|  |  |
| --- | --- |
| **СОГЛАСОВАНО**  Доцент кафедры  ИАНИ ННГУ, к.ф.-м.н.  \_\_\_\_\_\_\_\_\_\_\_\_\_ Д.А. Яшунин  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_2019 г. | **УТВЕРЖДАЮ**  Профессор кафедры  ИАНИ ННГУ, д.т.н.  \_\_\_\_\_\_\_\_\_\_\_\_\_ Н.В. Старостин  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_2019 г. |

**Пояснительная записка №7**

**«Метрики для задачи распознавания лиц»**

**Этап 1. Подготовка обзоров на существующие подходы к решению   
задачи и построение технологического стека**

**НИР «Разработка и реализация программного обеспечения   
для распознавания лиц на групповых фотографиях»**

**(Шифр ПО «AFR»)**

Ответственный исполнитель

\_\_\_\_\_\_\_\_\_\_\_\_ М.М. Годовицын

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_2019 г.

**Н. Новгород 2019**

**Оглавление**

[1 ROC (Receiver Operating Characteristic) 2](#_Toc4446106)

[2 mAP(mean Average Precision) 3](#_Toc4446107)

[3 Top-1 5](#_Toc4446108)

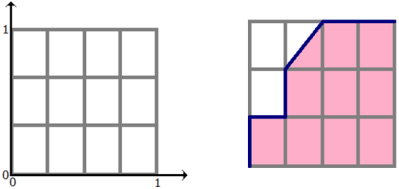
[4 Список источников 5](#_Toc4446109)

# ROC (Receiver Operating Characteristic)

Рассмотрим случай с бинарной классификацией, мы имеем два класса и 7 объектов.

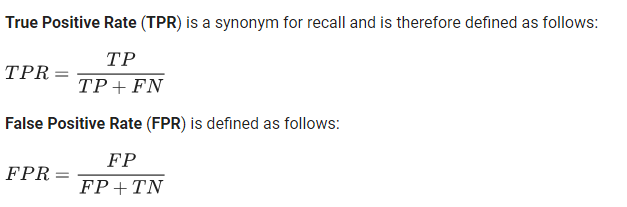
В таблице показаны метки и предсказанная вероятность принадлежности в отрезке [0;1].  
Далее мы сортируем по убыванию предсказанной вероятности, в случае идеальной классификации третий столбец должен быть автоматически упорядочен.

Далее строится квадрат 1 на 1.



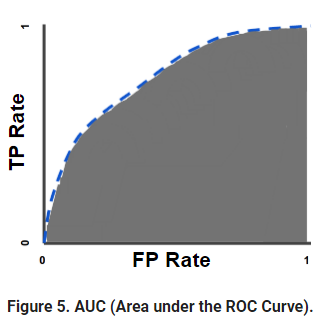
Начиная с (0,0) и в списке сверху вниз, если класс = 1, то делаем шаг вверх, иначе, вправо.  
Если подряд идут объекты с одинаковой оценкой, то мы делаем шаг вверх на количество классов 1 и шаг вправо на количество классов 0.

По сути данный график показывает зависимость TPR от FPR.



TPR показывает величину правильно предсказанных 1ек среди всех объектов с меткой класса 1.

FPR показывает величину неверно предсказанных 0ек среди всех объектов с меткой класса 0.



AUC ROC будет площадь под полученным графиком, в данном случае AUC ROC = 9.5 / 12 ~ 0.79.

Для n-классовых задач обычно используют n кривых с конфигурациями  
1. Class 1 vs Classes 2...n  
2. Class 2 vs Classes 1, 3...n  
и т.д.

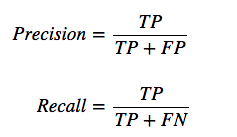
Так мы получим метрику для каждого класса при предсказании выборки объектов.

# mAP(mean Average Precision)

mAp (mean Average Precision) или просто AP (Average Precision). Основывается на двух более простых метриках, как Recall и Precision. Показывает среднюю точность из упорядоченных по recall данных.

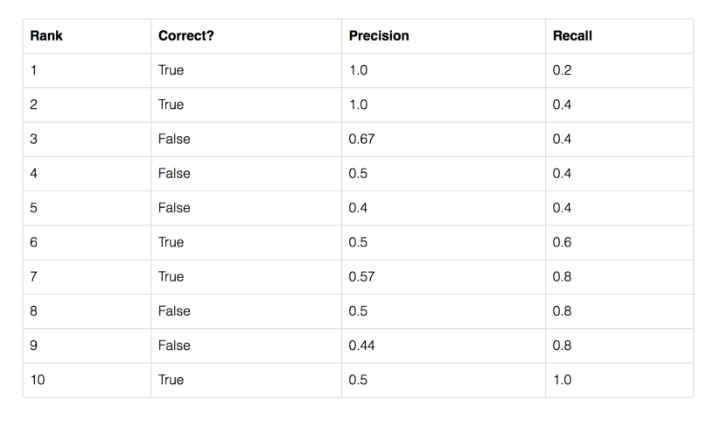
Precision показывает насколько точны ваши предсказания (доля верных позитивных прогнозов среди всех положительно предсказанных).

Recall показывает насколько хорошо вы нашли положительные прогнозы (доля верных позитивных прогнозов среди всех возможных верных прогнозов).



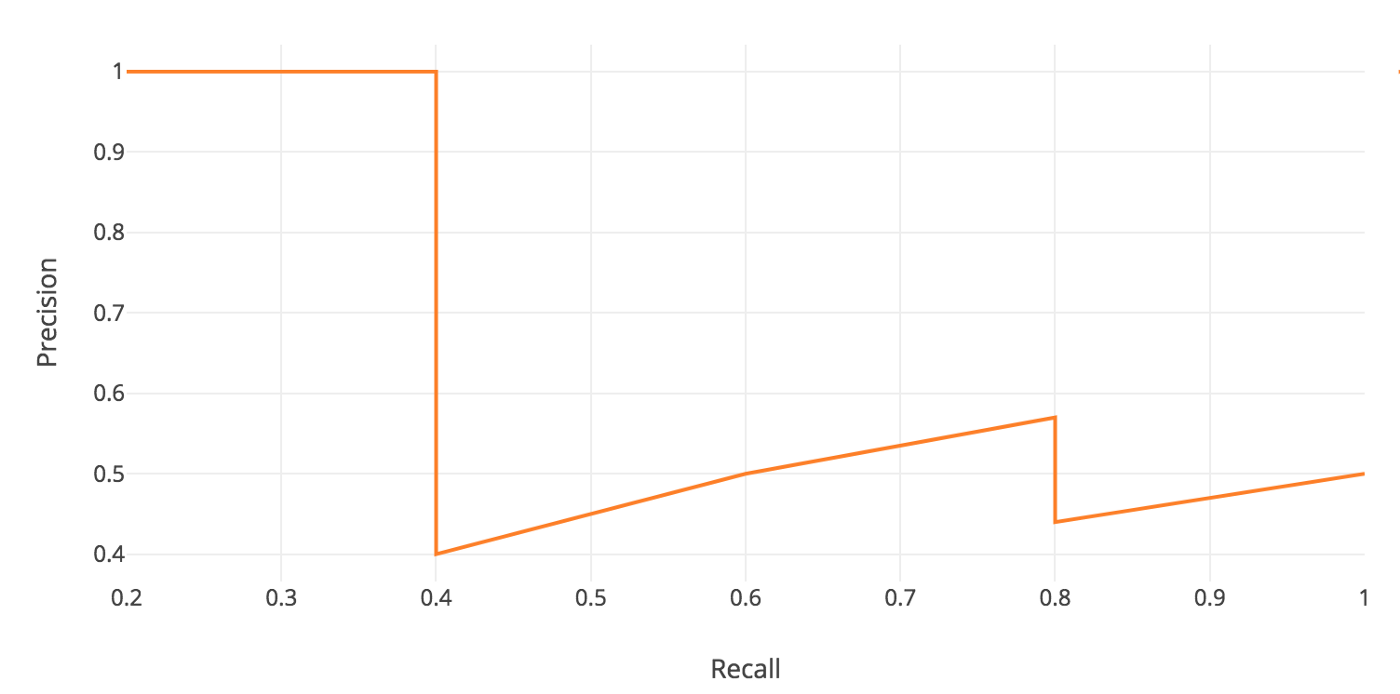
Для предварительной фильтрации используется IoU(Intersection over union), показывает насколько хорошо пересекаются предсказанная и корректная область.

Пусть мы имеем некоторый набор предсказаний отсортированный по степени уверенности.



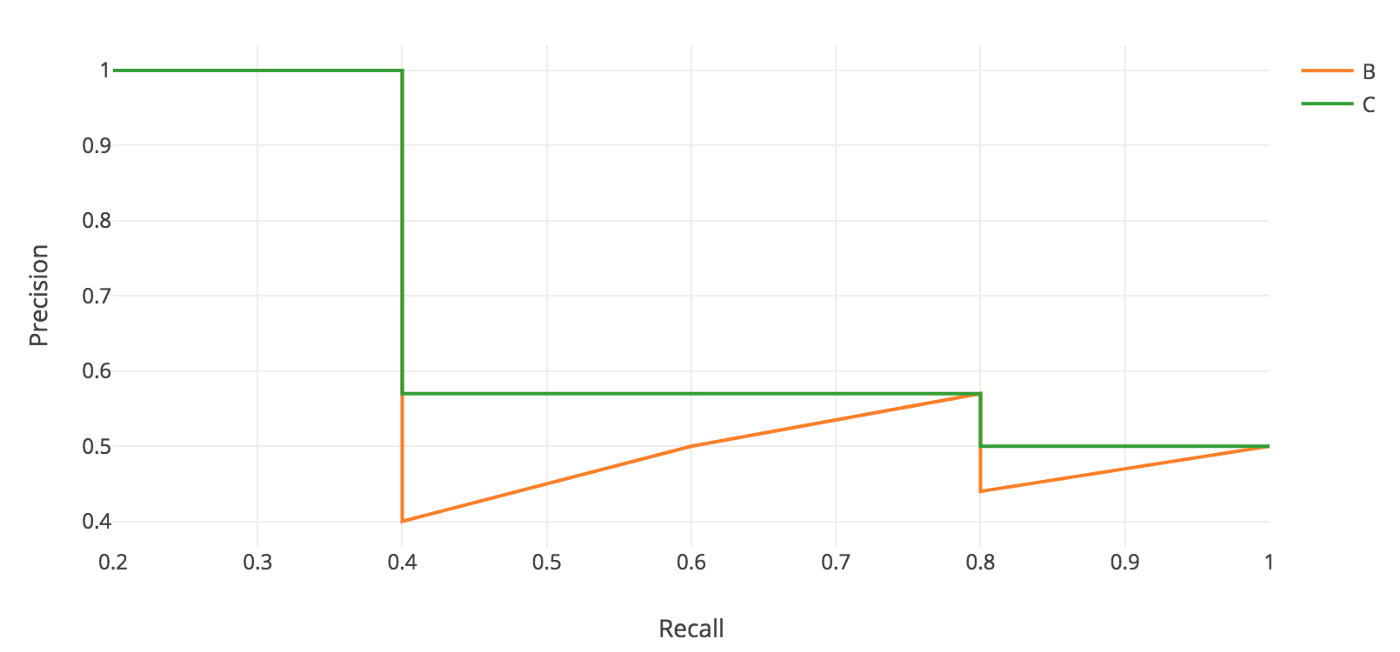
Precision для каждой строки вычисляется как доля корректных по мнению IoU предсказаний по top k предсказаний (где k - текущий ранг строки).

Recall вычисляется как доля корректных по всем корректным предсказаниям.



Далее мы строим сглаживание графика: для каждого recall 0.0, 0.1 ... 1.0  
выбираем максимальный precision в отрезке r >= текущий recall.

C:\Users\TrueSkit\Desktop\mAP4.PNG



AP вычисляется как среднее по всем максимальным precision для каждого recall 0.0, 0. 1 ... 1.0. C:\Users\TrueSkit\Desktop\mAP6.PNG

Для нашего случая.

mAP это просто среднее по всем классам.

# Top-1

Пусть есть набор изображений лиц, котором соответствует вектор предсказания сортированный по степени уверенности принадлежности к конкретному классу (пусть классов будет 5).

Итерируясь по каждому вектору предсказания определяем ошибку, если корректный класс находится в первом элементе массива, тогда распознавание корректно, иначе начисляется ошибка.   
Пройдя все лица top1= sum of error / number of faces.

Можно использовать и более простые модели, такие как recall, precision, F1(F-measure), accuarcy.

# Список источников

1) https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc

2) https://hackernoon.com/simple-guide-on-how-to-generate-roc-plot-for-keras-classifier-2ecc6c73115a

3) https://medium.com/@jonathan\_hui/map-mean-average-precision-for-object-detection-45c121a31173

4) https://stackoverflow.com/questions/37668902/evaluation-calculate-top-n-accuracy-top-1-and-top-5