

# PR-curve, ROC-curve, cost-curve

Полина Святокум

19 декабря 2016 г.

## 1 Что нельзя понять по ROC-кривым

- ROC-кривая и cost-кривая
- Визуальная оценка кривых
- Другие преимущества cost-кривых

## 2 Сходства и отличия PR-кривых и ROC-кривых

- PR-кривая и когда она лучше ROC-кривой
- Выпуклая оболочка и достижимая PR-кривая
- Построение достижимой PR-кривой
- AUC-ROC и AUC-PR

Будем рассматривать задачу бинарной классификации. Для оценки качества алгоритмов часто используют ROC-кривую (в частности, площадь под кривой).

ROC-пространство это плоскость с  $x$ -координатой отвечающей за  $FP$  rate и  $y$ -координатой отвечающей за  $TP$  rate.

Одна матрица ошибок порождает одну точку в этом пространстве. ROC-кривая получается при соединении ROC-точек алгоритма и  $(0, 0)$ ,  $(1, 1)$

## PCF(+) и cost-кривая

Мы хотим уметь оценивать качество алгоритма в зависимости от штрафов за неправильную классификацию. С этой целью вводим следующую функцию

$$PCF(+) = \frac{p(+)C(-|+)}{p(+)C(-|+) + p(-)C(+|-)}$$

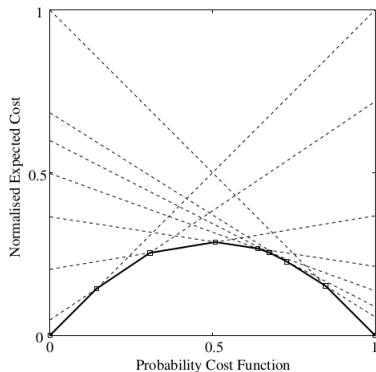
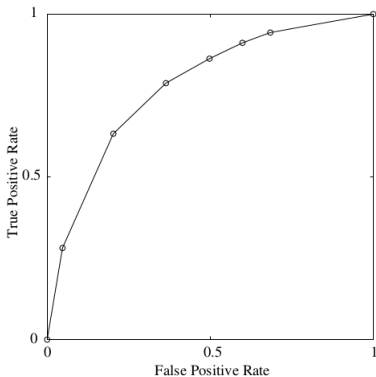
Cost-пространством является плоскость с x-координатой отвечающей за PCF(+) и y-координатой отвечающей за нормализованную ожидаемую cost error алгоритма.

При  $C(-|+) = C(+|-)$  по оси x будет отложена вероятность появления положительного класса, по оси y будет отложена относительная ошибка классификации.

# Связь ROC-кривых и cost-кривых

Каждая точка ROC-пространства соответствует прямой в cost-пространстве, и каждой прямой ROC-пространства соответствует точка в cost-пространстве.

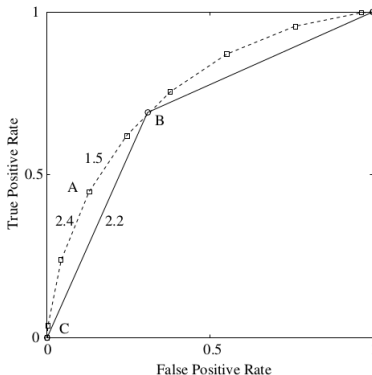
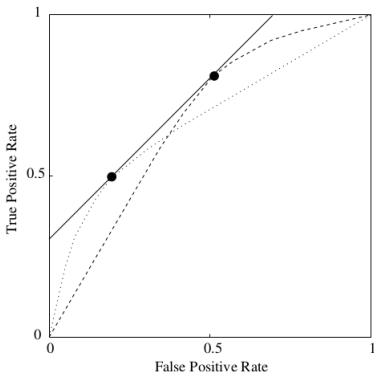
Точке  $(FP, TP)$  из ROC-пространства соответствует прямая, проходящая через точки  $(0, FP)$ ,  $(1, 1 - TP)$ , из cost-пространства.



- ROC-кривую сложно связать с конкретным способом оценки качества классификации
- По ROC-кривой сложно сравнить алгоритм с константным предсказанием

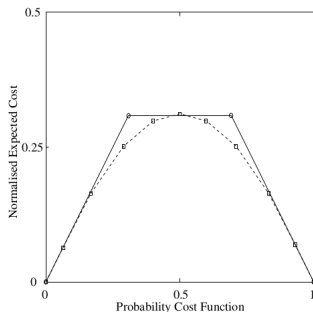
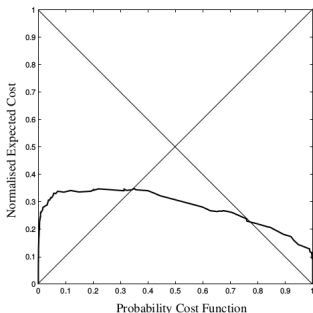
# Минусы ROC-кривой

- По ROC-кривым сложно понять при каких начальных условиях один алгоритм превосходит другой.
- По ROC-кривым сложно понять насколько один алгоритм превосходит другой (при разных параметрах нужно сравнивать разные точки)



# Как это сделать с помощью cost-кривых

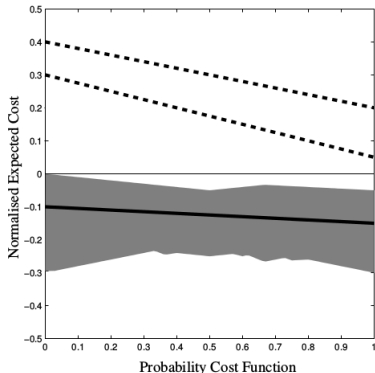
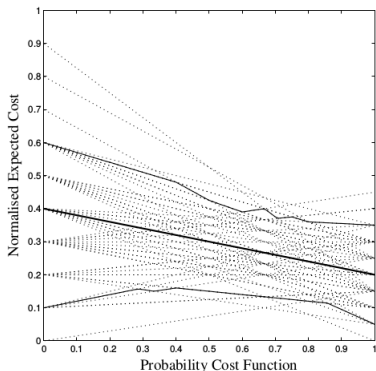
- Оценить качество алгоритма при различных начальных параметрах можно посмотрев на точку с соответствующей  $x$ -координатой
- Константные предсказания в cost-пространстве описываются диагоналями единичного квадрата
- В cost-пространстве можно оценивать норму разности  $y$ -координат при каждой фиксированной  $x$ -координате





# Другие преимущества cost-кривых

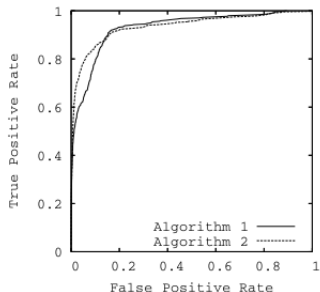
- Cost-кривые можно адекватно усреднить
- С помощью cost-кривых и бустрапа можно строить доверительные интервалы
- С помощью cost-кривых можно проверять гипотезу о существенном различии работы двух алгоритмов



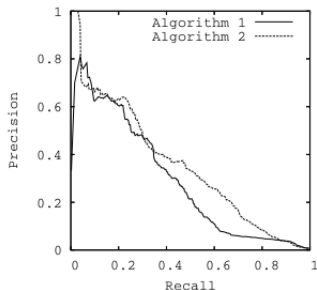
# PR-кривая

PR-пространство это плоскость с x-координатой отвечающей за Recall(TP rate) и y-координатой отвечающей за Precision.

Обычно сравнение сильно смещенных данных лучше проводить именно с помощью PR-кривой.



(a) Comparison in ROC space

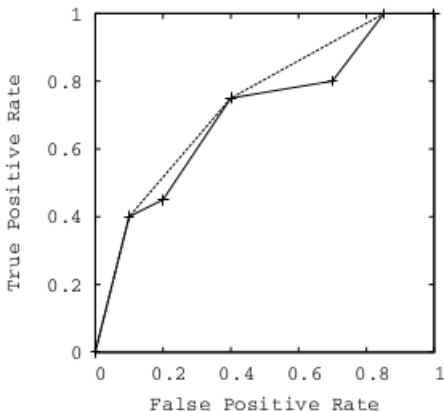


(b) Comparison in PR space

- Если Recall не равен 0, то по PR-точке можно однозначно восстановить матрицу ошибок. Следовательно, можно построить соответствующую ROC-точку. Обратное также верно. Потому между PR-кривыми и ROC-кривыми можно установить взаимно однозначное соответствие (при фиксированном наборе данных).
- На фиксированном наборе данных первая PR-кривая доминирует вторую тогда и только тогда, когда соответствующая первой ROC-кривая доминирует ROC-кривую соответствующую второй.

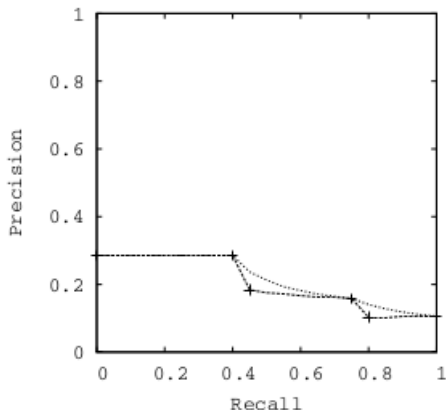
# Выпуклая оболочка ROC-кривой

- Между соседними точками используется линейная интерполяция
- Точки не лежат выше финальной кривой
- Для любых двух точек финальной кривой отрезок, который их соединяет, не лежит выше кривой



# Достижимая PR-кривая

- Достижимая PR-кривая доминирует все остальные валидные PR-кривые, которые могут быть построены на этих точках
- Может быть построена как двойственная к выпуклой оболочке ROC-кривой



# Интерполяция между соседними точками

- Линейная интерполяция между соседними точками PR-кривой нехороша по нескольким причинам. Во-первых, мы увеличиваем AUC-PR, во-вторых, мы получаем не двойственную к выпуклой оболочке ROC-кривой кривую.
- Для интерполяции применима следующая формула  
Если  $(FP_A, TP_A), (FP_B, TP_B)$  – точки из ROC-пространства, то в PR-пространстве мы их соединим кривой состоящей из точек

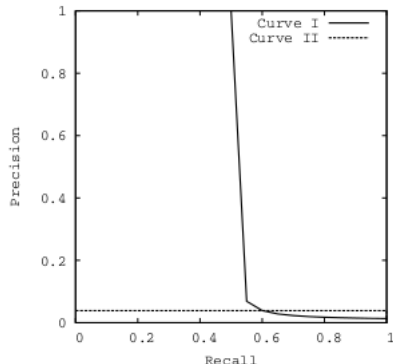
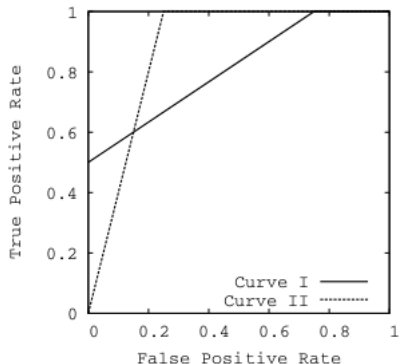
$$\left( \frac{TP_A + x}{TotalPos}, \frac{TP_A + x}{TP_A + x + FP_A + \frac{FP_B - FP_A}{TP_B - TP_A} x} \right), 1 \leq x \leq TP_B - TP_A$$

# Алгоритм построения достижимой PR-кривой

- Рассмотреть двойственные точки в пространстве ROC
- Построить их выпуклую оболочку
- Рассмотреть двойственные точки к выпуклой оболочке в пространстве PR
- Провести интерполяцию между соседними точками

# AUC-ROC и AUC-PR

- Алгоритмы, которые максимизируют AUC-ROC не всегда максимизируют AUC-PR
- Для сильно смещенных наборов данных классификаторы, максимизирующие AUC-ROC могут работать плохо





Ну как-то так