

Замечательно оформлены результаты, я давно прошу выполняющих этот проект сводить в таблицу результаты экспериментов, но почти никто так не делает.

```
print('Распаковываем картинки')
# Will unzip the files so that you can see them..
for data_zip in ['train.zip', 'test.zip']:
    with zipfile.ZipFile("../input/"+data_zip, "r") as z:
        z.extractall(PATH)
```

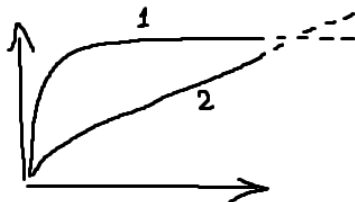
Можно сделать то же самое командой ОС:

```
!unzip -q ../input/train.zip -d {PATH}
```

В ReduceLROnPlateau указаны параметры: factor=0.9, patience=20. Во-первых если LR будет умножаться на 0.9 не чаще, чем раз в 20 эпох, то скорость его снижения будет очень низкой, вы же не будете обучать модель по 500 эпох, так что это можно считать эквивалентным константному LR. И кроме того у вас в EarlyStopping параметр patience=15 – это значит, что ReduceLROnPlateau никогда не сработает, потому что в нем patience=20. Хотя с другой стороны вы пишете, что меняли параметр patience в ReduceLROnPlateau.

Из кода не до конца понятно, точно ли вы использовали правильную нормализацию (то есть такую же нормализацию, какая использовалась при преобучении сети). Например, для Xception нужно использовать функцию `tf.keras.applications.xception.preprocess_input`, а для FixEfficientNet – смотря из какой библиотеки она взята.

При выборе между моделями могли бы пригодиться графики обучения, а не просто цифры. Может быть было бы видно, что одна модель обучается медленнее, но имеет больший потенциал, чем другая модель.



Еще можно при этом отрисовывать на графике используемый learning rate и его затухание.

> кажется единственное что сильно помогло увеличить качество модели это увеличение изображения.

Многие современные сверточные сети, увы, чувствительны к размеру изображения.

Отзыв подготовил ментор проекта Олег Зяблов. Если есть вопросы, можете задать их в канале #0-project_7-ford_vs_ferrari, постараюсь помочь разобраться. Успехов в дальнейшем обучении! Обязательно подключайтесь на итоговый созвон-вебинар по проекту 18 декабря. Анонс вебинара появится позже в канале #0-project_7-ford_vs_ferrari.