Министерство образования Республики Беларусь

Учреждение образования

«Брестский Государственный технический университет»

Кафедра ИИТ

**Лабораторная работа №1**

По дисциплине «Модели решения задач в интеллектуальных системах»

Тема: «Бинарная классификация»

**Выполнил:**

Студент 3 курса

Группы ИИ-21

Литвинюк Т. В.

**Проверил:**

Туз И. С.

Брест 2023

**Цель:** реализовать однослойную нейронную сеть для решения задачи классификации с использованием пороговой функции активации.

**Ход работы**

**Вариант 7**

|  |  |  |
| --- | --- | --- |
| **x1** | **x2** | **e** |
| **4** | **4** | **-9** |
| **4** | **7** | **-9** |
| **7** | **4** | **-3** |
| **7** | **7** | **-3** |

import numpy as np

class Network:

def \_\_init\_\_(self) -> None:

self.W = np.array([1.0, 0.0])

self.T = lambda x: -3 if x >= 5.5 else -9

self.a = 0.5

def train(self, input\_data: np.array, reference\_data: np.array):

epoch = 0

while True:

epoch += 1

E = 0

for input, reference in zip(input\_data, reference\_data):

output = self.T(input @ self.W)

error = output - reference

self.W -= self.a \* input \* (error)

E += abs(error)

if E == 0:

break

print(f"Всего эпох {epoch}")

def sort(self, input):

return self.T(input @ self.W)

input\_data = np.array([[4, 4], [4, 7], [7, 4], [7, 7]])

reference = np.array([-9, -9, -3, -3])

NN = Network()

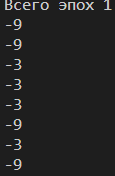
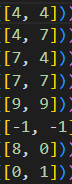
NN.train(input\_data, reference)

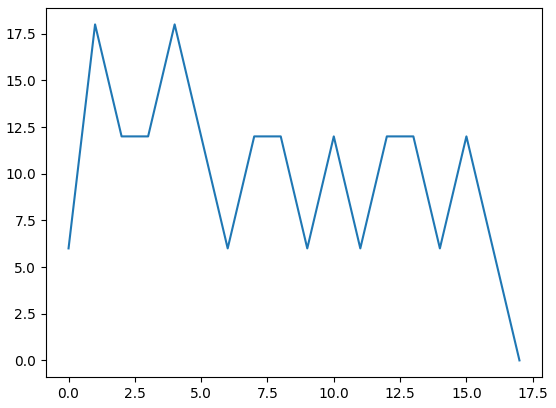
print(NN.sort(np.array([4, 4])))

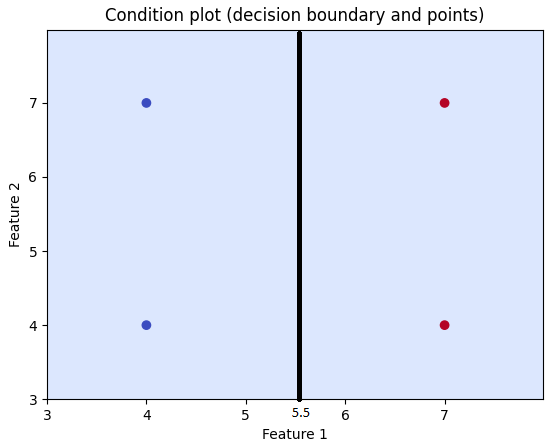
print(NN.sort(np.array([4, 7])))

print(NN.sort(np.array([7, 4])))

print(NN.sort(np.array([7, 7])))

Нейросеть обучилась на 18 эпохе. В ходе тестирования ошибок не было замечено.

График изменения ошибки:



**Вывод:** в ходе лабораторной работы я научился реализовывать однослойную нейронную сеть для решения задачи классификации с использованием пороговой функции активации.