**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**КАФЕДРА САПР**

**ОТЧЕТ**

**по Лабораторной работе № 4**

**«НЕЙРОННЫЕ СЕТИ»**

**по дисциплине «Автоматизация схематического проектирования»**

|  |  |  |
| --- | --- | --- |
| Студенты гр. 0302 |  | Сухарев Л. А.. |
| Преподаватель |  | Боброва Ю. О. |

Санкт-Петербург

2024

**Цель:** создание простейшей нейронной сети на Python без использования специализированных библиотек

1. **ОСНОВЫНЕ ТЕОРЕТИЧЕСКИЕ ПОЛОЖЕНИЯ**

Простейшая нейронная сеть состоит из слоя искусственных нейронов (ИН). ИН по своей сути представляет из себя алгоритм вычисления выхода из входных данных по известной формуле, с использованием набора весовых коэффициентов w и смещений b.

На вход нейрона поступают сигналы xi, каждый умножается на соответствующий коэффициент wi и суммируется (Рис.1). Полученная взвешенная сумма поступает на функцию активации, результат вычисления которой и является выходом нейрона

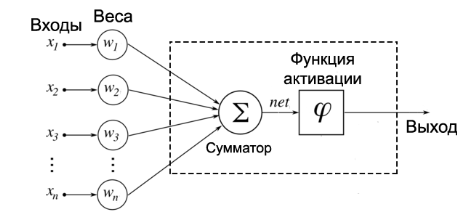


Рис. 1 — Структура искусственного нейрона

Простейшей функцией активации является функция единичного скачка, принимающая только два значения – 0, если net ниже порогового значения, 1 если превышает. Самой распространенной функцией активации является сигмоида, с которой вы уже знакомы. На выходе нейрон отдает значение от 0 до 1. Существует множество других видов функций активации, с которыми вы познакомитесь в рамках курса.

Нейронные сети состоят из набора нейронов, сгруппированных по слоям. В однослойных нейронных сетях сигналы с входного слоя сразу подаются на выходной слой (Рис.2). Он производит необходимые вычисления, результаты которых сразу подаются на выходы.

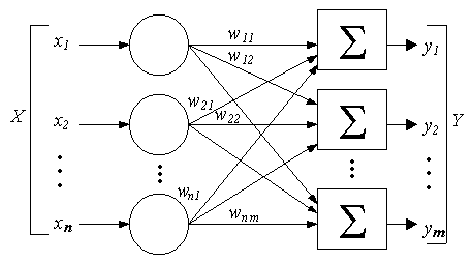


Рис. 2 — Однослойная нейронная сеть

Входы не считаются за входной слой сети и обозначены кружками. Справа (квадраты) расположены нейроны основного слоя.

Под обучением нейронной сети понимается поиск такого набора весовых коэффициентов, при котором входной сигнал после прохода по сети преобразуется с минимальной ошибкой в нужный нам выходной.

Используемые библиотеки: numpy, matplotlib

Дерево решений (распознающее дерево, recognition или decision tree) –

1. **КОД ПРОГРАММЫ. ПОЯСНЕНИЕ ЭТАПОВ**

Для начала создадим ряд функций для основных вычислений, происходящих внутри сети. Активационную функцию используем сигмоиду. Для расчетов также потребуется производная сигмоиды. (Рис.2)

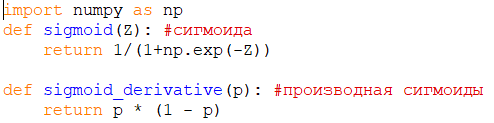


Рис. 3 — Функции для основных вычислений

Сама нейронная сеть будет представлять из себя класс с набором методов – инициализация (init), прямое распространение (feedforward), обратное распространение (backprop) и обучение (train). Также для дальнейших расчетов и для определения точности и прочих свойств работы нейронной сети создадим тестовый шаг (feedforward\_test) (Рис.4)

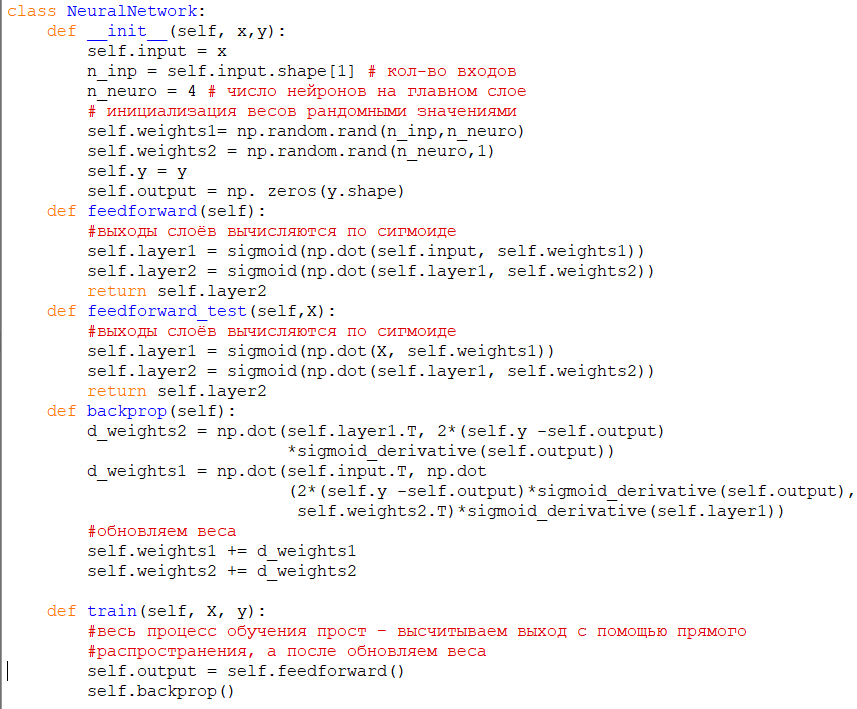


Рис. 4 — Нейронная сеть

Чтобы запустить созданную сеть сгенерируйте выборку, используем ранее написанный генератор данных. Единственное, надо будет изменить размер массива меток Y и из плоского сделать его одномерным. Чтобы точнее оценить работу НС разделим выборку на обучение и тест. (Рис.5)

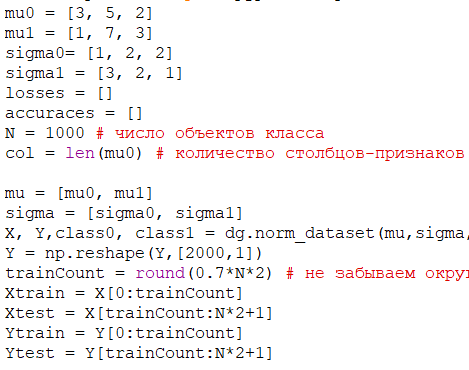


Рис. 5 — Подготовка данных

В цикле обучим сеть на достаточном количестве эпох. Для дальнейшего построения графиков в зависимости от номера эпохи, запомним на каждой эпохе значение точности и среднеквадратичной потери. (Рис.6)

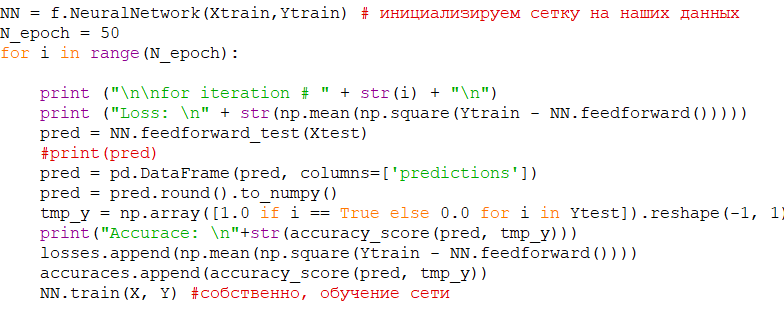


Рис. 6 — Обучение нейронной сети

Для получения предсказания необходимо вызвать метод feedforward. Заметим, что он выдает не номер класса, а значение от 0 до 1. (Рис. 7)

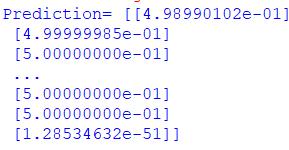


Рис. 7 — Получение предсказания

После обучения сети, вычисляя на каждой эпохе обучения значение точности и среднеквадратичной потери, построим графики их зависимости от номера эпохи. (Рис. 8). По графику можно заметить, что наилучшее значения точности и среднеквадратичной потери были достигнуты где-то на 27 эпохе.

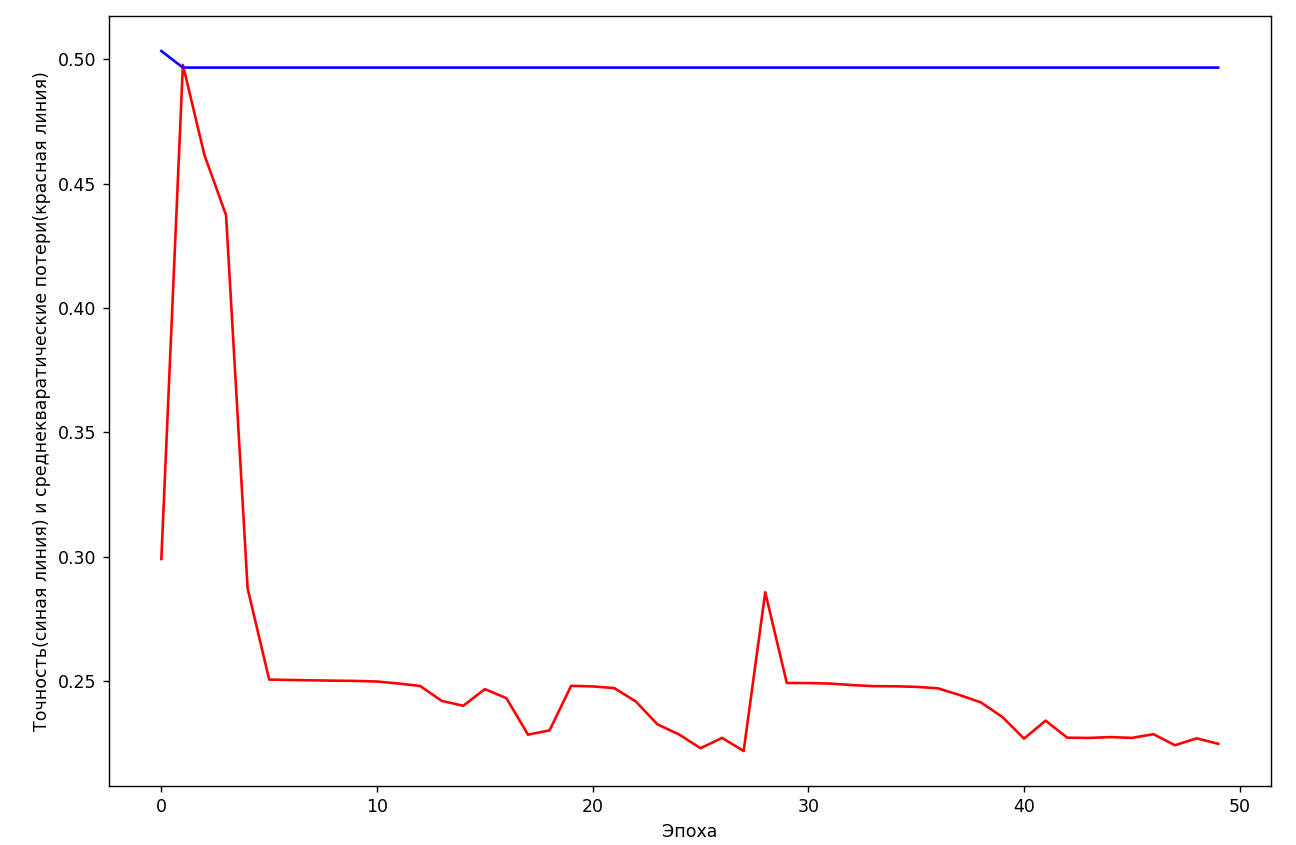


Рис. 8 — Зависимость точности и среднеквадратичной потери от эпохи

Также вынесем в отдельную переменную значение весов нейронов на каждом слое. (Рис. 9)



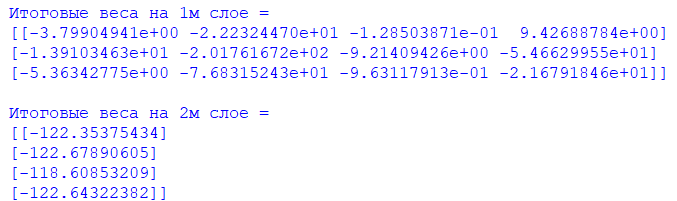


Рис. 9 — Веса нейронов на каждом слое

**Выводы**

В ходе данной работе были получены навыки создания простейшей нейронной сети на Python без использования специальных библиотек. Были исследованы устройство простейшей нейронной сети, и созданы функции для ее работы. Также был создан класс, собственно, самой нейронной сети, в ходе чего были исследованы особенности написания классов и методов класса на Python. Были построены и исследованы графики точности и среднеквадратичной потери. В итоге были закреплены все используемые в предыдущих лабораторных работах знания: вычисление, построение графиков и таблиц, работа с данными и выборками, а также построение классификаторов и анализ их работы.