

# Занятие 10. Метрики и валидация

---

Гирдюк Дмитрий Викторович

15 ноября 2025

СПбГУ, ПМ-ПУ, ДФС

# Задача обучения с учителем

- Постановка задачи обучения с учителем (supervised learning): необходимо предсказать значение целевой переменной  $y \in Y$  объекта по набору его признаков  $x \in X$
- Среда описывается совместным распределением  $f_{X,Y}(x, y)$ , а выборкой из нее является набор пар  $\mathcal{D} = \{(x^{(i)}, y^{(i)})\}_{i=1}^N$
- Результирующая модель возвращает значение  $y$  по признакам  $x$ :  $y = h(x; \theta)$
- Классификация:
  - $Y = \{0, 1\}$  – бинарная (binary)
  - $Y = \{1, 2, \dots, K\}$  – многоклассовая (multiclass)
  - $Y = \{0, 1\}^K$  – многозначная (multi-label)
- Регрессия:
  - $Y = R$  – одномерная (ordinal)
  - $Y = R^K$  – многомерная (multiple)

# Оценка качества моделей

- Как понять, что наша модель работает хорошо?
- У нас есть функция потерь/целевая функция, которую мы получаем при сведении задачи построения модели к задаче математической оптимизации
- Есть и метрики, которые позволяют адекватно оценить результаты работы модели на основе предсказанных меток и известных истинных значений
- В задачах регрессии многие метрики могут выступать в роли функций потерь. Но чаще всего метрики  $\neq$  функции потерь

# Метрики в задаче регрессии: MSE, RMSE, $R^2$

- RSS (Residual Sum of Squares)

$$RSS = \sum_{i=1}^N \left( y^{(i)} - h(\mathbf{x}^{(i)}) \right)^2$$

- MSE и RMSE (Root Mean Squared Error)

$$MSE = \frac{1}{N} RSS = \frac{1}{N} \sum_{i=1}^N \left( y^{(i)} - h(\mathbf{x}^{(i)}) \right)^2, \quad RMSE = \sqrt{MSE}$$

- Коэффициент детерминации  $R^2$

$$R^2 = 1 - \frac{\sum_{i=1}^N \left( y^{(i)} - h(\mathbf{x}^{(i)}) \right)^2}{\sum_{i=1}^N \left( y^{(i)} - \bar{y} \right)^2} = 1 - \frac{RSS}{TSS}$$

где  $\bar{y} = \frac{1}{N} \sum_{i=1}^N y^{(i)}$ .

# Метрики в задаче регрессии: MAE, MAPE, SMAPE

- MAE (Mean Average Error)

$$MAE = \frac{1}{N} \sum_{i=1}^N |y^{(i)} - h(\mathbf{x}^{(i)})|$$

- Относительные метрики: MAPE и SMAPE (Symmetric Mean Absolute Percentage Error)

$$MAPE = \frac{1}{N} \sum_{i=1}^N \frac{|y^{(i)} - h(\mathbf{x}^{(i)})|}{|y^{(i)}|}$$

$$SMAPE = \frac{1}{N} \sum_{i=1}^N \frac{2 |y^{(i)} - h(\mathbf{x}^{(i)})|}{|y^{(i)} + h(\mathbf{x}^{(i)})|}$$

SMAPE решает вопрос с нулем в знаменателе MAPE

# Метрики в задаче регрессии: замечания

- Все метрики выше естественным образом допускают обобщение путем добавления весовых коэффициентов для наблюдений
- MSE, RMSE, MAE могут быть использованы в качестве целевой функции.
- MAPE по сути есть MAE, в котором у наблюдений есть весовые коэффициенты  $\frac{1}{|y^{(i)}|}$

# Метрики в задаче бинарной классификации

- Метрики в задаче классификации весьма разнообразны. И чаще всего их нельзя использовать в качестве целевой функции
- Здесь и далее предполагаем, что у нас есть модель  $h(x; \theta)$ , которая возвращает метки предсказанных классов  $\{0, 1, \dots, K\}$
- Начнем с задачи бинарной классификации ( $K = 1$ ), анализ результатов которой проводят на основе матрицы ошибок (confusion matrix)

		Predicted condition	
		Positive (PP)	Negative (PN)
Actual condition	Total population = P + N		
	Positive (P)	True positive (TP)	False negative (FN)
	Negative (N)	False positive (FP)	True negative (TN)

# Accuracy

- Accuracy – доля правильно предсказанных объектов

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

- Что будет с метрикой при дисбалансе классов?
- Разная значимость FP и FN (медицинская диагностика)

		Predicted condition	
Total population = P + N		Positive (PP)	Negative (PN)
Actual condition	Positive (P)	True positive (TP)	False negative (FN)
	Negative (N)	False positive (FP)	True negative (TN)



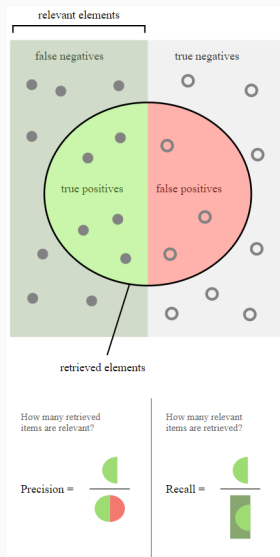
# Precision и Recall

- Точность (Precision) – доля правильно предсказанных положительных объектов среди всех отнесенных к положительным

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

- Полнота (Recall) – доля правильно предсказанных положительных объектов среди всех положительных объектов

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



## $F_\beta$ -меры

- Работать сразу с двумя метриками не очень удобно, потому precision и recall объединяют с помощью  $F_1$ -меры

$$F_1 = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}} = 2 \frac{Precision \times Recall}{Precision + Recall}$$

- Или  $F_\beta$ , если precision и recall не должны иметь одинаковую степень важности

$$F_\beta = (1 + \beta^2) \frac{Precision \times Recall}{Recall + \beta^2 Precision}$$

		Predicted condition	
		Positive (PP)	Negative (PN)
Actual condition	Total population = P + N		
	Positive (P)	True positive (TP)	False negative (FN)
	Negative (N)	False positive (FP)	True negative (TN)

# Пороги в задаче классификации

- Вернемся ненадолго к модели  $h(\mathbf{x}; \boldsymbol{\theta})$ . Как именно, например, в логистической регрессии мы получаем метки классов?
- Мы сравниваем полученные оценки вероятностей с некоторым порогом  $p_{\text{thd}}$ , по умолчанию с 0.5
- Вообще говоря, никто не мешает нам в каждой задаче задавать собственный порог, т.е. порог может выступать в роли гиперпараметра модели
- Понятно, что выбор порога имеет достаточно важное значение, ведь от него зависят итоговые метки, а от них — значения интересующих нас метрик

- Рассмотрим еще 2 метрики качества бинарной классификации
- True Positive Rate, TPR, синоним полноты (Recall), рассмотренной ранее

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

- False positive Rate, FPR

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

- Увеличивая порог  $p_{\text{thd}}$ , мы относим к положительным меньше наблюдений, но и количество неправильно классифицированных положительных объектов уменьшается
- Кривая, у которой по оси ординат TPR, а по оси абсцисс FPR, называется ROC-кривой (Receiver Operating Characteristic)

# Пример ROC-кривой

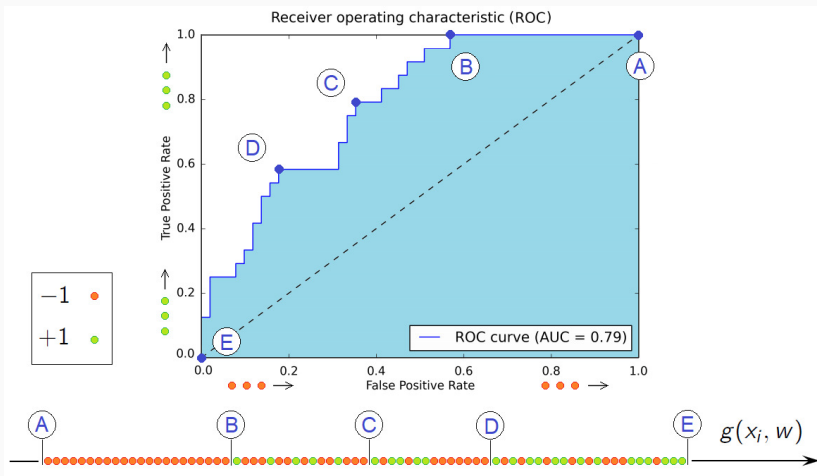


Рис. 1: ROC-кривая [1]

# Построение ROC-кривой [2]

id	оценка	класс
1	0.5	0
2	0.1	0
3	0.2	0
4	0.6	1
5	0.2	1
6	0.3	1
7	0.0	0

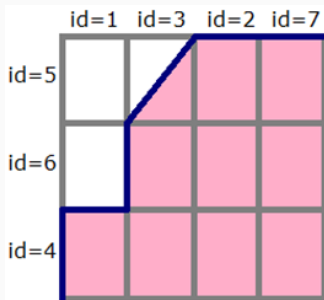
Табл. 1

id	оценка	класс
4	0.6	1
1	0.5	0
6	0.3	1
3	0.2	0
5	0.2	1
2	0.1	0
7	0.0	0

Табл. 2

id	> 0.25	класс
4	1	1
1	1	0
6	1	1
3	0	0
5	0	1
2	0	0
7	0	0

Табл. 3



- Понятно, что чем больше площадь (Area Under the Curve) под ROC-кривой, тем лучше классификатор справляется со своей задачей
- Из способа построения ROC-кривой и вычисления ее площади следует идея того, когда именно стоит применять ROCAUC – в задачах, где важнее не то, как хорошо выполняется предсказание на отдельных объектах, а то, где важнее правильный порядок предсказания

# Многоклассовая классификация

- Кратко о том, что делать, если классов больше 2
- Многие метрики бинарной классификации обобщаются на многоклассовый случай, но есть детали
- Для каждого класса можно посчитать свою собственную матрицу ошибок, а затем есть два варианта: мы либо усредняем все значения в матрице (микроусреднение), либо усредняем значения метрик по всем классам (макроусреднение)
- Первое слабо чувствительно к классам с небольшим числом наблюдений, второе наоборот



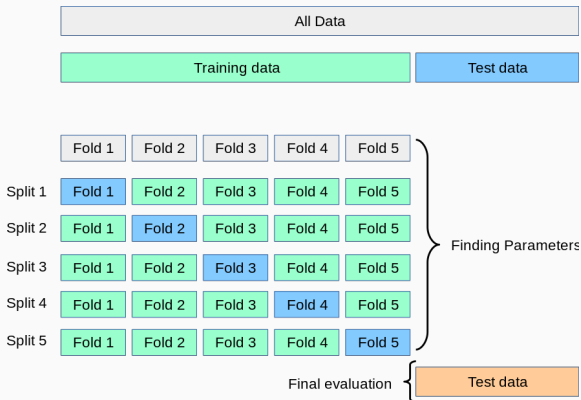
# Способы оценки качества моделей i

- Мы поговорили о метриках, используемых для оценки качества моделей в задачах обучения с учителем
- Теперь рассмотрим вопрос того, какие существуют процедуры для проведения этой оценки
- Замечание: оставим за скобками всевозможные "внутренние критерии" – оценки качества на данных, напрямую используемых для обучения. Сюда входят такие вещи (с ними, возможно, познакомят на практике по статистике) как информационный критерий Акаике (AIC), байесовский информационный критерий (BIC) и др
- С одной из процедур мы уже знакомы: разделение обучающей выборки на **тренировочную, валидационную и тестовую** (метод hold-out)

- Тренировочный датасет используем для обучения модели. Валидационный – для подбора гиперпараметров модели или оптимизационного алгоритма (или даже гиперпараметров процесса трансформации данных). А на тестовом датасете производим окончательную оценку обобщающей способности нашей модели
- Но есть проблема, при таком подходе приличная часть данных откладывается. Альтернативы?

# Кросс-валидация

- Кросс-валидация (перекрестная проверка, cross-validation) – общий способ оценки обобщающей способности модели, состоящий в ресемплировании данных и использовании их как для тренировки модели, так и для проверки



# Примеры кросс-валидации

- Скользящий контроль (Leave-one-out) – тренируем модель на всех данных кроме одного наблюдения, на котором проводим проверку. Усредняем значения метрики для каждого объекта. Трудоемко, высокая дисперсия
- Кросс-валидация по  $k$  блокам ( $k$ -fold cross-validation) – разбиваем данные на  $k$  частей, последовательно используем  $k - 1$  часть для обучения, оставшуюся для проверки, итоговые значения усредняем. Менее трудоемко чем скользящий контроль, но существенно зависим от разбиения
- Кросс-валидация по  $k$  блокам  $j$  раз ( $t \times k$ -fold cross-validation). Аналогично предыдущему, но процедуру производим  $t$  раз. Компромисс между трудоемкостью и точностью
- В scikit-learn реализовано достаточно много разнообразных схем произведения разбиения (например, с учетом стратификации по группирующему признаку)

# Дилемма смещения–дисперсии

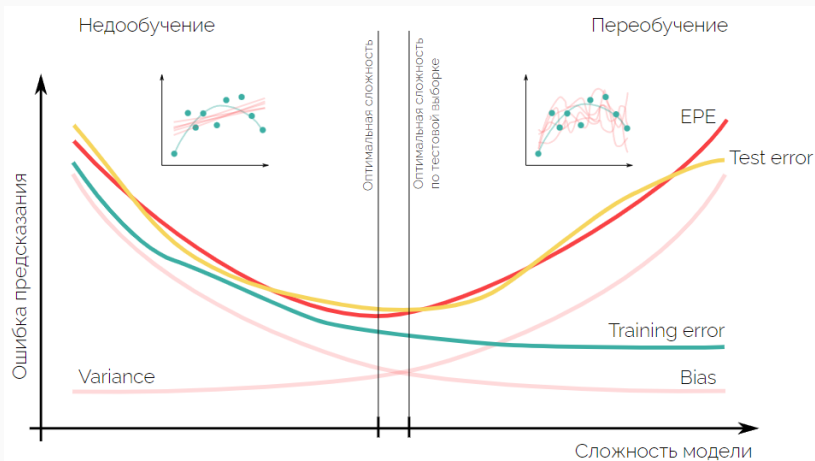


Иллюстрация дилеммы смещения-дисперсии.

1. *Воронцов К.* Презентация по методам оценки качества из курса лекций Воронцова К.В. URL:  
<http://www.machinelearning.ru/wiki/images/a/a2/08-Voron-ML-Quality-slides.pdf>.
2. *Дьяконов А.* AUC ROC (площадь под кривой ошибок). URL:  
<https://alexanderdyakonov.wordpress.com/2017/07/28/auc-roc-%D0%BF%D0%BB%D0%BE%D1%89%D0%B0%D0%B4%D1%8C-%D0%BF%D0%BE%D0%B4-%D0%BA%D1%80%D0%B8%D0%B2%D0%BE%D0%B9-%D0%BE%D1%88%D0%B8%D0%B1%D0%BE%D0%BA/>.