Дисциплина: Современные нейросетевые технологии

Лабораторная работа №1

**Реализация метода обратного распространения ошибки для двухслойной полностью связанной нейронной сети**

***Цель:*** настоящей работы состоит в том, чтобы изучить метод обратного распространения ошибки для обучения глубоких нейронных сетей на примере двухслойной полностью связанной сети (один скрытый слой).

***Задачи:***

Выполнение практической работы предполагает решение ***следующих задач***:

1. Изучение общей схемы метода обратного распространения ошибки.

2. Вывод математических формул для вычисления градиентов функции ошибки по параметрам нейронной сети и формул коррекции весов.

3. Проектирование и разработка программной реализации.

4. Тестирование разработанной программной реализации.

5. Подготовка отчета, содержащего минимальный объем информации по каждому этапу выполнения работы.

В процессе выполнения лабораторной работы предполагается, что сеть ориентирована на решение задачи классификации одноканальных изображений. Типичным примером такой задачи является задача классификации рукописных цифр. Именно ее предлагается использовать в качестве тестовой задачи на примере набора данных MNIST [1].

Метод обратного распространения ошибки разрабатывается, исходя из следующих предположений:

1. На входе сети имеется 𝑤×ℎ нейронов, что соответствует разрешению изображения.

2. На выходе сети имеется 𝑘 нейронов, что соответствует количеству классов изображений.

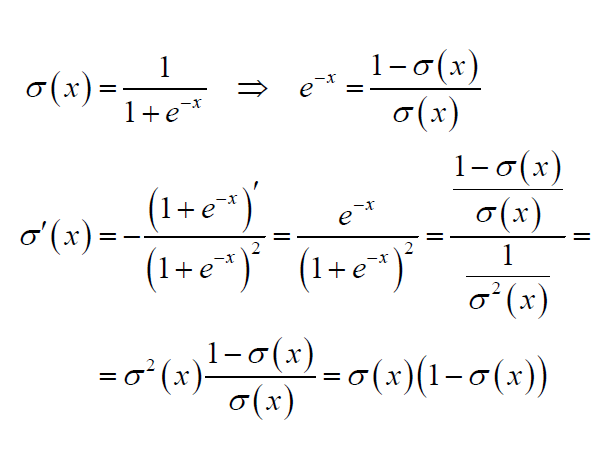
3. Скрытый слой содержит 𝑠 нейронов.

4. В качестве функции активации на втором слое используется функция softmax

.

**Выполнение работы**

1. Были изучены схема и математические формулы описывающие метод обратного распространения ошибки, функции активации, потерь и точности
2. Для реализации в качестве функции активации была выбрана сигмоида:



В качестве функции потерь была выбрана кросс энтропия



Где y – настоящие значения, a – предсказанные

Матричные формулы прямого хода слоя:

Z = Weights \* Input + Bias

Output = Sigmoid(Z)

Primes = SigmoidPrime(Z)

Матричные формулы получения обратного хода сети

DZ = ExternalFunctionPrime \* Primes

DW = DZ \* InputT

DB =

Error = WeightsT\*DZ

Где Weight – веса, Input – входные значения, Bias – смещение

ExternalFunctionPrime – производная внешней функции (Для внешнего слоя это производная кроссжнтропии далее значение Error от более внешнего слоя – по сути производная функции которая дает входные параметры этого слоя)

m – первая размерность матрицы DZ

1. Была разработана программная реализация на языке python использованием библиотек numpy, pytorch, matplotlib и оформлен в виде блокнота ipynb, представленного в репозитории
2. Програмная реализация была протестирована, получены следующие графики функции точности и потерь

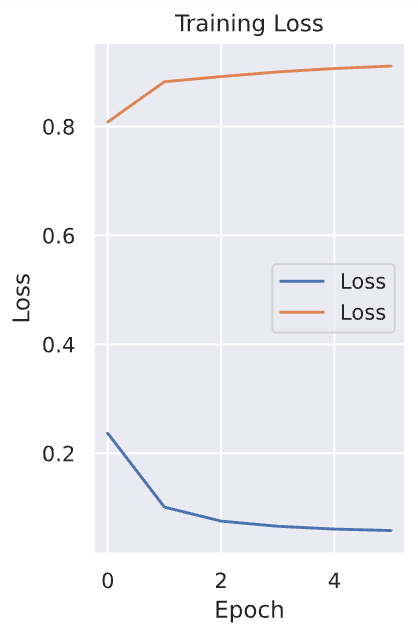


Рисунок 1 – графики функций потерь и точности (Синий потери, оранжевый – точности)

Итоговая точность составила 0.9109

1. По проведенной работе был составлен отчет