

1) Какие недостатки у функции relu?

Существует проблема умирающего ReLu, которая ограничивает пропускную способность сети.

Если мы посмотрим на график функции ReLu, мы заметим что часть этого графика (при $x < 0$) это прямая по ОХ, и градиент в этой части равен нулю. Так как градиент равен 0 \Rightarrow веса не будут меняться во время градиентного спуска \Rightarrow нейроны не будут реагировать на изменения в ошибке входных данных, т е ничего не будет меняться. Получается, что из-за этой проблемы некоторые нейроны просто выключатся и не будут отвечать, делая значительную часть нейросети пассивной.

2) В виде какого тензора можно представить датасет из двумерных сигналов?

Датасет из двумерных сигналов можно представить в виде трехмерного тензора, так как если двумерные сигналы можно представить, как тензор с формой (x, y) , то датасет из них можно представить как тензор с формой (n, x, y) , т е трехмерный тензор.

3) Что такое L1-Регуляризация весов?

L1-регуляризация является методом контроля емкости нейронной сети, который позволяет предотвратить переобучение.

В результате L1-регуляризации векторы весов становятся разреженными. Значит нейроны с L1-регуляризацией используют подмножество наиболее важных входов, а значит не подвергаются влиянию шумов - признаков, которые не влияют на конечный результат.

L1-регуляризацию еще называют лассо. При L1-регуляризации для каждого веса ω мы прибавляем к целевой функции сумму весов в модуле $\lambda|\omega|$. То есть, если весы на расстоянии λ ближе к 0, то они становятся равными 0 (так как обнуляются коэффициенты), а значит игнорируются, таким образом отбираются только важные признаки \Rightarrow уменьшается сеть.

4) Для чего нужна test data в Вашей программе?

В test_data мы загружаем данные для тестирования из boston_housing. Далее в программе она не используется, так как у нас и без того небольшой набор данных, проверочный набор получился бы слишком маленьким. Как следствие, оценки при проверке могли бы иметь слишком большой разброс. Это не позволило бы надежно оценить качество модели.

5) В каком месте задаются метрики для расчета во время обучения?

15 строка

```
model.compile(optimizer='rmsprop', loss='mse', metrics=['mae'])
```

В данном месте задаются метрики:

```
metrics=['mae']
```

6) В каких строчках у Вас происходит формирование обучающей и контрольной выборки?

47-54 строки

```
47 +     val_data = train_data[i * num_val_samples: (i + 1) * num_val_samples]
48 +     val_targets = train_targets[i * num_val_samples: (i + 1) * num_val_samples]
49 +     partial_train_data = np.concatenate([train_data[:i * num_val_samples],
50 +                                         train_data[(i + 1) * num_val_samples:]],
51 +                                         axis=0)
52 +     partial_train_targets = np.concatenate([train_targets[:i * num_val_samples],
53 +                                             train_targets[(i + 1) * num_val_samples:]],
54 +                                             axis=0)
```