Модель линейной классификации

Классификация

- $\mathbb{Y} = \{-1, +1\}$
- -1 — отрицательный класс
- $+1$ — положительный класс
- Алгоритм $a(x)$ должен возвращать одно из двух чисел

Линейная регрессия

$$a(x) = \boxed{w_0 + \sum_{j=1}^{d} w_j x_j}$$

Вещественное число!

Линейный классификатор

$$a(x) = \operatorname{sign}\left(w_0 + \sum_{j=1}^{d} w_j x_j\right)$$

Линейный классификатор

$$a(x) = \operatorname{sign}\left(w_0 + \sum_{j=1}^{d} w_j x_j\right)$$

Свободный коэффициент

Веса

Признаки

Линейный классификатор

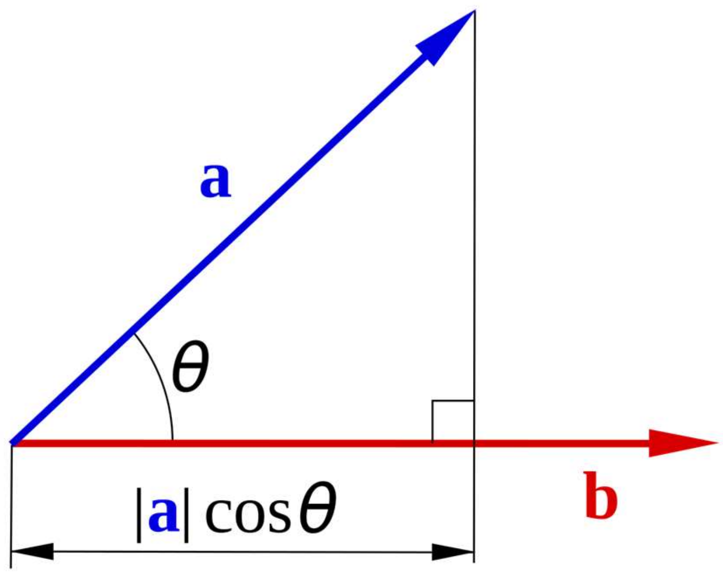
- Будем считать, что есть единичный признак

$$a(x) = \operatorname{sign} \sum_{j=1}^{d} w_j x_j = \operatorname{sign} \langle w, x \rangle$$

Геометрия линейного классификатора

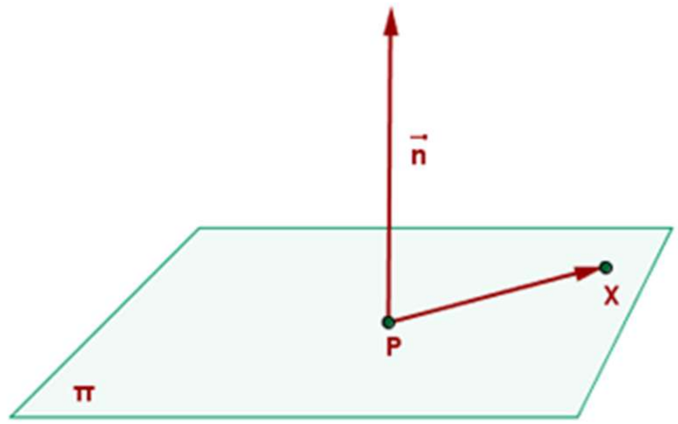
Скалярное произведение:

$$\langle a, b \rangle = |a||b|\cos(\Theta)$$



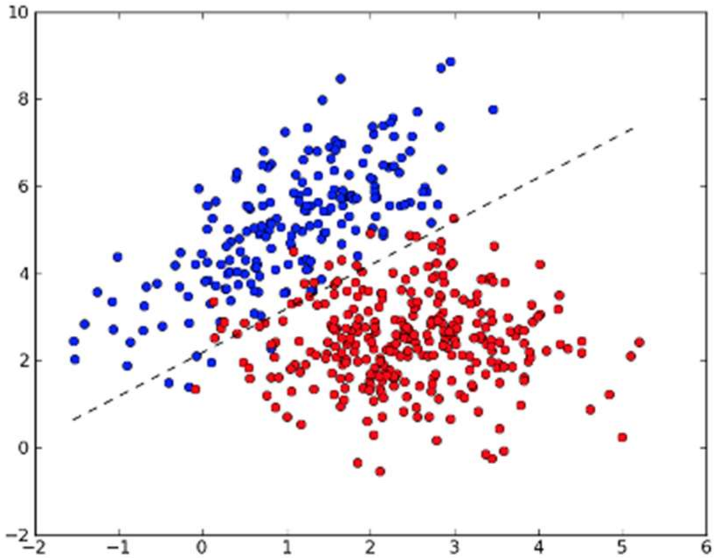
Геометрия линейного классификатора

Уравнение гиперплоскости: $\langle w, x \rangle = 0$



Геометрия линейного классификатора

- Линейный классификатор проводит гиперплоскость
- $\langle w, x \rangle < 0$ — объект «слева» от неё
- $\langle w, x \rangle > 0$ — объект «справа» от неё



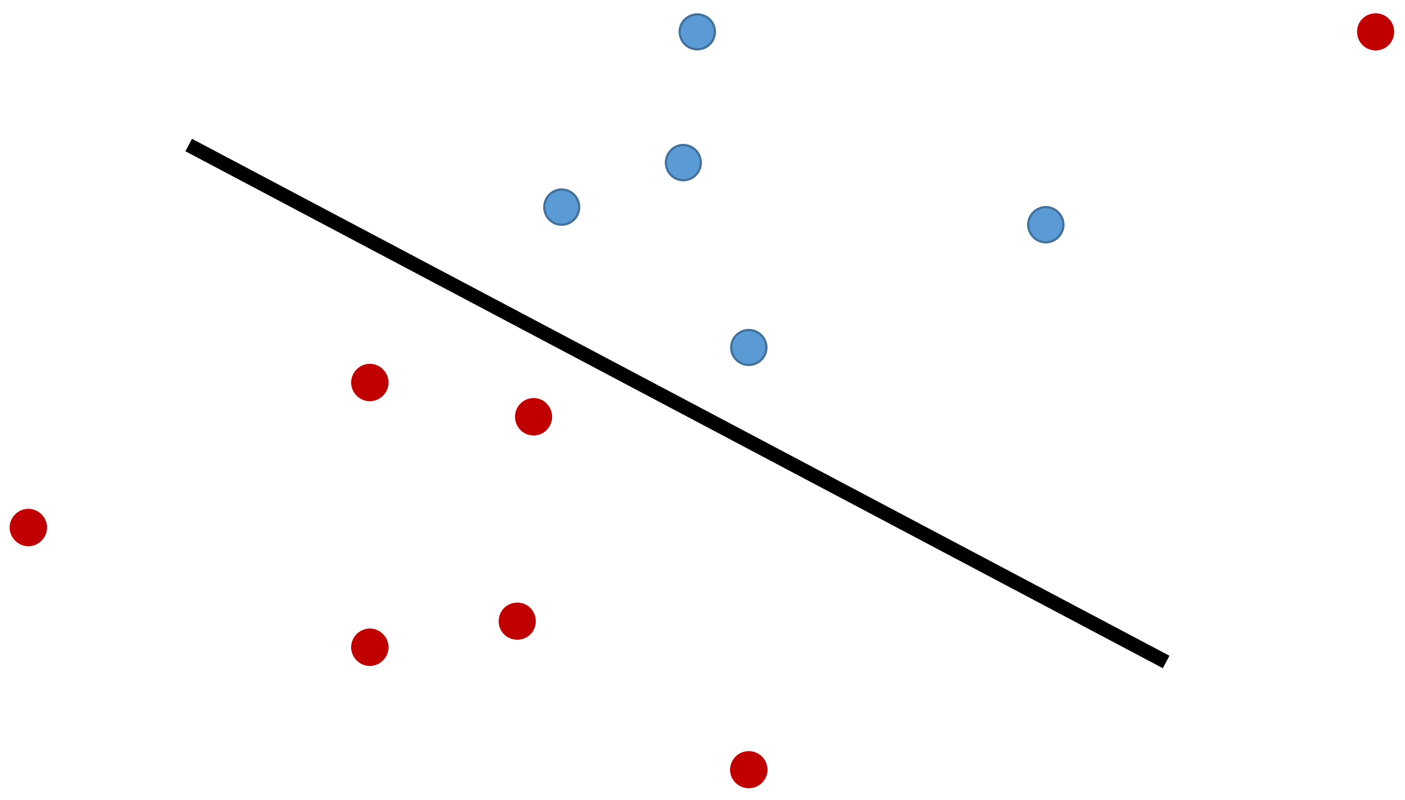
Геометрия линейного классификатора

- Расстояние от точки до гиперплоскости $\langle w, x \rangle = 0$:

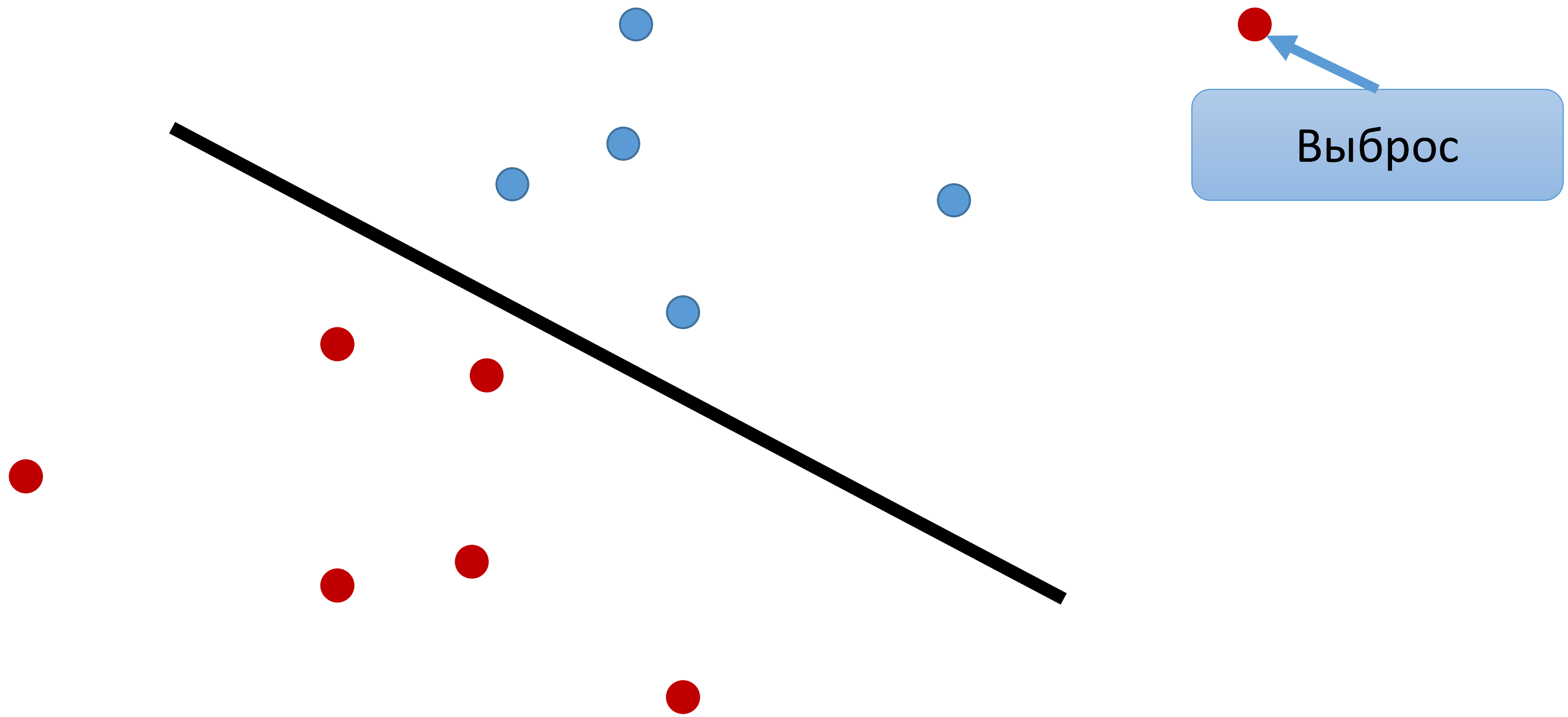
$$\frac{|\langle w, x \rangle|}{\|w\|} = \frac{|\langle w, x \rangle|}{\sqrt{\langle w, w \rangle}}$$

- Чем больше $\langle w, x \rangle$, тем дальше объект от разделяющей гиперплоскости

Геометрия линейного классификатора

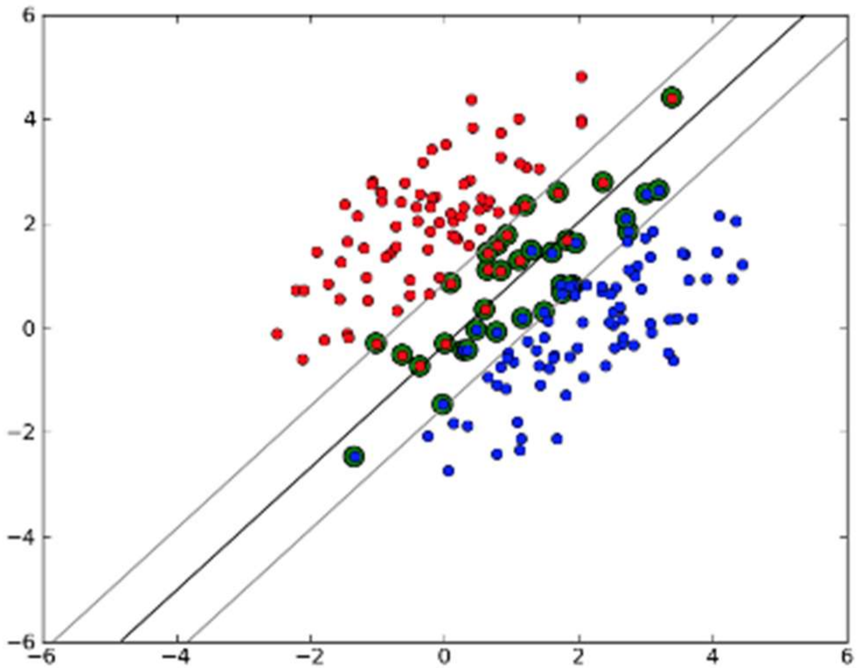


Геометрия линейного классификатора



Отступ

- $M_i = y_i \langle w, x_i \rangle$
- $M_i > 0$ — классификатор дает верный ответ
- $M_i < 0$ — классификатор ошибается
- Чем дальше отступ от нуля, тем больше уверенности



Порог

$$a(x) = \text{sign}(\langle w, x \rangle - t)$$

- t — порог классификатора

- Можно подбирать для оптимизации функции потерь, отличной от использованной при обучении

Линейный классификатор

- Линейный классификатор разделяет два класса гиперплоскостью
- Чем больше отступ по модулю, тем дальше объект от гиперплоскости
- Знак отступа говорит о корректности предсказания

Обучение линейных классификаторов

Функция потерь в классификации

- Частый выбор — бинарная функция потерь

$$L(y, a) = [a \neq y]$$

- Функционал ошибки — доля ошибок (error rate)

$$Q(a, X) = \frac{1}{\ell} \sum_{i=1}^{\ell} [a(x_i) \neq y_i]$$

- Нередко измеряют долю верных ответов (accuracy):

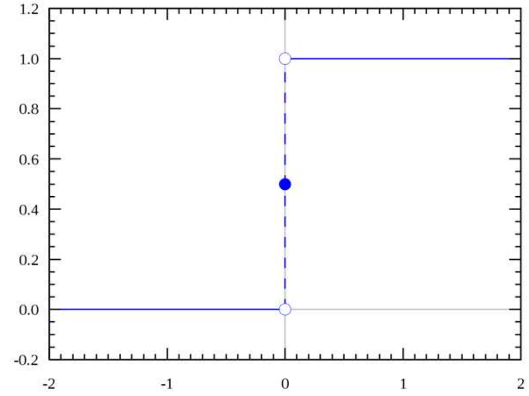
$$Q(a, X) = \frac{1}{\ell} \sum_{i=1}^{\ell} [a(x_i) = y_i]$$

Доля ошибок для линейного классификатора

- Функционал ошибки:

$$Q(w, X) = \frac{1}{\ell} \sum_{i=1}^{\ell} [\operatorname{sign}(\langle w, x_i \rangle) \neq y_i]$$

- Индикатор — недифференцируемая функция



Отступы для линейного классификатора

- Функционал ошибки:

$$Q(w, X) = \frac{1}{\ell} \sum_{i=1}^{\ell} [\operatorname{sign}(\langle w, x_i \rangle) \neq y_i]$$

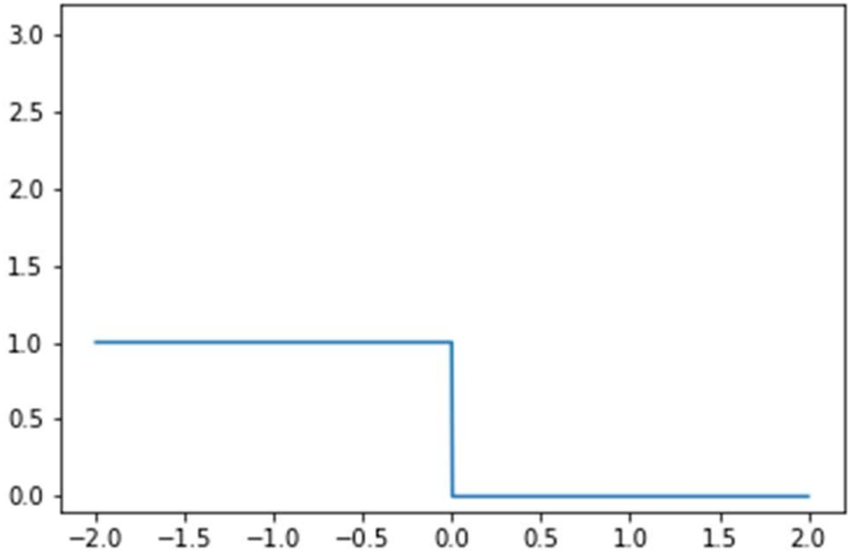
- Альтернативная запись:

$$Q(w, X) = \frac{1}{\ell} \sum_{i=1}^{\ell} [\underbrace{y_i \langle w, x_i \rangle}_{M_i} < 0]$$

Отступы для линейного классификатора

$$L(M) = [M < 0]$$

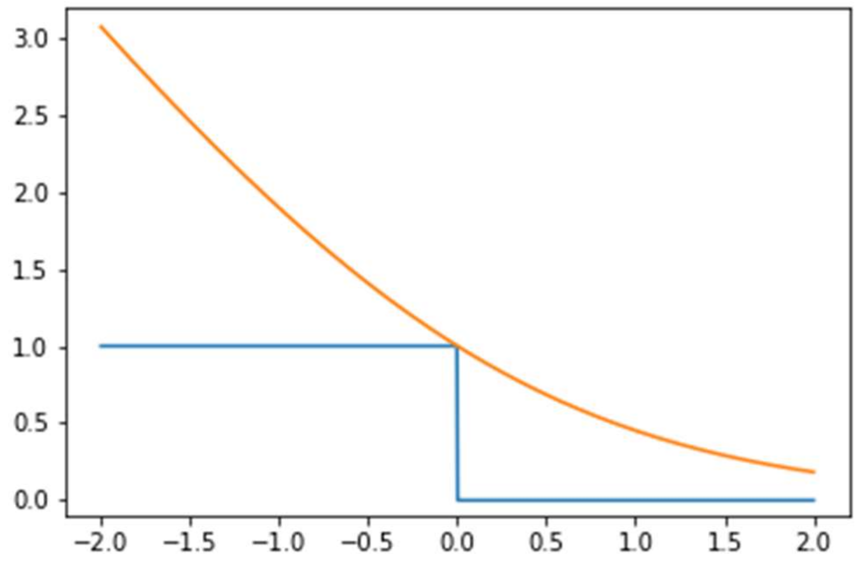
- Нельзя продифференцировать



Верхняя оценка

$$L(M) = [M < 0] \leq \tilde{L}\,(M)$$

- Оценим сверху дифференцируемой функцией



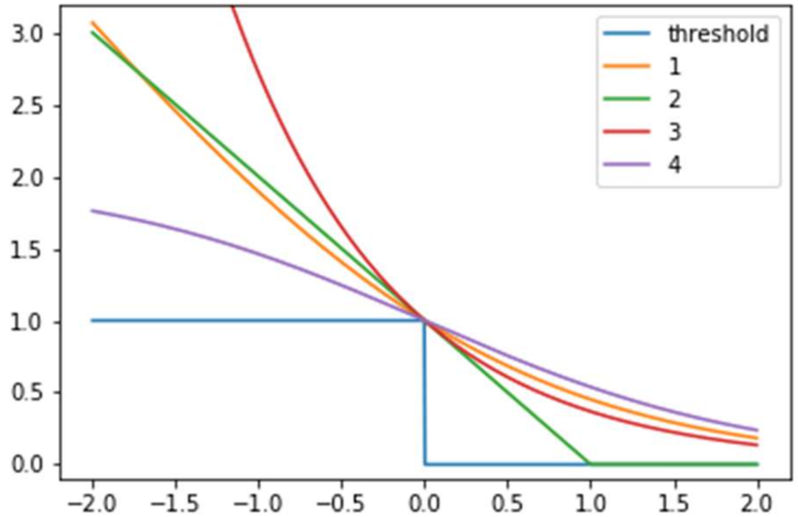
Верхняя оценка

$$0 \le \frac{1}{\ell}\sum_{i=1}^{\ell}[y_i\langle w, x_i\rangle < 0] \le \frac{1}{\ell}\sum_{i=1}^{\ell}\tilde{L}(y_i\langle w, x_i\rangle) \to \min_{w}$$

- Минимизируем верхнюю оценку
- Надеемся, что она прижмёт долю ошибок к нулю

Примеры верхних оценок

1. $\tilde{L}(M) = \log(1 + e^{-M})$ — логистическая
2. $\tilde{L}(M) = \max(0, 1 - M)$ — кусочно-линейная
3. $\tilde{L}(M) = e^{-M}$ — экспоненциальная
4. $\tilde{L}(M) = \frac{2}{1+e^{M}}$ — сигмоидная



Пример обучения

- Выбираем логистическую функцию потерь:

$$\tilde{Q}(w, X) = \frac{1}{\ell} \sum_{i=1}^{\ell} \log(1 + \exp(-y_i \langle w, x_i \rangle)) \to \min_{w}$$

- Вычисляем градиент:

$$\nabla_w \tilde{Q}(w, X) = -\frac{1}{\ell} \sum_{i=1}^{\ell} \frac{y_i x_i}{1 + \exp(y_i \langle w, x_i \rangle)}$$

Пример обучения

- Делаем градиентный спуск:

$$w^{(t)} = w^{(t-1)} + \eta \frac{1}{\ell} \sum_{i=1}^{\ell} \frac{y_i x_i}{1 + \exp(y_i \langle w, x_i \rangle)}$$

Пример регуляризации

$$\frac{1}{\ell}\sum_{i=1}^{\ell}\log(1+\exp(-y_i\langle w, x_i\rangle)) + \lambda\|w\|^2 \to \min_w$$

- Полностью аналогично линейной регрессии

- Важно не накладывать регуляризацию на свободный коэффициент[1]

1 https://medium.com/@shrutijadon10104776/why-we-dont-use-bias-in-regularization-5a86905dfcd6

Метрики качества классификации

Качество классификации

- Доля неправильных ответов:

$$\frac{1}{\ell}\sum_{i=1}^{\ell}[a(x_i) \neq y_i]$$

Качество классификации

- Доля правильных ответов (accuracy):

$$\frac{1}{\ell}\sum_{i=1}^{\ell}[a(x_i) = y_i]$$

Несбалансированные выборки

- Несбалансированная выборка — объектов одного класса существенного больше
- Пример: предсказание кликов по рекламе
- Пример: медицинская диагностика
- Пример: предсказание оттока клиентов
- Пример: специализированный поиск

Несбалансированные выборки

- Пример:
  - Класс -1: 950 объектов
  - Класс +1: 50 объектов
- $a(x) = -1$
- Доля правильных ответов: 0.95
- Почему результат нас не устраивает?

Несбалансированные выборки

- Пример:
  - Класс -1: 950 объектов
  - Класс +1: 50 объектов
- $a(x) = -1$
- Доля правильных ответов: 0.95
- Почему результат нас не устраивает?
- Модель не несёт экономической ценности
- Цены ошибок неравнозначны

Несбалансированные выборки

- q_0 — доля объектов самого крупного класса
- Для разумных алгоритмов:

$$\text{accuracy} \in [q_0, 1]$$

- Если получили большой accuracy — посмотрите на баланс классов

Улучшение метрики

- Два алгоритма
- Доли правильных ответов: r_1 и r_2
- Абсолютное улучшение: $r_2 - r_1$
- Относительное улучшение: $\frac{r_2 - r_1}{r_1}$

Улучшение метрики

- $r_1 = 0.8$
- $r_2 = 0.9$
- $\frac{r_2 - r_1}{r_1} = 12.5\%$

- $r_1 = 0.5$
- $r_2 = 0.75$
- $\frac{r_2 - r_1}{r_1} = 50\%$

- $r_1 = 0.001$
- $r_2 = 0.01$
- $\frac{r_2 - r_1}{r_1} = 900\%$

Цены ошибок

- Пример: кредитный скоринг
- Что хуже?
  - Выдать кредит «плохому» клиенту
  - Не выдать кредит «хорошему» клиенту
- Доля верных ответов не учитывает цены ошибок

Матрица ошибок

	$\boldsymbol{y = 1}$	$\boldsymbol{y = -1}$
$a(x) = 1$	True Positive (TP)	False Positive (FP)
$a(x) = -1$	False Negative (FN)	True Negative (TN)

Матрица ошибок

- Модель $a_1(x)$:

	$\boldsymbol{y = 1}$	$\boldsymbol{y = -1}$
$a(x) = 1$	80	20
$a(x) = -1$	20	80

- Модель $a_2(x)$:

	$\boldsymbol{y = 1}$	$\boldsymbol{y = -1}$
$a(x) = 1$	48	2
$a(x) = -1$	52	98

Точность (precision)

- Можно ли доверять классификатору при $a(x) = 1$?

$$\text{precision}(a, X) = \frac{TP}{TP + FP}$$

Точность (precision)

- Модель $a_1(x)$:

	$\boldsymbol{y = 1}$	$\boldsymbol{y = -1}$
$a(x) = 1$	80	20
$a(x) = -1$	20	80

- $\text{precision}(a_1, X) = 0.8$

- Модель $a_2(x)$:

	$\boldsymbol{y = 1}$	$\boldsymbol{y = -1}$
$a(x) = 1$	48	2
$a(x) = -1$	52	98

- $\text{precision}(a_2, X) = 0.96$

Полнота (recall)

- Как много положительных объектов находит классификатор?

$$\text{recall}(a, X) = \frac{TP}{TP + FN}$$

Полнота (recall)

- Модель $a_1(x)$:

	$y = 1$	$y = -1$
$a(x) = 1$	80	20
$a(x) = -1$	20	80

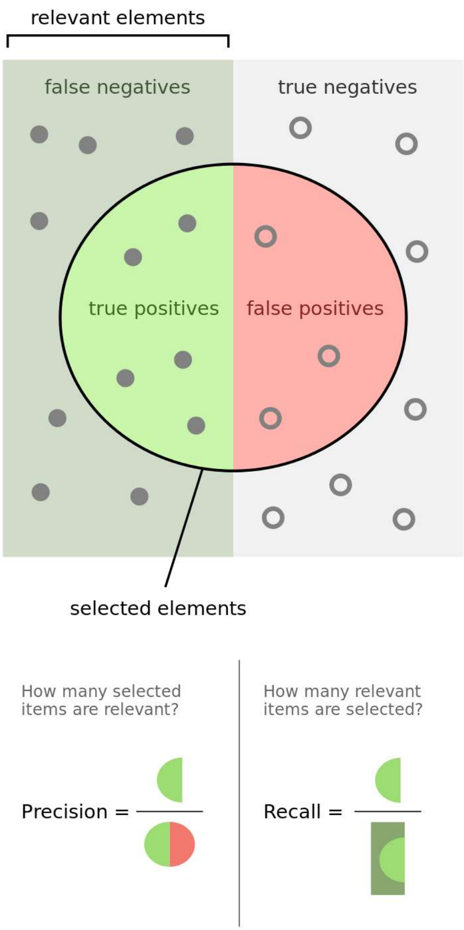
- $\text{recall}(a_1, X) = 0.8$

- Модель $a_2(x)$:

	$y = 1$	$y = -1$
$a(x) = 1$	48	2
$a(x) = -1$	52	98

- $\text{recall}(a_2, X) = 0.48$

Точность и полнота



https://en.wikipedia.org/wiki/Precision_and_recall

Антифрод

- Классификация транзакций на нормальные и мошеннические
- Высокая точность, низкая полнота:
  - Редко блокируем нормальные транзакции
  - Пропускаем много мошеннических
- Низкая точность, высокая полнота:
  - Часто блокируем нормальные транзакции
  - Редко пропускаем мошеннические

Кредитный скоринг

- Неудачных кредитов должно быть не больше 5%
- Ограничение: $\text{precision}(a, X) \geq 0.95$
- Максимизируем полноту

Медицинская диагностика

- Надо найти не менее 80% больных
- Ограничение: $\text{recall}(a, X) \geq 0.8$
- Максимизируем точность

Несбалансированные выборки

- $\text{accuracy}(a, X) = 0.99$
- $\text{precision}(a, X) = 0.33$
- $\text{recall}(a, X) = 0.1$

	$y = 1$	$y = -1$
$a(x) = 1$	10	20
$a(x) = -1$	90	10000

Совмещение точности и полноты

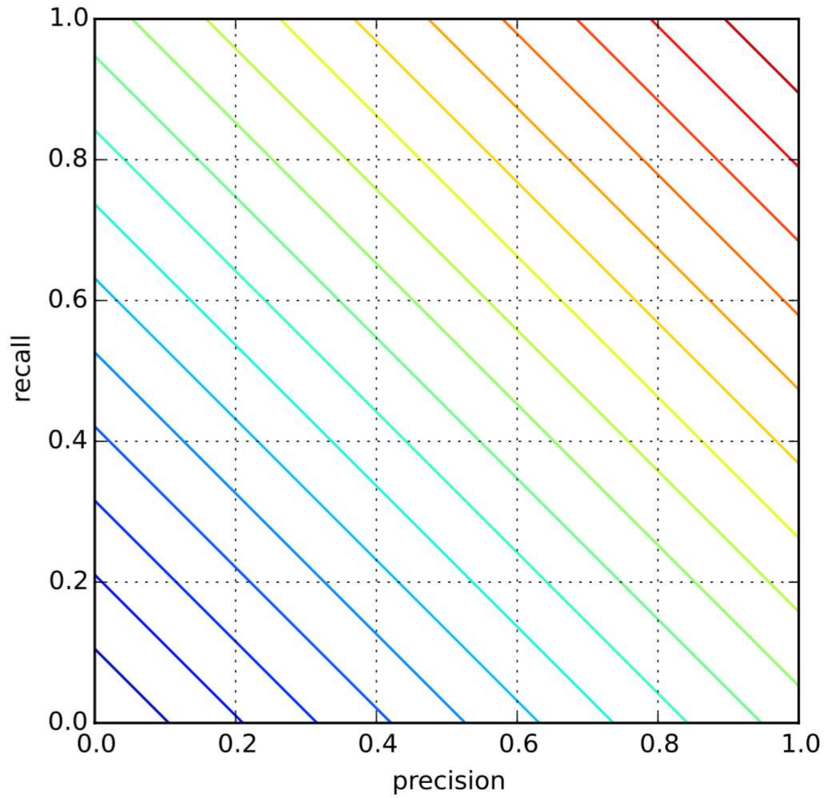
Точность и полнота

- Точность — можно ли доверять классификатору при $a(x) = 1$?
- Полнота — как много положительных объектов находит $a(x)$?

- Оптимизировать две метрики одновременно очень неудобно
- Как объединить?

Арифметическое среднее

$$A = \frac{1}{2}(\text{precision} + \text{recall})$$

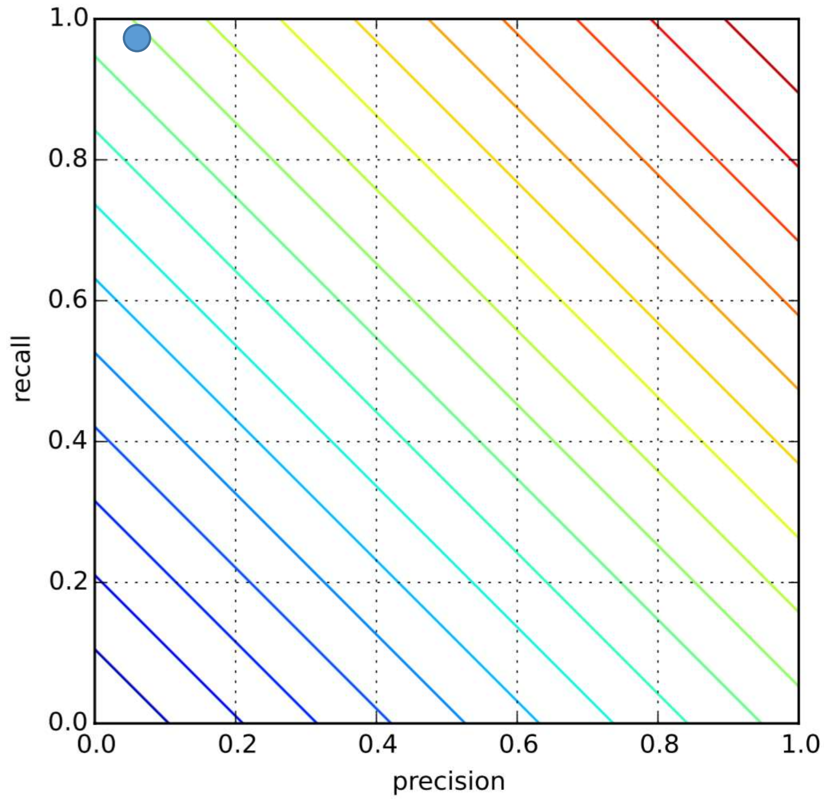


Арифметическое среднее

$$A = \frac{1}{2}(\text{precision} + \text{recall})$$

- $\text{precision} = 0.1$
- $\text{recall} = 1$
- $A = 0.55$

- Плохой алгоритм

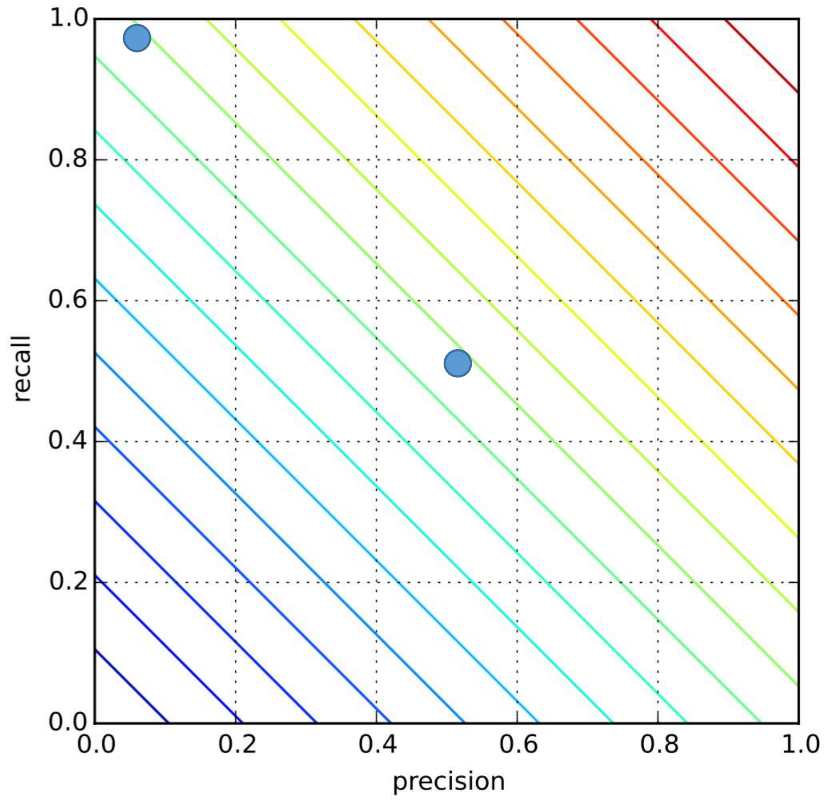


Арифметическое среднее

$$A = \frac{1}{2}(\text{precision} + \text{recall})$$

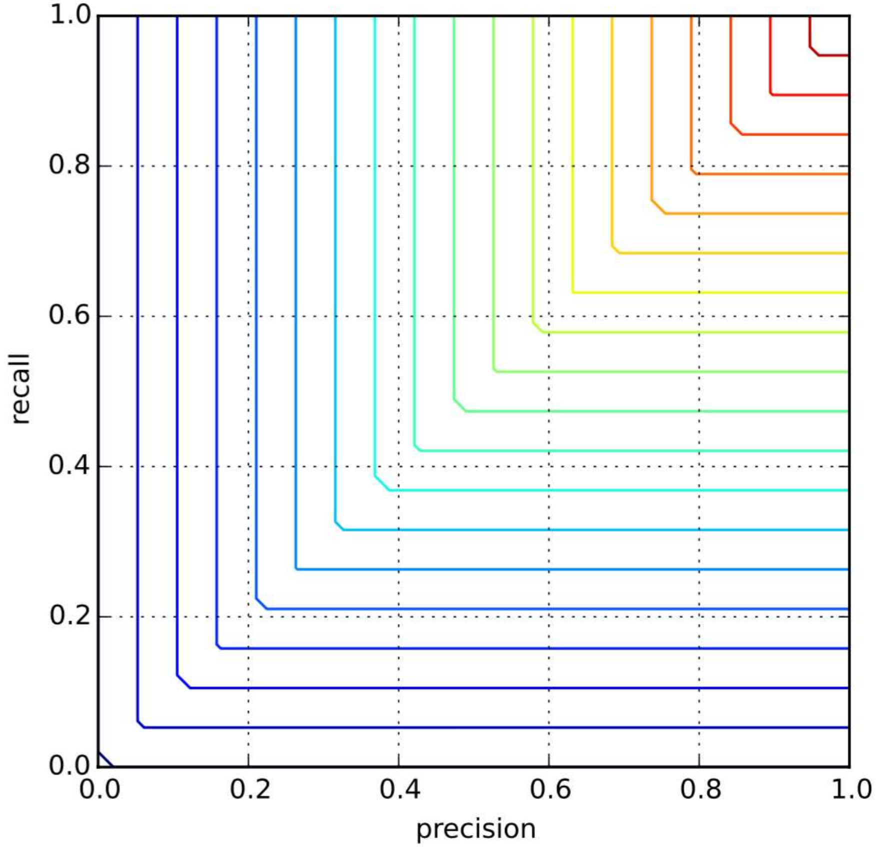
- $\text{precision} = 0.55$
- $\text{recall} = 0.55$
- $A = 0.55$

- Нормальный алгоритм
- Но качество такое же, как у плохого



Минимум

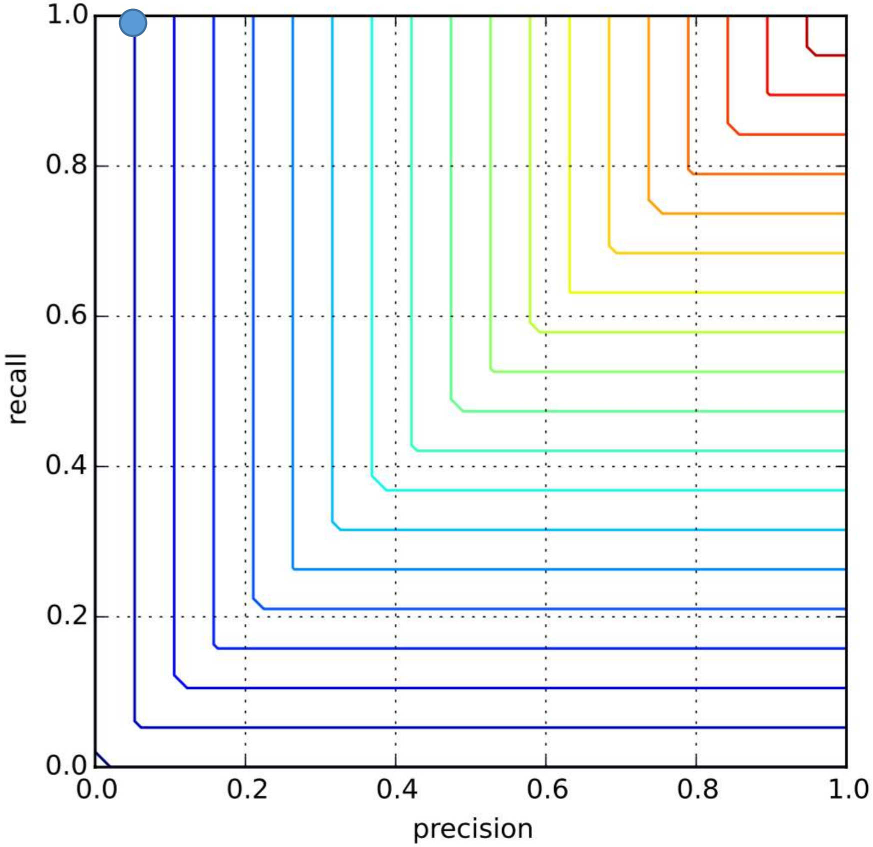
$$M = \min(\text{precision}, \text{recall})$$



Минимум

$$M = \min(\text{precision}, \text{recall})$$

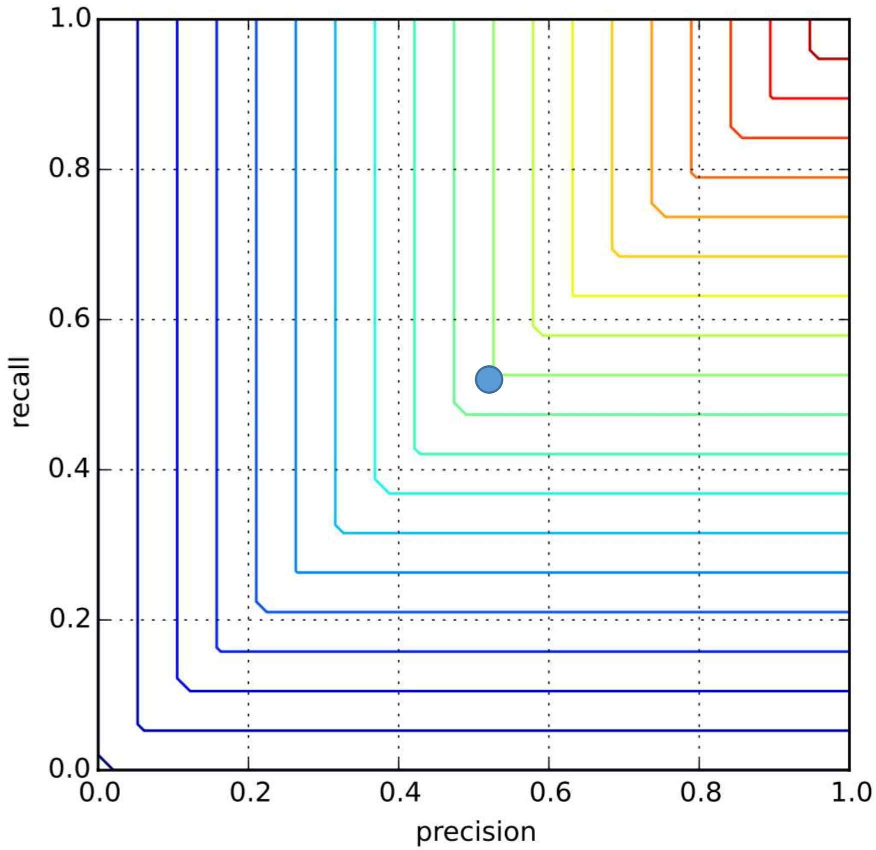
- $\text{precision} = 0.05$
- $\text{recall} = 1$
- $M = 0.05$



Минимум

$$M = \min(\text{precision}, \text{recall})$$

- $\text{precision} = 0.55$
- $\text{recall} = 0.55$
- $M = 0.55$



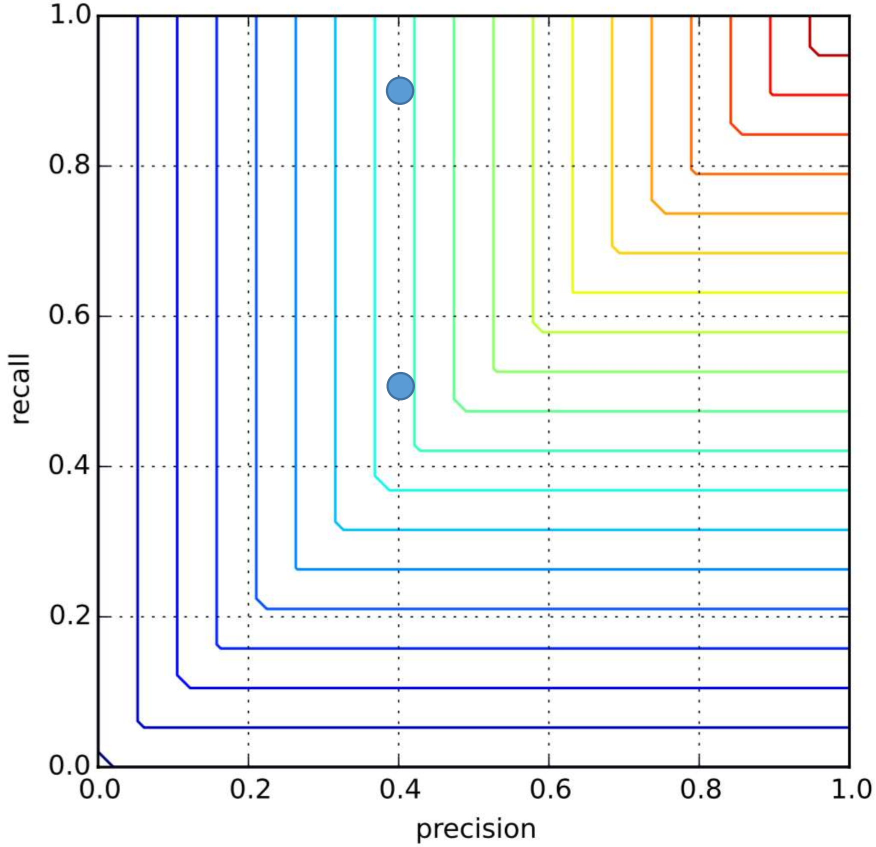
Минимум

$$M = \min(\text{precision}, \text{recall})$$

- $\text{precision} = 0.4, \text{recall} = 0.5$
- $M = 0.4$

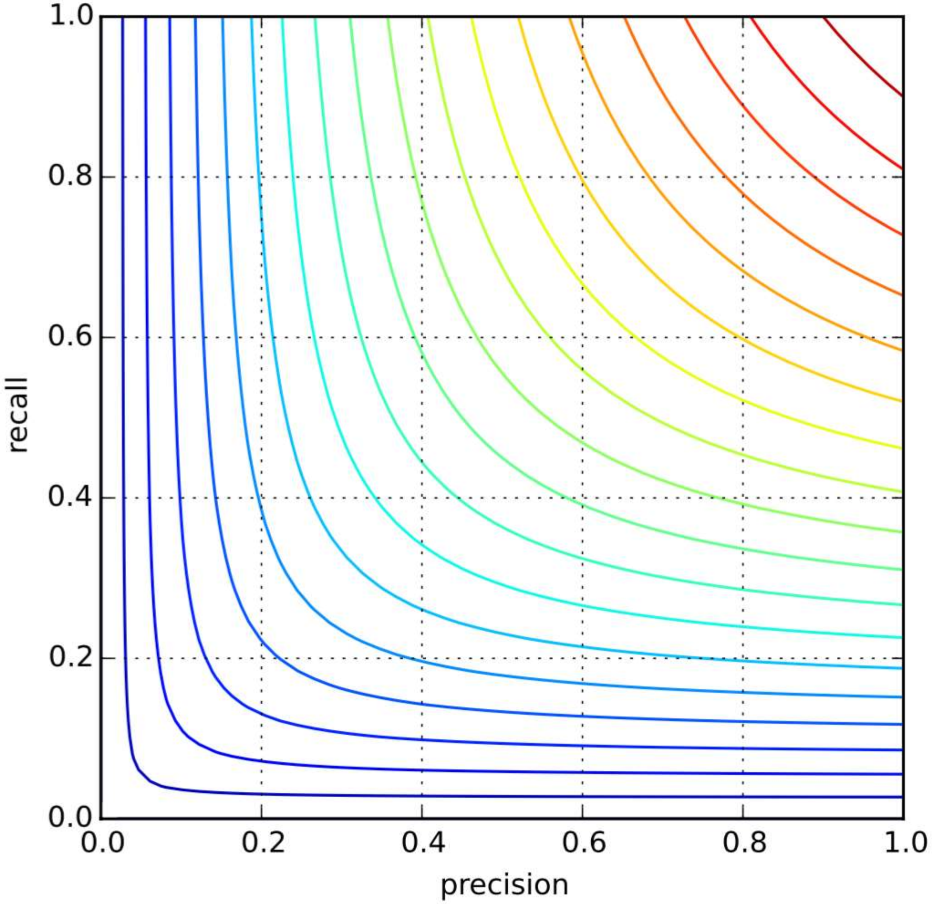
- $\text{precision} = 0.4, \text{recall} = 0.9$
- $M = 0.4$

- Но второй лучше!



F-мера

$$F = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

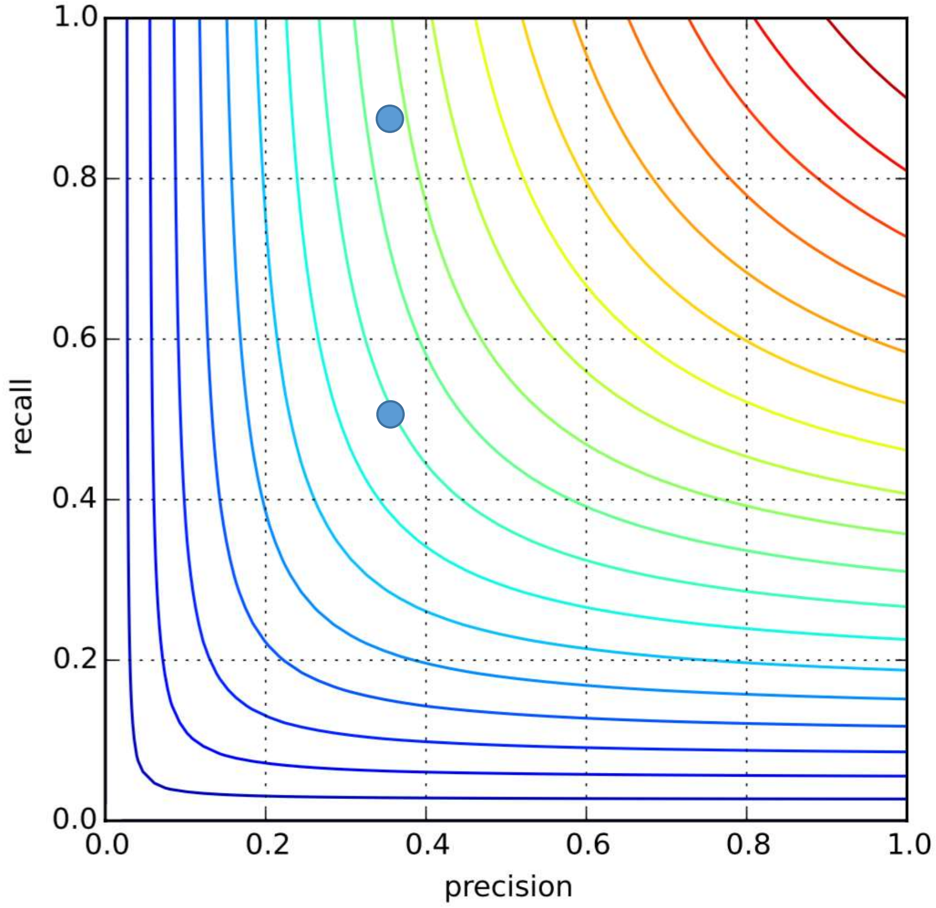


F-мера

$$F = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

- $\text{precision} = 0.4, \text{recall} = 0.5$
- $F = 0.44$

- $\text{precision} = 0.4, \text{recall} = 0.9$
- $\text{F} = 0.55$



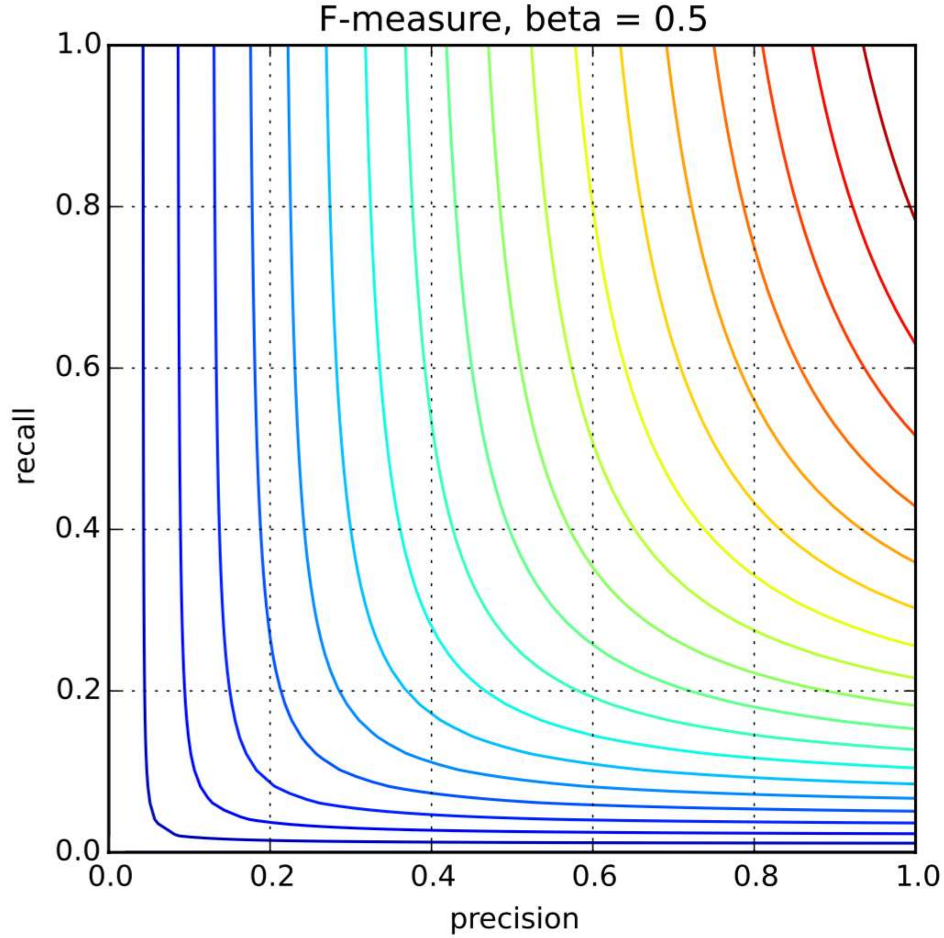
F-мера

$$F_\beta = (1+\beta^2)\frac{\text{precision} * \text{recall}}{\beta^2 * \text{precision} + \text{recall}}$$

F-мера

$$F_\beta = (1+\beta^2)\frac{\text{precision} * \text{recall}}{\beta^2 * \text{precision} + \text{recall}}$$

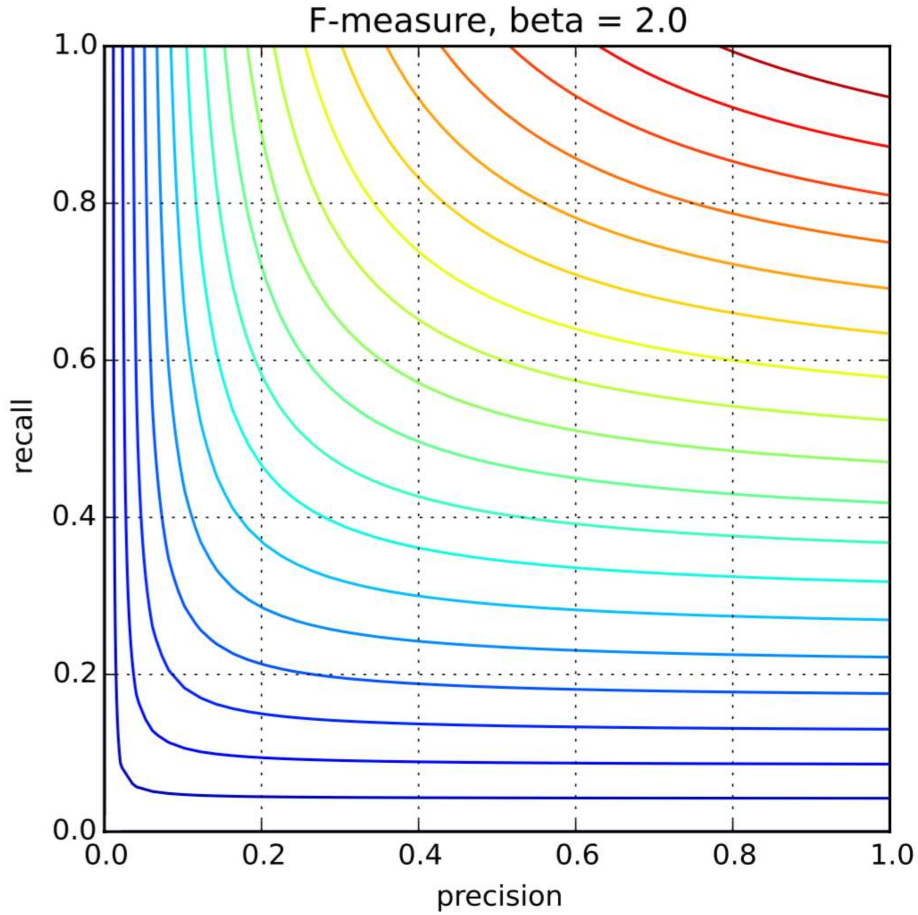
- $\beta = 0.5$
- Важнее точность



F-мера

$$F_\beta = (1+\beta^2)\frac{\text{precision} * \text{recall}}{\beta^2 * \text{precision} + \text{recall}}$$

- $\beta = 2$
- Важнее полнота



Метрики качества ранжирования

Классификатор

- Линейный классификатор:

$$a(x) = \mathrm{sign}(\langle w, x \rangle - t) = 2[\langle w, x \rangle > t] - 1$$

- $\langle w, x \rangle$ — оценка принадлежности классу +1
- Нередко $t = 0$

Оценка принадлежности

- Как оценить качество $b(x)$?
- Порог зависит от ограничений на точность или полноту

Оценка принадлежности

- Высокий порог:
  - Мало объектов относим к +1
  - Точность выше
  - Полнота ниже
- Низкий порог:
  - Много объектов относим к +1
  - Точность ниже
  - Полнота выше

Оценка принадлежности

-1	-1	+1	-1	-1	-1	+1	+1	-1	+1
0.01	0.09	0.12	0.15	0.29	0.4	0.48	0.6	0.83	0.9

Оценка принадлежности

-1	-1	+1	-1	-1	-1	+1	+1	-1	+1
0.01	0.09	0.12	0.15	0.29	0.4	0.48	0.6	0.83	0.9

Оценка принадлежности

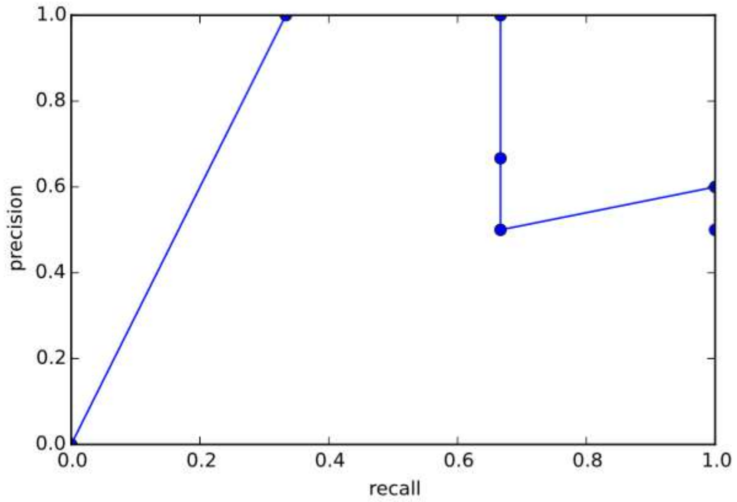
-1	-1	+1	-1	-1	-1	+1	+1	-1	+1
0.01	0.09	0.12	0.15	0.29	0.4	0.48	0.6	0.83	0.9

Оценка принадлежности

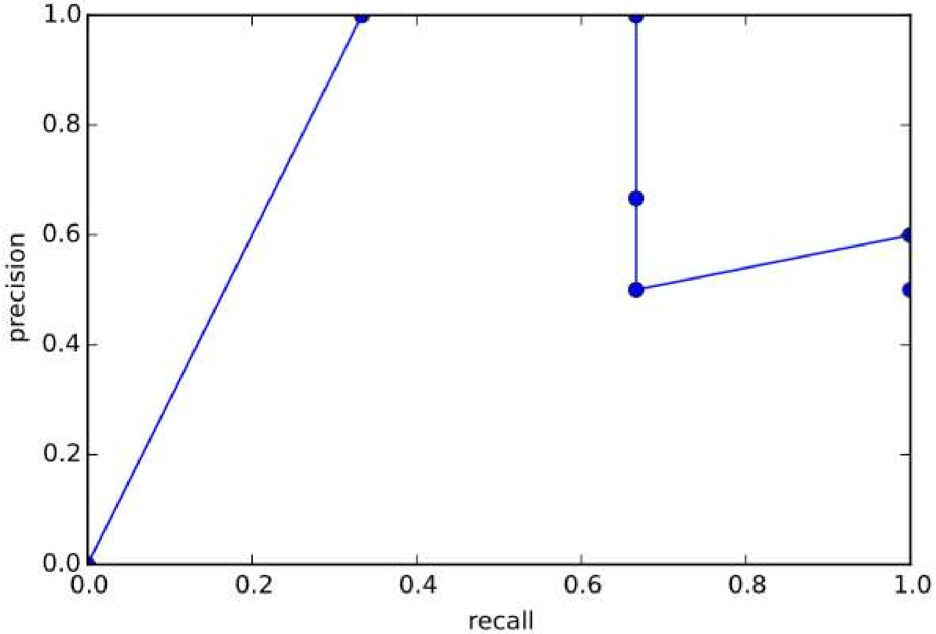
- Пример: кредитный скоринг
- $b(x)$ — оценка вероятности возврата кредита
- $a(x) = [b(x) > 0.5]$
- $\text{precision} = 0.1, \text{recall} = 0.7$
- В чем дело — в пороге или в алгоритме?

PR-кривая

- Кривая точности-полноты
- Ось X — полнота
- Ось Y — точность
- Точки — значения точности и полноты при последовательных порогах

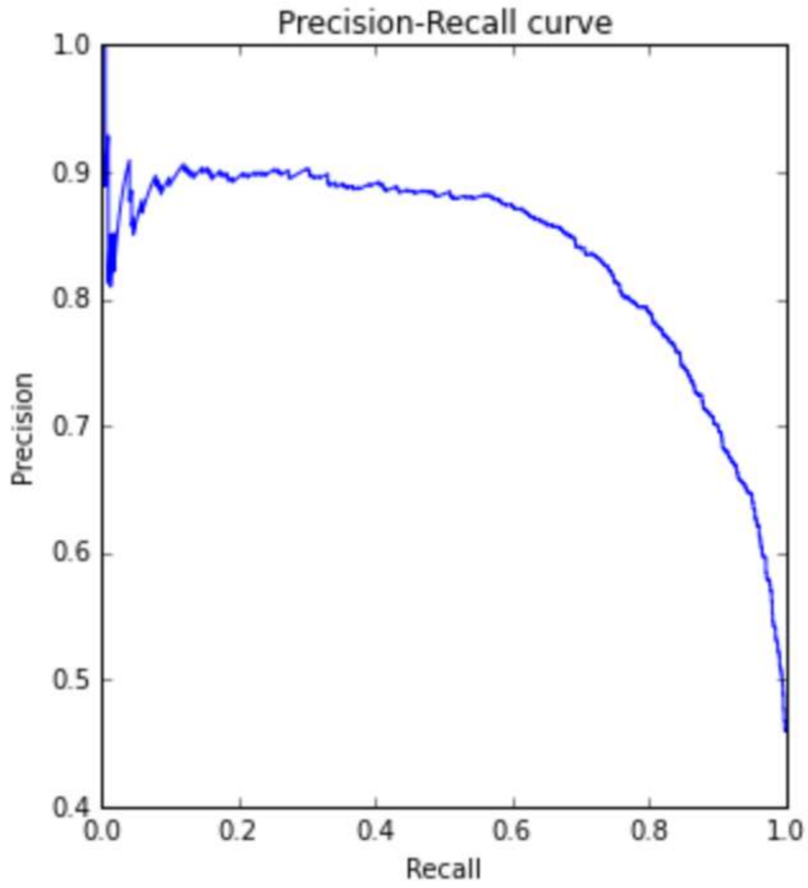


PR-кривая



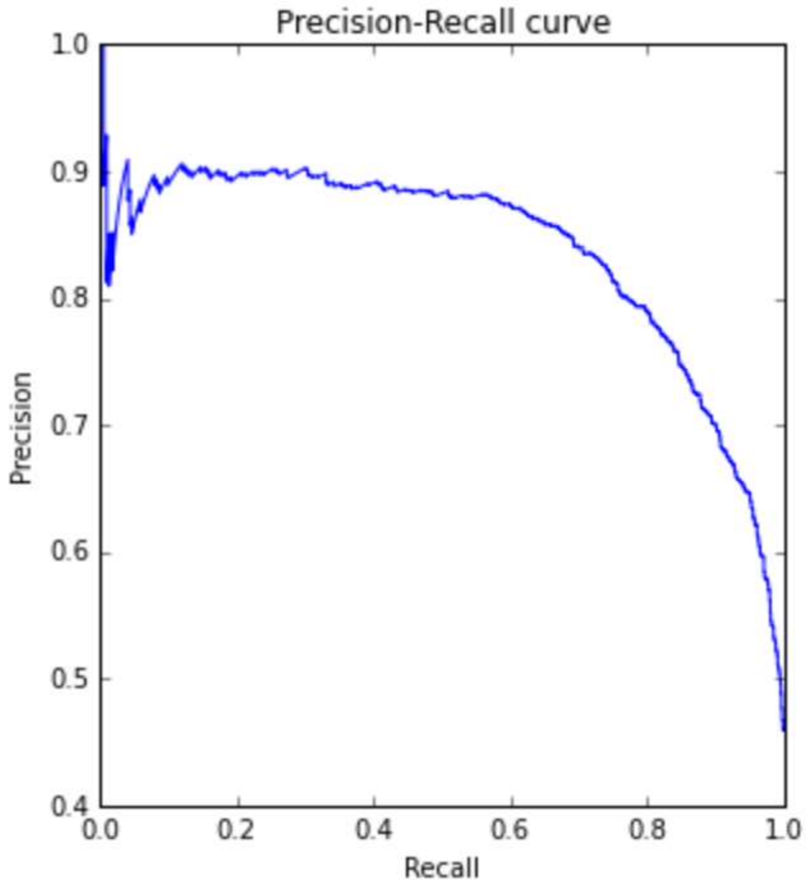
$b(x)$	0.14	0.23	0.39	0.52	0.73	0.90
y	0	1	0	0	1	1

PR-кривая в реальности

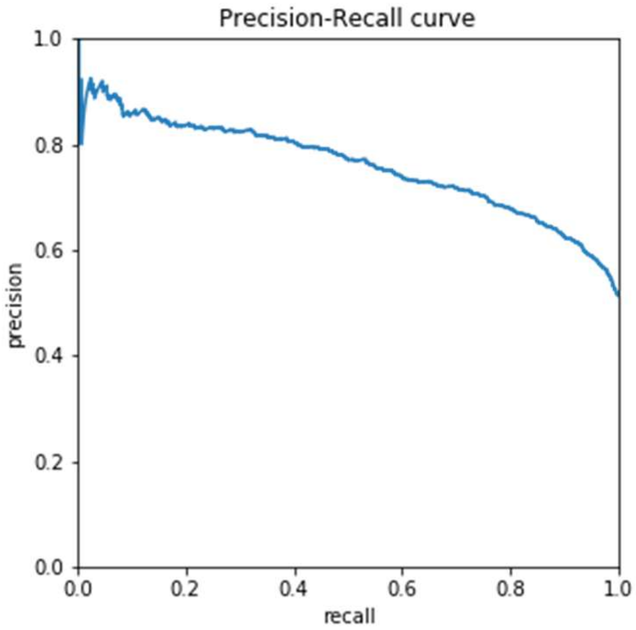


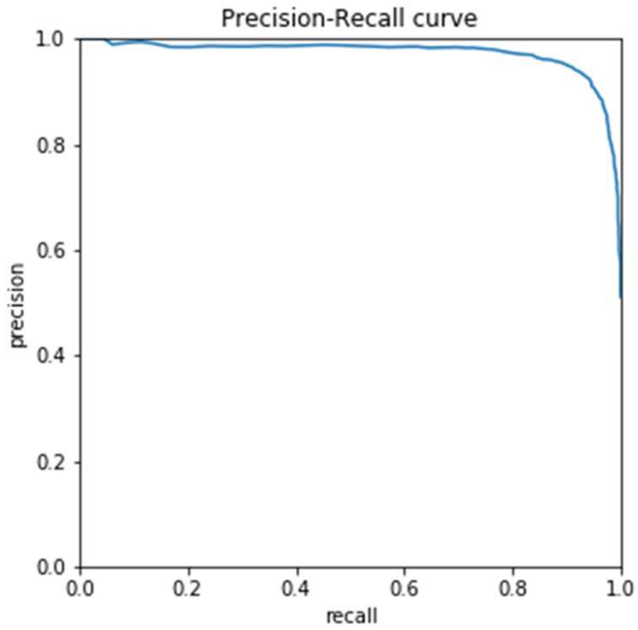
PR-кривая

- Левая точка: (0, 0) или (0, 1)
- Правая точка: $(1, r)$, r — доля положительных объектов
- Для идеального классификатора проходит через (1, 1)
- AUC-PRC — площадь под PR-кривой



PR-кривая


Precision-Recall curve
1.0
0.8
0.6
0.4
0.2
0.0
precision
0.0
0.2
0.4
0.6
0.8
1.0
recall


Precision-Recall curve
1.0
0.8
0.6
0.4
0.2
0.0
precision
0.0
0.2
0.4
0.6
0.8
1.0
recall

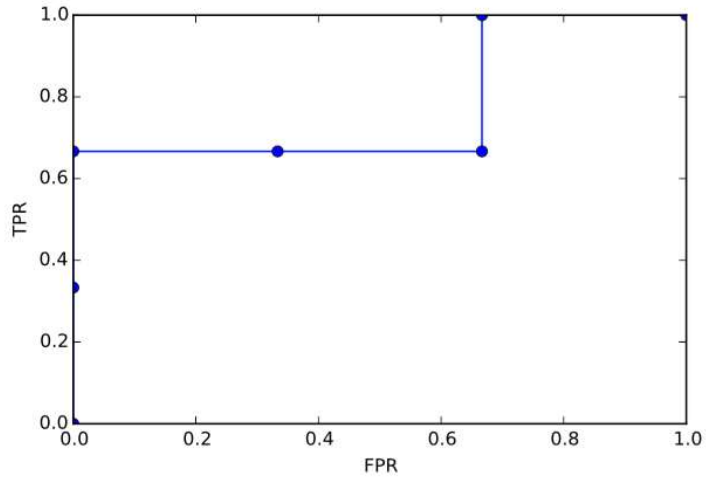
ROC-кривая

- Receiver Operating Characteristic
- Ось X — False Positive Rate

$$FPR = \frac{FP}{FP + TN}$$

- Ось Y — True Positive Rate

$$TPR = \frac{TP}{TP + FN}$$



ROC-кривая

- Receiver Operating Characteristic
- Ось X — False Positive Rate

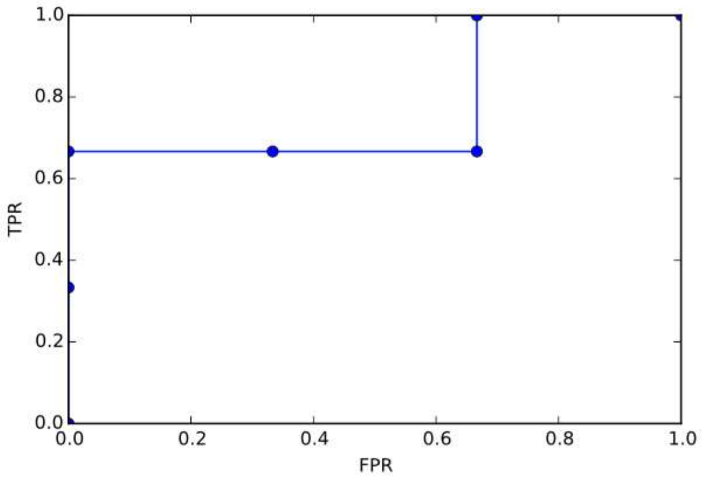
$$FPR = \frac{FP}{\boxed{FP + TN}}$$

Число отрицательных объектов

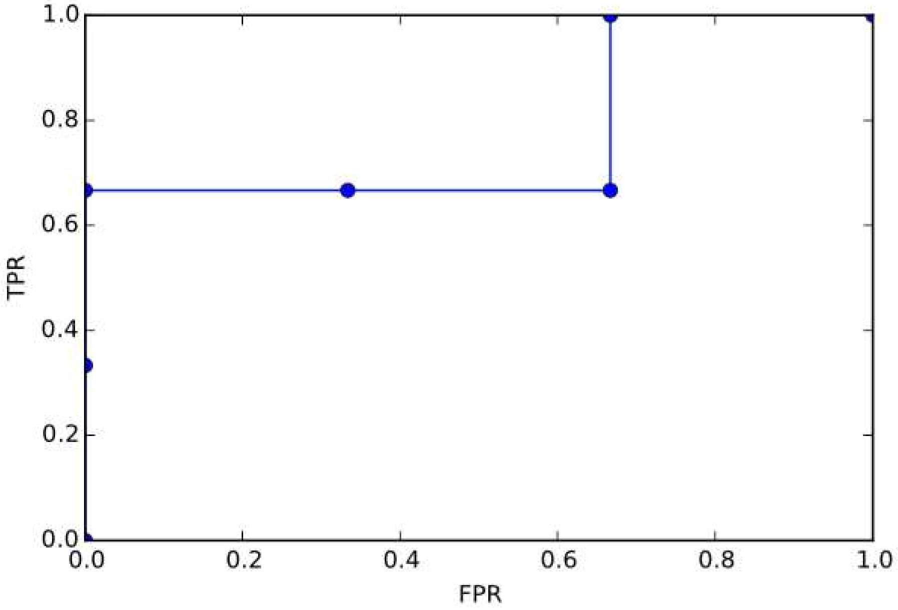
- Ось Y — True Positive Rate

$$TPR = \frac{TP}{\boxed{TP + FN}}$$

Число положительных объектов

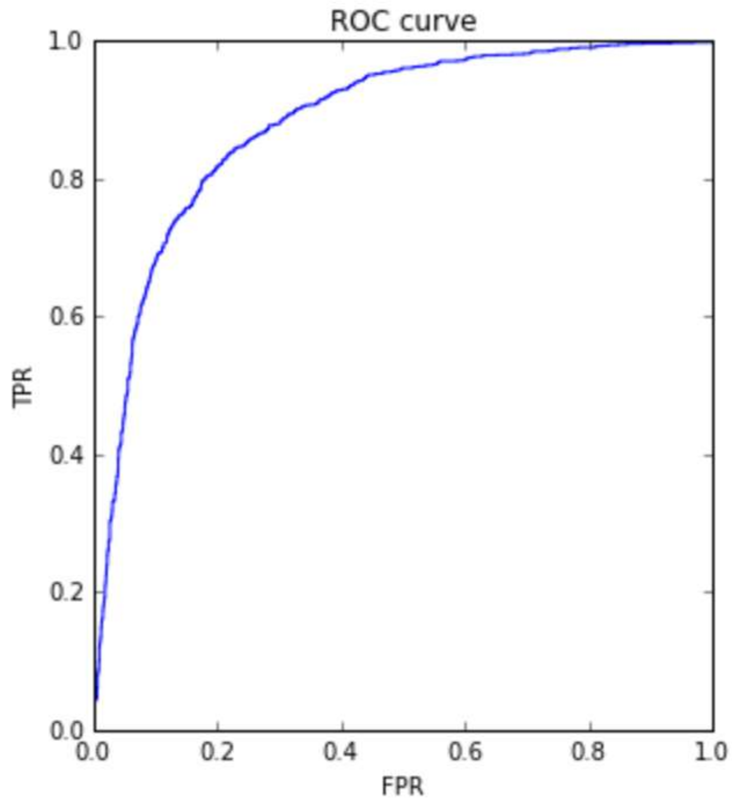


ROC-кривая



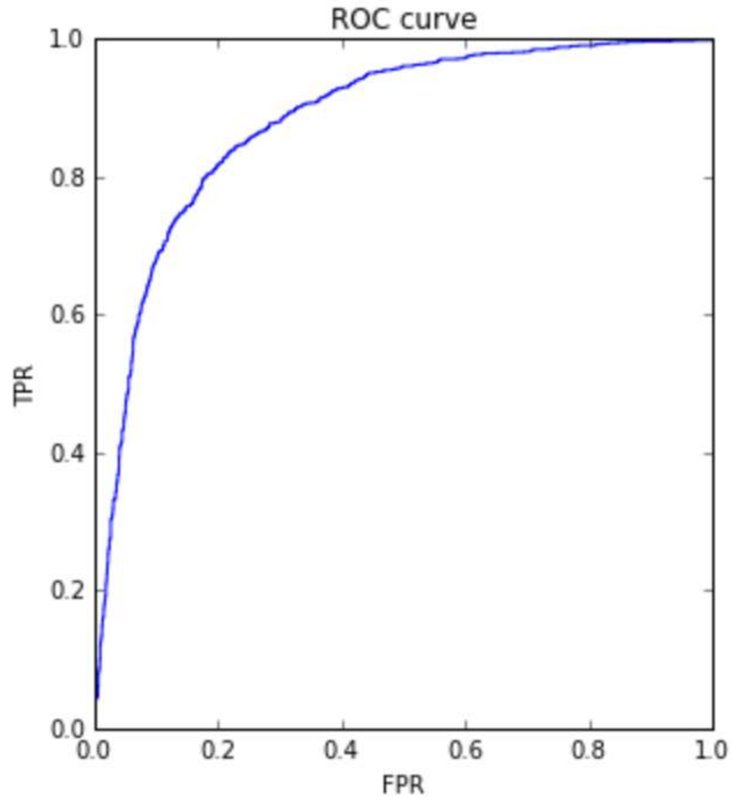
$b(x)$	0.14	0.23	0.39	0.52	0.73	0.90
y	0	1	0	0	1	1

ROC-кривая в реальности

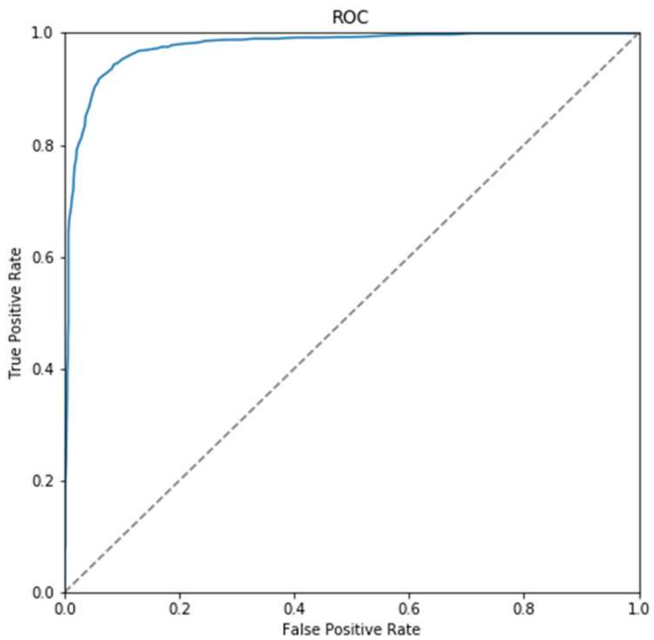


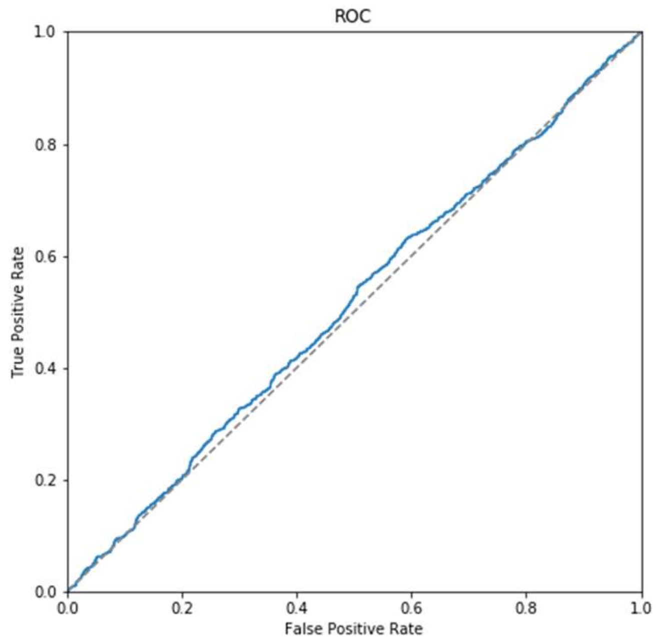
ROC-кривая

- Левая точка: (0, 0)
- Правая точка: $(1, 1)$
- Для идеального классификатора проходит через (0, 1)
- AUC-ROC — площадь под ROC-кривой



ROC-кривая


ROC
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0


ROC
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0

AUC-ROC

$$FPR = \frac{FP}{FP+TN}; \qquad TPR = \frac{TP}{TP+FN}$$

- FP и TP нормируются на размеры классов
- AUC-ROC не поменяется при изменении баланса классов
- Учитывает True Negatives
- Идеальный алгоритм: $\mathrm{AUC{-}ROC} = 1$
- Худший алгоритм: $\mathrm{AUC{-}ROC} \approx 0.5$

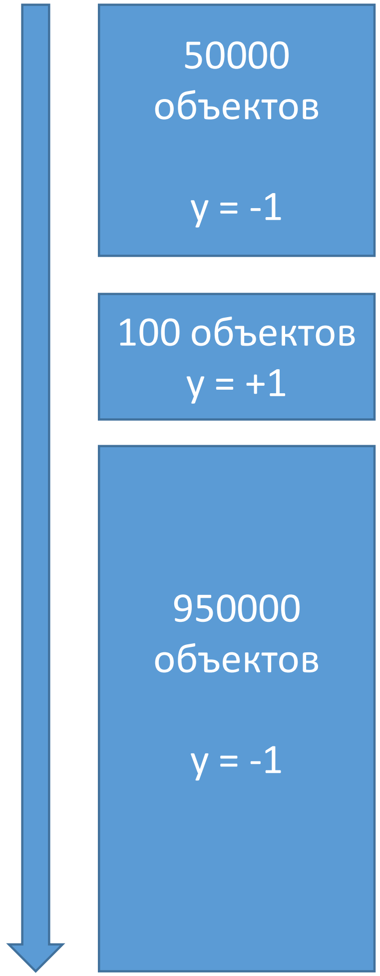
AUC-PRC

$$\text{precision} = \frac{TP}{TP+FP}; \qquad \text{recall} = \frac{TP}{TP+FN}$$

- Точность поменяется при изменении баланса классов
- AUC-PRC идеального алгоритма зависит от баланса классов
- Не учитывает True Negatives
- Проще интерпретировать, если выборка несбалансированная
- Лучше, если задачу надо решать в терминах точности и полноты

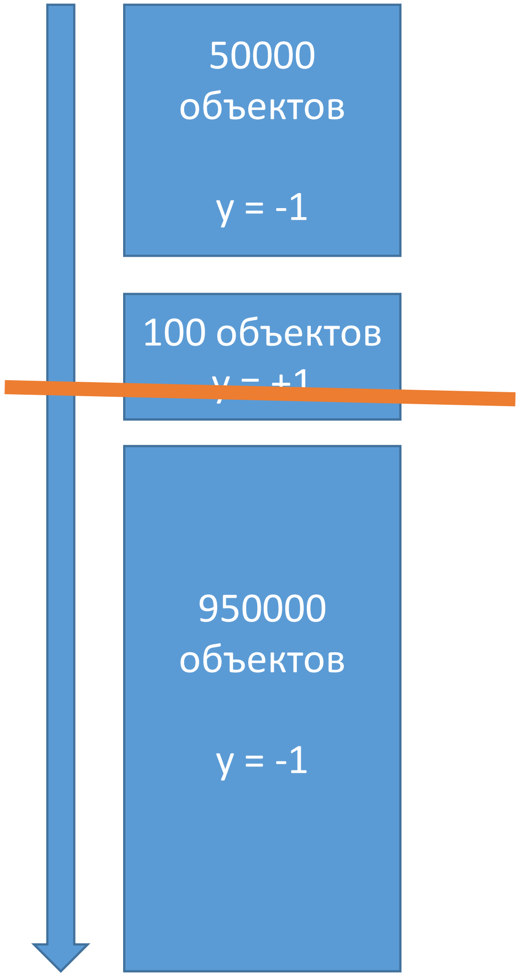
Пример

- AUC-ROC = 0.95
- AUC-PRC = 0.001



Пример

- Выберем конкретный классификатор
- $a(x) = 1$ — 50095 объектов
- Из них FP = 50000, TP = 95
- TPR = 0.95, FPR = 0.05
- precision = 0.0019, recall = 0.95



AUC-ROC vs AUC-PRC

Randy C • 6 years ago • Options • Reply   56

The way I think about the difference between ROC and precision-recall is in how each treats true negatives. Typically, if true negatives are not meaningful to the problem or negative examples just dwarf the number of positives, precision-recall is typically going to be more useful; otherwise, I tend to stick with ROC since it tends to be an easier metric to explain in most circles.

For illustration, let's take an example of an information retrieval problem where we want to find a set of, say, 100 relevant documents out of a list of 1 million possibilities based on some query. Let's say we've got two algorithms we want to compare with the following performance:

- Method 1: 100 retrieved documents, 90 relevant
- Method 2: 2000 retrieved documents, 90 relevant

Clearly, Method 1's result is preferable since they both come back with the same number of relevant results, but Method 2 brings a ton of false positives with it. The ROC measures of TPR and FPR will reflect that, but since the number of irrelevant documents dwarfs the number of relevant ones, the difference is mostly lost:

- Method 1: 0.9 TPR, 0.00001 FPR
- Method 2: 0.9 TPR, 0.00191 FPR (difference of 0.0019)

Precision and recall, however, don't consider true negatives and thus won't be affected by the relative imbalance (which is precisely why they're used for these types of problems):

- Method 1: 0.9 recall, 0.9 precision
- Method 2: 0.9 recall, 0.045 precision (difference of 0.855)

Obviously, those are just single points in ROC and PR space, but if these differences persist across various scoring thresholds, using ROC AUC, we'd see a very small difference between the two algorithms, whereas PR AUC would show quite a large difference.

https://www.kaggle.com/general/7517#41179