

Министерство образования и науки Украины
ХАРЬКОВСКИЙ НАЦИОНАЛЬНЫЙ АВТОМОБИЛЬНО-
ДОРОЖНЫЙ УНИВЕРСИТЕТ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

к выполнению контрольных работ по курсу "Нейронная мате-
матика"

2013

Министерство образования и науки Украины
ХАРЬКОВСКИЙ НАЦИОНАЛЬНЫЙ АВТОМОБИЛЬНО-
ДОРОЖНЫЙ УНИВЕРСИТЕТ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

к выполнению контрольных работ по курсу "Нейронная мате-
матика"

Утверждено методическим со-
ветом университета
протокол № _____ от
« _____ » _____ 20 ____ г.

Харьков – 2013

Составители: А.А.Тропина

Кафедра прикладной математики

ОБЩИЕ ПОЛОЖЕНИЯ

Методические указания предназначены для студентов заочной формы обучения инженерных специальностей технических высших учебных заведений и содержат описание и методические указания к контрольным работам, которые выполняются при изучении курса «Нейронная математика».

При выполнении контрольных работ (в объеме трех задач) студенты должны показать практическое применение знаний, полученных при изучении теоретических основ курса «Нейронная математика». Работа выполняется в соответствии с данными методических указаний. Часть работы выполняется на ЭВМ с использованием языка программирования системы **MATLAB**.

ЗАДАЧА №1 «МОДЕЛИ НЕЙРОННЫХ СЕТЕЙ И ЛОГИЧЕСКИЕ ФУНКЦИИ»

Цель работы – понять принцип действия однослойного и двухслойного персептрона. Изучив данную тему, студент должен:

знать:

- принцип действия нейронных сетей;
- что такое персептрон;

уметь:

- осуществлять математическую постановку исследуемых задач;

владеть:

- навыком построения таблиц истинности для булевых функций «NOT», «AND», «OR», «XOR».

- навыками построения однослойного и двухслойного персептронов.

- навыками обучения однослойного и двухслойного персептронов логическим операциям.

Теоретический минимум

Американские ученые У.Мак-Каллок и В.Питтс предложили математическую модель нейрона мозга человека, назвав ее математическим нейроном. Так же как и биологический нейрон, математический нейрон имеет несколько входов и один выход, может существовать в возбужденном и невозбужденном состояниях, причем переход в возбужденное состояние зависит от величины поступающих к нему сигналов и сил синаптических связей. Таким образом, математический нейрон весьма правдоподобно имитирует структуру и свойства своего прототипа — биологического нейрона мозга. На этом основании У.Мак-Каллок и В.Питтс предположили, что если математические нейроны связать между собой проводниками, имитирующими нервные волокна, то такой искусственный мозг будет способен решать интеллектуальные задачи, подобно тому, как это делает естественный человеческий мозг. Идея Мак-Каллока – Питтса была воплощена в жизнь в 1958 г. американским ученым Фрэнком Розенблаттом, также считающимся основателем нейроматематики. Он создал первую компьютерную программу для первой нейронной сети, которая была названа персептроном, от англ. слова perception — осознание.

Персептрон Розенблатта – простейшая модель нейронной сети, состоящая из одного нейрона. Единичный нейрон является основой адаптивного фильтра для обработки сигналов, поэтому работа однослойного персептрона лежит в основе метода адаптивной фильтрации. Математическая модель персептрона построена по аналогии с моделью биологического нейрона мозга. Эта модель представлена на рис.1. В ее основу входят входные сигналы, число которых равно n ($x_1, x_2, x_3, \dots, x_n$), а мощность определяется синаптическими весами ω_{kj} , где k -номер нейрона, j -номер синапса. При этом синаптический вес может принимать положительные и отрицательные значения, в отличие от синаптического веса биологического нейрона. Математическая модель также включает в себя сумматор, где все входные сигналы, поступающие в нейрон, суммируются в соответствии с синаптическими весами, формируя выход линейного сумматора u_k

$$u_k = \sum_{j=1}^n \omega_{kj} x_j \quad (1)$$

В модель нейрона, представленную на рис.1, дополнительно включен пороговый элемент b_k , который увеличивает или уменьшает входной сигнал, подаваемый на функцию активации $\varphi(u_k + b_k)$. После этого происходит формирование выходного сигнала нейрона y_k с учетом функции активации, которая масштабирует полученный потенциал активации (другое название - индуцированное локальное поле нейрона) $v_k = u_k + b_k$ в заданных рамках, ограничивая амплитуду выходного сигнала нейрона в интервале $[0,1]$ или $[-1,1]$.

Для модели нейронной сети, состоящей из одного нейрона, т.е. для однослойного персептрона, индекс k можно опустить. В роли функции активации для однослойного персептрона обычно выступает пороговая функция вида единичного скачка или функция Хевисайда (рис.2)

$$\varphi(v) = \begin{cases} 1 & v \geq 0 \\ 0 & v < 0 \end{cases}. \quad (2)$$

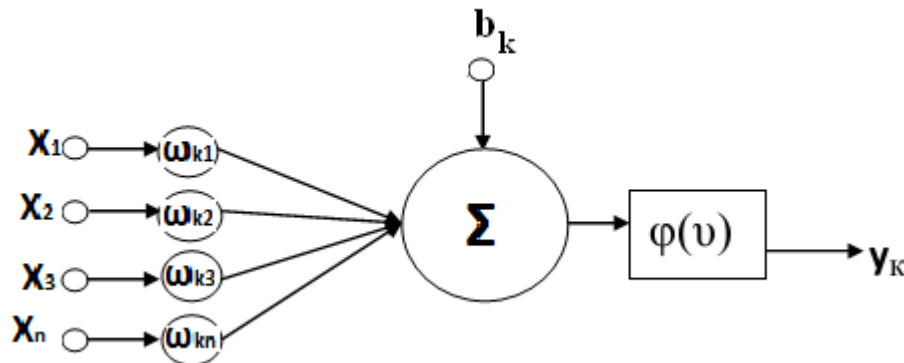


Рисунок 1. Схема нейрона

Соответственно выходной сигнал нейрона с такой функцией активации может принимать следующие значения

$$y(v) = \begin{cases} 1 & v \geq 0 \\ 0 & v < 0 \end{cases}. \quad (3)$$

Эту математическую модель однослойного персептрона в литературе называют моделью Мак-Каллока-Питтса. Согласно уравнениям модели (1)-(3) выходной сигнал нейрона принимает значение 1, если индуцированное локальное поле этого нейрона не отрицательно, т.е. входные сигналы достаточно интенсивны. В таком случае говорят, что нейрон возбужден. С точки зрения биологического нейрона весовые коэффициенты ω_j , имитируют электропроводность нервных волокон – силу синаптических связей между нейронами. Чем эти силы выше, тем больше вероятность перехода нейрона в возбуждённое состояние.

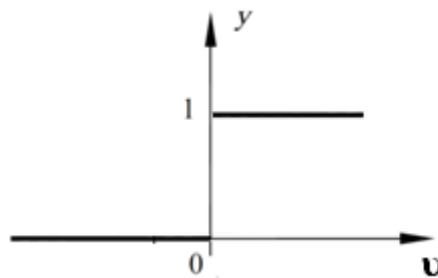


Рисунок 2. Функция активации единичного скачка

С помощью модели однослойного персептрона можно моделировать различные логические функции, например, функцию логического отрицания «NOT». Таблица истинности этой логической функции имеет вид:

Таблица 1

Таблица истинности логической функции «NOT»

x	y
0	1
1	0

Используя данные табл. 1 и формулы (1)-(3) нетрудно убедиться, что однослойный персептрон с одним входом моделирует функцию «NOT» при задании $\omega=-1$ и $b=0$ (Рис. 3).

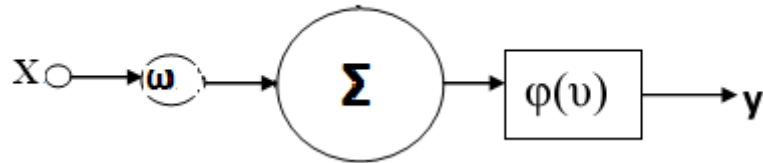


Рисунок 3. Однослойный персептрон, моделирующий логическую функцию «NOT»

Задачи, которые с помощью однослойного персептрона решены быть не могут, называют линейно неразделимыми задачами.

Нелинейное преобразование зачастую можно обеспечить путем специальной обработки входных сигналов. Именно по такому принципу работает квадратичный нейрон, выходной сигнал которого рассчитывается по следующей формуле

$$y_k = \varphi \left(\sum_{j=1}^n \omega_{kj} x_j + b_k + \sum_{l=1}^n \sum_{m=1}^n \omega_{klm} x_l x_m \right). \quad (4)$$

В качестве функции активации в такой модели выступает функция-выпрямитель вида

$$\varphi(v) = \begin{cases} v & v \geq 0 \\ 0 & v < 0 \end{cases}.$$

Обучение нейронных сетей

Обучение нейронной сети – это процесс, в котором свободные параметры нейронной сети меняются таким образом, что бы сеть ответила на внешний сигнал соответствующим образом. Существует пять моделей обучения: на основе коррекции ошибок, с использованием памяти, Хеббовское обучение, конкурентное обучение и обучение методом Больцмана. Обучение, основанное на коррекции ошибок, реализует метод оптимальной фильтрации. Обучение на основе памяти предполагает явное использование обучающих данных. Метод Хебба и конкурентный подход к обучению основаны на принципах нейробиологии. Метод Больцмана базируется на принципах статистической механики. Наиболее распространенным ме-

тодом обучения является метод обратного распространения ошибки.

Известно, что среди различных структур нейронных сетей одной из наиболее распространенных является многослойная структура, в которой каждый нейрон произвольного слоя связан со всеми аксонами нейронов предыдущего слоя или, в случае первого слоя, со всеми входами нейронной сети. Когда в сети только один слой, алгоритм ее обучения довольно очевиден, так как правильные выходные состояния нейронов единственного слоя заведомо известны, и подстройка синаптических связей идет в направлении, минимизирующем ошибку на выходе сети. По этому принципу строится, например, алгоритм обучения однослойного перцептрона. В многослойных же сетях оптимальные выходные значения нейронов всех слоев, кроме последнего, как правило, не известны, и двух или более слойный перцептрон уже невозможно обучить, руководствуясь только величинами ошибок на выходах нейронных сетей. Один из вариантов решения этой проблемы – разработка наборов выходных сигналов, соответствующих входным, для каждого слоя нейронной сети, что, конечно, является очень трудоемкой операцией и не всегда осуществимо. Вторым вариантом – динамическая подстройка весовых коэффициентов синапсов, в ходе которой выбираются, как правило, наиболее слабые связи и изменяются на малую величину в ту или иную сторону, а сохраняются только те изменения, которые повлекли уменьшение ошибки на выходе всей сети. И, наконец, третий, более приемлемый вариант и не очень трудоемкий вариант – распространение сигналов ошибки от выходов нейронной сети к ее входам, в направлении, обратном прямому распространению сигналов в обычном режиме работы. Этот алгоритм обучения нейронных сетей получил название процедуры обратного распространения ошибки. Рассмотрим основные детали этого алгоритма.

Согласно методу наименьших квадратов, минимизируемой целевой функцией ошибки нейронной сети является величина:

$$E(w) = \frac{1}{2} \sum_{j,p} (y_{j,p}^{(N)} - d_{j,p})^2, \quad (5)$$

где $y_{j,p}^{(N)}$ – реальное выходное состояние нейрона j выходного слоя N нейронной сети при подаче на ее входы p -го образа;
 d_{jp} – идеальное (желаемое) выходное состояние этого нейрона.

Суммирование ведется по всем нейронам выходного слоя и по всем обрабатываемым сетью образам. Минимизация ведется методом градиентного спуска, что означает подстройку весовых коэффициентов следующим образом:

$$\Delta w_{ij}^{(n)} = -\eta \cdot \frac{\partial E}{\partial w_{ij}}, \quad (6)$$

где w_{ij} – весовой коэффициент синаптической связи, соединяющей i -ый нейрон слоя $n-1$ с j -ым нейроном слоя n ,
 η – коэффициент скорости обучения, $0 < \eta < 1$.

Производную от целевой функции можно выразить следующим образом

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial y_j} \cdot \frac{dy_j}{ds_j} \cdot \frac{\partial s_j}{\partial w_{ij}}, \quad (7)$$

где y_j – выходной сигнал нейрона j ,

s_j – взвешенная сумма его входных сигналов, то есть аргумент активационной функции.

Так как множитель dy_j/ds_j является производной этой функции по ее аргументу, из этого следует, что производная активационной функции должна быть определена на всей оси абсцисс. В связи с этим функция единичного скачка и прочие активационные функции с неоднородностями не подходят для рассматриваемых нейронных сетей. В них применяются такие гладкие функции, как гиперболический тангенс или классическая сигмоидальная функция с экспонентой. В случае гиперболического тангенса

$$\frac{dy}{ds} = 1 - s^2. \quad (8)$$

Третий множитель $\partial s_j / \partial w_{ij}$ это выход нейрона предыдущего слоя $y_i^{(n-1)}$. Кроме того

$$\frac{\partial E}{\partial y_j} = \sum_k \frac{\partial E}{\partial y_k} \cdot \frac{dy_k}{ds_k} \cdot \frac{\partial s_k}{\partial y_j} = \sum_k \frac{\partial E}{\partial y_k} \cdot \frac{dy_k}{ds_k} \cdot w_{jk}^{(n+1)}. \quad (9)$$

Здесь суммирование по k выполняется среди нейронов слоя $n+1$. Введя новую переменную

$$\delta_j^{(n)} = \frac{\partial E}{\partial y_j} \cdot \frac{dy_j}{ds_j} \quad (10)$$

получаем рекурсивную формулу для расчетов величин $\delta_j^{(n)}$ слоя n из величин $\delta_k^{(n+1)}$ более старшего слоя $n+1$.

$$\delta_j^{(n)} = \left[\sum_k \delta_k^{(n+1)} \cdot w_{jk}^{(n+1)} \right] \cdot \frac{dy_j}{ds_j}. \quad (11)$$

Для выходного слоя

$$\delta_l^{(N)} = (y_l^{(N)} - d_l) \cdot \frac{dy_l}{ds_l}. \quad (12)$$

С учетом соотношений (11,12) коррекция синаптических весов определяется следующим образом:

$$\Delta w_{ij}^{(n)} = -\eta \cdot \delta_j^{(n)} \cdot y_i^{(n-1)}. \quad (13)$$

Иногда для придания процессу коррекции весов некоторой инерционности, сглаживающей резкие скачки при перемещении по поверхности целевой функции, выражение (13) дополняется значением изменения веса на предыдущей итерации

$$\Delta w_{ij}^{(n)}(t) = -\eta \cdot (\mu \cdot \Delta w_{ij}^{(n)}(t-1) + (1-\mu) \cdot \delta_j^{(n)} \cdot y_i^{(n-1)}), \quad (14)$$

где μ – коэффициент инерционности,
 t – номер текущей итерации.

Таким образом, полный алгоритм обучения нейронной сети с помощью процедуры обратного распространения ошибки выглядит следующим образом:

1. При заданном входном сигнале в режиме обычного функционирования нейронной сети, когда сигналы распространяются от

входов к выходам, рассчитать значения выходных сигналов. При этом

$$s_j^{(n)} = \sum_{i=0}^M y_i^{(n-1)} \cdot w_{ij}^{(n)}, \quad (15)$$

где M – число нейронов в слое $n-1$ с учетом нейрона с постоянным выходным состоянием $+1$, задающего смещение;

$y_i^{(n-1)} = x_{ij}^{(n)}$ – i -ый вход нейрона j слоя n , $y_j^{(n)} = f(s_j^{(n)})$,

$f()$ – сигмоидальная функция, $y_q^{(0)} = I_q$,

I_q – q -ая компонента вектора входного сигнала.

2. Рассчитать $\delta^{(N)}$ для выходного слоя по формуле (12).

Рассчитать по формуле (13) или (14) изменения весов $\Delta w^{(N)}$ слоя N .

3. Рассчитать по формулам (11) и (13) (или (11) и (14)) соответственно $\delta^{(n)}$ и $\Delta w^{(n)}$ для всех остальных слоев, $n=N-1, \dots, 1$.

4. Скорректировать все веса нейронной сети согласно следующему выражению

$$w_{ij}^{(n)}(t) = w_{ij}^{(n)}(t-1) + \Delta w_{ij}^{(n)}(t).$$

5. Если ошибка сети существенна, перейти на шаг 1. В противном случае, обучение сети заканчивается.

Условие задачи №1

1. Построить таблицы истинности логических функций «AND», «OR», «XOR».

2. Нарисовать схему однослойного или двухслойного персептрона, реализующего логическую функцию в соответствии с вариантом в табл.2.

3. Ответить на контрольные вопросы в соответствии с вариантом.

4. Нейрон j получает входной сигнал от четырех других нейронов, состояния которых определены в табл.3. Вычислите выходной сигнал нейрона j случая, когда нейрон представлен моделью Мак-Каллока-Питтса или квадратичной моделью нейрона. Соответствующие веса связей этого нейрона и

порог приведены в табл.3. Для квадратичного нейрона все дополнительные синаптические веса находятся перемножением соответствующих исходных синаптических весов. Например, синаптический вес $\omega_{123} = \omega_2 \cdot \omega_3$.

Замечание. Поскольку в задаче рассматривается модель одного нейрона индекс k в формулах 1, 4 равен 1. Номер варианта в таблице определяется по последней цифре номера зачетной книжки. Выбор модели Мак-Каллока-Питтса или квадратичной модели соответствует четной (нечетной) предпоследней цифре номера зачетной книжки.

Таблица 2

Варианты для задачи №1

Вариант	Логическая функция	Контрольные вопросы
0	AND	1,3,5
1	NOT	2,4,6
2	XOR	7,8,9
3	OR	10,11,12
4	AND	13,14,15
5	NOT	16,7,11
6	XOR	14,12,5
7	OR	2,3,7
8	AND	10,13,8
9	OR	4,8,16

Контрольные вопросы

1. Сколько входов и сколько выходов может иметь нейрон в модели Мак-Каллока-Питтса?
2. Напишите формулы, с помощью которых происходит преобразование сигналов в однослойном персептроне.
3. Нарисуйте графическое изображение функции активации нейрона Мак-Каллока – Питтса.

4. Нарисуйте схему персептрона, реализующего логическую функцию «OR», и приведите соответствующие ей значения сил синаптических связей и порогов.
5. Назовите основные элементы математической модели нейрона.
6. Перечислите основные виды функций активации и нарисуйте их графическое изображение.
7. Нарисуйте схему персептрона, реализующего логическую функцию «NOT» и приведите соответствующие ей значения сил синаптических связей и порогов.
8. Найти производную функции активации сигмоидального типа и вычислить ее значение в начале координат

$$\varphi(v) = \frac{v}{\sqrt{1+v^2}}.$$

9. Найти производную функции активации сигмоидального типа и вычислить ее значение в начале координат

$$\varphi(v) = \tanh(av/4).$$

10. Найти производную функции активации сигмоидального типа и вычислить ее значение в начале координат

$$\varphi(v) = \frac{1}{1 + \exp(-av)}.$$

11. Назовите основные свойства нейронных сетей.
12. Переформулируйте модель Мак-Каллока – Питтса вводя новый синапс, вес которого равен порогу b .
13. Дайте определение потенциала активации нейрона.
14. Сформулируйте основную функцию порогового элемента нейрона.
15. Сформулируйте отличительные признаки многослойного персептрона.
16. Сформулируйте теорему сходимости персептрона.

Ва риант	Состояние ней- ронов (В – возбужденное, Н- невозбужденное)	Синаптический вес нейронов (модель Мак- Каллока Питтса)	П орог
0	1- Н, 2- В, 3 – В, 4- В	1- 0.5, 2- -0.2, 3 – – 0.8, 4- 0.1	1 0
1	1- В, 2- Н, 3 – В, 4- В	1- 0.6, 2- -0.8, 3 – – 0.6, 4- 0.6	- 2
2	1- В, 2- В, 3 –В, 4- В	1- 0.7, 2- -0.9, 3 – – 0.1, 4- 0.7	4
3	1- В, 2- В, 3 – Н, 4- В	1- 0.3, 2- -0.1, 3 – 0.7, 4- 0.15	1 3
4	1- В, 2- В, 3 –В, 4- В	1- 0.2, 2- -0.4, 3 – 0.65, 4- 0.1	2 5
5	1- В, 2- В, 3 –В, 4- Н	1- 0.1, 2- -0.6, 3 – 0.75, 4- 0.2	5 0
6	1- В, 2- В, 3 –В, 4- В	1- 0.15, 2- -0.7, 3 – 0.15, 4- 0.3	4 0
7	1- Н, 2- В, 3 – В, 4- В	1- 0.25, 2- -0.5, 3 – 0.1, 4- -0.7	6
8	1- В, 2- Н, 3 – В, 4- В	1- 0.35, 2- -0.3, 3 – 0.5, 4- -0.8	1 1
9	1- В, 2- В, 3 – Н, 4- В	1- 0.8, 2- -0.25, 3 – 0.7, 4- -0.6	- 8

Задача № 2

АППРОКСИМАЦИЯ ФУНКЦИИ ОДНОЙ ПЕРЕМЕННОЙ

Цель работы- научиться работать с сетью прямой передачи сигнала с помощью функции системы **MATLAB newff** . Разобраться с алгоритмом обратного распространения ошибки.

Краткие теоретические сведения

В задаче рассматривается нейронная сеть с прямой передачей сигнала (с прямой связью), то есть сеть, в которой сигналы передаются только в направлении от входного слоя к выходному, и элементы одного слоя связаны со всеми элементами следующего слоя. Выберем в качестве модели обучения обучение методом распространения ошибки.

Пронумеруем выходы нейронной сети индексом $j = 1, 2, \dots, n$, а обучающие примеры индексом $M = 1, 2, \dots, M_0$. Тогда в качестве целевой функции можно выбрать функцию ошибки как сумму квадратов расстояний между реальными выходными состояниями y_{jM} нейронной сети, выдаваемых сетью на входных данных примеров, и правильными значениями функции d_{jM} , соответствующими этим примерам. Пусть $x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$ – столбец входных значений, где $i=1, 2, \dots, n$. Тогда $y = \{ y_j \}$ – выходные значения, где $j=1, 2, \dots, m$. В общем случае $n \neq m$. Рассмотрим разность $y_{jM} - d_{jM}$, где d_{ji} – точное (правильное) значение из примера. Эта разность, которая часто называется ошибкой, должна быть минимальна. Введем расстояния согласно евклидовой метрике, определим норму следующим образом

$$\|y - d\| = \sqrt{(y - d, y - d)} \quad (16)$$

Пусть целевая функция имеет вид

$$E = \frac{1}{2} \sum_{j,M} (y_{j,M} - d_{j,M})^2 \quad (17)$$

Задача обучения нейронной сети состоит в том, чтобы найти такие коэффициенты $w_{\beta k}$, при которых достигается минимум $E(\mathbf{w}) (E \geq 0)$.

На рис. 4 показана архитектура нейронной сети с прямой передачей сигнала.

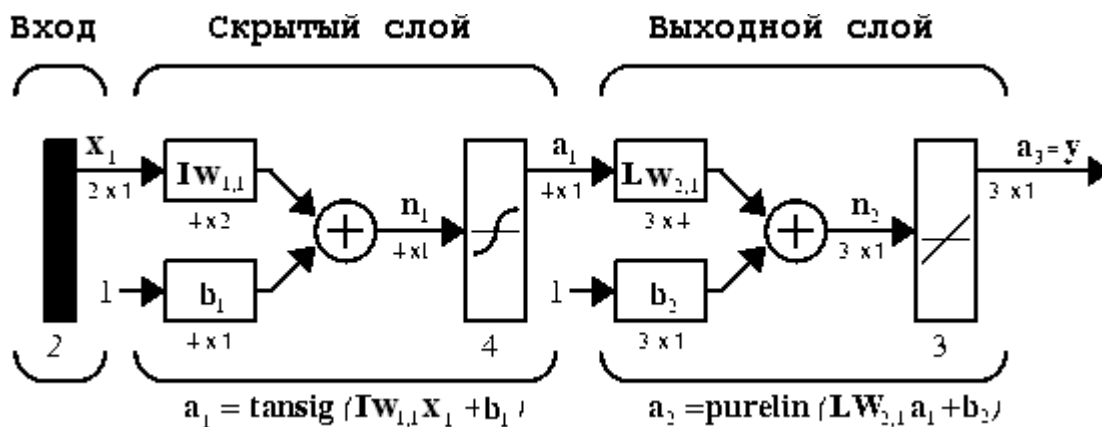


Рисунок 4. Схема архитектуры нейронной сети с прямой передачей сигнала

Здесь приняты следующие обозначения x_i - вектор входа, Iw_{ij} , Lw_{ij} - матрицы весов входа и выхода, b_i - смещение, a_i - выход слоя, y - выход сети, tansig (гиперболическая тангенциальная), purelin (линейная) - соответствующие функции активации.

Весы и смещения определяются с помощью алгоритма обратного распространения ошибок.

Обучение сети обратного распространения требует выполнения следующих операций:

1. Выбрать очередную обучающую пару из обучающего множества; подать входной вектор на вход сети.
2. Вычислить выход сети.
3. Вычислить разность между выходом сети и требуемым выходом (целевым вектором обучающей пары).
4. Скорректировать веса сети так, чтобы минимизировать ошибку.
5. Повторять шаги с 1 по 4 для каждого вектора обучающего множества до тех пор, пока ошибка на всем множестве не достигнет приемлемого уровня.

Пример решения типовой задачи

Решение задачи состоит из следующих этапов: прежде всего, необходимо оцифровать график функции $y = f(x)$, то есть получить ряд соответствующих значений по горизонтальной и вертикальной осям.

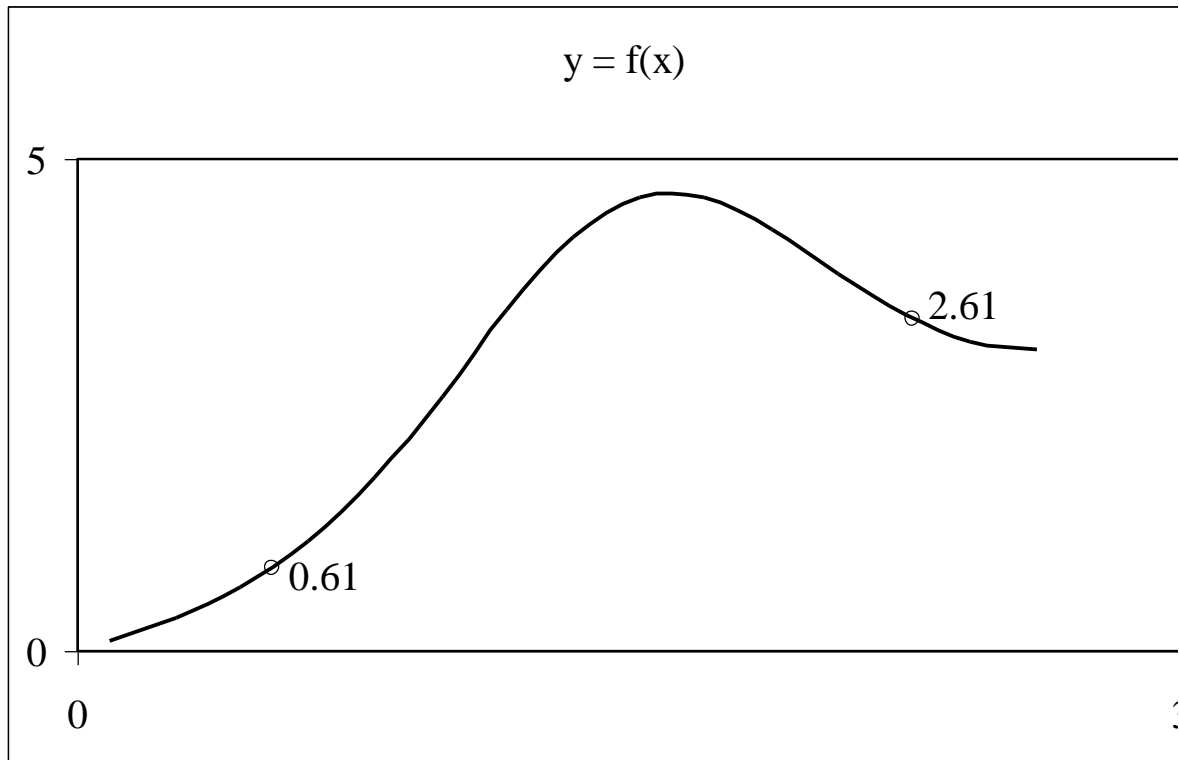


Рисунок 5. Пример зависимости для функции одной переменной

В примере, показанном на рис. 5 были получены два массива, каждый из которых состоит из 15 значений. По горизонтальной оси – [0.10 0.31 0.51 0.72 0.93 1.14 1.34 1.55 1.76 1.96 2.17 2.38 2.59 2.79 3.00]. По вертикальной оси – [0.1010 0.3365 0.6551 1.1159 1.7632 2.5847 3.4686 4.2115 4.6152 4.6095 4.2887 3.8349 3.4160 3.1388 3.0603].

Ниже приводится программа создания, обучения нейронной сети и вывода результатов в системе MathLab.

```
x=[0.10 0.31 0.51 0.72 0.93 1.14 ...
    1.34 1.55 1.76 1.96 2.17 2.38 ...
    2.59 2.79 3.00];
y=[0.1010 0.3365 0.6551 1.1159 1.7632 2.5847 ...
    3.4686 4.2115 4.6152 4.6095 4.2887 3.8349 ...
    3.4160 3.1388 3.0603];
net=newff([0 3],[5,1],{'tansig','purelin'},'trainbfg');
net.trainParam.epochs=300;
net.trainParam.show=50;
net.trainParam.goal=1.37e-2;
[net,tr]=train(net,x,y);
```

```

an=sim(net,x);
plot(x,y,'+r',x,an,'-g'); hold on;
xx=[0.61 2.61];
v=sim(net,xx)
plot(xx,v,'ob','MarkerSize',5,'LineWidth',2)

```

В результате выполнения программы получают следующие результаты, отражённые на рис. 6 и 7:

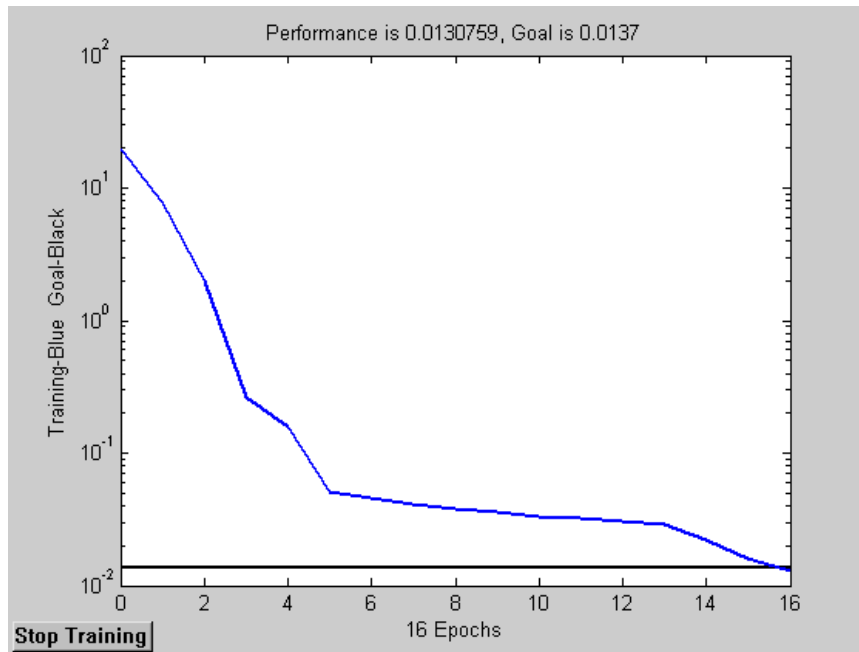


Рисунок 6. Характеристика точности обучения в зависимости от числа эпох обучения

В массиве v содержатся приближённые значения для двух контрольных точек, указанных на графике (рис. 5) $xx=[0.61\ 2.61]$. При данных параметрах сети получены значения: $v = [1.05\ 3.35]$. Сравнив эти приближённые значения с точными значениями $[0.85\ 3.37]$, можно сделать вывод о корректности построения нейронной сети.

Варианты задачи приведены на рис.8. При оформлении решения задачи необходимо включить следующие пункты:

1. Исходные данные в соответствии с вариантом (рис. 8);
2. Текст программы с подробными комментариями;
3. Характеристику точности обучения (рис. 6);
4. Результаты моделирования (рис. 7);
5. Сопоставление результатов в контрольных точках;
6. Краткие выводы о результатах работы.

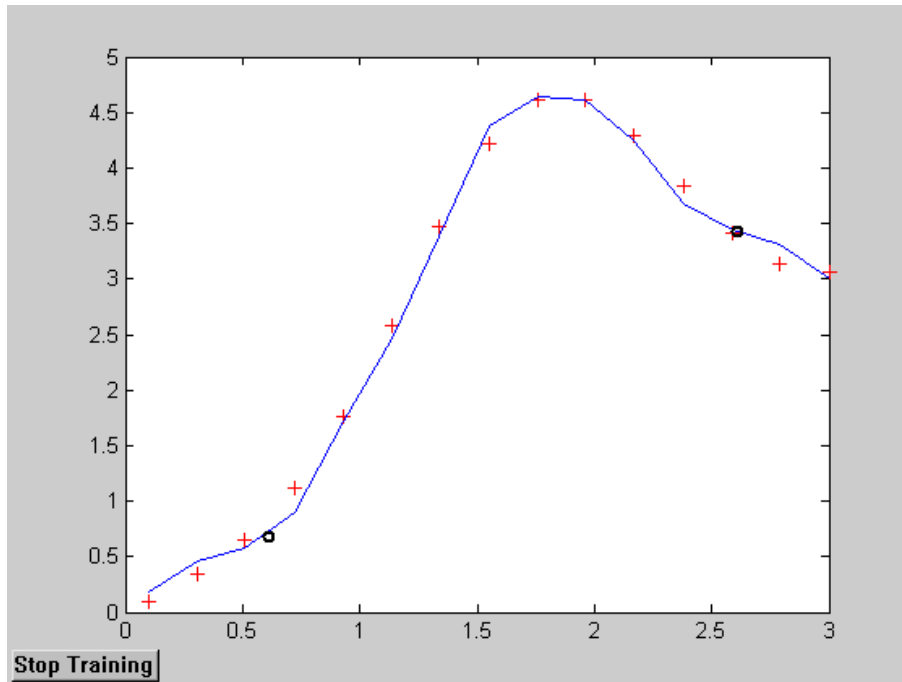
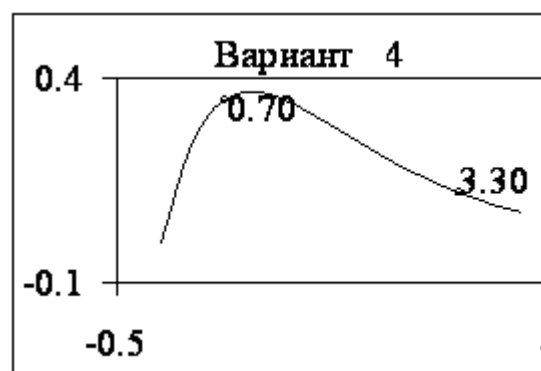
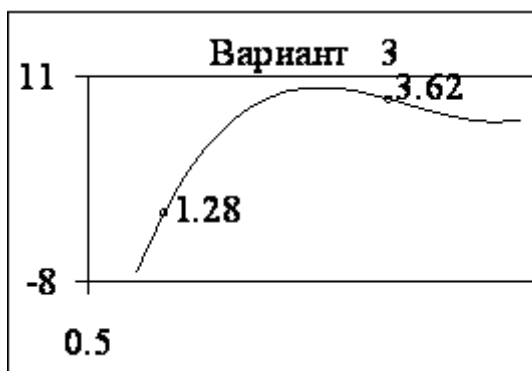
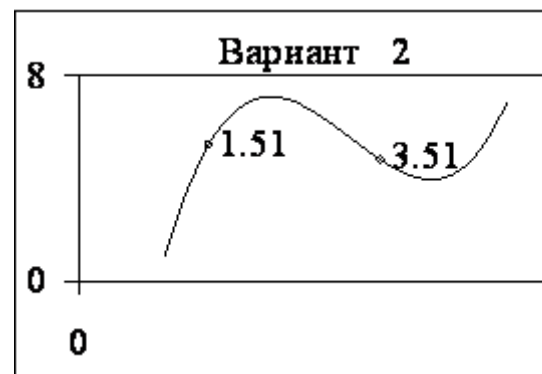
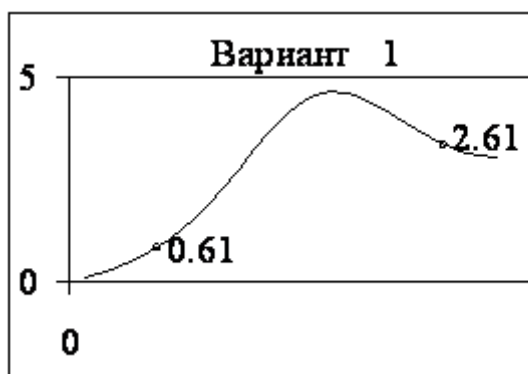


Рисунок 7. Результаты моделирования сети: + - исходные данные; сплошная линия и символ «o» – результаты моделирования всей зависимости и в контрольных точках



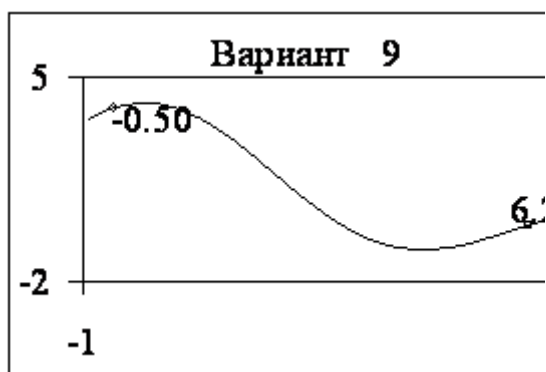
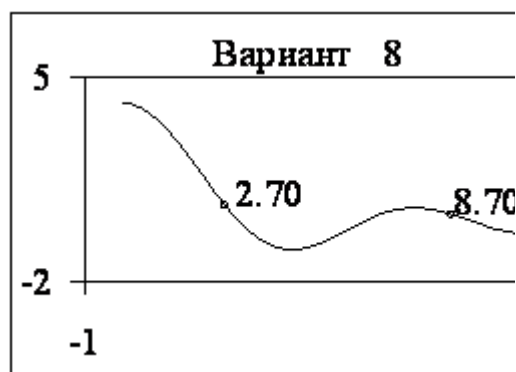
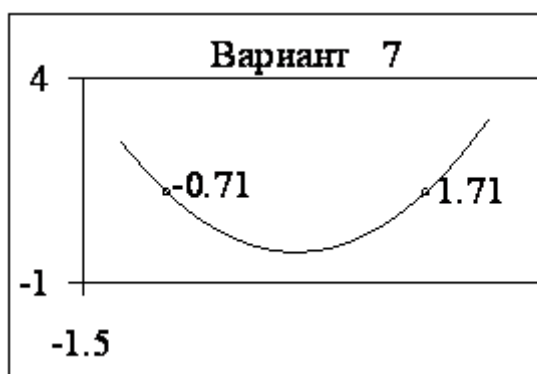
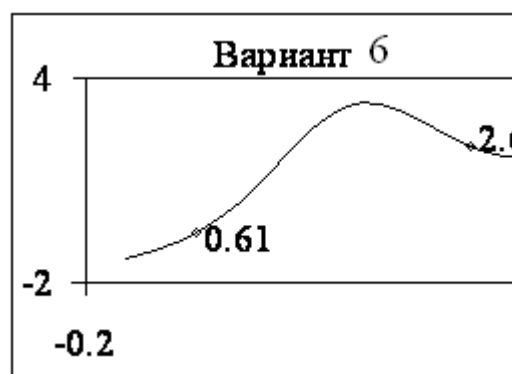
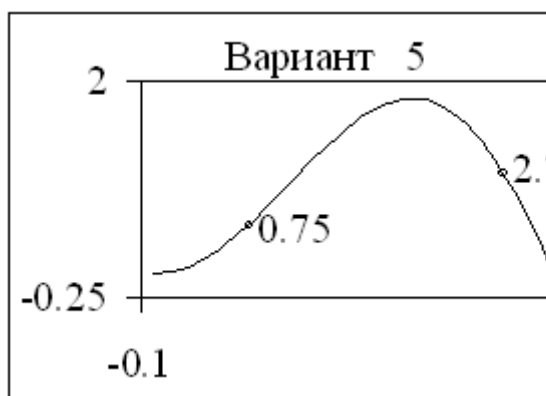


Рисунок 8. Варианты заданий

Список литературы

1. Медведев В.С., Потемкин В.Г. Нейронные сети. Матлаб 6. М.: Диалог МИФИ, 2002. – 496с.
2. Каллан Р. Основные концепции нейронных сетей. М.: Издательский дом «Вильямс», 2001. – 287с.
3. Уоссермен Ф. Нейрокомпьютерная техника: Теория и практика. – М.: Мир, 1992.
4. Круглов В.В., Борисов В.В. Искусственные нейронные сети. М.: Горячая линия - Телеком, 2001. – 382с.
5. Осовский С. Нейронные сети для обработки информации. М.: Финансы и статистика, 2002. – 344с.
6. Зайцевский И.В., Свиридов А.П., Слесарев Д.А. Нейронные сети и их приложения. М.: МЭИ, 2002. – 95с.
7. Дьяконов В.П., Круглов В.В. MATLAB 6.5 SP1/7/7 SP1/7 SP2 Simulink 5/6. Инструменты искусственного интеллекта и биоинформатики. М.: СОЛОН-ПРЕСС, 2006. – 456с.
8. Хайкин С. Нейронные сети: полный курс, 2-е издание. – М.: Изд-во «Вильямс». – 2006. – 1104 с.