

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Локтионова Оксана Геннадьевна
Должность: проректор по учебной работе
Дата подписания: 27.09.2024 07:37:09
Уникальный программный ключ:
0b817ca911e6668a

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Юго-Западный государственный университет» (ЮЗГУ)

Кафедра программной инженерии



Утверждаю:
Проректор по учебной работе

О.Г. Локтионова

« 11 » 06

2024г.

**КЛАССИФИКАЦИИ ИНФОРМАТИВНЫХ
ПРИЗНАКОВ НА ОСНОВЕ НЕЙРОННОЙ СЕТИ
ПРЯМОГО РАСПРОСТРАНЕНИЯ ТИПА ОДНОСЛОЙНЫЙ
ПЕРЦЕПТРОН**

Методические указания для выполнения лабораторной работы по дисциплине «Нейронные сети и нейрокомпьютеры» для студентов направления подготовки 09.04.04 ОПОП ВО Программная инженерия, направленность (профиль) «Предпринимательство, инновации и технологии будущего в программной инженерии»

Курск 2024

УДК 004.932

Составитель: Р.А. Томакова

Рецензент

Кандидат технических наук, к.т.н., доцент А.В. Малышев

Классификация информативных признаков на основе нейронной сети прямого распространения типа однослойный перцептрон: методические указания для выполнения лабораторной работы по дисциплине «Нейронные сети и нейрокомпьютеры» для студентов направления подготовки 09.04.04 ОПОП ВО Программная инженерия, направленность (профиль) «Предпринимательство, инновации и технологии будущего в программной инженерии» / Юго-Зап. гос. ун-т; сост. Р.А. Томакова. Курск, 2024.–26 с.

Рассмотрена методика классификации информативных признаков на основе однослойной нейронной сети типа перцептрон, приведено исследование различных моделей ИНС в системе **MATLAB** с использованием графического инструмента пользователя **NNTOOL** из пакета **NeuralNetworkToolbox**.

Методические указания составлены в соответствии с ФГОС ВО – магистратура по направлению подготовки 09.04.04 «Программная инженерия» на основании учебного плана ОПОП ВО 09.04.04 Программная инженерия, направленность (профиль) «Предпринимательство, инновации и технологии будущего в программной инженерии».

Предназначены для студентов, обучающихся по направлению подготовки 09.04.04 Программная инженерия, направленность (профиль) «Предпринимательство, инновации и технологии будущего в программной инженерии» всех форм обучения.

Текст печатается в авторской редакции

Подписано в печать 11.06.2024. Формат 60×84 1/16.

Усл. печ. л. 0,8 . Уч.- изд. л. 0,7. Тираж 100 экз. Заказ 480. Бесплатно.

Юго-Западный государственный университет.

305040, г. Курск, ул. 50 лет Октября, 94.

КЛАССИФИКАЦИИ ОБРАЗОВ ИНФОРМАТИВНЫХ ПРИЗНАКОВ НА ОСНОВЕ НЕЙРОННОЙ СЕТИ ПРЯМОГО РАСПРОСТРАНЕНИЯ

Цель работы: изучение модели нейрона и архитектуры однослойной искусственной нейронной сети (ИНС) типа персептрон; создание и исследование моделей ИНС в системе **MATLAB** с использованием графического инструмента пользователя **NNTOOL** из пакета **NeuralNetworkToolbox**.

Основные задачи лабораторного занятия (лабораторной работы):

- познакомиться с основными понятиями методологии нейрокомпьютеров для построения интеллектуальных систем;
- изучить классификацию методов на основе нейронной сети прямого распространения и этапы реализации этих методов;
- выделить способы построения и развертывания архитектуры рекуррентной нейронной сети прямого распространения;
- проанализировать проблемную ситуацию как систему, выявляя ее составляющие и связи между ними;
- выделить структурные компоненты исследуемой модели, критически оценить надежность источников информации.

Планируемые результаты обучения (формируемые знания, умения, навыки и компетенции):

Код и наименование индикатора достижения компетенции, закрепленного за дисциплиной:

ПК-6.1 Определяет набор инструментальных средств разработки и библиотек повторно используемых модулей;

ПК-6.2 Выбирает средства создания и учёта базы знаний и задач, сборки и непрерывной интеграции;

ПК-6.3 Формирует управленческие решения на основе результатов мониторинга функционирования инфраструктуры.

Необходимое материально-техническое оборудование и материалы:

1. Класс ПЭВМ - Athlon 64 X2-2.4; Cel 2.4, Cel 2.6, Cel 800.
2. Мультимедиа центр: ноутбук ASUS X50VL PMD T2330/14"/1024Мб/160Gb/ сум-ка/проектор inFocus IN24+ .
3. Экран мобильный Draper Diplomat 60x60
4. Доступ в сеть Интернет.

Шкала оценивания и критерии оценивания выполненной практической работы:

Форма контроля	Минимальный балл		Максимальный балл	
	алл	примечание	алл	примечание
1		3		5
Лабораторное занятие Распознавание образов в обучаемых системах		Выполнил но «не защитил»	2	Выполнил и «защитил»

План проведения практического занятия (лабораторной работы)

1. ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОГО ВЫПОЛНЕНИЯ

1. Построить нейронную сеть, которая производит классификацию информативных признаков на заданное количество классов.

2. Построить обучающую выборку, позволяющую правильно классифицировать заданную проверочную выборку. Координаты точек обучающей выборки должны быть подобраны геометрически.

3. Произвести обучение нейронной сети на составленной обучающей выборке.

4. Используя обученную нейронную сеть, произвести классификацию для каждого варианта проверочного множества. Поскольку номер класса для каждой точки может меняться при каждом новом обучении сети, номера классов в таблице приведены условно. Точки с разными номерами классов в таблице при классификации должны иметь разные номера классов, точки с одинаковыми – одинаковые.

5. Предоставить отчет. Содержание отчета:

- титульный лист;
- тема и цель работы;
- формальная постановка задачи классификации;
- основные теоретические сведения;
- задание, учитывая свой вариант;
- ход создания, обучения и тестирования персептронной однослойной ИНС с использованием графического инструмента пользователя **nntool**;
- выводы.

2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

2.1 Постановка задачи классификации

Пусть задано X – множество информативных признаков объектов (множество образов), Y – конечное множество номеров классов. Предполагается, что существует неизвестная *целевая зависимость* – отображение $f^* : X \xrightarrow{f} Y$, значения которой Также заданы значения конечной обучающей выборки $X^m = \{(x_1, y_1), \dots, (x_m, y_m)\}$.

Требуется построить алгоритм: $X \longrightarrow Y$, позволяющий классифицировать произвольный объект $x \in X$.

Основу каждой ИНС составляют относительно простые элементы (ячейки), имитирующие работу нейронов мозга, называемые *нейронами*. Далее под нейроном будет понимать искусственный нейрон, то есть ячейка ИНС.

Каждый нейрон характеризуется:

- *текущим состоянием* по аналогии с нервными клетками головного мозга, которые могут быть возбуждены или заторможены;
- группой *синапсов* – *однаправленных входных связей, соединенных с выходами других нейронов*;
- имеет *аксон* – выходную связь данного нейрона, с которой сигнал (возбуждения или торможения) поступает на синапсы следующих нейронов.

Общий вид модели искусственного нейрона приведен на рисунке 1. Каждый синапс характеризуется величиной синоптической связи или ее *весом* w_i , который по физическому смыслу эквивалентен электрической проводимости.

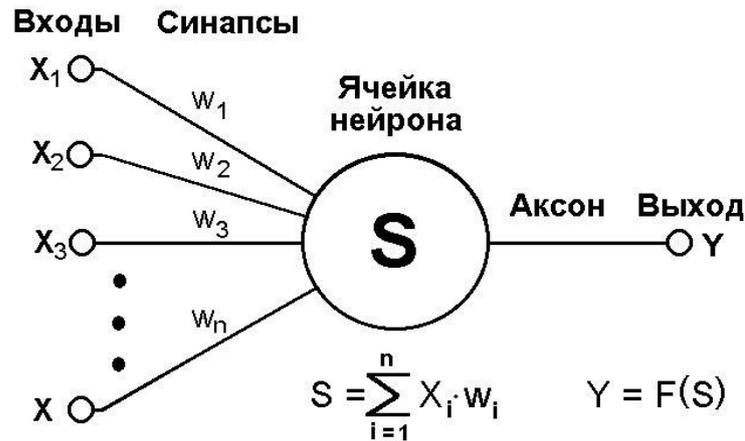


Рисунок 1. Модель искусственного нейрона

Текущее состояние нейрона определяется, как взвешенная сумма его входов:

$$S = \sum_{i=1}^n x_i \cdot w_i . \quad (1)$$

Выход нейрона представляет в виде функции состояния:

$$Y = f(S) . \quad (2)$$

Нелинейная функция f называется **функцией активации** и может иметь различный вид, проиллюстрированный на рисунке 2.

Одной из наиболее распространенных функций активации является нелинейная функция с насыщением, так называемая **логистическая функция**, или **сигмоид** (т.е. функция S -образного вида):

$$f(x) = \frac{1}{1 + e^{-\alpha x}} . \quad (3)$$

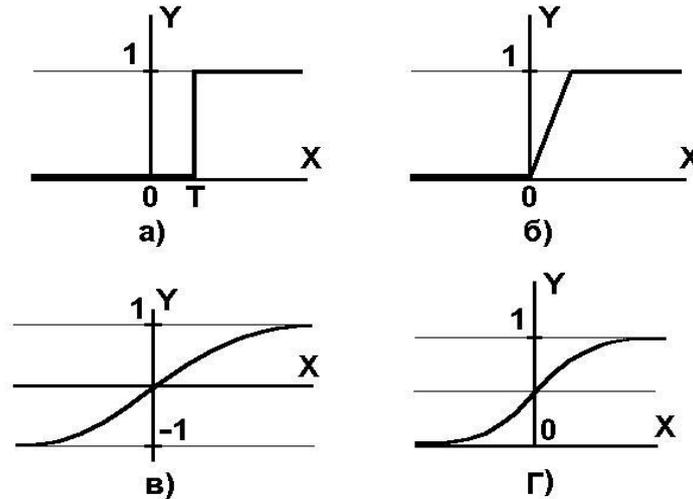


Рисунок 2. Виды функций активации: а) функция единичного скачка; б) линейный пороговая функция (гистерезис); в) сигмоидальная функция – гиперболический тангенс; г) логистическая сигмоидальная функция

Важным преимуществом и характерной особенностью ИНС является принцип параллельной обработки сигналов. Для решения сложных задач применяют достаточно большие группы нейронов, образующие слои, связанные между собой. Нейроны в сети могут быть связаны внутри отдельных слоев, так и между слоями. Существует множество способов организации искусственных нейронных сетей, которые могут содержать различное число слоев нейронов.

В зависимости от направления передачи сигнала эти связи могут быть *прямыми или обратными*.

Слой нейронов, принимающий информацию из внешней среды, называется входным слоем, а слой, передающий информацию во внешнюю среду, - выходным слоем. Остальные слои, если они имеются в сети, называются промежуточными, или *скрытыми*.

Следует отметить, что *обработка взаимодействия всех нейронов в сети выполняется послойно.*

В качестве примера рассмотрим ИНС типа однослойного персептрона, на входы которого подаются сигналы, проходящие по синапсам на три нейрона, и имеющие три выходных сигнала (рис.3):

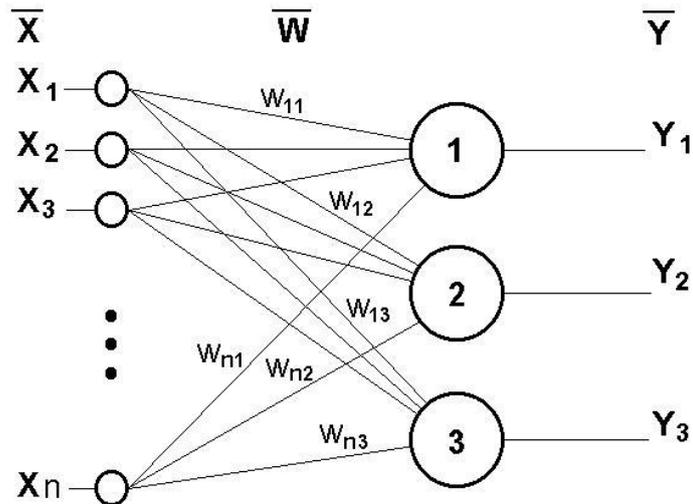


Рисунок 3. Однослойный персептрон

$$Y_j = f\left(\sum_{i=1}^n w_{ij} \cdot x_i\right), \quad j = 1, 2, 3 \quad (4)$$

Весовые коэффициенты синапсов одного слоя нейронов можно свести в матрицу W , каждый которой элемент w_{ij} задает величину i -ой синаптической связи j -ого нейрона.

Таким образом, процесс, происходящий в ИНС может быть записан в матричной форме:

$$Y = F(X \cdot W), \quad (5)$$

где X и Y – соответственно входной и выходной векторы, $F(XW)$ – функция активации, применяемая поэлементно к компонентам вектора (XW) .

Структура, или *архитектура* сети искусственных нейронов зависит от конкретной решаемой задачи. Сеть может быть однослойной без обратных связей или с обратными связями, двухслойной с прямыми связями, трехслойной с обратными связями и т. д. Сети с обратными связями называют часто *рекуррентными*.

Описание архитектуры искусственной нейронной сети должно включать:

- указания числа слоев и связей между ними;
- сведения о количестве нейронов в каждом слое;
- вид функций активации в каждом слое;
- наличие смещений для каждого слоя;

- наличие компонент входных, выходных и целевых векторов;
- в ряде случаев характеристики топологии слоев.

Теоретически число слоев и число нейронов в каждом слое может быть произвольным, однако фактически оно ограничено ресурсами компьютера. Чем сложнее ИНС, тем масштабнее задачи, которые она реализует.

Выбор структуры ИНС осуществляется в соответствии с особенностями и сложностью задачи. Для решения некоторых отдельных типов задач уже существуют оптимальные конфигурации. Если же задача не может быть сведена ни к одному из известных типов, разработчику приходится решать сложную проблему *синтеза новой конфигурации*.

Очевидно, что процесс функционирования ИНС, то есть сущность действий, которые она способна выполнять, зависит от величин синаптических связей, поэтому, задавшись определенной структурой ИНС, отвечающей какой-либо задаче, разработчик сети должен найти оптимальные значения всех переменных весовых коэффициентов (некоторые синаптические связи могут быть постоянными).

Этот этап называется обучением ИНС, и от того, насколько качественно он будет выполнен, зависит способность сети решать поставленные перед ней проблемы во время эксплуатации. На этапе обучения *кроме параметра качества подбора весов важную роль играет время обучения*. Как правило, эти два параметра связаны обратной зависимостью и их приходится выбирать на основе компромисса.

Обучение ИНС может вестись с учителем или без него. В первом случае сети предъявляются значения как входных, так и желательных выходных сигналов, и она по некоторому внутреннему алгоритму подстраивает веса своих синаптических связей. Во втором случае выходы ИНС формируются самостоятельно, а веса изменяются по алгоритму, учитывающему только входные и производные от них сигналы.

2. 2 Однослойный персептрон

Однослойный персептрон представляет простейшую ИНС, все нейроны которой имеют жесткую пороговую функцию активации. Веса и смещения нейронов в этой сети могут быть настроены таким образом, чтобы решить задачу классификации входных векторов.

Каждое значение элемента вектора входа персептрона умножено на соответствующий вес, и сумма полученных взвешенных элементов является входом функции активации. Если вход функции активации $n \geq 0$, то нейрон персептрона возвращает 1, если $n < 0$, то 0.

Функция активации с жесткими ограничениями придает персептрону способность классифицировать векторы входа, разделяя пространство входов на две области. Пространство входов делится на две области разделяющей линией. Эта линия перпендикулярна к вектору весов w и смещена на величину b . Векторы входа выше линии соответствуют положительному потенциалу нейрона, и, следовательно, выход персептрона для этих векторов будет равен 1; векторы входа ниже линии соответствуют выходу персептрона, равному 0.

При изменении значений смещения и весов граница линии изменяет свое положение.

Персептрон без смещения всегда формирует разделяющую линию, проходящую через начало координат; добавление смещения формирует линию, которая не проходит через начало координат.

В случае, когда размерность вектора входа превышает 2, т. е. входной вектор имеет более 2 элементов, разделяющей границей будет служить гиперплоскость. Работа сети сводится к классификации (обобщению) входных сигналов, принадлежащих n -мерному гиперпространству, по некоторому числу классов. С математической точки зрения это происходит путем разбиения гиперпространства гиперплоскостями (запись для случая однослойного персептрона)

$$\sum_{i=1}^n x_i \cdot w_{ij} = T_j, \quad j = \overline{1, m}. \quad (6)$$

Каждая полученная область является областью определения отдельного класса. Число таких классов для одной ИНС типа персептрон не превышает 2^m , где m – число выходов сети. Однако не

все классы могут быть разделимы ИНС.

Например, однослойный персептрон, состоящий из одного нейрона с двумя входами, не способен разделить плоскость (двумерное гиперпространство) на две полуплоскости так, чтобы осуществить классификацию входных сигналов по двум классам:

$$A \in \{(0,0), (1,1)\}$$

$$B \in \{(0,1), (1,0)\}$$

Уравнение сети для этого случая

$$x_1 \cdot w_1 + x_2 \cdot w_2 = T \quad (7)$$

является уравнением прямой (одномерной гиперплоскости), которая ни при каких условиях не может разделить плоскость так, чтобы точки из множества входных сигналов, принадлежащие разным классам, оказались по разные стороны от прямой (рис. 4).

Анализ данных, показывает, что разбиение на классы реализует логическую функцию исключающего ИЛИ для входных сигналов. Невозможность реализации однослойным персептроном этой функции получила название проблемы исключающего ИЛИ.

Функции, которые не реализуются однослойной сетью, называются *линейно неразделимыми*. Решение задач, подпадающих под это ограничение, заключается в применении 2-х и более слойных сетей или сетей с нелинейными синапсами, однако и тогда существует вероятность, что корректное разделение некоторых входных сигналов на классы невозможно.

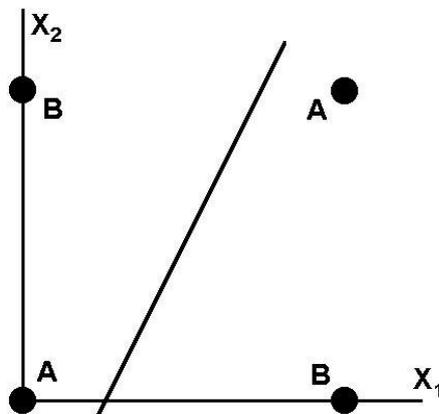


Рисунок 4. Визуальное представление работы нейронной сети

Обучение ИНС однослойного персептрона заключается в следующих действиях. Проинициализировать элементы весовой матрицы (обычно небольшими случайными значениями).

1. Подать на входы один из входных векторов, которые сеть должна научиться различать, и вычислить ее выход Y .

2. Если выход правильный, перейти на шаг 4.

Иначе вычислить разницу между идеальным и полученным значениями выхода:

$$\delta = Y_1 - Y \quad (8)$$

Модифицировать веса в соответствии с формулой:

$$w_{ij}(t+1) = w_{ij}(t) + v \cdot \delta \cdot x_i, \quad (9)$$

где t и $t+1$ – номера соответственно текущей и следующей итераций;

v – коэффициент скорости обучения, $0 < v < 1$;

i – номер входа;

j – номер нейрона в слое.

Очевидно, что если Y_1 меньше Y весовые коэффициенты будут увеличены и тем самым уменьшат ошибку. В противном случае они будут уменьшены, и Y тоже уменьшится, приближаясь к Y_1 .

4. Цикл с шага 2, пока сеть не перестанет ошибаться.

На втором шаге на разных итерациях поочередно в случайном порядке предъявляются все возможные входные вектора. К сожалению, нельзя заранее определить число итераций, которые потребуется выполнить, а в некоторых случаях и гарантировать полный успех.

2.3 Пример выполнения работы

1. В окне среды **MATLAB “CommandWindow”** введите команду **nntool**. Откроется окно графического инструмента пользователя **nntool**, проиллюстрированное на рисунке 5.

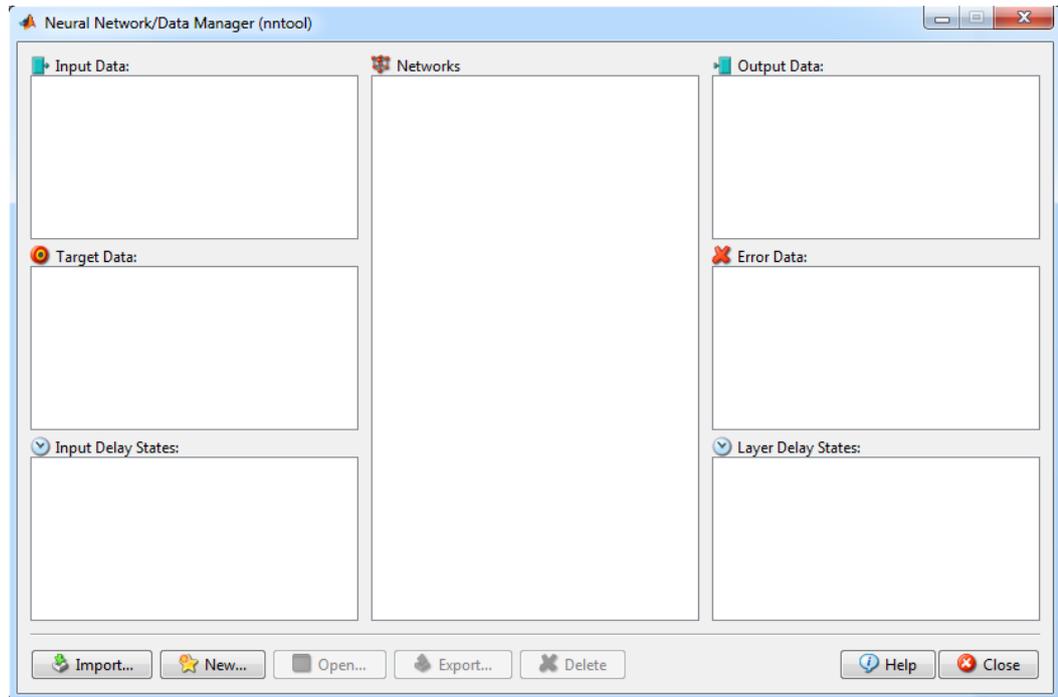


Рисунок 5. Окно графического инструмента пользователя nntool

2. Нажмите кнопку **“New”** в окне графического инструмента пользователя **nntool**. Откроется окно, показанное на рисунке 6, для создания новой ИНС, задания входных и целевых данных и обучения ИНС.

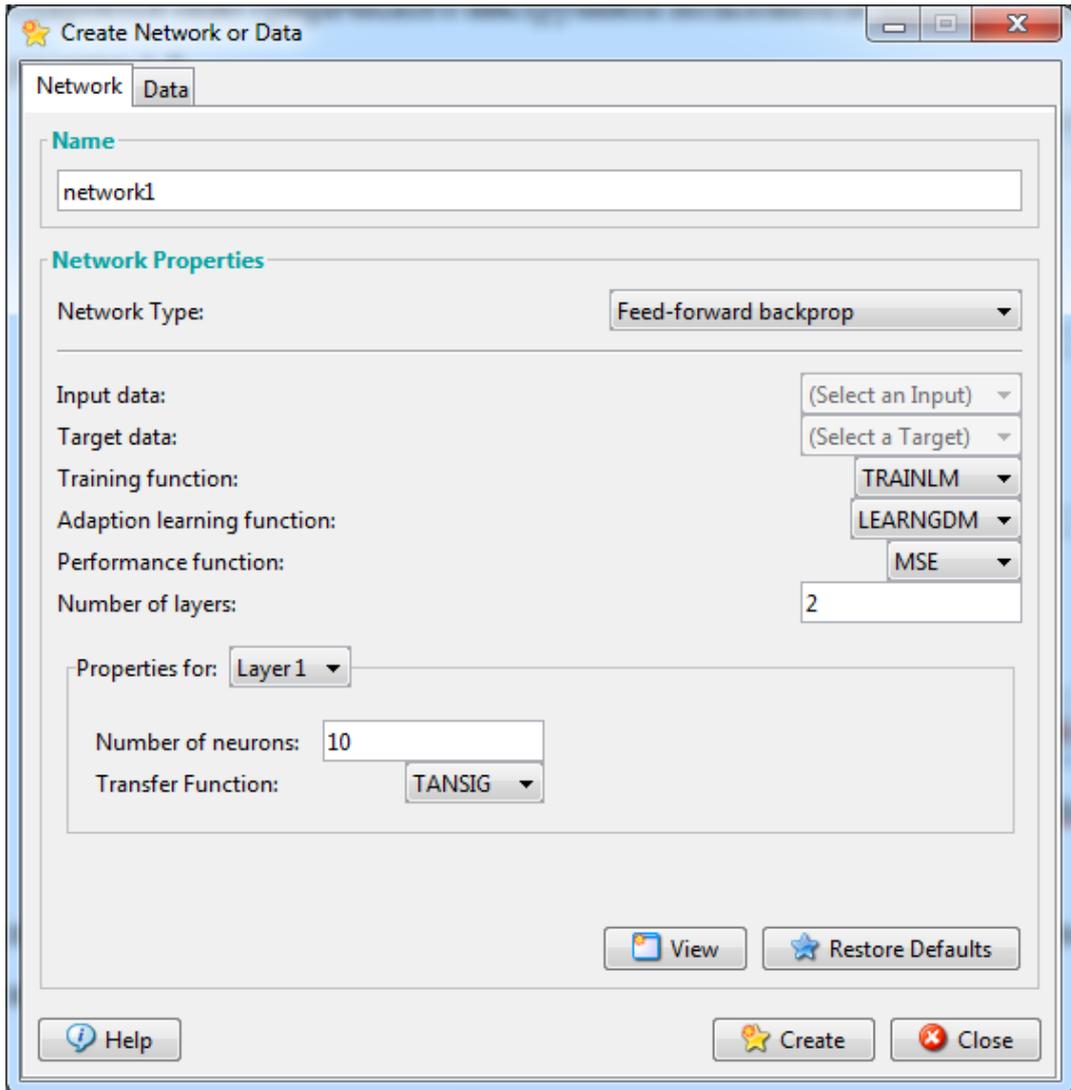


Рисунок 6. Окно создания новой ИНС, задания входных и целевых данных для обучения ИНС

3. В поле “**Name**” окна создания новой ИНС, задания входных и целевых данных для обучения ИНС введите наименование ИНС, например “**nnclass**”. В выпадающем списке “**NetworkType**” выберите позицию “**Perceptron**”. В результате текущее окно примет вид, показанный на рисунке 7.

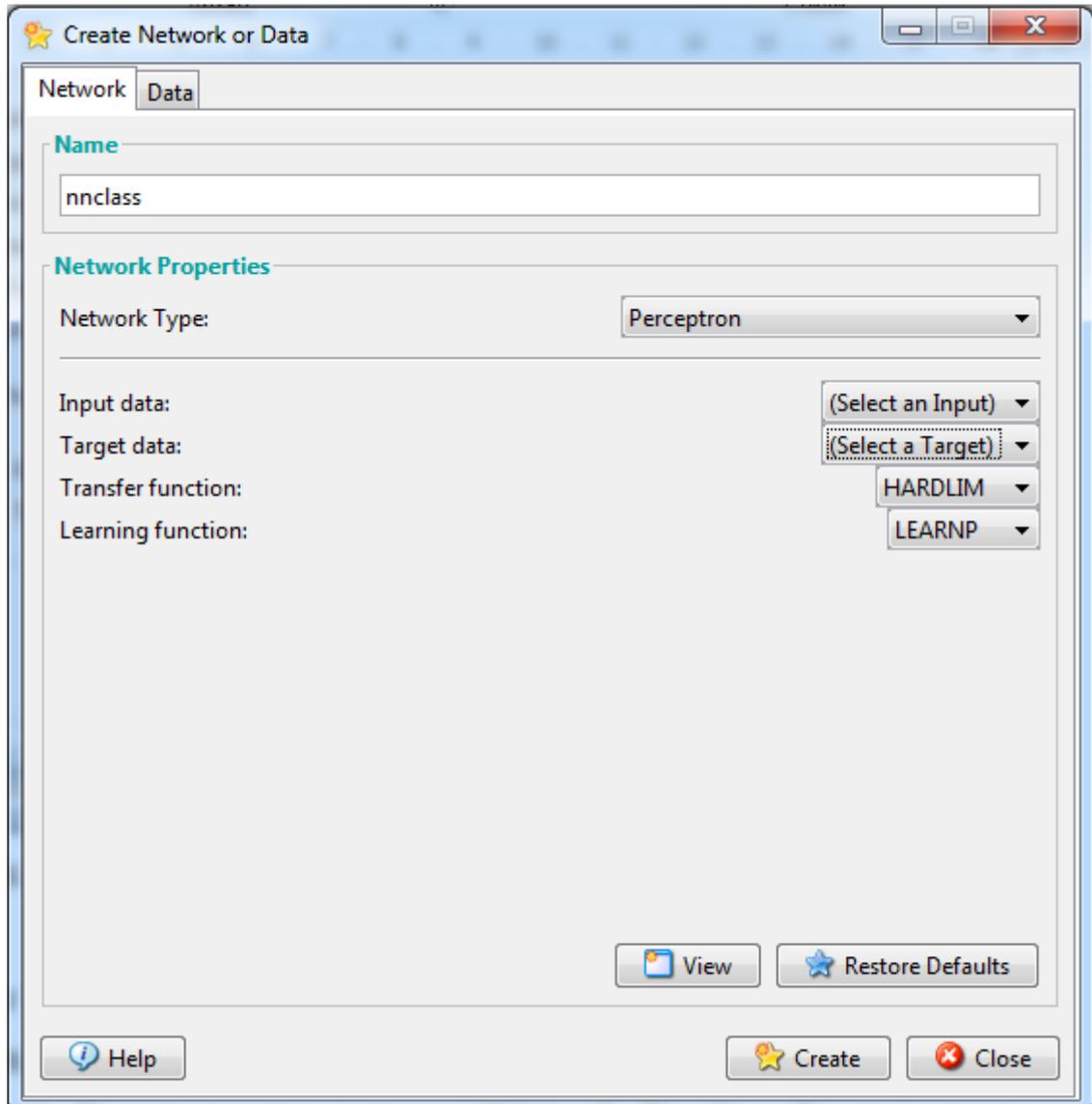


Рисунок 7. Окно создания новой ИНС, задания входных и целевых данных для обучения ИНС, с выбранным типом ИНС “Perceptron”

4. Перейдите на вкладку “**Data**” окна создания новой ИНС, задания входных и целевых данных для обучения ИНС, с выбранным типом ИНС “**Perceptron**” (рисунок 8).

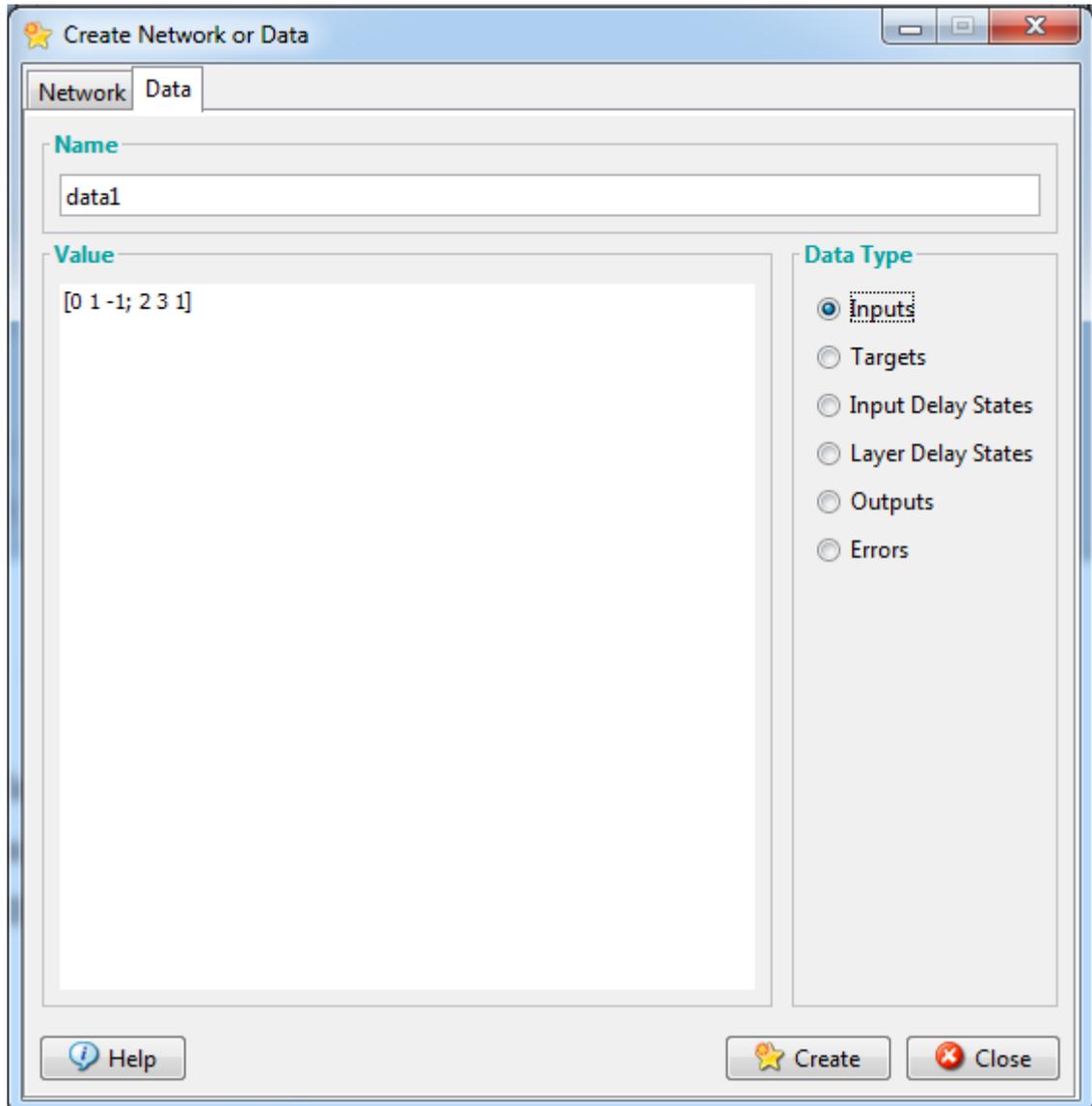


Рисунок 8. Открытая вкладка **“Data”** окна создания новой ИНС, задания входных и целевых данных для обучения ИНС, с выбранным типом ИНС **“Perceptron”**

5. В поле **“Name”** открытой вкладки **“Data”** окна создания новой ИНС, задания входных и целевых данных для обучения ИНС, с выбранным типом ИНС **“Perceptron”** введите Р. В поле **“Value”** задайте входные данные для обучения ИНС, например $[-1 \ 0 \ -1 \ 1.5 \ 1 \ 2; -0.5 \ -1 \ 1 \ 1.5 \ 1.5 \ 1]$ (рисунок 9).

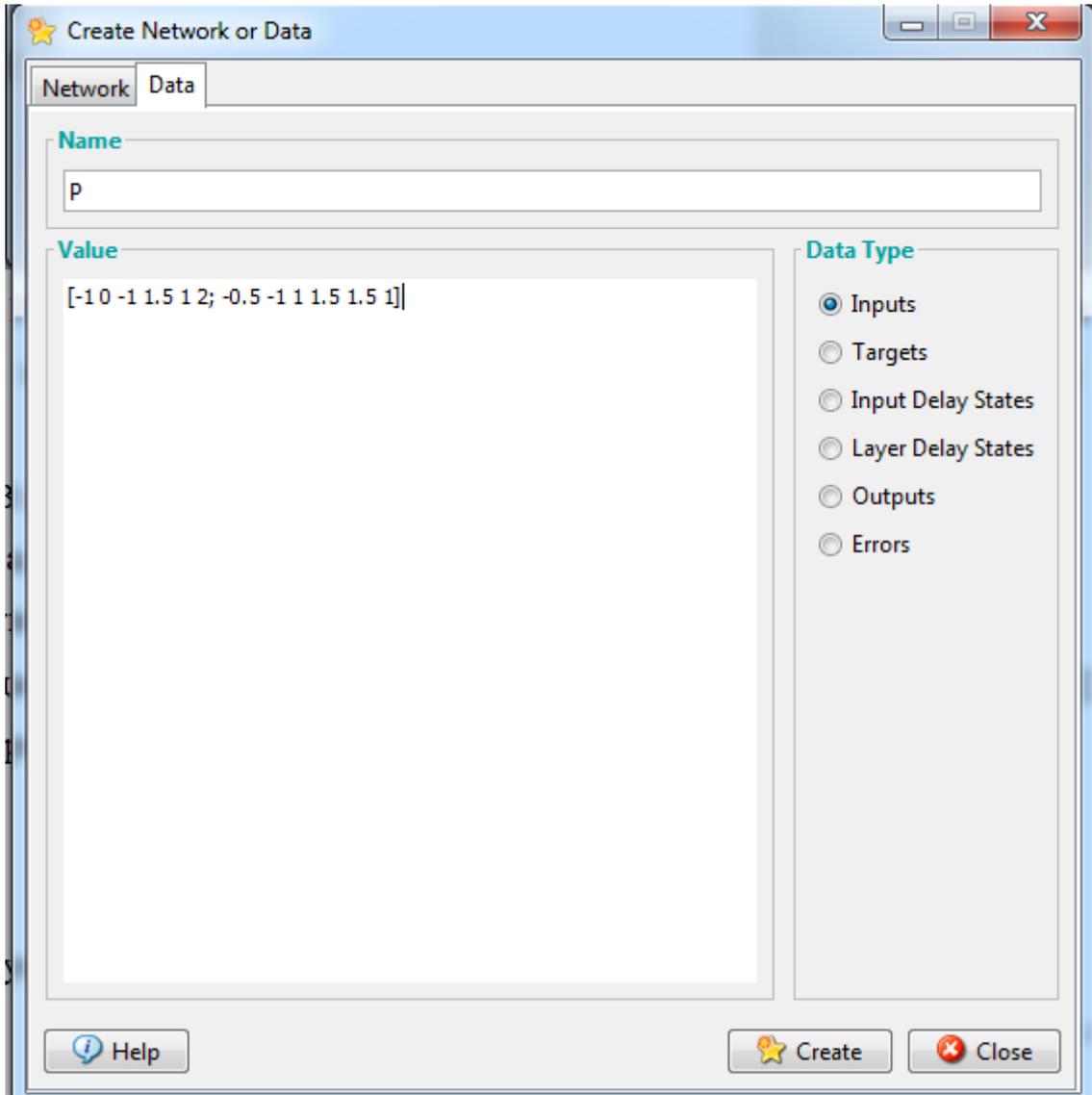


Рисунок 9. Задание имени входных данных и самих входных данных

6. Нажмите кнопку **“Create”**.

7. Появится окно, сообщающее о том, что входные данные P добавлены в менеджер данных ИНС (рисунок 10).

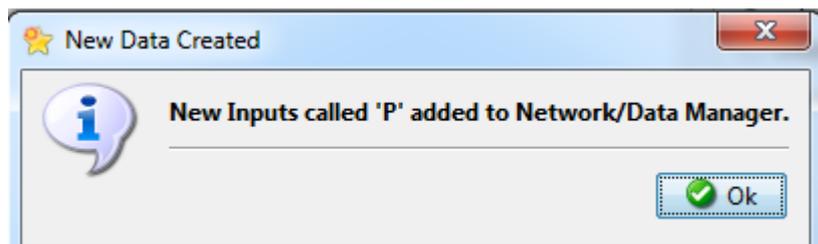


Рисунок 10. Окно с сообщением об успешном добавлении входных данных P в менеджер данных ИНС

8. Нажмите кнопку “Ok”.

9. В окне, представленном на рисунке 9 в разделе “DataType” выберите позицию “Targets”, в поле “Name” введите T, а в поле “Value” введите целевые данные для обучения ИНС, например [0 0 0 1 1 1] (рис. 11).

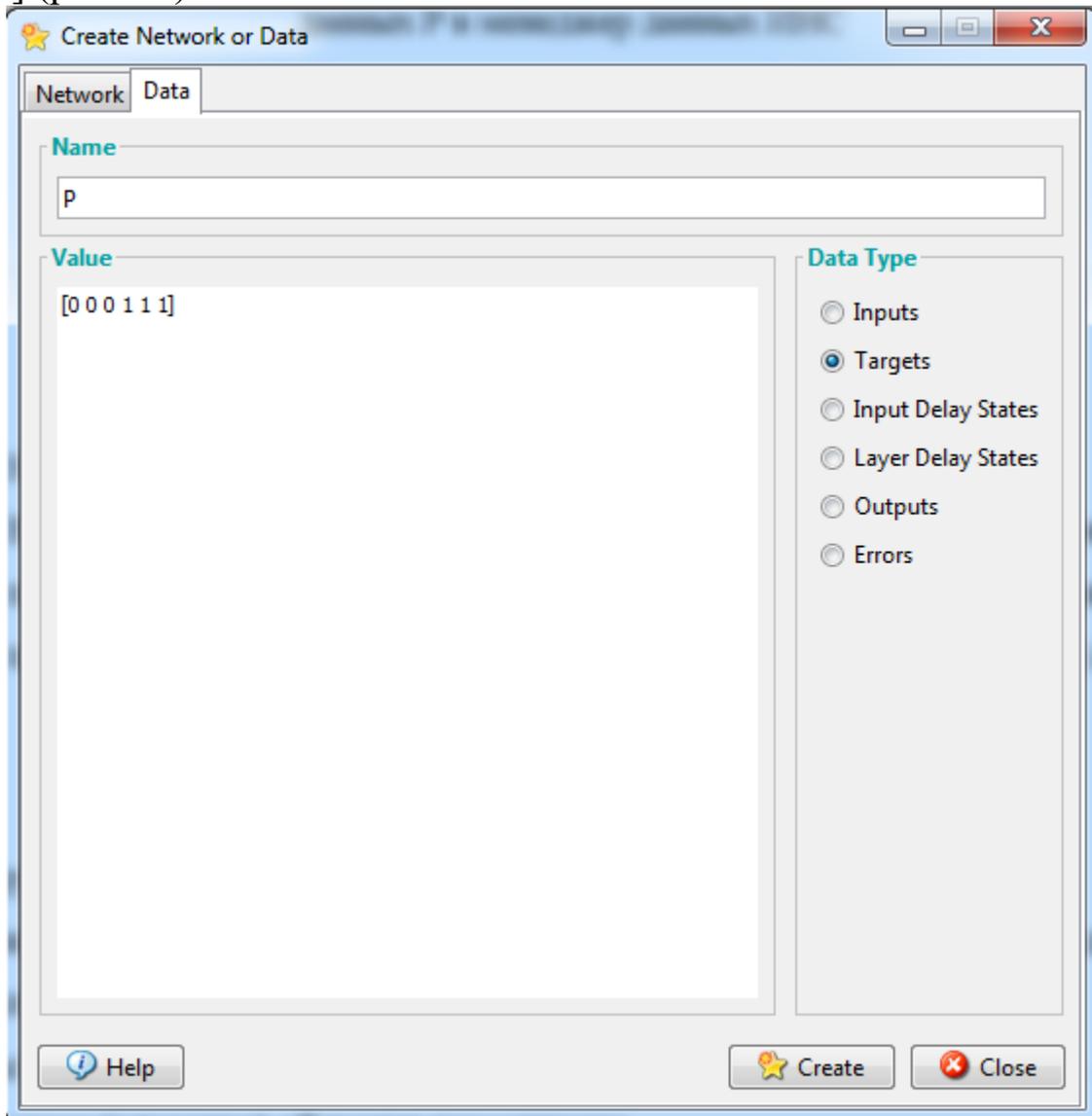


Рисунок 11. Задание имени целевых данные и самих целевых данных

10. Нажмите кнопку “Create”.

11. Появится окно, сообщающее о том, что целевые данные T добавлены в менеджер данных ИНС (рисунок 12).

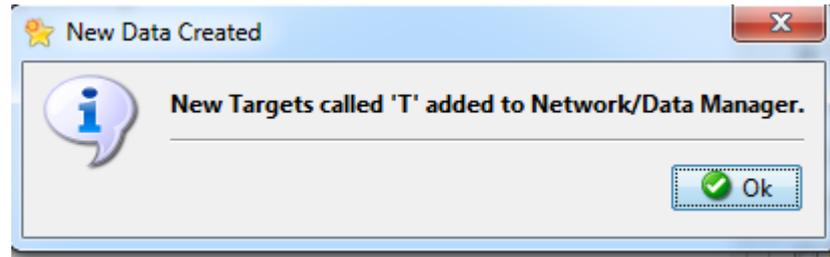


Рисунок 12. Окно с сообщением об успешном добавлении целевых данных Т в менеджер данных ИНС

12. Нажмите кнопку “Ok”.
13. Откройте вкладку “Network”.
14. В выпадающем списке “Inputdata” выберите позицию Р, а в выпадающем списке “Targetdata” позицию Т (рисунок 13).

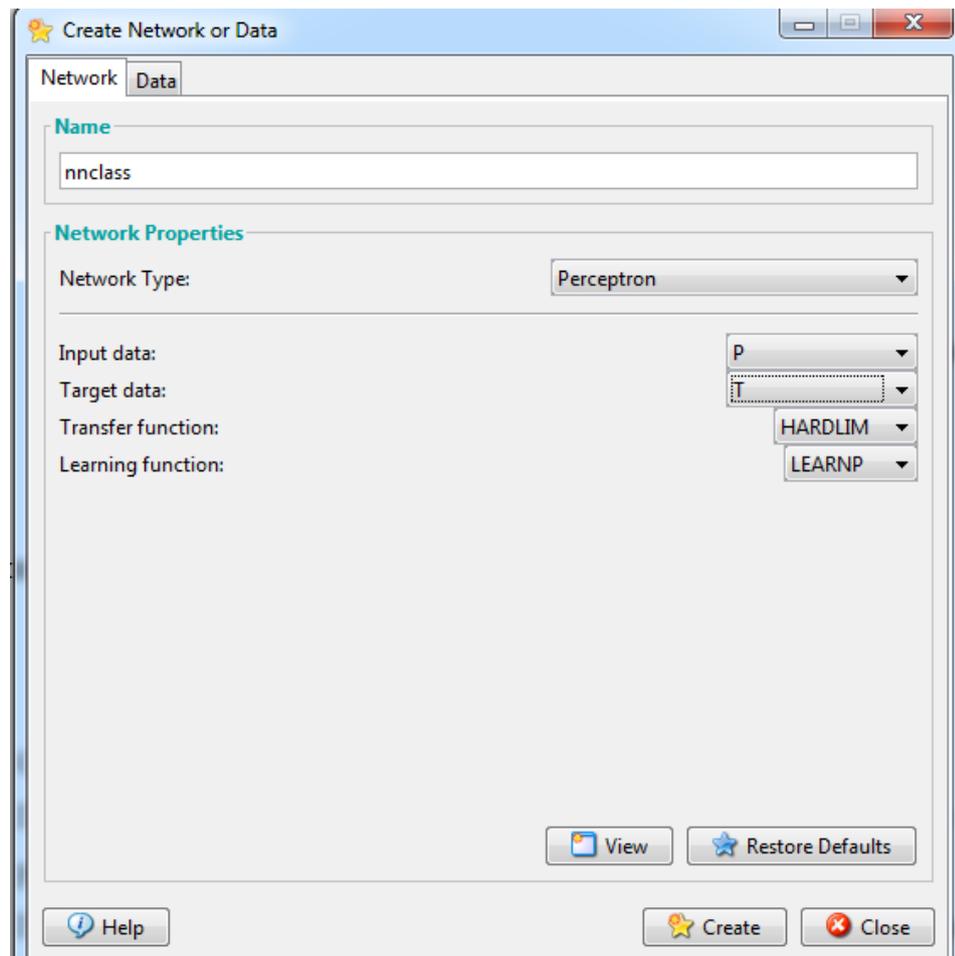


Рисунок 13. Окно создания новой ИНС, задания входных и целевых данных для обучения ИНС, с выбранным типом ИНС “Perceptron” и входными и целевыми данными

15. Нажмите кнопку “**Create**”.

16. Появится окно, сообщающее о том, что ИНС **nnclass** успешно добавлена в менеджер данных ИНС (рис. 14).

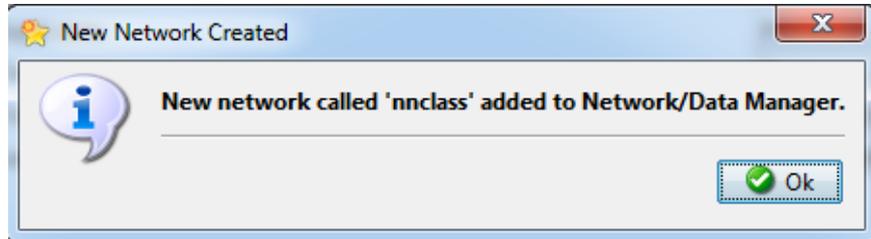


Рисунок 14. Окно с сообщением об успешном добавлении ИНС **nnclass** в менеджер данных ИНС

17. Нажмите кнопку “**Ok**”.

18. В окне создания новой ИНС, задания входных и целевых данных для обучения ИНС, с выбранным типом ИНС “**Perceptron**” и входными и целевыми данными (рис. 13) нажмите кнопку “**Close**”.

19. В окне графического инструмента пользователя **ntool** (рисунок 5) в разделе “**Networks**” выберите ИНС **nnclass** (рисунок 15), дважды кликнув курсором мыши по позиции “**nnclass**”.

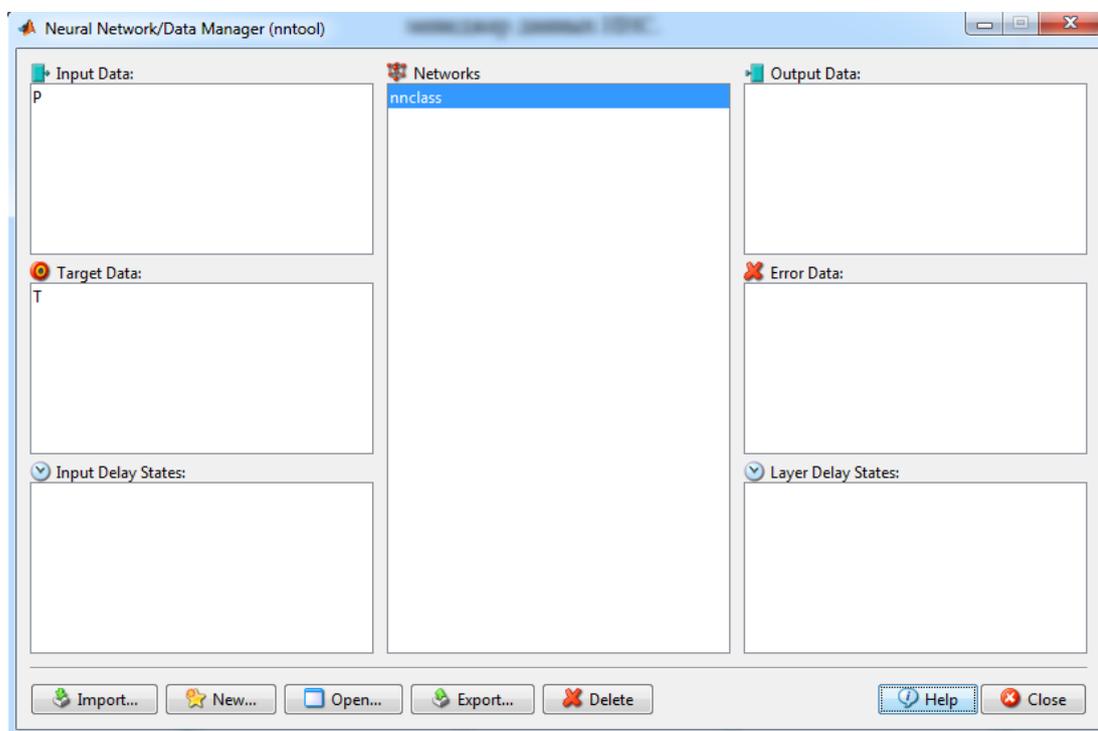


Рисунок 15. Выбор ИНС **nnclass** в окне графического инструмента пользователя **ntool**

20. В открывшемся окне откройте вкладку **“Train”**, в выпадающих списках **“Inputs”** и **“Targets”** выберите позиции **P** и **T** соответственно (рис. 16).

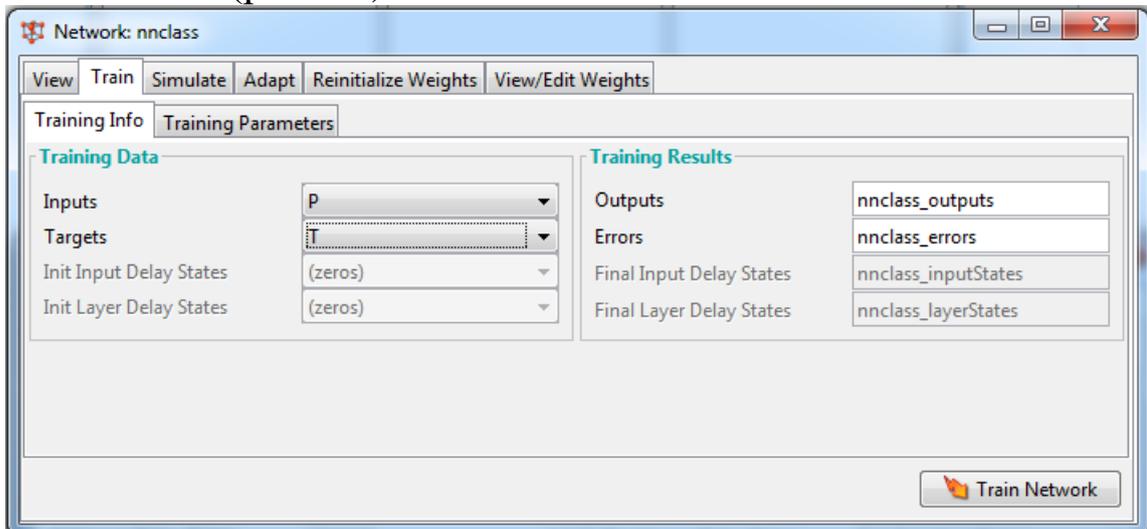


Рисунок 16. Выбор позиций в выпадающих списках **“Inputs”** и **“Targets”**

21. Нажмите кнопку **“TrainNetwork”**. Откроется окно обучения ИНС (рис. 17) и дождитесь окончания обучения. Затем его закройте.

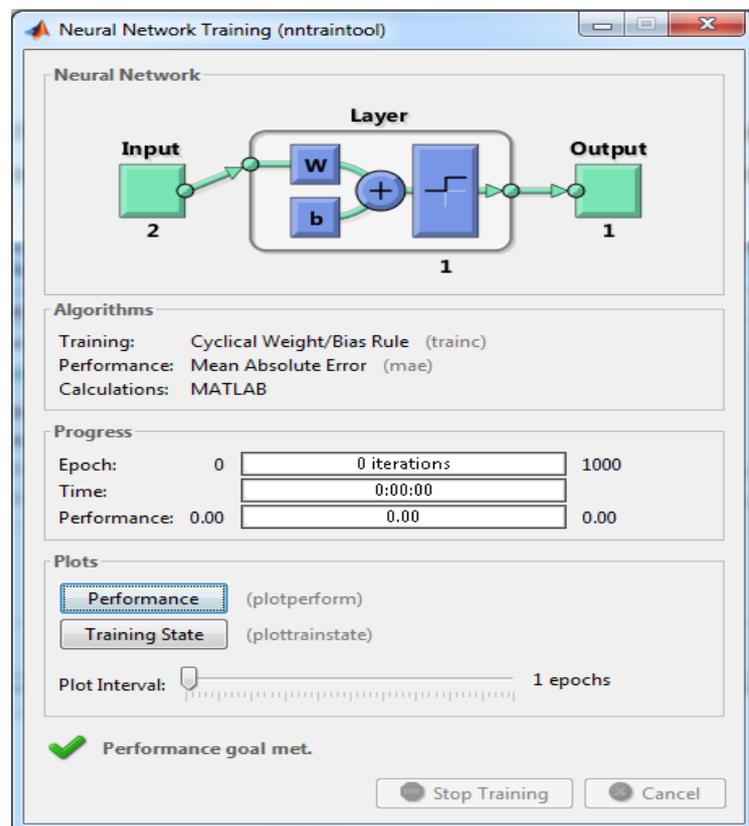


Рисунок 17. Окно обучения ИНС

22. Откройте вкладку “**Simulate**” и в выпадающем списке “**Inputs**” выберите позицию P (рисунок 18). Нажмите кнопку “**SimulateNetwork**”.

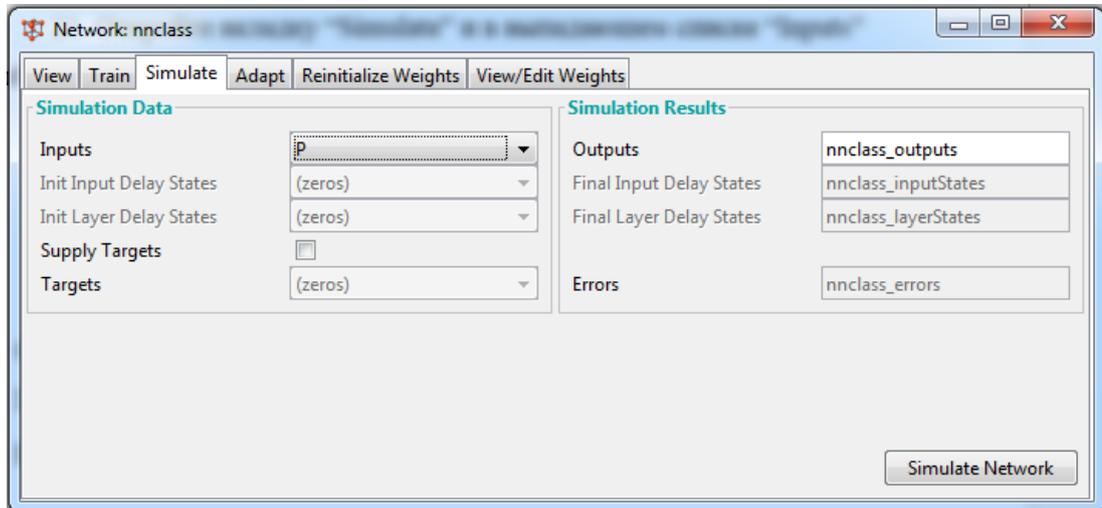


Рисунок 18. Вкладка “**Simulate**”

23. Появится окно, с информацией о том, что симуляция ИНС завершена и результат храниться в менеджере данных ИНС (рисунок 19).

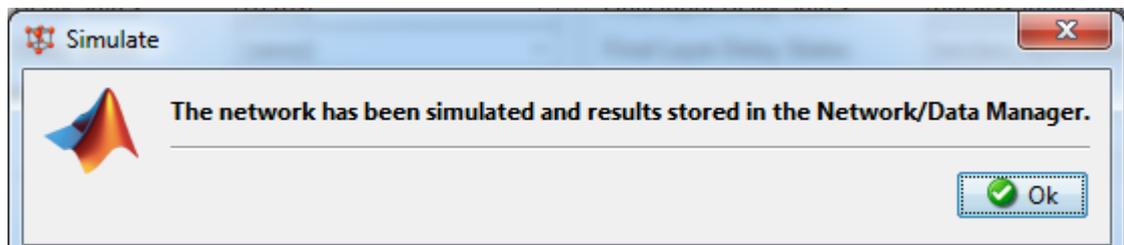


Рисунок 19. Окно, содержащее информацию о завершении прогона сети, результат храниться в менеджере данных ИНС

24. В разделе “**OutputData**” в окне графического инструмента пользователя **nntool** (рисунок 20) выберите позицию “**nnclass_outputs**” двойным кликом курсора мыши. Откроется окно с результатом симуляции ИНС (рис. 21).

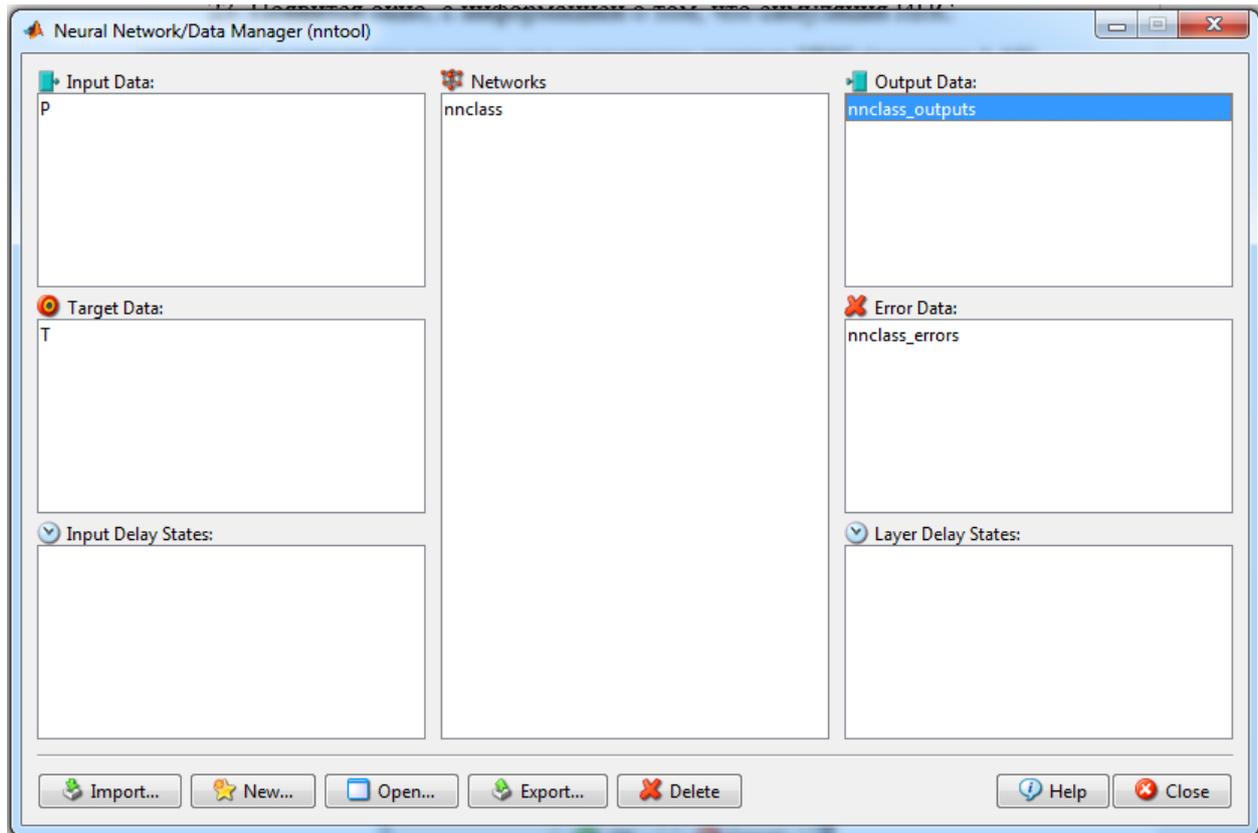


Рисунок 20. Выбор позиции “**nnclass_outputs**” в разделе “OutputData” в окне графического инструмента пользователя **nntool**

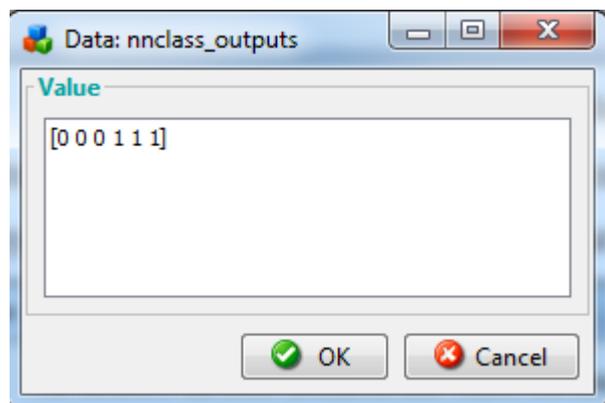


Рисунок 21. Окно, содержащее информацию о результате прогона (симуляции) ИНС

25. Сравните результат симуляции, показанный в окне на рисунке 20, с целевыми данными T, на которых производилось обучение ИНС.

Таблица 1.

Индивидуальные варианты заданий

№ варианта	Количество классов для классификации	Координаты точек проверочного множества и номер класса, к которому принадлежит каждая точка
1	3	[0;0]-1; [1;1]-2; [-1;-1]-3;
2	4	[2;1]-1; [1;0]-2; [0;-1]-3;
3	3	[0;0]-1; [1;1]-1; [-1;-1]-3;
4	4	[1;0]-1; [-1;1]-2; [-1;-1]-2;
5	3	[0;0]-1; [1;1]-2; [-1;-1]-3;
6	4	[0;0]-1; [1;1]-1; [-1;-1]-1;
7	3	[-1;0]-1; [-1;1]-2; [-1;-1]-3;
8	4	[0;1]-1; [1;-1]-2; [-1;-1]-3;
9	3	[2;0]-1; [1;1]-2; [-1;1]-2;
10	4	[0;0]-1; [1;1]-2; [-1;-1]-1;
11	3	[0;0]-1; [1;1]-1; [-1;-1]-2;
12	4	[0;0]-1; [1;1]-2; [-1;-1]-3;
13	3	[0;0]-1; [1;1]-1; [-1;-1]-1;
14	4	[0;1]-1; [1;-1]-2; [-1;-1]-3;

3. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Записать вид структуры (модель) нейрона типа однослойного персептрона.
2. Сформулируйте правило нахождения количества нейронов в персептроне для распознавания заданного числа классов.
3. Как осуществляется построение линий классификации персептрона на основании его весов?
4. В чем заключается алгоритм обучения персептрона?

РЕКОМЕНДАТЕЛЬНЫЙ СПИСОК ЛИТЕРАТУРЫ

1. Борисовский, С.А. Нейросетевые модели с иерархическим пространством информативных признаков для сегментации плохоструктурированных изображений/ С.А. Борисовский, А.Н. Брежнева, Р.А. Томакова// Биомедицинская радиоэлектроника, 2010, - № 2. – С. 49-53.

2. Дж Ту, Гонсалес, Р. Принципы распознавания образов / Р. Гонсалес, Дж Ту – М.: Мир, 1978. – 411 с.

3. Кореневский, Н.А. Нейронные сети с макрослоями для классификации и прогнозирования патологий сетчатки глаза/ Н.А. Кореневский, Р.А. Томакова, С.П. Серегин, А.Ф. Рыбочкин// Медицинская техника, 2013. - № 4. – С. 16-18.

4. Томакова Р.А. Теоретические основы и методы обработки и анализа микроскопических изображений биоматериалов: монография / Р.А. Томакова, С.Г. Емельянов, С.А. Филист. – Курск, Юго-Зап. гос. ун-т, 2011. –202с.

5. Томакова Р.А. Интеллектуальные технологии сегментации и классификации биомедицинских изображений: монография / Р.А. Томакова, С.Г. Емельянов, С.А. Филист. – Курск, Юго-Зап. гос. ун-т, 2012. – 222с.

6. Томакова, Р.А. Структурно-функциональные решения нечетких нейронных сетей для интеллектуальных систем анализа разнотипных признаков/ Р.А. Томакова, С.А. Филист, В.В. Жилин, С.А. Горбатенко//Фундаментальные и прикладные проблемы техники и технологии, 2011. - № 1. –С. 85-91.

7. Томакова, Р.А. Универсальные сетевые модели для задач классификации биомедицинских данных/ Р.А. Томакова, С.А. Филист, Яа Зар До// Известия Юго-Западного государственного университета, 2012. - № 4-2(43). – С.44-50.

8. Томакова, Р.А. Метод обработки и анализа сложноструктурируемых изображений на основе встроенных функций среды MATLAB / Р.А. Томакова, С.А. Филист //Вестник Читинского государственного университета. – 2012. – № 1 (80). – С. 3-9.

9. Томакова, Р.А. Нейросетевые модели принятия решений для диагностики заболеваний легких на основе анализа флюорограмм грудной клетки/ Р.А. Томакова, М.В. Дюдин, М.В. Томаков// Биомедицинская радиоэлектроника. 2014. №9. С. 12-15.

10. Хайкин, С. Нейронные сети: полный курс / С. Хайкин. – М.: ООО «И.Д. Вильямс», 2006. – 1104с.