

## ОПИСАНИЕ АЛГОРИТМА ОБУЧЕНИЯ НЕЙРОННОЙ СЕТИ, РЕАЛИЗОВАННОГО СРЕДСТВАМИ MICROSOFT VISUAL BASIC 6.0

САВЧЕНКО Е.Ю.

МУК

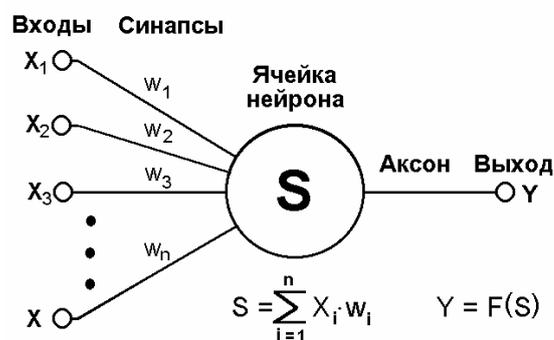
*Рассматривается алгоритм обучения трехслойного перцептрона реализованного средствами Microsoft Visual Basic. Данный алгоритм отражает метод наискорейшего спуска. Результат работы, рассматриваемой нейронной сети, являются блоком оценки ответа в алгоритме отбора и предъявления заданий в адаптивной тестирующей системе.*

*The algorithm of the education трехслойного перцептрона marketed facility Microsoft Visual Basic is Considered. The Givenned algorithm reflects the method of the наискорейшего lowering. The Result of the work, considered нейронной network, are a block of the estimation of the answer in algorithm of the selection and presenting the tasks in adaptive testing system.*

Одним из современных направлений в тестовых методах оценивания подготовленности обучаемых является компьютерное адаптивное тестирование. Под адаптивным тестовым контролем понимают компьютерную систему научно обоснованной проверки и оценки результатов обучения, обладающую высокой эффективностью за счет оптимизации процедур генерации, предъявления и оценки результатов выполнения адаптивного теста [1]. Адаптивный тест – это тест, в котором сложность заданий меняется в зависимости от правильных ответов испытуемого: если обучаемый правильно отвечает на тестовые задания, сложность последующих заданий повышается, если не правильно понижается. Эффективность контрольно-оценочных процедур повышается при использовании многошаговой стратегии отбора и предъявления заданий, основанной на алгоритмах с полной контекстной зависимостью, в которых очередной шаг совершается только после оценки результатов выполнения предыдущего шага. После выполнения испытуемым очередного задания возникает потребность в принятии решения о подборе трудности следующего задания в зависимости от того, верным или неверным был предыдущий ответ. Алгоритм отбора и предъявления заданий строится по принципу обратной связи и работает на базе нейронных сетей. При правильном ответе испытуемого очередное задание выбирается более трудным, а неверный ответ влечет за собой предъявление последующего более легкого задания, чем то, на которое испытуемым был дан неверный ответ. Блок оценки ответа в алгоритме отбора и предъявления заданий реализован на базе

нейронных сетей.

Нейронные сети (Neural Networks) это модели биологических нейронных сетей мозга, в которых нейроны имитируются относительно простыми, часто однотипными, элементами (искусственными нейронами). Нейрон представляет собой единицу обработки информации в нейронной сети. На рис. 1 показана модель нейрона, лежащего в основе искусственных нейронных сетей.



**Рис.1. Искусственный нейрон.**

В этой модели можно выделить три основных элемента.

1. Набор синапсов и связей, каждый из которых характеризуется своим весом или силой. В частности, сигнал  $x_i$  на входе синапса  $i$ , связанного с нейроном  $k$ , умножается на вес  $w_{ki}$ . Первый индекс относится к рассматриваемому нейрону, а второй – ко входному окончанию синапса, с которым связан данный вес. В отличие от синапсов мозга синаптический вес искусственного нейрона может иметь как положительные, так и отрицательные значения.

2. Сумматор складывает входные

сигналы, взвешенные относительно соответствующих синапсов нейрона. Эту операцию можно описать как линейную комбинацию.

3. Функция активации ограничивает амплитуду выходного сигнала нейрона. Эта функция также называется функцией сжатия [2]. Текущее состояние нейрона определяется, как взвешенная сумма его входов:

$$S = \sum x_i w_i. \quad (1)$$

Выход нейрона есть функция его состояния:

$$y = f(s). \quad (2)$$

Нелинейная функция  $f$  называется активационной и может иметь различный вид, как показано на рис.3. Одной из наиболее распространенных является нелинейная функция с насыщением, так называемая логистическая функция или сигмоид (т.е. функция S-образного вида):

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

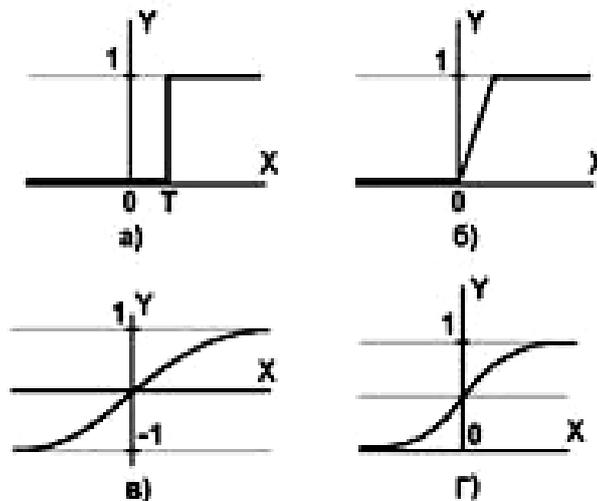


Рис. 3 Активационная функция:

- а) функция единичного скачка; б) линейный порог (гистерезис); в) сигмоид – гиперболический тангенс; г) сигмоид – формула (3)

При уменьшении  $\alpha$  сигмоид становится более пологим, в пределе при  $\alpha=0$  вырождаясь в горизонтальную линию на уровне 0.5, при увеличении  $\alpha$  сигмоид приближается по внешнему виду к функции единичного скачка с порогом  $T$  в точке  $x=0$ . Из выражения для сигмоида очевидно, что выходное значение нейрона лежит в диапазоне  $[0,1]$ . [3].

Самым важным свойством нейронных сетей является их способность обучаться на основе данных окружающей среды. Процесс обучения - это процесс, в котором свободные параметры нейронной сети настраиваются посредством моделирования среды, в которую сеть встроена. Тип обучения определяется способом надстройки этих параметров. Это определение процесса обучения предполагает следующую последовательность событий:

1. В нейронную сеть поступают

стимулы из внешней среды.

2. В результате этого изменяются свободные параметры нейронной сети.

3. После изменения внутренней структуры нейронная сеть отвечает на возбуждения уже иным образом.

Данный список четких правил решения проблемы обучения называется алгоритмом обучения. Рассматриваемая в статье нейронная сеть, реализована на базе Microsoft Visual Basic 6.0.

Исходные данные:

Вектор входа  $x_0$ , состоящий из трех элементов.

1.  $U_i$  количество уровней сложности заданий в тесте  $i=1,2,3$ .
2.  $O$  правильность ответа студента на вопрос  $O=0,1$ .
3.  $It$  количество пройденных вопросов в тесте по одному уровню сложности. Вектор целевых значений  $d$ .

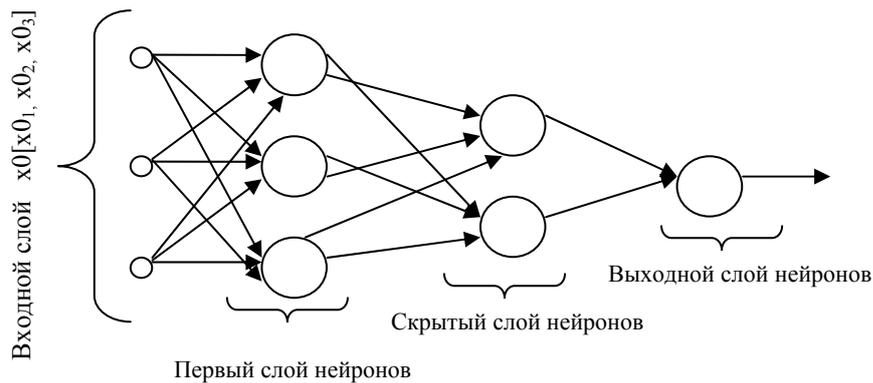
1. Уровень сложности следующего задания, предоставляемого тестирующемуся.

Последовательность работы алгоритма:

Тестирующая система предлагает испытуемому начальный средний уровень сложности,  $U_i=2$ , в переменную  $I_t$  записывается количество пройденных вопросов в тесте по одному уровню сложности. Правильность ответа на вопрос данного уровня сложности фиксируется в переменную  $O$ . Полученные данные служат входным вектором для расчета в нейронной

сети. Результатом работы нейронной сети будет предложение о повышении или понижении уровня сложности последующего вопроса.

Для решения поставленной задачи была выбрана нейронная сеть, имеющая архитектуру трехслойного персептрона с тремя нейронами во входном слое, тремя нейронами в скрытом слое и одним нейроном в выходном слое рис. 4, с сигмоидальной функцией активации во всех слоях.



**Рис.4. Архитектура трехслойного персептрона с тремя нейронами во входном слое, тремя нейронами в скрытом слое и одним нейроном в выходном слое.**

Тестовое множество составляет 22 примера. Входное тестовое множество с вектором целей записывается в текстовый файл. Считывание данных из файла организуется с помощью процедуры  $r\_file()$ .

Листинг программа 1.

```
Public Sub r_file()
i = 0
vektor_v = 3
Open "g:/text3.txt" For Input As #1
Do While Not EOF(1)
i = i + 1
Line Input #1, textline
m(i) = textline
Loop
Close #1
.....
End Sub
Значения матрицы весовых
коэффициентов устанавливаются в
процедуре wesa(). Листинг программа 2.
Начальные значения формируются с
помощью генератора случайных числе в
диапазоне от 0 до 1.
```

Листинг программа 2

```
Public Sub wesa()
Randomize 'Initialize random-number generator.
For i = 1 To kol_neir_1
For j = 1 To vektor_v
w1(i, j) = ((1 * Rnd) - 0.5)
Next j
Next i
For i = 1 To kol_neir_2
For j = 1 To kol_neir_1
w2(i, j) = ((1 * Rnd) - 0.5)
Next j
Next i
For i = 1 To kol_neir_3
For j = 1 To kol_neir_2
w3(i, j) = ((1 * Rnd) - 0.5)
Next j
Next i
End Sub
```

Сформировав вектора входа, матрицы весовых коэффициентов, вычисляется выходной сигнал с помощью процедуры  $raschet()$ . Листинг программа 3.

Листинг программа 3

```
Sub raschet()
kol_sloy = 3
vektor_v = 3
```

```

kol_neir_1 = 3
kol_neir_2 = 2
kol_neir_3 = 1
nu = 0.6
kol_epox = 700
r_file
raziw(1) = kol_neir_1
razjw(1) = vektor_v
raziw(2) = kol_neir_1
razjw(2) = kol_neir_2
raziw(3) = kol_neir_2
razjw(3) = kol_neir_2
wesa
For epoxa = 1 To kol_epox
For dann = 1 To test_mnoj
    prpr
    matv2
    grad

```

```

Next dann
nu = nu - 0.0001
nak(epoxa) = ee_st
Next epoxa
End Sub

```

Смоделированная нейронная сеть обучена с помощью алгоритма обучения с учителем. Концептуально участие учителя можно рассматривать как наличие знаний об окружающей среде, представленных в виде пар вход-выход. Учителю и обучаемой сети подается обучающий вектор из окружающей среды. На основе сформированных знаний учитель передает обучаемой нейронной сети желаемый отклик, соответствующий данному входному вектору. Параметры сети корректируются с учетом обучающего вектора и сигнала ошибки рис.5.

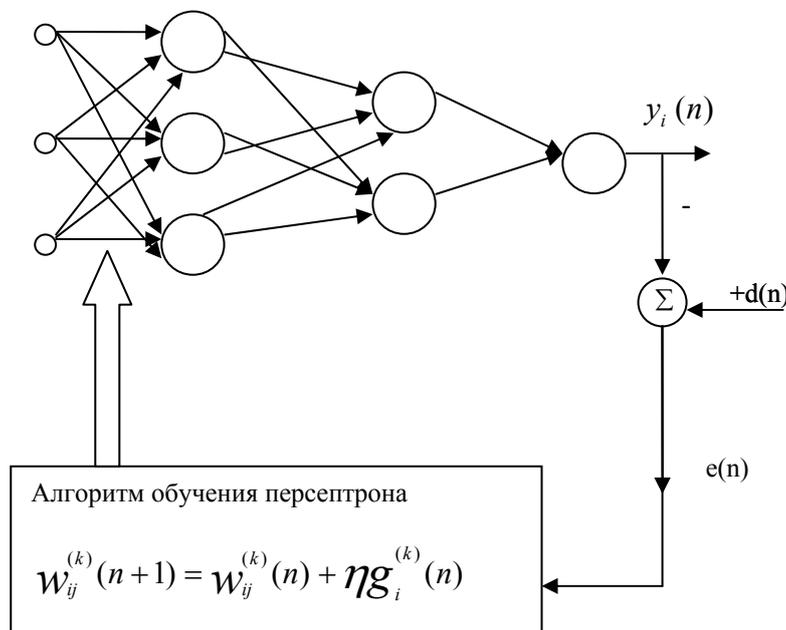


Рис.5. Алгоритм обучения нейронной сети.

Описанный механизм обучения в рассматриваемой нейронной сети был реализован с помощью метода наискорейшего спуска. В общем виде данный алгоритм можно записать как

$$y_i^{(k)}(n) = f(s_i^{(k)}(n)) \quad (4)$$

$$s_i^{(k)}(n) = \sum_{j=0}^{N_{k-1}} w_{ij}^{(k)}(n) x_j^{(k)}(n) \quad (5)$$

$$e_i^k(n) = d_i^k(n) - y_i^{(k)}(n) \quad (6)$$

$$E(n) = \frac{1}{2} e^2(n) \quad (7)$$

$$\nabla = \left[ \frac{\partial}{\partial w_1}, \frac{\partial}{\partial w_2}, \frac{\partial}{\partial w_3}, \dots, \frac{\partial}{\partial w_m} \right]^T \quad (8)$$

$$\nabla E = \left[ \frac{\partial E}{\partial w_1}, \frac{\partial E}{\partial w_2}, \frac{\partial E}{\partial w_3}, \dots, \frac{\partial E}{\partial w_m} \right]^T \quad (9)$$

$$g_i^{(k)}(n) = \nabla E(w) \quad (10)$$

$$w_{ij}^{(k)}(n+1) = w_{ij}^{(k)}(n) + \eta g_i^{(k)}(n) \quad (11)$$

Представленная нейронная сеть состоит из  $L$  слоев. В каждом слое расположено  $N_k$

элементов,  $k = 1, \dots, L - 1, i = 1, \dots, N_k$ ;  $y_i^{(k)}$  выходной сигнал  $i$ -го нейрона в  $k$ -м слое,  $k = 1, \dots, L - 1, i = 1, \dots, N_k$ ;  $w_{ij}^{(k)}$  обозначает вес связи  $i$ -го нейрона в  $k$ -м слое, которая соединяет этот нейрон с  $j$ -м входным сигналом  $x_j^{(k)}, j = 0, 1, \dots, N_k$ ;  $d_i^{(L)}$  эталонный выходной сигнал сети;  $E(n)$  текущее

значение энергии ошибки;  $\eta$  величина шага коррекции,  $g_i^{(k)}(n)$ -вектор градиента вычисленный в точке  $w(n)$ .

Смоделированный трехслойный персептрон был обучен 300 эпохами Рис.6.

Результаты моделирования нейронной сети

Персептрон	Размер обучающего множества	Количество эпох	Среднеквадратическая ошибка	Размер экспериментального множества	Экспериментальные результаты
Трехслойный персептрон	22	300	0.00114	15	1 ошибка на экспериментальном множестве.

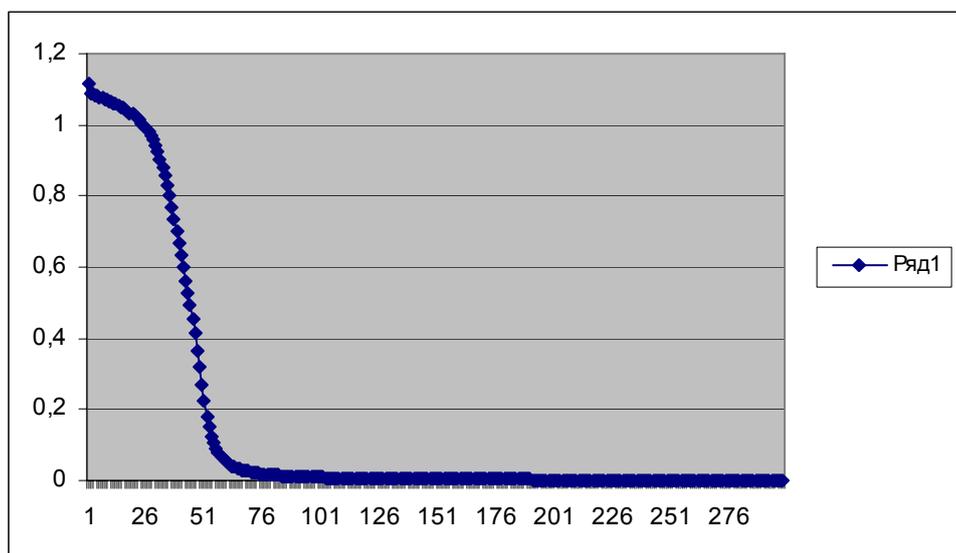


Рис. 6. Трехслойный персептрон. Среднеквадратическая ошибка.

В таблице приведены результаты работы нейронной сети с экспериментальным множеством данных. Нейронная сеть на экспериментальном множестве ошибается 1 раз. Таким образом, полученный результат показывает, что для блока оценки ответа в алгоритме отбора и предъявления заданий могут быть успешно использованы искусственные нейронные сети.

### **Литература**

1. Соловей Е. Методы и модели интеллектуального автоматизированного контроля знаний, 2001. КГТУ Калининград.
2. Саймон Хайкин Нейронные сети Полный курс 2-ое изд. 2006.- М. Спб. Киев Вильямс.
3. Короткий С. Электронная публикация. [http://serg-dobrinin.narod.ru/intelect/s\\_korotriy2.htm](http://serg-dobrinin.narod.ru/intelect/s_korotriy2.htm)
4. Рутковская Д., Пилинский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы; М. 2007