Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

Лабораторная работа №3

за 1 семестр

По дисциплине: «МиАПР»

Тема: «Нелинейные ИНС в задачах прогнозирования»

Выполнил:

Студент 2 курса

Группы ПО-4(1)

Грибовский Д.С.

Проверил:

Крощенко А.А.

2020

**Лабораторная работа №3**

Нелинейные ИНС в задачах прогнозирования

Цель работы: Изучить обучение и функционирование нелинейной ИНС при решении задач прогнозирования.

**Вариант 6**

**Задание:**

**Код программы:**

import math

import random

def func(x):

a = 0.2

b = 0.6

c = 0.05

d = 0.6

return a \* math.cos(b \* x) + c \* math.sin(d \* x)

def sigmoid(x):

e = 2.718

return 1 / (1 + e \*\* (-x))

def derivative(x):

return x \* (1 - x)

n\_in = 10 #количество входов

n\_el = 4 #количество элементов скрытого слоя

n\_ob = 30 #размер выборки для обучения

n\_pr = 15 #размер выборки для прогназированния

alpha = 0.1 #скорость обучения

Em = 1e-6 #минимальная ошибка

Err = [0 for i in range(n\_ob - n\_in)]

E\_hid = [] #ошибка скрытого слоя

F\_hid = [] #функция активности скрытого слоя

Y\_hid = [] #выходная активность скрытого слоя, выходного слоя

T\_hid = [] #пороговые значение скрытого слоя, выходного слоя

W\_hid = [[0] \* n\_el for i in range(n\_in)] #весовые коэффициенты скрытого слоя

W\_out = [] #весовые коэффициенты выходного слоя

Y\_out = 0.0

t = []; #массив эталонных значений

y = []; #массив обучаемых значений

z = []; #массив прогнозируемых значений

for i in range(n\_pr + n\_ob + n\_in):

t.append(func(i \* 0.1))

T\_out = random.uniform(-0.1, 0.1)

print("Случайно заданные весовые коэффициенты: ")

for j in range(n\_el):

Y\_hid.append(0)

F\_hid.append(0)

for i in range(n\_in):

W\_hid[i][j] = round(random.uniform(-0.1, 0.1), 2)

W\_out.append(round(random.uniform(-0.1, 0.1), 2))

T\_hid.append(round(random.uniform(-0.1, 0.1), 2))

print(W\_hid)

print("\n")

count = 0

while True:

count += 1

E = 0.0

for k in range(n\_ob - n\_in):

Y\_out = 0.0

for i in range(n\_el):

for j in range(n\_in):

Y\_hid[i] += W\_hid[j][i] \* t[k + j]

Y\_hid[i] -= T\_hid[i]

F\_hid[i] = sigmoid(Y\_hid[i])

Y\_hid[i] = 0.0

Y\_out += F\_hid[i] \* W\_out[i]

Y\_out -= T\_out;

Err[k] = Y\_out - t[k];

T\_out += alpha \* Err[k];

for i in range(n\_el):

W\_out[i] -= alpha \* Err[k] \* F\_hid[i]

for i in range(n\_el):

deriv = derivative(F\_hid[i])

for j in range(n\_in):

W\_hid[j][i] -= alpha \* Err[k] \* deriv \* W\_out[i] \* t[k + j]

T\_hid[i] += alpha \* Err[k] \* deriv \* W\_out[i]

E += 0.5 \* Err[k] \*\* 2

#print(E," ",Em)

if E < Em:

break

print("Эпохи: ", count)

print("Ошибка", E)

#Обучение

print("Результаты обучения:")

print(" %2s %2s %2s %2s " % (

"y[]",

"Эталонное значение",

"Полученное значение",

"Отклонение"

))

for k in range(n\_ob):

y.append(0)

for i in range(n\_el):

for j in range(n\_in):

y[k] += W\_hid[j][i] \* t[k + j]

Y\_hid[i] -= T\_hid[i];

F\_hid[i] = sigmoid(Y\_hid[i]); #функция активации

Y\_hid[i] = 0;

y[k] += F\_hid[i] \* W\_out[i];

y[k] -= T\_out

print(" %2d %9lf %18lf %19lf " % (

k,

t[k],

y[k],

t[k] - y[k]

))

#Прогнозированние

print("Результаты прогнозирования:")

print(" %2s %2s %2s %2s " % (

"y[]",

"Эталонное значение",

"Полученное значение",

"Отклонение"

))

for k in range(n\_pr):

z.append(0)

for i in range(n\_el):

for j in range(n\_in):

Y\_hid[i] += W\_hid[j][i] \* t[k + n\_ob + j]

Y\_hid[i] -= T\_hid[i];

F\_hid[i] = sigmoid(Y\_hid[i]); # функция активации

Y\_hid[i] = 0;

z[k] += F\_hid[i] \* W\_out[i];

z[k] -= T\_out

print(" %2d %9lf %18lf %19lf " % (

k + n\_ob,

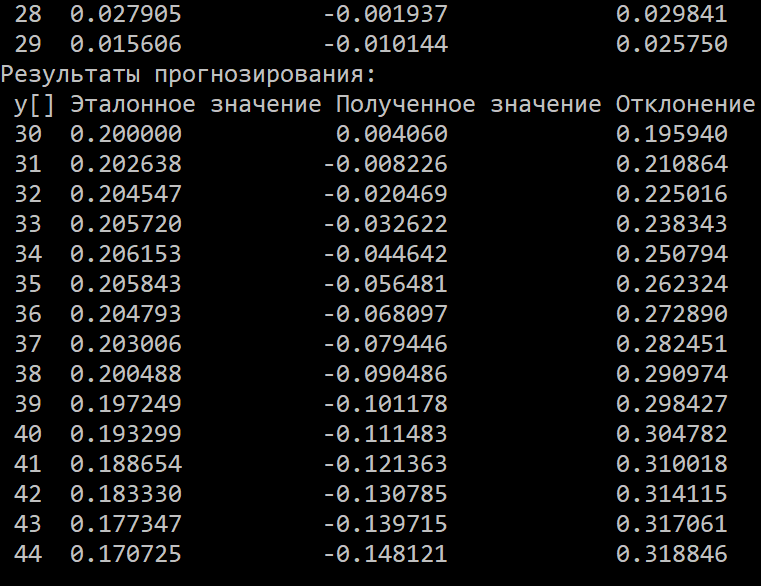
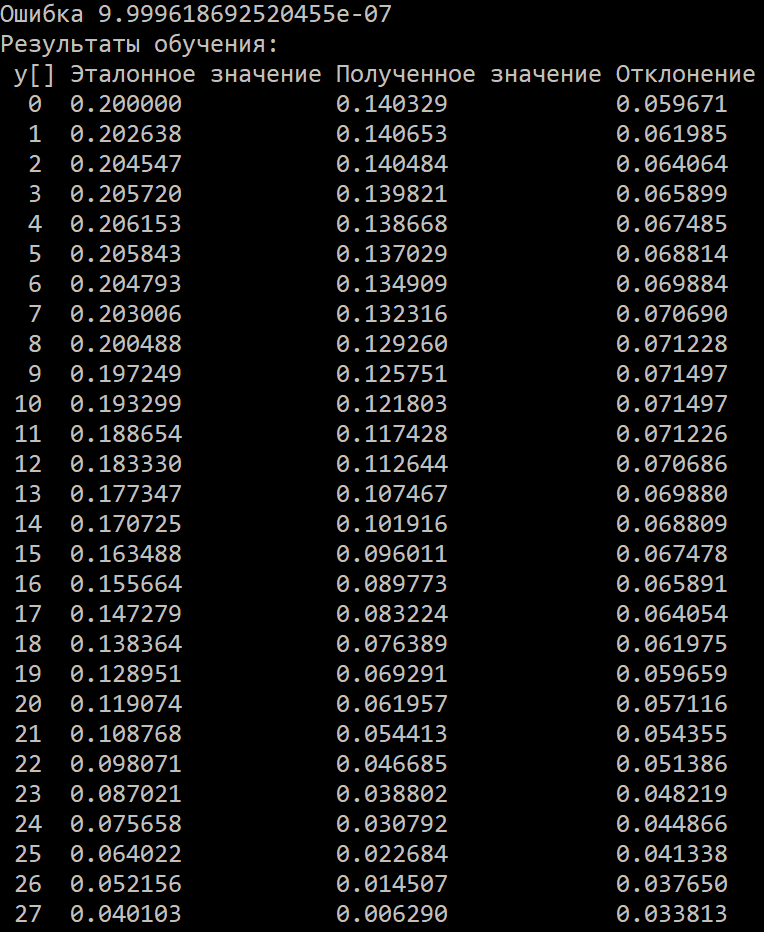
t[k],

z[k],

t[k] - z[k]

))

Результат:



Вывод: В ходе выполнения работы спроектировал линейную ИНС с использованием адаптивного шага обучения.