

Министерство науки и высшего образования Российской Федерации  
**Муромский институт (филиал)**  
федерального государственного бюджетного образовательного учреждения  
высшего образования  
**«Владимирский государственный университет  
имени Александра Григорьевича и Николая Григорьевича Столетовых»  
(МИ ВлГУ)**

**А.Д. ВАРЛАМОВ**

## **ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ**

Практикум

для студентов образовательной программы  
09.03.02 *Информационные системы и технологии*

Текстовое электронное издание

Учебно-методический центр МИ ВлГУ  
Муром 2019

© Варламов А.Д., 2019

© МИ ВлГУ, 2019

**УДК [004.8 + 004.93] (075.8)**  
**ББК 32.813**

**Ответственный за выпуск:**

заведующий кафедрой информационных систем,  
доктор технических наук Андрианов Дмитрий Евгеньевич

Варламов А.Д. Интеллектуальные системы и технологии: Практикум для студентов образовательной программы 09.03.02 Информационные системы и технологии / [Электронный ресурс]. – Электрон. текстовые дан. (2,8 Мб). - Мум.: МИ ВлГУ, 2019. - 1 электрон. опт. диск (CD-R). – Систем. требования: процессор x86 с тактовой частотой 500 МГц и выше; 512 Мб ОЗУ; Windows XP/7/8; видеокарта SVGA 1280x1024 High Color (32 bit); привод CD-ROM. - Загл. с экрана.

Практикум содержит сведения, необходимые для выполнения лабораторных работ по дисциплине «Интеллектуальные системы и технологии» для студентов образовательной программы 09.03.02 Информационные системы и технологии. Тематика работ направлена на приобретение студентами знаний и навыков, необходимых для освоения специальных дисциплин и выполнения выпускной квалификационной работы.

**Текстовое электронное издание**

**Минимальные системные требования:**

Компьютер: процессор x86 с тактовой частотой 500 МГц и выше; ОЗУ 512 Мб;  
10 Мб на жестком диске; видеокарта SVGA 1280x1024 High Color (32 bit);  
привод CD-ROM

Операционная система: Windows XP/7/8

Программное обеспечение: Adobe Acrobat Reader версии 6 и старше.

© Варламов А.Д., 2019  
© МИ ВлГУ, 2019

## Содержание

Лабораторная работа №1 «МАШИННОЕ ОБУЧЕНИЕ С УЧИТЕЛЕМ. ОСНОВЫ РАБОТЫ С НЕЙРОННЫМИ СЕТЯМИ» .....	5
1. Цель работы.....	5
2. Методические указания.....	5
2.1. Общие сведения о нейронных сетях и программах-нейроимитаторах .....	5
2.2. Типовой пример выполнения работы.....	6
2.2.1 Пример выполнения работы в Deductor Studio Academic.....	6
2.2.2 Пример выполнения работы в NNTool .....	12
3. Задание на лабораторную работу.....	19
4. Требования к содержанию отчета .....	19
5. Контрольные вопросы .....	19
6. Индивидуальные задания. ....	20
Лабораторная работа №2 «ПОДГОТОВКА ДАННЫХ ДЛЯ МАШИННОГО ОБУЧЕНИЯ И ОЦЕНКА КАЧЕСТВА ОБУЧЕННОГО КЛАССИФИКАТОРА» .....	25
1. Цель работы.....	25
2. Методические указания.....	25
2.1. Введение .....	25
2.2. Понижение размерности данных.....	25
2.3. Проверка репрезентативности выборки и анализ результатов .....	26
2.4. Типовой пример выполнения работы.....	26
3. Задание на лабораторную работу.....	32
3.1. Порядок выполнения работы.....	32
3.2. Требования к работе .....	33
4. Требования к содержанию отчета .....	33
5. Контрольные вопросы .....	33
6. Индивидуальные задания .....	33
Лабораторная работа №3 «ПРОГРАММНАЯ РЕАЛИЗАЦИЯ РЕЗУЛЬТАТОВ МАШИННОГО ОБУЧЕНИЯ» .....	34
1. Цель работы.....	34
2. Методические указания.....	34
2.1. Типовой пример решения задачи экспертного выбора .....	34
2.2. Типовой пример решения задачи экспертного выбора на основе модели линейной регрессии.....	38
3. Задание на лабораторную работу.....	41
4. Требования к содержанию отчета .....	41
5. Контрольные вопросы .....	41
6. Индивидуальные задания .....	42
Лабораторная работа №4 «ТЕХНОЛОГИИ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА В ОБРАБОТКЕ И АНАЛИЗЕ ИЗОБРАЖЕНИЙ» .....	47
1. Цель работы.....	47
2. Методические указания.....	47
2.1. Задача классификации изображений.....	47
2.2. Типовой пример выполнения работы классификации изображений .....	47

2.3. Задача обработки изображений .....	49
2.4. Типовой пример выполнения работы обработки изображений .....	50
3. Задание на лабораторную работу .....	51
4. Требования к содержанию отчета .....	51
5. Контрольные вопросы .....	51
6. Индивидуальные задания .....	52
Библиографический список.....	53

## Лабораторная работа №1

### «МАШИННОЕ ОБУЧЕНИЕ С УЧИТЕЛЕМ. ОСНОВЫ РАБОТЫ С НЕЙРОННЫМИ СЕТЯМИ»

#### 1. Цель работы

Приобретение навыков построения и обучения нейронных сетей при помощи программ-нейроимитаторов (нейросимуляторов) и их использование в практических задачах.

#### 2. Методические указания

##### 2.1. Общие сведения о нейронных сетях и программах-нейроимитаторах

Нейронные сети представляют собой самообучающиеся модели, имитирующие деятельность человеческого мозга. Они способны не только выполнять однажды запрограммированную последовательность действий над заранее определенными данными, но и сами анализировать вновь поступающую информацию.

Идея нейронных сетей родилась в рамках теории искусственного интеллекта, в результате попыток имитировать способность биологических нервных систем обучаться и исправлять ошибки. Изучение и использование формальных нейронных сетей началось в начале 20 века, но широкую известность они получили позже, когда стали появляться вычислительные устройства, мощности которых были достаточны для работы с искусственными нейронными сетями.

По аналогии с биологической сетью формальная нейронная сеть состоит из (формальных) нейронов. Соединяясь друг с другом, нейроны образуют нейронные сети (рис. 1.1).

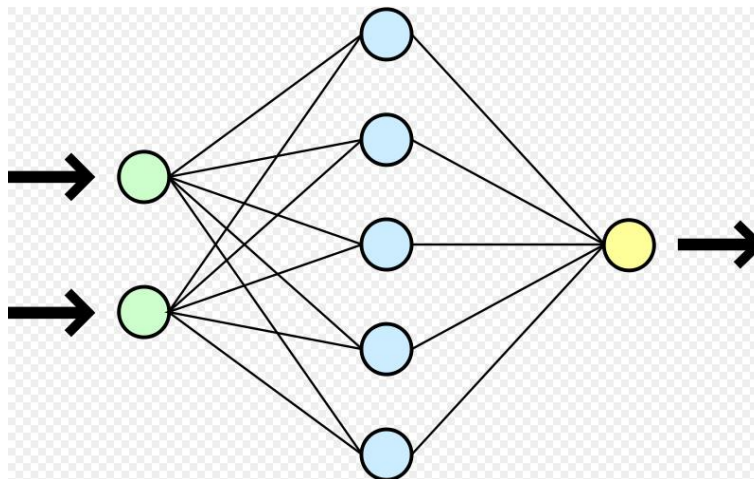


Рис. 1.1. Пример структуры нейронной сети

Нейронная сеть способна обучаться решению задач, для которых у человека не существует формализованных, быстрых или работающих с приемлемой точностью теоретических или эмпирических алгоритмов. Наряду с обучающими данными требуется лишь задать некоторый критерий качества решения задачи, который нейронная сеть при своём обучении должна будет минимизировать или оптимизировать.

В числе задач, решение которых доверяют искусственным нейронным сетям, можно назвать следующие: распознавание текстов, игра на бирже, контекстная реклама в Интернете, фильтрация спама, проверка проведения подозрительных операций по банковским картам, системы безопасности и видеонаблюдения и многие другие [4, 6].

## 2.2. Типовой пример выполнения работы

Рассмотрим типовой пример решения задачи:

Петя	Было желание съездить	Дали отпуск	Бизнес в норме	Поехал на юг
Вася	Было желание съездить	Не дали отпуск	Бизнес в норме	Не поехал на юг
Витя	Не было желания съездить	Дали отпуск	Фирма обанкротилась	Не поехал на юг
Вова	Не было желание съездить	Не дали отпуск	Бизнес в норме	Не поехал на юг
Антон	Не было желание съездить	Дали отпуск	Бизнес в норме	Поедет ли он в отпуск?

Вариант содержит набор прецедентов, то есть ситуаций из прошлого с известным результатом (данные о Пете, Васе, Вите, Вове) и новые данные (сведения по Антону), по которым необходимо предсказать пока еще не известный результат.

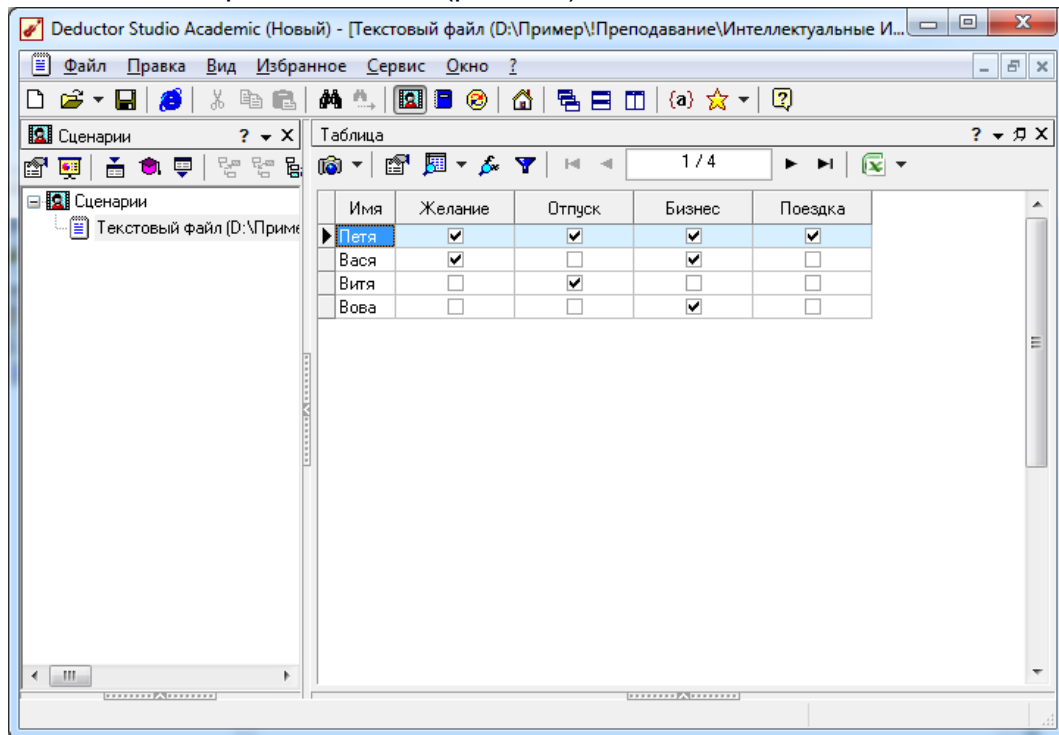
Нейронную сеть можно обучить на тех данных, для которых известны выходные данные (поездка на юг). Обученная нейронная сеть способна предсказывать результат для нового объекта, не входящего в обучающую выборку. Этот этап называется прогоном сети. Так, в рассматриваемом варианте задания требуется узнать, поедет ли Антон на юг.

### 2.2.1 Пример выполнения работы в Deductor Studio Academic

Для работы в Deductor Studio Academic исходные данные (четыре прецедента, прописанные в четырех первых строчках таблицы) необходимо представить в формате \*.csv (текстовый формат с разделителями полей). Текстовые значения ячеек таблицы прецедентов следует заменить условными значениями: 1 – истина, а 0 – ложь. Исходные данные в данном формате будут выглядеть следующим образом:

Имя;Желание;Отпуск;Бизнес;Поездка
Петя;1;1;1;1
Вася;1;0;1;0
Витя;0;1;0;0
Вова;0;0;1;0

При помощи мастера импорта загрузим эти данные в программу. Получим их табличное представление (рис. 1.2).

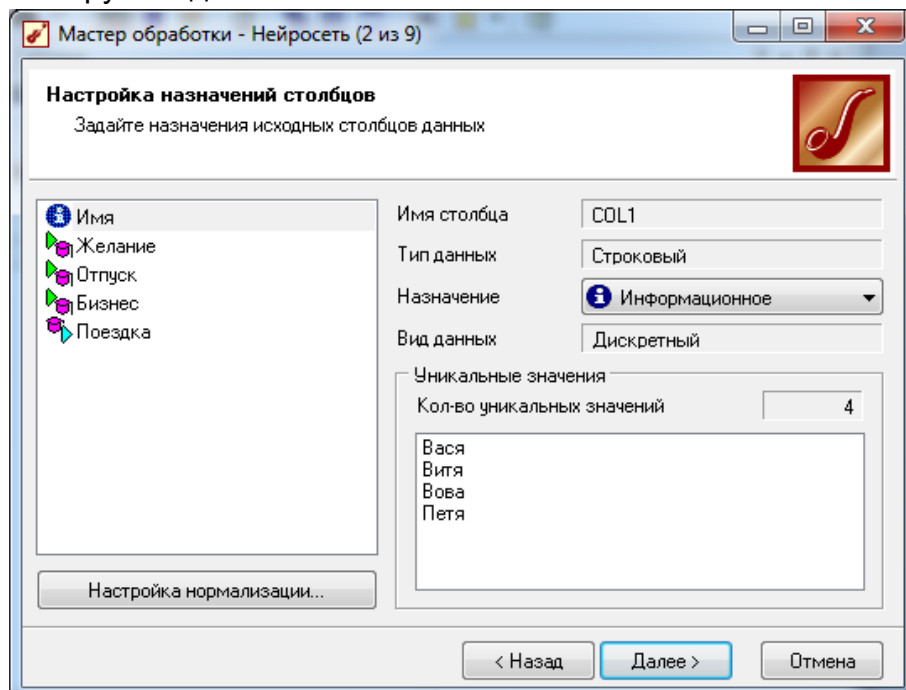


The screenshot shows the 'Deductor Studio Academic' window. On the left, a tree view shows 'Сценарии' (Scenarios) with a sub-item 'Текстовый файл (D:\Приме...)' (Text file (D:\Primer...)). The main area displays a table with the following data:

Имя	Желание	Отпуск	Бизнес	Поездка
Петя	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Вася	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Витя	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Вова	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Рис. 1.2. Результат импорта данных

Вызвав мастер обработки "Нейросеть", проверим типы данных полей и настроим их назначение. "Имя" – строковое, информационное поле; "Желание", "Отпуск", "Бизнес" – логические, входные данные; "Поездка" – логическое, выходное поле. Если типы данные другие, нужно вернуться к предыдущему шагу и настроить параметры мастера импорта. При наличии поля с неизменяемым назначением "непригодное" необходимо исправить ошибки в файле .csv и снова попытаться загрузить данные.



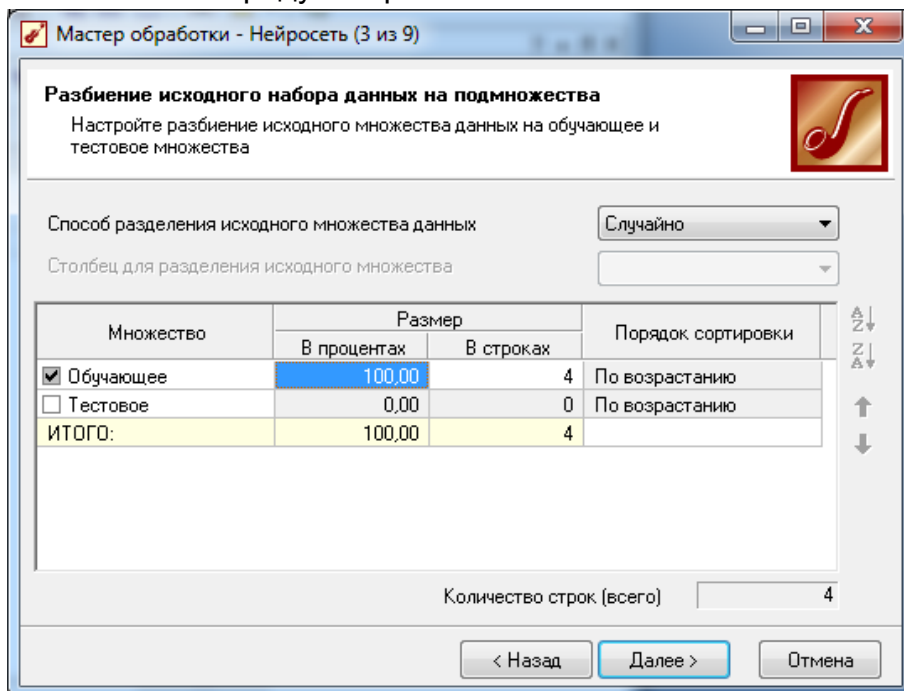
The screenshot shows the 'Мастер обработки - Нейросеть (2 из 9)' (Data Processing Master - Neural Network (2 of 9)) window. The title bar indicates it's step 2 of 9. The main area is titled 'Настройка назначений столбцов' (Column Assignment Settings) with the instruction 'Задайте назначения исходных столбцов данных' (Specify the assignments of the original data columns). On the left, a list of columns is shown: 'Имя' (Name), 'Желание' (Wish), 'Отпуск' (Vacation), 'Бизнес' (Business), and 'Поездка' (Trip). The 'Имя' column is selected. On the right, the settings for the selected column are displayed:

- Имя столбца: COL1
- Тип данных: Строковый (String)
- Назначение: Информационное (Informational)
- Вид данных: Дискретный (Discrete)
- Уникальные значения: Кол-во уникальных значений: 4
- Уникальные значения (List): Вася, Витя, Вова, Петя

At the bottom, there are buttons for '< Назад' (Back), 'Далее >' (Next), and 'Отмена' (Cancel).

Рис. 1.3. Настройка назначений полей

На следующем этапе надо указать, что все прецеденты следует отнести к обучающему множеству (100% строк). В рамках данной работы использование тестового множества не предусмотрено.



**Мастер обработки - Нейросеть (3 из 9)**

**Разбиение исходного набора данных на подмножества**  
Настройте разбиение исходного множества данных на обучающее и тестовое множества

Способ разделения исходного множества данных: Случайно

Столбец для разделения исходного множества:

Множество	Размер		Порядок сортировки
	В процентах	В строках	
<input checked="" type="checkbox"/> Обучающее	100,00	4	По возрастанию
<input type="checkbox"/> Тестовое	0,00	0	По возрастанию
<b>ИТОГО:</b>	100,00	4	

Количество строк (всего): 4

< Назад    Далее >    Отмена

Рис. 1.4. Отнесение всех данных к тестовому множеству

Далее мастер предлагает нам настроить структуру нейронной сети. Количество нейронов входного и выходного слоя менять нельзя, так как они зависят от числа полей входных и выходных данных. Можно настроить количество скрытых слоев, количество нейронов в каждом из них, тип нелинейной функции формального нейрона и ее параметры.

Отметим: тип архитектуры данной нейронной сети – многослойный персептрон.



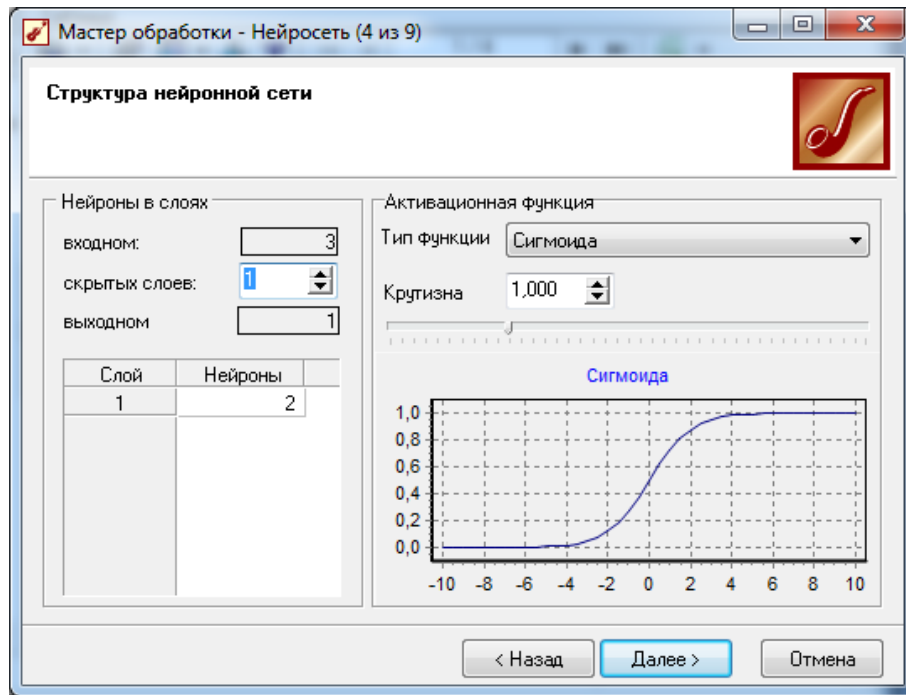


Рис. 1.5. Задание параметров структуры нейронной сети

На шестом этапе диалога применения мастера обработки задаются условия остановки процесса обучения. По умолчанию установлены следующие параметры: обучение прекратится, если все примеры обучающей выборки будут распознаны, либо пройдет 10000 эпох.

Отметим: эпоха – одна итерация цикла обучения нейронной сети, когда происходит корректирование ее параметров.

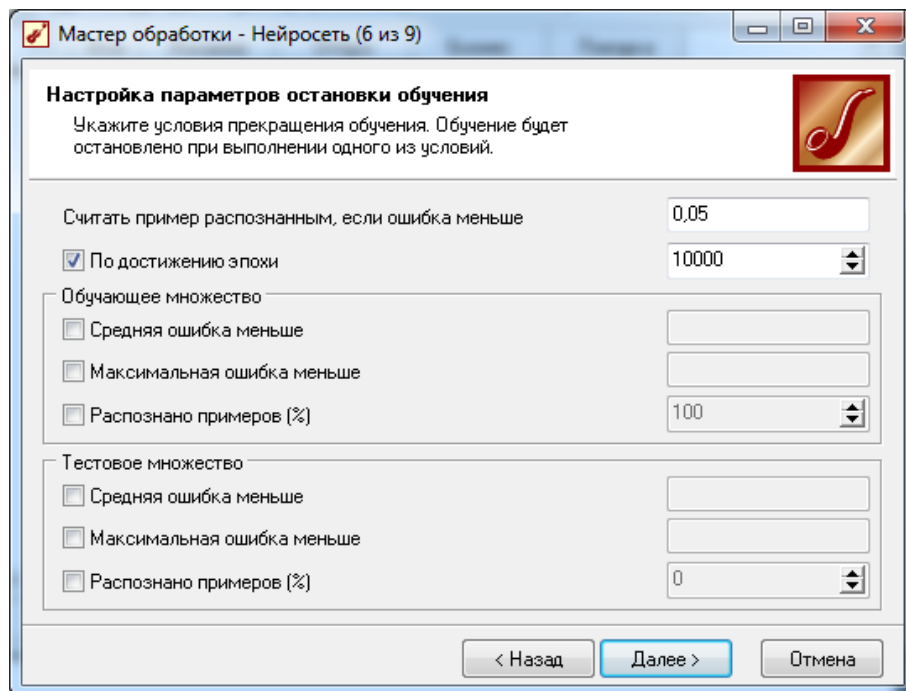


Рис. 1.6. Настройка параметров остановки обучения

И, наконец, можно запустить процесс обучения, нажав на кнопку "Пуск".

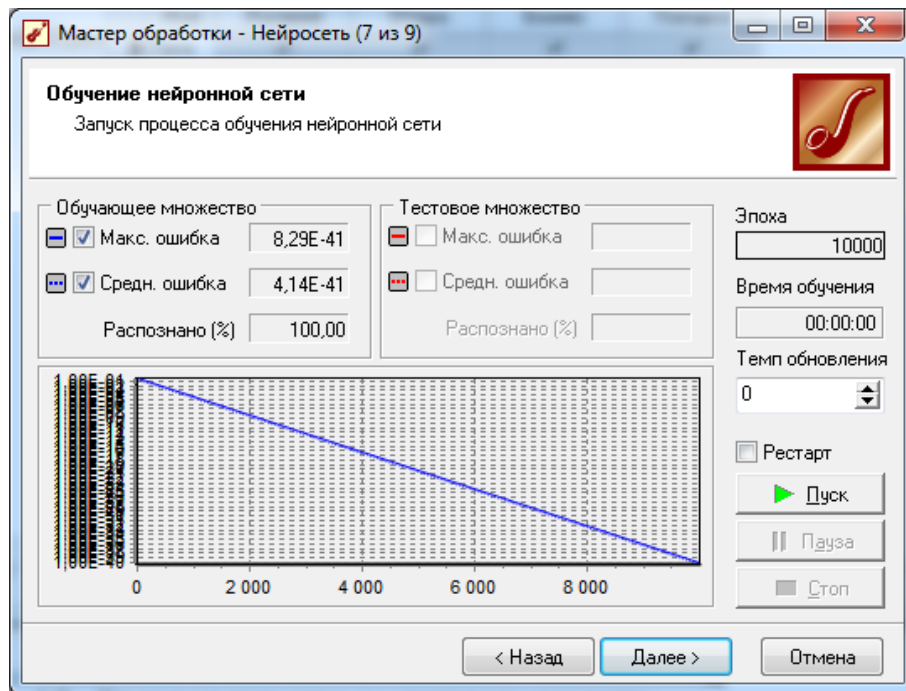


Рис. 1.7. Процесс обучения

На графике отображается изменение средней ошибки в процессе обучения. По снижению величины ошибки и приближению ее к нулю можно сделать вывод, что нейронная сеть хорошо обучилась. Перейдя на следующую страницу после завершения обучения, укажем способы отображения данных: "Граф нейросети", "Таблица сопряженности", "Что-если".

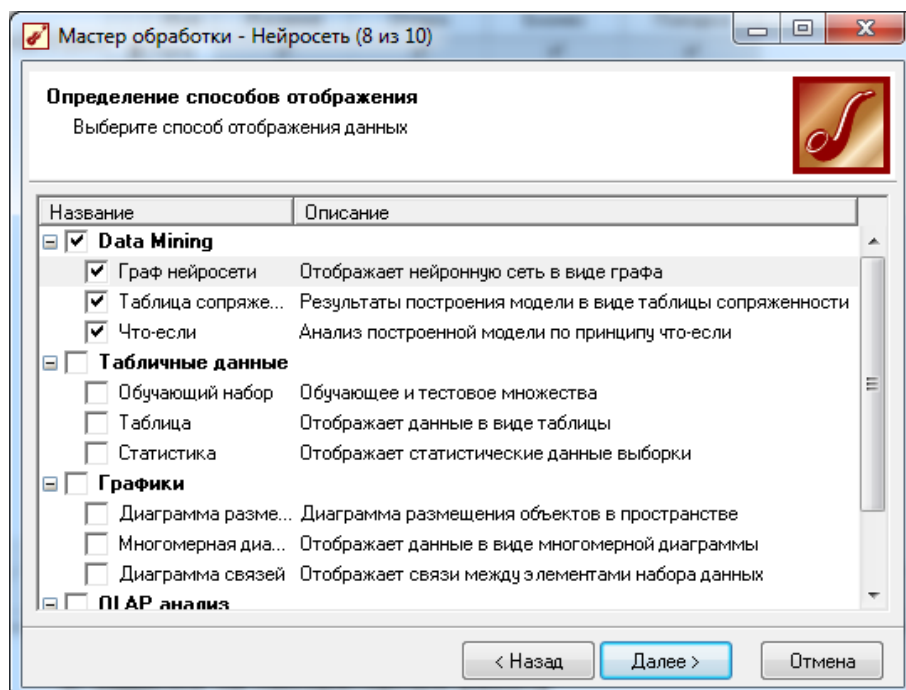


Рис. 1.8. Выбор способов отображения результатов

На вкладке "Граф нейросети" следующей страницы показана структура нейронной сети.

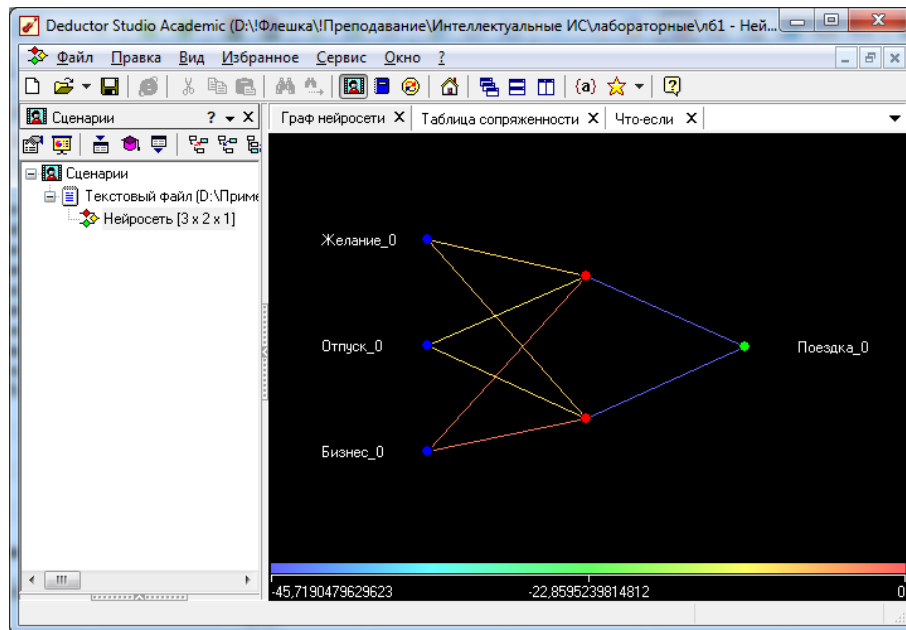


Рис. 1.9. Визуализация структуры нейронной сети

Таблица сопряженности позволяет статистически оценить результаты обучения. В ней все прецеденты выборки разделены на четыре группы, две из которых соответствуют распознанным примерам и две нераспознанным. По таблице сопряженности мы удостоверяемся, что все примеры распознаны.

Фактически	Классифицировано		Итого
	False	True	
False	3	0	3
True	0	1	1
Итого	3	1	4

Рис. 1.10. Отнесение всех данных к тестовому множеству

Прогон нейронной сети можно выполнить, используя анализ "Что-если". Введя в поля входных данных сведения для Антона из таблицы варианта задания, получим результат в виде логического значения "True".

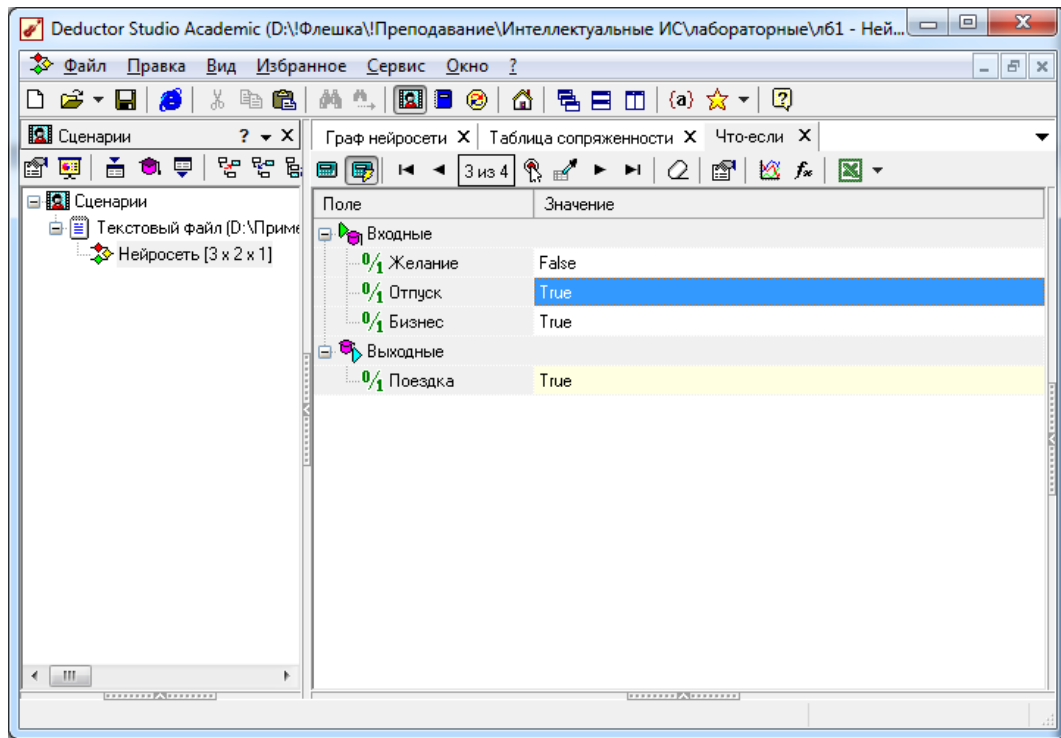


Рис. 1.11. Прогон нейронной сети

Результат прогона интерпретируется как "Антон поедет в отпуск".

### 2.2.2 Пример выполнения работы в NNTool

"NNTool" – это инструментальное средство программной среды MatLab. Чтобы запустить "NNTool", необходимо выполнить команду в командном окне:

MATLAB:

```
>> nntool
```

После этого появится главное окно инструмента "Network/DataManager" (рис. 1.12).

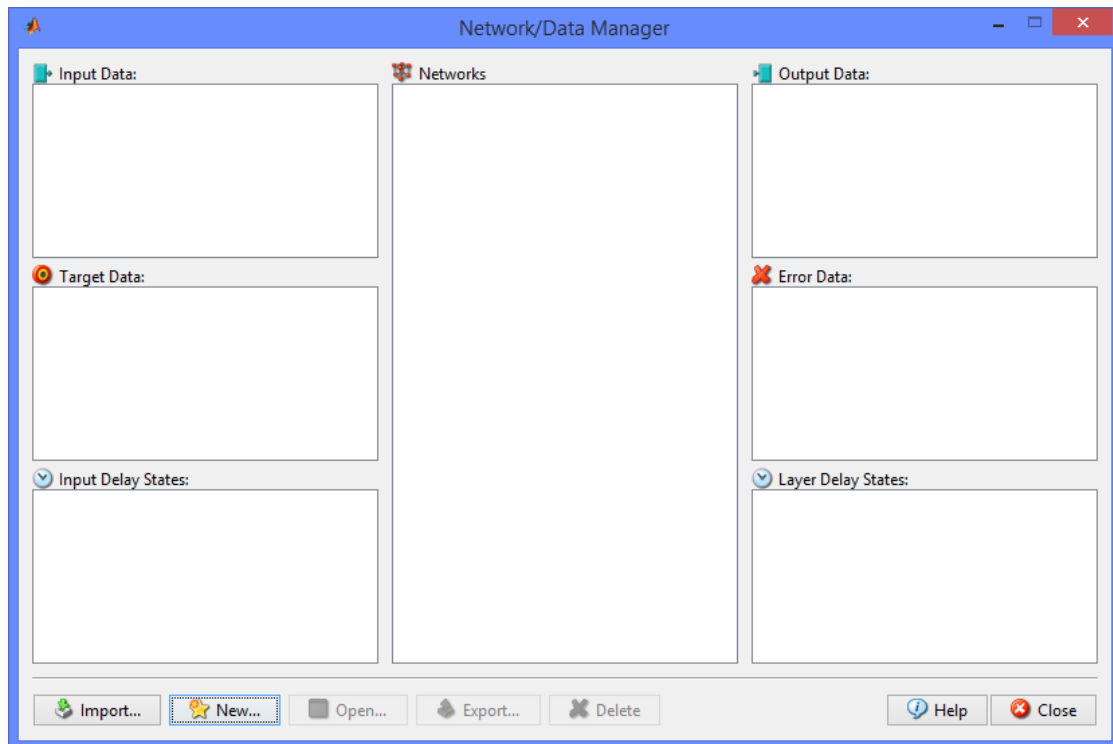


Рис. 1.12. Главное окно NNTool

Выберем нейронную сеть типа "Персептрон" с тремя входами (по количеству входных полей задачи). Перед созданием сети необходимо заготовить набор обучающих и целевых данных.

Отметим, что входные данные задачи (желание, отпуск и состояние бизнеса) и целевые значения (поездка на юг) задаются отдельно. Для входных данных указывается тип "Input", а для целевых – "Target". Воспользуемся кнопкой "New", чтобы задать исходные данные. В появившемся окне следует перейти на вкладку "Data" и произвести изменения, показанные на рис. 1.13.

The screenshot shows a dialog box titled "Create Network or Data" with a blue header bar. It has two tabs: "Network" and "Data", with "Data" being the active tab. The "Name" field contains the text "data1". The "Value" field contains a 3x4 matrix of numbers:  $\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}$ . To the right of the "Value" field is a "Data Type" section with six radio button options: "Inputs" (which is selected), "Targets", "Input Delay States", "Layer Delay States", "Outputs", and "Errors". At the bottom of the dialog, there are three buttons: "Help" (with an information icon), "Create" (with a star icon), and "Close" (with a red X icon).

Рис. 1.13. Задание входных данных для обучения

Вектор целей задаётся подобным образом (рис. 1.14).

The image shows a MATLAB dialog box titled "Create Network or Data". It has two tabs: "Network" and "Data", with "Data" currently selected. The "Name" field contains the text "target1". The "Value" field contains the vector "[1 0 0 0]". To the right of the "Value" field is a "Data Type" section with six radio button options: "Inputs", "Targets" (which is selected), "Input Delay States", "Layer Delay States", "Outputs", and "Errors". At the bottom of the dialog are three buttons: "Help" (with an information icon), "Create" (with a star icon), and "Close" (with a red X icon).

Рис. 1.14. Задание целевого вектора

После нажатия на "Create" в разделе "Targets" появится вектор "target1". Теперь следует приступить к созданию нейронной сети. Переходим во вкладку "Network" и заполняем форму, как показано на рис. 1.15.

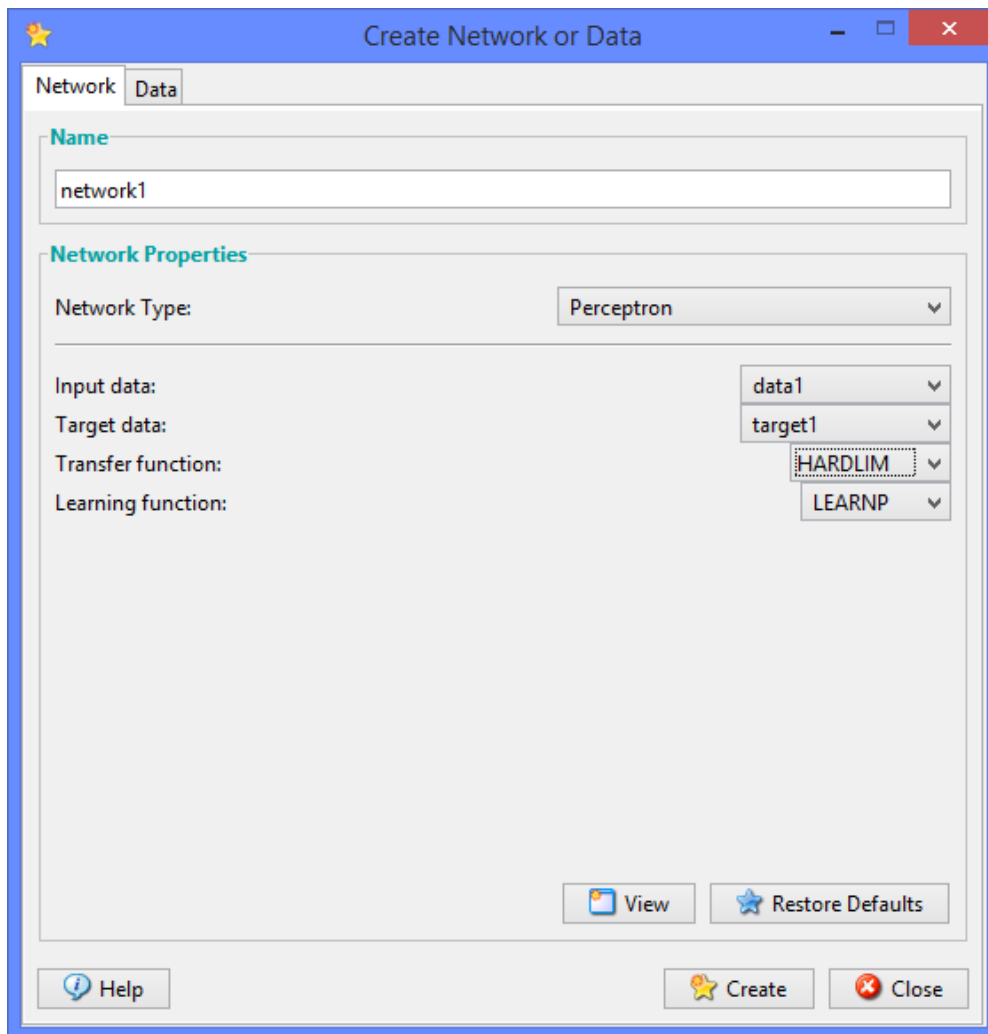


Рис. 1.15. Окно создания сети

При этом поля имеют следующие смысловые нагрузки:

- имя сети (NetworkName) – это имя объекта создаваемой сети;
- тип сети (NetworkType) – определяет тип архитектуры нейронной сети;
- входные диапазоны (Input ranges) – матрица с числом строк, равным числу входов сети;
- получить из входа (Getfrominput): позволяет автоматически сформировать необходимые данные, указав имя входной переменной;
- количество нейронов (Numberofneurons) – число нейронов в слое;
- передаточная функция (Transferfunction): в этом пункте выбирается передаточная функция (функция активации) нейронов;
- функция обучения (Learningfunction) – функция, отвечающая за обновление весов и смещений сети в процессе обучения.

С помощью клавиши "View" можно посмотреть архитектуру создаваемой сети (рис. 1.16).



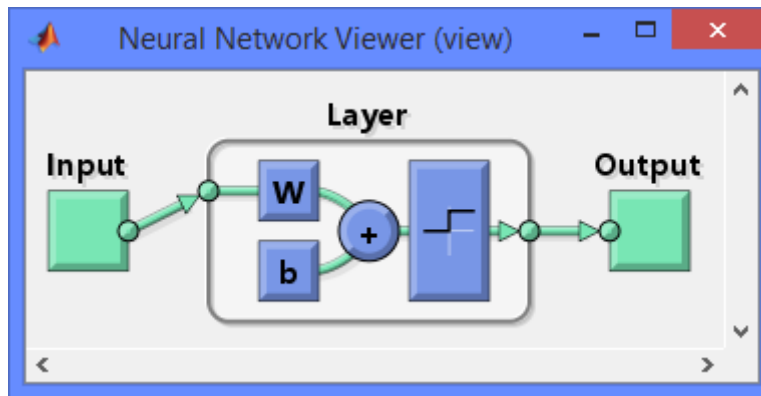


Рис. 1.16. Предварительный просмотр сконструированной сети

Отметив указателем мыши объект сети, вызовем окно управления сетью нажатием кнопки "Train". Перед нами появится вкладка "Train" окна свойств сети, содержащая, в свою очередь, ещё одну панель вкладок (рис. 1.17). Их главное назначение – управление процессом обучения. На вкладке "Информация обучения" ("Traininginfo") требуется указать набор обучающих данных в поле "Входы" ("Inputs") и набор целевых данных в поле "Цели" ("Targets"). Поля "Выходы" ("Outputs") и "Ошибки" (Errors) "NNTool" заполняет автоматически. При этом результаты обучения, к которым относятся выходы и ошибки, будут сохраняться в переменных с указанными именами.

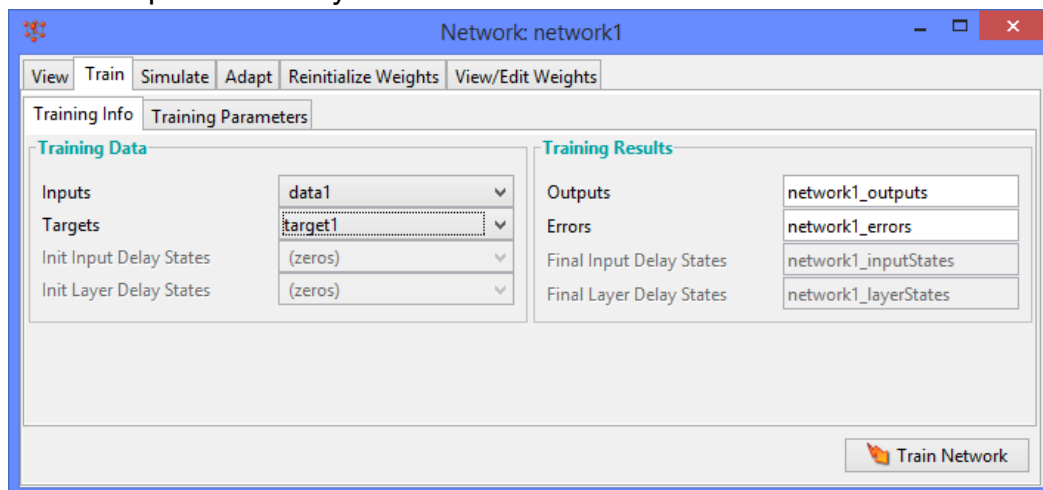


Рис. 1.17. Окно параметров нейронной сети

На вкладке "Параметры обучения" ("Trainingparameters") для нашей сети можно установить условия завершения обучения сети (рис. 1.18).

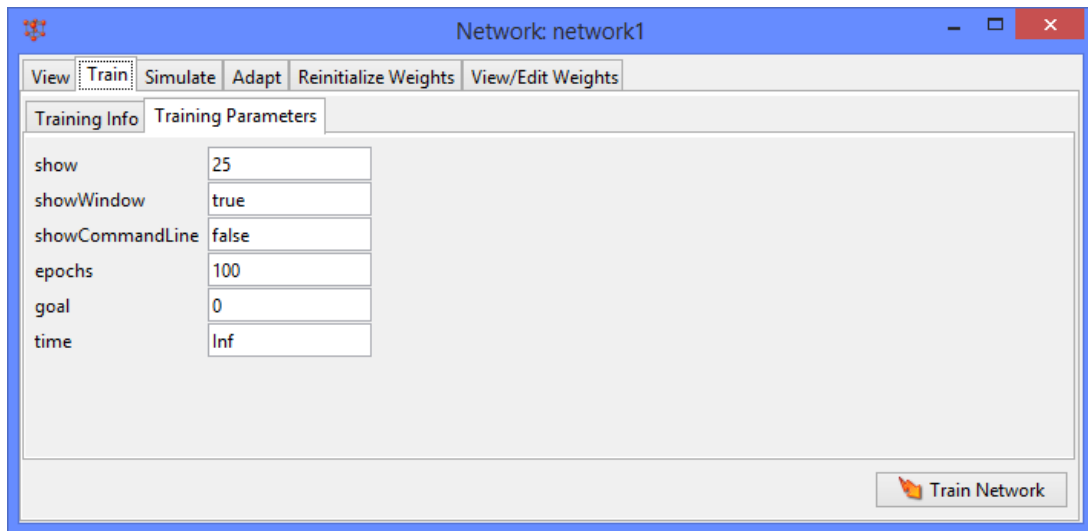


Рис. 1.18. Вкладка параметров обучения

Чтобы начать обучение, нужно нажать кнопку "Обучить сеть" (TrainNetwork). После этого появится окно, иллюстрирующее динамику целевой функции – кривую обучения. В нашем случае график может выглядеть так, как показано на рис. 1.19.

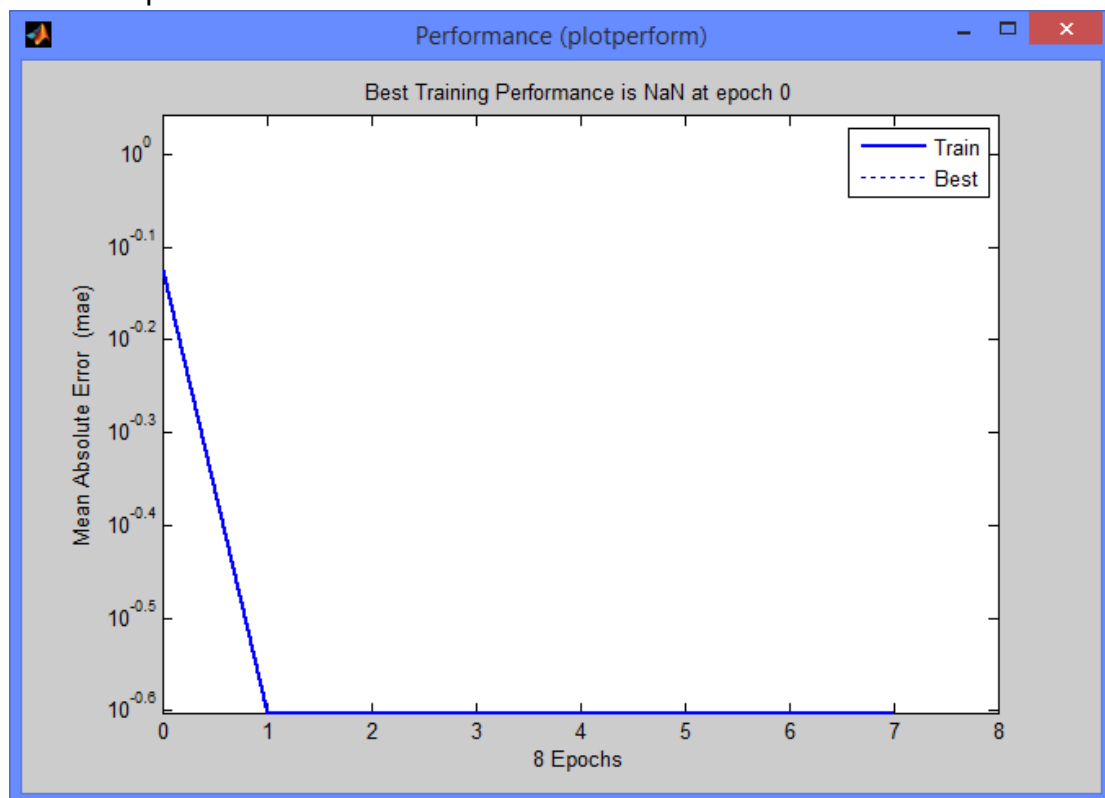


Рис. 1.19. Кривая обучения

Из величины ошибки, близкой к нулю, делаем вывод о том, что нейронная сеть обучилась. Убедимся в правильности решения задачи путём прогона обученной сети. Для этого необходимо открыть вкладку "Прогон" ("Simulate") и выбрать в выпадающем списке "Входы" ("Inputs") данные для Антона. Они будут представлять из себя вектор-столбец из трех значений. После прогона в

выходных данных получим значение "[0]", что означает, что по сети предсказывает Антону отказ от поездки на юг.

### **3. Задание на лабораторную работу**

1. Изучить принципы работы формальных нейронных сетей.
2. При помощи двух (на выбор) нейроимитаторов создать нейронную сеть, обучить ее, и решить на ее основе поставленную практическую задачу. Рекомендуется использовать "NNTool" из пакета "MatLab" и "Deductor Studio".
3. Отобразить результаты обучения нейронной сети и ее прогона.

### **4. Требования к содержанию отчета**

1. Цель лабораторной работы.
2. Индивидуальное задание.
3. Для каждой программы работы с нейронными сетями: подготовленные исходные данные для обучения сети; граф или схема нейронной сети; график обучения; результаты прогона (ответ на поставленный в задании вопрос).
4. Выводы по полученным результатам.

### **5. Контрольные вопросы**

1. Какой метод машинного обучения вами использовался в работе? Почему?
2. Опишите структуру вашей нейронной сети (тип архитектуры, количество слоев, формальных нейронов и так далее).
3. Как узнать, что нейронная сеть обучилась?
4. Каким образом вы подготовили исходные данные для их использования в качестве обучающей выборки для нейронной сети? Какой они имеют формат?
5. Назовите преимущества и недостатки программной и аппаратной реализации нейронной сети.
6. Что такое обучение нейронной сети и что такое прогон сети?

## 6. Индивидуальные задания.

### Варианты заданий на лабораторную работу

#### Вариант 1

Петя	Списал контрольную работу	Не выучил уроки	Гулял до 12	Получил 2
Вася	Сам сделал контрольную работу	Не выучил уроки	Гулял до 12	Получил 2
Витя	Сам сделал контрольную работу	Выучил уроки	Не гулял до 12	Получил 5
Вова	Сам сделал контрольную работу	Не выучил уроки	Не гулял до 12	Получил 5
Антон	Списал контрольную работу	Выучил уроки	Не гулял до 12	Что получил?

#### Вариант 2

Петя	Занимается боксом	Хорошо бегают	Отбился от хулиганов
Вася	Занимается боксом	Плохо бегают	Отбился от хулиганов
Витя	Занимается шахматами	Плохо бегают	Не отбился от хулиганов
Антон	Занимается шахматами	Хорошо бегают	Предсказать результат

#### Вариант 3

Петя	Не учился в институте	Не учился в техникуме	Явился по повестке	Забрали в армию
Вася	Не учился в институте	Не учился в техникуме	Не явился по повестке	Забрали в армию
Витя	Учился в институте	Не учился в техникуме	Явился по повестке	Не забрали в армию
Антон	Не учился в институте	Учился в техникуме	Явился по повестке	Не забрали в армию
Лева	Учился в институте	Учился в техникуме	Не явился по повестке	Где пребывает?

**Вариант 4**

Петя	Вымыл окна	Поругался с начальником	Перепутал документы	Уволен с работы
Вася	Вымыл окна	поладил с начальством	Перепутал документы	Работает
Витя	Разбил окно	поладил с начальством	Разобрал документы	Работает
Вова	Разбил окно	поладил с начальством	Перепутал документы	Работает
Антон	Разбил окно	Поругался с начальником	Перепутал документы	Как дела на работе?

**Вариант 5**

Петя	Умеет читать	Умеет писать	Любит аттракционы	Взяли в космонавты
Вася	Умеет читать	Не умеет писать	Любит аттракционы	Не взяли в космонавты
Витя	Не умеет читать	Не умеет писать	Не любит аттракционы	Не взяли в космонавты
Вова	Не умеет читать	Не умеет писать	Любит аттракционы	Сбудется ли его мечта стать космонавтом?

**Вариант 6**

Выключатель 1	Выключатель 2	Выключатель 3	В подъезде
-	-	-	Темно
-	-	+	Темно
-	+	-	Светло
-	+	+	Светло
+	-	-	Темно
+	-	+	Светло
+	+	-	Светло
+	+	+	?

**Вариант 7**

Вася	Ловил рыбу сетью	утром	Хватило на уху
Петя	Ловил рыбу сетью	вечером	Не хватило на уху
Коля	Ловил рыбу удочкой	вечером	Не хватило на уху
Дима	Ловил рыбу сетью	утром	хватило на уху
Жора	Ловил рыбу удочкой	утром	?

**Вариант 8**

Рома	Завел будильник	Не ходил на дискотеку	Проснулся вовремя
Соня	Завела будильник	ходила на дискотеку	Прospала
Рома	Не завел будильник	ходил на дискотеку	Прospал
Соня	Завела будильник	Не ходила на дискотеку	Проснулась вовремя
Рома	Не завел будильник	Не ходил на дискотеку	?

**Вариант 9**

Женя	Не посещал занятия в автошколе	Не учил билеты	Явился на экзамен	Не сдал экзамен
Коля	Не посещал занятия в автошколе	Не учил билеты	Не явился на экзамен	Не сдал экзамен
Саша	Посещал занятия в автошколе	Не учил билеты	Явился на экзамен	Сдал экзамен
Данила	Не посещал занятия в автошколе	Учил билеты	Явился на экзамен	Сдал экзамен
Степа	Посещал занятия в автошколе	Учил билеты	Не явился на экзамен	?

**Вариант 10**

	Сдал отчёты во время	Хорошо проявил себя	Имеет Хорошие отношения с начальством	Получил повышение
Вася	+	-	+	+
Петя	+	-	-	-
Сергей	-	+	-	-
Алексей	+	-	+	+
Иван	+	+	-	?

**Вариант 11**

Петя	Учил лекции	Делал шпаргалки	Гулял до 12	Сдал экзамен
Вася	Учил лекции	Не делал шпаргалки	Гулял до 12	Сдал экзамен
Витя	Не учил лекции	Не делал шпаргалки	Не гулял до 12	Не сдал экзамен
Вова	Не учил лекции	Делал шпаргалки	Не гулял до 12	Не сдал экзамен
Антон	Учил лекции	Не делал шпаргалки	Не гулял до 12	Сдал ли экзамен?

**Вариант 12**

Температура, С°	Осадки, мм	Продажа мороженого, коробок
33	7,80	76
25	14,0	71
19	18,0	58
31	9,60	73
24	14,2	62
19	18,6	61
32	8,20	82
30	10,0	69
26	12,6	69
31	9,60	68
31	10,0	70
18	19,0	52

**Вариант 13**

Зависимость электрического сопротивления  $S$  некоторого медного стержня от температуры  $t$ :

$t$	19,1	25,0	30,1	36,0	40,0	45,2	50
$S$	76,30	77,80	79,75	80,80	82,35	83,90	85

**Вариант 14**

Вероника	Любит историю	Не любит математику	Не любит физику	Любит литературу	Гуманитарий
Алексей	Не любит историю	Любит математику	Любит физику	Не любит литературу	Технарь
Ирина	Любит историю	Любит математику	Не любит физику	Любит литературу	Гуманитарий
Иван	Не любит историю	Не любит математику	Любит физику	Не любит литературу	Технарь
Петр	Любит историю	Любит математику	Любит физику	Не любит литературу	Технарь
Женя	Не любит историю	Не любит математику	Не любит физику	Любит литературу	?

**Вариант 15**

Вячеслав	Хорошо питается	Занимается в спортзале	Работает в офисе	Нормальный вес
Алексей	Плохо питается	Лежит на диване	Работает грузчиком	Ненормальный вес
Иван	Хорошо питается	Лежит на диване	Работает грузчиком	Нормальный вес
Иннокентий	Плохо питается	Лежит на диване	Работает в офисе	Ненормальный вес
Илья	Хорошо питается	Занимается в спортзале	Работает грузчиком	?

**Вариант 16**

Иван	Нагрешил	Исповедался	Умер	Не мучается
Степан	Нагрешил	Не исповедался	Умер	Горит в аду
Руслан	Нагрешил	Исповедался	Живой	Не мучается
Денис	Нагрешил	Не исповедался	Живой	Не мучается
младенец Петя	Не грешил	Не исповедался	Умер	Не мучается
младенец Оля	Не грешила	Не исповедалась	Живая	?

**Вариант 17**

Ксения	Не чихает	Не кашляет	Улыбается	Здорова
Вика	Не чихает	Кашляет	Сопли и слёзы	Больна
Рита	Чихает	Не кашляет	Улыбается	Здорова
Танечка	Не чихает	Не кашляет	Сопли и слёзы	Здорова
Вениамин	Чихает	Кашляет	Сопли и слёзы	Болен
Фаина	Чихает	Не кашляет	Сопли и слёзы	?

**Вариант 18**

	Талант	Выносливость	Сила воли	Станет лыжником?
Ваня	Есть	Хорошая	Есть	Да
Петя	Есть	Плохая	Есть	Да
Саша	Нет	Хорошая	Нет	Нет
Егор	Нет	Плохая	Есть	Нет
Вася	Есть	Хорошая	Нет	?

**Вариант 19**

	Учился хорошо	Трудолюбие	Активность	Работает по специальности
Игорь	+	+	-	+
Саша	+	-	+	-
Настя	+	+	+	+
Паша	-	+	+	-
Гена	+	-	-	?

**Вариант 20**

	Знание законов	Виноват	Репутация	Защитил свои права?
Гриша	+	-	+	+
Ваня	-	+	+	-
Боря	-	-	-	-
Маша	+	-	-	+
Вова	+	+	-	-
Вера	-	-	+	?



## **Лабораторная работа №2**

### **«ПОДГОТОВКА ДАННЫХ ДЛЯ МАШИННОГО ОБУЧЕНИЯ И ОЦЕНКА КАЧЕСТВА ОБУЧЕННОГО КЛАССИФИКАТОРА»**

#### **1. Цель работы**

Изучить методы анализа и трансформации данных, применяемых в машинном обучении.

#### **2. Методические указания**

##### **2.1. Введение**

Для того чтобы эффективно применять методы анализа данных, часто требуется выполнить предобработку данных. Данные могут содержать пропуски, шумы, аномальные значения и т.д. Кроме того, данные могут быть избыточны. Помимо этого размерность исходного пространства данных может быть очень большой, и тогда желательно применение специальных алгоритмов понижения размерности: отсеивание наименее значимых признаков.

##### **2.2. Понижение размерности данных**

Понизить размерность данных можно при помощи корреляционного и факторного анализа данных.

Корреляционный анализ – метод обработки статистических данных, заключающийся в изучении коэффициентов (корреляции) между переменными. Корреляция – статистическая взаимосвязь двух или нескольких случайных величин (либо величин, которые можно с некоторой допустимой степенью точности считать таковыми). Математической мерой корреляции двух случайных величин служит коэффициент корреляции. Модуль коэффициента свидетельствует о степени зависимости: чем ближе его значение к 0, тем слабее линейная зависимость. Чем ближе коэффициент корреляции к 1, тем сильнее прямая линейная зависимость, чем ближе к -1, тем сильнее обратная линейная зависимость. На практике считается, что если модуль коэффициента корреляции больше 0,6, то линейная зависимость сильная, а если менее 0,3, то почти отсутствует. Если корреляционный анализ покажет слишком слабую корреляцию между входным признаком и выходными данными, то данный признак можно исключить из рассмотрения, таким образом, понизив размерность данных.

Факторный анализ – это методика комплексного и системного изучения и измерения воздействия факторов на величину результативного показателя. Одной из главных целей факторного анализа является сокращение числа переменных (редукция данных). Поэтому факторный анализ используется как метод сокращения данных.

### 2.3. Проверка репрезентативности выборки и анализ результатов

В машинном обучении применяют следующие виды выборок:

2.3.1. Обучающая выборка (training sample) – выборка, по которой производится настройка (оптимизация параметров) модели зависимости. Фактически на этих данных обучается алгоритм.

Если модель зависимости построена по обучающей выборке, то оценка качества этой модели, сделанная по той же выборке, оказывается, как правило, оптимистически смещённой. Это нежелательное явление называют переобучением. На практике оно встречается очень часто. Хорошую эмпирическую оценку качества построенной модели даёт её проверка на независимых данных, которые не использовались для обучения.

2.3.2. Тестовая (или контрольная) выборка (test sample) – выборка, по которой оценивается качество построенной модели. Если обучающая и тестовая выборки независимы, то оценка, сделанная по тестовой выборке, является несмещённой.

Оценку качества, сделанную по тестовой выборке, можно применить для выбора наилучшей модели. Однако тогда она снова окажется оптимистически смещённой. Для получения несмещённой оценки выбранной модели приходится выделять третью выборку.

2.3.3. Проверочная выборка (validation sample) – выборка, по которой осуществляется выбор наилучшей модели из множества моделей, построенных по обучающей выборке.

Чаще всего в машинном обучении задействуют обучающую и тестовую выборки.

### 2.4. Типовой пример выполнения работы

Предположим, мы хотим решить задачу предсказания настроения человека по произошедшим с ним за текущий день событиям (рис. 2.1).

	Имя	Часов спал	Находится в отпуске	Гулял с собакой	Провел часов в пробках	Поругался с женой	Получил премии	Настроение
▶	Алексей	10	1	1	0	0	10	1
	Дмитрий	3	1	0	2	0	5	0
	Виктор	8	1	0	0	0	15	1
	Иван	9	1	0	0	0	15	1
	Игнат	10	0	0	0	0	0	0
	Федот	6	0	1	5	1	10	0
	Кирилл	9	0	0	0	0	0	0
	Стас	5	0	0	0	0	10	1
	Никита	8	1	0	0	0	5	1
	Андрей	7	1	0	1	0	15	1

Рис. 2.1. Данные задачи

Исходные данные представим в виде CSV-файла и импортируем их в "Deductor Studio" (рис. 2.2).

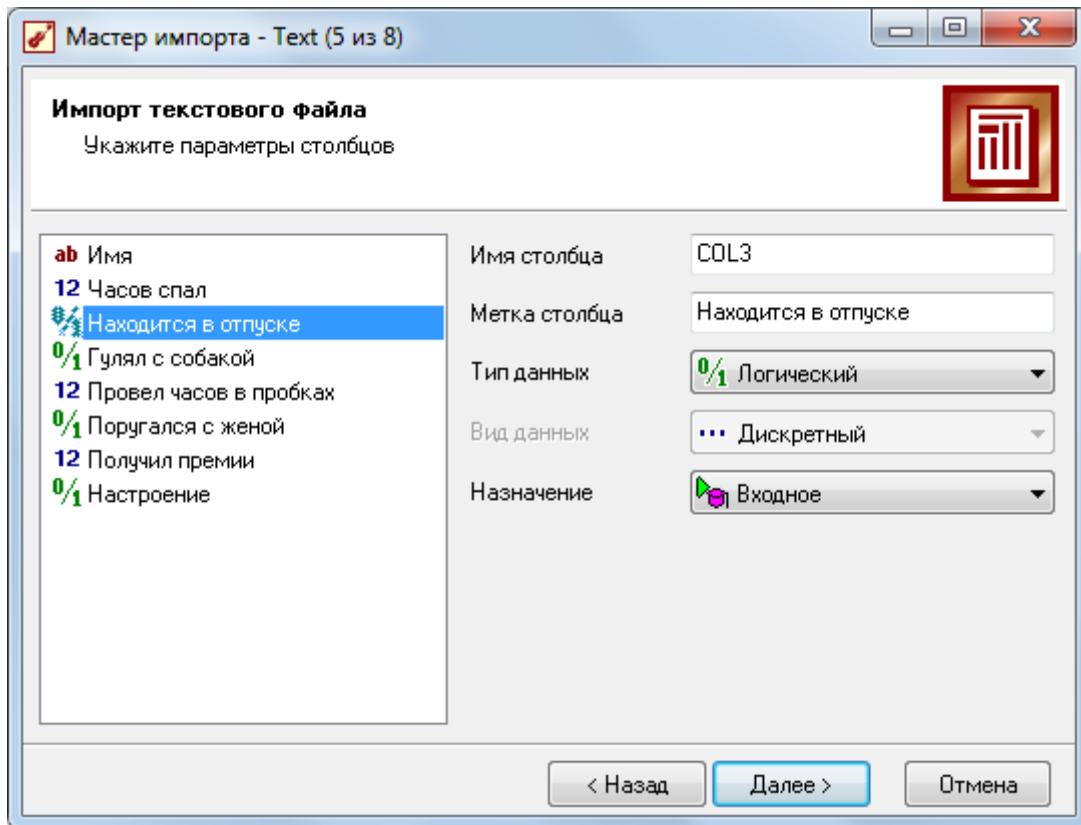


Рис. 2.2. Импорт данных

Отметим, что поля “Часов спал”, “Находился в отпуске”, “Гулял с собакой”, “Провел часов в пробках”, “Поругался с женой”, “Получил премию” являются входными, а поле “Настроение” – выходным (целевое значение для обучения).

Теперь к импортированным данным можно применить инструмент мастера обработки “Выявление дубликатов и противоречий” (Рис. 2.3).

Отметим, что **дубликатом** называется набор прецедентов, имеющих одинаковые значения входных и целевых полей. **Противоречием** называется набор прецедентов, имеющих одинаковые значения входных полей, но различающиеся значениями целевых полей.

В результате обработки данных в таблице появляются новые столбцы: “Дубликаты” и “Противоречия”, имеющие не пустые значения для строк, относящимся соответственно к дубликатам и противоречиям.

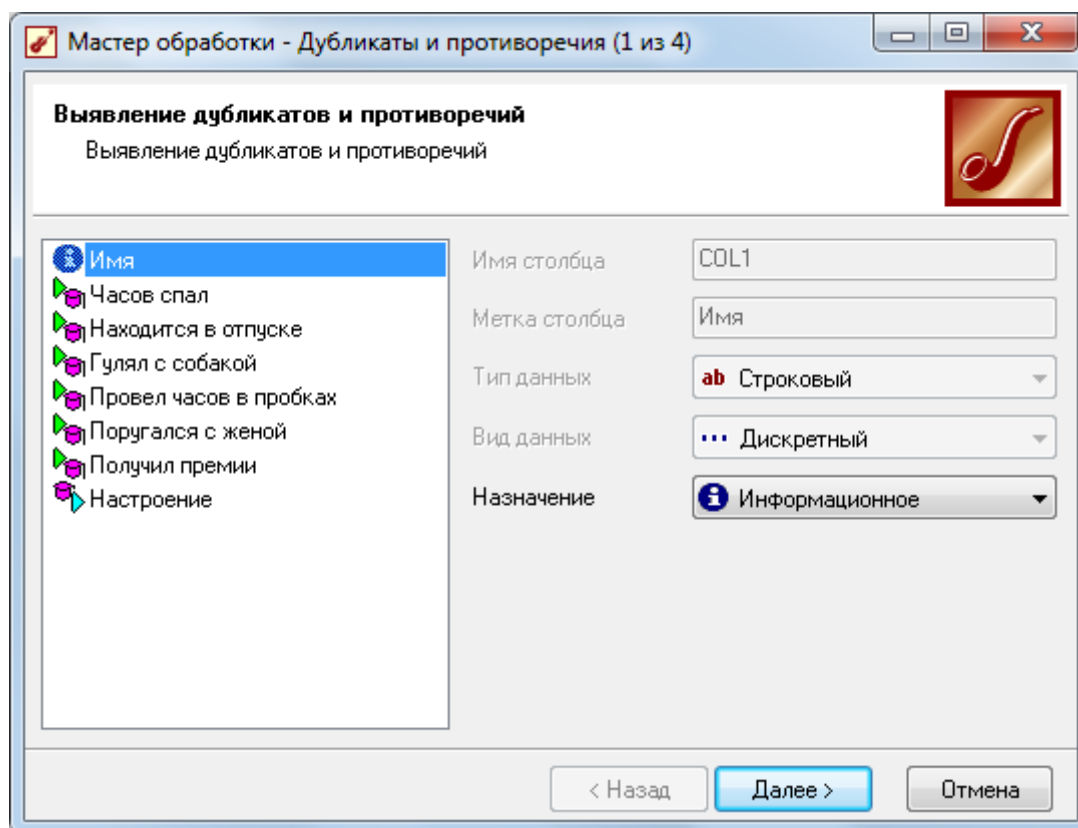


Рис. 2.3. Поиск дубликатов и противоречий

Чтобы убрать дубликаты и противоречия рекомендуется применить инструмент “Группировка данных” (рис. 2.4).

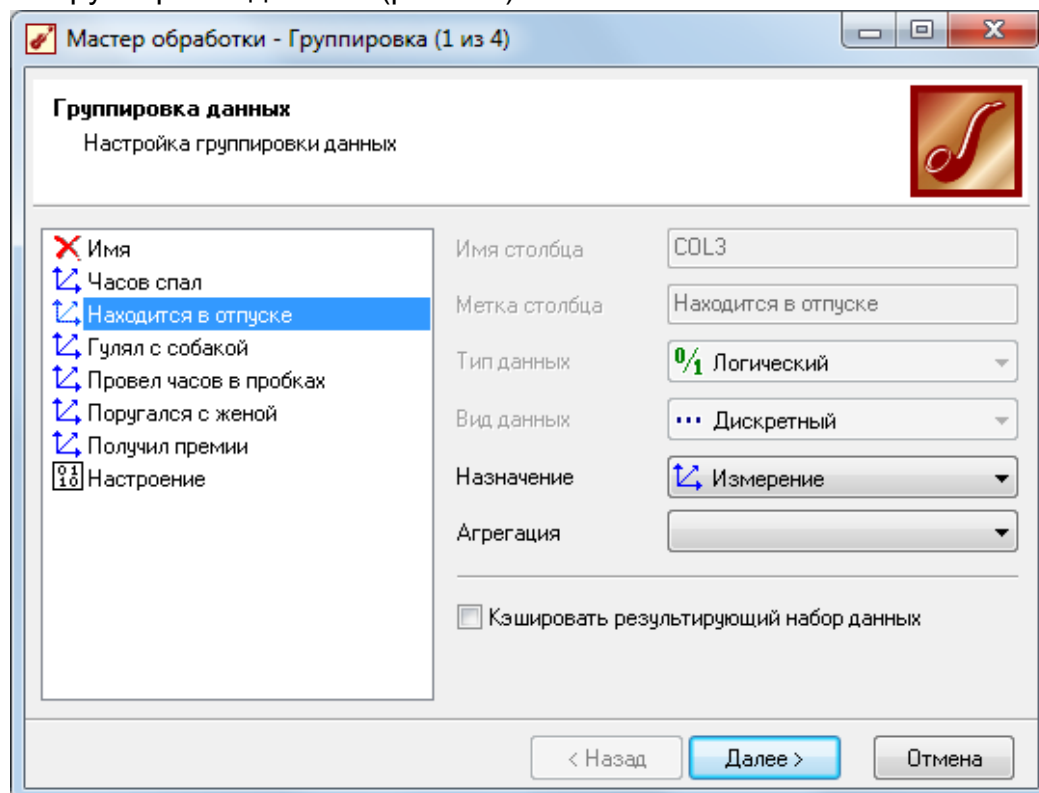


Рис. 2.4. Устранение дубликатов и противоречий при помощи группировки данных

Далее необходимо произвести корреляционный анализ. В настройках обработчика указываются входные и выходные поля (рис. 2.5). Затем выбираем пункт Коэффициент корреляции Пирсона (рис. 2.6).

**Мастер обработки - Корреляционный анализ (1 из 6)**

**Корреляционный анализ**  
Задание входных, выходных и информационных столбцов

Часов спал  
Находится в отпуске  
Гулял с собакой  
**Провел часов в пробках**  
Поругался с женой  
Получил премии  
Настроение

Имя столбца: COL5  
Тип данных: Целый  
Назначение: Входное  
Вид данных: Непрерывный

Статистика

Минимум	0
Максимум	5
Среднее	0,631578947368421
Стандартное откл.	1,30002249193318

< Назад    Далее >    Отмена

Рис. 2.5. Настройка данных для корреляционного анализа

**Мастер обработки - Корреляционный анализ (2 из 6)**

**Параметры расчета корреляции**  
Укажите метод расчета корреляции

☒ **Коэффициент корреляции Пирсона**  
При помощи коэффициента корреляции Пирсона можно определить силу и направление линейной зависимости между двумя процессами происходящих одновременно и без учета временного лага.

☐ **Максимум взаимнокорреляционной функции**  
Будет вычислен максимум из коэффициентов корреляций двух процессов, рассчитанных при всевозможных временных сдвигах. Следует применять, если необходимо узнать линейную зависимость между двумя процессами или частями процессов происходящих с определенным временным лагом.

< Назад    Далее >    Отмена

Рис. 2.6. Выбор параметра расчета корреляции

В результате получается таблица с коэффициентами корреляции для каждой возможной пары из комбинации входного и выходного параметров (рис. 2.7). Коэффициент корреляции принимает значения от -1 до 1.

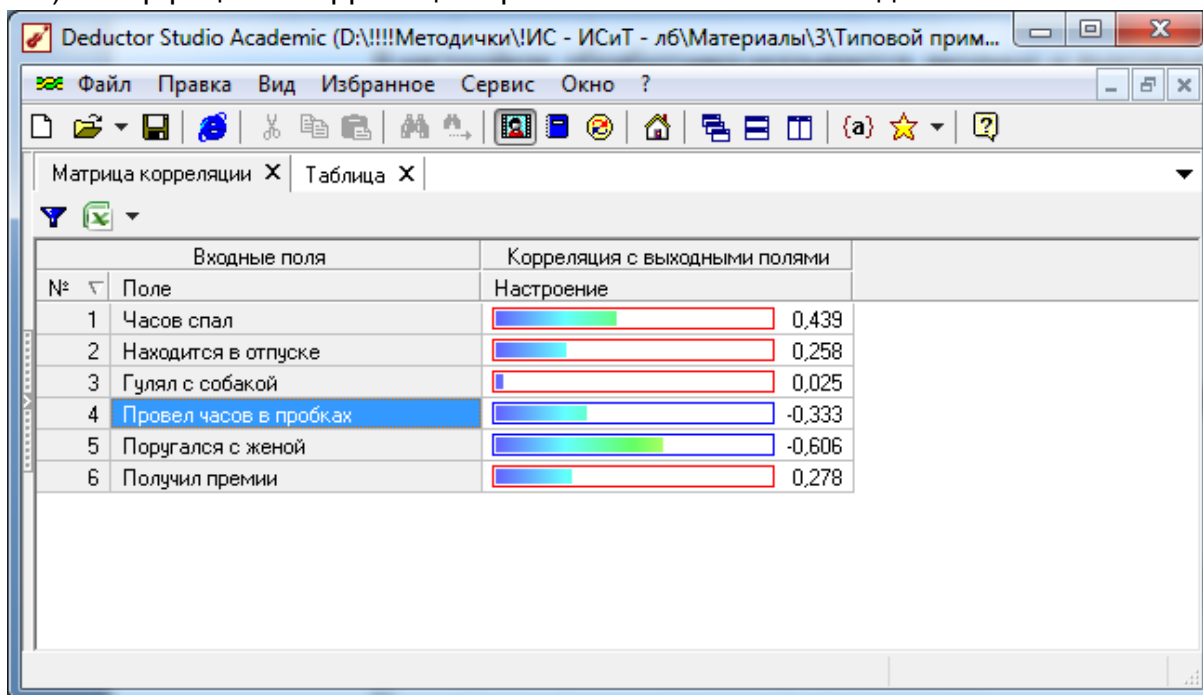


Рис. 2.7. Результат корреляционного анализа

Анализируя полученные результаты, принимаем решение об исключении признака "Гулял с собакой", как практически не влияющего на результат (значение модуля коэффициента корреляции слишком мало и составляет 0,025). Размерность данных снижается с 6 до 5.

Следующим этапом необходимо произвести факторный анализ. Нередко в наборе данных объекты описываются огромным числом признаков. Однако, здесь может возникнуть проблема наличия слабо информативных и неинформативных признаков. Анализировать большое число признаков на наличие указанных проблем крайне затруднительно. В данном случае уместнее избавиться от некоторого их количества, применив факторный анализ. В результате будет получено новое пространство признаков меньшей размерности.

В настройках обработчика "Факторный анализ" указываются входные и выходные поля. При уточнении числа выделяемых факторов отбрасываем один фактор, жертвуя 8,17% полезности исходных данных в формировании результата, снижая размерность данных с 5 до 4 (рис. 2.8).

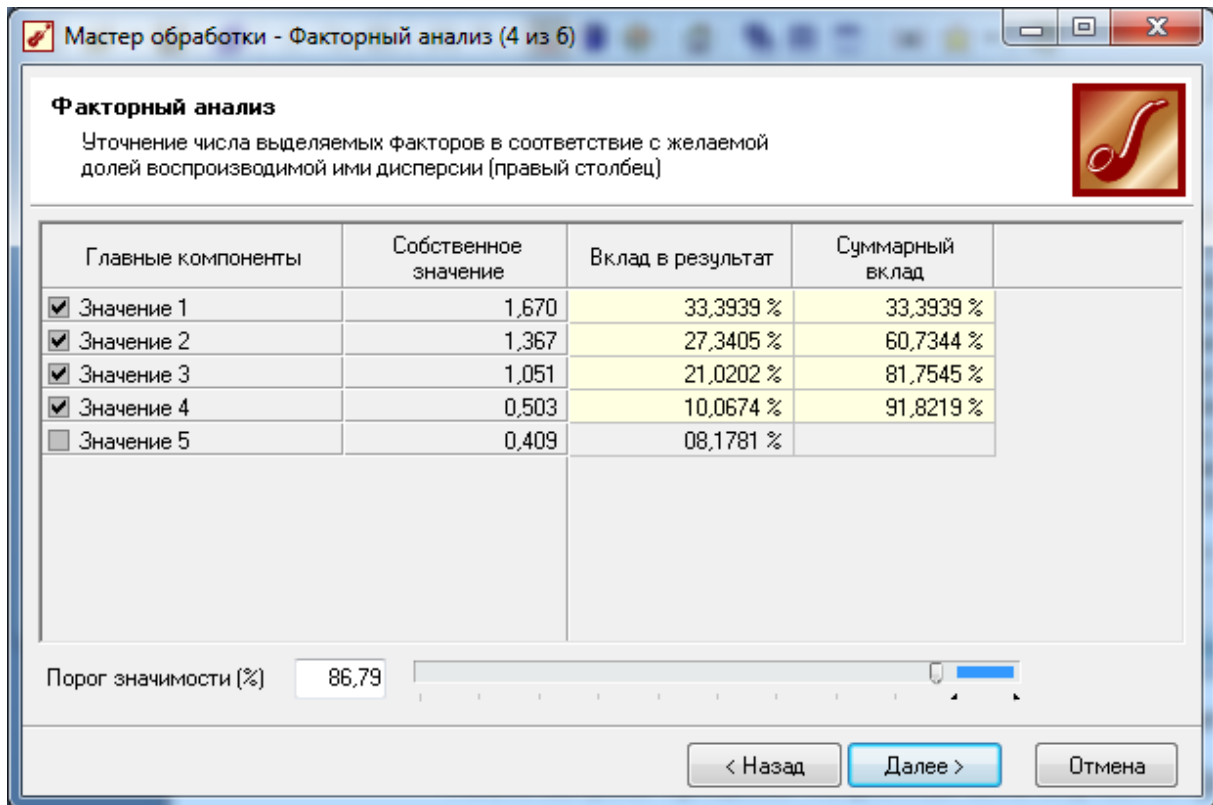


Рис. 2.8. Настройка данных для факторного анализа

Подготовку данных для машинного обучения теперь можно считать завершенной. Заметим, что размерность данных при проведении подготовительных этапов снизилась на треть. Такое сокращение стало бы очень существенным при работе с огромными данными.

Запустим мастер обработки "Нейросеть" (рис. 2.9).

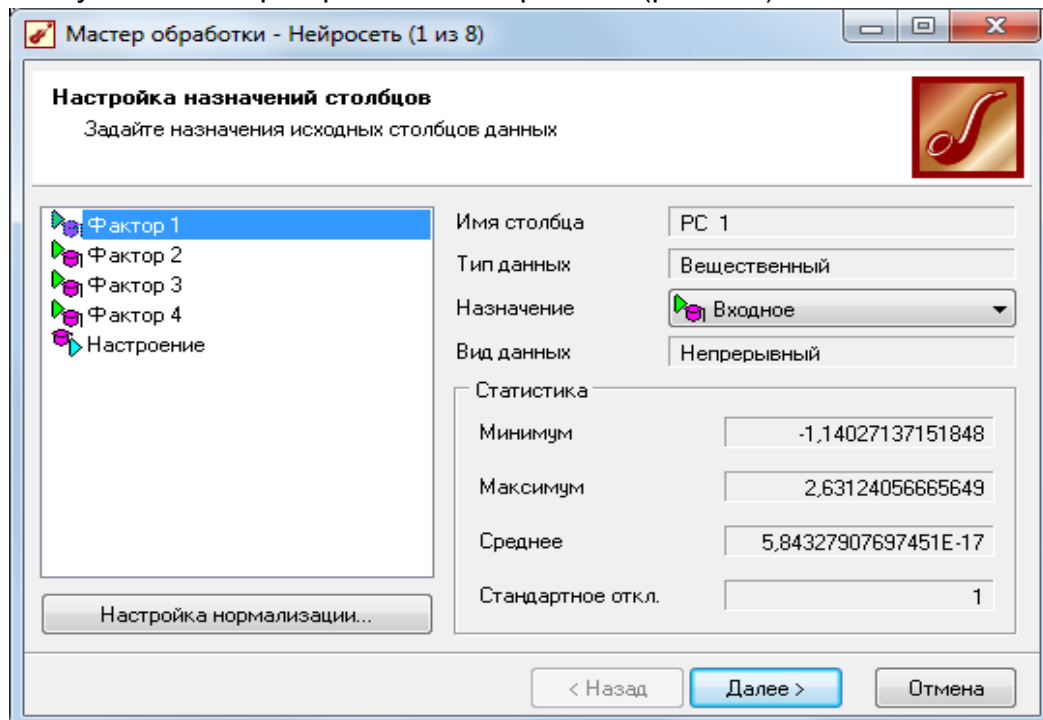


Рис. 2.9. Мастер обработки "Нейросеть"

Выборку данных необходимо разделить пополам на обучающую и тестовую. Далее мы обучим нейронную сеть (рис. 2.10).

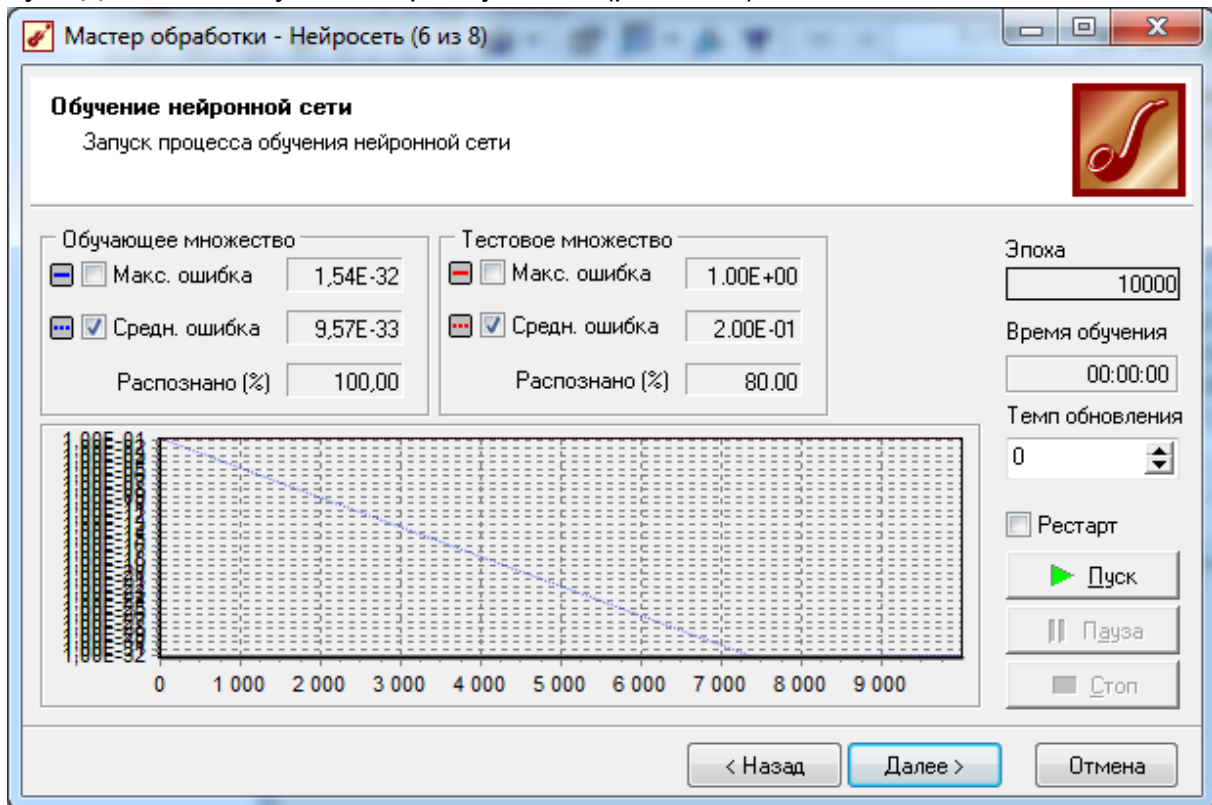


Рис. 2.10. Запуск процесса обучения нейронной сети

Анализируя полученные результаты, по существенным различиям показателей качества на обучающем и тестовом множествах, делаем вывод о не репрезентативности обучающей выборки. Поэтому следует увеличить количество прецедентов исходных данных и повторить обучение с анализом выборки на репрезентативность.

### 3. Задание на лабораторную работу

#### 3.1. Порядок выполнения работы

1. Создать выборку для обучения нейронной сети.
2. Убрать из выборки дубликаты и противоречия.
3. Провести корреляционный анализ данных. Определить корреляцию между всеми входными данными и целью. При наличии очень малой корреляции между некоторым входным параметром и целью можно исключить этот параметр.
4. Провести факторный анализ входных параметров. Определить уровень значимости каждого фактора. При очень низком уровне значимости исключить фактор.
5. Выборку данных разделить пополам на обучающую и тестовую. Обучить нейронную сеть.
6. Сравнить величину ошибки относительно данных тестовой и обучающей выборки. Если они сильно различаются, то сеть является переобученной, а



обучающая выборка мала (не репрезентативна) и ее следует дополнить и вернуться к пункту 2.

### **3.2. Требования к работе**

1. Обучающая и тестовая выборка на начальном этапе работы должна содержать не менее 20 прецедентов каждая.
2. Использовать не менее 5 признаков.

### **4. Требования к содержанию отчета**

1. Тема и цель работы.
2. Таблица с исходными данными.
3. Результат устранения дубликатов и противоречий.
3. Результат корреляционного анализа данных.
4. Результат факторного анализа данных обучающей выборки.
5. Результат проверки обучающей выборки на репрезентативность.

### **5. Контрольные вопросы**

1. Что такое обучающая выборка?
2. Для чего нужна тестовая выборка?
3. В каком случае следует применять проверочную выборку?
4. Перечислите, методы анализа данных, которые использовались в работе.
5. Какой метод(ы) нужно применить к данным, чтобы исключить дубликаты и противоречия?
6. Что такое корреляция? Какие значения может принимать коэффициент корреляции?
7. С какой целью выполняется корреляционный анализ данных?
8. С какой целью выполняется факторный анализ данных?
9. Каким образом оценивается репрезентативность выборки данных для машинного обучения?

### **6. Индивидуальные задания**

1. Предсказать температуру на улице на следующий день по параметрам текущей погоды.
2. Классифицировать автомобили.
3. Оценить платежеспособность должника.
4. Предсказать успешность.

## Лабораторная работа №3

### «ПРОГРАММНАЯ РЕАЛИЗАЦИЯ РЕЗУЛЬТАТОВ МАШИННОГО ОБУЧЕНИЯ»

#### 1. Цель работы

Изучить особенности применения нейросетевых систем для решения задач экспертного выбора; научиться формализовать результаты обучения нейронной сети и внедрять их в программный код.

#### 2. Методические указания

Существуют два подхода к реализации экспертных систем:

1. Rule-based. При этом подходе используется модель представления знаний в виде набора правил. Правила задаются на этапе разработки экспертной системы, являются строгими и неизменными. Пример правила: “если у человека болят зубы, ему надо обратиться к зубному врачу”.

2. Case-based. Это адаптивный подход, основанный на обучении экспертной системы на конкретных примерах.

Нейронные системы позволяют реализовать второй подход. Они используют машинное обучение и обучающую выборку данных, которая содержит любое количество образцовых примеров работы экспертной системы.

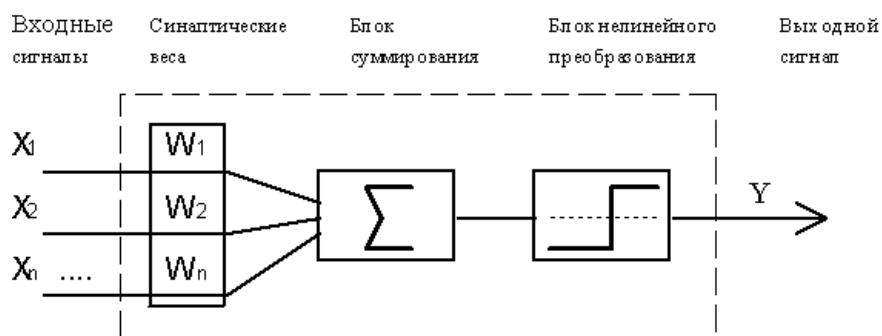


Рис. 3.1. Структура формального нейрона

#### 2.1. Типовой пример решения задачи экспертного выбора

Реализацию экспертного выбора нейронной сетью рассмотрим на примере следующей простой задачи: по двум характеристикам транспортного средства (наличие двух колес и мотора) необходимо определить его тип из следующих возможных: автомобиль, мотоцикл, велосипед. Исходные данные, подготовленные для машинного обучения, могут быть представлены следующей таблицей:

Таблица 3.1

**Исходные данные для решения задачи**

Имеет два колеса	Имеет мотор	Автомобиль	Мотоцикл	Велосипед
нет	да	да	нет	нет
да	да	нет	да	нет
да	нет	нет	нет	да

Отметим, что количество входных полей соответствует числу характеристик классов объектов, а количество выходных – числу классов объектов. Подготовим исходные данные для работы в "NNTool": Входные данные (input data) имеют структуру в виде матрицы размером  $3 \times 2$  (рис. 3.2).

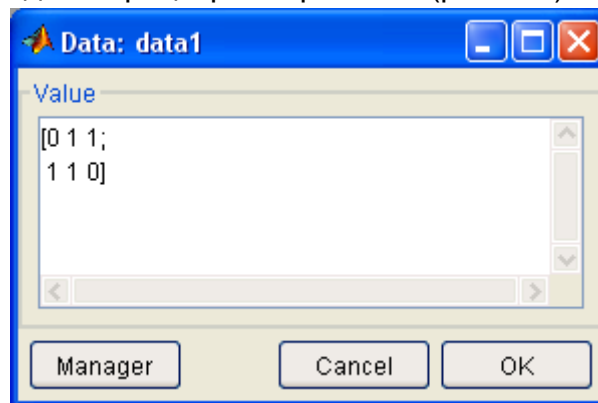


Рис. 3.2. Входные данные для задачи

Целевые данные (target data) также имеют размерность 2 и представлены матрицей  $3 \times 3$  (рис. 3.3).

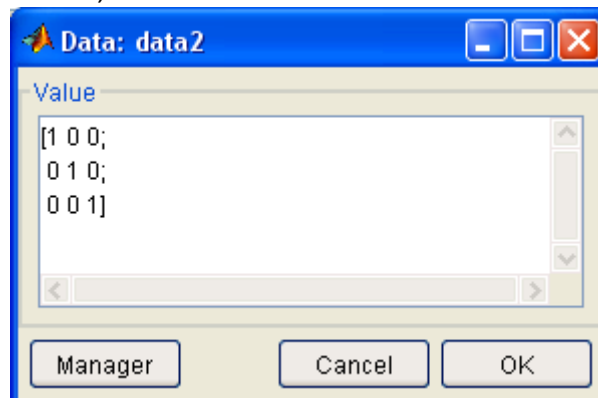


Рис. 3.3. Ввод целевых данных для задачи

Знакомым нам способом обучаем нейронную сеть, имеющую 2 входа и 3 выхода (рис. 3.4).

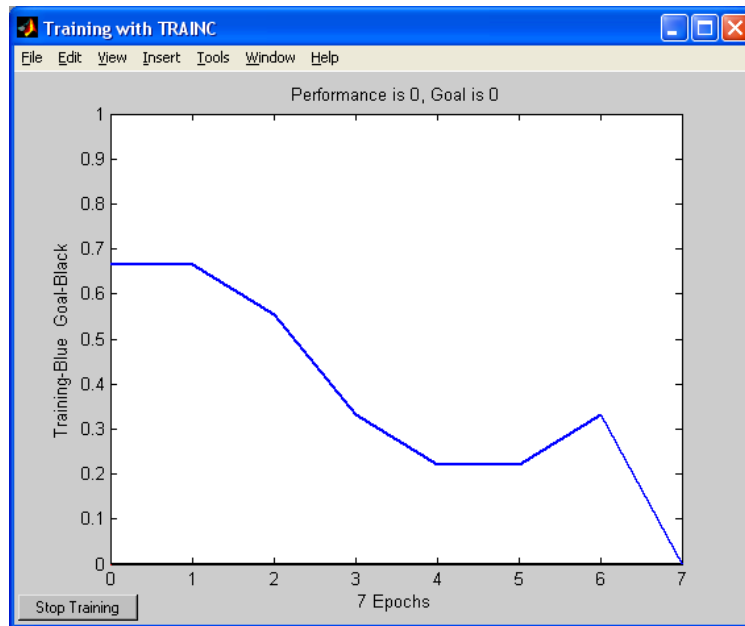


Рис. 3.4. Обучение нейронной сети

Мы знаем, что при обучении нейронной сети происходит корректировка синаптических весов и порогов. Их подобранные значения нам нужны для реализации алгоритма работы обученной сети [2]. Но для начала обратим внимание на архитектуру нейронной сети, являющейся однослойной и состоящей из 3 нейронов (рис. 3.5). Каждый нейрон имеет два входа и один выход, соответственно, его состояние описывается двумя значениями синаптических весов и одной пороговой величиной.

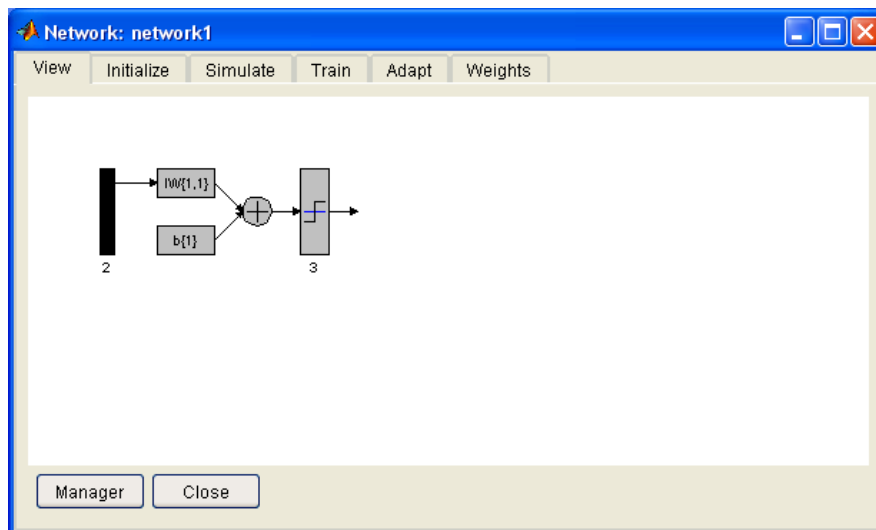


Рис. 3.5. Структура нейронной сети

Таким образом, учитывая архитектуру сети, веса имеют структуру матрицы размером  $2 \times 3$  (рис. 3.6), а пороги вектор-столбец трех элементов (рис. 3.7).

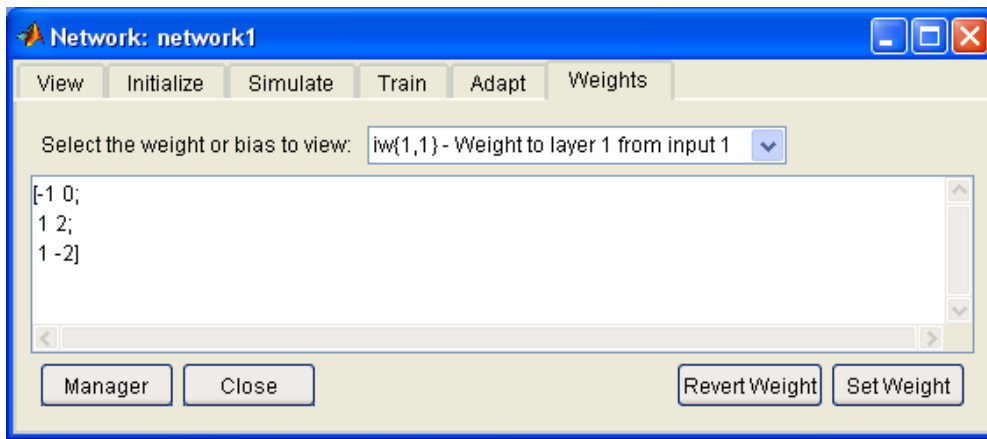


Рис. 3.6. Веса

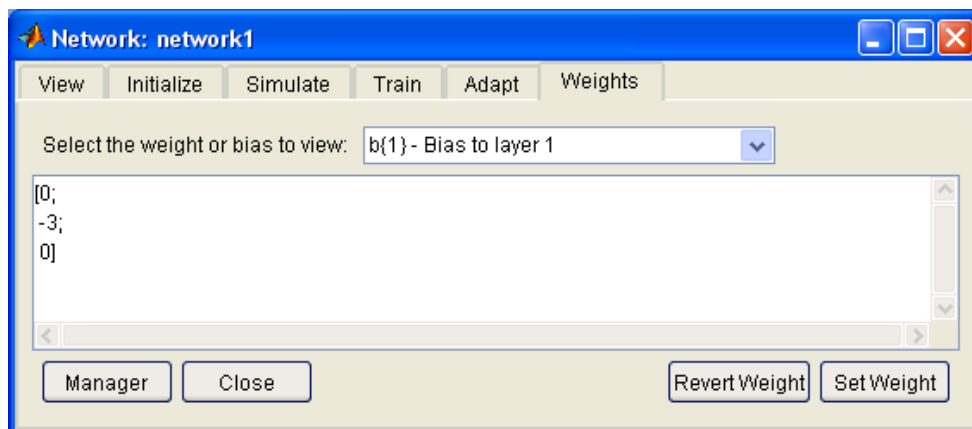


Рис. 3.7. Пороги

Получив веса, мы можем математическим расчетом проверить работу нейронной сети на примерах из выборки. В выборке для объекта с четырьмя колесами и мотором расчет выглядит следующим образом:

1. Похож ли объект на автомобиль?

$0*(-1)+1*0+0 = 0$ ; 0 не отрицательное число, следовательно, ответ ДА

2. Похож ли объект на мотоцикл?

$0*1+1*2-3 = -1$ ; -1 отрицательное число, следовательно, ответ НЕТ

3. Похож ли объект на велосипед?

$0*1+1*(-2)+0 = -1$ ; отрицательное число, следовательно, ответ НЕТ

Таким образом, по мнению нейронной сети и, реализованной на ее основе экспертной системы (рис. 3.8), объект является автомобилем.

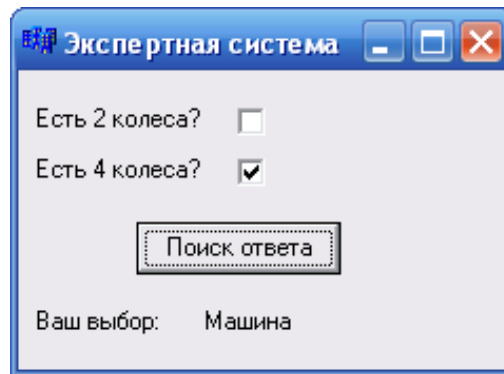


Рис. 3.8. Работа экспертной системы

Листинг программы на языке С++ выглядит следующим образом:

```
int a=0,b=0,i,y=0,x=0,promej_result=0;
int wesa[3][2]={{-1,0},{1,2},{1,-2}};
int porogi[3]={0,-3,0};
int res[3];
if (CheckBox1->Checked)
a=1;
else
a=0;
if(CheckBox2->Checked)
b=1;
else b=0;
for(i=0;i<3;i++)
{
y=0;
promej_result=0;
promej_result=a*wesa[i][y]+b*wesa[i][y+1]+porogi[i];
if(promej_result<0)
res[x]=0;
else
res[x]=1;
x++;
y++;
}
if(res[0]==1)
Label4->Caption = "Машина";
if(res[1]==1)
Label4->Caption = "Мотоцикл";
if(res[2]==1)
Label4->Caption = "Велосипед";
if(a==1 && b==1)
Label4->Caption = "Не существует такого транспорта";
```

## 2.2. Типовой пример решения задачи экспертного выбора на основе модели линейной регрессии

Данный типовый пример является демонстрацией решения задачи классификации на основе тех же данных (таблица 3.1), но с применением другой

модели машинного обучения: линейной регрессии. Итак, подготовим данные в CSV формате и загрузим им в новых проект в системе Deductor Studio Academic (рис. 3.9).

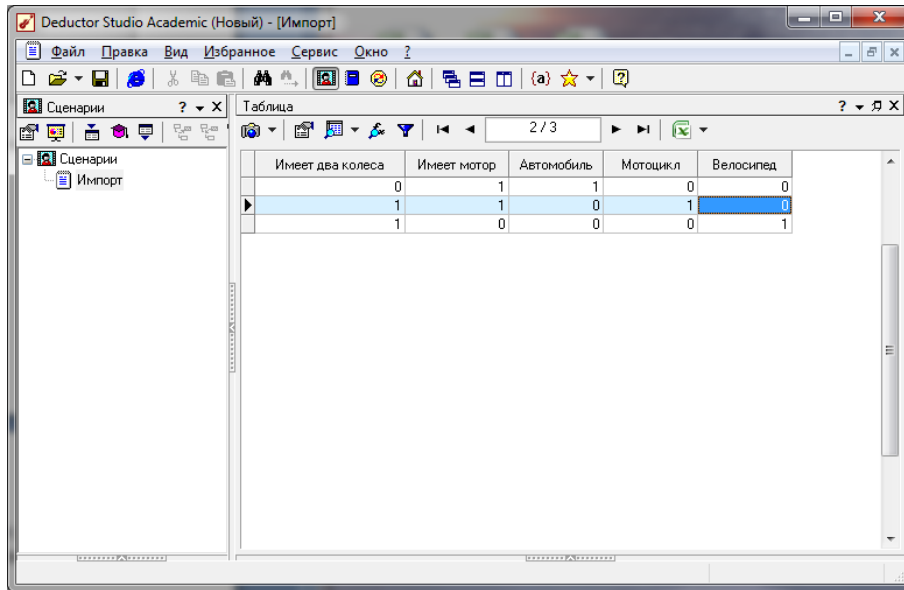


Рис. 3.9. Загрузка данных в Deductor Studio

В данной задаче имеются три выхода, значения которых нужно предсказать: поля "Автомобиль", "Мотоцикл" и "Велосипед". Последовательно построим модели для прогнозирования каждого из этих выходов. Запустим мастер обработки "Линейная регрессия" и укажем типы столбцов, разделив их на входные, выходные и не используемые данные (рис. 3.10), чтобы получить модель прогноза значения поля "Автомобиль".

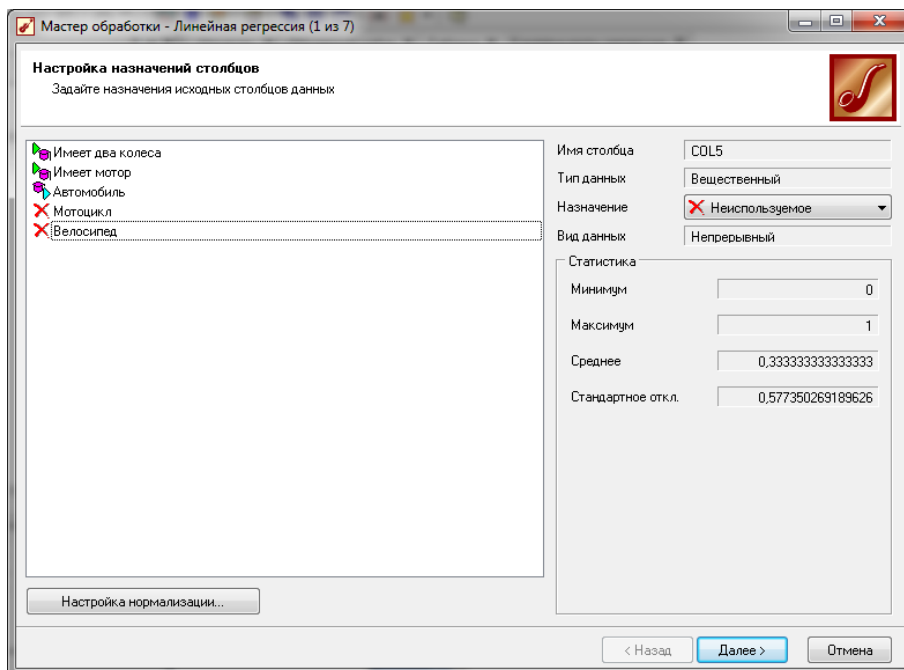


Рис. 3.10. Настройка назначений полей данных

Настройки модели регрессии можно не менять и оставить по умолчанию. После обучения модели укажем на необходимость отображения коэффициентов регрессии (рис. 3.11).

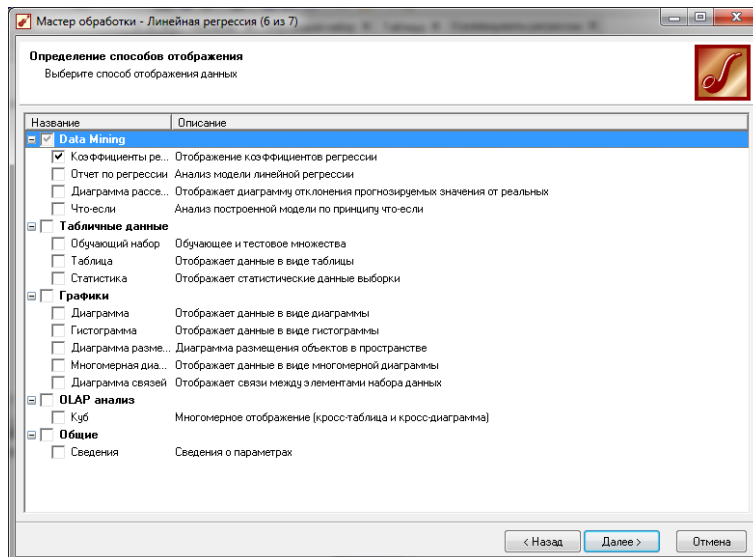


Рис. 3.11. Выбор способов отображения результатов обучения

После работы мастера получим коэффициенты регрессии (рис. 3.12).

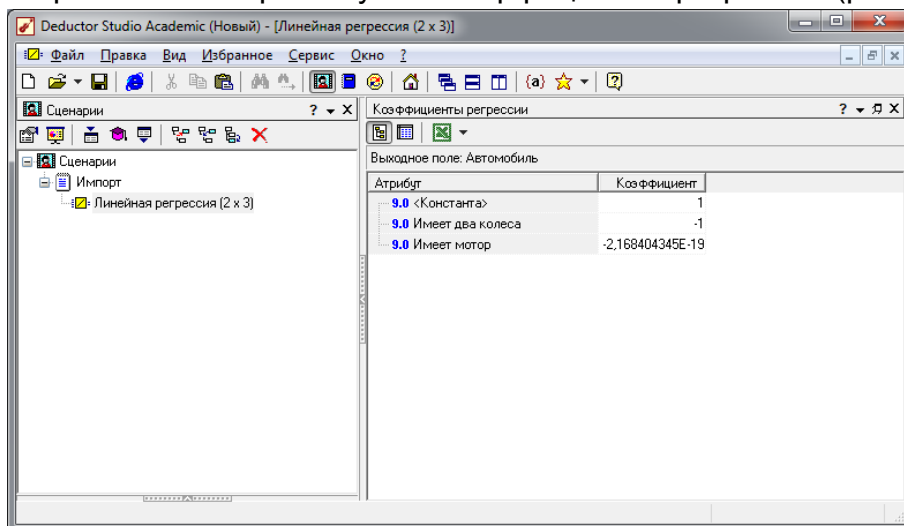


Рис. 3.12. Коэффициенты регрессии для поля "Автомобиль"

Аналогичным образом получим коэффициенты регрессии для предсказания значений полей "Мотоцикл" и "Велосипед" (рис. 3.13).

Выходное поле: Мотоцикл		Выходное поле: Велосипед	
Атрибут	Коэффициент	Атрибут	Коэффициент
9.0 <Константа>	-1	9.0 <Константа>	1
9.0 Имеет два колеса	1	9.0 Имеет два колеса	-2,168404345E-19
9.0 Имеет мотор	1	9.0 Имеет мотор	-1

Рис. 3.13. Коэффициенты регрессии для полей "Мотоцикл" (слева) и "Велосипед" (справа)

Используя коэффициенты регрессии, получим следующую математическую модель:

$$\begin{aligned}
 Y_1 &= -X_1 + 1; \\
 Y_2 &= X_1 + X_2 - 1; \\
 Y_3 &= -X_2 + 1;
 \end{aligned}$$



Здесь  $Y_1$  – предсказанное моделью значение выходного поля "Автомобиль";  $Y_2$  – предсказанное моделью значение выходного поля "Мотоцикл";  $Y_3$  – предсказанное моделью значение выходного поля "Велосипед";  $X_1$  – характеристика объекта в форме ответа на вопрос "Имеет два колеса". Значения всех перечисленных условных обозначений интерпретируются как "да", при большей близости к 1, и "нет" в противном случае.

Реализовать данную модель на языке программирования не составит никакого труда.

### **3. Задание на лабораторную работу**

1. Изучить особенности проектирования моделей машинного обучения для экспертного выбора объектов.
2. Решить задачу экспертного выбора.
  - 2.1. Определить классы объектов
  - 2.2. Выделить признаки классифицируемых объектов.
  - 2.3. Сконструировать и обучить любые две модели машинного обучения (например, нейронная сети и линейная регрессия).
  - 2.4. Программно реализовать экспертную систему на базе алгоритмов функционирования обученных классификаторов.
3. Сделать выводы.

### **4. Требования к содержанию отчета**

1. Цель лабораторной работы.
2. Индивидуальное задание.
3. Таблица данных обучающей выборки.
4. Структура нейронной сети.
5. Свободные параметры обученной модели (веса и смещения).
6. Листинг программы, моделирующей работу обученных классификаторов.
7. Выводы по лабораторной работе.

### **5. Контрольные вопросы**

1. Case-based подход к разработке экспертных систем.
2. Какие можно использовать модели машинного обучения для решения задачи экспертного выбора объектов?
3. Напишите общую математическую модель линейной регрессии.
4. Нарисуйте общую структурную схему формального нейрона. Опишите в общем виде каким образом формируется значение выхода формального нейрона.
5. Какие параметры моделей изменяются при обучении ваших классификаторов.

6. Сравните между собой ваши модели. Какая из них на ваш взгляд более сложная? Для какой модели вероятность появления эффекта переобучения выше?

7. Если одним из ваших классификаторов была нейронная сеть, охарактеризуйте ее архитектуру.

## 6. Индивидуальные задания

### Вариант 1

Задание – классификация животных.

Имеет хобот	Хищник	Слон	Обезьяна	Крокодил
Да	нет	да	нет	нет
Нет	нет	нет	да	нет
Нет	да	нет	нет	да

### Вариант 2

Задание – определение типа устройства.

Наличие SIM-карты	Маленький размер	ПК	КПК	Мобильный телефон
да	да	нет	нет	да
нет	да	нет	да	нет
нет	нет	да	нет	нет

### Вариант 3

Задание – определение типа музыкального инструмента.

Наличие провода	Имеет струны	Фортепиано	Гитара	Электрогитара
Да	да	нет	нет	да
Нет	да	нет	да	нет
Нет	нет	да	нет	нет

### Вариант 4

Задание – определение типа офисного устройства.

Печатает	Использует формат A4	Принтер	Плоттер	Сканер
да	да	да	нет	нет
да	нет	нет	да	нет
нет	нет	нет	нет	да

### Вариант 5

Задание – определение вида сказочного мифического существа.

Длинные уши	Зеленая кожа	Человек	Орк	Эльф	Троль
да	да	нет	нет	да	нет
нет	да	нет	да	нет	нет
нет	нет	да	нет	нет	нет
да	нет	нет	нет	нет	да

**Вариант 6**

Задание – определение вида музыкального инструмента.

Есть струны	Подключается к колонкам и усилителю	Барабаны	Синтезатор	Гитара
да	да	нет	нет	да
нет	да	нет	да	нет
нет	нет	да	нет	нет

**Вариант 7**

Задание – определение типа летательного аппарата.

Имеет двигатель	Имеет крылья	Имеет пропеллер	Наполнен воздухом	Самолет	Вертолет	Планер	Дирижабль
да	да	нет	нет	да	нет	нет	нет
да	нет	да	нет	нет	да	нет	нет
нет	да	нет	нет	нет	нет	да	нет
нет	нет	нет	да	нет	нет	нет	да

**Вариант 8**

Задание – выбор вида транспорта.

Использует седло	Имеет гусеницы	Велосипед	Автомобиль	Самолет	Танк
да	нет	да	нет	нет	нет
нет	нет	нет	да	да	нет
нет	да	нет	нет	нет	да

**Вариант 9**

Задание – выбор представителя фауны.

Имеет когти	Оно может Вас съесть	Кошка	Тигр	Крокодил	Амёба
да	нет	да	нет	нет	нет
да	да	нет	да	нет	нет
нет	да	нет	нет	да	нет
нет	нет	нет	нет	нет	да

**Вариант 10**

Задание – определение типа мебели

Жесткий	Обитый тканью	Большой	Стул	Кресло	Диван
да	нет	нет	да	нет	нет
нет	да	нет	нет	да	нет
нет	да	да	нет	нет	да

**Вариант 11**

Задание – определение типа стола

Стоит компьютер	Лежит скатерть	Большой размер	Компьютерный стол	Обеденный стол	Серверный стол	Царский стол
да	нет	нет	да	нет	нет	нет
нет	да	нет	нет	да	нет	нет
да	нет	да	нет	нет	да	нет
нет	да	да	нет	нет	нет	да

**Вариант 12**

Задание – определение типа лампы

Нить накаливания	Диод	Лампа накаливания	Светодиодная лампа	Люминесцентная лампа
да	нет	да	нет	нет
нет	да	нет	да	нет
нет	нет	нет	нет	да

**Вариант 13**

Задание – классификация типа посадочного места.

Сидит царь	Сидит холоп	Сидит боярин	Сидит несколько человек	Трон	Земля	Боярский стул	Лавка
да	нет	нет	нет	да	нет	нет	нет
нет	да	нет	нет	нет	да	нет	нет
нет	нет	да	нет	нет	нет	да	нет
нет	нет	нет	да	нет	нет	нет	да

**Вариант 14**

Задание – определение вида спорта

Тарелка	Мяч	Палка	Ружьё	Фризби	Футбол	Городки	Стендовая стрельба	Бейсбол
да	нет	нет	нет	да	нет	нет	нет	нет
нет	да	нет	нет	нет	да	нет	нет	нет
нет	нет	да	нет	нет	нет	да	нет	нет
нет	нет	нет	да	нет	нет	нет	да	нет
да	нет	нет	да	нет	нет	нет	да	нет
нет	да	да	нет	нет	нет	нет	нет	да

**Вариант 15**

Задание – определение эмоционального состояния женщины

Повышенная раздражительность	Не разговаривает	Нормальное состояние	Обиделась	Хочет подарок
да	нет	да	нет	нет
нет	да	нет	да	нет
нет	нет	нет	нет	да

**Вариант 16**

Задание – определение вида птицы

Черный цвет	Большой клюв	Бело-черная раскраска	Больших размеров	Галка	Ворона	Сорока	Ворон
да	нет	нет	нет	да	нет	нет	нет
да	да	нет	нет	нет	да	нет	нет
нет	нет	да	нет	нет	нет	да	нет
да	да	нет	да	нет	нет	нет	да

**Вариант 17**

Задание – определение типа питомца

Мяукает	Гавкает	Кошка	Собака	Грызун	Котопёс
да	нет	да	нет	нет	нет
нет	да	нет	да	нет	нет
нет	нет	нет	нет	да	нет
да	да	нет	нет	нет	да

**Вариант 18**

Задание – определение типа объекта по его поведению после броска тапком.

Шевелит усами	Прячется под диван	Бежит за тапком	Таракан	Кот	Пес	Второй тапок
да	нет	нет	да	нет	нет	нет
нет	да	нет	нет	да	нет	нет
нет	нет	да	нет	нет	да	нет
нет	нет	нет	нет	нет	нет	да

**Вариант 19**

Задание – определение типа шарика

Шарик летит вверх	Шарик падает	Шарик падает с грохотом	Гелиевый	Воздушный	Же- лез- ный	Мыль- ный пу- зырь
да	нет	нет	да	нет	Нет	нет
нет	да	нет	нет	да	нет	нет
нет	нет	да	нет	нет	да	нет
нет	нет	нет	нет	нет	нет	да

**Вариант 20**

Задание – определение типа жидкости

Противно пахнет	Прозрач- ная	Пахнет фруктами	Без запаха	Спирт	Щелочь	Сок
да	да	нет	нет	да	нет	нет
нет	да	нет	да	нет	да	нет
нет	нет	да	нет	нет	нет	да

## **Лабораторная работа №4**

### **«ТЕХНОЛОГИИ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА В ОБРАБОТКЕ И АНАЛИЗЕ ИЗОБРАЖЕНИЙ»**

#### **1. Цель работы**

Приобретение опыта использования технологий искусственного интеллекта в задачах обработки и анализа изображений.

#### **2. Методические указания**

На сегодняшний день искусственные нейронные сети и системы искусственного интеллекта эффективно решают множество сложных и трудно формализуемых задач в различных областях. Практически все крупные компании, представленные на рынке информационных услуг, занимаются разработками в области нейрокомпьютеров и нейроинформатики. Особенно эффективно ими решаются задачи обработки и распознавания сложных изображений [1, 5].

##### **2.1. Задача классификации изображений**

Задача классификации изображений относится к категории задач распознавания образов. Образ – это объект, который можно представить конечным набором признаков. Решение о принадлежности картинки к определенному классу принимается на основе анализа набора признаков, которые характеризуют это изображение. При классификации изображений используются глобальные признаки, то есть такие признаки, которые рассчитываются по всему изображению. Примерами глобальных признаков являются:

- Средняя яркость изображения (или цветовой компоненты);
- Медиана яркости изображения (или цветовой компоненты);
- Среднеквадратическое отклонение яркости (или цветовой компоненты);
- другие.

##### **2.2. Типовой пример выполнения работы классификации изображений**

В качестве примера рассмотрим задачу классификации изображений на 4 класса:

- 1 - летний пейзаж,
- 2 - весенний пейзаж,
- 3 - зимний пейзаж,
- 4 - осенний пейзаж.

Для формирования обучающей выборки были отобраны изображения, которые приведены на рисунке 4.1.



Изображение 1



Изображение 2



Изображение 3



Изображение 4



Изображение 5



Изображение 6



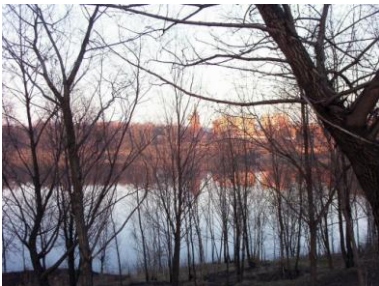
Изображение 7



Изображение 8



Изображение 9



Изображение 10



Изображение 11



Изображение 12

Рис. 4.1. Изображения разных классов для формирования обучающей выборки

Теперь необходимо определиться с признаками, на основе которых будет выполняться классификация. Визуально анализируя изображения, мы видим, что они отличаются по составу цветов. В каждом классе изображений можно выделить свой доминирующий цвет. Поэтому можно с большой вероятностью предположить, что в качестве признаков достаточно будет использовать средние значения трех цветовых компонент изображений: красной, зеленой и синей.

Напишем программу, которая будет вычислять значения признаков на некотором языке программирования. Также определимся с целевыми значениями для машинного обучения. Поскольку в задаче используются 4 класса,



целевых полей тоже будет 4. Значение целевого поля будет равно 1, если изображение принадлежит соответствующему классу и 0 в противном случае. Значения признаков и целевые поля в совокупности представляют собой обучающую выборку, которая в табличной форме может быть представлена следующим образом:

Таблица 4.1

**Вычисленные значения признаков изображений**

Изображение	Среднее значение $R$	Среднее значение $G$	Среднее значение $B$	Класс 1	Класс 2	Класс 3	Класс 4
1	92	111	161	0	0	1	0
2	165	189	217	0	0	1	0
3	138	163	209	0	0	1	0
4	105	114	73	1	0	0	0
5	85	94	32	1	0	0	0
6	123	139	100	1	0	0	0
7	177	101	29	0	0	0	1
8	150	105	43	0	0	0	1
9	91	71	44	0	0	0	1
10	128	122	126	0	1	0	0
11	125	129	130	0	1	0	0
12	134	142	149	0	1	0	0

Задача машинного обучения с таким образом подготовленными данными решалась на лабораторной работе 3. Выполним машинное обучение с учителем, а его результаты программно реализуем. Очевидно, что программа помимо реализации алгоритма прогона нейронной сети (или другого алгоритма на основе машинного обучения) должна в своем составе иметь функции вычисления признаков изображений.

### 2.3. Задача обработки изображений

Принципиальное отличие данной задачи от классификации заключается в том, что решение принимается не по всему изображению (к какому классу его отнести), а по каждой его точке (какого цвета должен быть очередной пиксель в результирующем изображении). В связи с этим можно отметить следующие отличительные особенности данного класса задач [3]:

- вместо глобальных признаков изображений используются локальные признаки.
- для вычисления признаков преимущественно используются алгоритмы обработки, а не анализа изображений.
- объем обучающей выборки (количество прецедентов) значительно превосходит количество изображений,
- как правило, задачи обработки изображений с использованием искусственного интеллекта являются более трудоемкими из-за многократного запуска классификатора принятия решения.

## 2.4. Типовой пример выполнения работы обработки изображений

В качестве примера рассмотрим задачу выделения кровоизлияний на снимках глазного дна для диагностирования диабетической ретинопатии, которая сводится к бинаризации изображения. Эталонное бинарное изображение строится экспертом путем указания действительных областей кровоподтеков (рис. 4.2).

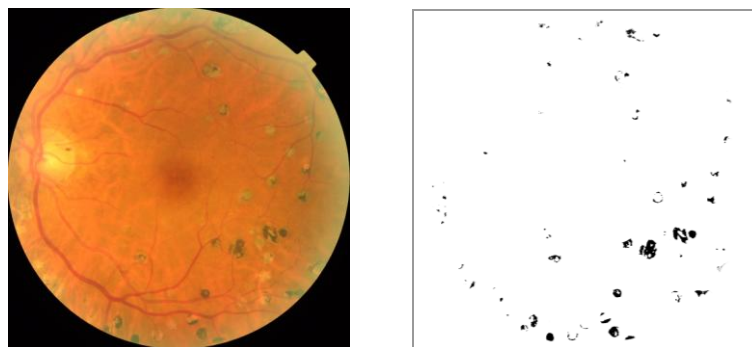
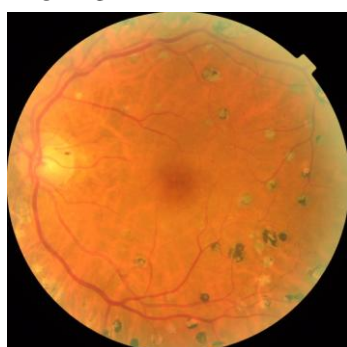
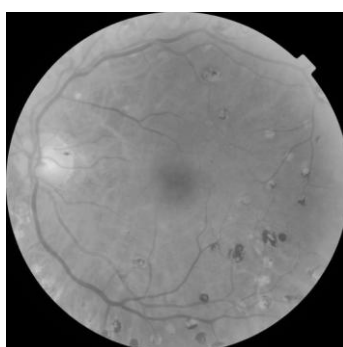


Рис. 4.2. Исходное и эталонное бинарное изображения

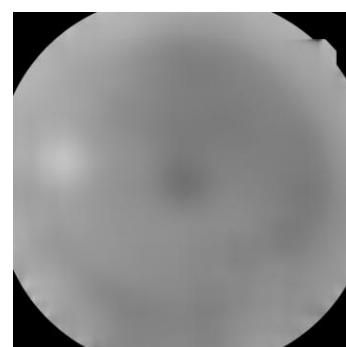
Локальные признаки для исходного изображения могут быть получены обработкой (некоторым алгоритмом) этого изображения с получением на выходе полутонового изображения [3, 7, 8]. То есть набором признаков для классификации точки с координатами  $(x,y)$  будут яркости точек с координатами  $(x,y)$  выходных полутоновых изображений. Примеры полутоновых изображений (которые также являются визуализацией локальных признаков) приведены на рисунке 4.3.



Исходное изображение



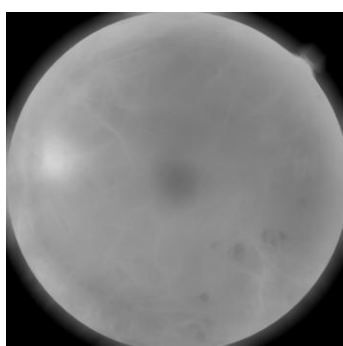
Признак яркости точек изображения



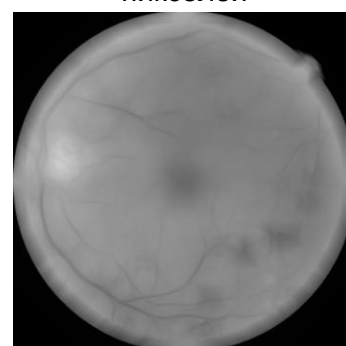
Признак максимальной яркости в окрестности 20\*20 пикселей



Признак протяженности выделенных темных объектов



Признак максимальной яркости линейных сегментов



Признак минимальной яркости линейных сегментов

Рис. 4.3. Изображения извлеченных признаков

Напишем программу, которая выполняет  $A$  алгоритмов обработки изображений. По набору исходных изображений сформируем обучающую выборку. Одно исходное изображение, размером  $X \times Y$ , позволит создать  $X \cdot Y$  прецедентов. Количество полей-признаков равно  $A$ , а целевое поле будет одно.

Выполнив машинное обучение, мы получим алгоритм, который по совокупности значений локальных признаков относительно некоторой точки с координатами  $(x, y)$  предскажет значение сегмента этой точки и, таким образом, позволит выполнять сегментацию произвольного изображения сетчатки глаза.

### **3. Задание на лабораторную работу**

1. Изучить возможности искусственного интеллекта в задачах обработки и анализа изображений.
2. Сконструировать алгоритм классификации (или обработки, в соответствии с вариантом) изображения путем обучения, для этого:
  - сформировать обучающую выборку;
  - выполнить подбор признаков для обучения;
  - выполнить обучение. Если результаты неудовлетворительные, вернуться к предыдущим пунктам;
  - внедрить достигнутый результат в собственную разработку.
3. Протестировать полученную программу.
4. Сделать выводы.

### **4. Требования к содержанию отчета**

1. Цель лабораторной работы.
2. Индивидуальное задание.
3. Примеры отобранных изображений для обучающей выборки.
4. Описание признаков с формулами их вычисления.
5. Данные обучающей выборки (в табличной или другой форме).
6. Описание используемого классификатора (например, граф нейронной сети и его краткое описание).
7. Результаты обучения классификатора.
8. Листинг программы.
9. Описание результатов работы программы.
10. Выводы по лабораторной работе.

### **5. Контрольные вопросы**

1. Задачи обработки изображений, решаемые системами искусственного интеллекта.
2. Какую архитектуру нейронной сети можно использовать для обработки изображений?
3. Что из себя представляет обучающая выборка для обучения нейронной сети обработке изображений?

4. Особенность обучения нейронной сети обработке изображений на основе вычисления локальных признаков изображений.

5. Основные параметры работы генетического алгоритма обучения нейронной сети?

6. Приведите пример задач распознавания образов.

## **6. Индивидуальные задания**

### **Варианты заданий на лабораторную работу**

1. Классификация изображений волнистых попугаев по цветам.
2. Классификация изображений бабочек.
3. Классификация изображений пиццы.
4. Классификация спутниковых изображений по времени года.
5. Классификация спутниковых изображений по типу местности.
6. Классификация типов кирпичной кладки.
7. Классификация изображений автомобилей по их цвету.
8. Классификация изображений обоев по цветам.
9. Классификация типов облачности по изображениям неба.
10. Классификация изображений из ленты социальной сети по типам.
11. Преобразование полутонового изображения в цветное (использовать однотипные изображения, объекты которых имеют одинаковый цвет, например, небо, лес, закат солнца и так далее).
12. Сегментация текстового блока на изображении, содержащем текст.
13. Поиск ярлычков рабочего стола на его изображении.
14. Разделение жирного и обычного текста на изображении текстового документа.
15. Выделение линий чертежного документа.
16. Нахождение точки лазерной указки.
17. Выделение текстовой информации на изображении документа (например, пластиковая карточка, права, страховое свидетельство и так далее)
18. Распознавание фигур на шахматной доске (можно взять шашки или подобную игру с доской).
19. Автоматическая корректировка яркости и контрастности изображения.
20. Выделение номера банкноты.
21. Поиск линий на изображении ладони.
22. Поиск отметок на листе (опросник, лист голосования, анкета)
23. Поиск окон на фотографии дома.
24. Поиск элемента лица человека на его фотографии.
25. Выделение записей на (маркерной или меловой) доске.
26. Выделение почтового индекса на конверте.

### Библиографический список

1. **Alexey Varlamov, Ruslan Sharapov.** Machine Learning of Visually Similar Images Search // Proceedings of the 14th All-Russian Scientific Conference "Digital libraries: Advanced Methods and Technologies, Digital Collections", vol. 934, Pereslavl-Zalessky, Russia, October 15-18, 2012, pp. 113-120.
2. **Ахметова М.** Использование стилей программирования для лабораторных работ по искусственным нейронным сетям // Перспективы, организационные формы и эффективность развития сотрудничества российских и зарубежных вузов III Ежегодная международная научно-практическая конференция. Технологический университет. 2015. С. 60-63.
3. **Варламов А.Д.** Восстановление цвета полутоновых изображений нейронной сетью // Алгоритмы, методы и системы обработки данных: Сборник научных статей/Под ред. С.С. Садыкова, Д.Е. Андрианова – М.: Горячая линия – Телеком, 2011. – С. 55-62.
4. **Исаин Н.В.** Нейронный прогноз мировых цен на нефть // Экономические стратегии. 2013. Т. 15. № 2 (110). С. 66-67.
5. **Муравьева Е.А., Сагдатуллин А.М., Емекеев А.А.** Artificial intelligence and artificial neural network in images recognition // Материалы научной сессии ученых Альметьевского государственного нефтяного института. 2014. Т. 1. № 2. С. 98-101.
6. **Рутковская Д.** Нейронные сети, генетические алгоритмы и нечеткие системы / М. Пилиньский, Л. Рутковский, пер.: И.Д. Рудинский, Д. Рутковская. – 2-е изд., стер. – М.: Горячая линия – Телеком, 2013. – пер. с польск. – ISBN 978-5-9912-0320-3
7. **Старожилова О.В.** Решение задач идентификации неоднородностей на изображениях с использованием нейронной сети // Международный научно-исследовательский журнал. 2015. № 2-1 (33). С. 83-84.
8. **Чижов В.С., Ковалев Ю.А., Варламов А.Д.** Разработка метода повышения качества поиска лиц на изображениях анализом их биометрических признаков // Алгоритмы, методы и системы обработки данных. Выпуск 2(27), 2014, с. 55-63.