Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

Лабораторная работа №1

за 1 семестр

По дисциплине: «МиАПР»

Тема: «Линейная искусственная нейронная сеть. Правило Видроу-Хоффа»

Выполнил:

Студент 2 курса

Группы ПО-4(1)

Калиновский В.Е.

Проверил:

Крощенко А.А.

Брест 2020

Лабораторная работа №1

Линейная искусственная нейронная сеть. Правило Видроу-Хоффа

Цель работы: изучить обучение и функционирование линейной ИНС при решении

задач прогнозирования.

Вариант 11

Задание:

Написать на любом ЯВУ программу моделирования прогнозирующей линейной

ИНС. Для тестирования использовать функцию

y = a\*sin(bx) + d

a = 3, b = 5, d = 0.5, кол-во входов ИНС = 4.

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.

Код программы:

#include <iostream>

#include <iomanip>

#include <ctime>

using namespace std;

int main() {

setlocale(0, "");

int a = 3,

b = 5,

enter = 4, //кол-во входов ИНС

N = 30, //количество значений для обучения

val = 15; //количество значений для прогноза

double d = 0.5,

Em = 0.05, //минимальная среднеквадратичная ошибка сети

E, //суммарная среднеквадратичная ошибка сети

T = 1; //порог нейронной сети

double\* W = new double[enter]; //весовые коэффициенты (3)

for (int i = 0; i < enter; i++) { //рандом весовых коэффициентов

W[i] = (double)(rand()) / RAND\_MAX;

//cout << "W[" << i << "] = " << W[i] << endl;

}

cout << endl;

double\* ref\_values = new double[N + val]; //эталонные значения y

for (int i = 0; i < N + val; i++) { //вычисляем эталонные значения

double step = 0.1; //шаг

double x = step \* i;

ref\_values[i] = a \* sin(b \* x) + d;

}

int count = 0; //для индексов

while (1) {

double y1; //выходное значение нейронной сети

double A = 0.04; //скорость обучения

E = 0; //ошибка

for (int i = 0; i < N - enter; i++) {

y1 = 0;

for (int j = 0; j < enter; j++) { //векторы выходной активности сети

y1 += W[j] \* ref\_values[j + i];

}

y1 -= T;

for (int j = 0; j < enter; j++) { //изменение весовых коэффициентов

W[j] -= A \* (y1 - ref\_values[i + enter]) \* ref\_values[i + j];

}

T += A \* (y1 - ref\_values[i + enter]); //изменение порога нейронной сети

E += 0.5 \* pow(y1 - ref\_values[i + enter], 2); //расчет суммарной среднеквадратичной ошибки

count++;

}

cout << count << " | " << E << endl;

if (E < Em) break;

} //далее сеть обучена

cout << endl;

cout << "Результаты обучения" << endl;

cout << setw(27) << right << "Эталонные значения" << setw(23) << right << "Полученные значения";

cout << setw(23) << right << "Отклонение" << endl;

double\* predict\_values = new double[N + val];

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

predict\_values[i] = 0;

for (int j = 0; j < enter; j++) {

predict\_values[i] += W[j] \* ref\_values[j + i]; //получаемые значения в результате обучения

}

predict\_values[i] -= T;

cout << "y[" << i + 1 << "] = " << setw(20) << right << ref\_values[i + enter] << setw(23) << right;

cout << predict\_values[i] << setw(23) << right << ref\_values[i + enter] - predict\_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 < val; i++) {

predict\_values[i + N] = 0;

for (int j = 0; j < enter; j++) {

//прогнозируемые значения

predict\_values[i + N] += W[j] \* ref\_values[N - enter + j + i];

}

predict\_values[i + N] -= T;

cout << "y[" << N + i + 1 << "] = " << setw(20) << right << ref\_values[i + N] << setw(23) << right;

cout << predict\_values[i + N] << setw(23) << right << ref\_values[i + N] - predict\_values[i + N] << endl;

}

delete[]ref\_values;

delete[]predict\_values;

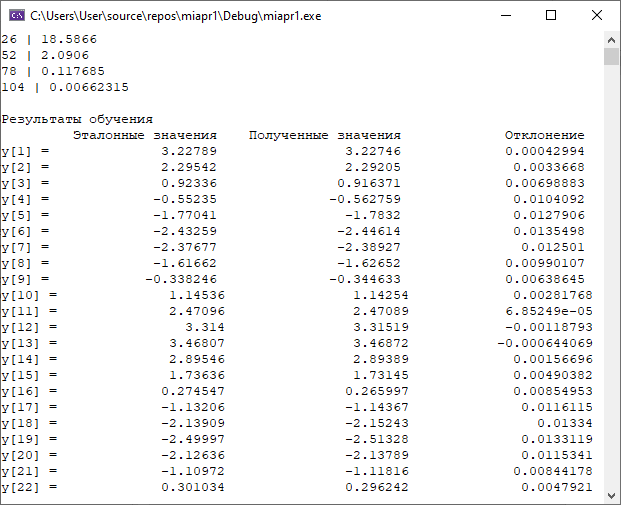
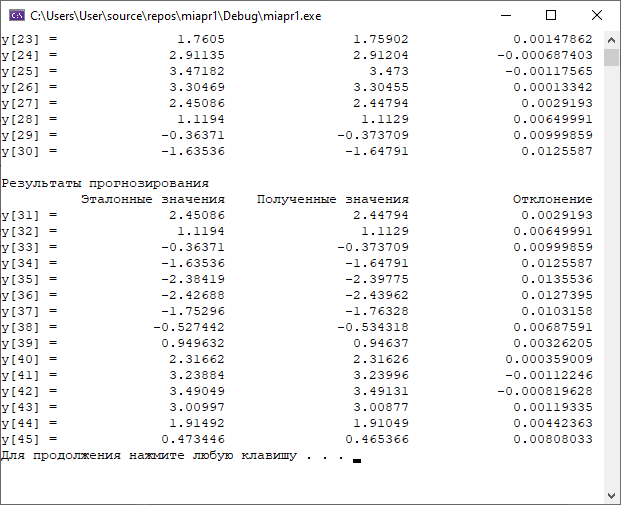
delete[]W;

system("pause");

return 0;

}

Результат выполнения:

Вывод: Изучил обучение и функционирование линейной ИНС при решении задач

прогнозирования.