**КОНТРОЛЬНАЯ РАБОТА**

По дисциплине: Интеллектуальные информационные системы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

на тему: Искусственные нейронные сети на основе алгоритма обратного распространения ошибки\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Выполнил

Студент: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Специальность: Прикладная информатика

(в экономике)\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Группа \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Работу проверил:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ставрополь, 2014

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_Toc355462349)

[1. МАТЕМАТИЧЕСКАЯ МОДЕЛЬ НЕЙРОННОЙ СЕТИ С АЛГОРИТМОМ ОБРАТНОГО РАСПРОСТАНЕНИЯ ОШИБКИ 4](#_Toc355462350)

[2. РАСЧЕТ НЕЙРОННОЙ СЕТИ (1 ПРОХОД)](#_Toc355462350) 6

[2.1 Шаг №0](#_Toc355462352) 7

[2.2 Шаг №1](#_Toc355462352) 7

[2.3 Шаг №2](#_Toc355462352) 8

[2.4 Шаг №3](#_Toc355462352) 9

[2.5 Шаг №4](#_Toc355462352) 9

[2.6 Шаг №5](#_Toc355462352) 10

[3. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ НЕЙРОННОЙ СЕТИ](#_Toc355462351) 11

[3.1 Общие сведения](#_Toc355462352) 11

[3.2 Функциональное назначение](#_Toc355462353) 12

[3.3 Описание логической структуры](#_Toc355462354) 12

[3.4 Требования к техническому обеспечению.](#_Toc355462355) 16

[3.5 Вызов программы.](#_Toc355462355) 16

[3.6 Входные данные.](#_Toc355462355) 16

[3.7 Выходные данные.](#_Toc355462355) 17

[3.8 Описание тестовых прогонов.](#_Toc355462355) 17

[ЗАКЛЮЧЕНИЕ](#_Toc355462358) 18

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ](#_Toc355462359) 18

ВВЕДЕНИЕ

В современном мире все чаще мы сталкиваемся с проблемами, которые подчас решить может только человек — ЭВМ с ними справиться не в силах. Примером таких задач могут служить задачи прогнозирования (экономического, климатического), задачи классификации объектов, задачи распознавания символов и т.п. Для решения таких задач используются т.н. искусственные нейронные сети (ИНС) — тривиальные модели функционирования человеческого головного мозга. На данный момент существует огромное количество видов ИНС, которые различаются поставленными перед ними задачами, физической структурой сети, внутренними параметрами: функцией активации, алгоритмом обучения и т.д. В данной работе нами была построена и изучена ИНС с одним скрытым слоем, с алгоритмом обучения по методу обратного распространения ошибки. Раздел 1 содержит математическую модель моделируемой нами сети. В разделе 2 описаны расчеты ИНС, в соответствии с вариантом работы. Раздел 3 содержит описание разработанной нами программы на ЭВМ, которая инициализирует ИНС с начальными параметрами, обучает ее и тестирует на некотором наборе входных данных.

1. МАТЕМАТИЧЕСКАЯ МОДЕЛЬ НЕЙРОННОЙ СЕТИ С АЛГОРИТМОМ ОБРАТНОГО РАСПРОСТАНЕНИЯ ОШИБКИ

В соответствие с вариантом работы нам необходимо построить ИНС со структурой 2-2-1, то есть два входных нейрона, два нейрона в скрытом слое и один выходной нейрон. Также необходимо учесть два нейрона смещения, один из которых находиться на входном слое, второй на скрытом слое. Схематически рассматриваемую сеть изобразим следующим образом

Схема 1. Структура ИНС 2-2-1.

В виду того, что ожидаемым откликом от сети, в соответствии с нашим вариантом, является сигнал , то в качестве функции активации для нейронов на скрытом слое и для выходного нейрона выберем функцию

Опишем математическую модель ИНС.

Взвешенная сумма нейрона

где суммирование ведется по всем номерам нейронов на предыдущем слое. Отклик нейрона

Ошибка выходного нейрона

Ошибка нейронов скрытого слоя

где . Изменение веса нейрона после этапа обучения

где — начальный вес ребра, соединяющего нейроны и , — вес ребра, соединяющего нейроны и на -ом шаге обучения, — норма обучения, — норма инерции обучения. В результате одного шага обучения веса ребер в ИНС изменяются по следующему правилу

Обучение ИНС проходит до тех пор, пока невязка выходного сигнала превышает допустимый порог

После того, как выходной сигнал ИНС по модулю не будет превышать ожидаемого выходного сигнала более чем на процесс обучения останавливается и сеть считается обученной.

2. РАСЧЕТ НЕЙРОННОЙ СЕТИ (1 ПРОХОД)

В соответствие с вариантом работы начальные данные для ИНС

Коэффициенты и допуски выберем следующим образом

(обучение ИНС необходимо продолжить)

3. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ НЕЙРОННОЙ СЕТИ

3.1 Общие сведения

С целью визуализации процесса обучения ИНС нами была разработана программа на ЭВМ, которая позволяет наглядно увидеть каждый шаг обучения ИНС по методу обратного распространения ошибки. Интерфейс программы представлен на рисунке 2. Назначение каждого отдельного элемента этого интерфейса мы опишем нижу. Сейчас же заметим, что в целом программа оформлена в виде ИНС типа 2-2-1.

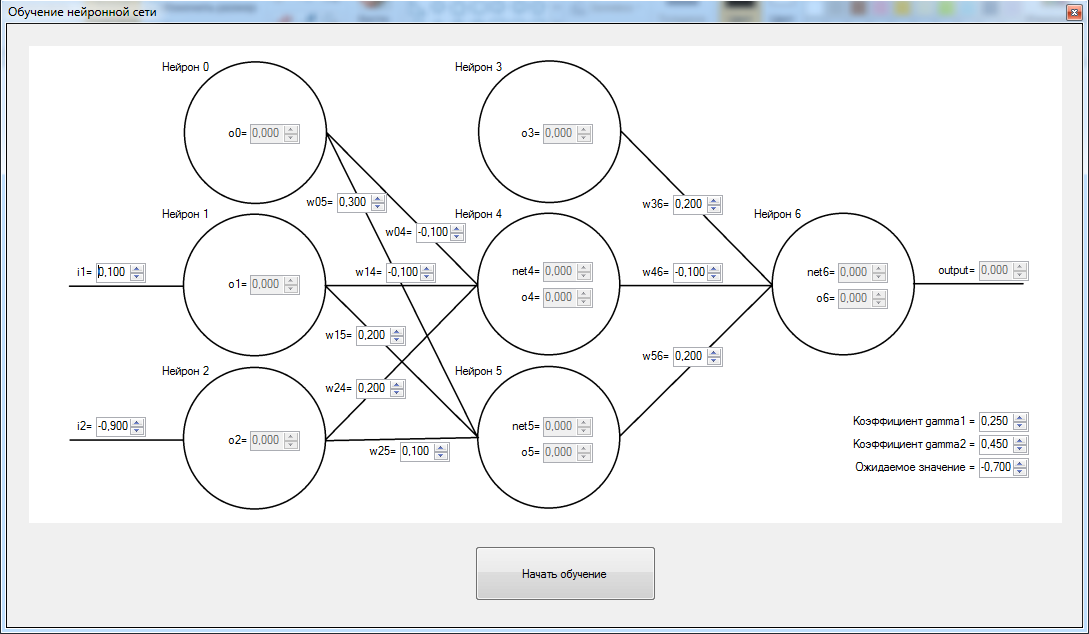


Рис 2. Интерфейс приложения.

3.2 Функциональное назначение

Данное приложение создано с целью обучения ИНС типа 2-2-1. Изначально, до обучения, ИНС представляет из себя структуру нейронов, которые, принимая входные сигналы, на выходе образуют сигнал, не имеющий никакого смысла. С помощью метода обратного распространения ошибки, шаг за шагом, при помощи данной программы мы можем обучить ИНС таким образом, чтобы выход сети начал выдавать нужную нам информацию, в частности сигнал, который мы от ИНС ожидаем.

3.3 Описание логической структуры

Алгоритм работы программы представлен на схеме 3.

step = 0

isFirstEra = 0

Ввод i1, i2

Ввод w04, w05

Ввод w14, w15

Ввод w24, w25

Ввод w36, w46, w56

Ввод gamma1, gamma2

Ввод th

step

step = 1

0

1

o0 = 1

o1 = i1

o2 = i2

Вывод o0

Вывод o1

Вывод o2

net4 = o0\*w04+o1\*w14+o2\*w24

o4 = tanh(net4)

net5 = o0\*w05+o1\*w15+o2\*w25

o5 = tanh(net5)

o3 = 1

2

step = 2

Вывод net4,o4

Вывод net5,o5

Вывод o3

step = 3

3

net6 = o3\*w36+o4\*w46+o5\*w56

o6 = tanh(net6)

Вывод net6,o6

step = 4

4

output = o6

|output – th| < 0.001

step = 5

Вывод “ИНС обучена!»

step = 6

5

6

delta6 = (th-o6)(o6+1)(1-o6)

delta5 = (o5+1)(1-o5)\*delta6\*w56

delta4 = (o4+1)(1-o4)\*delta6\*w46

isFirstEra = true

dw56 = gamma1\*delta6\*o5

dw46 = gamma1\*delta6\*o4

dw36 = gamma1\*delta6\*o3

dw56 = gamma1\*delta6\*o5+gamma2\*dw56

dw46 = gamma1\*delta6\*o4+gamma2\*dw46

dw36 = gamma1\*delta6\*o3+gamma2\*dw36

w56 = w56 + dw56

w46 = w46 + dw46

w36 = w36 + dw36

Вывод w56,w46,w36

step = 7

7

delta6 = (th-o6)(o6+1)(1-o6)

delta5 = (o5+1)(1-o5)\*delta6\*w56

delta4 = (o4+1)(1-o4)\*delta6\*w46

isFirstEra = true

dw25 = gamma1\*delta5\*o2

dw24 = gamma1\*delta4\*o2

dw15 = gamma1\*delta5\*o1

dw14 = gamma1\*delta4\*o1

dw05 = gamma1\*delta5\*o0

dw04 = gamma1\*delta4\*o0

dw25 = gamma1\*delta5\*o2+gamma2\*dw25

dw24 = gamma1\*delta4\*o2+gamma2\*dw24

dw15 = gamma1\*delta5\*o1+gamma2\*dw15

dw14 = gamma1\*delta4\*o1+gamma2\*dw14

dw05 = gamma1\*delta5\*o0+gamma2\*dw05

dw04 = gamma1\*delta4\*o0+gamma2\*dw04

w25 = w25 + dw25

w24 = w24 + dw24

w15 = w15 + dw15

w14 = w14 + dw14

w05 = w05 + dw05

w04 = w04 + dw04

Вывод w04,w14,w24

Вывод w05,w15,w25

isFirstEra = false

step = 2

Схема 3. Блок-схема приложения.

3.4 Требования к техническому обеспечению

Реализованное приложение создано в среде Visual Studio 2010 и использует .NetFramework v4.5. На клиентском ЭВМ должна быть установлена видеокарта, поддерживающая технологию OpenGL. Также должен быть установлен пакет .NetFramework версии 4.5. Так как приложение выполняет расчеты, то необходим хороший процессор, не ниже Intel Pentium 4, 1.8 GHz. Оперативная память также используется, поэтому необходимо от 128 Mb оперативной памяти на материнской плате.

3.5 Вызов программы

Программа имеет структуру приложения Win32, может быть вызвана из проводника двойным нажатие левой клавиши мыши на значке neuro.exe. Также программа может быть вызвана командой neuro.exe из консоли (без параметров).

3.6 Входные данные

Программа не принимает параметры из консоли. Ввод входных данных организован в диалоговом окне приложения непосредственно через элементы управления. Пользователь, перед началом обучения ИНС, может изменить входные сигналы сети, начальные синаптические веса, коэффициенты обучения и ожидаемый отклик. Все эти параметры, до начала процесса обучения, доступны пользователю. После начала процесса обучения все поля становятся недоступными для редактирования, так как программа начинает изменять эти поля в соответствии с запрограммированным алгоритмом.

3.7 Выходные данные

После некоторого количества шагов ИНС обучается и выдает на выход сигнал, который пользователь ожидает от нее. В качестве результата работы программы пользователь имеет обученные значения синаптических весов, которые он может использовать в дальнейшем для практических приложений данной ИНС.

3.8 Описание тестовых прогонов

Результаты тестовых прогонов приложения представлены в таблице 4.



Таблица 4. Результаты тестовых прогонов приложения.

ЗАКЛЮЧЕНИЕ

В данной работе была построена математическая модель ИНС типа 2-2-1, рассчитаны веса ребер сети, написано приложение для ЭВМ, имитирующее процесс обучения ИНС. Данную ИНС в дальнейшем можно доработать, усложнив модель, и использовать для решения прикладных задач.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Круглов В.В., Борисов В.В. Искусственные нейронные сети. Теория и практика. — М.: Горячая линия – Телеком, 2001. 382 с.
2. Горбань А.Н., Россиев Д.А. Нейронные сети на персональном компьютере. — Новосибирск: Наука, 1996. 275 с.
3. Р. Тадеусевич, Б. Боровик, Т. Гончаж, Б. Леппер, Элементарное введение в технологию нейронных сетей, М.: Горячая линия – Телеком, 2011. 408 с.
4. Г. Шилдт, C# 4.0. Полное руководство, Вильямс, 2013, 1056 с.
5. Уоссермен Ф. Нейрокомпьютерная техника. — М.: Мир, 1992. 127 с.