

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Емельянов Сергей Геннадьевич

Должность: ректор

Дата подписания: 27.04.2023 09:21:17

Уникальный программный ключ:

9ba7d3e34c012eba476ffd2d064cf2781953be730df2374

МИНИСТЕРСТВО НАУКИ РОССИИ

Федеральное государственное бюджетное образовательное  
учреждение высшего образования

«Юго-Западный государственный университет»  
(ЮЗГУ)

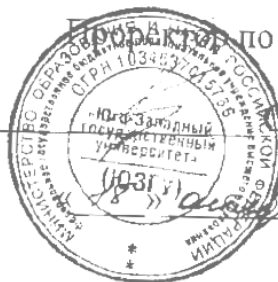
Кафедра программной инженерии

УТВЕРЖДАЮ

Профессор по учебной работе

О.Г. Локтионова

2020 г.



**НЕЙРОННЫЕ СЕТИ. ОБУЧЕНИЕ НЕЙРОННОЙ СЕТИ  
ВЫПОЛНЕНИЮ ЗАДАННОЙ ОПЕРАЦИИ**

Методические указания по выполнению лабораторной работы  
по дисциплине «Экспертные системы» для студентов направления  
подготовки 09.04.04 «Программная инженерия»

Курск 2020

УДК 004.65

Составители: В.Г. Белов, Т.М. Белова

Рецензент

Кандидат технических наук, доцент кафедры программной инженерии ЮЗГУ И.Н. Ефремова

**Нейронные сети. Обучение нейронной сети выполнению заданной операции:** методические указания по выполнению лабораторной работы по дисциплине " Экспертные системы" для студентов направления подготовки 09.04.04 "Программная инженерия" / Юго-Зап. гос. ун-т; сост.: В.Г. Белов, Т.М. Белова, – Курск, 2017. – 13 с.: ил. 4.

Изложены основные алгоритмы настройки нейронных сетей при помощи системы MATLAB.

Материал предназначен для студентов направления подготовки 09.04.04 «Программная инженерия» (профиль "Разработка информационно-вычислительных систем")

Подписано в печать . Формат 60x84 1/16.

Усл. печ. л. . Уч.-изд. л. . Тираж 100 экз. Заказ . Бесплатно.

Юго-Западный государственный университет

305040, Курск, ул.50 лет Октября, 94.

## Содержание

1 Цель работы .....	4
2 Общие сведения.....	5
3 Порядок выполнения работы .....	20
4 Содержание отчета.....	21
5 Контрольные вопросы .....	22
6 Список использованных источников .....	23

## 1 Цель работы

Цель работы: ознакомиться с интерфейсом NNTool системы MATLAB, создать и обучить нейронную сеть выполнению заданной операции на основании сведений [1-4].

## 2 Общие сведения

GUI-интерфейс для NNTool позволяет, не обращаясь к командному окну системы MATLAB, выполнять создание, обучение, моделирование, а также импорт и экспорт нейронных сетей и данных. Конечно, такие инструменты наиболее эффективны лишь на начальной стадии работы с пакетом, поскольку имеют определенные ограничения. В частности, интерфейс NNTool допускает работу только с простейшими однослойными и двухслойными нейронными сетями, но при этом пользователь выигрывает во времени и эффективности освоения новых объектов.

Вызов GUI-интерфейса NNTool возможен либо командой `nntool` из командной строки, либо из окна запуска приложений Launch Pad с помощью опции NNTool из раздела Neural Network Toolbox. После вызова на экране терминала появляется окно **Network/Data Manager** (Управление сетью/данными).

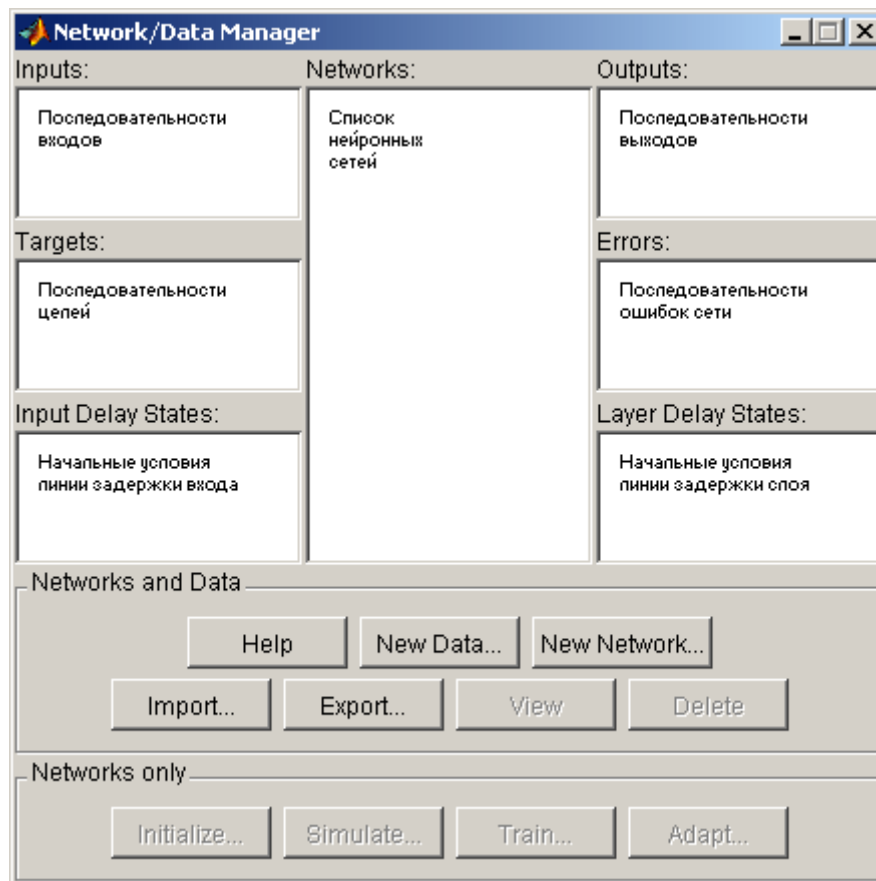


Рис. 1

- **Help** – кнопка вызова окна подсказки **Network/Data Manager Help**;
- **New Data...** – кнопка вызова окна формирования данных **Create New Data** (рис. 2);
- **New Network...** – кнопка вызова окна создания новой нейронной сети **Create New Network** (рис. 3);
- **Import...** – кнопка вызова окна для импорта или загрузки данных **Import or Load to Network/Data Manager** (рис. 4);
- **Export...** – кнопка вызова окна для экспорта или записи данных в файл **Export or Save from Network/Data Manager** (рис. 5).

Кнопки **View**, **Delete** становятся активными только после создания и активизации данных, относящихся к последовательностям входа, цели, выхода или ошибок сети. Кнопка **View** позволяет просмотреть, а кнопка **Delete** удалить активизированные данные.

Кнопки **View**, **Delete**, **Initialize...**, **Simulate...**, **Train...**, **Adapt...** становятся активными после создания и активизации самой нейронной сети. Они позволяют просмотреть, удалить, инициализировать, промоделировать, обучить или адаптировать нейронную сеть и будут описаны ниже.

Прежде всего, рассмотрим назначение и способы работы с перечисленными выше окнами. Окно **Network/Data Manager Help**. Это окно подсказки описывает правила работы с диспетчером **Network/Data Manager** при создании нейронной сети.

Чтобы создать нейронную сеть, необходимо выполнить следующие операции:

- Сформировать последовательности входов и целей (кнопка **New Data**), либо загрузить их из рабочей области системы MATLAB или из файла (кнопка **Import**).
- Создать новую нейронную сеть (кнопка **New Network**), либо загрузить ее из рабочей области системы MATLAB или из файла (кнопка **Import**).
- Выбрать тип нейронной сети и нажать кнопку **Train...**, чтобы открыть окно для задания параметров процедуры обучения.
- Открыть окно **Network** для просмотра, инициализации, моделирования, обучения и адаптации сети.

Окно **Create New Data**. Это окно показано на рис. 2 и включает 2 области редактирования текста для записи имени вводимых данных (область **Name**) и ввода самих данных (область **Value**), а также 6 кнопок для указания типа вводимых данных.

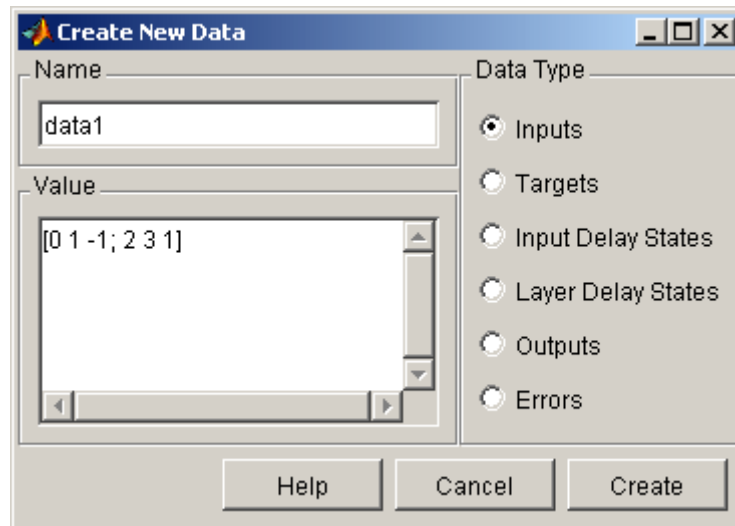


Рис. 2

Различают следующие типы данных:

- **Inputs** (Входы) – последовательность значений входов;
- **Targets** (Цели) – последовательность значений цели;
- **Input Delay States** (Состояния ЛЗ входа) – начальные условия линии задержки на входе;
- **Layer Delay States** (Состояния ЛЗ слоя) – начальные условия линии задержки в слое;
- **Outputs** (Выходы) – последовательность значений выхода сети;
- **Errors** (Ошибки) – разность значений целей и выходов.

Как правило, пользователь задает только последовательности входа и цели, т.е. типы данных **Inputs** и **Targets**. При этом следует помнить, что при адаптации нейронной сети данные должны быть представлены в виде массива ячеек.

Окно **Create New Network**. Это окно показано на рис. 3 и включает поля для задания параметров создаваемой сети. В зависимости от типа сети количество полей и их названия изменяются.



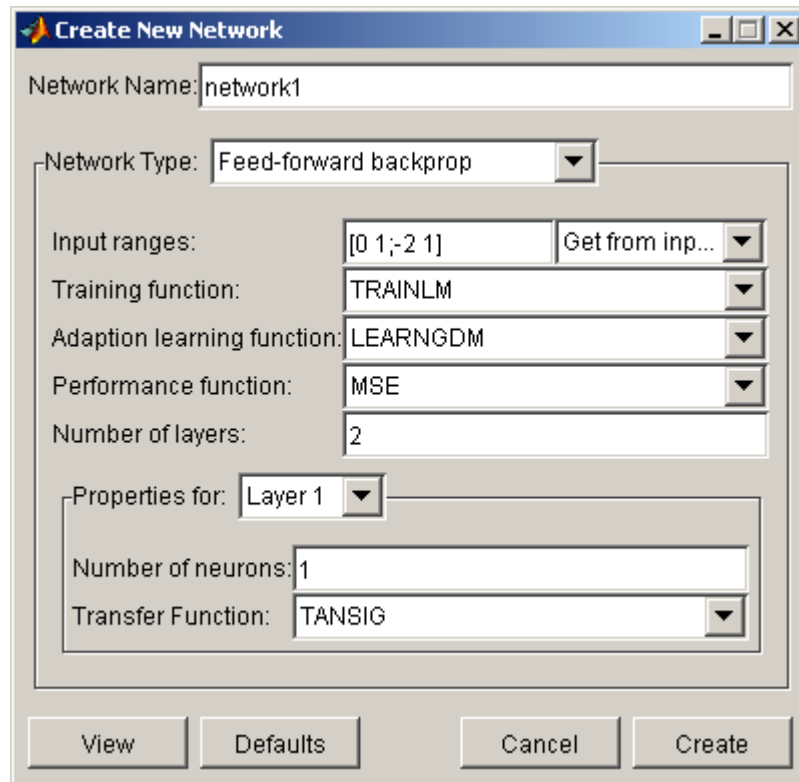


Рис. 3

Обратимся к описанию полей:

- **Network Name** (Имя сети) – стандартное имя сети, присваиваемое GUI-интерфейсом NNTool; в процессе создания новых сетей порядковый номер будет изменяться автоматически.
- **Type** (Тип сети) – список сетей, доступных для работы с интерфейсом NNTool. Для удобства этот список повторен в нижеследующей таблице. Интерфейс NNTool позволяет создавать нейронные сети только с одним или двумя слоями.

Таблица 1

<i>№ n/n</i>	<i>Тип сети</i>	<i>Название сети</i>	<i>Число слоев</i>	<i>Обучаемые параметры</i>
1	Competitive	Конкурирующая сеть	1	$IW\{1, 1\}, b\{1\}$
2	Cascade- forward backprop	Каскадная сеть с прямым распространением сигнала и обратным распространением ошибки	2	$IW\{1, 1\}, b\{1\},$ $LW\{2, 1\},$ $IW\{2, 1\}, b\{2\}$

3	Elman backprop	Сеть Элмана с обратным распространением ошибки	2	$IW\{1, 1\}, b\{1\},$ $LW\{2, 1\}, b\{2\},$ $LW\{1, 1\}$
4	Feed-forward backprop	Сеть с прямым распространением сигнала и обратным распространением ошибки	2	$IW\{1, 1\}, b\{1\},$ $LW\{2, 1\}, b\{2\}$
5	Time delay backprop	Сеть с запаздыванием и обратным распространением ошибки	2	$IW\{1, 1\}, b\{1\},$ $LW\{2, 1\}, b\{2\}$
6	Generalized regression	Обобщенная регрессионная сеть	2	$IW\{1, 1\}, b\{1\},$ $LW\{2, 1\}$
7	Hopfield	Сеть Хопфилда	1	$LW\{1, 1\}, b\{1\}$
8	Linear layer (design)	Линейный слой (создание)	1	$IW\{1, 1\}, b\{1\}$
9	Linear layer (train)	Линейный слой (обучение)	1	$IW\{1, 1\}, b\{1\}$
10	LVQ	Сеть для классификации входных векторов	2	$IW\{1, 1\},$ $LW\{2, 1\}$
11	Perceptron	Перцептрон	1	$IW\{1, 1\}, b\{1\}$
12	Probabalistic	Вероятностная сеть	2	$IW\{1, 1\}, b\{1\},$ $LW\{2, 1\}$
13	Radial basis (exact fit)	Радиальная базисная сеть с нулевой ошибкой	2	$IW\{1, 1\}, b\{1\},$ $LW\{2, 1\}$
14	Radial basis (fewer neurons)	Радиальная базисная сеть с минимальным числом нейронов	2	$IW\{1, 1\}, b\{1\},$ $LW\{2, 1\}, b\{2\}$
15	Self-organizing map	Самоорганизующаяся карта Кохонена	1	$IW\{1, 1\}$

- **Input ranges** (Диапазоны входа) – допустимые границы входов, которые либо назначаются пользователем, либо определяются автоматически по имени входной последовательности, выбираемой из списка **Get from Inp...**
- **Training function** (Функция обучения) – список обучающих функций.
- **Adaption learning function** (Функции настройки для режима адаптации) – список функций настроек.

- **Performance function** (Функция качества обучения) – список функций оценки качества обучения.
- **Number of layers** (Количество слоев) – количество слоев нейронной сети.
- **Properties for** (Свойства) – список слоев: Layer 1 (Слой 1), Layer 2 (Слой 2).
- **Number of neurons** (Количество нейронов) – количество нейронов в слое.
- **Transfer function** (Функция активации) – функция активации слоя.

Окно **Import or Load to Network/Data Manager**. Это окно показано на рис. 4 и включает 3 поля.

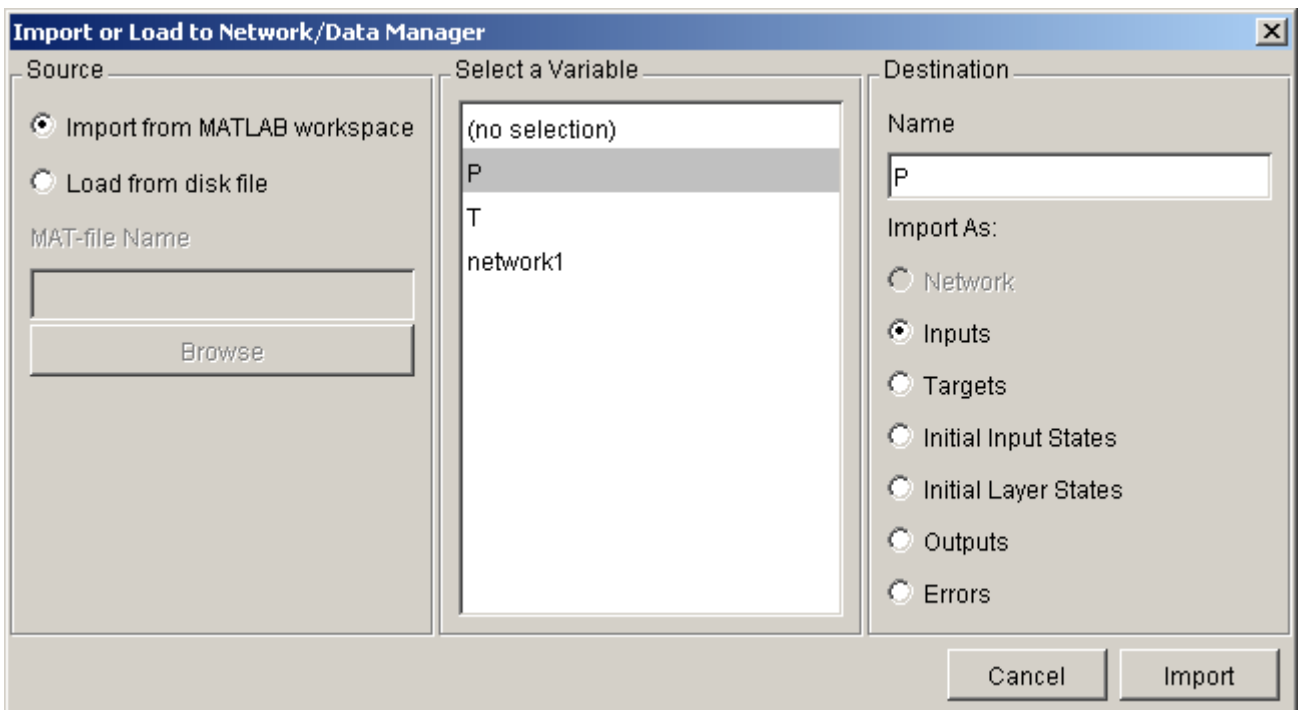


Рис. 4

**Source** (Источник) – поле для выбора источника данных. Это либо рабочая область системы MATLAB (кнопка выбора **Import from MATLAB Workspace**), либо файл (кнопка выбора **Load from disk file**).

Если выбрана первая кнопка, то в поле **Select a Variable** вы можете видеть все переменные рабочей области и, выбрав одну из них, например P1, можете определить ее в поле **Destination** (Назначение) как последовательность входа **Inputs** (Входы).

Если выбирается кнопка **Load from disk file**, то активизируется поле **MAT-file Name** и кнопка **Browse**, что позволяет начать поиск и загрузку файла из файловой системы.

Окно **Export or Save from Network/Data Manager**. Это окно показано на рис. 5 и позволяет передать данные из рабочей области GUI-интерфейса NNTool в рабочую область системы MATLAB или записать их в виде файла на диске.

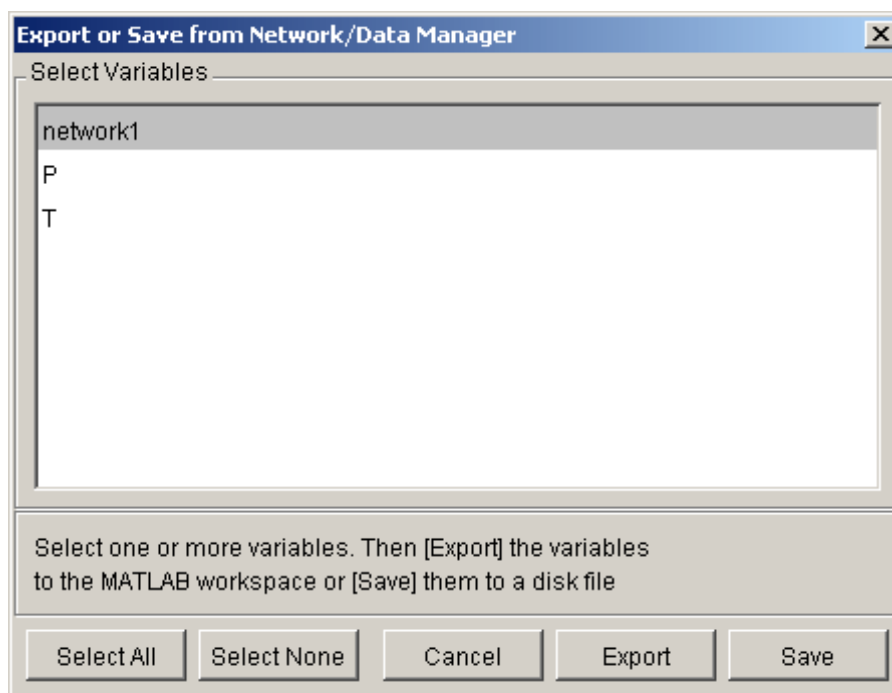


Рис. 5

В данном случае нами выбрана переменная `network1`, которая принадлежит к классу `network object` и описывает нейронную сеть. После того как эта переменная экспортирована в рабочую область, можно, например, построить модель нейронной сети в системе Simulink с помощью оператора `gensim`.

Диалоговая панель **Network** показана на рис. 6.

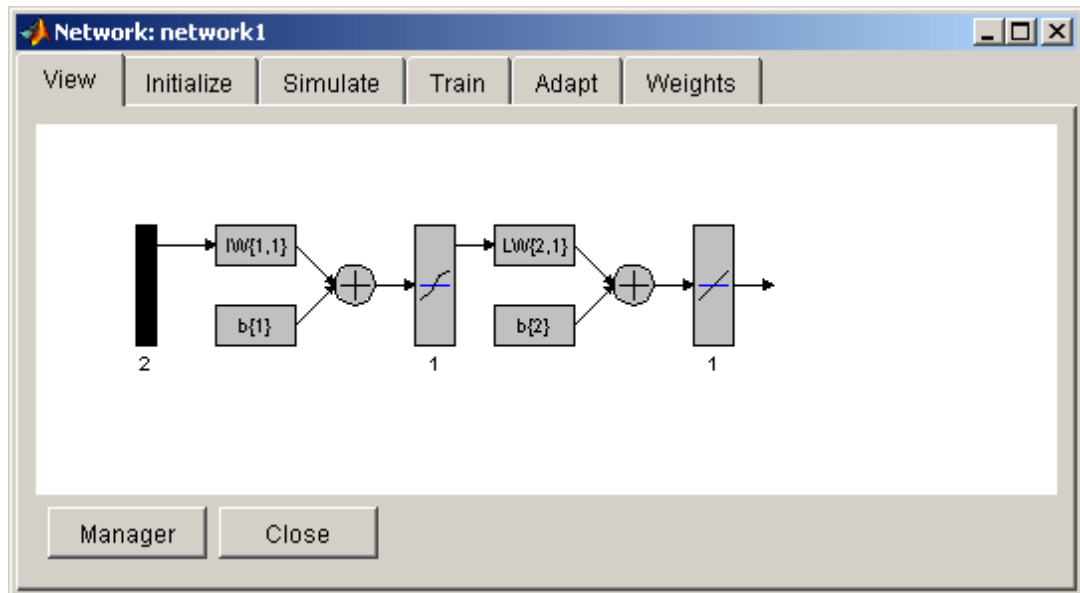


Рис. 6

Она открывается только в том случае, когда в окне **Network/Data Manager** выделена созданная сеть останоятся активными кнопки **View**, **Initialize**, **Simulate**, **Train**, **Adapt**.

Панель имеет 6 закладок:

- **View** (Просмотреть) – структура сети;
- **Initialize** (Инициализация) – задание начальных весов и смещений;
- **Simulate** (Моделирование) – моделирование сети;
- **Train** (Обучение) – обучение сети;
- **Adapt** (Адаптация) – адаптация и настройка параметров сети;
- **Weights** (Весы) – просмотр установленных весов и смещений.

*Пример.* Нейронная сеть с прямой передачей сигнала.

Создать и обучить нейронную сеть выполнению операции  $y = x_1^2 + x_2$ ,

если заданы последовательности входа  $P = [1 \ 0.5 \ 0 \ 1; -2 \ 0 \ 0.5 \ 1]$  и цели  $T = [-1 \ 0.25 \ 0.5 \ 2]$ .

Сформируем последовательности входов и целей в рабочей области GUI-интерфейса NNTool, используя окно **Create New Data**. Выберем нейронную сеть типа feed-forward backprop с прямой передачей сигнала и с обратным распространением ошибки. Схема этой сети показана на рис. 6.

Выполним инициализацию сети, для чего выберем закладку **Initialize**, откроется диалоговая панель, показанная на рис. 7. Диапазоны значений исходных данных выберем по входам из ниспадающего меню **Get from input**. Для ввода установленных диапазонов и инициализации весов надо воспользоваться кнопками **Set Ranges** (Установить диапазоны) и **Initialize Weights** (Инициализировать веса). Если требуется вернуться к прежним диапазонам, то следует выбрать кнопки **Revert Ranges** (Вернуть диапазоны) и **Revert Weights** (Вернуть веса).

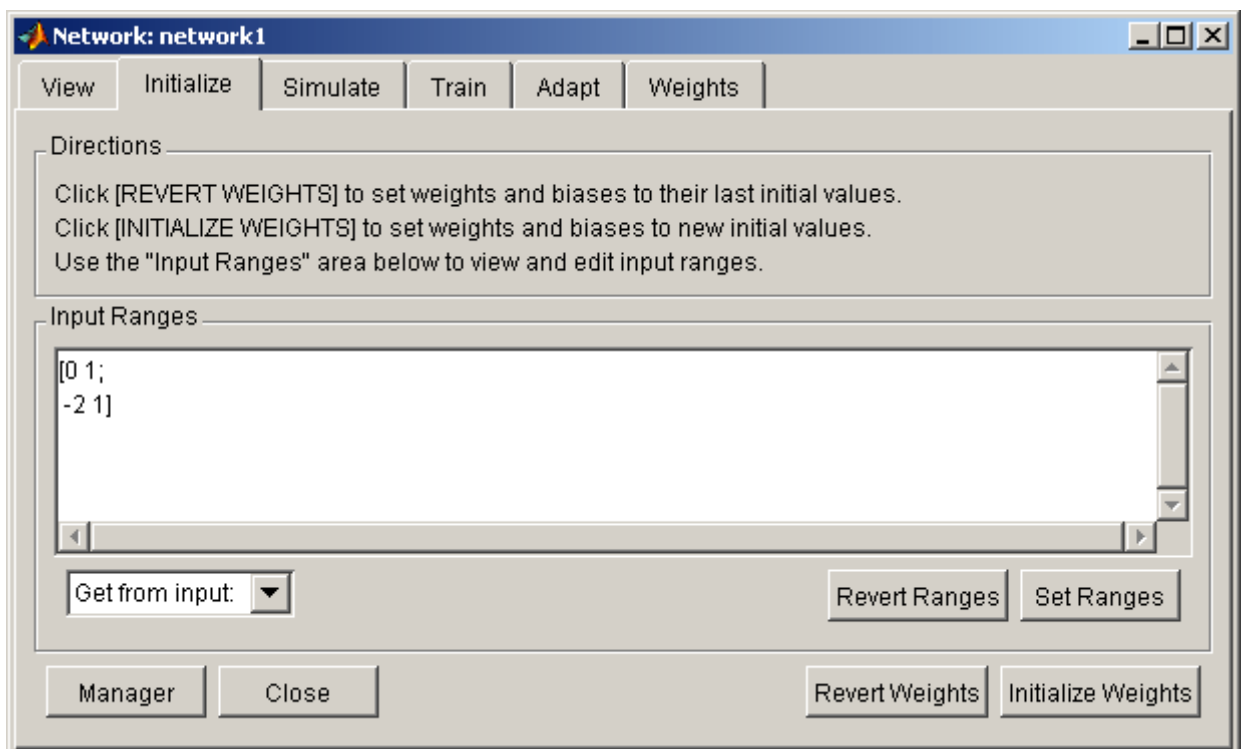


Рис. 7

Затем выполняется обучение сети, для чего выбирается закладка **Train** и открывается диалоговая панель, показанная на рис. 8.

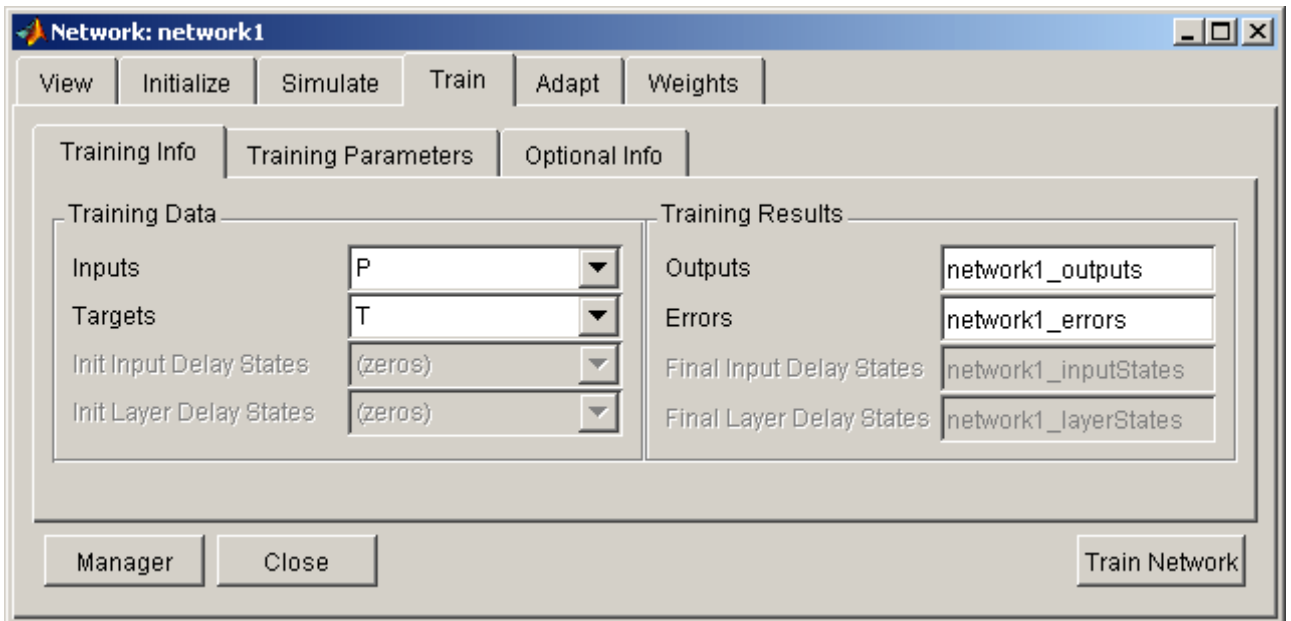


Рис. 8

Панель имеет три закладки:

- **Training Info** (Информация об обучающих последовательностях) (рис. 8);
- **Training Parameters** (Параметры обучения) (рис. 9);
- **Optional Info** (Дополнительная информация) (рис. 10).

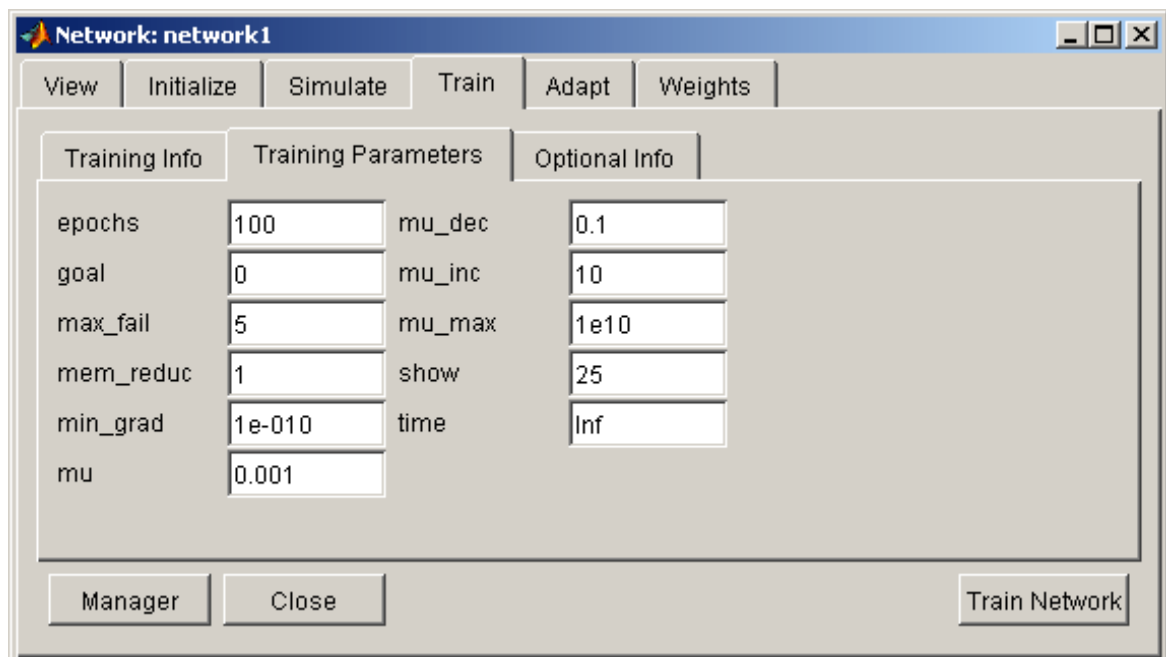


Рис. 9

Последняя закладка применяется, когда в процессе обучения используются контрольные и тестовые последовательности.

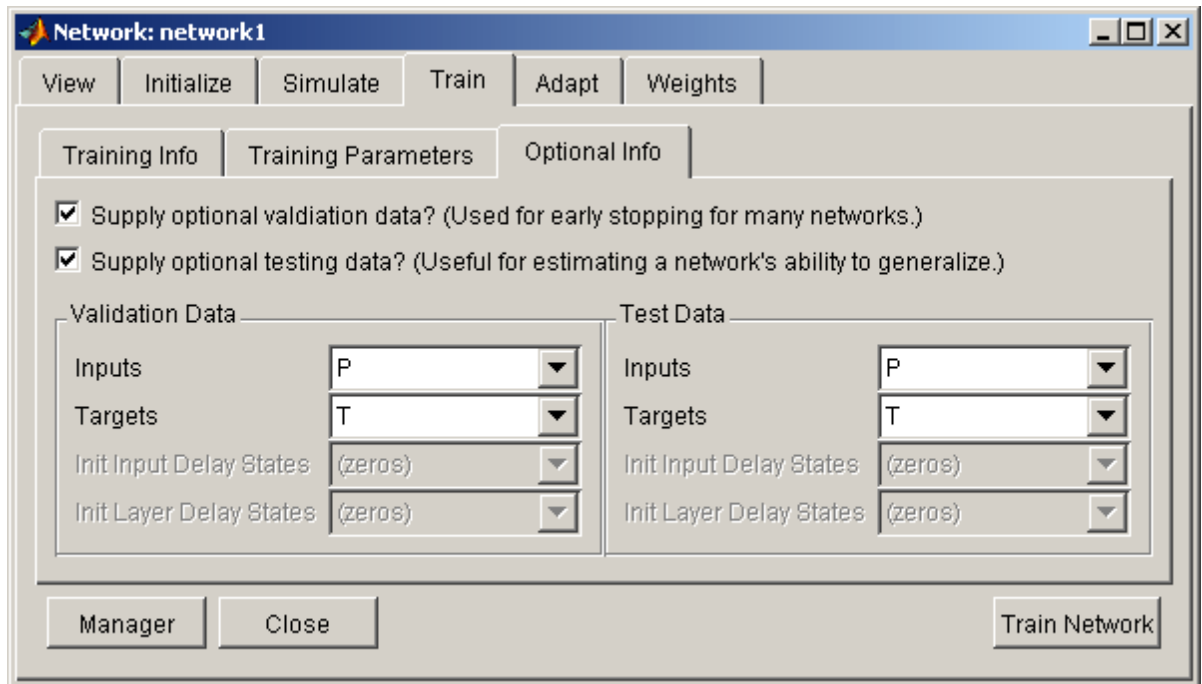


Рис. 10

Применяя эти закладки, можно установить имена последовательностей входа и цели, а также параметров процедуры обучения.

Теперь можно приступить к обучению сети (кнопка **Train Network**).

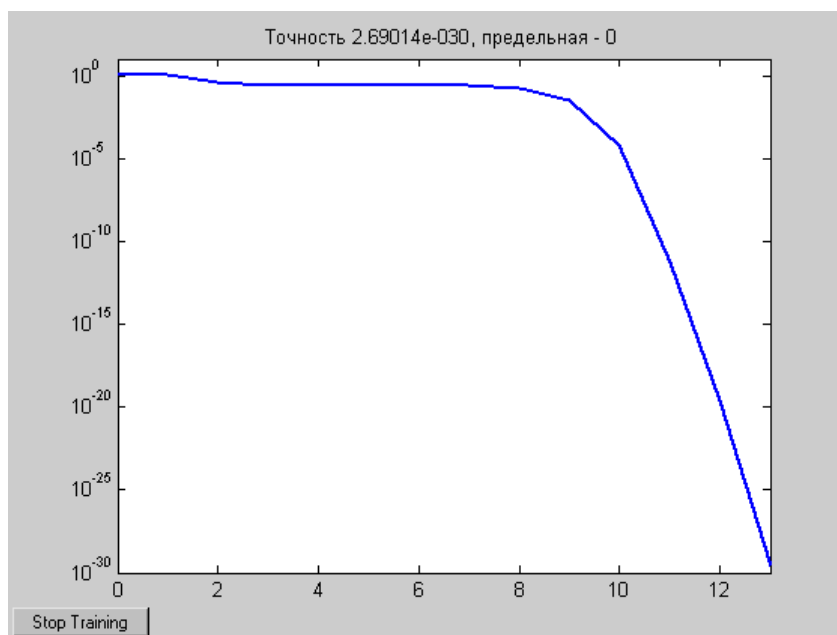


Рис. 11



Качество обучения сети с прямой передачей сигнала на выбранной обучающей последовательности поясняется на рис. 11. Практически нулевая точность достигается за 13 циклов обучения.

Соответствующие веса и смещения можно увидеть, если выбрать закладку **Weights** (рис. 12).

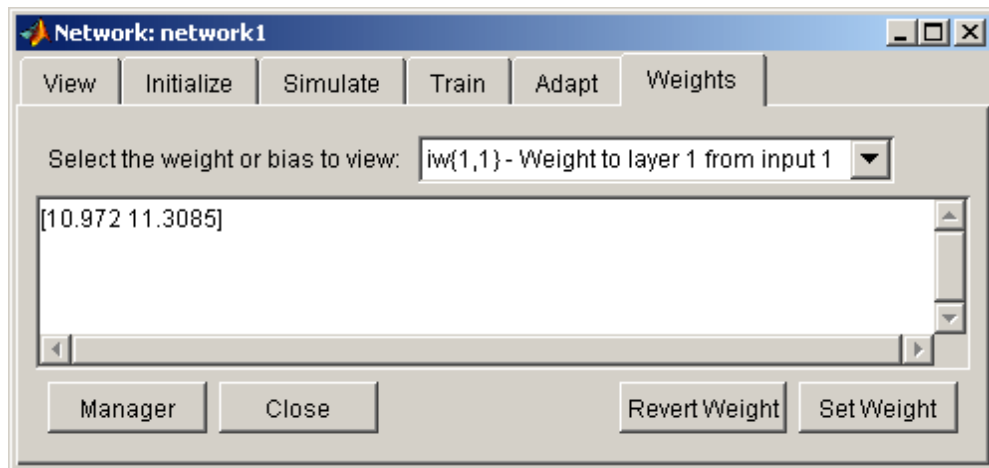


Рис. 12

Для удобства работы можно экспортировать созданную нейронную сеть в рабочую область системы MATLAB и получить информацию о весах и смещениях непосредственно в рабочем окне системы:

```
network1.IW{1, 1}, network1.b{1}
ans = 10.972 11.3085
ans = -5.6542
```

```
network1.LW{2, 1}, network1.b{2}
ans = 1.5
ans = 0.5
```

Результаты обучения можно просмотреть в окне **Network/Data Manager**, выбрав кнопку **Manager** (рис. 12). Появляется окно (рис. 13); теперь, активизируя имена последовательностей выхода или ошибок `network1_outputs` и `network1_errors`, можно просмотреть результаты, используя кнопку **View**. Получаем выходные данные, практически равные целям, и ошибки порядка  $10^{-15}$ .

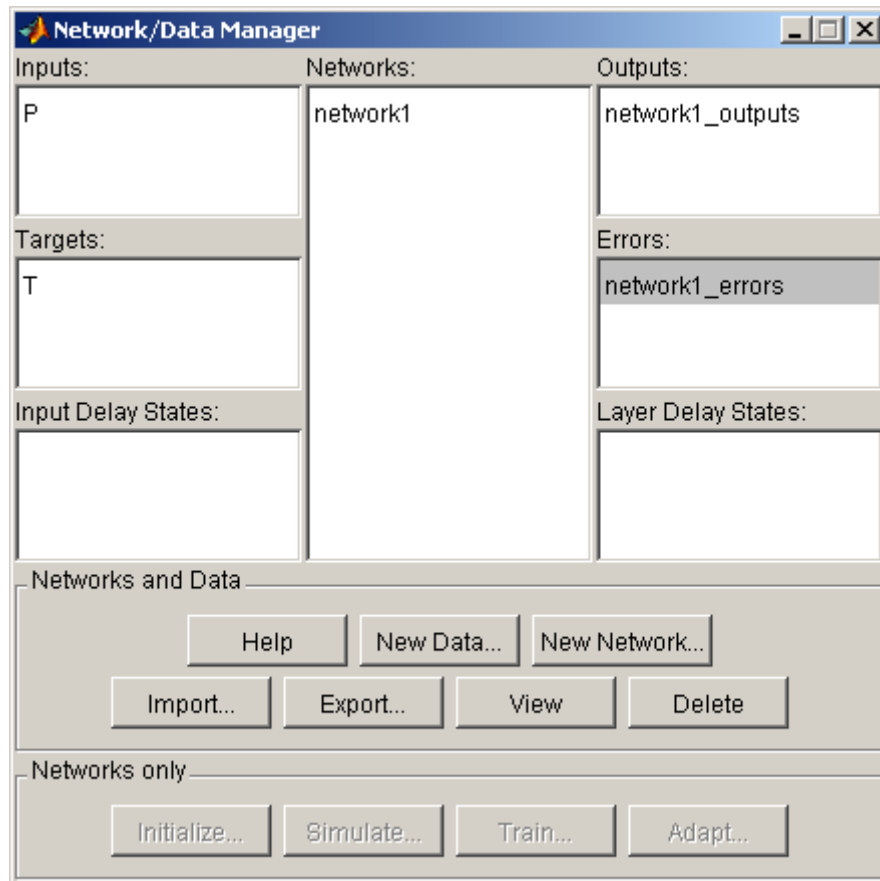


Рис. 13

При моделировании работы нейронной сети в Excel следует использовать следующие формулы для вычисления функций активации (передаточных функций) нейрона:

- Линейная (PURELIN):  $y = x$
- Логистическая (LOGSIG):  $y = 1/(1+\exp(-x))$
- Гиперболический тангенс (TANSIG):  

$$y = (\exp(x)-\exp(-x))/(\exp(x)+\exp(-x)) = \tanh(x)$$

При этом математическая модель нейрона описывается следующими соотношениями:

$$y = f(w_1x_1 + w_2x_2 + \dots + w_Nx_N + b)$$

где  $w_i$  – вес синапса (в данной работе используется нейронная сеть с двумя слоями, которые характеризуются матрицами весов, обозначаемыми IW (Input Weight) и LW (Layer Weight)),  $b$  – значение смещения (bias),  $x_i$  –

компонента входного вектора (входной сигнал),  $y$  – выходной сигнал нейрона,  $N$  – число входов нейрона,  $f$  – нелинейное преобразование (функция активации).

### 3 Порядок выполнения работы

1. Изучить теоретическую часть.
2. Получить у преподавателя вариант задания.
3. Создать и обучить нейронную сеть выполнению заданной операции.
4. Промоделировать созданную нейронную сеть в Excel, используя соответствующие веса и смещения, полученные в результате обучения.
5. Распечатать полученные результаты.
6. Оформить отчет.

## 4 Содержание отчета

1. Вариант задания.
2. Структура используемой нейронной сети.
3. Распечатки экрана Excel с результатами моделирования (в качестве входных данных использовать также данные, которые не входили в обучающую выборку).
4. Выводы по работе.

## 5 Контрольные вопросы

1. Назначение и преимущества использования нейронных сетей.
2. Какие действия необходимо выполнить для создания нейронной сети?
3. Что такое функция активации нейрона?

## 6 Список использованных источников

1. Искусственный интеллект. – В 3-х кн. Кн. 2. Модели и методы: Справочник / под ред. Д. А. Поспелова – М.: Радио и связь, 1990.
2. Галушкин А.И. Теория нейронных сетей. – М.: ИПРЖР, 2000.
3. Горбань А. Н. Обучение нейронных сетей. – М.: СП Параграф, 1990. Круглов В. В., Борисов В. В. Искусственные нейронные сети. Теория и практика. – М.: Горячая линия - Телеком, 2001.
4. Системы искусственного интеллекта [Электронный ресурс]: Методические указания к выполнению лабораторных работ / Сост. Гудков П.А. - Пенза: Пензенский гос. ун-т, 2007. - 53 с. – Режим доступа: <http://window.edu.ru/resource/709/59709>