

# Лекция 9

Линейная регрессия с точки зрения МО. Задача классификации

---

**Курс:** Введение в DS на УБ и МиРА (весна, 2022)

**Преподаватель:** Владимир Омелюсик

30 мая 2022 г.

## В предыдущих сериях

- Тестирование гипотез в линейной регрессии.
- Основные понятия машинного обучения.
- Виды задач машинного обучения.

# Линейная регрессия

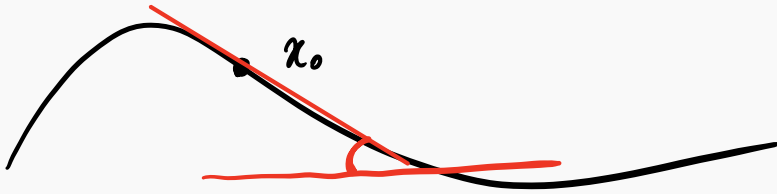
- Всё то же самое, что обсуждали до этого.
- Важно только качество предсказаний.
- Проблемы с обучением по формулам:


$$\hat{\beta} = (X^T X)^{-1} X^T y$$

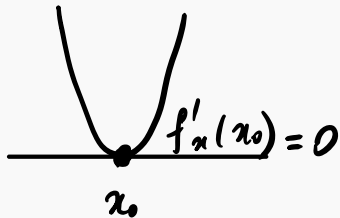
- Если матрица  $(X^T X)$  необратима, то будут проблемы с вычислениями.
- Произведение матриц – долгая операция.

# Обучение: градиентный спуск

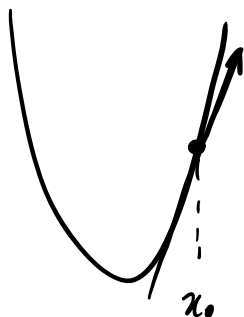
$$\lim_{x \rightarrow x_0} \frac{f(x) - f(x_0)}{x - x_0} = f'_x(x_0)$$



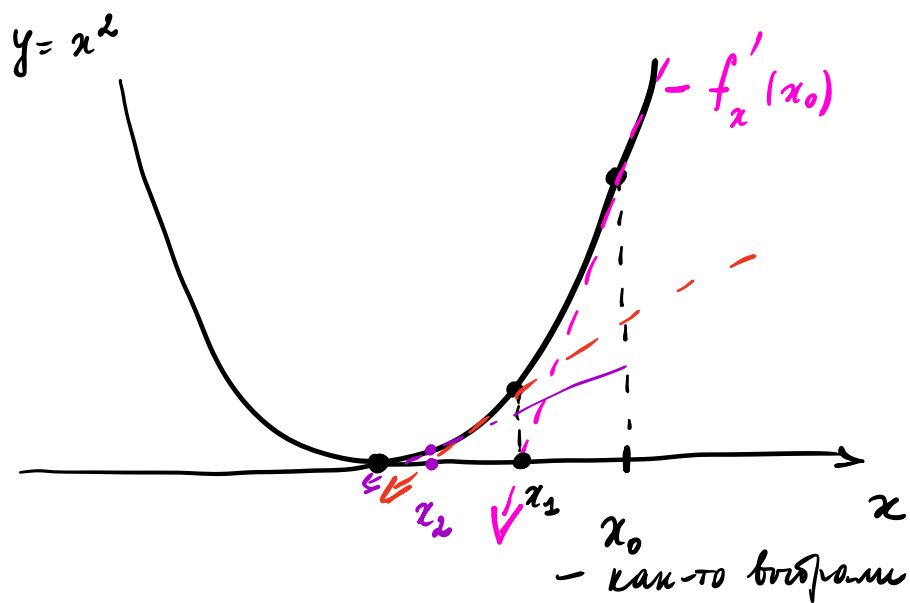
$x_0$  - экстремум и если  $f'_x(x_0)$  существ., то  $f'_x(x_0) = 0$



$f'_x(x_0)$  показыв. напрвл. максим. возраст.  
функции



$\Rightarrow$  ф-я убыв. быстрее всего в  
напрл.  $-f'_x(x_0)$



Останавли., если  $f'_x(x_i) = 0$

# Обучение: градиентный спуск

- В многомерном случае рассчитываем градиент:

$$\nabla_x f(x) = \left( \frac{df}{dx_1}, \dots, \frac{df}{dx_d} \right) - \text{градиент}$$

- Например, градиент MSE:

$$\nabla_{\beta} MSE = \frac{2}{N} X^T (Xw - y)$$

- Градиентный спуск для обучения:

$$\rightarrow \hat{\beta}_{t+1} = \hat{\beta}_t - \alpha \nabla_{\beta} MSE(\hat{\beta}_t),$$

где  $\alpha > 0$  – длина шага.

*длина шага*

# Алгоритм градиентного спуска

1. Выбираем начальное приближение  $\hat{\beta}_0$ .

2. Повторяем

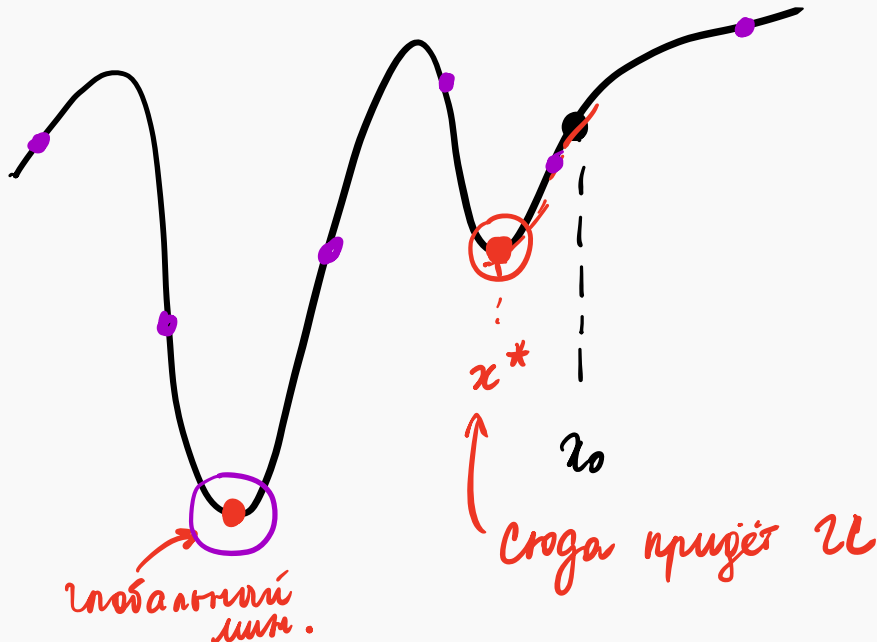
$$\longrightarrow \hat{\beta}_{t+1} = \hat{\beta}_t - \underbrace{\alpha \nabla_{\beta} \text{MSE}(\hat{\beta}_t)},$$

3. Останавливаемся, если

$$\underline{\|\hat{\beta}_t - \hat{\beta}_{t-1}\|_2 \leq \varepsilon}$$

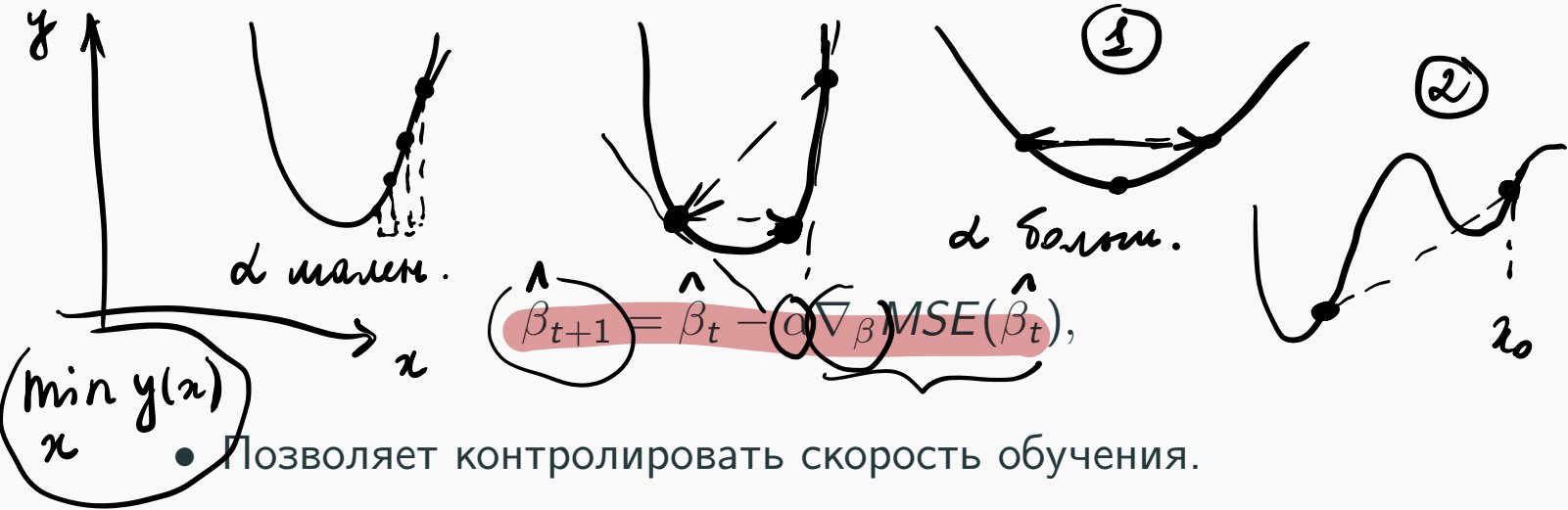
# Проблема градиентного спуска

- Градиентный спуск находит только локальные минимумы.
- Решение: мультистарт

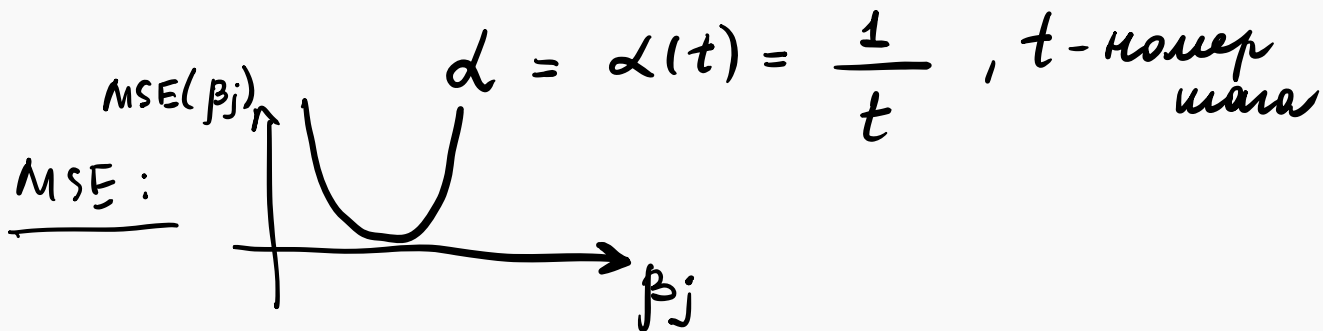




## Длина шага



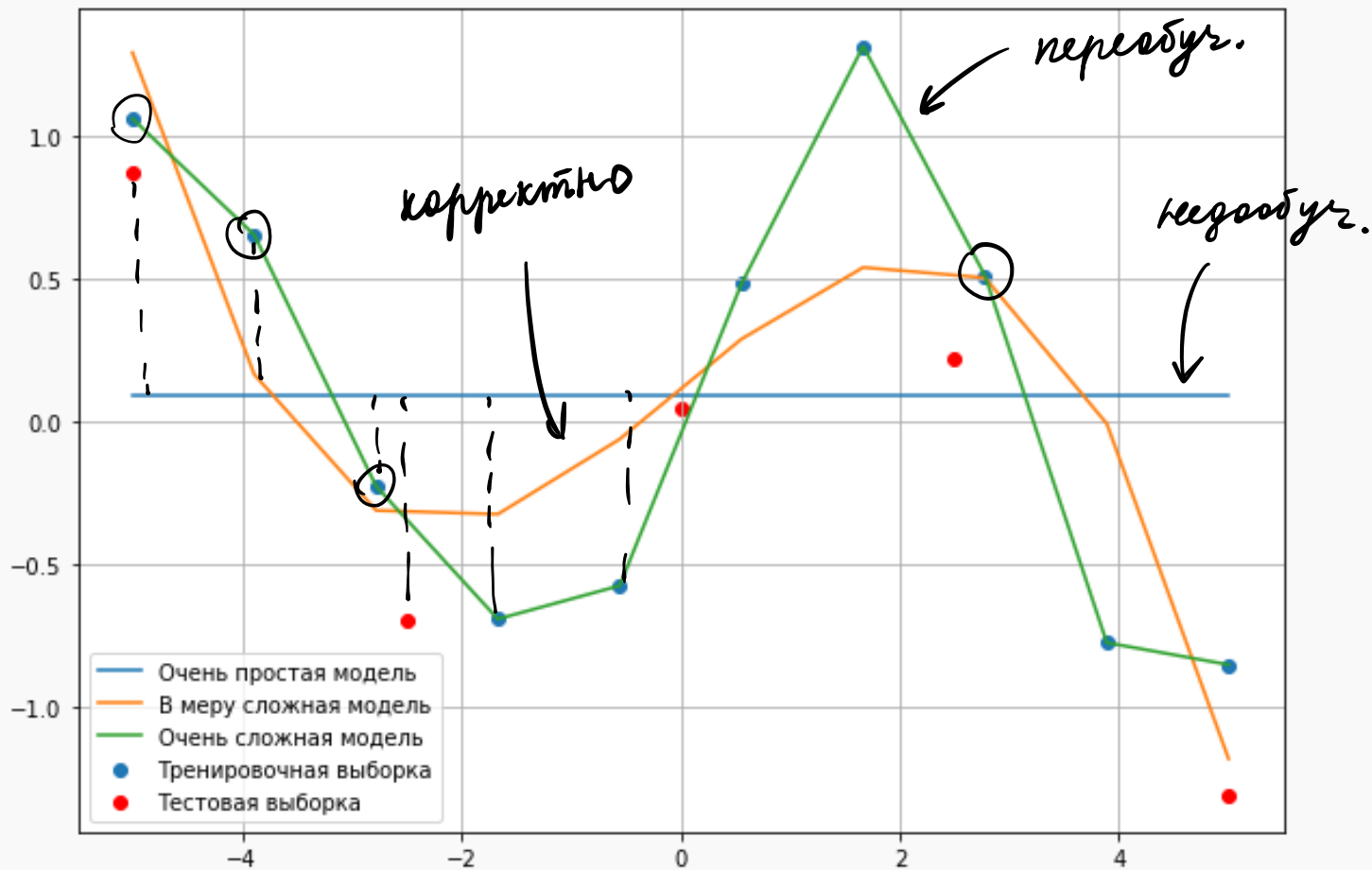
- Позволяет контролировать скорость обучения.
- Если сделать слишком большой, можно «перепрыгнуть» минимум.
- Гиперпараметр, нужно подбирать.



# Обобщающая способность модели

- Обобщающая способность – способность модели давать корректные предсказания на новых данных, не участвовавших при обучении.
- Недообучение – ситуация, когда модели не удалось правильно «запомнить» зависимости в данных. В этом случае качество будет низким как на обучающей выборке, так и на тестовой.
- Переобучение – ситуация, когда модель идеально «запомнила» свойства обучающей выборки, но не общие зависимости в ней. В этом случае качество будет высоким на обучающей выборке, но низким на новых данных.

# Обобщающая способность модели



# Переобучение в линейной регрессии

- Наблюдение: большие веса могут свидетельствовать о переобучении.

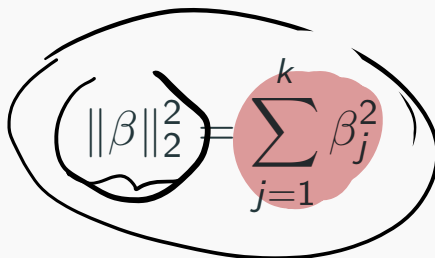
$$\hat{y}_i = 0.2 + \underline{1495.23x_i} + \dots,$$

если  $x_i$  — вес человека в кг, а  $y_i$  — рост человека в см — странно.

- Идея — штрафовать большие веса.

# Регуляризация

- Добавим к функции потерь регуляризатор. Например,



A diagram showing the L2 norm formula  $\|\beta\|_2^2 = \sum_{j=1}^k \beta_j^2$  enclosed in a hand-drawn oval. The summation part is highlighted with a red background.

$\|\beta\|$  — норма вект.

$\|\beta\|_2$  — евклидова норма

$$\|\beta\|_2^2$$

- Новая функция потерь:

$$\underbrace{\|y - X\beta\|_2^2}_{\text{ош.}} + \lambda \underbrace{\|\beta\|_2^2}_{\text{регул.}}$$

где  $\lambda$  — коэффициент регуляризации.

- Запускаем градиентный спуск на новой функции потерь.
- Важно не включать в регуляризатор  $\beta_0$ .

# Метрики качества на тестовой выборке

Всё те же, что были для статистики:  $MSE$ ,  $MAE$ ,  $R^2$ , ...

# Метод k ближайших соседей

- Дано:  $(X)$  и  $(y)$ .
- Решаем задачу многоклассовой классификации: каждое наблюдение может относиться к одному из  $K$  классов:  
 $y_i \in \{1, 2, \dots, K\}$

*предполагаем*

Гипотеза компактности

У «похожих» друг на друга объектов будут «похожие» ответы.

- Как определить похожесть? Для числовых признаков, например, так:

$x_1$     3    4    5

$x_2$     1    2    3

$$d(x_1, x_2) = \sqrt{(3-1)^2 + (4-2)^2 + (5-3)^2}$$

$$d(x_1, x_2) = \sqrt{\sum_{j=1}^k (x_1^j - x_2^j)^2} \quad \text{— евклидово расст.}$$

# Метод k ближайших соседей

1. **Обучение.** В kNN отсутствует. На этапе обучения происходит запоминание обучающей выборки  $X$ ,  $y$ .
2. **Предсказание.**
  - 2.1 Пусть нужно сделать предсказание для нового объекта  $x_i$ . Отсортируем объекты обучающей выборки по расстоянию до этого объекта.

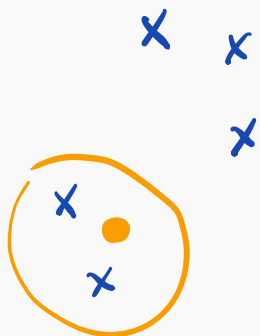
$$d(x_i, x_{(1)}) \leq d(x_i, x_{(2)}) \leq \dots$$

- 2.2 Предсказываем самый популярный класс среди  $k$  ближайших соседей.

$$\hat{y}_i = \arg \max_C \sum_{i=1}^k [y_{(i)} = C]$$

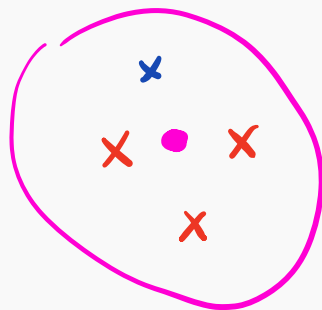


# Пример: kNN



(N2)  $k=4$

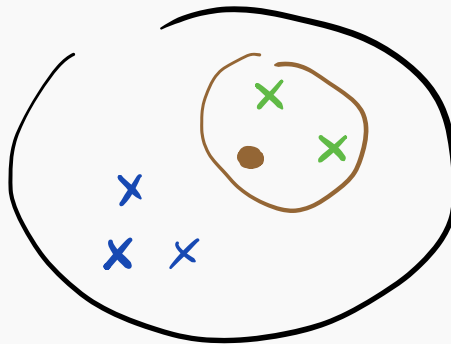
● - новая точка  
 $\Rightarrow$  красный



(N3)  $k=2$

● - н.т. = зелен. класс

(N4)  $k=5 \rightarrow$  синий класс



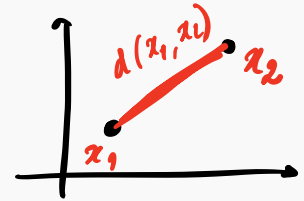
(N1)  $k=2$

● - новая точка  
 $\Rightarrow$  синий

# Расстояния

- Числовые признаки.
  - Евклидово расстояние.

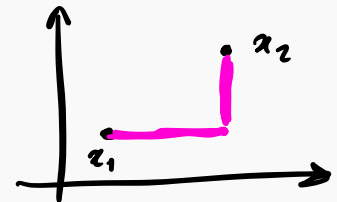
$$d(x_1, x_2) = \sqrt{\sum_{j=1}^k (x_1^j - x_2^j)^2}$$



- Манхэттэнское расстояние.



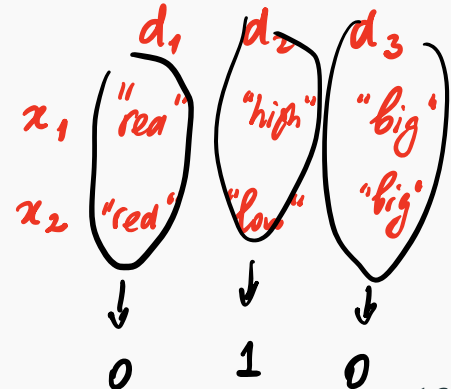
$$d(x_1, x_2) = \sum_{j=1}^k |x_1^j - x_2^j|$$



- Категориальные признаки.
  - Считающее расстояние.

$$d(x_1, x_2) = \sum_{j=1}^k [x_1^j \neq x_2^j]$$

$$d(x_1, x_2) = 0 + 1 + 0$$



$$\textcircled{1} \quad x_j := \frac{x_j - \text{mean}(x_j)}{\text{std}(x_j)}$$

$$\textcircled{2} \quad x_j := \frac{x_j - \min(x_j)}{\max(x_j)}$$

- Простой метод, основанный на расчётах расстояний.
- Гиперпараметры: число соседей  $k$  и функция расстояния.
- Проблема: поиск соседей может занимать долгое время.