

# Лекция 3

# Линейные методы

# регрессии. Часть 2.

Юлия Конюшенко

ТГ: @ko\_iulia

[koniushenko.iun@phystech.edu](mailto:koniushenko.iun@phystech.edu)

- линейная регрессия

$$\alpha(x) = w_0 + w_1 x_1 + w_2 x_2$$

$$\alpha(x) = w_0 + \sum_{j=1}^n w_j x_j = w_0 \cdot 1 + \sum_{j=1}^n w_j x_j = \sum_{j=0}^n w_j x_j = (\omega, x)$$

обучение  $\equiv$  миним. MSE

погреш. MSE ?  $\equiv$  вероятн. интерпр

Град спуск

$$w^{(k)} = w^{(k-1)} - \eta \nabla Q(w^{(k-1)})$$

Стохастический  $\rightarrow$  по 1 объекту

Mini-batch gradient descent  $\rightarrow$  по батчу

# ПЛАН ЛЕКЦИИ

- метрики качества и функционалы ошибки в задаче регрессии; бизнес-метрики
- признаки переобученной модели и методы выявления переобучения и борьбы с ним: кросс-валидация и регуляризация, полезное свойство L1-регуляризации

# 1. МЕТРИКИ КАЧЕСТВА И ФУНКЦИОНАЛЫ ОШИБКИ В ЗАДАЧАХ РЕГРЕССИИ

# МЕТРИКИ КАЧЕСТВА И ФУНКЦИИ ОШИБКИ

- **Функционал (функция) ошибки** – функция, которую минимизируют в процессе обучения модели для нахождения неизвестных параметров (весов).
- **Метрика качества** – функция, которую используют для оценки качества построенной (уже обученной) модели.

# МЕТРИКИ КАЧЕСТВА И ФУНКЦИИ ОШИБКИ

- **Функционал (функция) ошибки** – функция, которую минимизируют в процессе обучения модели для нахождения неизвестных параметров (весов).
- **Метрика качества** – функция, которую используют для оценки качества построенной (уже обученной) модели.

*Иногда одна и та же функция может использоваться и для обучения модели (функция ошибки), и для оценки качества модели (метрика качества).*

# ЛИНЕЙНАЯ РЕГРЕССИЯ

Линейная регрессия:

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

Обучение линейной регрессии - минимизация  
среднеквадратичной ошибки:

$$\frac{1}{l} \sum_{i=1}^l (a(x_i) - y_i)^2 \rightarrow \min_w$$

# СРЕДНЕКВАДРАТИЧНОЕ ОТКЛОНЕНИЕ: MSE (MEAN SQUARED ERROR)

Среднеквадратичное отклонение:

$$MSE(a, X) = \frac{1}{l} \sum_{i=1}^l (a(x_i) - y_i)^2$$

- $\beta$  чем MSE хороша?
- $\beta$  чем MSE плоха?

# СРЕДНЕКВАДРАТИЧНОЕ ОТКЛОНЕНИЕ: MSE (MEAN SQUARED ERROR)

Среднеквадратичное отклонение:

$$MSE(a, X) = \frac{1}{l} \sum_{i=1}^l (a(x_i) - y_i)^2$$

Плюсы:

- Позволяет сравнивать модели
- Подходит для контроля качества во время обучения

# СРЕДНЕКВАДРАТИЧНОЕ ОТКЛОНение: MSE

Среднеквадратичное отклонение:

$$MSE(a, X) = \frac{1}{l} \sum_{i=1}^l (a(x_i) - y_i)^2$$

Плюсы:

- Позволяет сравнивать модели
- Подходит для контроля качества во время обучения

Минусы:

- Плохо интерпретируется, т.к. не сохраняет единицы измерения (если целевая переменная – кг, то MSE измеряется в кг в квадрате) - как исправить?
- Тяжело понять, насколько хорошо данная модель решает задачу, так как MSE не ограничена сверху. 500 - это много, а 5? а 10000? как поправить?

# RMSE (ROOT MEAN SQUARED ERROR)

Корень из среднеквадратичной ошибки:

$$RMSE(a, X) = \sqrt{\frac{1}{l} \sum_{i=1}^l (a(x_i) - y_i)^2}$$

Плюсы:

- Все плюсы MSE
- Сохраняет единицы измерения (в отличие от MSE)

Минусы:

- Тяжело понять, насколько хорошо данная модель решает задачу, так как RMSE не ограничена сверху.

## КОЭФИЦИЕНТ ДЕТЕРМИНАЦИИ ( $R^2$ )

Коэффициент детерминации:

$$R^2(a, X) = 1 - \frac{\sum_{i=1}^l (a(x_i) - y_i)^2}{\sum_{i=1}^l (y_i - \bar{y})^2},$$

где  $\bar{y} = \frac{1}{l} \sum_{i=1}^l y_i$ .

что означает?

- $R^2$  чем больше, тем лучше!
- Какой максимум?
- Какой минимум?

# КОЭФФИЦИЕНТ ДЕТЕРМИНАЦИИ ( $R^2$ )

Коэффициент детерминации:

$$R^2(a, X) = 1 - \frac{\sum_{i=1}^l (a(x_i) - y_i)^2}{\sum_{i=1}^l (y_i - \bar{y})^2},$$

где  $\bar{y} = \frac{1}{l} \sum_{i=1}^l y_i$ .

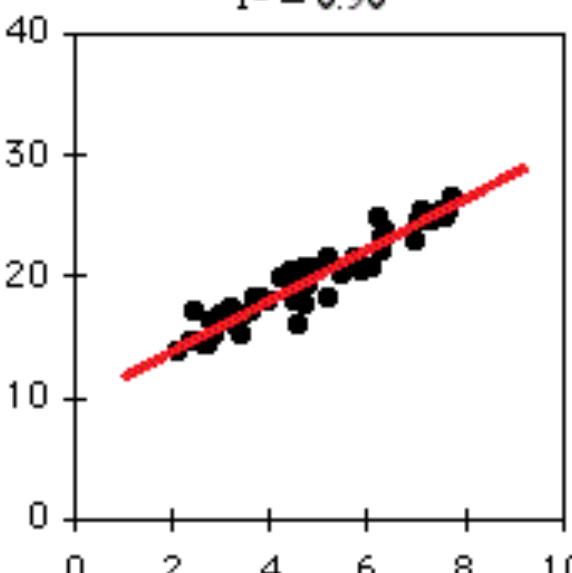
Коэффициент детерминации это доля дисперсии целевой переменной, объясняемая моделью.

- Чем ближе  $R^2$  к 1, тем лучше модель объясняет данные
- Чем ближе  $R^2$  к 0, тем ближе модель к константному предсказанию
- Отрицательный  $R^2$  говорит о том, что модель плохо решает задачу

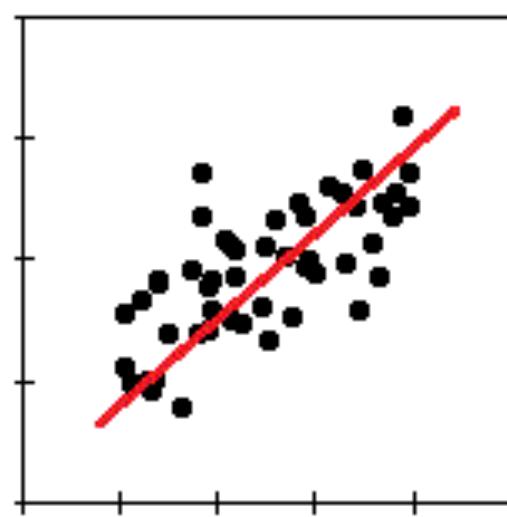
# КОЭФФИЦИЕНТ ДЕТЕРМИНАЦИИ ( $R^2$ )

$$R^2 \leq 1$$

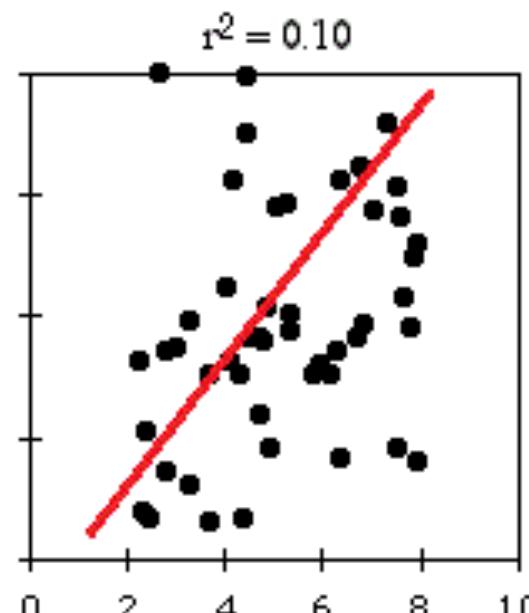
$r^2 = 0.90$



$r^2 = 0.50$



$r^2 = 0.10$



# MAE (MEAN ABSOLUTE ERROR)

Средняя абсолютная ошибка:

$$MAE(a, X) = \frac{1}{l} \sum_{i=1}^l |a(x_i) - y_i|$$

# MAE (MEAN ABSOLUTE ERROR)

Средняя абсолютная ошибка:

$$MAE(a, X) = \frac{1}{l} \sum_{i=1}^l |a(x_i) - y_i|$$

Плюсы:

- Менее чувствителен к выбросам, чем MSE *- потому?*

*Всегда ли нужно удалять выбросы?*

$y$	$a_1(x)$	$a_2(x)$	$a_3(x)$
$y_1$	2	1	2
2	3	3	3
3	4	4	4
100	5	5	6

# MAE (MEAN ABSOLUTE ERROR)

Средняя абсолютная ошибка:

$$MAE(a, X) = \frac{1}{l} \sum_{i=1}^l |a(x_i) - y_i|$$

Плюсы:

- Менее чувствителен к выбросам, чем MSE

Минусы:

- MAE - не дифференцируемый функционал

# MSLE (MEAN SQUARED LOGARITHMIC ERROR)

Среднеквадратичная логарифмическая ошибка:

$$MSLE(a, X) = \frac{1}{l} \sum_{i=1}^l (\log(a(x_i) + 1) - \log(y + 1))^2$$

- Подходит для задач с неотрицательной целевой переменной ( $y \geq 0$ )
- Штрафует за отклонения в порядке величин
- Штрафует заниженные прогнозы сильнее, чем завышенные

# MAPE

*MAPE – Mean Absolute Percentage Error:*

$$MAPE(a, X) = \frac{1}{l} \sum_{i=1}^l \frac{|y_i - a(x_i)|}{|y_i|}$$

MAPE измеряет относительную ошибку.

# MAPE

$$MAPE(a, X) = \frac{1}{l} \sum_{i=1}^l \frac{|y_i - a(x_i)|}{|y_i|}$$

Плюсы:

- Ограничена:  $0 \leq MAPE \leq 1$
- Хорошо интерпретируема: например, MAPE=0.16 означает, что ошибка модели в среднем составляет 16% от фактических значений.

Какие есть недостатки?

# MAPE

$$MAPE(a, X) = \frac{1}{l} \sum_{i=1}^l \frac{|y_i - a(x_i)|}{|y_i|}$$

Плюсы:

- Ограничена:  $0 \leq MAPE \leq 1$
- Хорошо интерпретируема: например,  $MAPE=0.16$  означает, что ошибка модели в среднем составляет 16% от фактических значений.

Минусы:

- По-разному относится к недо- и перепрогнозу. Например, если правильный ответ  $y = 10$ , а прогноз  $a(x) = 20$ , то ошибка  $\frac{|10-20|}{|10|} = 1$ , а если ответ  $y = 30$ , то ошибка  $\frac{|30-20|}{|30|} = \frac{1}{3} \approx 0.33$ .

Как исправить?

# SMAPE

SMAPE – *Symmetric Mean Absolute Percentage Error*  
(симметричный вариант MAPE):

$$SMAPE(a, X) = \frac{1}{l} \sum_{i=1}^l \frac{|y_i - a(x_i)|}{(|y_i| + |a(x_i)|)/2}$$

SMAPE – попытка сделать симметричным прогноз (то есть дать одинаковую ошибку для недо- и перепрогноза).

# SMAPE

SMAPE – *Symmetric Mean Absolute Percentage Error*  
(симметричный вариант MAPE):

$$SMAPE(a, X) = \frac{1}{l} \sum_{i=1}^l \frac{|y_i - a(x_i)|}{(|y_i| + |a(x_i)|)/2}$$

SMAPE – попытка сделать симметричным прогноз (то есть дать одинаковую ошибку для недо- и перепрогноза).

Проверим:

Пусть правильный ответ  $y = 10$ , а прогноз  $a(x) = 20$ , то

ошибка  $\frac{|10-20|}{|10+20|/2} = \frac{2}{3} \approx 0.67$ , а если ответ  $y = 30$ , то ошибка

$$\frac{|30-20|}{|30+20|/2} = \frac{2}{5} = 0.4.$$

## SMAPE

SMAPE – попытка сделать симметричным прогноз (то есть дать одинаковую ошибку для недо- и перепрогноза).

Проверим:

Пусть правильный ответ  $y = 10$ , а прогноз  $a(x) = 20$ , то

ошибка  $\frac{|10-20|}{|10+20|/2} = \frac{2}{3} \approx 0.67$ , а если ответ  $y = 30$ , то ошибка

$\frac{|30-20|}{|30+20|/2} = \frac{2}{5} = 0.4$ .

*Ошибки стали меньше отличаться друг от друга, но всё-таки не равны.*

# SMAPE

SMAPE – попытка сделать симметричным прогноз (то есть дать одинаковую ошибку для недо- и перепрогноза).

*“Сейчас уже в среде прогнозистов сложилось более-менее устойчивое понимание, что SMAPE не является хорошей ошибкой. Тут дело не только в завышении прогнозов, но и в том, что наличие прогноза в знаменателе позволяет манипулировать результатами оценки.” (см. [источник](#))*

# ОПТИМУМЫ MSE И MAE

Рассмотрим вероятностную постановку задачи.

Предположим, что на объектах с одинаковым признаковым описанием могут быть разные ответы. В этом случае на всех таких объектах MSE (или MAE) должна выдать один и тот же ответ.

**Теорема.** Пусть даны  $l$  объектов с одинаковым признаковым описанием и значениями целевой переменной  $y_1, \dots, y_l$ .

Тогда:

1. Оптимум MSE достигается на среднем значении ответов:

$$\alpha_{MSE} = \sum_{i=1}^l y_i$$

2. Оптимум MAE достигается на медиане ответов:

$$\alpha_{MAE} = median\{y_1, \dots, y_l\}$$

$$MSE: \quad a(x) = a^*$$

$$MSE = \frac{1}{l} \sum_{i=1}^l (y_i - a(x_i))^2$$

$$MSE = \frac{1}{l} \sum (y_i - a)^2 \rightarrow \min$$

$$\frac{\partial MSE}{\partial a} = -\frac{2}{l} \sum (y_i - a) = -\frac{2}{l} \sum y_i + \frac{2}{l} \sum a = 0$$

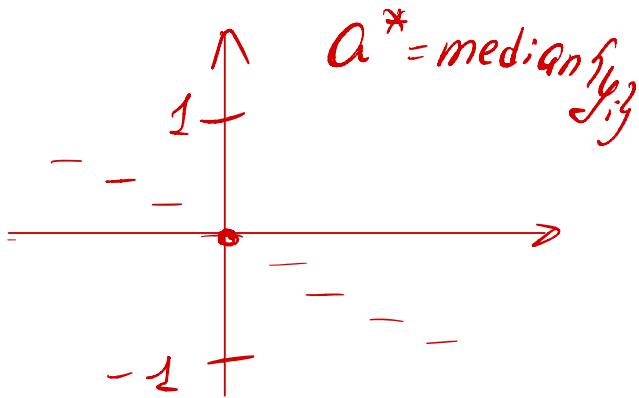
$$-\frac{2}{l} \sum y_i + \frac{2}{l} \cdot l a = 0 \quad \Leftrightarrow \boxed{a^* = \frac{1}{l} \sum y_i}$$

MAE:

$$|x| = \begin{cases} x, & x \geq 0 \\ -x, & x < 0 \\ 0, & 0 \end{cases}$$

$$\frac{\partial MAE}{\partial a} = \frac{1}{l} \sum_{i=1}^l \text{sign}(y_i - a) = 0$$

$$a < y_1 \leq y_2 \leq \dots \leq y_l$$



# КВАНТИЛЬНАЯ РЕГРЕССИЯ

Квантильная функция потерь:

$$Q(a, X^\ell) = \sum_{i=1}^{\ell} \rho_\tau(y_i - a(x_i))$$

если заранее знаем,  
что за что-то  
хотим штрафовать  
сильнее

Здесь

$$\rho_\tau(z) = (\tau - 1)[z < 0]z + \tau[z \geq 0]z = (\tau - \frac{1}{2})z + \frac{1}{2}|z|$$

- что будет, если подставить  $1/2$ ?
- на каком значении достигается минимум?

Параметр  $\tau \in [0; 1]$ .

- Чем больше  $\tau$ , тем больше штрафуем за занижение прогноза.

# ВЕРОЯТНОСТНЫЙ СМЫСЛ КВАНТИЛЬНОЙ ФУНКЦИИ ПОТЕРЬ

## Теорема.

Пусть в каждой точке  $x \in X$  (пространство объектов) задано распределение  $p(y|x)$  на ответах для данного объекта.

Тогда оптимизация функции потерь  $\rho_\tau(z)$  дает алгоритм  $a(x)$ , приближающий  $\tau$ -квантиль распределения ответов в каждой точке  $x \in X$ .

# МЕТРИКИ: ОНЛАЙН, ОФЛАЙН, БИЗНЕС

Бизнес

Показатели бизнеса

Например:

- Lifetime value
- Прибыль
- Расходы
- Доля аудитории
- Цена акций

Мы хотели бы смотреть, как модель влияет на них, но не можем

Измеряются месяцами

Метрики

Онлайн

Оффлайн

Связаны с показателями бизнеса  
Можно сделать быстрый тест

Например:

- Конверсия в клик
- Оценка сервиса
- Средний чек
- MAU, DAU, WAU

Мы можем оценить эти метрики, проведя A/B-тест

Измеряются неделями

Являются приближением  
онлайн-метрик

Считываются на исторических  
данных

Например:

- Precision, recall
- Accuracy

Считываются минуты-часы

Можем почти бесплатно  
проверить наши модели

# СВОЙСТВА МЕТРИК

- Чувствительность
- Шум
- Интерпретация
- Иерархия

# ПРИМЕР ИЕРАРХИИ МЕТРИК

- Хотим внедрить новое ML-ранжирование рекомендаций товаров
- Находимся в ситуации, когда этот элемент уже есть на сайте

Ваша подборка для покупок у нас



Что измеряем?

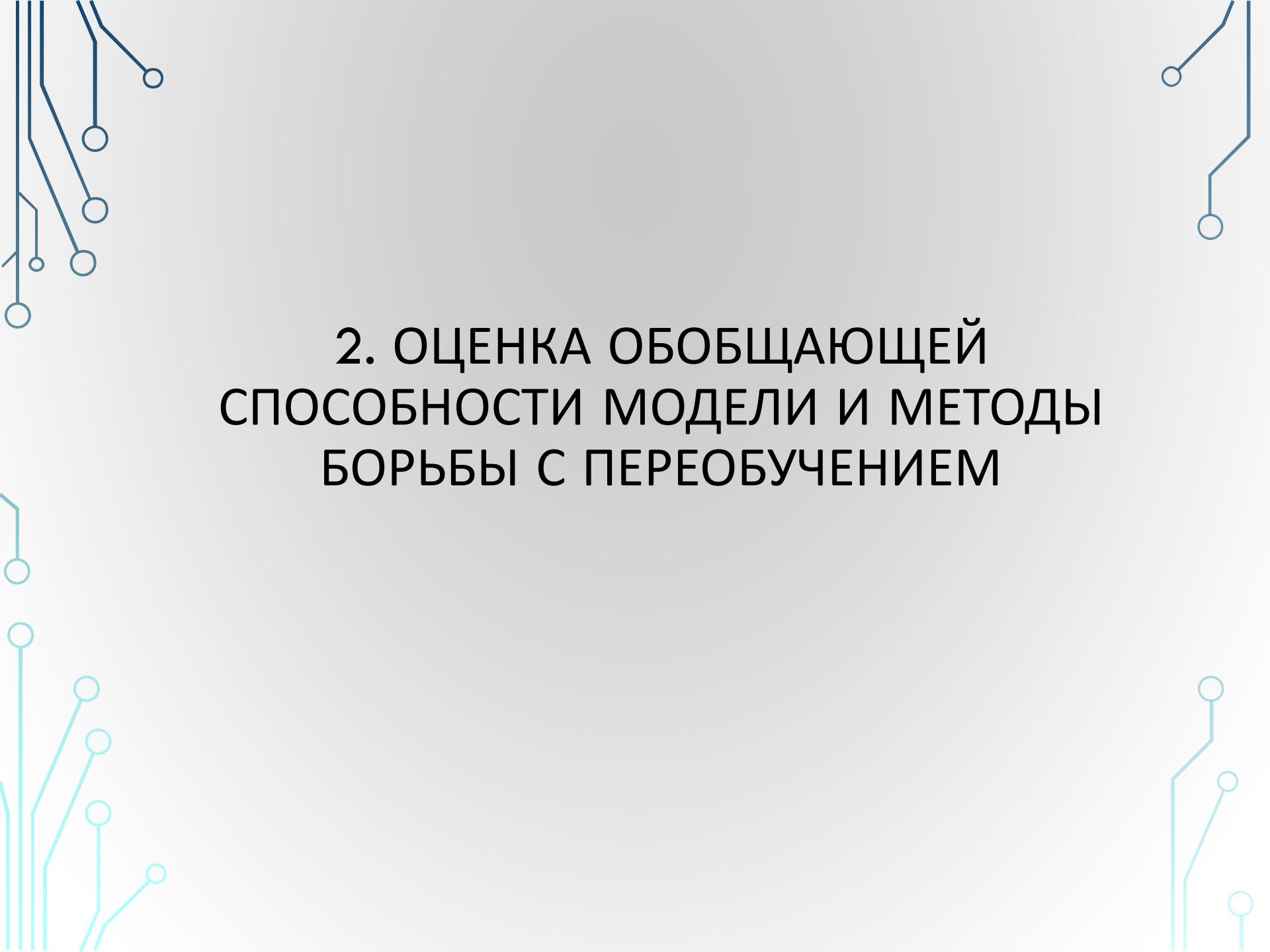
# ИЕРАРХИЯ МЕТРИК

Бизнес-  
метрика

Онлайн-  
метрики

Оффлайн-  
метрики

- Выручка
- Средний чек / Число купивших пользователей
- Выручка проданных товаров, через наш элемент
- CTR элемента
- Оффлайн метрики ранжирования
- Accuracy на валидационной выборке

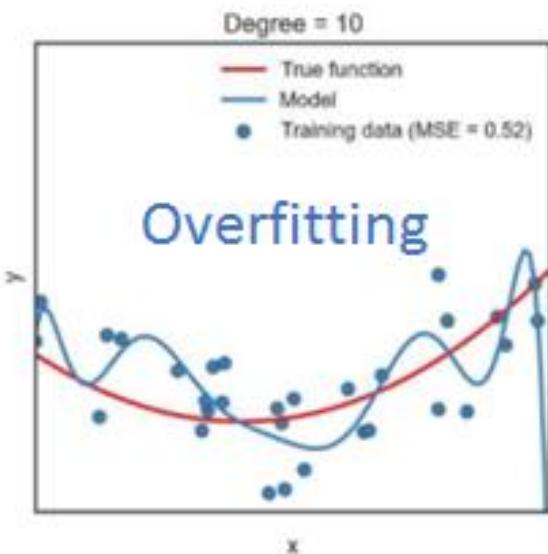
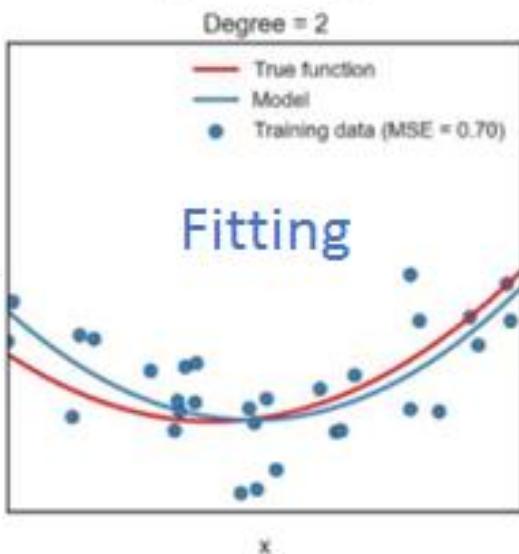
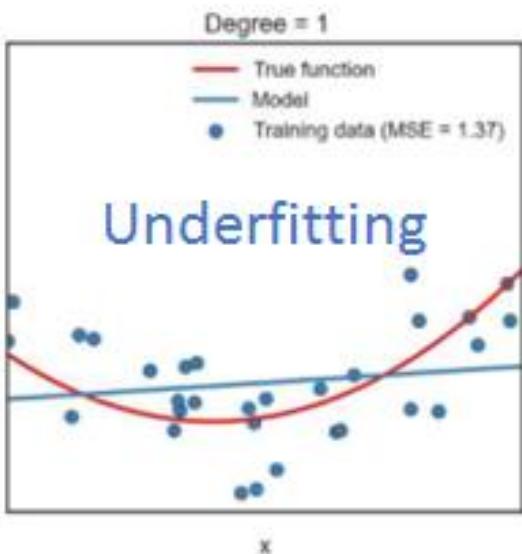


## 2. ОЦЕНКА ОБОБЩАЮЩЕЙ СПОСОБНОСТИ МОДЕЛИ И МЕТОДЫ БОРЬБЫ С ПЕРЕОБУЧЕНИЕМ

# ОЦЕНКА ОБОБЩАЮЩЕЙ СПОСОБНОСТИ МОДЕЛИ

**Переобучение (overfitting)** – явление, при котором качество модели на новых данных сильно хуже, чем качество на тренировочных данных.

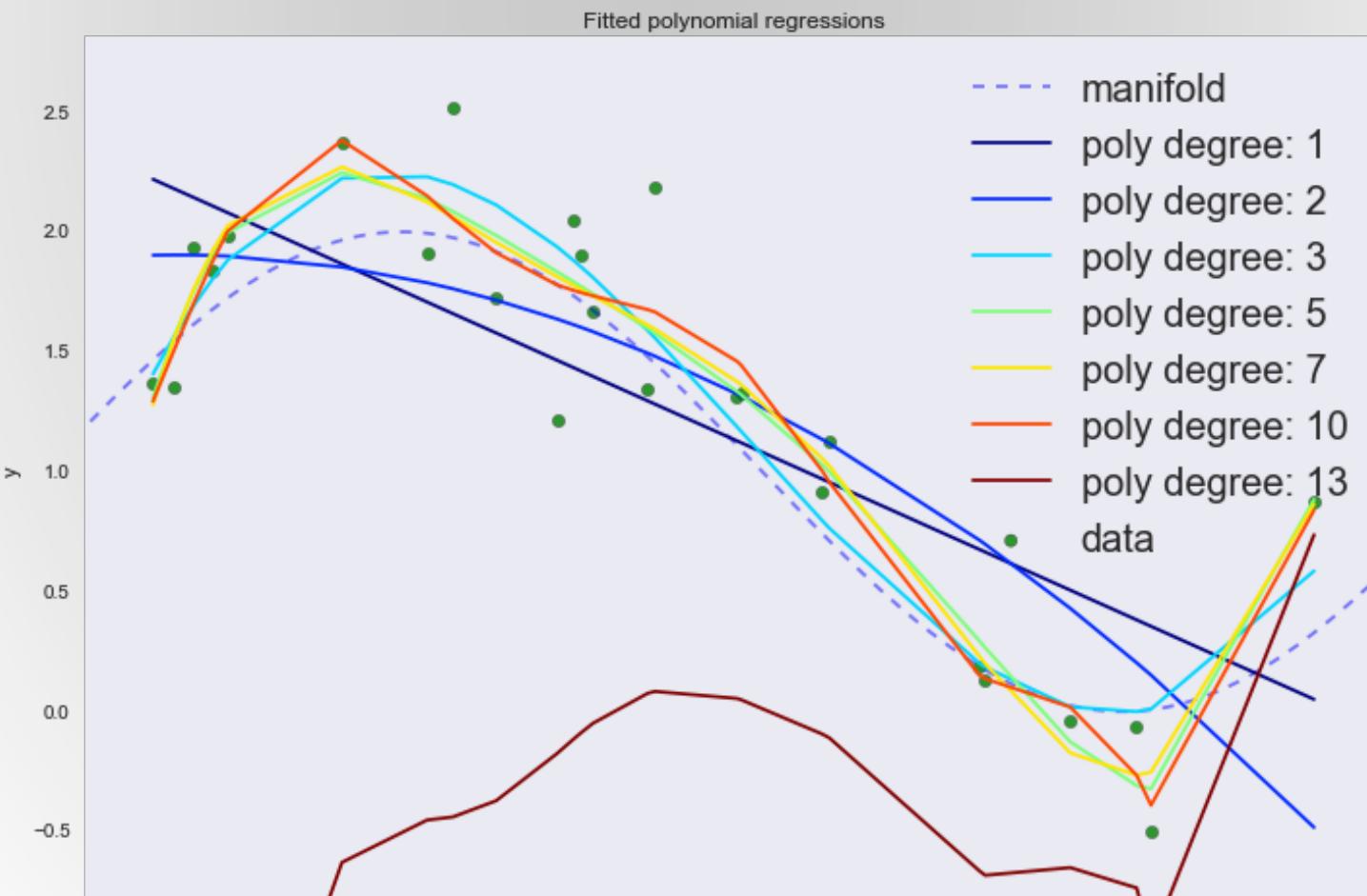
Fitting training data



# ПРИЗНАКИ ПЕРЕОБУЧЕННОЙ МОДЕЛИ

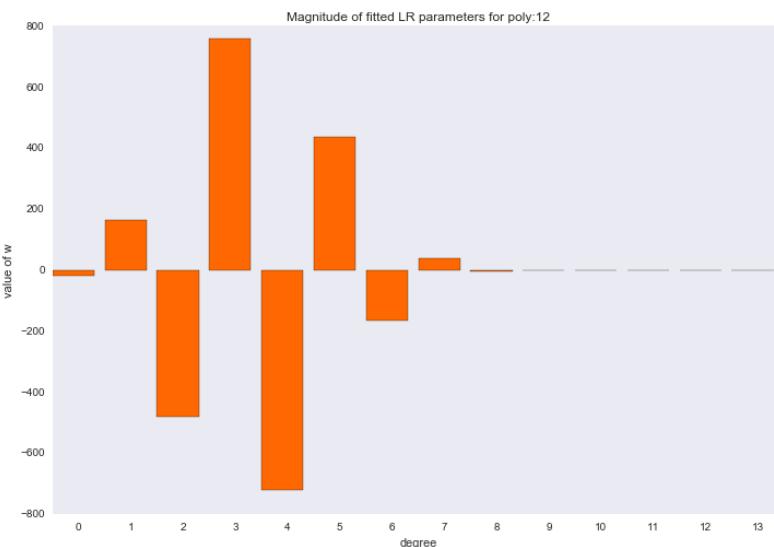
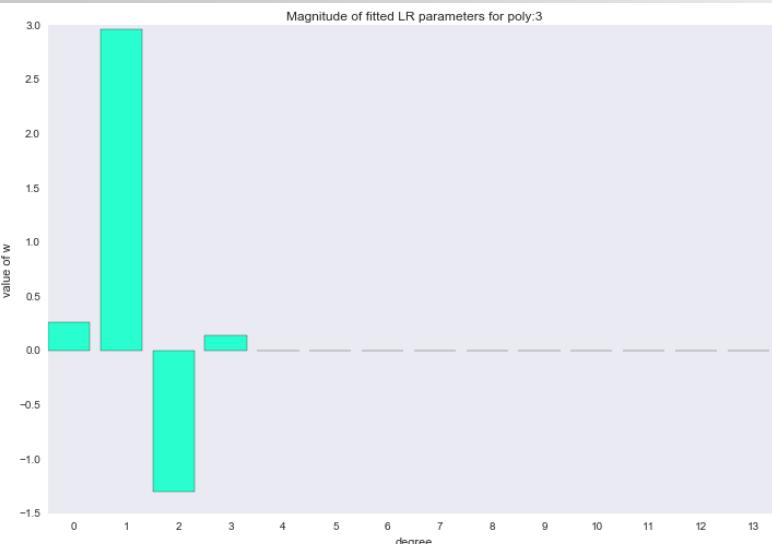
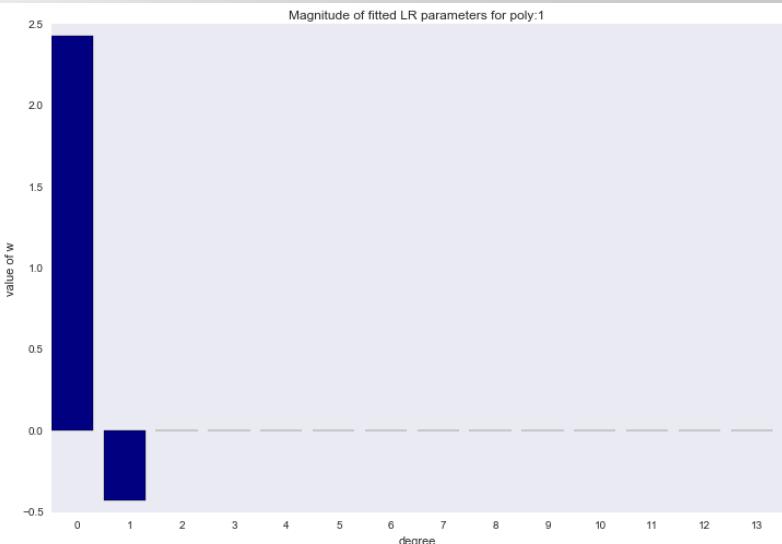
- Большая разница в качестве на тренировочных и тестовых данных (модель подгоняется под тренировочные данные и не может найти истинную зависимость)
- Большие значения параметров (весов)  $w_j$  модели
- Неустойчивость дискриминантной (разделяющей) функции ( $w, x$ ).

# ПЕРЕОБУЧЕНИЕ: ПРИМЕР



Какая кривая описывает данные лучше всего?

# ПЕРЕОБУЧЕНИЕ: ПРИМЕР



# МЕТОД БОРЬБЫ С ПЕРЕОБУЧЕНИЕМ: РЕГУЛЯРИЗАЦИЯ

**Утверждение.** Если в выборке есть линейно-зависимые признаки, то задача оптимизации  $Q(w) \rightarrow \min$  имеет бесконечное число решений.

- Большие значения параметров (весов) модели  $w$  – признак переобучения.

# МЕТОД БОРЬБЫ С ПЕРЕОБУЧЕНИЕМ: РЕГУЛЯРИЗАЦИЯ

**Утверждение.** Если в выборке есть линейно-зависимые признаки, то задача оптимизации  $Q(w) \rightarrow \min$  имеет бесконечное число решений.

- Большие значения параметров (весов) модели  $w$  – признак переобучения.

Решение проблемы – *регуляризация*.

Будем минимизировать регуляризованный функционал ошибки:

$$Q_{alpha}(w) = Q(w) + \alpha \cdot R(w) \rightarrow \min_w ,$$

где  $R(w)$  - регуляризатор.

$\alpha - ?$

# РЕГУЛЯРИЗАЦИЯ

- Регуляризация штрафует за слишком большие веса.

Наиболее используемые регуляризаторы:

- $L_2$ -регуляризатор:  $R(w) = \|w\|_2 = \sum_{i=1}^d w_i^2$
- $L_1$ -регуляризатор:  $R(w) = \|w\|_1 = \sum_{i=1}^d |w_i|$

# РЕГУЛЯРИЗАЦИЯ

- Регуляризация штрафует за слишком большие веса.

Наиболее используемые регуляризаторы:

- $L_2$ -регуляризатор:  $R(w) = \|w\|_2 = \sum_{i=1}^d w_i^2$
- $L_1$ -регуляризатор:  $R(w) = \|w\|_1 = \sum_{i=1}^d |w_i|$

Пример регуляризованного функционала:

$$Q(a(w), X) = \frac{1}{l} \sum_{i=1}^l ((w, x_i) - y_i)^2 + \alpha \sum_{i=1}^d w_i^2,$$

где  $\alpha$  – коэффициент регуляризации.

# АНАЛИТИЧЕСКОЕ РЕШЕНИЕ ЗАДАЧИ МНК С $L_2$ -РЕГУЛЯРИЗАТОРОМ

Задача оптимизации в матричном виде:

$$Q(w) = (y - Xw)^T(y - Xw) + \alpha w^T I w \rightarrow \min \quad (*)$$

где  $I$  – единичная матрица.

Эта задача имеет аналитическое решение:

$$\mathbf{w} = (X^T X + \alpha I)^{-1} X^T y$$

- Матрица  $X^T X + \alpha I$  всегда положительно определена, поэтому её можно обратить. Следовательно, задача (\*) имеет единственное решение.

# ПОЛЕЗНОЕ СВОЙСТВО L1-РЕГУЛЯРИЗАЦИИ

*Все ли признаки в задаче нужны?*

- Некоторые признаки могут не иметь отношения к задаче, т.е. они не нужны.
- Если есть ограничения на скорость получения предсказаний, то чем меньше признаков, тем быстрее
- Если признаков больше, чем объектов, то решение задачи будет неоднозначным.

*Поэтому в таких случаях надо делать отбор признаков, то есть убирать некоторые признаки.*

*Как доказать, что  $L_1$ -регуляризация занумерует веса?*

# $L_1$ -РЕГУЛЯРИЗАЦИЯ

**Утверждение.** В результате обучения модели с  $L_1$ -регуляризатором происходит зануление некоторых весов, т.е. отбор признаков.

Можно показать, что задачи

$$(1) \quad Q(w) + \alpha \left\| w \right\|_1 \rightarrow \min_w$$

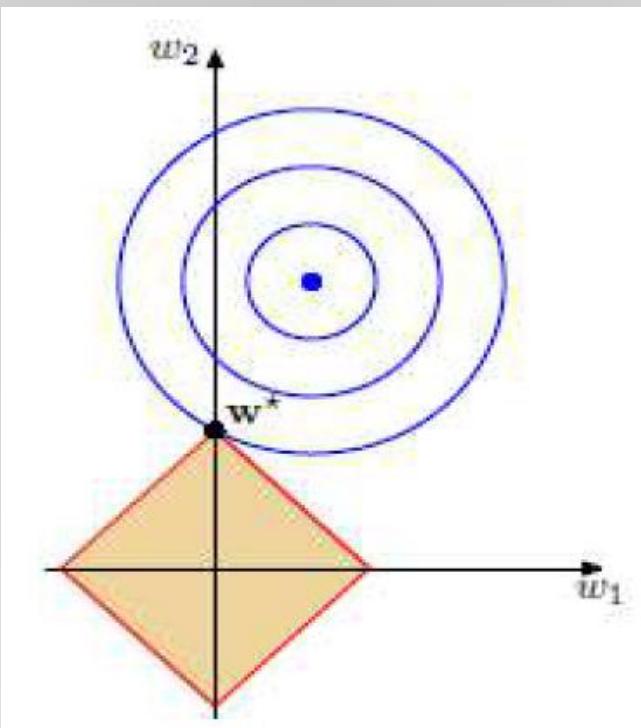
и

$$(2) \quad \begin{cases} Q(w) \rightarrow \min_w \\ \left\| w \right\|_1 \leq C \end{cases}$$

эквивалентны.

# ОТБОР ПРИЗНАКОВ ПО L1-РЕГУЛЯРИЗАЦИИ

Нарисуем линии уровня  $Q(w)$  и область  $\|w\|_1 \leq C$ :

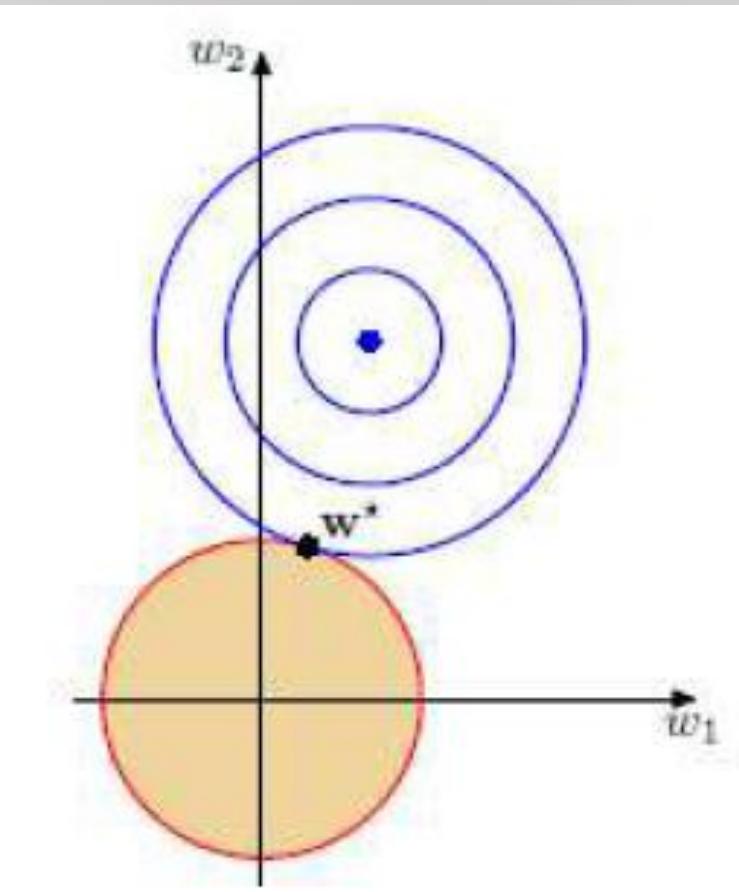


$$Q(w) = \text{MSE}$$

$$|w_1| + |w_2| \leq C$$

Если признак незначимый, то соответствующий вес близок к 0. Отсюда получим, что в большинстве случаев решение нашей задачи попадает в вершину ромба, т.е. обнуляет незначимый признак.

# L2-РЕГУЛЯРИЗАЦИЯ НЕ ОБНУЛЯЕТ ПРИЗНАКИ



$$w_1^2 + w_2^2 \leq C$$

Уменьшает веса,  
но не зачумка

# РАЗРЕЖЕННЫЕ МОДЕЛИ

Модели, в которых часть весов равна 0, называются ***разреженными моделями***.

- L1-регуляризация зануляет часть весов, то есть делает модель разреженной.

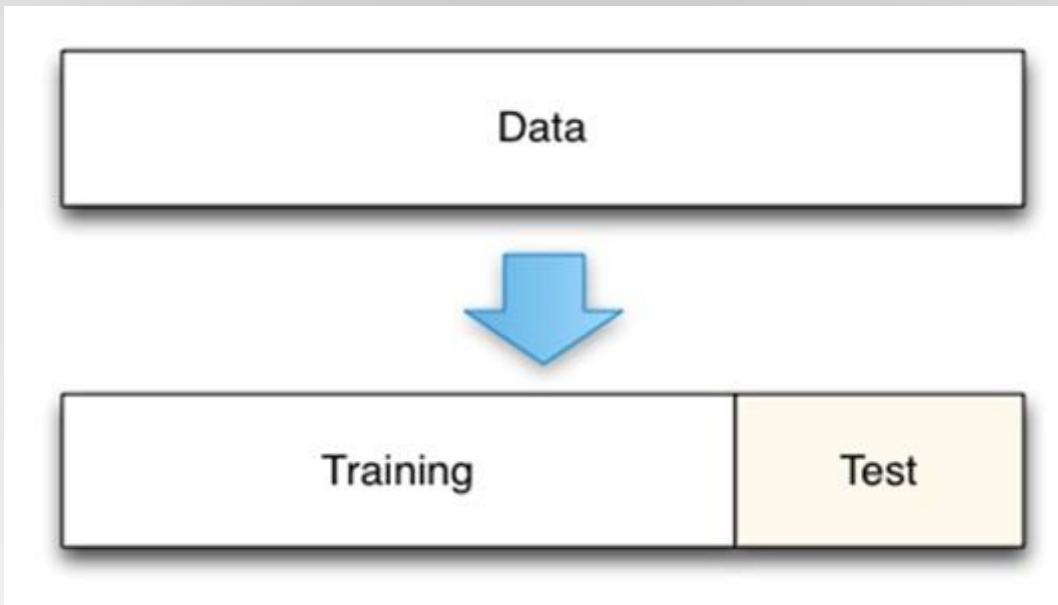
# ОЦЕНИВАНИЕ КАЧЕСТВА МОДЕЛИ

- Отложенная выборка
- Кросс-валидация

# ОТЛОЖЕННАЯ ВЫБОРКА

Делим тренировочную выборку на две части:

- По первой части обучаем модель (*train*)
- По оставшимся данным – оцениваем качество (*test*)

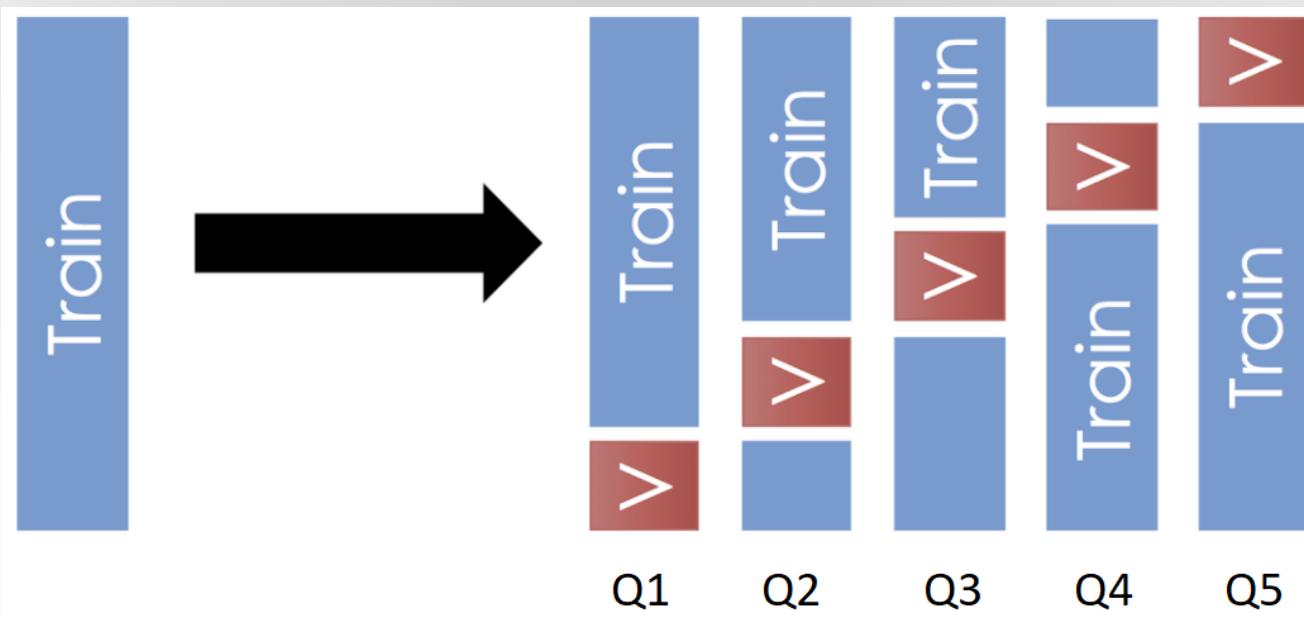


Недостаток:

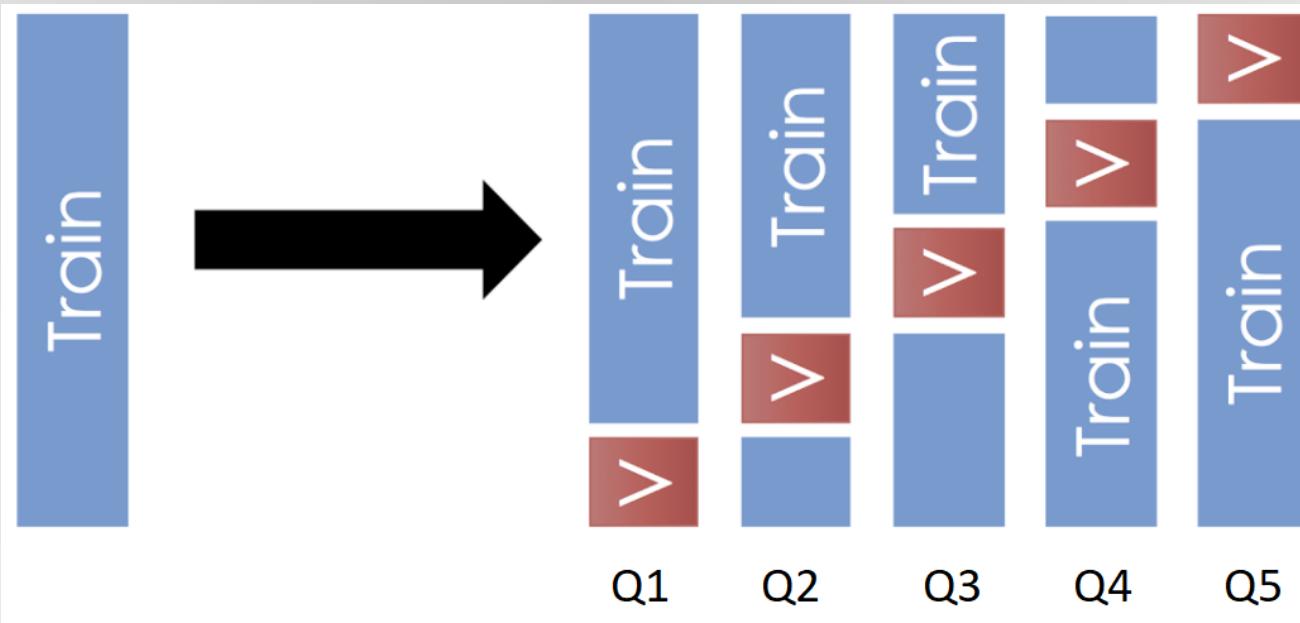
- Результат сильно зависит от разбиения на *train* и *test*

# КРОСС-ВАЛИДАЦИЯ

- Разбиваем объекты на тренировку (train) и валидацию (validation) несколько раз (при разбиении k раз получаем k-fold кросс-валидацию)
- Для каждого разбиения вычисляем качество на валидационной части
- Усредняем полученные результаты



# КРОСС-ВАЛИДАЦИЯ



$$CV = \frac{1}{k} \sum_{i=1}^k Q(a_i(x), X_i) = \frac{1}{k} \sum_{i=1}^k Q_i$$

# ВИДЫ КРОСС-ВАЛИДАЦИИ

- **k-fold cross-validation** – разбиваем данные на k блоков, каждый из которых по очереди становится контрольным (валидационным)
- **Complete cross-validation** – перебираем ВСЕ разбиения
- **Leave-one-out cross-validation** – каждый блок состоит из одного объекта (число блоков = числу объектов)

# ВЫБОР КОЛИЧЕСТВА БЛОКОВ В K-FOLD КРОСС-ВАЛИДАЦИИ



• Проблемы при маленьком  $k$ ?

• Проблемы при большом  $k$ ?

# ВЫБОР КОЛИЧЕСТВА БЛОКОВ В K-FOLD КРОСС-ВАЛИДАЦИИ



- Маленькое  $k$  – оценка может быть пессимистично занижена из-за маленького размера тренировочной части
- Большое  $k$  – оценка может иметь большую дисперсию из-за маленького размера валидационной части