

Бердникова Анастасия, группа 8382

Вариант 1

Задание.

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
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.

Примечание: так как множество всех наблюдений ограничен, то обучение проводить можно на всем датасете без контроля.

(a and b) or (a and c)

Загрузка данных и создание ИНС.

Модель:

```
model = Sequential()
model.add(Dense(16, activation='relu',
input_shape=(3,)))
model.add(Dense(8,
activation='relu'))
model.add(Dense(1,
activation='sigmoid'))
```

```
model.compile(optimizer='adam',  
loss='binary_crossentropy', metrics=['accuracy'])    return  
model
```

Для симуляции работы нейронной сети на основе весов были реализованы две функции:

- `element_operations` - операции над тензорами реализованы без использования `numpy`
- `numpy_operations` - операции над тензорами реализованы с помощью `numpy`

результаты до обучения поэлементные

операции над тензорами:

[[0.5]

[0.53591688]

[0.52009833]

[0.51721061]

[0.51408638]

[0.50071916]

[0.49505] [0.47822068]]

операции над тензорами из

numpy:

[[0.5]

[0.53591688]

[0.52009833]

[0.51721061]

[0.51408638]

[0.50071916]

[0.49505]

[0.47822068]]

предсказание модели:

[[0.5]

[0.53591686]

[0.5200983]

[0.5172106]

[0.51408637]

[0.5007192]

[0.49504998] [0.47822067]] результаты

после обучения поэлементные

операции над тензорами: [[0.1494457]

[0.05686541]

[0.07749248]

[0.03758528]

[0.44947513]

[0.85107282]

[0.89333006] [0.94861534]]

операции над тензорами из

numpy:

[[0.1494457]

[0.05686541]

[0.07749248]

[0.03758528]

[0.44947513]

[0.85107282]

[0.89333006]

[0.94861534]]

предсказание модели:

[[0.14944568]

[0.05686539]

[0.07749248]

[0.03758535]

[0.44947514]

[0.8510728]

[0.8933301]

[0.9486153]]

Как до обучения, так и после обучения значения, полученные симуляциями и реальные совпадают. Значения, полученные симуляциями с помощью numpy и без него полностью совпадают.