Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

**Лабораторная работа №5**

**Тема: «Нелинейные ИНС в задачах распознавания образов»**

**Выполнил:**

Студент 4 курса

Факультета ЭИС

Группы АС-55

Черноокий И.В.

**Проверил:**

Савицкий Ю.В.

Брест 2022

**Лабораторная работа №5**

**«Нелинейные ИНС в задачах распознавания образов»**

**Цель работы:** Изучить обучение и функционирование нелинейной искусственной нейронной сети (ИНС) при решении задач распознавания образов.

**Задание:** Написать на любом ЯВУ программу моделирования нелинейной ИНС для распознавания образов. Рекомендуется использовать сигмоидную функцию, но это не является обязательным. Количество НЭ в скрытом слое взять согласно варианту работы №3. Его можно варьировать, если сеть не обучается или некорректно функционирует.

**Текст программы**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <iostream>

#include <math.h>

#include <time.h>

#include <windows.h>

#include <conio.h>

#define INPUT 30

#define HIDE 8

#define OUTPUT 10

#define alpha 0.1

#define E 0.4

using namespace std;

double w\_layer2[INPUT][HIDE], //Весовые коэффициенты между входным и скрытым слоем

T\_layer2[HIDE], //Пороговые значения НЭ скрытого слоя

w\_layer3[HIDE][OUTPUT], //Весовые коэффициенты связей между скрытым и выходным слоем

T\_layer3[OUTPUT], //Пороговые значения НЭ выходного слоя

y\_layer2[HIDE], //Выходные значения НЭ скрытого

y\_layer3[OUTPUT]; //Выходные значения НЭ выходного

char input\_buffer[INPUT+1]; //Образ, введенный с клавиатуры

double g\_layer2[HIDE]; //Ошибки нейронов скрытого

double g\_layer3[OUTPUT]; //Ошибки нейронов выходного

bool input\_sample[OUTPUT][INPUT] ={

{

0,1,1,1,0,

1,0,0,0,1,

1,0,0,0,1,

1,0,0,0,1,

1,0,0,0,1,

0,1,1,1,0

},

{

0,0,1,0,0,

0,1,1,0,0,

1,0,1,0,0,

0,0,1,0,0,

0,0,1,0,0,

1,1,1,1,1

},

{

0,1,1,1,0,

1,0,0,0,1,

0,0,0,1,0,

0,0,1,0,0,

0,1,0,0,0,

1,1,1,1,1

},

{

0,1,1,1,0,

1,0,0,0,1,

0,0,1,1,0,

0,0,1,1,0,

1,0,0,0,1,

0,1,1,1,0

},

{

0,0,0,1,0,

0,0,1,1,0,

0,1,0,1,0,

1,1,1,1,1,

0,0,0,1,0,

0,0,0,1,0

},

{

1,1,1,1,1,

1,0,0,0,0,

1,1,1,1,0,

0,0,0,0,1,

0,0,0,0,1,

1,1,1,1,0

},

{

0,1,1,1,0,

1,0,0,0,0,

1,1,1,1,0,

1,0,0,0,1,

1,0,0,0,1,

0,1,1,1,0

},

{

0,1,1,1,1,

0,0,0,0,1,

0,0,0,1,0,

0,0,1,0,0,

0,0,1,0,0,

0,0,1,0,0

},

{

0,1,1,1,0,

1,0,0,0,1,

0,1,1,1,0,

1,0,0,0,1,

1,0,0,0,1,

0,1,1,1,0

},

{

0,1,1,1,0,

1,0,0,0,1,

1,0,0,0,1,

0,1,1,1,0,

0,0,0,0,1,

0,1,1,1,0

}};

FILE \*fo;

//Возвращает определённый класс образа

int out\_network(char temp[]){

//Считаем выходные значения скрытого слоя

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

y\_layer2[i] = 0;

for (int j = 0; j < INPUT; j++)

if (temp[j])

y\_layer2[i] += w\_layer2[j][i];

y\_layer2[i] -= T\_layer2[i];

y\_layer2[i] = 1 / (1 + exp (- y\_layer2[i]));

}

//Считаем выходные значения выходного слоя

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

y\_layer3[i] = 0;

for (int j = 0; j < HIDE; j++)

y\_layer3[i] += y\_layer2[j] \* w\_layer3[j][i];

y\_layer3[i] -= T\_layer3[i];

}

//Определяем класс входного образа

double max\_val = y\_layer3[0];

int result = 0;

for (int i = 1; i < OUTPUT; i++)

if (y\_layer3[i] > max\_val){

max\_val = y\_layer3[i];

result = i;

}

return result;

}

//Распознавание строки

int recognition(char temp[]){

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

if (temp[i] == '0')

input\_buffer[i] = 0;

else

input\_buffer[i] = 1;

}

return out\_network(input\_buffer);

}

int main(){

fo = fopen("error.txt","wt");

setlocale (0,"");

srand(time(NULL));

/// Случайно задаем веса и пороги

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

for (int j = 0; j < INPUT; j++)

w\_layer2[j][i] = double (rand() - 16384) / 16384;

for (int j = 0; j < OUTPUT; j++)

w\_layer3[i][j] = double (rand() - 16384) / 16384;

T\_layer2[i] = double (rand() - 16384) / 16384;

}

for (int i = 0; i < OUTPUT; i++)

T\_layer3[i] = double (rand() - 16384) / 16384;

/// Вывод входных значений

for (int k = 0; k < 6; k++) {

for (int i = 0; i < 10; i++) {

for (int j = 0; j < 5; j++) {

if (input\_sample[i][j + k \* 5] == 0)

printf (" ");

else

printf ("%d", input\_sample[i][j + k \* 5]);

}

printf (" ");

}

printf ("\n");

}

/// Обучение

int counter = 0;

bool flag = 1; //Признак окончания обучения

while (flag){

flag = 0;

//Для каждой обучающей выборки:

for (int k = 0; k < OUTPUT; k++){

//Формируем обучающую выборку

for (int i = 0; i < INPUT; i++)

input\_buffer[i] = input\_sample [k][i];

//Получаем выходные значения нейронов

out\_network(input\_buffer);

//Считаем ошибки нейронов выходного слоя

double E\_sqrt = 0;

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

if (i == k)

g\_layer3[i] = y\_layer3[i] - 1;

else

g\_layer3[i] = y\_layer3[i];

E\_sqrt += pow(g\_layer3[i], 2);

}

fprintf(fo, "%f\n", E\_sqrt);

//Если одна из них больше допустимой, переходим к следующей итерации обучения

for (int i = 0; i < OUTPUT; i++)

if (abs(g\_layer3[i]) >= E)

flag = 1;

//Считаем ошибки нейронов скрытого

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

g\_layer2[i] = 0;

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

g\_layer2[i] += g\_layer3[j] \* y\_layer3[j] \* ( 1 - y\_layer3[j] ) \* w\_layer3[i][j];

}

}

//Пересчитываем значения весовых коэффициентов и пороговых значений скрытого слоя

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

for (int i = 0; i < INPUT; i++)

w\_layer2[i][j] -= alpha \* g\_layer2[j] \* y\_layer2[j] \* ( 1 - y\_layer2[j] ) \* input\_buffer[i];

T\_layer2[j] += alpha \* g\_layer2[j] \* y\_layer2[j] \* ( 1 - y\_layer2[j] );

}

//Пересчитываем значения весовых коэффициентов и пороговых значений выходного слоя

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

for (int i = 0; i < HIDE; i++)

w\_layer3[i][j] -= alpha \* g\_layer3[j] \* y\_layer2[i];

T\_layer3[j] += alpha \* g\_layer3[j];

}

counter++;

}

}

printf("Обучено за %d итераций\n", counter);

///Распознавание образов

while (true){

printf("Введите образ для распознавания: ");

gets\_s(input\_buffer);

printf("Распознанный образ - число %d\n", recognition(input\_buffer));

}

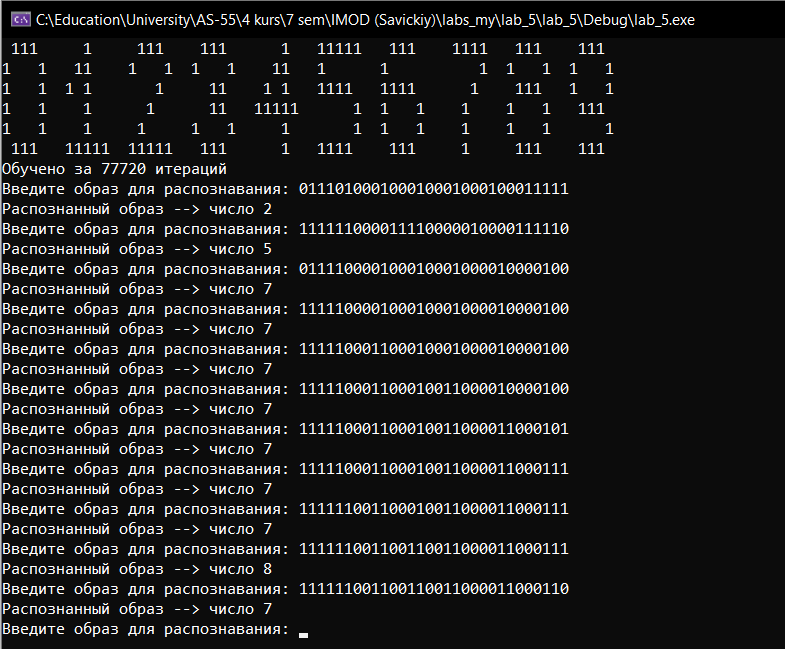
fclose(fo);

return 0;

}

**Результаты работы**

Образ для 7 ввели изначально без помех, затем помехи в одном бите, затем 2, 3, 5, 6, 7. Распознало корректно. Но когда различие было в 8 битах, то н.с. распознала ошибочный образ.



**Вывод:** изучили обучение и функционирование нелинейной ИНС при решении задач распознавания образов.