Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**(ФГБОУ ВО «КубГУ»)**

**Факультет компьютерных технологий и прикладной математики**

**Кафедра вычислительных технологий**

**ЛАБОРАТОРНАЯ РАБОТА №1**

**Дисциплина: Приложение нейросетевых алгоритмов**

Работу выполнила: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В. С. Паничева

Направление подготовки: 02.03.03 Математическое обеспечение и администрирование информационных систем

Преподаватель: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В. И. Шиян

**Тема:** задачи классификации, многослойные нейронные сети

**Цель работы:** получение знаний и практических навыков построения однослойных нейронных сетей.

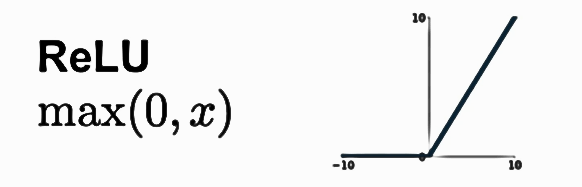
**Задание:** требуется разработать алгоритмы и программы решения задач 1–3 на языке Python без использования специализированных библиотек.

**Задача 1.** Пусть имеется функция XOR, в которой две точки {(0, 0), (1, 1)} принадлежат к одному классу, а две другие точки {(1, 0), (0, 1)} – к другому. Покажите, как разделить два этих класса, используя функцию активации ReLU.

Точки можно разделить на два класса (класс 0 и класс 1) с использованием операции XOR:



Функция ReLU возвращает входное значение, если оно положительное, в нашем случае 1, иначе возвращает 0.

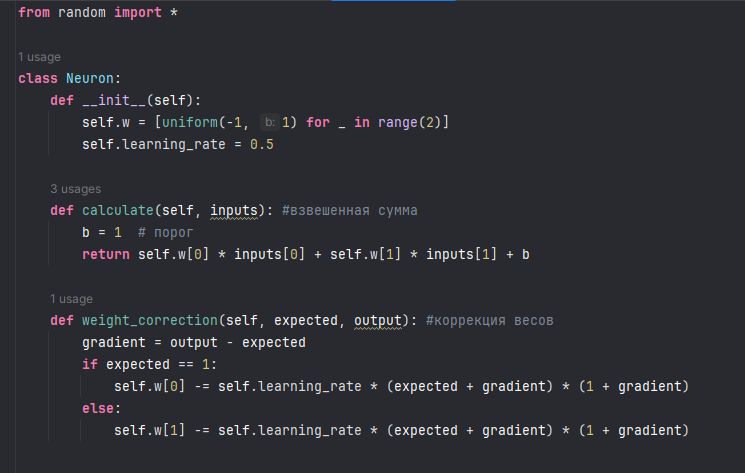


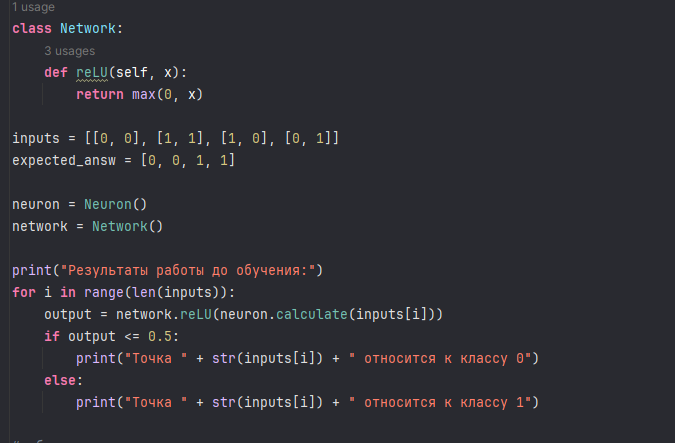
В данной программе реализован простая нейросеть с функцией активации ReLU, которая обучается на примере задачи XOR. В процессе обучения она пытается разделить два класса XOR, используя функцию активации ReLU. В функцию ReLU значения передаются в следующем виде:

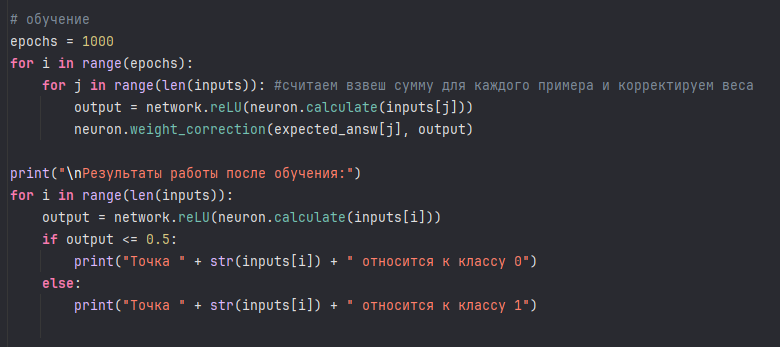
где w – веса, x – входы, b – порог

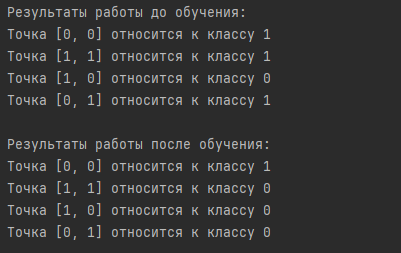
Класс Neuron содержит методы для инициализации весов, вычисления взвешенной суммы и коррекции весов нейрона.

В классе Network определена функция reLU, которая возвращает максимум из нуля и значения аргумента. Далее идет определение входных данных inputs и ожидаемых ответов expected\_answ, создается экземпляр нейрона и сети. Затем происходит обучение нейрона на протяжении 1000 эпох путем вычисления выхода нейрона и коррекции весов.





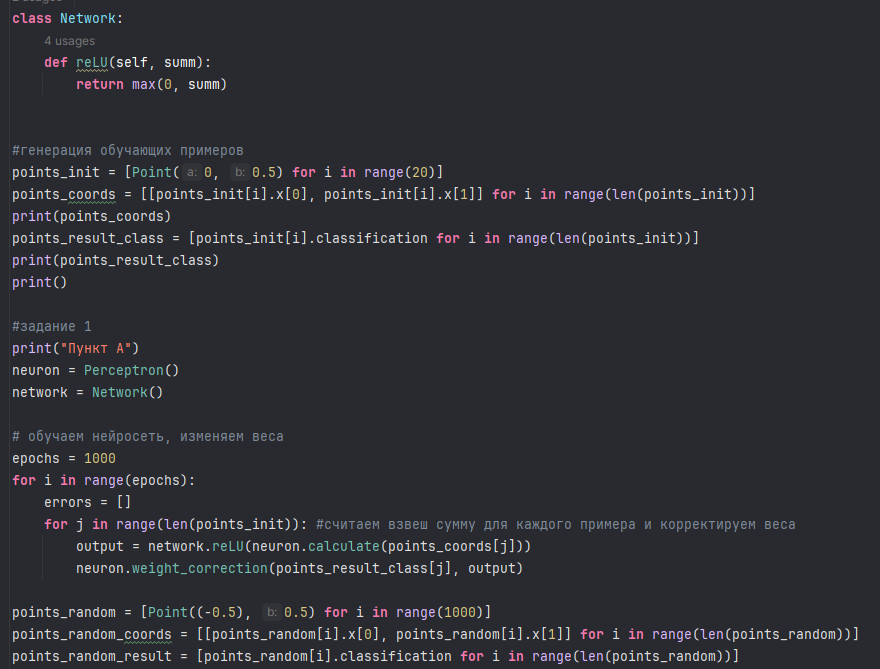




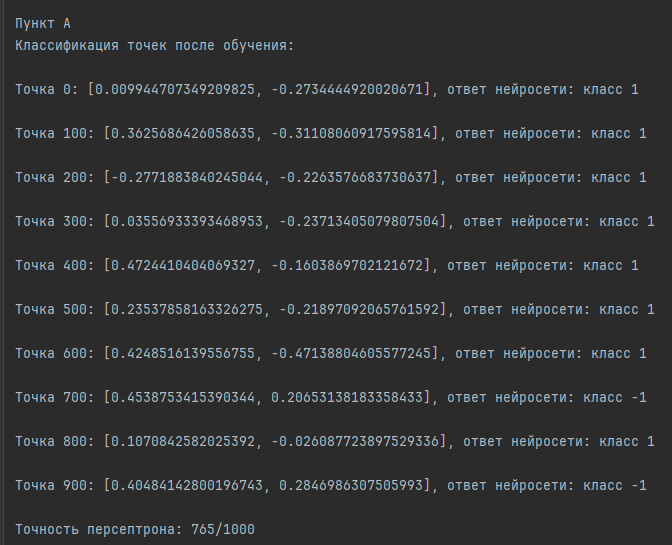
**Задача 2.** Пусть имеется двухмерный набор данных, в котором все точки с x1 > x2 принадлежат к положительному классу, а все точки с x1 < x2 к отрицательному. Разделителем для этих двух классов является линейная гиперплоскость (прямая линия), определяемая уравнением x1 − x2 = 0. Создайте набор тренировочных данных с 20 точками, сгенерированными случайным образом в положительном квадранте единичного квадрата. Снабдите каждую точку меткой, указывающей на то, превышает или не превышает её первая координата x1 вторую координату x2.

А. Реализуйте алгоритм перцептрона, обучите его на полученных выше 20 точках и протестируйте его точность на 1000 точках, случайно сгенерированных в единичном квадрате. Используйте для генерирования тестовых точек ту же процедуру, что и для тренировочных.

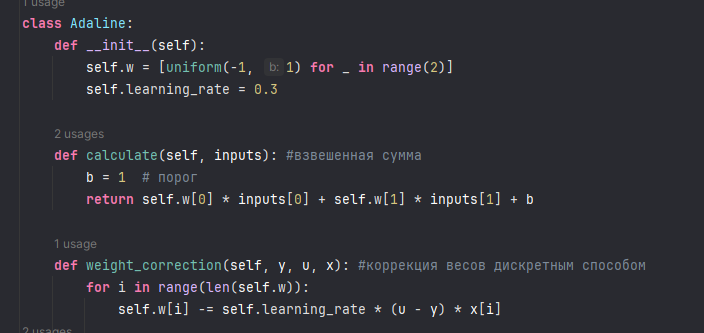




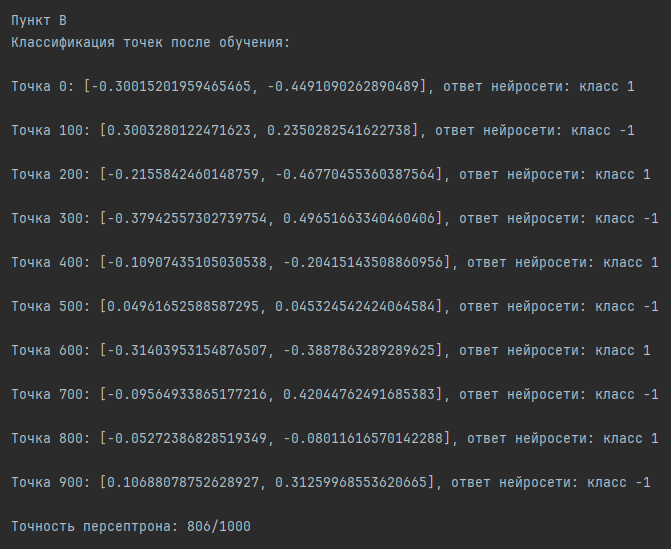




Б. Замените критерий перцептрона на нейрон типа адалайн (рассмотреть дискретный случай) при реализации тренировки и повторите определение точности вычислений на тех же тестовых точках, которые использовали перед этим. Удалось ли вам в каком-то из способов получить лучшую точность? Как вы считаете, в каком случае классификация тех же 1000 тестовых точек не изменится значительно, если использовать другой набор из 20 тренировочных точек?







**Задача 3.** Требуется разработать и исследовать нейронную сеть обратного распространения, предназначенную для распознавания образов.

Даны в виде матрицы 3 х 3 4 латинские буквы X, Y, L, I.

Требуется:

1. Построить и обучить нейронную сеть, которая могла бы решать задачу распознавания символов.

2. Произвести тестирование нейронной сети при добавлении шума.