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

Пучинский А.А.

**Проверил:**

Туз И. С.

Брест 2023

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

**Ход работы**

**Вариант 1**

|  |  |  |
| --- | --- | --- |
| **X1** | **X2** | **E** |
| **0** | **0** | **0** |
| **0** | **1** | **0** |
| **1** | **0** | **0** |
| **1** | **1** | **1** |

import numpy as np

# Входные данные

X = np.array([[0, 0], [0, 1], [1, 0], [1, 1]])

# Ожидаемые выходы

reference = np.array([0, 0, 0, 1])

# Инициализация весов и порога

weights = np.array([1.0, 1.0]) # Инициализируем веса случайным образом

bias = 2 # Инициализируем порог

# Функция пороговой активации

def threshold\_activation(x):

return 1 if x >= 0 else 0

# Процесс обучения (подбор весов)

learning\_rate = 0.1

epochs = 1000

for epoch in range(epochs):

for i in range(len(X)):

input\_data = X[i]

net\_input = np.dot(input\_data, weights) - bias

predicted\_output = threshold\_activation(net\_input)

error = reference[i] - predicted\_output

weights += learning\_rate \* error \* input\_data

bias += learning\_rate \* error

def predict(input\_data):

for input\_data\_ in input\_data:

net\_input = np.dot(input\_data\_, weights) - bias

predicted\_output = threshold\_activation(net\_input)

print(f"Input: {input\_data\_}, Predicted Output: {predicted\_output}")

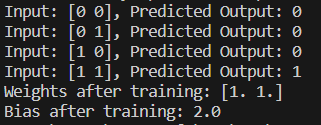
predict(X)

# Веса и порог после обучения

print(f"Weights after training: {weights}")

print(f"Bias after training: {bias}")

В ходе тестирования ошибок не было замечено.



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