

Постановка задачи.

Вариант 7

Логическая операция:

(a **or** b) and (a **xor not** b)

Необходимо реализовать нейронную сеть, вычисляющую результат заданной логической операции. Затем реализовать функции, которые будут симулировать работу построенной модели. Функции должны принимать тензор входных данных и список весов. Должно быть реализовано 2 функции:

1. Функция, в которой все операции реализованы как поэлементные операции над тензорами
2. Функция, в которой все операции реализованы с использованием операций над тензорами из NumPy

Для проверки корректности работы функций необходимо:

1. Инициализировать модель и получить из нее веса (Как получить веса слоя, Как получить список слоев модели)
2. Прогнать датасет через не обученную модель и реализованные 2 функции. Сравнить результат.
3. Обучить модель и получить веса после обучения
4. Прогнать датасет через обученную модель и реализованные 2 функции. Сравнить результат.

Реализация

Программа считывает матрицу входных значений из файла, затем создает модель из трех слоев, два первых слоя имеют по 16 нейронов и активацию relu, а последний слой 1 нейрон и активацию sigmoid. Так же было реализовано две функции: `native_simulation(data, m_weights)` – которая симулирует работу модели с помощью поэлементных операций над тензорами и `np_simulation(data, m_weights)` – которая симулирует работу модели

используя функции `numpy` и функция. Затем датасет был прогнан через необученную сеть и две функции полученные результаты:

Модель до обучения:

```
[[0.5      ]  
 [0.5479686 ]  
 [0.5072564 ]  
 [0.56113875]  
 [0.48240337]  
 [0.48234034]  
 [0.4644779 ]  
 [0.4898047 ]]
```

Функция с поэлементными операциями:

```
[[0.5      ]  
 [0.54796861]  
 [0.50725641]  
 [0.56113877]  
 [0.48240337]  
 [0.48234034]  
 [0.4644779 ]  
 [0.48980469]]
```

Функция использующая `numpy`:

```
[[0.5      ]  
 [0.54796861]  
 [0.50725641]  
 [0.56113877]  
 [0.48240337]  
 [0.48234034]  
 [0.4644779 ]  
 [0.48980469]]
```

Из полученных данных можно сделать вывод, что результаты двух функций идентичны и совпадают с данным модели в пределах погрешности.

Затем модель была обучена через 300 эпох и получен следующий результат:

Модель после обучения:

[[0.07636824]

[0.03091606]

[0.05005556]

[0.95297897]

[0.8954799]

[0.04442203]

[0.07516047]

[0.9567746]]

Функция с поэлементными операциями:

[[0.5]

[0.54796861]

[0.50725641]

[0.56113877]

[0.48240337]

[0.48234034]

[0.4644779]

[0.48980469]]]

Функция использующая numpy:

[[0.5]

[0.54796861]

[0.50725641]

[0.56113877]

[0.48240337]

[0.48234034]

[0.4644779]

[0.48980469]]

После обучения значения, полученные моделью и функциями совпадают в пределах погрешности