Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

Лабораторная работа №4

По дисциплине: «МиАПР»

Тема: «Изучить обучение и функционирование нелинейной ИНС при решении задач распознавания образов.»

Вариант №1

Выполнил:

Студент 2 курса

Группы ПО-7

Белев С.В.

Проверил:

Крощенко А.А.

Брест 2021

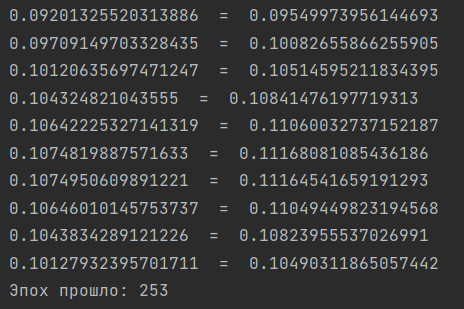
**Цель работы:** изучить обучение и функционирование нелинейной ИНС при решении задач распознавания образов.

**Задание:** Спрогнозировать нелинейный временной ряд, применяя параметры лабораторной работы №3. При этом необходимо использовать алгоритм обучения многослойной ИНС с адаптивным шагом.

**Код программы:**

from math import \*  
from random import random as r  
  
  
def formula(x):  
 return 0.1 \* cos(0.1 \* x) + 0.05 \* sin(0.1 \* x)  
  
  
def sigm(x):  
 return 1 / (1 + exp(-x))  
  
  
def sigm\_der(x):  
 return x \* (1 - x)  
  
  
x\_list = [formula(i) for i in range(55)]  
e\_list = [formula(i) for i in range(6, 56)]  
test\_list = [formula(i) for i in range(56, 73)]  
Wki = [[r() for i in range(6)], [r() for i in range(6)]]  
Wij = [r() for i in range(2)]  
T = [r() for i in range(3)]  
h\_Summ = [0, 0]  
ls = 0.1  
Errs = [0, 0]  
MSE = 0  
x\_summ = 0  
Age = 0  
  
def hcalc(Wki, T, model, i, x\_list):  
 res = 0  
 for ne in range(6):  
 res += Wki[i][ne] \* x\_list[ne + model]  
 return sigm(res - T[i])  
  
  
def ocalc(Wij, T, h\_Summ):  
 return Wij[0] \* h\_Summ[0] + Wij[1] \* h\_Summ[1] - T[2]  
  
  
while True:  
 Age += 1  
 for model in range(50):  
  
  
  
 for i in range(2):  
 h\_Summ[i] = hcalc(Wki, T, model, i, x\_list)  
 strcalc = ocalc(Wij, T, h\_Summ)  
 Err = strcalc - e\_list[model]  
 MSE += Err \*\* 2  
  
 for i in range(2):  
 Errs[i] = Err \* Wij[i]  
  
 for i in range(6):  
 x\_summ += x\_list[i + model] \*\* 2  
 temp1 = ((Errs[0] \*\* 2) \* sigm\_der(h\_Summ[0])) + ((Errs[1] \*\* 2) \* sigm\_der(h\_Summ[1]))  
 temp2 = (Errs[0] \* sigm\_der(h\_Summ[0]))\*\*2 + (Errs[1] \* sigm\_der(h\_Summ[1]))\*\*2  
 hls = (4 \* temp1) / ((1 + x\_summ) \* temp2)  
  
 ls = ((Err\*\*2) / (1 + Err \*\* 2))  
  
 for i in range(2):  
 Wij[i] = Wij[i] - ls \* Err \* h\_Summ[i]  
 T[2] = T[2] + ls \* Err  
  
 for i in range(2):  
 for k in range(6):  
 Wki[i][k] = Wki[i][k] - hls \* Errs[i] \* sigm\_der(h\_Summ[i]) \* x\_list[k+model]  
 for i in range(2):  
 T[i] = T[i] + hls \* Errs[i] \* sigm\_der(h\_Summ[i])  
 if MSE < 10e-5:  
 break;  
 MSE = 0  
 x\_summ = 0  
for test\_model in range(10):  
 for i in range(2):  
 h\_Summ[i] = hcalc(Wki, T, test\_model, i, test\_list)  
 strcalc = ocalc(Wij, T, h\_Summ)  
 print(strcalc, " = ", test\_list[6 + test\_model])  
print("Эпох прошло:", Age)

**Результат работы программы:**



**Вывод:** Количество эпох, необходимых на обучение заметно снизилось (пусть это так-же зависит от начальной инициализации, но результат действительно значительный)