

ПРОБЛЕМА ПЕРЕОБУЧЕНИЯ

ПРОСТОЙ ПРИМЕР

- › Измеряем долю ошибок
- › На обучающей выборке: 0.2
- › Означает ли это, что алгоритм *обучился*?

ПРОСТОЙ ПРИМЕР

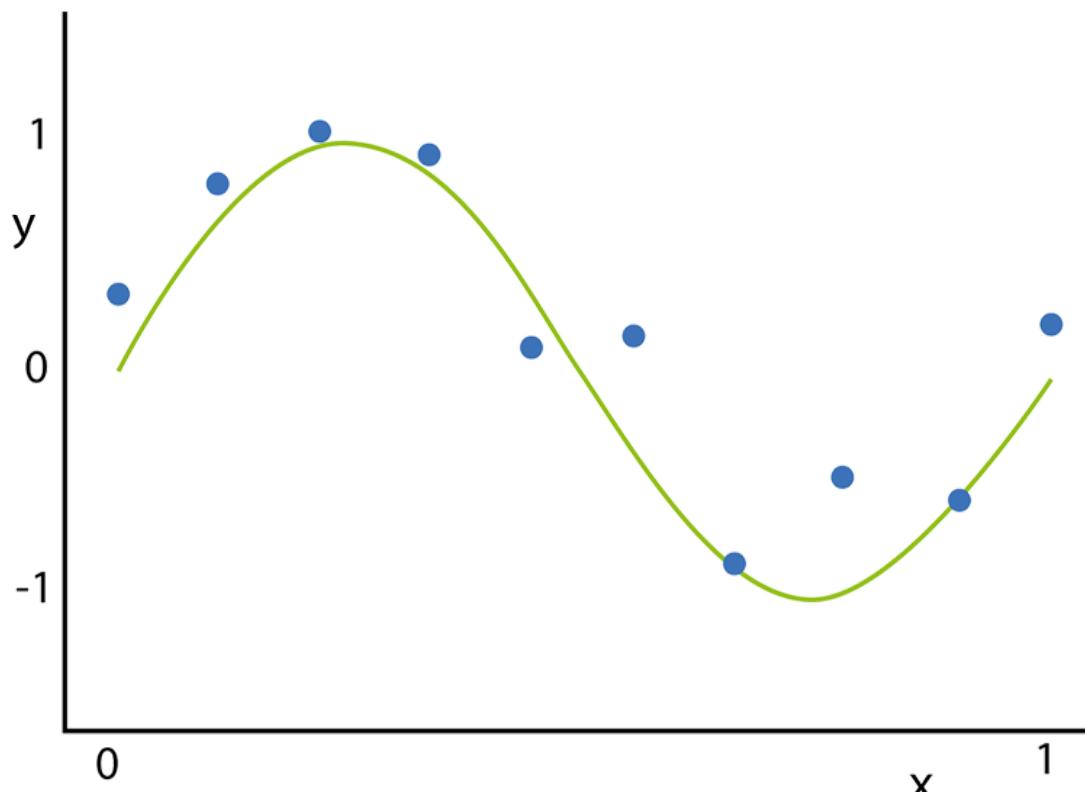
- › Измеряем долю ошибок
- › На обучающей выборке: 0.2
- › Новые данные: 0.9
- › Алгоритм не обладает обобщающей способностью

ПРОСТОЙ ПРИМЕР

- › Но при этом показал хорошее качество на обучении
- › Переобучение

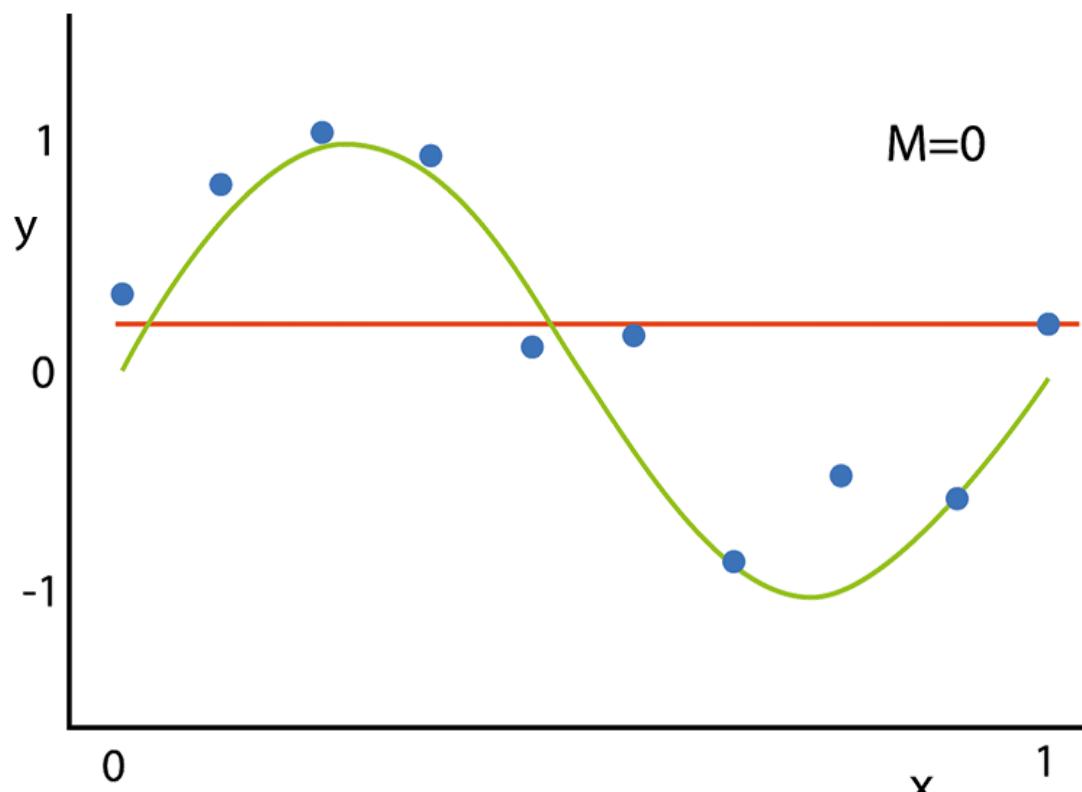
ЛИНЕЙНАЯ РЕГРЕССИЯ

- › Зеленый — истинная зависимость
- › Синий — выборка



ЛИНЕЙНАЯ РЕГРЕССИЯ

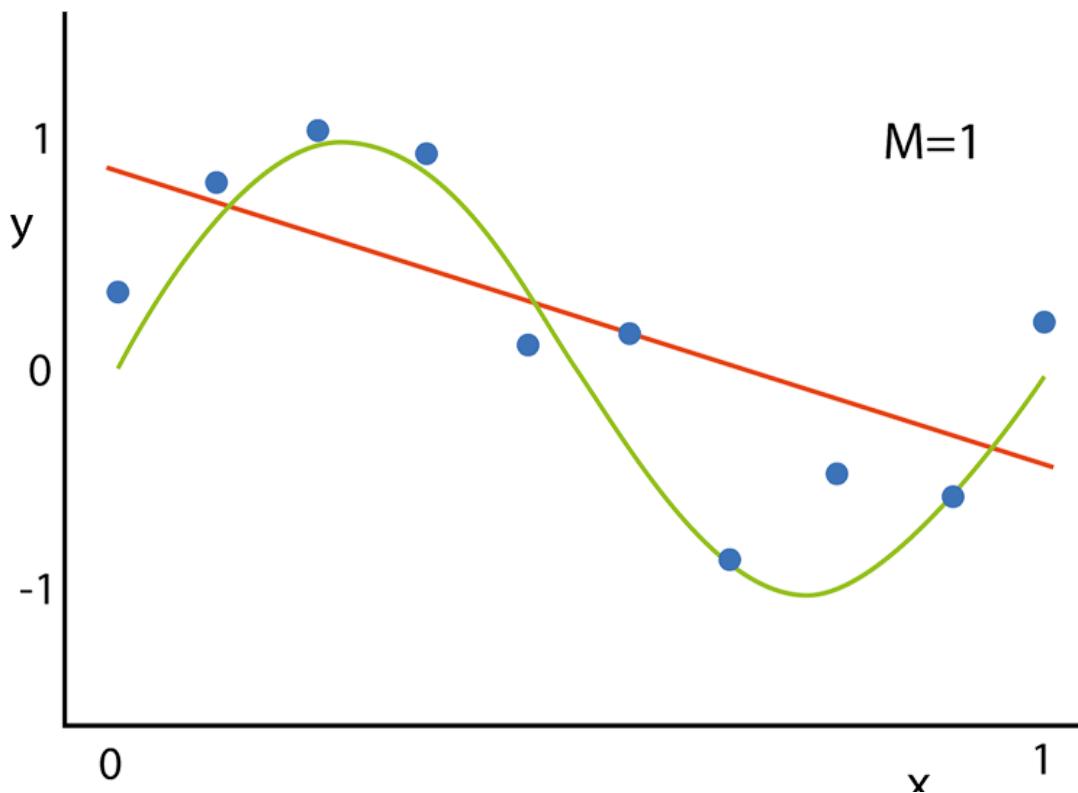
- » $a(x) = w_0$
- » Недообучение



ЛИНЕЙНАЯ РЕГРЕССИЯ

» $a(x) = w_0 + w_1 x$

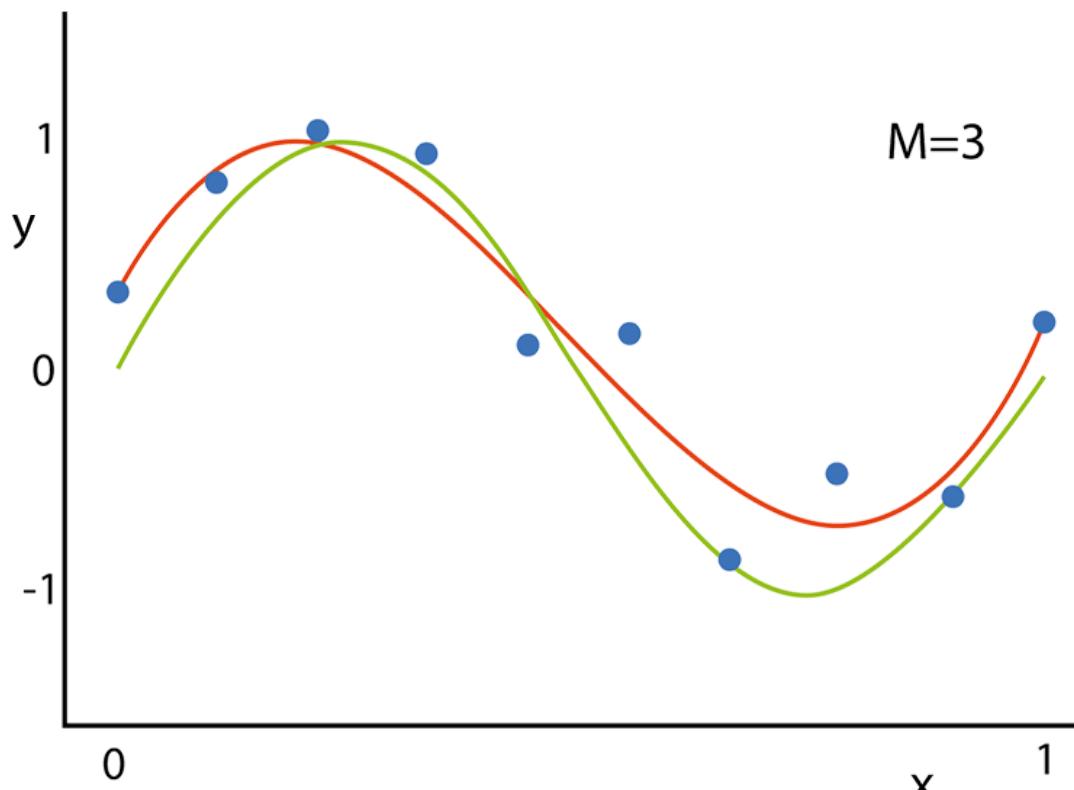
» Недообучение



ЛИНЕЙНАЯ РЕГРЕССИЯ

» $a(x) = w_0 + w_1 x + w_2 x^2 + w_3 x^3$

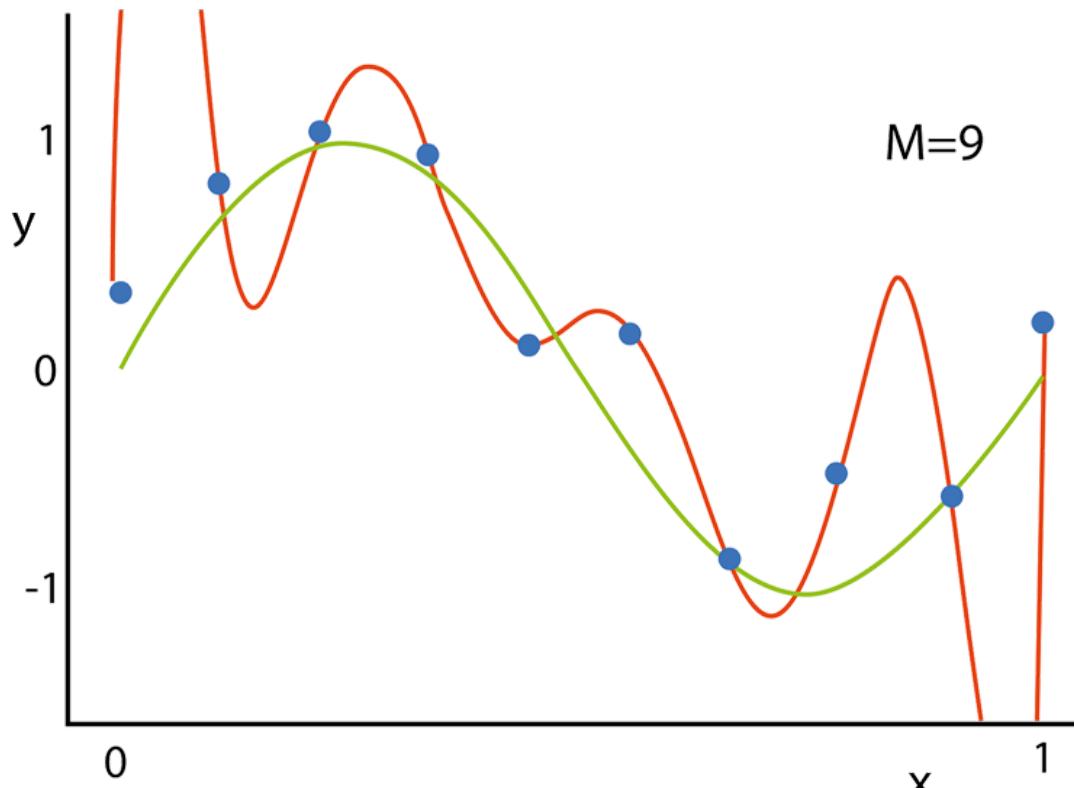
» То, что надо



ЛИНЕЙНАЯ РЕГРЕССИЯ

» $a(x) = w_0 + w_1 x + w_2 x^2 + \dots + w_9 x^9$

» Переобучение



ОБОБЩАЮЩАЯ СПОСОБНОСТЬ

- › Недообучение — **плохое** качество на обучении и на новых данных
- › Переобучение — **хорошее** качество на обучении, **плохое** на новых данных

КАК ВЫЯВИТЬ ПЕРЕОБУЧЕНИЕ?

- › Хороший алгоритм — хорошее качество на обучении
- › Переобученный алгоритм — хорошее качество на обучении
- › Нужны дополнительные данные

КАК ВЫЯВИТЬ ПЕРЕОБУЧЕНИЕ?

- › Отложенная выборка — данные, на которых не обучались
- › Кросс-валидация
- › Меры сложности модели

СЛОЖНОСТЬ ЛИНЕЙНОЙ РЕГРЕССИИ

» Модель:

$$a(x) = w_0 + w_1 x + w_2 x^2 + \dots + w_9 x^9$$

» В примере:

$$a(x) = 0.5 + 13458922x + \\ + 43983740x^2 + \dots + 2740x^9$$

» $a(x) = 0.4 + 8x - 23x^2 + 19x^3$

РЕЗЮМЕ

- › Переобучение — излишняя подгонка под обучающую выборку
- › Приводит к низкому качеству на новых данных
- › Большие веса — признак переобученности линейных моделей

РЕГУЛЯРИЗАЦИЯ

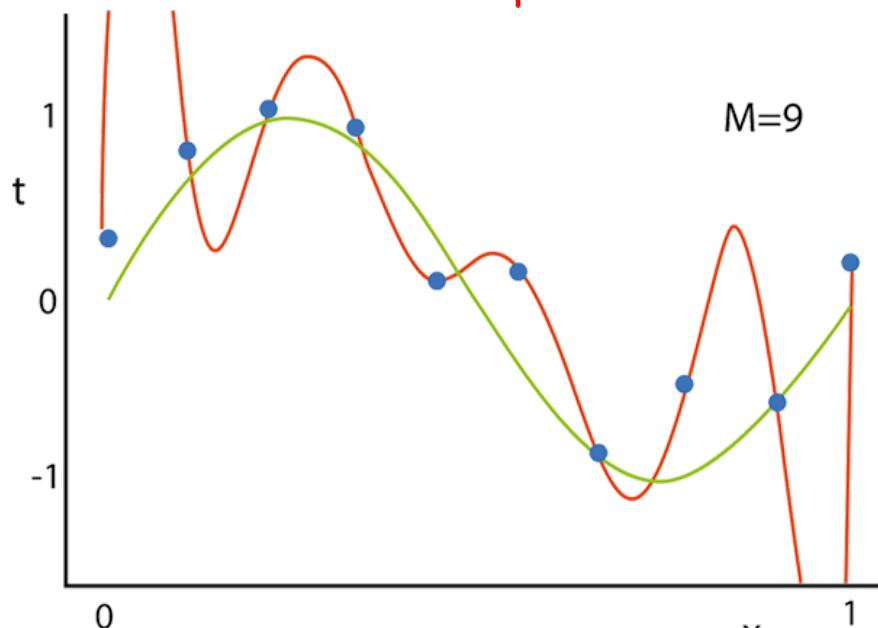
СЛОЖНОСТЬ ЛИНЕЙНОЙ РЕГРЕССИИ

» Модель:

$$a(x) = w_0 + w_1 x + w_2 x^2 + \dots + w_9 x^9$$

» В примере:

$$a(x) = 0.5 + 13458922x + \\ + 43983740x^2 + \dots + 2740x^9$$



МУЛЬТИКОЛЛИНЕАРНОСТЬ

- › Линейная зависимость признаков
- › Для любого объекта x_i обучающей выборки:
 $\alpha_1 x_i^1 + \dots + \alpha_d x_i^d = 0$
или
 $\langle \alpha, x_i \rangle = 0$

МУЛЬТИКОЛЛИНЕАРНОСТЬ

› Допустим, мы нашли решение:

$$\mathbf{w}_* = \underset{\mathbf{w}}{\operatorname{argmin}} \frac{1}{\ell} \sum_{i=1}^{\ell} (\langle \mathbf{w}, \mathbf{x}_i \rangle - y_i)^2$$

› Изменим вектор весов:

$$\mathbf{w}_1 = \mathbf{w}_* + t\alpha$$

› Ответ алгоритма на объекте:

$$\langle \mathbf{w}_* + t\alpha, \mathbf{x} \rangle = \langle \mathbf{w}_*, \mathbf{x} \rangle + t\langle \alpha, \mathbf{x} \rangle = \langle \mathbf{w}_*, \mathbf{x} \rangle$$

МУЛЬТИКОЛЛИНЕАРНОСТЬ

- › Бесконечно много оптимальных алгоритмов
- › Многие из них имеют большие веса
- › Не все из них имеют хорошую обобщающую способность

ПЕРЕОБУЧЕНИЕ

- › Большие веса в линейной модели — высокий риск переобучения
- › Будем штрафовать за это!

РЕГУЛЯРИЗАЦИЯ

› Функционал ошибки: $Q(\mathbf{w}, \mathbf{x})$

› Квадратичный регуляризатор:

$$\|\mathbf{w}\|^2 = \sum_{j=1}^d w_j^2$$

› Новый функционал:

$$Q(\mathbf{w}, \mathbf{X}) + \lambda \|\mathbf{w}\|^2 \rightarrow \min_{\mathbf{w}}$$

КОЭФФИЦИЕНТ РЕГУЛЯРИЗАЦИИ

$$Q(w, X) + \lambda \|w\|^2 \rightarrow \min_w$$

- › Чем больше λ , тем ниже сложность модели
- › Чем меньше λ , тем выше риск переобучения
- › Нужен баланс
- › Выбор λ – по кросс-валидации

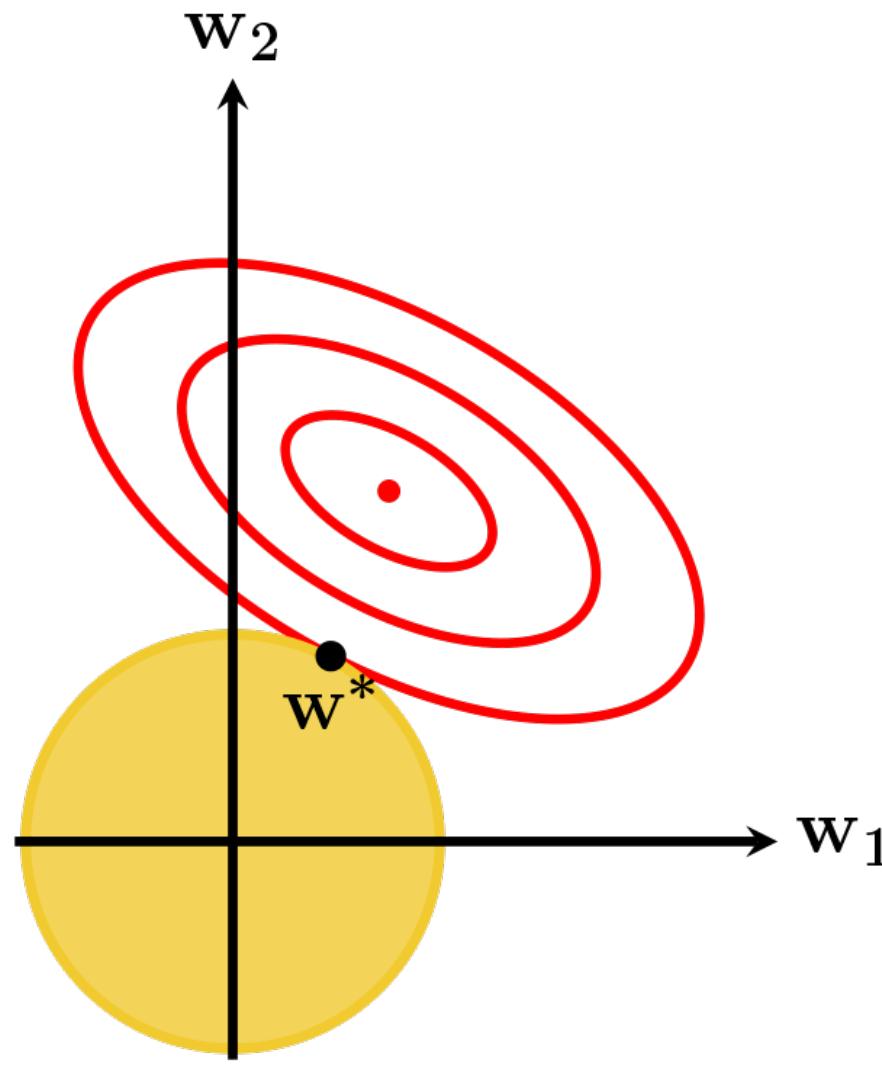
СМЫСЛ РЕГУЛЯРИЗАЦИИ

$$Q(\mathbf{w}, \mathbf{X}) + \lambda \|\mathbf{w}\|^2 \rightarrow \min_{\mathbf{w}}$$

› Эквивалентная задача:

$$\begin{cases} Q(\mathbf{w}, \mathbf{X}) \rightarrow \min_{\mathbf{w}} \\ \|\mathbf{w}\|^2 \leq C \end{cases}$$

СМЫСЛ РЕГУЛЯРИЗАЦИИ



ВИДЫ РЕГУЛЯРИЗАТОРОВ

L_2 -регуляризатор

- ▶ Штрафует модель за сложность
- ▶ Гладкий и выпуклый

ВИДЫ РЕГУЛЯРИЗАТОРОВ

L_1 -регуляризатор:

- ▶ $\| w \|_1 = \sum_{j=1}^d | w_j |$
- ▶ Негладкий
- ▶ Некоторые веса оказываются нулевыми
- ▶ Позволяет отбирать признаки

РЕЗЮМЕ

- › Большие веса в линейной модели — симптом переобучения
- › Регуляризация вводит штраф за большие веса
- › L_2 -регуляризация — частый выбор
- › L_1 -регуляризация — сложнее оптимизировать, но можно отбирать признаки

ОЦЕНИВАНИЕ КАЧЕСТВА АЛГОРИТМОВ

КАК ВЫЯВИТЬ ПЕРЕОБУЧЕНИЕ?

- › Хороший алгоритм — хорошее качество на обучении
- › Переобученный алгоритм — хорошее качество на обучении
- › Нужны дополнительные данные

КАК ОЦЕНИТЬ КАЧЕСТВО?

- › Как алгоритм будет вести себя на новых данных?
- › Какая у него будет доля ошибок?
- › По обучающей выборке нельзя это оценить

ОТЛОЖЕННАЯ ВЫБОРКА

- › Разбиваем выборку на две части
- › На первой обучаем алгоритм
- › На второй измеряем качество (тестовая выборка)
 - ▶ Доля ошибок
 - ▶ MSE
 - ▶ ...



ПРОПОРЦИИ РАЗБИЕНИЯ

- › Маленькая отложенная часть
 - ▶ (+) Обучающая выборка
репрезентативная
 - ▶ (-) Оценка качества ненадёжная
- › Большая отложенная часть
 - ▶ (+) Оценка качества надёжная
 - ▶ (-) Оценка качества смещённая
- › Обычно: 70/30, 80/20, 0.632/0.368

ОТЛОЖЕННАЯ ВЫБОРКА

- › (+) Обучаем алгоритм один раз
- › (-) Зависит от разбиения
- › Подходит, если данных очень много

Особые объекты



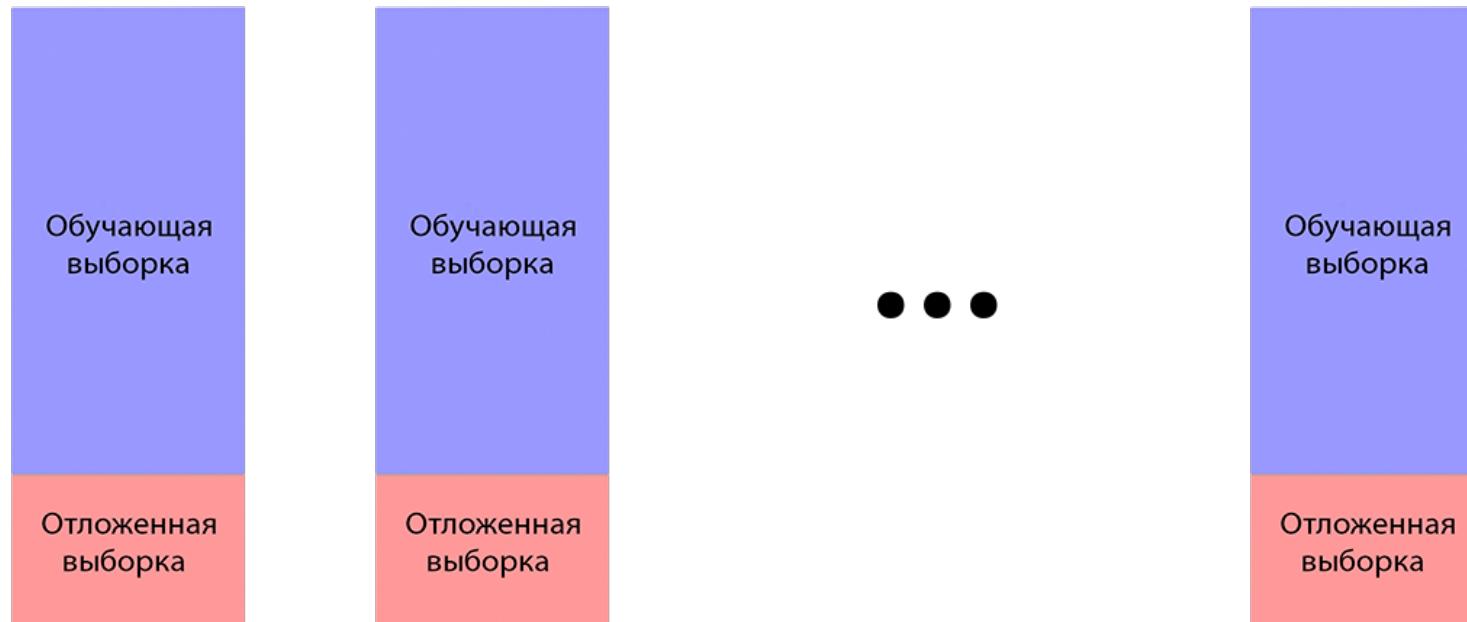
МНОГО ОТЛОЖЕННЫХ ВЫБОРОК

- Улучшение: разбиваем выборку на две части n раз
- Усредняем оценку качества



МНОГО ОТЛОЖЕННЫХ ВЫБОРОК

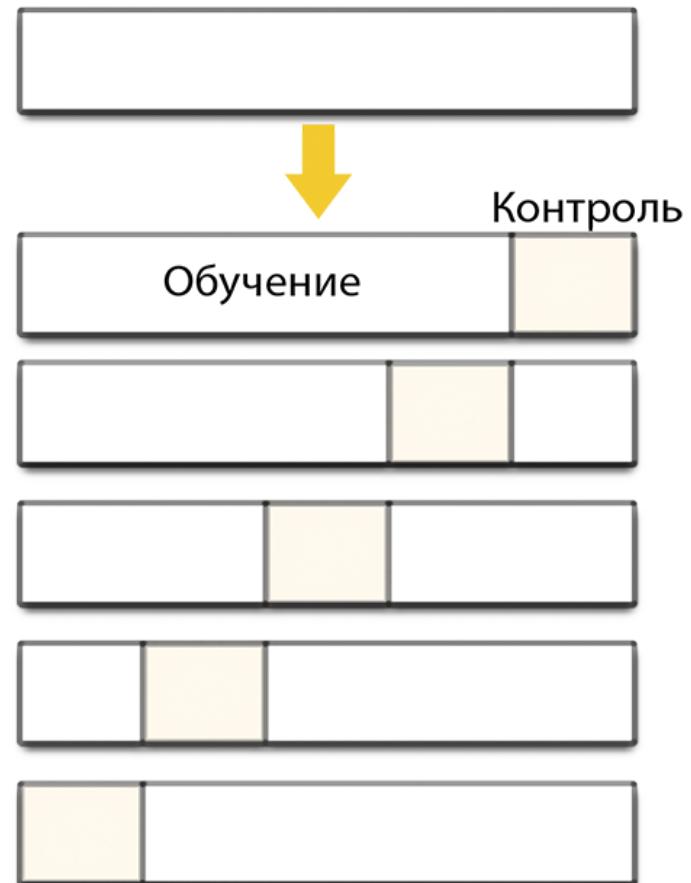
- › Нет гарантий, что каждый объект побывает в обучении



КРОСС-ВАЛИДАЦИЯ

› Разбиваем выборку на k блоков

› Каждая по очереди
выступает как тестовая



ЧИСЛО БЛОКОВ

› Мало блоков

- ▶ (+) Надёжные оценки
- ▶ (-) Смешённые оценки

› Много блоков

- ▶ (-) Ненадёжные оценки
- ▶ (+) Несмешённые оценки

ЧИСЛО БЛОКОВ

- › Обычно: $k = 3, 5, 10$
- › Чем больше выборка, тем меньше нужно k
- › Чем больше k , тем больше раз надо обучать алгоритм

СОВЕТ

- › Перемешивайте выборку!
- › Если не предсказываете будущее

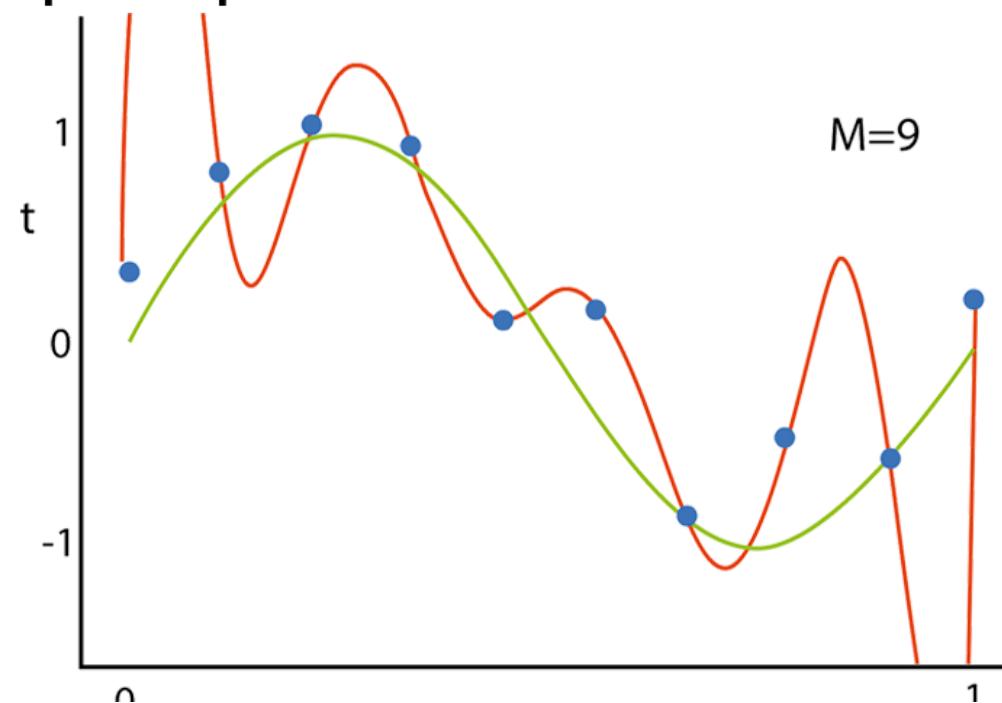
РЕЗЮМЕ

- › Для оценивания качества надо использовать данные вне обучения
- › Отложенная выборка
- › Кросс-валидация

СРАВНЕНИЕ АЛГОРИТМОВ И ВЫБОР ГИПЕРПАРАМЕТРОВ

ГИПЕРПАРАМЕТРЫ

- › Те параметры, которые нельзя настроить по обучающей выборке
- › Пример: параметр регуляризации
- › Пример: степень полинома



СРАВНЕНИЕ АЛГОРИТМОВ

- › Какой функционал лучше — **MSE** или **MAE**?
- › Какая регуляризация лучше?
- › Что лучше — линейная модель или решающее дерево?

ЧТО ИСПОЛЬЗОВАТЬ?

- › Отложенная выборка
- › Кросс-валидация
- › С осторожностью

ПРИМЕР

- › Сравним 1000 алгоритмов
- › Выбираем лучший на отложенной выборке
- › Отложенная выборка превращается в обучающую
- › Легко переобучиться

УЛУЧШЕННАЯ СХЕМА

Обучение

Валидация

Контроль

- › Настраиваем все алгоритмы на обучении
- › Сравниваем на валидации
- › Лучший проверяем на контроле

УЛУЧШЕННАЯ СХЕМА

Кросс-валидация

Контроль

- › Обучаем и сравниваем алгоритмы с помощью кросс-валидации
- › Лучший проверяем на контроле

РЕЗЮМЕ

- › При выборе гиперпараметров и сравнении моделей есть риск переобучения
- › Надо выделять контрольную выборку