Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

Лабораторная работа №1

По дисциплине: «Методы и алгоритмы принятие решиений»

Вариант №11

Выполнил:

Студент 2-го курса

Группы ПО-7

Лобан К.Ю.

Проверил:

Крощенко А.А.

Брест 2021

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

***Задание.***

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

**Код программы:**

from math import sin

from random import uniform

def function(s):

return 3 \* sin(5 \* s) + 0.5

def y\_calculating(w, x, T):

s = 0

for j in range(4):

s += w[j] \* x[j]

return s - T

alpha = 0.05 # Скорость обучения

min\_error = 1e-28 # Желанная минимальная ошибка

xe\_train = [function(i / 10) for i in range(34)] # Эталонные значения для обучения

xe\_test = [function(i / 10) for i in range(30, 49)] # Эталонные значения для тестирования

w = [uniform(0, 1) for \_ in range(4)] # Создание случайных весов

T = uniform(0, 1) # Создание случайного порога

repeat = 0 # Счетчик повторений

error = 1 # Среднеквадратичная ошибка = 1, чтобы пройти первый while

# Обучение

while error >= min\_error and repeat < 100: # Остановка по достижению мин. ошибки или 100-ого повтора

error = 0

repeat += 1

for epoch in range(30): # 30 выборок

y = y\_calculating(w, xe\_train[epoch:epoch + 4], T) # Вычисление (y) нейронной сетью

e\_out: float = xe\_train[epoch + 4] # Эталонное значение этой выборки

delta: float = y - e\_out # Разница

error += delta \*\* 2 / 2 # Подсчет среднеквадратичной ошибки

for t in range(4): # Изменение всех весов и порога

w[t] -= alpha \* delta \* xe\_train[epoch + t]

T += alpha \* delta

# Конец обучения и начало тестирования

print(f'Нейросеть обучена за {repeat} повтор.')

print('Тестирование:')

print(' N: Идеальное значение Полученное значение Разница Среднекв. ошибка')

for epoch in range(15):

y = y\_calculating(w, xe\_test[epoch:epoch + 4], T) # Вычисление (y) нейронной сетью

e\_out: float = xe\_test[epoch + 4] # Эталонное значение

delta: float = y - e\_out # Разница

error += delta \*\* 2 / 2 # Подсчет среднеквадратичной ошибки

print(f'{epoch + 1:2}: {e\_out:21} {y:21} {delta:24} {error:24}')

**Полученные результаты:**



**Вывод:** я изучил обучение и функционирование линейной ИНС при решении задач прогнозирования.