

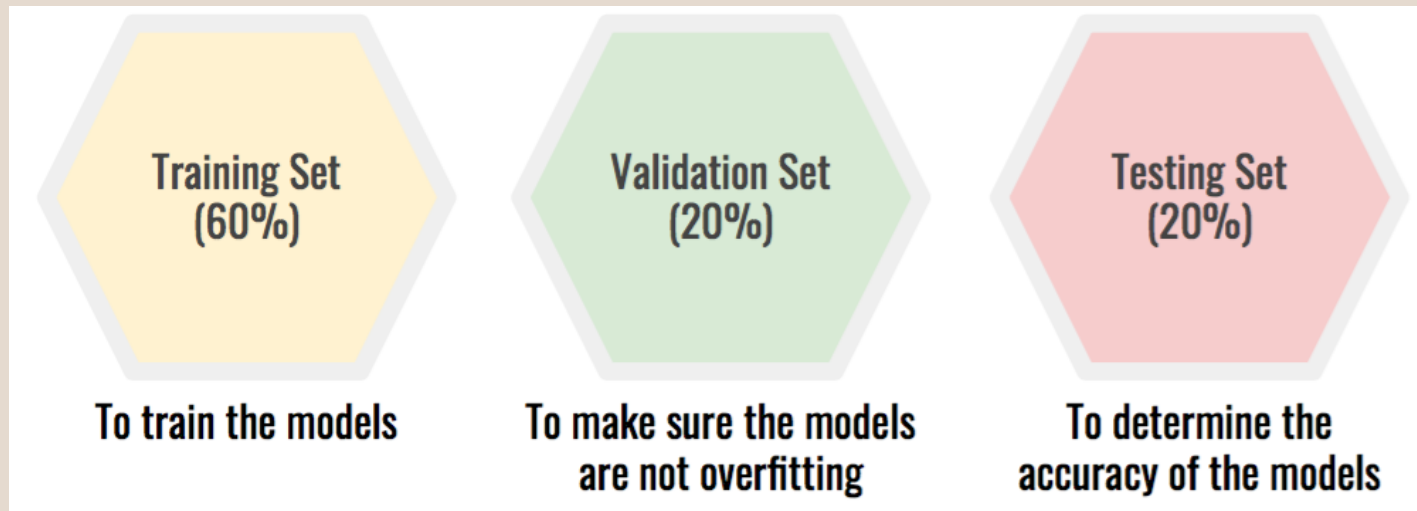


# Оценка качества. Кросс- валидация. Регуляризация. Lasso и Ridge

2022

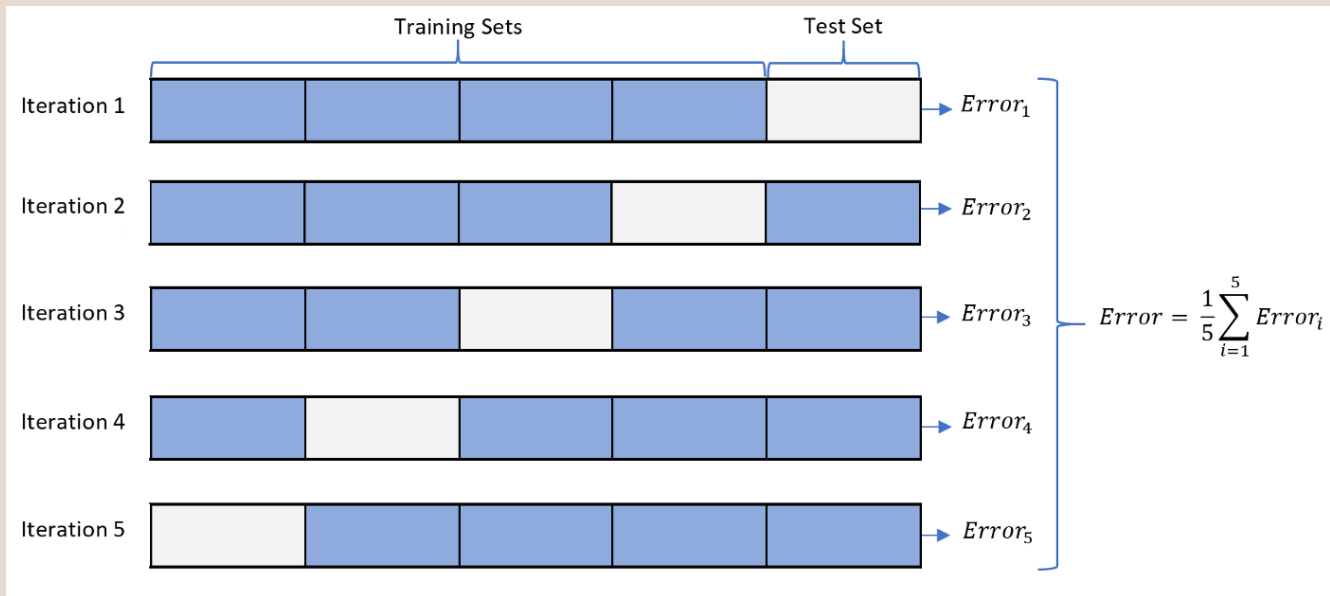
# Оценка качества работы алгоритма

- Делим выборку на несколько частей:
  - train и test – стандартная практика
  - train, validation & test – стандартная практика для нейронных сетей
- Недостаток: результат сильно зависит от разбиения



# 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

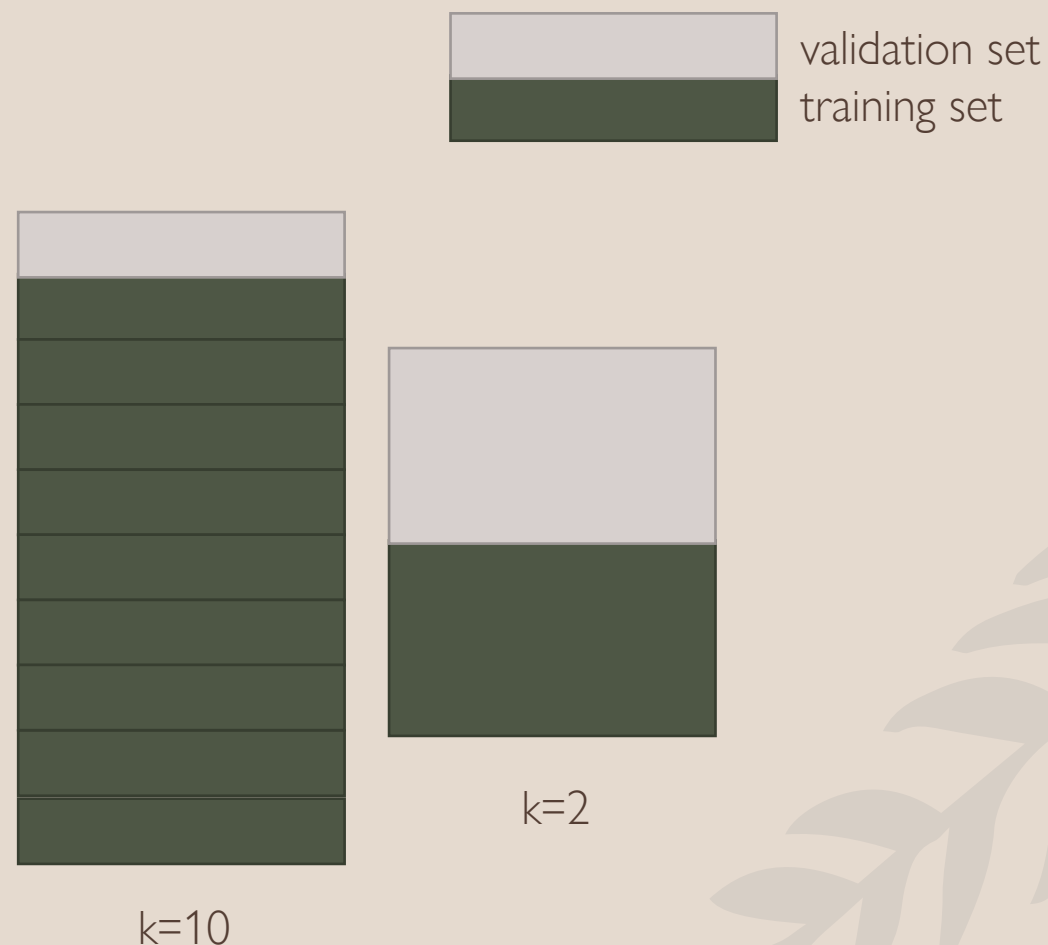


# Виды кросс-валидации

- k-fold cross-validation — разбиваем данные на k блоков, каждый из которых по очереди становится контрольным (валидационным)
- stratified k-fold cross-validation — стремимся, чтобы выборка равномерно распределялась по фолдам
- Complete cross-validation — перебираем ВСЕ разбиения
- Leave-one-out cross-validation — каждый блок состоит из одного объекта (число блоков = числу объектов)

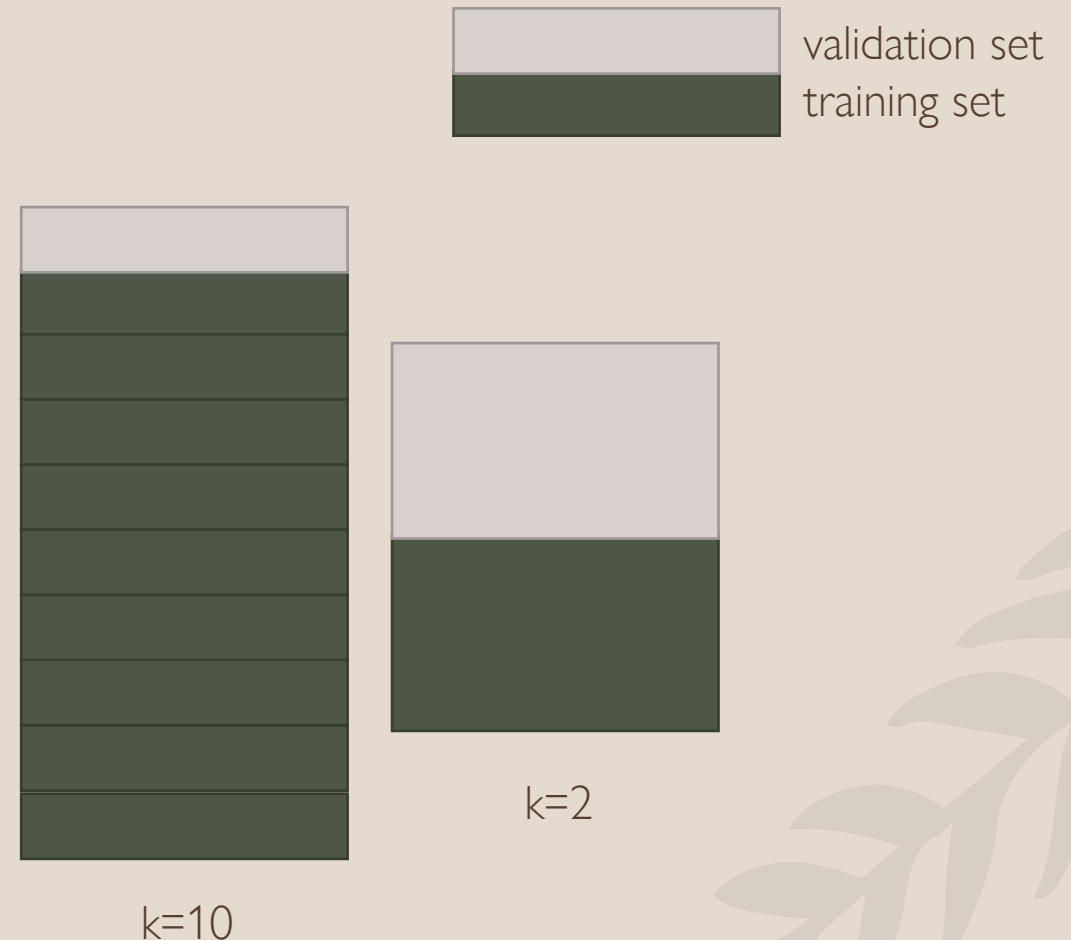
# Выбор количества блоков в k-fold

- Проблемы при маленьком  $k$ ?
- Проблемы при большом  $k$ ?



# Выбор количества блоков в k-fold

- 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

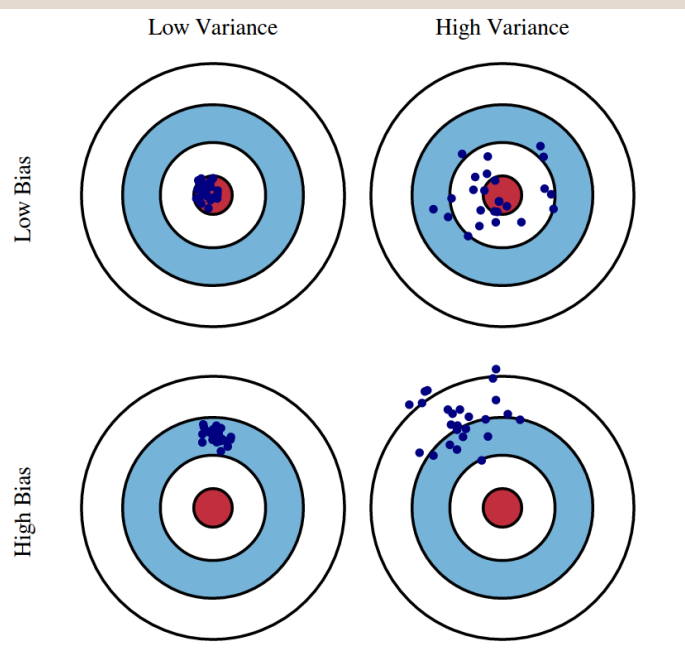


# Bias-Variance Tradeoff

**Bias** (смещение) – разность между средним предсказанием модели и истинным значением.

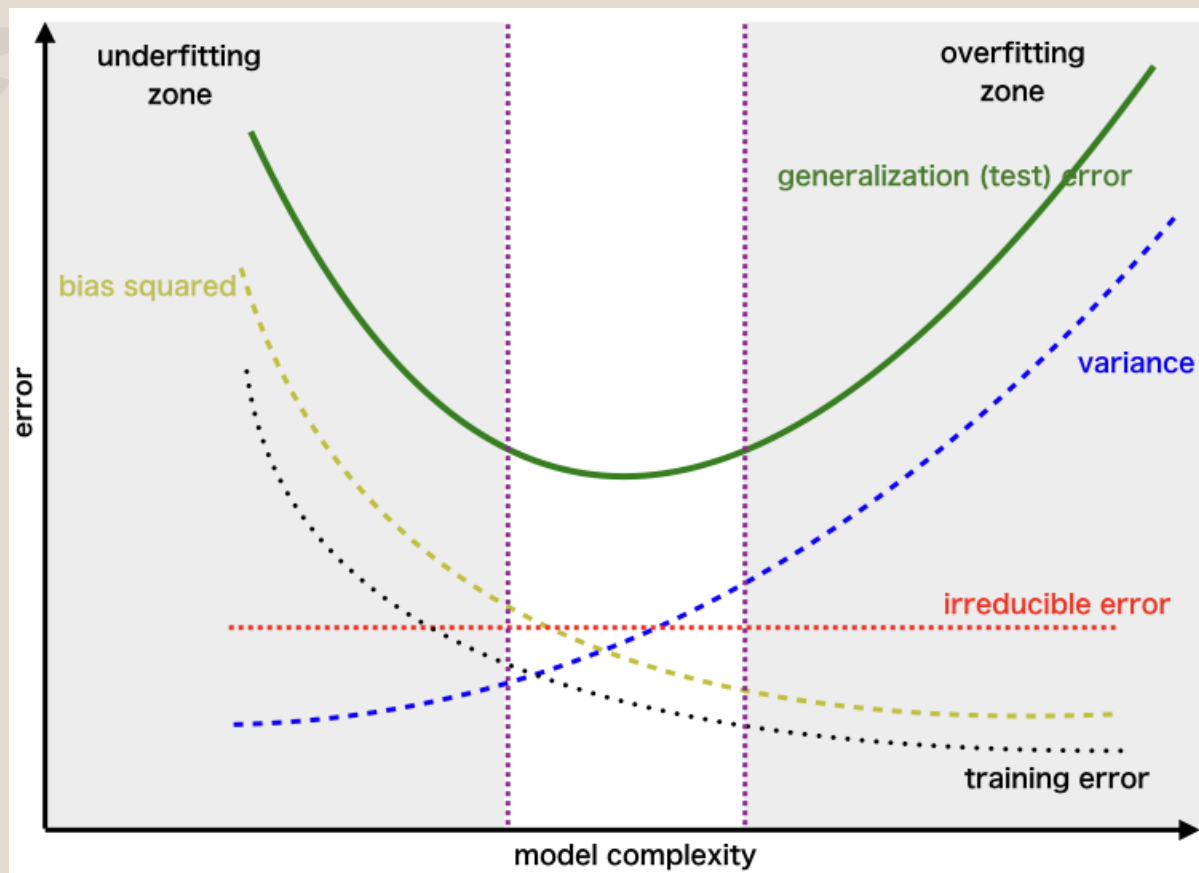
**Variance** (разброс) – вариативность предсказаний модели в данной точке.

Ошибку модели можно представить в виде:  $Err(x) = Bias^2(a(x)) + Var(a(x)) + \sigma^2$ .



- Модель с высоким смещением мало внимания обращает на обучающие данные, она переупрощена. Высокое смещение – недообучение.
- Модель с высоким разбросом очень чувствительна к изменениям обучающей выборки. Высокий разброс – сильное переобучение.

# Bias-Variance Tradeoff



Наша задача – найти баланс между смещением и разбросом...



# 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.

# 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:

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

где  $R(w)$  — регуляризатор.

# Регуляризация

Регуляризация штрафует за слишком большие веса.

Чаще всего используются регуляризаторы:

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

Пример регуляризованного функционала ошибки:

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

где  $\alpha$  — коэффициент регуляризации.

# Для чего это вообще?

## 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.

# $L_1$ -регуляризация

**Утверждение.** В результате обучения модели с  $L_1$ -регуляризатором происходит зануление некоторых признаков.

Можно показать, что задачи

1)

$$Q(w) + \alpha \|w\|_1 \rightarrow \min_w$$

и

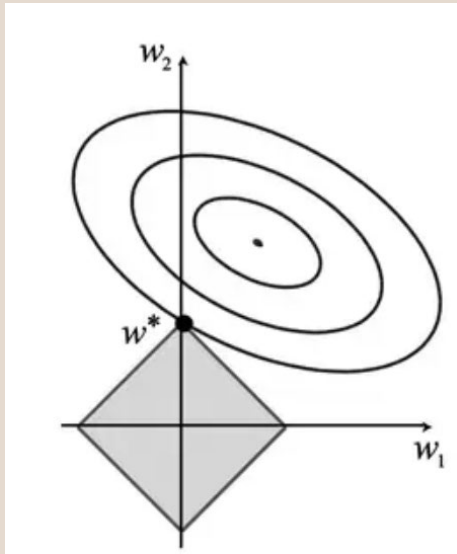
2)

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

эквивалентны.

# Отбор признаков по $L_1$ -регуляризации

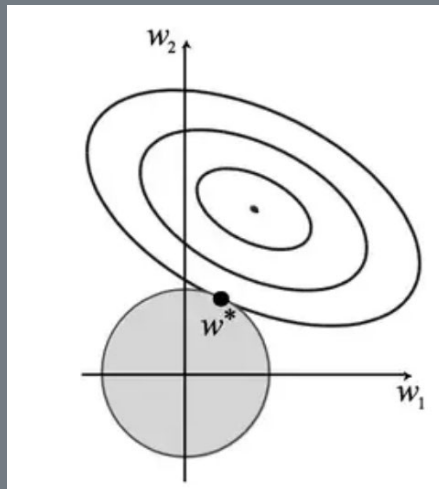
- Нарисуем линии уровня  $Q(w)$  и область  $\|w\|_1 \leq C$ :



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

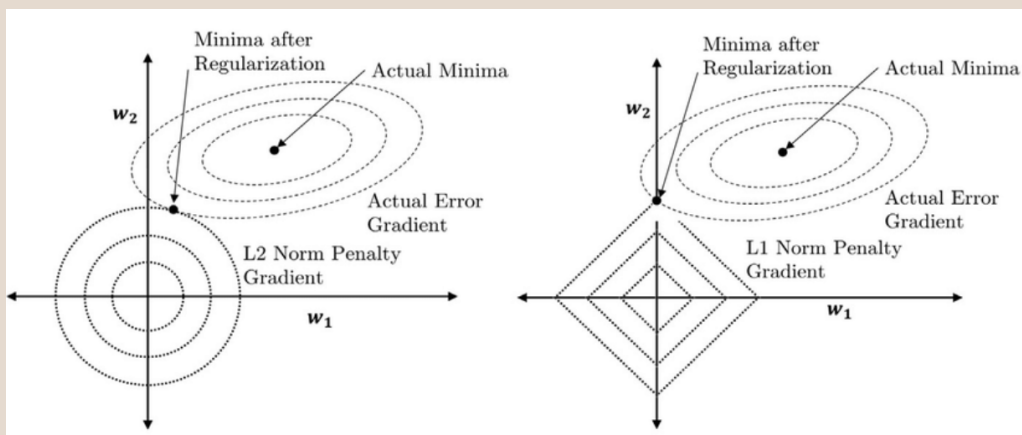
# $L_2$ -регуляризация

$L_2$ -регуляризация не обнуляет признаки.



# Регуляризация

- $L_1$ -регуляризация называется Lasso;
- $L_2$ -регуляризация называется Ridge.





# Гиперпараметры модели

- **Параметры** модели — величины, настраивающиеся по обучающей выборке (например, веса  $\mathbf{w}$  в линейной модели).
- **Гиперпараметры** модели — величины, контролирующие процесс обучения. Поэтому они не могут быть настроены по обучающей выборке (например, коэффициент регуляризации  $\alpha$ ).

*Проблема:* если подбирать гиперпараметры по кросс-валидации, то мы будем использовать отложенную (валидационную) выборку для поиска лучших значений гиперпараметров. То есть, отложенная выборка становится обучающей.

# Гиперпараметры модели

- Поэтому надо делать еще отдельную отложенную выборку для валидации:

