

Метрики классификации

Семинар 6

решения.

Задача 5 (0.5 балла). Если у вас осталось время, то напишите куда можно сходить в Москве.

ИМХ ВШЭ

План

- Мотивация
- Бинарная классификация
- Confusion matrix
- Accuracy
- Precision
- Recall
- F1-Score
- ROC-AUC и PR-AUC

Мотивация

Метрики нужны бизнесу

MAE = 100



MAE = 1000



Мотивация

А какие метрики для классификации?

Мотивация

А какие метрики для классификации?

Доля верных ошибок

Бинарная классификация

Бинарная классификация – это когда модель решает, принадлежит ли объект к **одному из двух** возможных классов.

$$a(x) = \sigma(< w, x >)$$

$$y \in \{0,1\}$$

Бинарная классификация

Пример	1	0
Диагностика заболеваний	Болен	Здоров
Детекция фрода	Фрод	Не фрод
Детекция спама	Спам	Не спам
Кредитный scoring	Вернет кредит	Не вернет
Списывал на проверочной	Да	Нет

Confusion matrix

		Actual Value	
		positives	negatives
Predicted Value	positives	TP True Positive	FP False Positive
	negatives	FN False Negative	TN True Negative

TP (True Positive) – модель правильно нашла позитив

FP (False Positive) – сработала зря

FN (False Negative) – пропустила нужный случай

TN (True Negative) – всё верно, негатив остался негативом

Confusion matrix

Пример: Мы работаем в службе безопасности банка. У нас есть модель, которая выявляет мошеннические транзакции (Фрод)

		Actual Value	
		positives	negatives
Predicted Value	positives	TP True Positive	FP False Positive
	negatives	FN False Negative	TN True Negative

TP

Факт: Транзакция действительно мошенническая

Предсказание: Модель сказала Фрод

Итог: Все круто

Confusion matrix

		Actual Value	
		positives	negatives
Predicted Value	positives	TP True Positive	FP False Positive
	negatives	FN False Negative	TN True Negative

TN

Факт: Обычная покупка (легально)

Предсказание: Модель сказала Не
Фрод

Итог: Все круто

Confusion matrix

		Actual Value	
		positives	negatives
Predicted Value	positives	TP True Positive	FP False Positive
	negatives	FN False Negative	TN True Negative

FP

Факт: Обычная покупка (легально)

Предсказание: Модель сказала Фрод.
Блокируем карту

Итог: Мы вызвали ложную тревогу
(клиент обиделся и ушел от нас)

Confusion matrix

		Actual Value	
		positives	negatives
Predicted Value	positives	TP True Positive	FP False Positive
	negatives	FN False Negative	TN True Negative

FN

Факт: Транзакция мошенническая

Предсказание: Модель сказала Не Фрод. Пропускаем платеж

Итог: Мы допустили мошенническую операцию, где-то пострадала 1 бабушка

Accuracy

Accuracy - это доля правильных предсказаний среди всех примеров

Accuracy отвечает на очень простой вопрос:

“Какую долю примеров модель угадала правильно?”

		Actual Value	
		positives	negatives
Predicted Value	positives	TP True Positive	FP False Positive
	negatives	FN False Negative	TN True Negative

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Accuracy

У нас есть модель, которая диагностирует редкую болезнь.
Она угадывает правильно в 99% случаев.

Accuracy = 0.99

Хорошая ли это модель?

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$



Accuracy

У нас есть модель, которая диагностирует редкую болезнь.
Она угадывает правильно в 99% случаев.

Accuracy = 0.99

Хорошая ли это модель? (Конечно нет)

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Всего 1000 пациентов.

True Negative = 990

True Positive = 0

False Negative = 10

False Positive = 0



Precision

Precision показывает, какая доля из всех примеров, которые модель предсказала как положительные, действительно оказалась положительной.

		Actual Value	
		positives	negatives
Predicted Value	positives	TP True Positive	FP False Positive
	negatives	FN False Negative	TN True Negative

$$\text{Precision} = \frac{TP}{TP + FP}$$

Precision

Задача скоринга - модель предсказывает, выдавать клиенту кредит или нет.

Сравним между собой две модели. Ниже - их матрицы ошибок на тестовых данных (данные сбалансированы: всего 200 клиентов, по 100 клиентов каждого класса):

	$y = 1$ Могут вернуть	$y = -1$ Не могут вернуть
$a(x) = 1$ Получили кредит	80	20
$a(x) = -1$ Не получили кредит	20	80

	$y = 1$ Могут вернуть	$y = -1$ Не могут вернуть
$a(x) = 1$ Получили кредит	48	2
$a(x) = -1$ Не получили кредит	52	98

$$\text{Precision} = \frac{TP}{TP + FP}$$

Precision

Задача скоринга - модель предсказывает, выдавать клиенту кредит или нет.

Сравним между собой две модели. Ниже - их матрицы ошибок на тестовых данных (данные сбалансированы: всего 200 клиентов, по 100 клиентов каждого класса):

	$y = 1$ Могут вернуть	$y = -1$ Не могут вернуть
$a(x) = 1$ Получили кредит	80	20
$a(x) = -1$ Не получили кредит	20	80

👎 Precision_m1 =
0.8

	$y = 1$ Могут вернуть	$y = -1$ Не могут вернуть
$a(x) = 1$ Получили кредит	48	2
$a(x) = -1$ Не получили кредит	52	98

👍 Precision_m2 =
0.96

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall

Recall показывает, какую долю из всех положительных примеров, модель действительно нашла

		Actual Value	
		positives	negatives
Predicted Value	positives	TP True Positive	FP False Positive
	negatives	FN False Negative	TN True Negative

$$\text{Recall} = \frac{TP}{TP + FN}$$

Recall

Как много кредитоспособных клиентов (тех, кто вернет кредит) находят модели?
Среди всех клиентов, кто вернет кредит, посчитаем долю клиентов, которым модель выдала кредит

	$y = 1$ Могут вернуть	$y = -1$ Не могут вернуть
$a(x) = 1$ Получили кредит	80	20
$a(x) = -1$ Не получили кредит	20	80

	$y = 1$ Могут вернуть	$y = -1$ Не могут вернуть
$a(x) = 1$ Получили кредит	48	2
$a(x) = -1$ Не получили кредит	52	98

$$\text{Recall} = \frac{TP}{TP + FN}$$

Recall

Как много кредитоспособных клиентов (тех, кто вернет кредит) находят модели?
Среди всех клиентов, кто вернет кредит, посчитаем долю клиентов, которым модель выдала кредит

	$y = 1$ Могут вернуть	$y = -1$ Не могут вернуть
$a(x) = 1$ Получили кредит	80	20
$a(x) = -1$ Не получили кредит	20	80

👍 Recall_m1 = 0.8

	$y = 1$ Могут вернуть	$y = -1$ Не могут вернуть
$a(x) = 1$ Получили кредит	48	2
$a(x) = -1$ Не получили кредит	52	98

👎 Recall_m2 = 0.48

$$\text{Recall} = \frac{TP}{TP + FN}$$

Recall

Как много кредитоспособных клиентов (тех, кто вернет кредит) находят модели?
Среди всех клиентов, кто вернет кредит, посчитаем долю клиентов, которым модель выдала кредит

	$y = 1$ Могут вернуть	$y = -1$ Не могут вернуть
$a(x) = 1$ Получили кредит	80	20
$a(x) = -1$ Не получили кредит	20	80

👍 Recall_m1 = 0.8

	$y = 1$ Могут вернуть	$y = -1$ Не могут вернуть
$a(x) = 1$ Получили кредит	48	2
$a(x) = -1$ Не получили кредит	52	98

👎 Recall_m2 = 0.48

$$\text{Recall} = \frac{TP}{TP + FN}$$

Precision и Recall

Нельзя однозначно ответить, какая из этих двух моделей лучше. Возможно, банку, выдающему кредиты, важно выдавать кредиты только тем, кто их вернёт, и не выдавать кредиты тем, кто не вернет - тогда нужна высокая точность (**precision**). А если же банк хочет выдать кредиты наибольшему количеству клиентов, кто теоретически может вернуть кредит - максимизируется полнота (**recall**).

Так что метрика выбирается, исходя из требований заказчика!

Precision vs Recall

Recall важнее

Сфера

Почему

Пропустить больного (FN)
хуже, чем тревожно проверить
здорового

Поиск
уязвимостей /
безопасность

Пропущенная угроза может
стоить дорого

Поисковая
система

Лучше показать больше
результатов (даже
нерелевантных)

Метео-прогноз /
предупреждения

Пропустить шторм хуже, чем
предупредить зря

Precision важнее

Сфера

Почему

FP → показываем
пользователю неинтересный
контент

Поисковые
системы

FP → мусор и нерелевантные
результаты в выдаче

Автоматическая
модерация
контента

FP → удаление легитимных
постов, репутационные риски

Юридические
системы

FP → юристы тратят время на
проверку корректных
документов

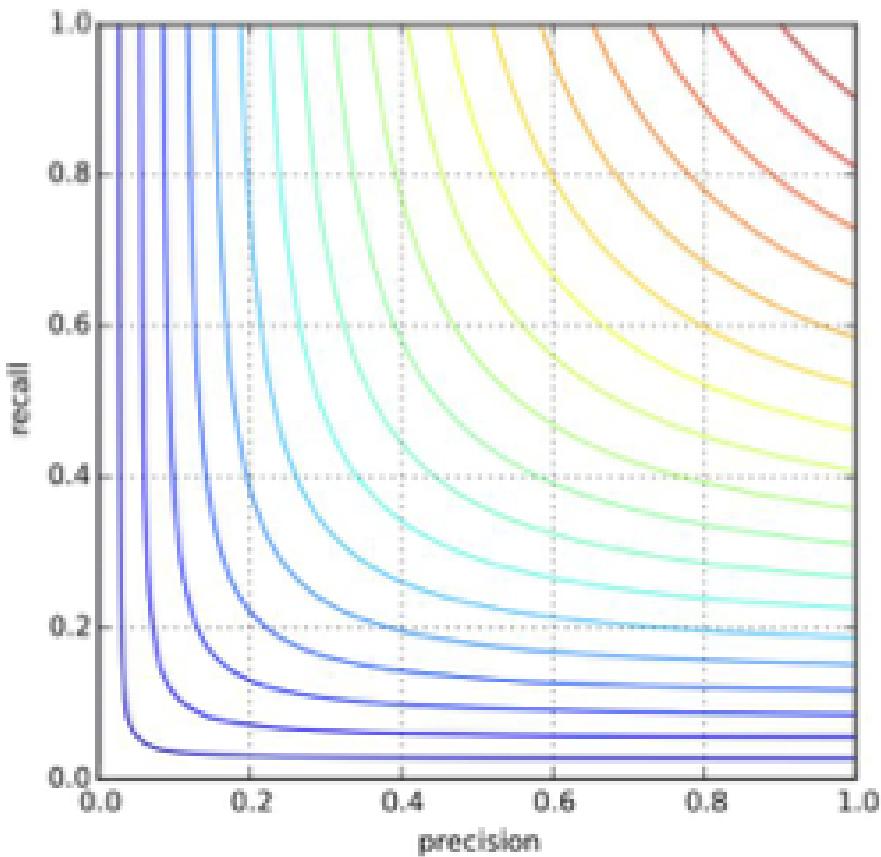
Precision и Recall

Иногда важно, чтобы и точность, и полнота были побольше, при этом нет отдельных требований к точности и полноте. Тогда, конечно, можно использовать accuracy.

Но как считать качество, если данные **несбалансираны**?

F1-Score

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$



F1-score – это гармоническое среднее, а не арифметическое.
Гармоническое среднее "штрафует" дисбаланс, если одна из метрик низкая, то F1 резко падает.

Precision	Recall	F1
1.0	0.0	0.00
0.9	0.9	0.90
0.9	0.5	0.64
0.5	0.9	0.64

[Про то почему в f1 именно гармоническое среднее](#)

ROC-AUC

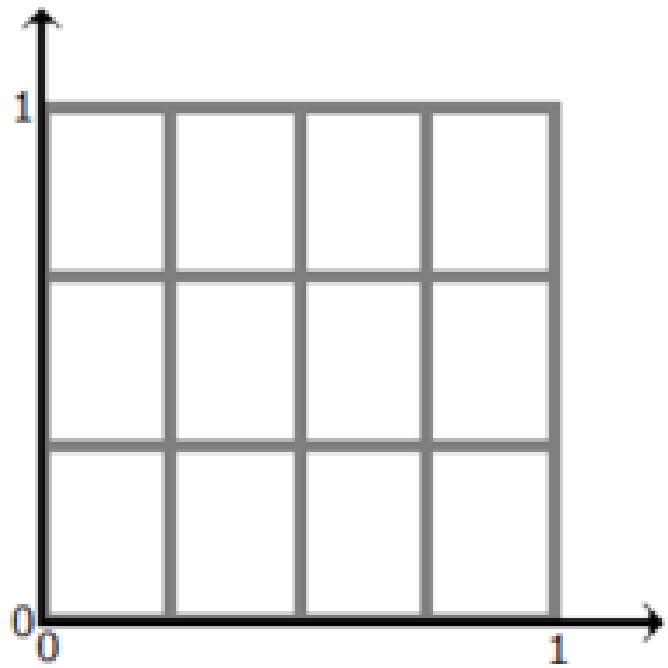
Интегральные метрики

r	класс
0.5	0
0.1	0
0.25	0
0.6	1
0.2	1
0.3	1
0.0	0



r	класс
0.6	1
0.5	0
0.3	1
0.25	0
0.2	1
0.1	0
0.0	0

ROC-AUC

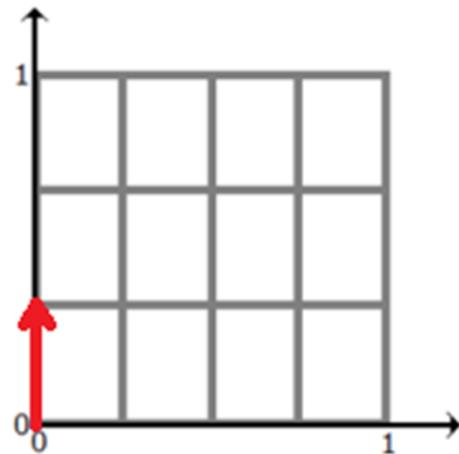


ROC-AUC

Алгоритм:

- идем сверху вниз по отсортированной таблице;
- если встречаем 1, то делаем шагок на 1 вверх;
- если встречаем 0, то делаем шагок на 1 вправо.

p	класс
0.6	1
0.5	0
0.3	1
0.25	0
0.2	1
0.1	0
0.0	0

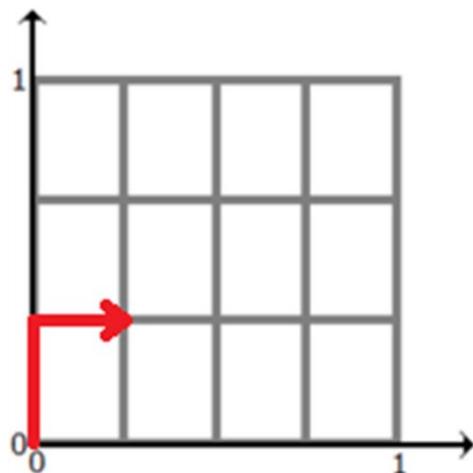


ROC-AUC

Алгоритм:

- идем сверху вниз по отсортированной таблице;
- если встречаем 1, то делаем шагок на 1 вверх;
- если встречаем 0, то делаем шагок на 1 вправо.

p	класс
0.6	1
0.5	0
0.3	1
0.25	0
0.2	1
0.1	0
0.0	0

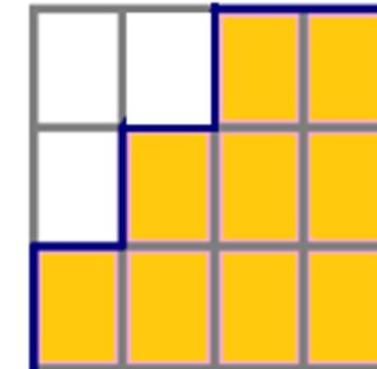
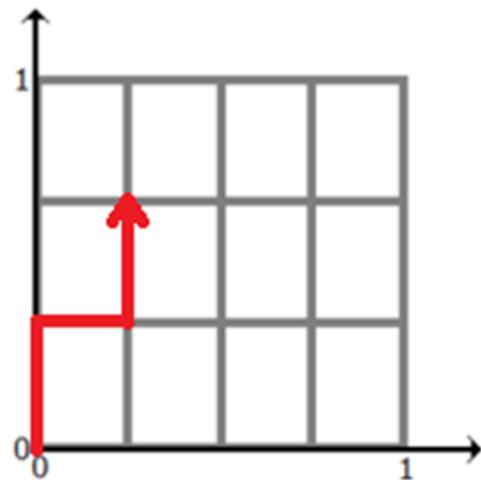


ROC-AUC

Алгоритм:

- идем сверху вниз по отсортированной таблице;
- если встречаем 1, то делаем шагок на 1 вверх;
- если встречаем 0, то делаем шагок на 1 вправо.

p	класс
0.6	1
0.5	0
0.3	1
0.25	0
0.2	1
0.1	0
0.0	0



ROC-AUC

Свойства **ROC-AUC**:

- ROC-AUC принимает значение от 0 до 1.
- ROC-AUC идеальной модели равен 1 (так как в таблице после сортировки стоят сначала все 1, а потом все 0, и мы идем по квадрату до конца вверх, а потом до конца вправо - получаем, что ROC-кривая идет по сторонам квадрата).
- ROC-AUC модели, которая случайным образом предсказывает классы, равен 0.5.
- Чем больше ROC-AUC, тем лучше наш алгоритм сортирует объекты по вероятностям - то есть тем лучше алгоритм решает задачу классификации.

ROC-AUC

Формальный алгоритм построения ROC-кривой

False Positive Rate (FPR) - доля неверно принятых объектов отрицательного класса

True Positive Rate (TPR) - доля верно принятых объектов положительного класса

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

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

ROC-AUC

Формальный алгоритм построения ROC-кривой

- Рисуем плоскость, ось x называем FPR, ось у - TPR.
- Для каждого возможного порога перевода вероятностей в классы (на практике достаточно рассматривать только пороги, равные вероятностям из таблицы, так как промежуточные пороги не изменяют TPR и FPR) вычисляем TPR и FPR. Отмечаем точку с координатами (FPR, TPR) на плоскости.
- Соединяя полученные точки кривой - получаем ROC-кривую