

Министерство образования Республики Беларусь  
Учреждение образования  
«Брестский государственный технический университет»  
Кафедра ИИТ

Лабораторная работа №2  
за 1 семестр  
По дисциплине: «МиАПР»  
Тема: «Линейная искусственная нейронная сеть. Адаптивный шаг обучения»

Выполнил:  
Студент 2 курса  
Группы ПО-4(1)  
Прокопюк Н.О.  
Проверил:  
Крощенко А.А.

## Лабораторная работа №2

### Линейная искусственная нейронная сеть. Адаптивный шаг обучения

Цель работы: изучить обучение и функционирование линейной ИНС с применением адаптивного шага.

#### Вариант 7

##### Задание:

Написать на любом ЯВУ программу моделирования прогнозирующей линейной ИНС. Для тестирования использовать функцию

$$y = a \cdot \sin(bx) + d$$

$a = 3$ ,  $b = 7$ ,  $d = 0.3$ , кол-во входов ИНС = 5.

Обучение и прогнозирование производить на 30 и 15 значениях соответственно табулируя функцию с шагом 0.1. Скорость обучения выбирается студентом самостоятельно, для чего моделирование проводится несколько раз для разных альфа. Результаты оцениваются по двум критериям - скорости обучения и минимальной достигнутой ошибке. Необходимо заметить, что эти критерии в общем случае являются взаимоисключающими, и оптимальные значения для каждого критерия достигаются при разных альфа.

Нейронная сеть представляет собой последовательность связанных нейронов. К нейрону поступают входящие сигналы, каждому из которых присвоен определенный вес. Сигнал умножается на свой вес, значения суммируются, и получается единое число, которое получает активационная функция. На выходе она принимает решение, транслировать ли сигнал дальше. Самая сложная задача в работе с нейросетью – грамотно подобрать коэффициенты к нейронам. Для этого используется обучение – процесс нахождения корректных весов для нейросети. От того, как именно обучают нейросеть, будут зависеть ее решения.

Для ускорения процедуры обучения вместо постоянного шага обучения предложено использовать адаптивный шаг обучения, который из возможных направлений выбирает наиболее оптимальный вариант.

##### Исходный код:

```
#include <iostream>
#include <iomanip>
#include <ctime>

using namespace std;

int main() {
    setlocale(0, "");
    int a = 3,
        b = 6,
        enteries = 3,
        n = 30,
        values = 15;

    double d = 0.1,
        Em = 0.00001,
        E,
        T = 1;

    double* W = new double[enteries];
    for (int i = 0; i < enteries; i++) {
        W[i] = (double)(rand()) / RAND_MAX;
        cout << "W[" << i << "] = " << W[i] << endl;
    }
}
```

```

cout << endl;

double* etalon_values = new double[n + values];
for (int i = 0; i < n + values; i++) {
    double step = 0.1;
    double x = step * i;
    etalon_values[i] = a * sin(b * x) + d;
}
int era = 0;

while (1) {
    double y1;
    double Alpha = 0.05;
    E = 0;

    for (int i = 0; i < n - enteries; i++) {
        y1 = 0;

        double temp = 0.0;
        for (int j = 0; j < enteries; j++) {
            temp += pow(etalon_values[i + j], 2);
        }
        Alpha = 1 / (1 + temp);

        for (int j = 0; j < enteries; j++) {
            y1 += W[j] * etalon_values[j + i];
        }
        y1 -= T;

        for (int j = 0; j < enteries; j++) {
            W[j] -= Alpha * (y1 - etalon_values[i + enteries]) * etalon_values[i +
j];

        }

        T += Alpha * (y1 - etalon_values[i + enteries]);
        E += 0.5 * pow(y1 - etalon_values[i + enteries], 2);
        era++;

    }

    cout << era << " | " << E << endl;
    if (E < Em) break;
}
cout << endl;

cout << "РЕЗУЛЬТАТЫ ОБУЧЕНИЯ" << endl;
cout << setw(27) << right << "Эталонные значения" << setw(23) << right << "Полученные
значения";
cout << setw(23) << right << "Отклонение" << endl;
double* prognos_values = new double[n + values];

for (int i = 0; i < n; i++) {
    prognos_values[i] = 0;
    for (int j = 0; j < enteries; j++) {
        prognos_values[i] += W[j] * etalon_values[j + i];
    }
    prognos_values[i] -= T;

    cout << "y[" << i + 1 << "] = " << setw(20) << right << etalon_values[i + enteries]
<< setw(23) << right;
    cout << prognos_values[i] << setw(23) << right << etalon_values[i + enteries] -
prognos_values[i] << endl;
}

cout << endl << "РЕЗУЛЬТАТЫ ПРОГНОЗИРОВАНИЯ" << endl;
cout << setw(28) << right << "Эталонные значения" << setw(23) << right << "Полученные
значения" << setw(23) << right << "Отклонение" << endl;

for (int i = 0; i < values; i++) {

```

```

        prognoz_values[i + n] = 0;
        for (int j = 0; j < enteries; j++) {
            prognoz_values[i + n] += W[j] * etalon_values[n - enteries + j + i];
        }
        prognoz_values[i + n] -= T;

        cout << "y[" << n + i + 1 << "] = " << setw(20) << right << etalon_values[i + n] <<
        setw(23) << right;
        cout << prognoz_values[i + n] << setw(23) << right << etalon_values[i + n] -
        prognoz_values[i + n] << endl;
    }

    delete[] etalon_values;
    delete[] prognoz_values;
    delete[] W;

    system("pause");
    return 0;
}

```

## Результат выполнения программы:

```

C:\Users\uzzzz\source\repos\ConsoleApplication1\Debug\ConsoleApplication1.exe
W[0] = 0.00125126
W[1] = 0.563585
W[2] = 0.193304

27 | 13.4957
54 | 0.103327
81 | 0.000866647
108 | 7.57515e-06

РЕЗУЛЬТАТЫ ОБУЧЕНИЯ
      Эталонные значения      Полученные значения      Отклонение
y[1] = 3.02154 3.02138 0.000163235
y[2] = 2.12639 2.12631 7.911e-05
y[3] = 0.52336 0.523362 -1.84794e-06
y[4] = -1.22756 -1.22751 -5.13584e-05
y[5] = -2.51473 -2.51468 -5.21259e-05
y[6] = -2.88849 -2.88849 -3.8823e-06
y[7] = -2.21829 -2.21837 7.65194e-05
y[8] = -0.738246 -0.738407 0.000160993
y[9] = 1.03462 1.0344 0.000220029
y[10] = 2.481 2.48077 0.000233004
y[11] = 3.09563 3.09543 0.000195387
y[12] = 2.6638 2.66368 0.000120317
y[13] = 1.33636 1.33632 3.4019e-05
y[14] = -0.42298 -0.422947 -3.33607e-05
y[15] = -1.99962 -1.99957 -5.82846e-05
y[16] = -2.84281 -2.84278 -3.2046e-05
y[17] = -2.65799 -2.65802 3.61891e-05
y[18] = -1.50972 -1.50984 0.000122584
y[19] = 0.200869 0.200672 0.000196959
y[20] = 1.87622 1.87599 0.000233333
y[21] = 2.93109 2.93087 0.000218998
y[22] = 2.99697 2.99681 0.000158963
y[23] = 2.05086 2.05079 7.42e-05
y[24] = 0.423261 0.423267 -5.68157e-06
y[25] = -1.31727 -1.31721 -5.27764e-05
y[26] = -2.5627 -2.56265 -5.0633e-05
y[27] = -2.87798 -2.87798 0
y[28] = -2.15296 -2.15304 8.1435e-05
y[29] = -0.640921 -0.641086 0.000165224
y[30] = 1.12994 1.12972 0.000222098

РЕЗУЛЬТАТЫ ПРОГНОЗИРОВАНИЯ
      Эталонные значения      Полученные значения      Отклонение
y[31] = -2.15296 -2.15304 8.1435e-05
y[32] = -0.640921 -0.641086 0.000165224
y[33] = 1.12994 1.12972 0.000222098
y[34] = 2.54102 2.54079 0.000232189
y[35] = 3.09938 3.09919 0.000191971
y[36] = 2.60997 2.60985 0.000115494
y[37] = 1.24375 1.24372 2.94741e-05
y[38] = -0.522009 -0.521973 -3.60402e-05
y[39] = -2.07048 -2.07043 -5.81626e-05
y[40] = -2.86075 -2.86072 -2.91652e-05
y[41] = -2.61674 -2.61678 4.08225e-05
y[42] = -1.42369 -1.42382 0.000127352
y[43] = 0.301624 0.301424 0.000200195
y[44] = 1.95651 1.95627 0.000233907
y[45] = 2.96286 2.96264 0.00021671

Для продолжения нажмите любую клавишу . . .

```

Вывод: Изучил обучение и функционирование линейной ИНС с использованием адаптивного шага.