

1. Постановка задачи

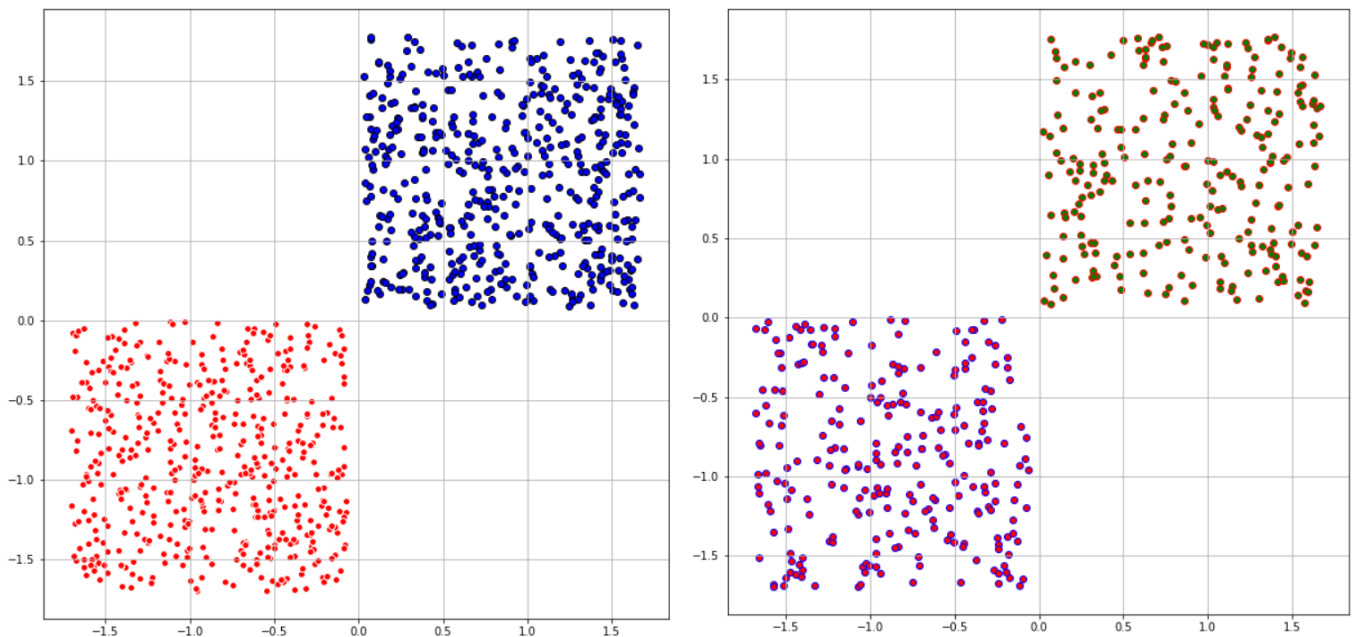
Обучить нейронную сеть с одним нейроном и последующей функцией активации разделять два непересекающихся множества.

Обучение выполнить с использованием смещения и без него.

Реализация на TensorFlow. Классификация координат на плоскости.

2. Отображение набора данных, обучающих и проверочных множеств на рисунках (после стандартизации)

обучающее множество | проверочное множество



3. Обучение НС

Функция активация – leaky_relu, эпох 8, размер всего обучающего пакета – 1000, batch_size = 250, оптимизатор Адам, функция потерь – MSE.

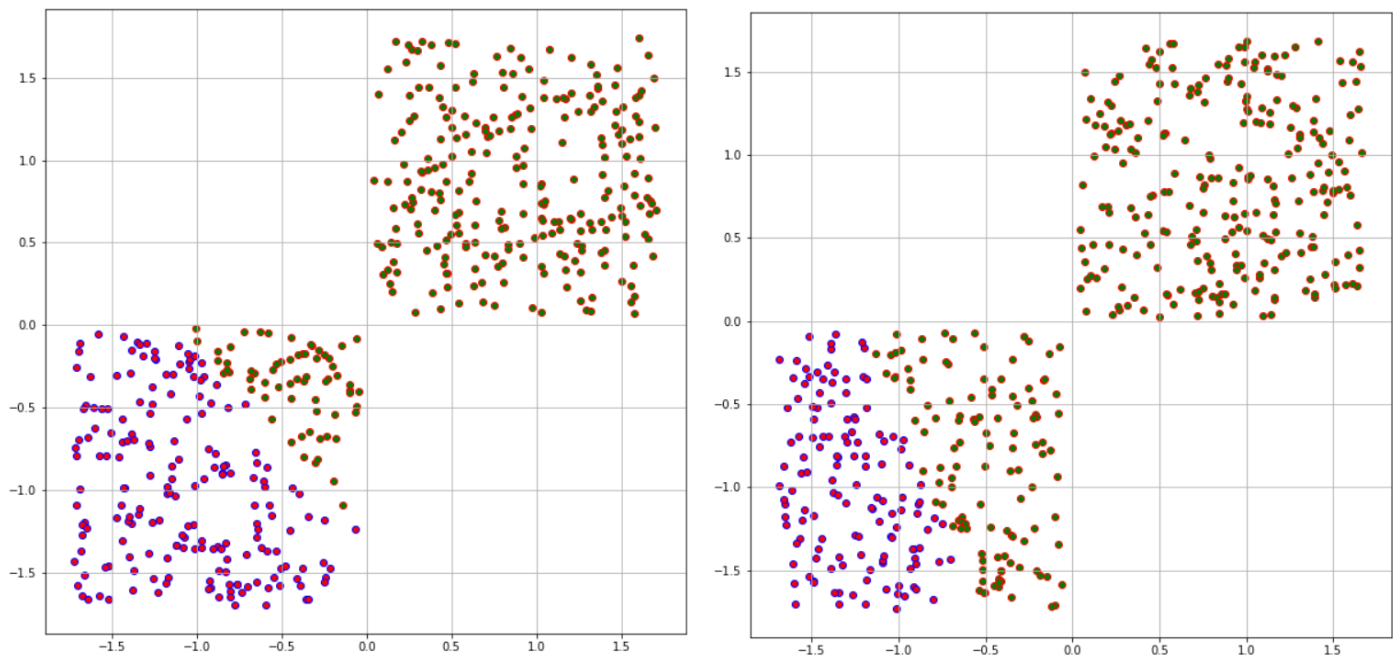
Вывожу потери и точность только последнего пакета.

с нейроном смещения | без нейрона смещения

loss 0.0147309	acc 0.984	loss 0.019605042	acc 0.976
loss 0.01871446	acc 0.976	loss 0.024280433	acc 0.968
loss 0.012510688	acc 0.98	loss 0.010704865	acc 0.988
loss 0.01756749	acc 0.964	loss 0.01793336	acc 0.972
loss 0.013864515	acc 0.972	loss 0.024253624	acc 0.968
loss 0.004180588	acc 0.996	loss 0.011793653	acc 0.996
loss 0.0083858855	acc 0.988	loss 0.011432586	acc 0.996
loss 0.0040376317	acc 1.0	loss 0.023183057	acc 0.976

4. Прогноз модели на проверочном множестве

с нейроном смещения (точность 87%) | без нейрона смещения (точность 76.8%)



5. Вывод весов обученного нейрона и смещения (при наличии последнего)

Со смещением:

```
: print(model.trainable_variables)

(<tf.Variable 'Variable:0' shape=(1,) dtype=float32, numpy=array([0.23971723], dtype=float32)>, <tf.Variable 'Variable:0' shape
=(2, 1) dtype=float32, numpy=
array([[ -0.23310749],
       [-0.20735678]], dtype=float32)>)
```

Без смещения:

```
# вывод весов обученного нейрона и смещения (при наличии)
print(model.trainable_variables)

(<tf.Variable 'Variable:0' shape=(2, 1) dtype=float32, numpy=
array([[ -0.4073395 ],
       [-0.16363926]], dtype=float32)>,,)
```