. 9.42 Сеть на основе многослойного персептрона

4. В поле Test Options выбрать «Use training set» (см. Рис.9.43).
5. Для того чтобы увидеть результаты классификации нужно нажать More Options и выбрать Output predictions. Обратите внимание, что отображается также полезная информация о вероятности принадлежности классу. Результат работы алгоритма представлен на Рисунке 9.43.

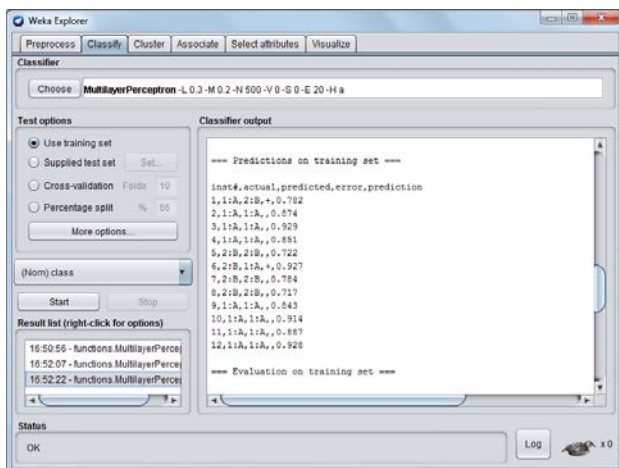


Рис. 9.43 Результат работы

### Отбор решающих признаков для задач классификации

Вкладка Select Attributes позволяет выбрать признаки для последующей классификации объектов (см. Рис. 9.44). Отбор

признаков включает перебор всех возможных комбинаций признаков данных для поиска подмножества признаков, дающих наилучший результат предсказания.

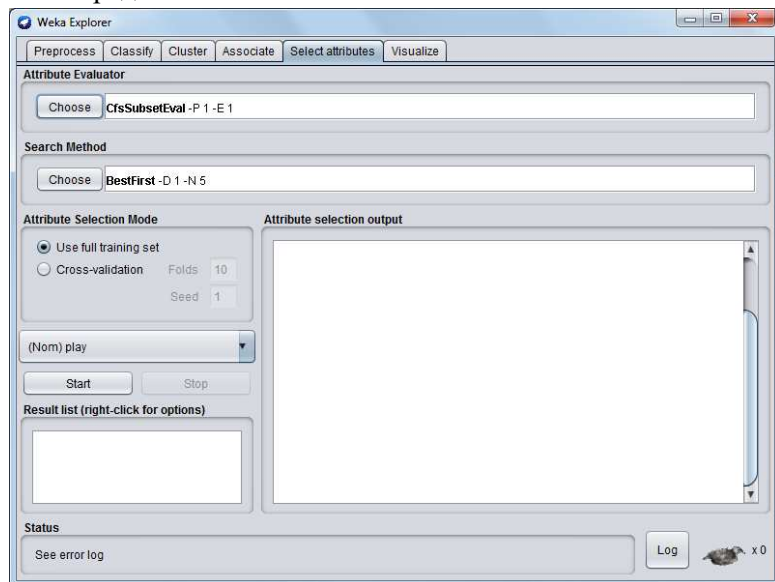


Рис. 9.44 Вкладка Select Attributes

Для отбора решающих признаков должны быть настроены два параметра: оценка атрибутов (Attribute Evaluator) и метод поиска (Search method). На панели Attribute Evaluator выбирается функция оценки качества признаков, а на панели Search Method – метод поиска оптимального признакового пространства (например, генетический алгоритм, случайный поиск или полный перебор).

Область «Attribute Selection Mode» имеет два параметра:

- Use full training set - значимость подмножества атрибутов определяется для полного набора обучающих данных.
- Cross-validation (Скользящий контроль, кросс-проверка) - значимость подмножества атрибутов определяется при помощи кросс-валидации. Поля Fold и Seed определяют количество блоков (folds) и случайный сид (seed), используемый при перетасовке данных. Внизу находится выпадающий список, задающий целевой признак, который будет использоваться в качестве класса.



По нажатию кнопки «Start» запускается процесс выбора атрибутов. Когда процесс закончен, результаты выводятся в область результатов «Attribute selection output» и добавляются в список результатов. Нажатие правой кнопки мыши на результаты выдает контекстное меню.

Для отбора решающих признаков нужно выполнить следующие действия:

1. Загрузить БД в WEKA, например, weather.numeric.arff.
2. Перейти на вкладку Select Attributes.
3. Выбрать алгоритмы для Attribute Evaluator и Search Method.
4. Выбрать Attribute Selection Mode.
5. Выбрать параметр, по которому будет происходить классификация, например – play.
6. Нажать кнопку Start. Результат работы представлен на Рисунке 9.45.

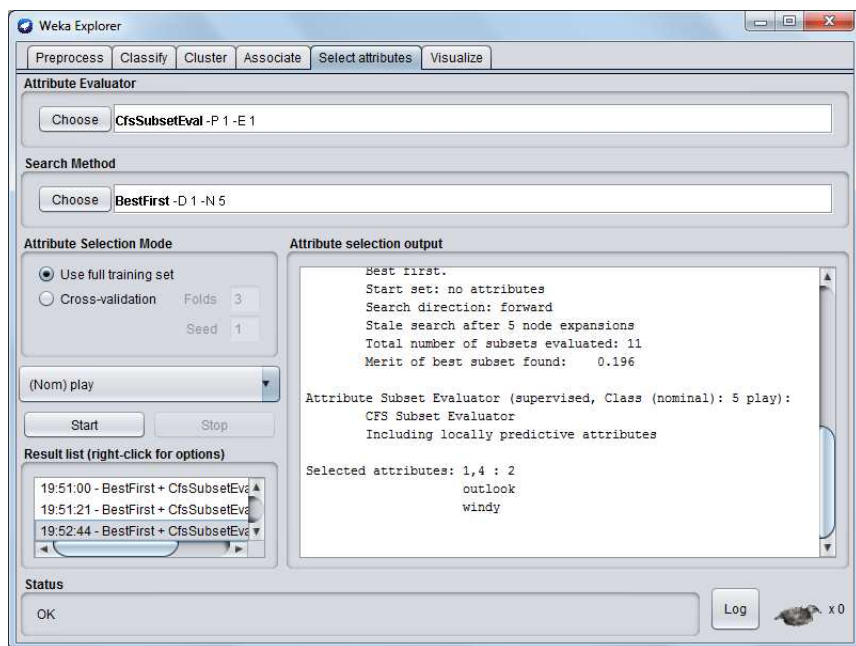
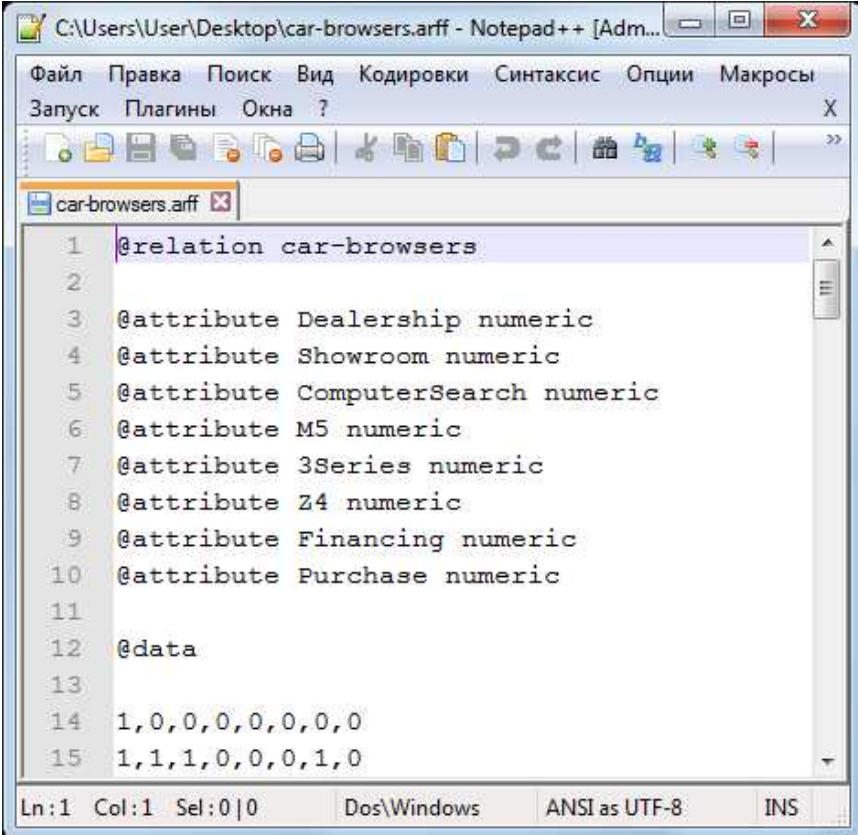


Рис. 9.45 Отбор решающих признаков

## Кластеризационный анализ

Рассмотрим задачу кластеризации: есть информация о посетителях демонстрационного зала некоего дилерского центра, нужно выделить различные группы посетителей и определить какие-либо тенденции в их поведении.

В примере используется 100 записей, и каждый столбец описывает определенный этап, который покупатель проходит в процессе выбора и приобретения автомобиля (1 - прошел, 0 – не прошел). Данные в формате ARFF представлены следующим образом (см. Рис. 9.46):



```
1 @relation car-browsers
2
3 @attribute Dealership numeric
4 @attribute Showroom numeric
5 @attribute ComputerSearch numeric
6 @attribute M5 numeric
7 @attribute 3Series numeric
8 @attribute Z4 numeric
9 @attribute Financing numeric
10 @attribute Purchase numeric
11
12 @data
13
14 1,0,0,0,0,0,0,0
15 1,1,1,0,0,0,1,0
```

Рис. 9.46 Данные в формате ARFF

Для создания модели нужно выполнить следующие действия:

1. Загрузить файл car-browsers.arff в WEKA.
2. Открыть закладку Cluster. В панели Chose выбрать опцию SimpleKMeans.
3. Настроить параметры алгоритма кластеризации: щелкнуть левой кнопкой на опции SimpleKMeans и указать число кластеров – numClusters (см. Рис. 9.47). Изменить значение по умолчанию - 2 на 5.

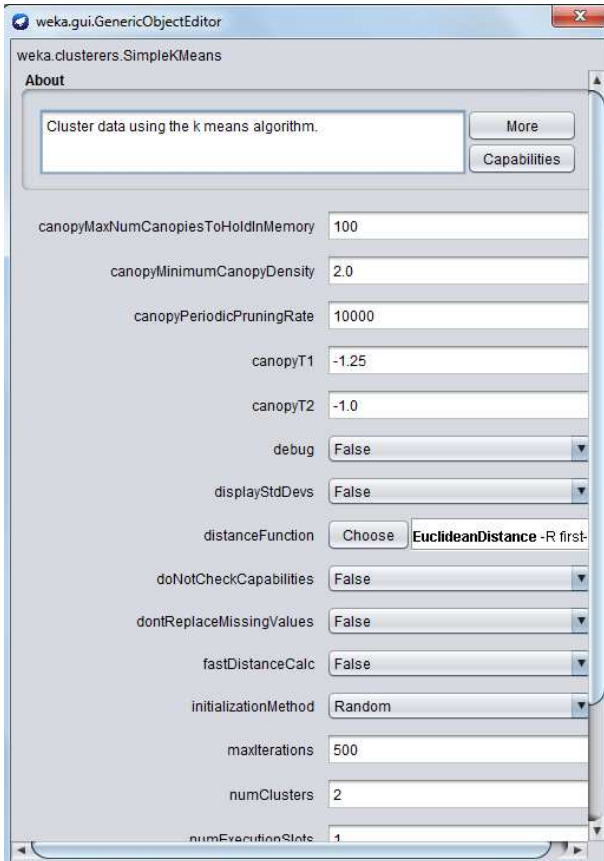


Рис. 9.47 Настройка параметров алгоритма кластеризации

4. Нажать кнопку Start. Результаты работы модели представлены на Рисунке 9.48.

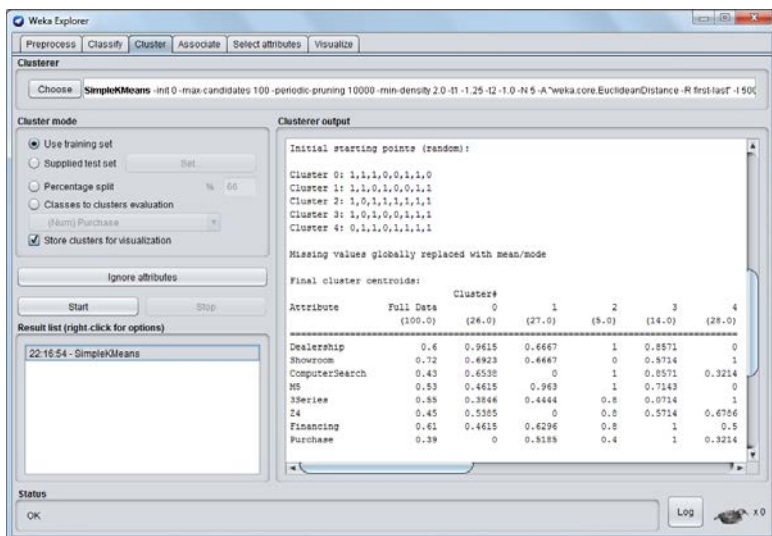


Рис. 9.48 Результаты работы WEKA

Данные кластеризации показывают, каким образом сформирован каждый кластер: значение «1» означает, что у всех данных в этом кластере соответствующий атрибут равен 1, а значение «0» - 0. Каждый кластер характеризует определенный тип поведения клиентов, поэтому на основании разбиения можно сделать следующие выводы:

Кластер 0 - эта группа посетителей рассматривает машины, но никогда ничего не покупает.

Кластер 1 - эта группа посетителей является поклонниками M5, т.к. они сразу подходят к автомобилям этой модели, практически полностью игнорируя автомобили серии 3 или Z4. Показатель покупки машин – 52%. Это может свидетельствовать о недостаточно продуманной стратегии продаж и о необходимости улучшить работу дилерского центра.

Кластер 2 - данные этой группы статистически довольно разбросаны, и нельзя сделать каких-либо определенных заключений относительно поведения посетителей, попавших в этот кластер (возможно, нужно подобрать другое количество кластеров в модели).

Кластер 3 - в эту группу попадают посетители, которые всегда покупают машину и всегда получают одобрение по кредиту. Данные этого кластера демонстрируют модель поведения таких покупателей:

сначала они осматривают выставленные на парковке машины, а затем обращаются к поисковой системе дилерского центра. Как правило, они покупают модели M5 или Z4, но никогда не берут модели третьей серии. Данные этого кластера указывают на то, что дилерскому центру следует активнее привлекать внимание к поисковым компьютерам, и кроме того, следует найти какой-нибудь способ выделить модели M5 и Z4 в результатах поиска, чтобы гарантированно обратить на них внимание посетителей. После того, как посетитель, попавший в этот кластер, выбрал определенную модель автомобиля, он гарантированно получает необходимый кредит и совершает покупку.

Кластер 4 – посетители этой группы всегда ищут модели 3 серии и никогда не интересуются более дорогими моделями. Они сразу же проходят в демонстрационный зал, не тратя время на осмотр машин на внешней стоянке. Кроме того, они практически не пользуются поисковой системой центра. Примерно 50% этой группы получают одобрение по кредиту, тем не менее, покупку совершают всего 32% участников. Анализируя данные этого кластера, можно сделать следующий вывод: посетители этой группы хотели бы купить автомобиль и точно знают, какая машина им нужна (3 серия). Однако, для того чтобы купить машину, им нужно получить положительное решение по кредиту. Чтобы повысить уровень продаж среди посетителей 4 кластера, дилерскому центру следовало бы понизить уровень требований для получения кредита или снизить цены на модели 3 серии.

Рассмотрим возможности WEKA для визуального представления данных. Для этого нужно щёлкнуть правой кнопкой мышки в секции Result List и в контекстном меню выбрать опцию Visualize Cluster Assignments. В результате откроется окно с графическим представлением результатов кластеризации (см. Рис. 9.49).

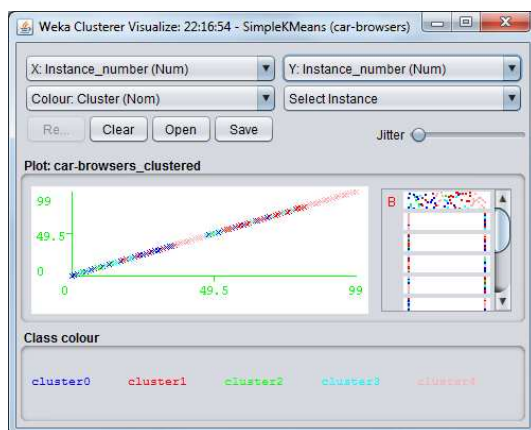


Рис. 9.49 Окно с графическим представлением результатов кластеризации

В секции Class Color, кликнув на название кластера, установите каждому кластеру свой цвет. Кроме того, двигая указатель Jitter, увеличьте разброс между группами точек - чтобы было удобнее их просматривать.

Измените настройку оси X так, чтобы она соответствовала количеству автомобилей M5 - M5 (Num), а настройку оси Y – так, чтобы она показывала количество купленных автомобилей - Purchase (Num) (см. Рис. 9.50). Такие настройки помогут оценить распределение по кластерам в зависимости от того, сколько человек интересовалось моделью M5, и сколько человек купило эту модель.

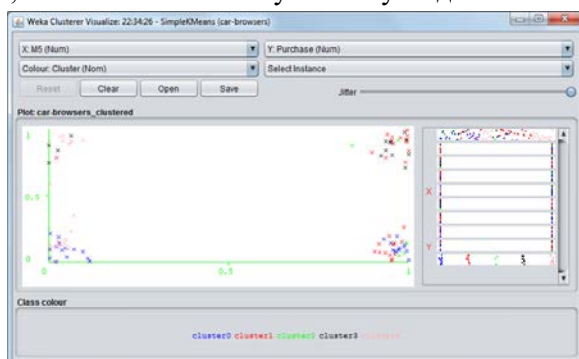


Рис. 9.50 Окно с графическим представлением результатов кластеризации

Соответствует ли визуальное отображение кластеризации тем заключениям, которые были сделаны ранее? Как можно заметить, в окрестности точки  $X=1$ ,  $Y=1$  (посетители, которые интересовались автомобилями модели M5 и купили их) расположены только два кластера: 1 и 3. Аналогично, в окрестности точки  $X=0$ ,  $Y=0$  расположены только два кластера: 4 и 0. Соответствует ли это сделанным выводам? Да, соответствует. Кластеры 1 и 3 покупают BMW M5, в то время как кластер 0 не покупает ничего, а кластер 4 ищет BMW серии 3.

Чтобы посмотреть, к какому кластеру относится какой посетитель нажмите кнопку Save, после чего WEKA сгенерирует файл с результатами кластеризации, пример которого представлен на Рисунке 9.51.

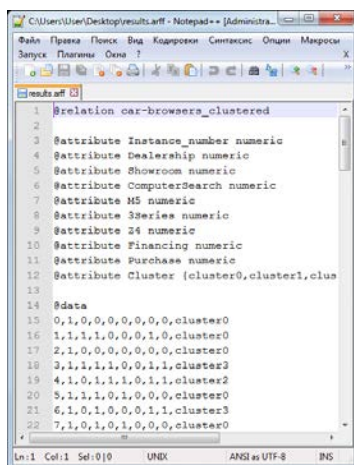


Рис. 9.51 Файл с результатами кластеризации

### Пример задачи Ассоциации

В ходе решения задачи поиска ассоциативных правил отыскиваются закономерности между связанными событиями в наборе данных. Стоит отметить, что в бесплатной версии WEKA алгоритмы ассоциации заблокированы.

Для того чтобы решить задачу ассоциации нужно выполнить следующие действия:

1. Загрузить файл БД в WEKA, например, weather.numeric.arff.

2. Открыть закладку Associate. В панели Chose выбрать опцию Apriori. Результат работы программы представлен на Рисунке 9.52.

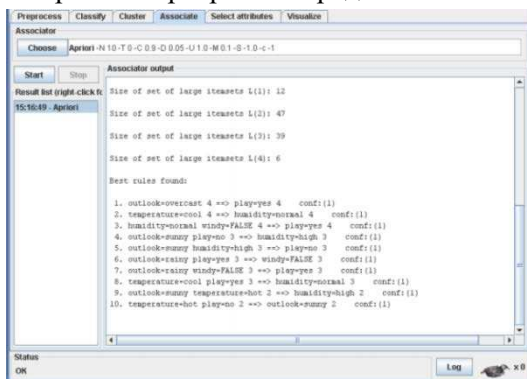


Рис. 9.52 Результат работы программы

## Модуль Experimenter

Этот модуль позволяет проводить эксперименты: запускать несколько алгоритмов на нескольких задачах и получать сводный отчёт (см. Рис. 9.53).



Рис. 9.53 Окно Weka GUI Chooser

На вкладке Setup (см. Рис. 9.54) изначально активны две кнопки: Open (открыть файл эксперимента) и New (создать новый эксперимент).



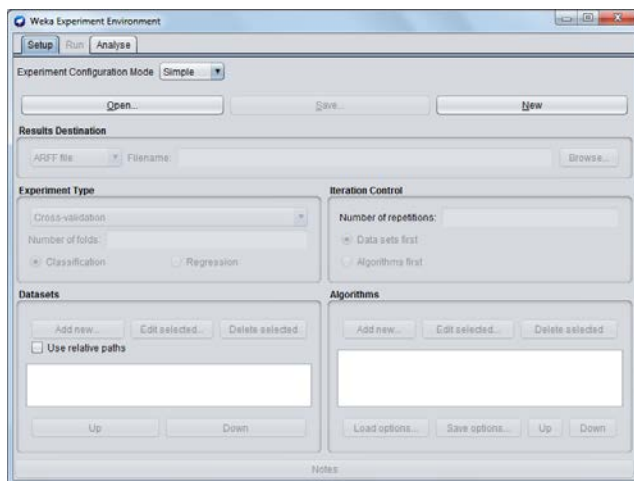


Рис. 9.54 Вкладка Setup

Для создания нового эксперимента нужно выполнить следующие действия:

1. Нажать на кнопку New, после чего активируются все опции.
2. На панели Result Destination выбрать или создать файл для записи отчёта, например, Experimenter\_Results.
3. На панели Experiment Type выбрать тип эксперимента: проводить верификацию методом контроля по блокам/фолдам или использовать разделение выборки на контроль/обучение: чем больше фолдов будет указано, тем дольше будут идти эксперименты.
4. На панели Iteration Control выбрать число повторений экспериментов. При повторениях происходят новые разбиения на обучение/контроль и на фолды. Также здесь указывается порядок проведения эксперимента: перебирать сначала все задачи (Data set first) или все алгоритмы (Algorithms first).
5. На панели Datasets нажать кнопку AddNew и выбрать базы данных для проведения эксперимента, например, weather.arff и iris.arff (эти БД можно найти в папке \_адрес\_установки\_программы\Weka-3-8\data\\_имя\_БД).
6. На панели Algorithms нажать кнопку AddNew, потом на кнопку Choose и выбрать алгоритм, например: weka.classifiers.bayes.NaiveBayes, нажать на кнопку Ок (см. Рис. 5.55).

Аналогичным образом выбрать еще несколько алгоритмов, например: `weka.classifiers.functions.MultilayerPerceptron`, `weka.classifiers.lazy.IBK`, `weka.classifiers.trees.J48`.

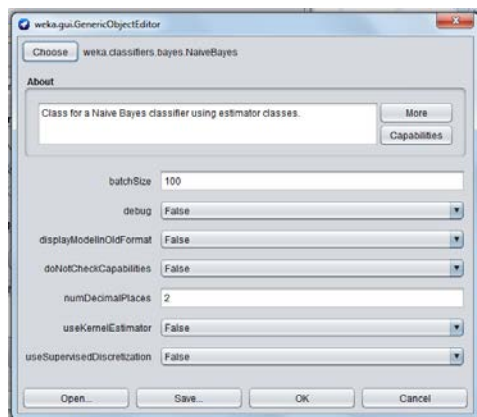


Рис. 9.55 Выбор алгоритма

Вид вкладки **Setup** после выполненных действий представлен на Рисунке 9.56.

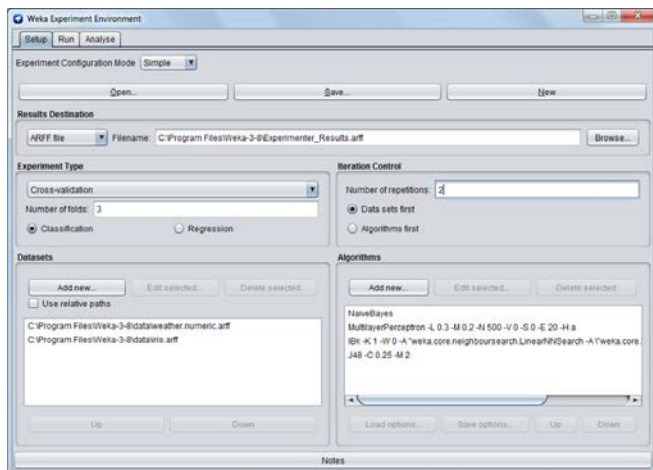


Рис. 9.56 Настроенная вкладка Setup

7. Перейти на вкладку **Run** и нажать кнопку **Start**: после чего в окне **Log** будут выведены сообщения о проведении эксперимента: время старта и окончания эксперимента, сообщения об ошибках (см. Рис. 9.57). Результаты проведения эксперимента записываются в файл, указанный в пункте 2.

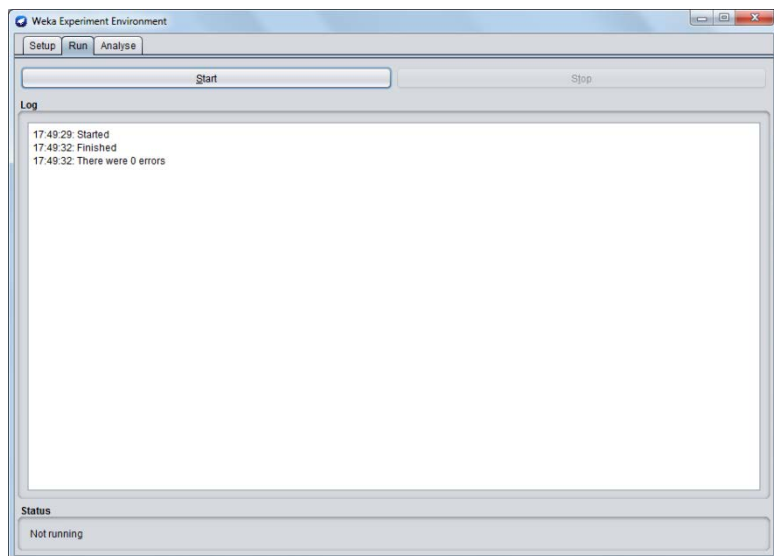


Рис. 9.57 Запуск эксперимента

8. Перейти на вкладку *Analyse* - она позволяет анализировать результаты экспериментов.

9. На панели *Source* выбрать эксперимент для анализа (только что проведённый или записанный в файл): кликнуть на кнопку *File* выбрать файл, созданный на втором шаге.

10. На панели *Configure test* определить вид статистики для анализа. Кнопка *Rows* позволяет выбрать, что будет записано по строкам выводимой матрицы (для примера выберем из открывшегося списка *Dataset*, *Run*, *Fold*), а кнопка *Column* – что будет записано по столбцам (выберем *Scheme*). В поле *Comparison field* выбирается, чем будет заполнена таблица (выбор *Percent\_correct* обеспечивает заполнение процентами верной классификации). Остальные кнопки и опции также позволяют уточнить вид отчёта. Например, при выборе *Output Format/Output Format/LaTeX* будет произведена генерация TeX-файла (по умолчанию выбрана опция *Plain Text*). Вид вкладки *Analyse* после выполненных действий представлен на Рисунке 9.58.

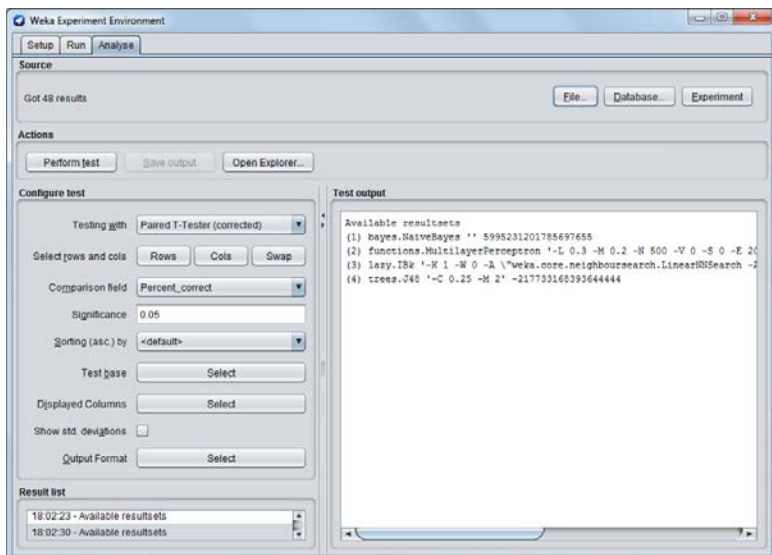


Рис. 9.58 Вкладка Analyse

11. Нажать кнопку Perform test на панели Actions для генерации отчёта. Вид вкладки Analyse после выполненных действий представлен на Рисунке 9.59.

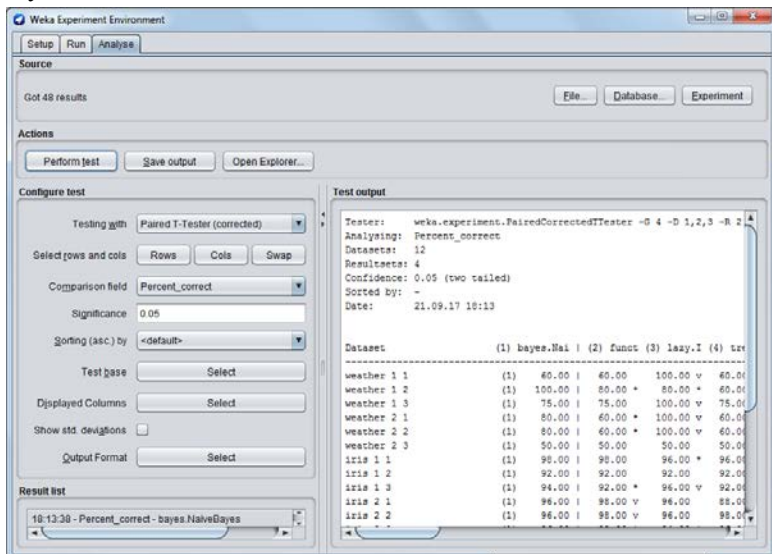


Рис. 9.59 Вкладка Analyse после проведения эксперимента  
Рассмотрим результаты работы программы (см. Рис. 9.60).

Dataset		(1) bayes.Nai	(2) funct	(3) lazy.I	(4) trees
weather 1 1	(1)	60.00	60.00	100.00 v	60.00
weather 1 2	(1)	100.00	80.00 *	80.00 *	60.00 *
weather 1 3	(1)	75.00	75.00	100.00 v	75.00
weather 2 1	(1)	80.00	60.00 *	100.00 v	60.00 *
weather 2 2	(1)	80.00	60.00 *	100.00 v	60.00 *
weather 2 3	(1)	50.00	50.00	50.00	50.00
iris 1 1	(1)	98.00	98.00	96.00 *	96.00 *
iris 1 2	(1)	92.00	92.00	92.00	92.00
iris 1 3	(1)	94.00	92.00 *	96.00 v	92.00 *
iris 2 1	(1)	96.00	98.00 v	96.00	88.00 *
iris 2 2	(1)	96.00	98.00 v	96.00	98.00 v
iris 2 3	(1)	96.00	92.00 *	94.00 *	96.00

Рис. 9.60 Результаты работы программы

Здесь показана статистика работы на двух задачах iris и weather четырёх алгоритмов: наивного байесовского классификатора (bayes.NaiveBayes), многослойного персептрона (functions.MultilayerPerceptron), ближайшего соседа (lazy.IB1) и одного из алгоритмов построения решающего дерева (trees.J48).

Запись «weather 1 3» означает, что в соответствующей строке собрана статистика работы на задаче weather в первом запуске на третьем фолде, т.к. кнопкой Row были выбраны пункты Dataset, Run, Fold.

Для того чтобы посмотреть средние результаты по всем фолдам (см. Рис. 5.62), нужно на панели Configure test, кликнуть на OutPutFormat и чекнуть опцию ShowAverage (см. Рис. 9.61).

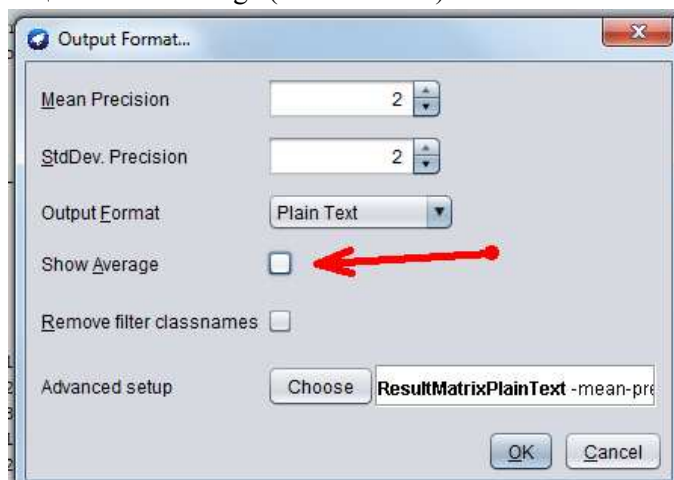


Рис. 9.61 Опция OutPutFormat

Dataset	(1)	bayes.Nai	(2)	funct	(3)	lazy.I	(4)	trees
weather 1 1	(1)	60.00		60.00	100.00	v	60.00	
weather 1 2	(1)	100.00		80.00	*	80.00	*	60.00 *
weather 1 3	(1)	75.00		75.00	100.00	v	75.00	
weather 2 1	(1)	80.00		60.00	*	100.00	v	60.00 *
weather 2 2	(1)	80.00		60.00	*	100.00	v	60.00 *
weather 2 3	(1)	50.00		50.00	50.00			50.00
iris 1 1	(1)	98.00		98.00	96.00	*	96.00	*
iris 1 2	(1)	92.00		92.00	92.00			92.00
iris 1 3	(1)	94.00		92.00	*	96.00	v	92.00 *
iris 2 1	(1)	96.00		98.00	v	96.00		88.00 *
iris 2 2	(1)	96.00		98.00	v	96.00		98.00 v
iris 2 3	(1)	96.00		92.00	*	94.00	*	96.00
Average		84.75		79.58		91.67		77.25

Рис. 9.62 Средние результаты эксперимента

## ЗАДАНИЕ НА ЛАБОРАТОРНУЮ РАБОТУ

1. Создайте регрессионную модель расчета расхода бензина (MPG - количества миль на галлон), исходя из нескольких параметров автомобиля. Модель учитывает несколько параметров машины – количество цилиндров, рабочий объем двигателя, его мощность, вес автомобиля, время разгона, год выпуска, производителя и марку автомобиля. БД можно найти по следующему адресу: <https://cs.nyu.edu/courses/fall00/G22.3033-001/weka/weka-3-0-2/data/auto-mpg.arff>

2. Создайте регрессионную модель расчета стоимости машины модели M5. Модель в качестве независимых параметров будет учитывать данные проданных автомобилей и параметры модели M5, а в качестве зависимого параметра – стоимость автомобилей, проданных дилерским центром.

3. Решите задачу Фишера о классификации цветков ириса. БД можно найти по следующему адресу: <http://archive.ics.uci.edu/ml/datasets/Iris>

4. Решите задачу классификации дней в зависимости от погоды. БД можно найти в папке \_адрес\_установки\Weka-3-8\data\weather.nominal.arff

5. Решите задачу классификации стекла в зависимости от типа. БД можно найти в папке \_адрес\_установки\Weka-3-8\data\glass.arff

6. Решите задачу кластеризации цветков ириса. БД можно найти по следующему адресу: <http://archive.ics.uci.edu/ml/datasets/Iris>

Решите задачу кластеризации дней в зависимости от погоды. БД можно найти в папке \_адрес\_установки\Weka-3-8\data\weather.nominal.arff

## **ФОРМА ОТЧЕТА ПО ЛАБОРАТОРНОЙ РАБОТЕ**

На выполнение лабораторной работы отводится 2 занятия (4 академических часа: 3 часа на выполнение и сдачу лабораторной работы и 1 час на подготовку отчета).

Отчет на защиту предоставляется в печатном виде.

Структура отчета (на отдельном листе(-ах)): титульный лист, формулировка задания, описание процесса выполнения лабораторной работы, результаты выполнения работы, выводы.