Машинное обучение

Лекция 4

Линейная регрессия

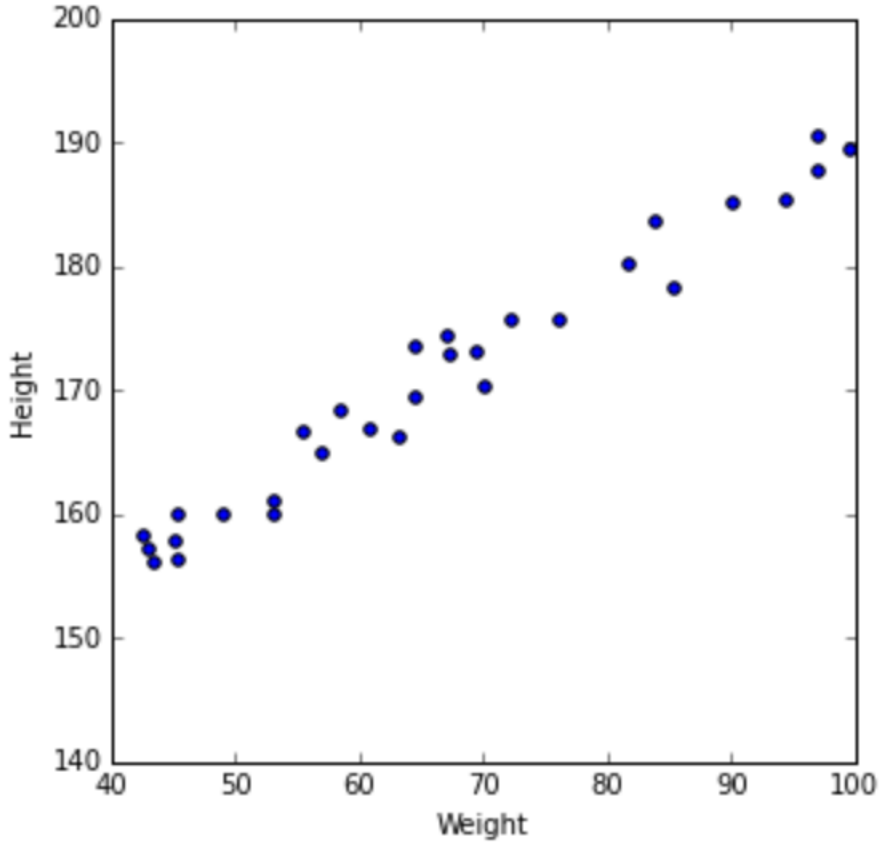
Градиентный спуск

Сергей Корпачев

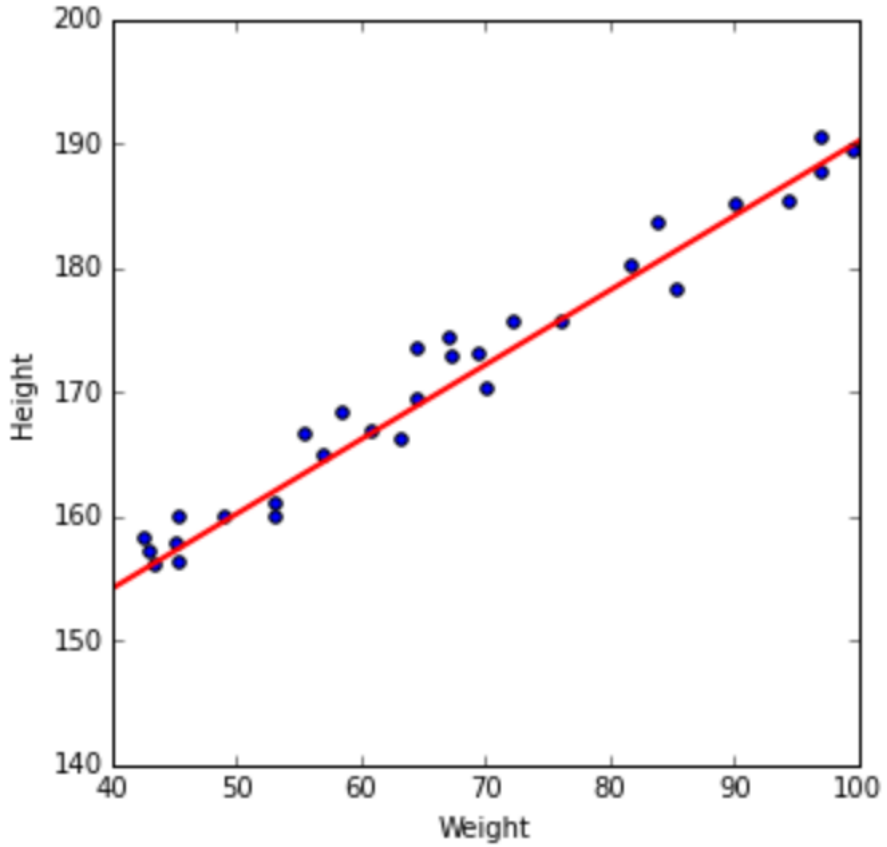
НИУ ВШЭ, 2026

Линейная регрессия

Парная регрессия



Парная регрессия



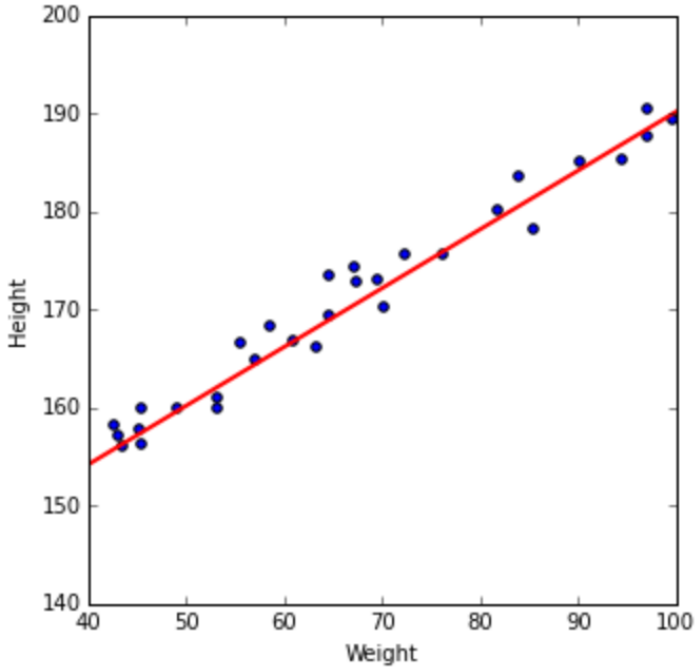
Парная регрессия

- Простейший случай: один признак
- Модель: $a(x) = w_1\, x + w_0$
- Два параметра: w_1 и w_0
- w_1 — тангенс угла наклона
- w_0 — где прямая пересекает ось ординат

Почему модель *линейная*?

$$a(x) = 2\,x + 1$$

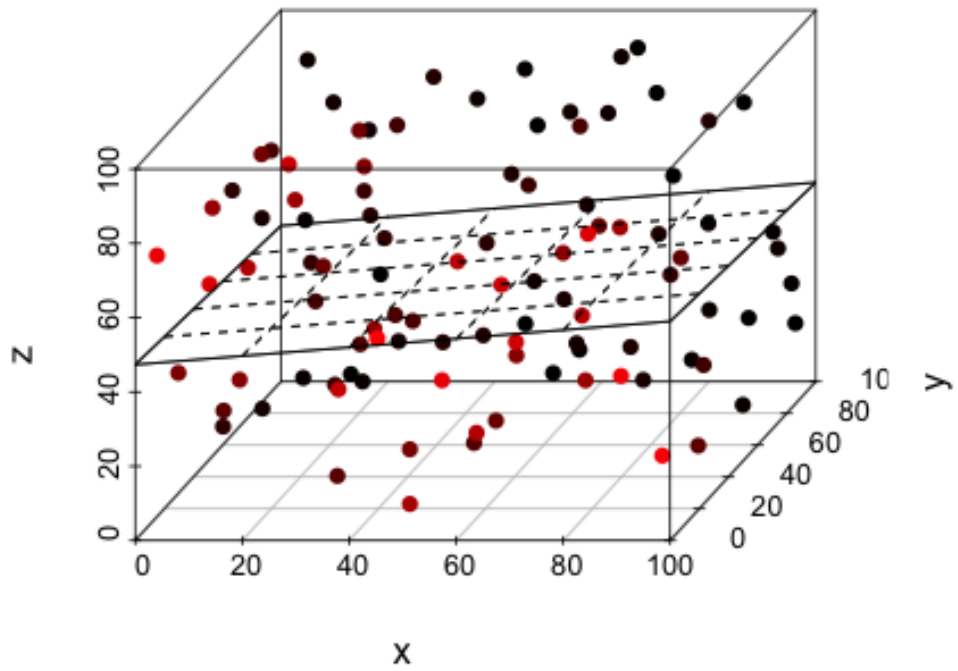
- $x = 1, a(x) = 3$
- $x = 2, a(x) = 5$
- $x = 10, a(x) = 21$
- $x = 20, a(x) = 41$



Два признака

- Чуть более сложный случай: два признака
- Модель: $a(x) = w_0 + w_1\, x_1 + w_2 x_2$
- Три параметра

Два признака



Много признаков

- Общий случай: d признаков
- Модель

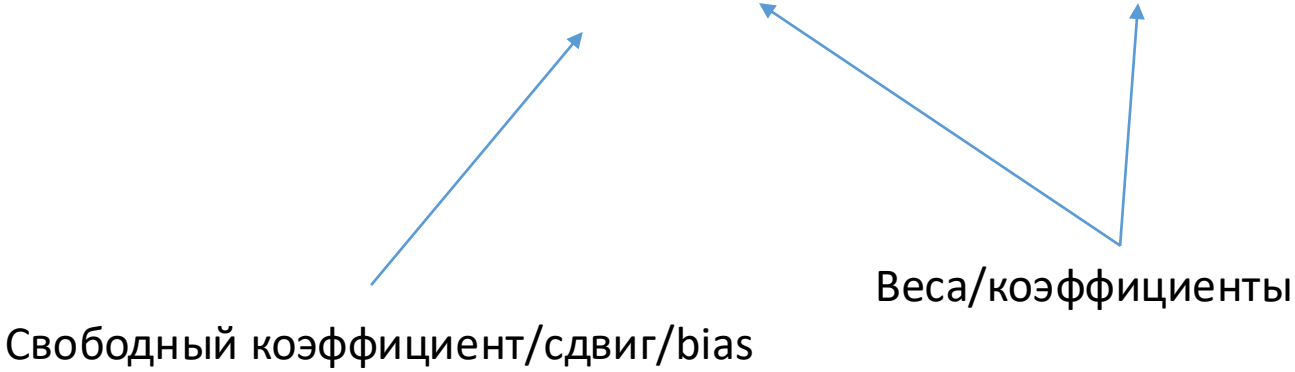
$$a(x) = w_0 + w_1 x_1 + \cdots + w_d x_d$$

- Количество параметров: $d + 1$

Много признаков

- Общий случай: d признаков
- Модель

$$a(x) = w_0 + w_1 x_1 + \cdots + w_d x_d$$



- Количество параметров: $d + 1$

Много признаков

Запишем через скалярное произведение:

$$a(x) = w_0 + w_1 x_1 + \cdots + w_d x_d =$$

$$= w_0 + \langle w, x \rangle$$

Будем считать, что есть признак, всегда равный единице:

$$a(x) = w_1 x_1 + \cdots + w_d x_d =$$

$$= w_1 * 1 + w_2 x_2 + \cdots + w_d x_d =$$

$$= \langle w, x \rangle$$

Применимость линейной регрессии

Модель линейной регрессии

$$a(x) = w_1 x_1 + \cdots + w_d x_d = \langle w, x \rangle$$

- Нет гарантий, что целевая переменная именно так зависит от признаков
- Надо формировать признаки так, чтобы модель подходила

Предсказание стоимости квартиры

- Признаки: площадь, район, расстояние до метро
- Целевая переменная: рыночная стоимость квартиры
- Линейная модель:

$$a(x) = w_0 + w_1 * (\text{площадь}) + w_2 * (\text{район}) + w_3 * (\text{расстояние до метро})$$

Предсказание стоимости квартиры

$$a(x) = w_0 + w_1 * (\text{площадь})$$
$$+ w_2 * (\text{район})$$
$$+ w_3 * (\text{расстояние до метро})$$

Предсказание стоимости квартиры

$$a(x) = w_0 + w_1 * (\text{площадь})$$
$$+ w_2 * (\text{район})$$
$$+ w_3 * (\text{расстояние до метро})$$

- За каждый квадратный метр добавляем w_1 к прогнозу

Предсказание стоимости квартиры

$$a(x) = w_0 + w_1 * (\text{площадь})$$
$$+ w_2 * (\text{район})$$
$$+ w_3 * (\text{расстояние до метро})$$

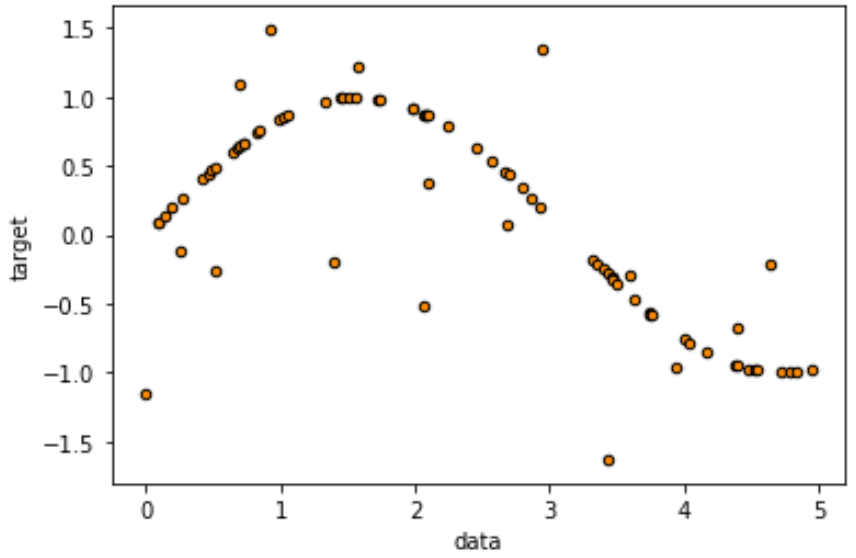
- Что-то странное

Предсказание стоимости квартиры

$$a(x) = w_0 + w_1 * (\text{площадь})$$

$$+ w_2 * (\text{район})$$

$$+ w_3 * (\text{расстояние до метро})$$



Кодирование категориальных признаков

- Значения признака «район»: $U = \{u_1, \dots, u_m\}$
- Новые признаки вместо x_j: $[x_j = u_1], \dots, [x_j = u_m]$
- One-hot кодирование

Кодирование категориальных признаков

Район
ЦАО
ЮАО
ЦАО
САО
ЮАО

ЦАО	ЮАО	САО
1	0	0
0	1	0
1	0	0
0	0	1
0	1	0

Кодирование категориальных признаков

Район
ЦАО
ЮАО
ЦАО
САО
ЮАО

ЦАО	ЮАО	САО
1	0	0
0	1	0
1	0	0
0	0	1
0	1	0

$$a(x) = w_0 + w_1 * (\text{площадь})$$

$$+ w_2 * (\text{квартира в ЦАО?})$$

$$+ w_3 * (\text{квартира в ЮАО?})$$

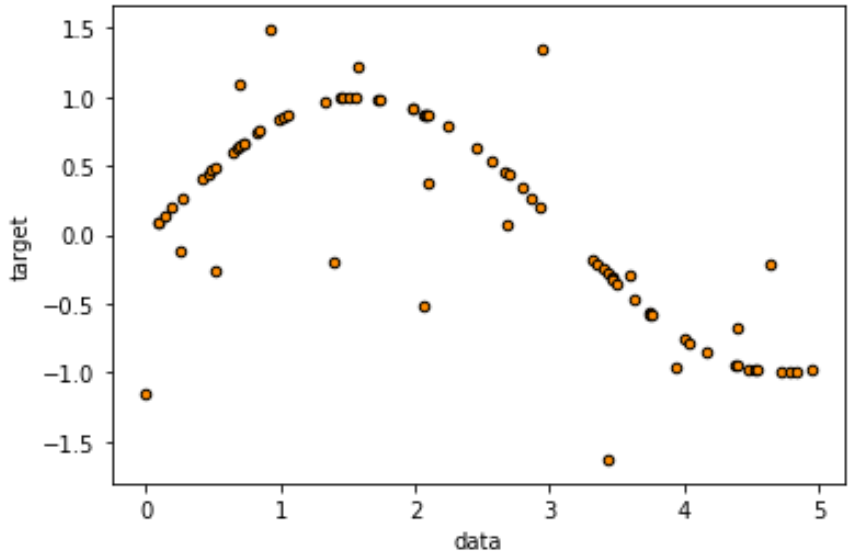
$$+ w_4 * (\text{квартира в САО?})$$

Предсказание стоимости квартиры

$$a(x) = w_0 + w_1 * (\text{площадь})$$

$$+ w_2 * (\text{район})$$

$$+ w_3 * (\text{расстояние до метро})$$

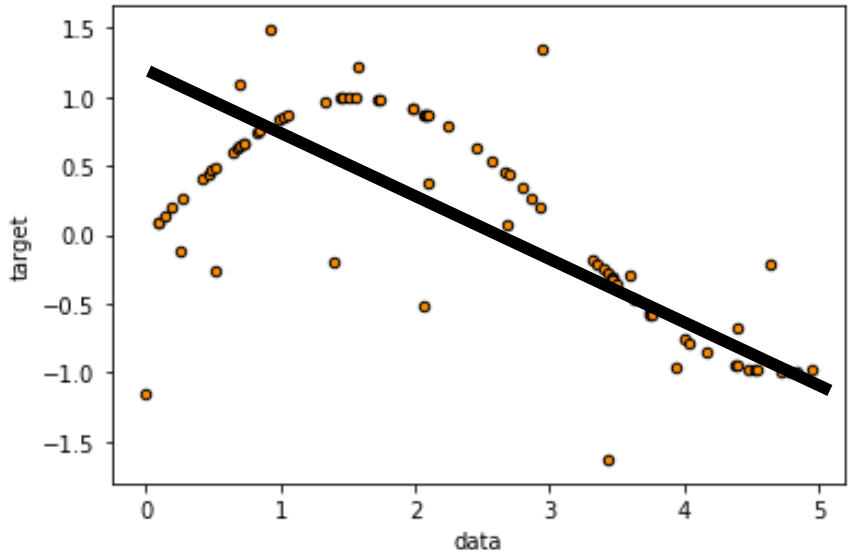


Предсказание стоимости квартиры

$$a(x) = w_0 + w_1 * (\text{площадь})$$

$$+ w_2 * (\text{район})$$

$$+ w_3 * (\text{расстояние до метро})$$

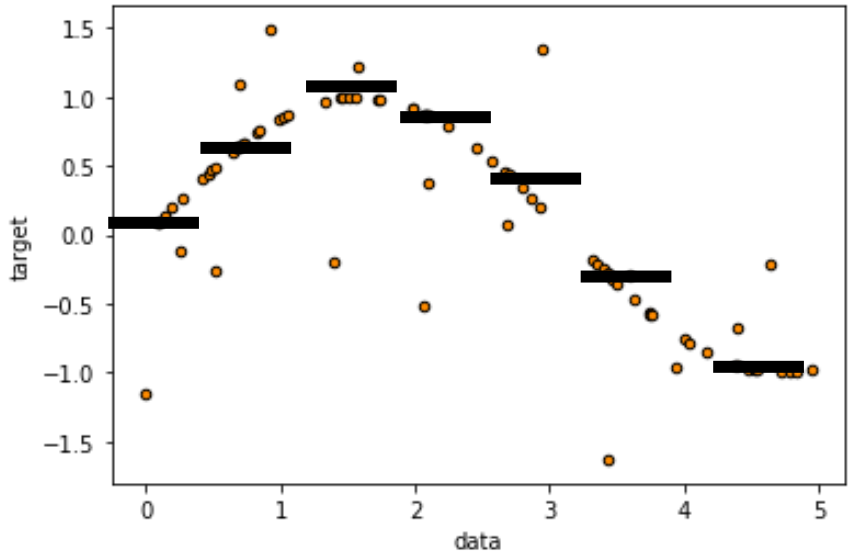


Предсказание стоимости квартиры

$$a(x) = w_0 + w_1 * (\text{площадь})$$

$$+ w_2 * (\text{район})$$

$$+ w_3 * (\text{расстояние до метро})$$

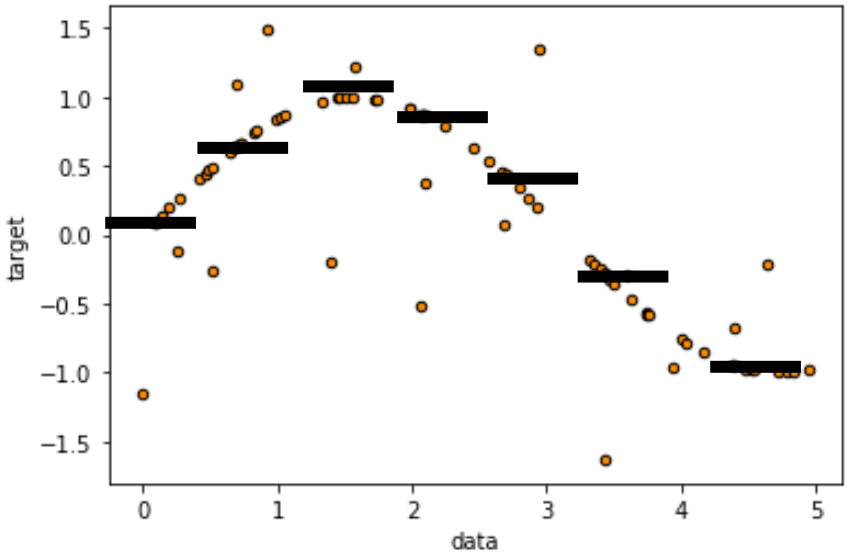


Предсказание стоимости квартиры

$$a(x) = w_0 + w_1 * (\text{площадь})$$

$$+ w_2 * (\text{район})$$

$$+ w_3 * [t_0 \leq x_3 < t_1] + \cdots + w_{3+n}[t_{n-1} \leq x_3 < t_n]$$



Нелинейные признаки

- Линейная модель с полиномиальными признаками:

$$a(x) = w_0 + w_1 * (\text{площадь}) + w_2 * (\text{этаж})$$
$$+w_3 * (\text{расстояние до метро}) + w_4 * (\text{площадь})^2$$
$$+w_5 * (\text{этаж})^2 + w_6 * (\text{расстояние до метро})^2$$
$$+w_7 * (\text{площадь}) * (\text{этаж}) + \cdots$$

Линейные модели

- Модель линейной регрессии хороша, если признаки сделаны специально под неё
- Пример: one-hot кодирование категориальных признаков или бинаризация числовых признаков

Линейная регрессия в векторном виде

Модель линейной регрессии

$$a(x) = \langle w, x \rangle$$

- Среднеквадратичная ошибка и задача обучения:

$$\frac{1}{\ell} \sum_{i=1}^{\ell} (\langle w, x_i \rangle - y_i)^2 \to \min_w$$

Матрицы

- Матрица — таблица с числами (для простоты)
- Матрица «объекты-признаки»:

$$X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{\ell 1} & x_{\ell 2} & \cdots & x_{\ell d} \end{pmatrix} \in \mathbb{R}^{\ell \times d}$$

Матрицы

- Матрица — таблица с числами (для простоты)
- Матрица «объекты-признаки»:

объект и его признаки

$$\begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{\ell 1} & x_{\ell 2} & \cdots & x_{\ell d} \end{pmatrix}$$

Матрицы

- Матрица — таблица с числами (для простоты)
- Матрица «объекты-признаки»:

$$\begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{\ell 1} & x_{\ell 2} & \cdots & x_{\ell d} \end{pmatrix}$$

значения признака на всех объектах

Векторы

- Вектор размера d — тоже матрица
- Вектор-строка: $w = (w_1, \dots, w_d) \in \mathbb{R}^{1 \times d}$
- Вектор-столбец: $w = \begin{pmatrix} w_1 \\ \dots \\ w_d \end{pmatrix} \in \mathbb{R}^{d \times 1}$

Матричное умножение

- Только для матриц $A \in \mathbb{R}^{m \times k}$ и $B \in \mathbb{R}^{k \times n}$
- Результат: $AB = C \in \mathbb{R}^{m \times n}$
- Правило:

$$c_{ij} = \sum_{p=1}^{k} a_{ip} b_{pj}$$

Пример

$$\begin{pmatrix} 1 & 2 \\ 0 & 1 \\ 10 & 0 \end{pmatrix} * \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 2 \end{pmatrix} = \begin{pmatrix} & & \\ & & \\ & & \end{pmatrix}$$

Пример

$$\begin{pmatrix} 1 & 2 \\ 0 & 1 \\ 10 & 0 \end{pmatrix} * \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 2 \end{pmatrix} = \begin{pmatrix} 1 & & \\ & & \\ & & \end{pmatrix}$$

Пример

$$\begin{pmatrix} 1 & 2 \\ 0 & 1 \\ 10 & 0 \end{pmatrix} * \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 2 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ & \end{pmatrix}$$

Пример

$$\begin{pmatrix} 1 & 2 \\ 0 & 1 \\ 10 & 0 \end{pmatrix} * \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 2 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 4 \end{pmatrix}$$

Пример

$$\begin{pmatrix} 1 & 2 \\ 0 & 1 \\ 10 & 0 \end{pmatrix} * \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 2 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 4 \\ 0 & & \end{pmatrix}$$

Применение линейной модели

- $a(x) = \langle w, x \rangle = w_1 x_1 + \cdots + w_d x_d$
- Как применить модель к обучающей выборке?

$$\begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{\ell 1} & x_{\ell 2} & \cdots & x_{\ell d} \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_d \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{d} w_i x_{1i} \\ \sum_{i=1}^{d} w_i x_{2i} \\ \vdots \\ \sum_{i=1}^{d} w_i x_{\ell i} \end{pmatrix}$$

Модель линейной регрессии

- Среднеквадратичная ошибка и задача обучения:

$$\frac{1}{\ell}\sum_{i=1}^{\ell}(\langle w, x_i\rangle - y_i)^2 \to \min_w$$

Вычисление ошибки

- Отклонения прогнозов от ответов:

$$Xw - y = \begin{pmatrix} \langle w, x_1 \rangle - y_1 \\ \vdots \\ \langle w, x_\ell \rangle - y_\ell \end{pmatrix}$$

Вычисление ошибки

- Евклидова норма:

$$\|z\| = \sqrt{\sum_{j=1}^{n} z_j^2}$$

$$\|z\|^2 = \sum_{j=1}^{n} z_j^2$$

Вычисление ошибки

- Отклонения прогнозов от ответов:

$$Xw - y = \begin{pmatrix} \langle w, x_1 \rangle - y_1 \\ \vdots \\ \langle w, x_\ell \rangle - y_\ell \end{pmatrix}$$

- Среднеквадратичная ошибка:

$$\frac{1}{\ell} \|Xw - y\|^2 = \frac{1}{\ell} \sum_{i=1}^{\ell} (\langle w, x_i \rangle - y_i)^2$$

Обучение линейной регрессии

$$\frac{1}{\ell}\|Xw - y\|^2 \to \min_w$$

- Вычисление MSE в NumPy:

```
np.square(X.dot(w) - y).mean()
```

Обучение линейной регрессии

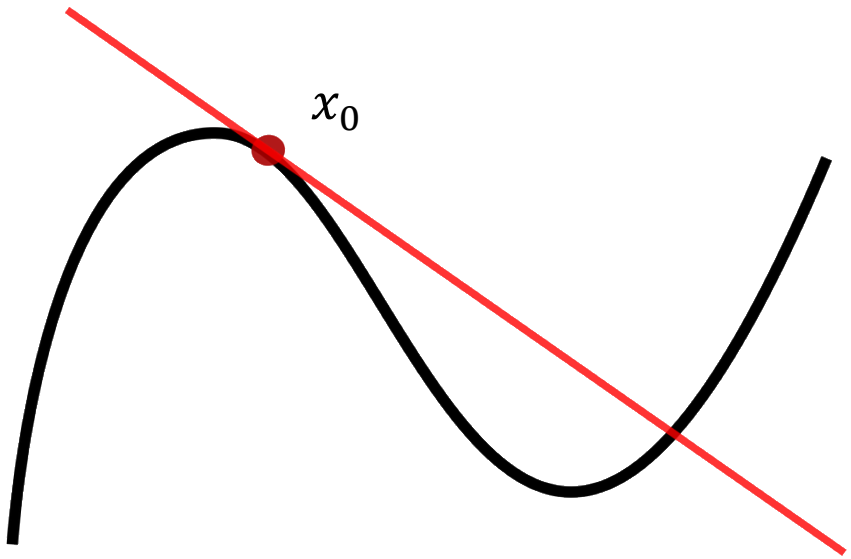
Среднеквадратичная ошибка

- MSE для линейной регрессии:

$$Q(w_1, \dots, w_d) = \frac{1}{\ell} \sum_{i=1}^{\ell} (w_1 x_1 + \cdots + w_d x_d - y_i)^2$$

Производная

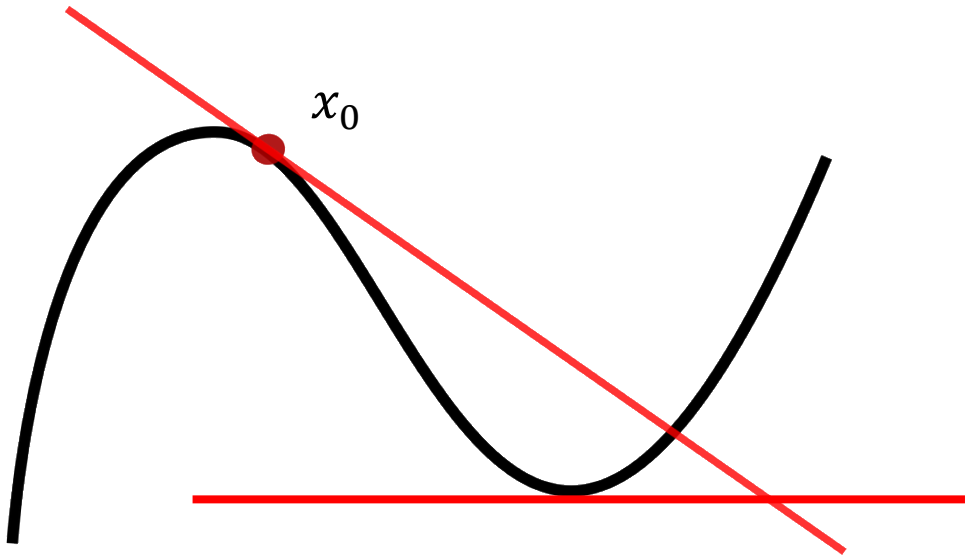
$$\lim_{x \to x_0} \frac{f(x) - f(x_0)}{x - x_0} = f'(x_0)$$



Производная

- Если точка x_0 — экстремум и в ней существует производная, то

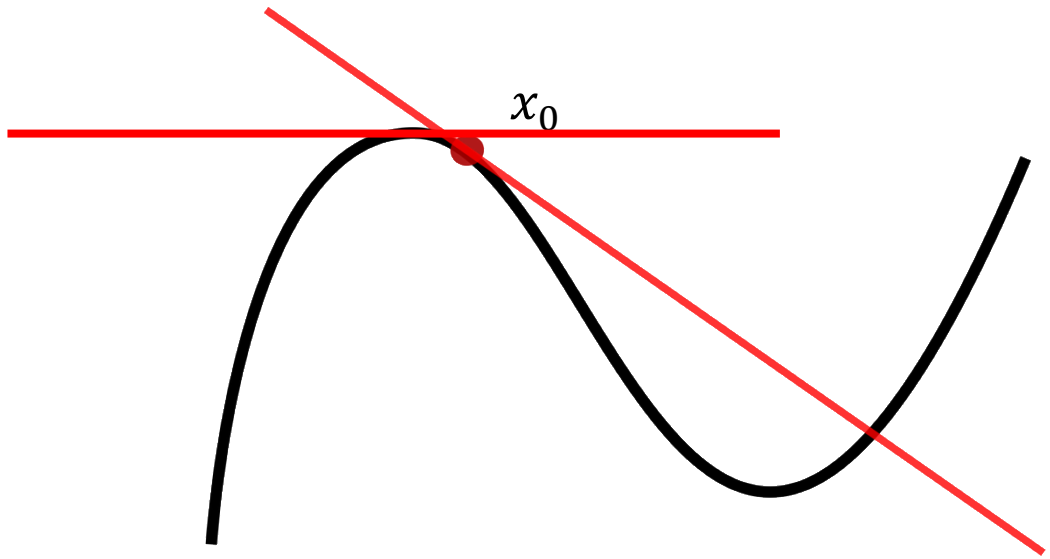
$$f'(x_0) = 0$$



Производная

- Если точка x_0 — экстремум и в ней существует производная, то

$$f'(x_0) = 0$$



Градиент

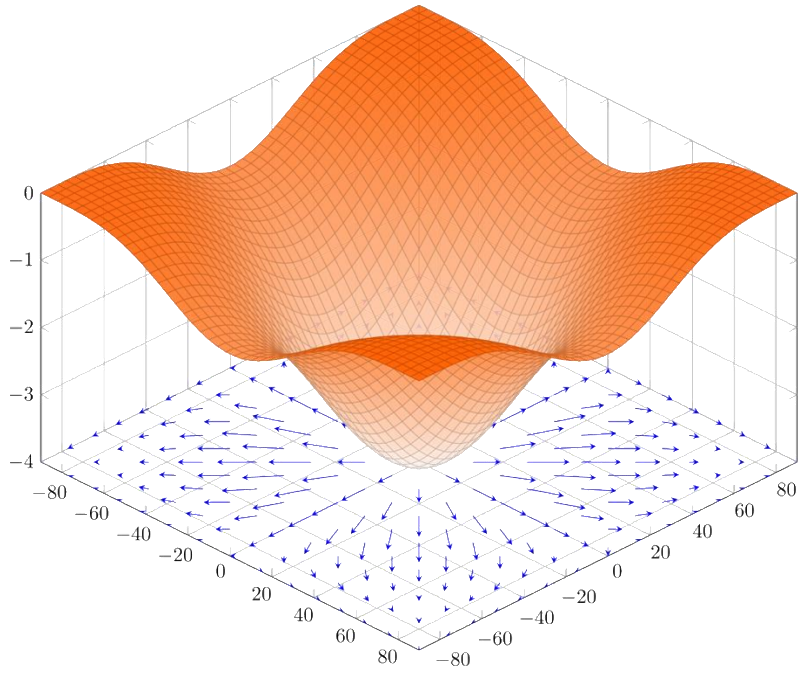
- Градиент — вектор частных производных

$$\nabla f(x) = \left(\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_d} \right)$$

- У градиента есть важное свойство!

Важное свойство

- Зафиксируем точку x_0
- В какую сторону функция быстрее всего растёт?



Важное свойство

- Зафиксируем точку x_0
- В какую сторону функция быстрее всего растёт?
- В направлении градиента!
- Если градиент равен нулю, то это экстремум

Условие экстремума

- Если точка x_0 — экстремум и в ней существует производная, то

$$\nabla f(x_0) = 0$$

Условие экстремума

- Если точка x_0 — экстремум и в ней существует производная, то

$$\nabla f(x_0) = 0$$

- Если функция выпуклая, то экстремум один
- MSE для линейной регрессии — выпуклая!
  - (при некоторых условиях)

Обучение линейной регрессии

- Можно посчитать градиент MSE:

$$\nabla \frac{1}{\ell} \|Xw - y\|^2 = \frac{2}{\ell} X^T (Xw - y)$$

- Приравниваем нулю и решаем систему линейных уравнений:

$$w = (X^T X)^{-1} X^T y$$

Аналитическое решение

$$w = (X^T X)^{-1} X^T y$$

- Если матрица $X^T X$ вырожденная, то будут проблемы
- Даже если она почти вырожденная, всё равно будут проблемы
- Если признаков много, то придётся долго ждать

ЛИНЕЙНАЯ РЕГРЕССИЯ

Линейная регрессия:

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

Обучение линейной регрессии – минимизация среднеквадратичной ошибки:

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

(здесь l – количество объектов)

ПОЧЕМУ MSE?

ВЕРОЯТНОСТНАЯ ПОСТАНОВКА

$$y = (w, x) + \varepsilon$$

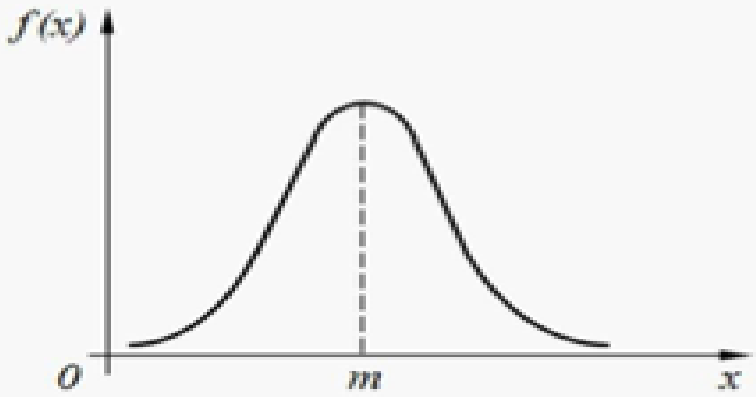
- Шум в данных обычно имеет некоторое распределение. В большинстве реальных задач считается, что

$$\varepsilon \sim N(0, \sigma^2).$$

- Отсюда получаем, что

$$y \sim N\big((w, x), \sigma^2\big).$$

График плотности нормального распределения



ВЕРОЯТНОСТНАЯ ПОСТАНОВКА

$$y \sim N((w, x), \sigma^2)$$

Это означает, что вероятность наблюдать y при данных значениях x равна

$$p(y|x, w) \sim N((w, x), \sigma^2)$$

Мы хотим подобрать оптимальные веса. Что это такое?

Мы хотим подобрать такой вектор w, что вероятность наблюдать некоторое значение y при наблюдаемых x максимальна.

МЕТОД МАКСИМУМА ПРАВДОПОДОБИЯ

Мы хотим подобрать оптимальные веса. Что это такое?

Мы хотим подобрать такой вектор w, что вероятность наблюдать некоторое значение y при наблюдаемых x максимальна.

Запишем это желание сразу для всех объектов выборки (в предположении, что объекты независимы):

$$p(\boldsymbol{y}|X,w) = p(y_1|x_1,w) \cdot p(y_2|x_2,w) \cdot \ldots \cdot p(y_i|x_i,w) \cdot \ldots \to \max_w$$

Величина $p(y|X,w)$ называется ***функцией правдоподобия (или правдоподобием) выборки***.

ММП ДЛЯ ЛИНЕЙНОЙ РЕГРЕССИИ

Модель данных с некоррелированным гауссовским шумом:

$$y_i = (w, x_i) + \varepsilon_i, \varepsilon_i \sim N(0, \sigma^2), i = 1, \dots, l$$

Тогда $y_i \sim N((w, x_i), \sigma^2), i = 1, \dots, l$

Метод максимума правдоподобия (ММП):

$$L(y_1, \dots, y_l | w) = \prod_{i=1}^{l} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2\sigma^2}(y_i - (w, x_i))^2\right) \to \max_w$$

$$-ln\ L(y_1, \dots, y_l | w) = const + \frac{1}{2\sigma^2}\sum_{i=1}^{l}(y_i - (w, x_i))^2 \to \min_w$$

В данном случае ММП совпадает с МНК.

АНАЛИТИЧЕСКОЕ РЕШЕНИЕ ЗАДАЧИ МЕТОДА НАИМЕНЬШИХ КВАДРАТОВ (МНК)

Задача обучения линейной регрессии (в матричной форме):

$$\frac{1}{\ell}\|Xw - y\|^2 \to \min_w$$

Точное (аналитическое) решение:

$$\boldsymbol{w = (X^T X)^{-1} X^T y}$$

НЕДОСТАТКИ АНАЛИТИЧЕСКОЙ ФОРМУЛЫ

- Обращение матрицы – сложная операция ($O(N^3)$ от числа признаков)
- Матрица $X^T X$ может быть вырожденной или плохо обусловленной
- Если заменить среднеквадратичный функционал ошибки на другой, то скорее всего не найдем аналитическое решение

ГРАДИЕНТНЫЙ СПУСК

МЕТОД ГРАДИЕНТНОГО СПУСКА

- Наша задача при обучении модели – найти такие веса w, на которых достигается минимум функции ошибки.
- В простейшем случае, если ошибка среднеквадратичная, то её график – это парабола.
- Идея метода градиентного спуска:

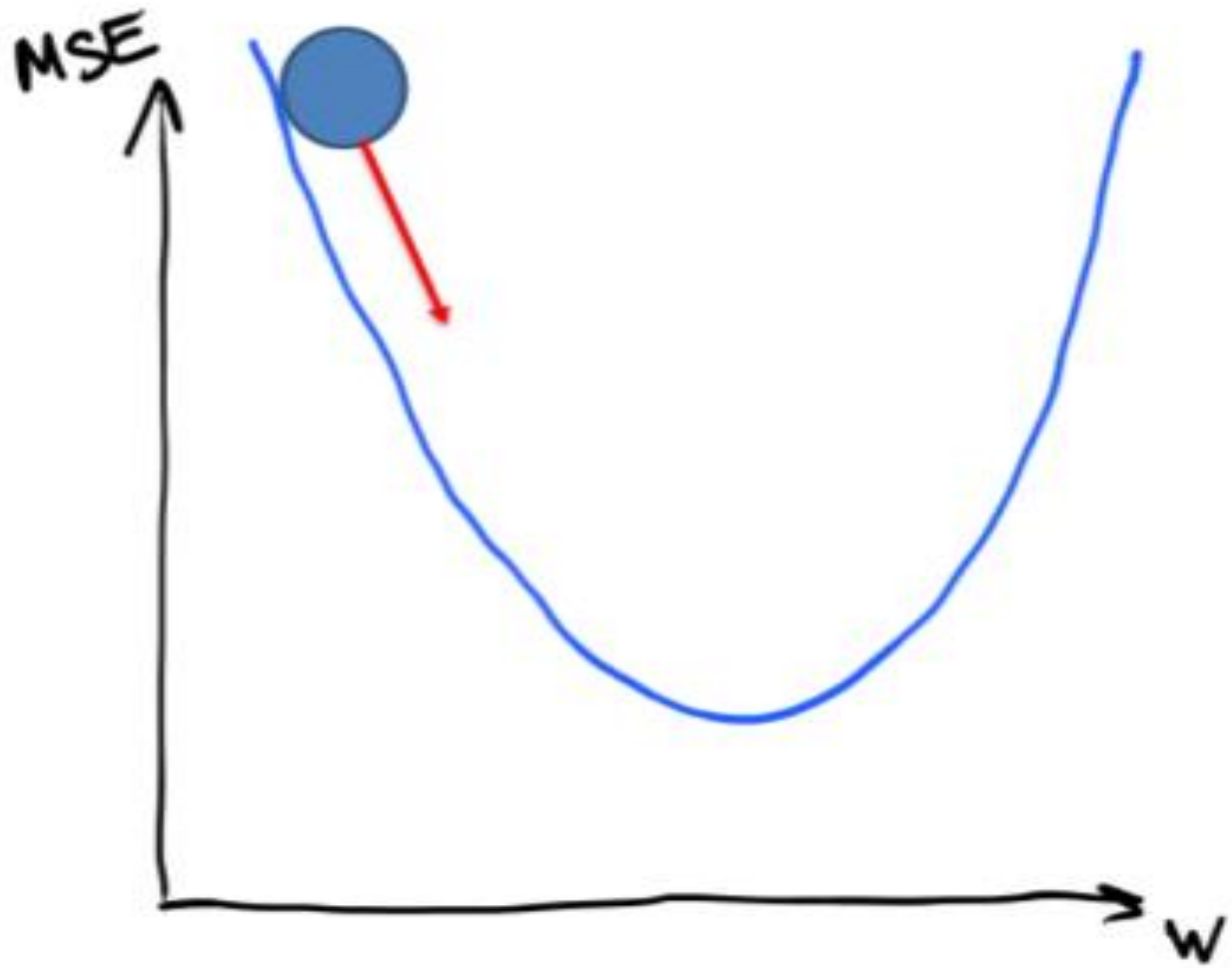
На каждом шаге (на каждой итерации метода) движемся в сторону антиградиента функции потерь!

То есть на каждом шаге движемся в направлении уменьшения ошибки.

Вектор градиента функции потерь обозначают $\mathbf{grad\ Q}$ *или* ∇Q.

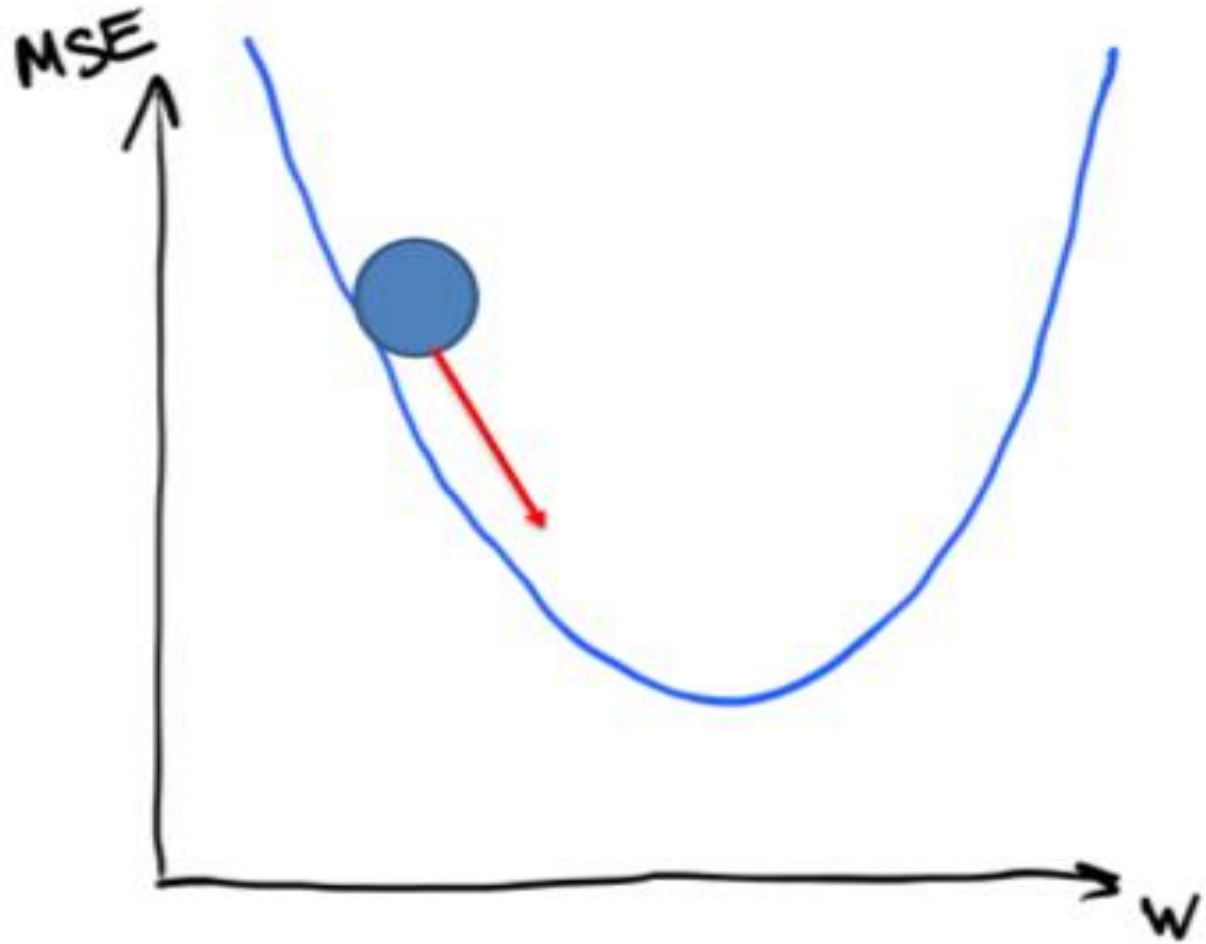
МЕТОД ГРАДИЕНТНОГО СПУСКА

На каждом шаге (на каждой итерации метода) движемся в сторону антиградиента функции потерь!



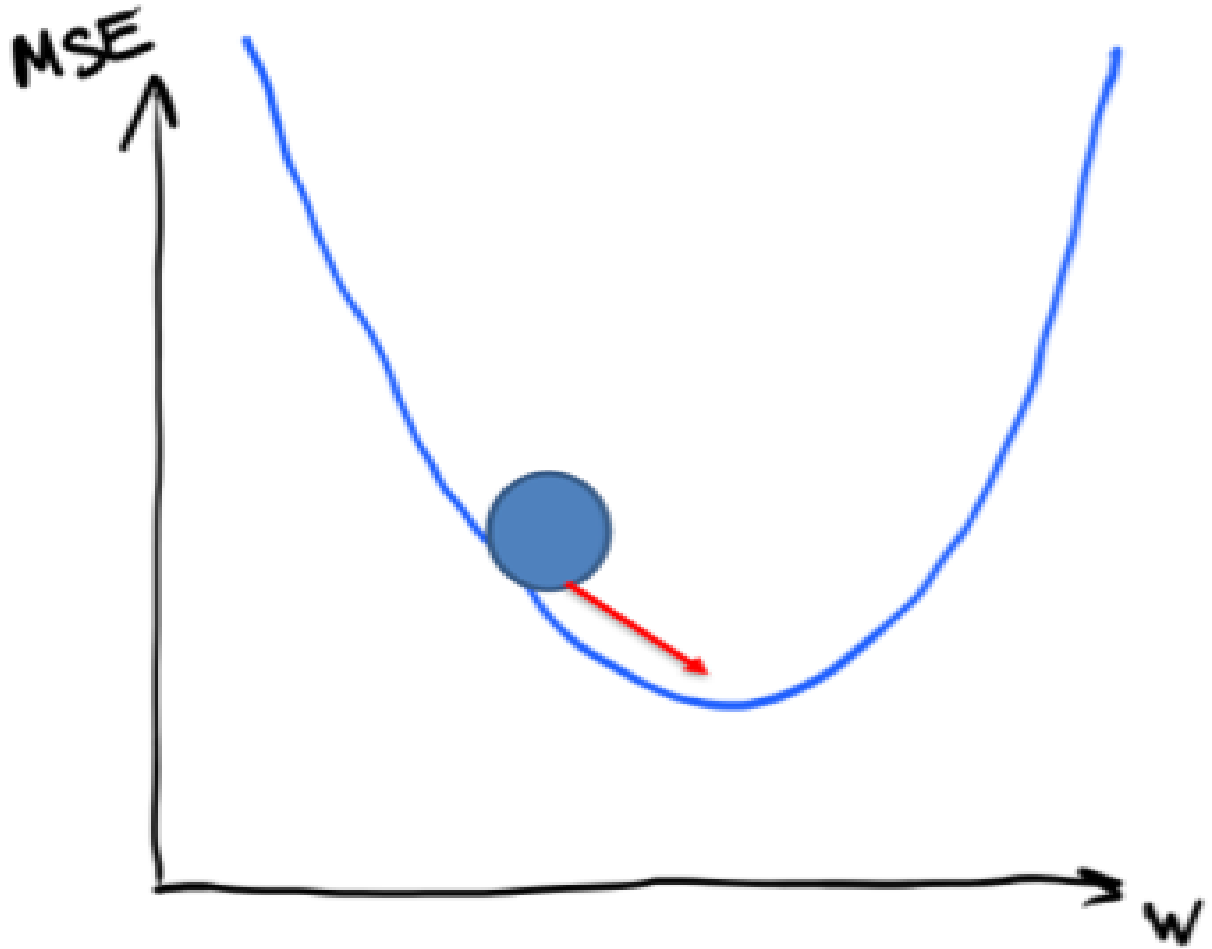
МЕТОД ГРАДИЕНТНОГО СПУСКА

На каждом шаге (на каждой итерации метода) движемся в сторону антиградиента функции потерь!



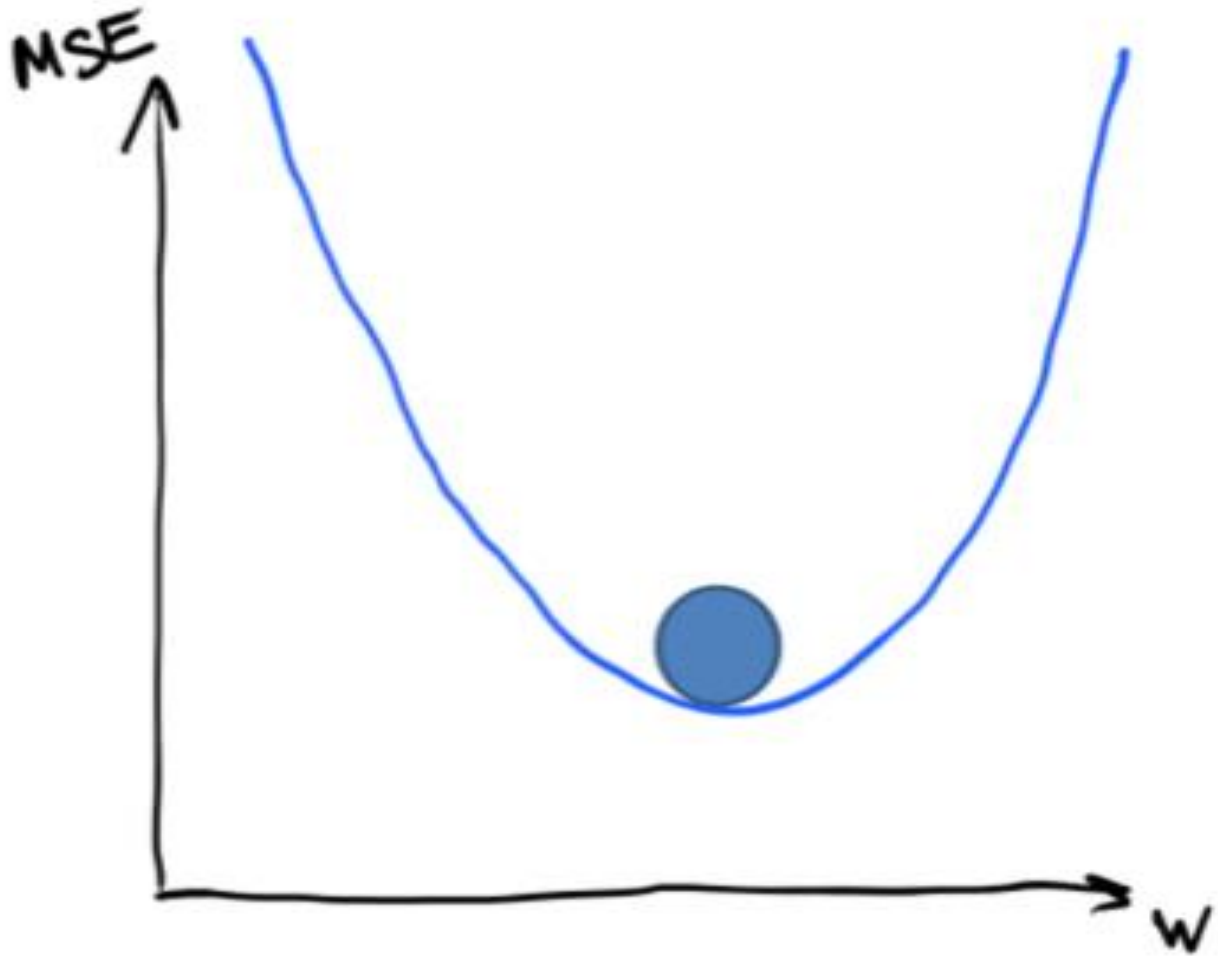
МЕТОД ГРАДИЕНТНОГО СПУСКА

На каждом шаге (на каждой итерации метода) движемся в сторону антиградиента функции потерь!



МЕТОД ГРАДИЕНТНОГО СПУСКА

На каждом шаге (на каждой итерации метода) движемся в сторону антиградиента функции потерь!



МЕТОД ГРАДИЕНТНОГО СПУСКА

Метод градиентного спуска (одномерный случай):

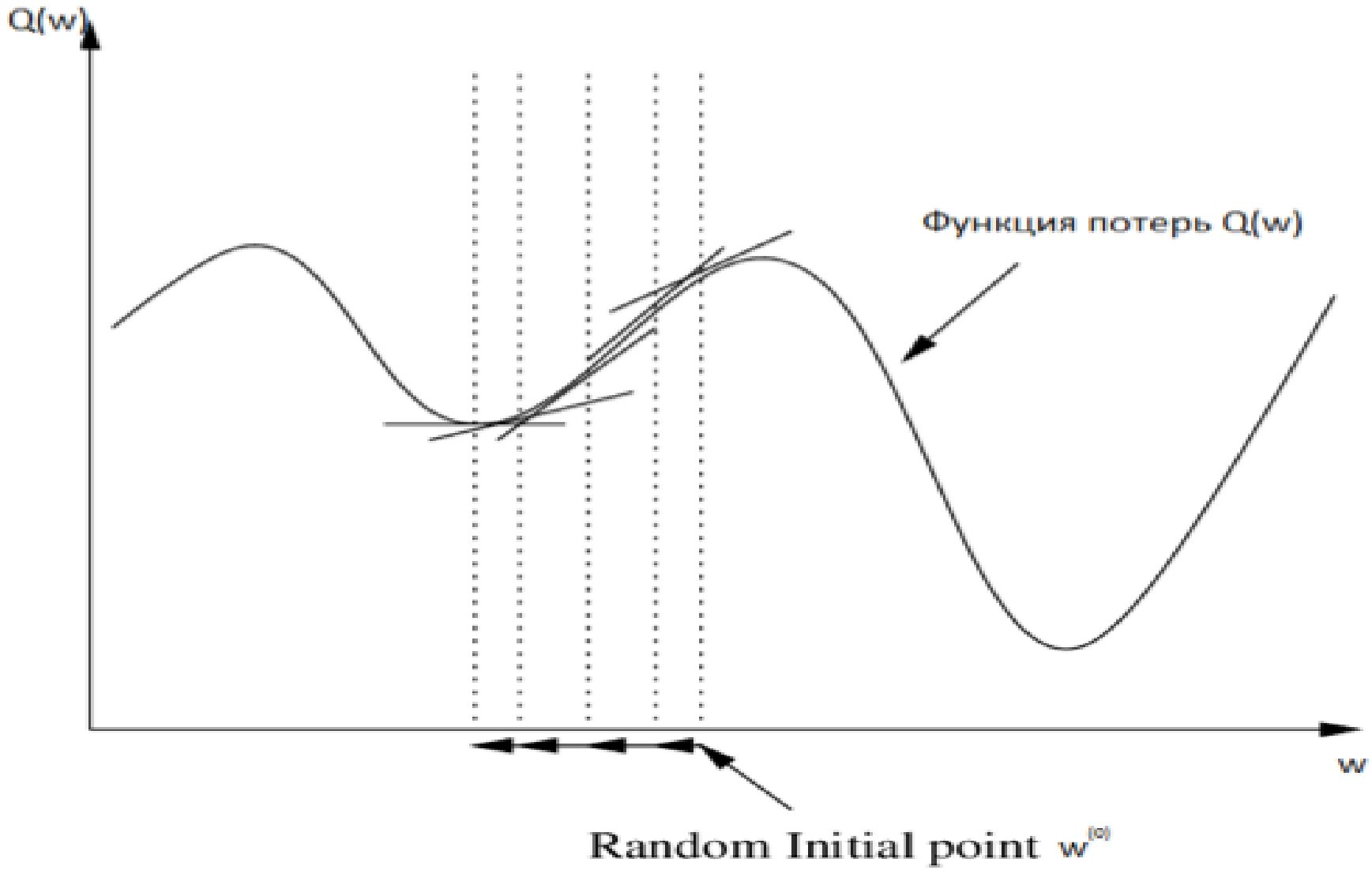
Пусть у нас только один вес - w.

Тогда при добавлении к весу w слагаемого $-\frac{\partial Q}{\partial w}$ функция $Q(w)$ убывает.

- Инициализируем вес $w^{(0)}$.
- На каждом следующем шаге обновляем вес, добавляя $-\frac{\partial Q}{\partial w}(w^{(k-1)})$:

$$\boldsymbol{w^{(k)} = w^{(k-1)} - \frac{\partial Q}{\partial w}(w^{(k-1)})}$$

МЕТОД ГРАДИЕНТНОГО СПУСКА



МЕТОД ГРАДИЕНТНОГО СПУСКА

Метод градиентного спуска (общий случай случай):

Пусть $w_0, w_1, \ldots, w_n$ - веса, которые мы ищем.

Тогда $\nabla Q(w) = \{\frac{\partial Q}{\partial w_0}, \frac{\partial Q}{\partial w_1}, \ldots, \frac{\partial Q}{\partial w_n}\}$

МЕТОД ГРАДИЕНТНОГО СПУСКА

Формулу для обновления весов можно записать в векторном виде:

- Инициализируем веса $\boldsymbol{w^{(0)}}$.
- На каждом следующем шаге обновляем веса по формуле:

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

МЕТОД ГРАДИЕНТНОГО СПУСКА

Формулу для обновления весов можно записать в векторном виде:

- Инициализируем веса $\boldsymbol{w^{(0)}}$.
- На каждом следующем шаге обновляем веса по формуле:

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

В формулу обычно добавляют параметр $\boldsymbol{\eta}$ – величина градиентного шага (learning rate). Он отвечает за скорость движения в сторону антиградиента:

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

Если функция $Q(w)$ выпуклая и гладкая, а также имеет минимум в точке w^*, то метод градиентного спуска при аккуратно подобранном η через некоторое число шагов гарантированно попадет в малую окрестность точки w^*.

ВАРИАНТЫ ИНИЦИАЛИЗАЦИИ ВЕСОВ

- $w_j = 0, j = 1, \ldots, n$
- Небольшие случайные значения:

$$w_j := random(-\varepsilon, \varepsilon)$$

- Обучение по небольшой случайной подвыборке объектов
- Мультистарт: многократный запуск из разных случайных начальных приближений и выбор лучшего решения

КРИТЕРИИ ОСТАНОВА

- $|Q(w^{(k)}) - Q(w^{(k-1)})| < \varepsilon$
- $||w^{(k)} - w^{(k-1)}|| < \varepsilon$
- $||\nabla Q(w^{(k)})|| < \varepsilon$

ГРАДИЕНТНЫЙ ШАГ

В общем случае градиентный шаг может зависеть от номера итерации, тогда будем писать не η, а η_k.

- $\eta_k = c$
- $\eta_k = \frac{1}{k}$
- $\eta_k = \lambda \left(\frac{s_0}{s_0+k}\right)^p$, λ, s_0, p - параметры

ОДИН ИЗ НЕДОСТАТКОВ ГРАДИЕНТНОГО СПУСКА

(с точки зрения реализации)

- На каждом шаге для вычисления $\nabla Q(w)$ мы вычисляем производную по каждому весу от каждого объекта. То есть вычисляем целую матрицу производных — это затратно и по времени, и по памяти.

СТОХАСТИЧЕСКИЙ ГРАДИЕНТНЫЙ СПУСК

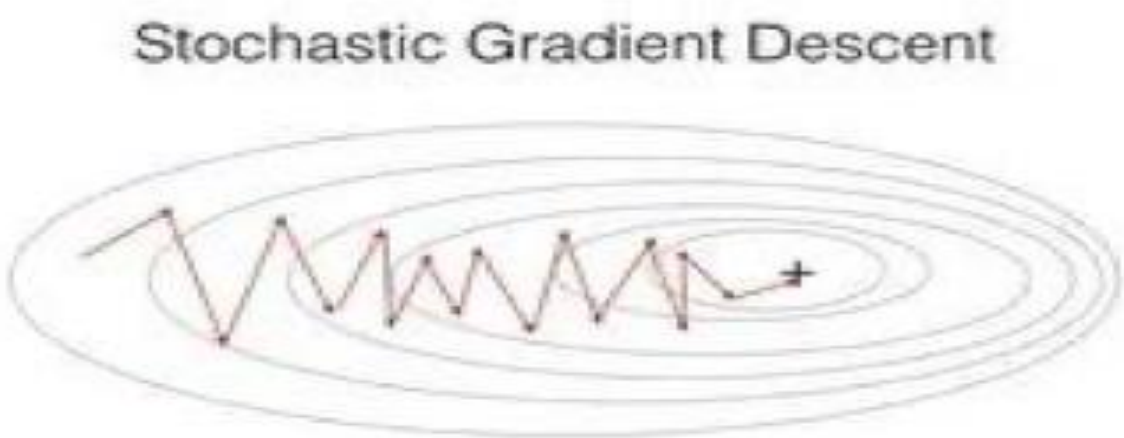
Stochastic gradient descent (SGD):

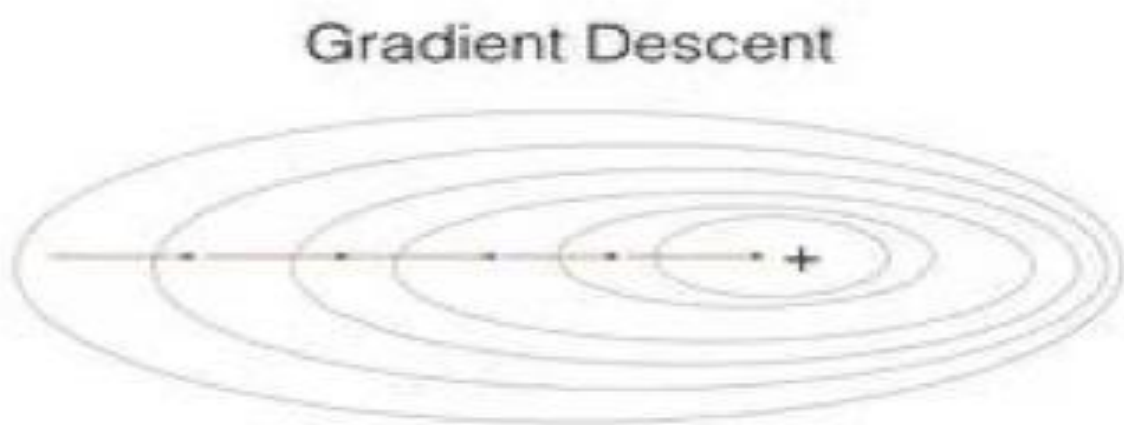
- на каждом шаге выбираем ***один случайный объект*** и сдвигаемся в сторону антиградиента по этому объекту:

$$w^{(k)} = w^{(k-1)} - \eta_k \cdot \nabla q_{i_k}\left(w^{(k-1)}\right),$$

где $\nabla q_{i_k}\left(w^{(k-1)}\right)$ - градиент функции потерь, вычисленный только по объекту с номером i_k (а не по всей обучающей выборке).

СТОХАСТИЧЕСКИЙ ГРАДИЕНТНЫЙ СПУСК





Если функция $Q(w)$ выпуклая и гладкая, а также имеет минимум в точке w^*, то метод стохастического градиентного спуска при аккуратно подобранном η через некоторое число шагов гарантированно попадет в малую окрестность точки w^*. Однако, сходится метод медленнее, чем обычный градиентный спуск

MINI-BATCH GRADIENT DESCENT

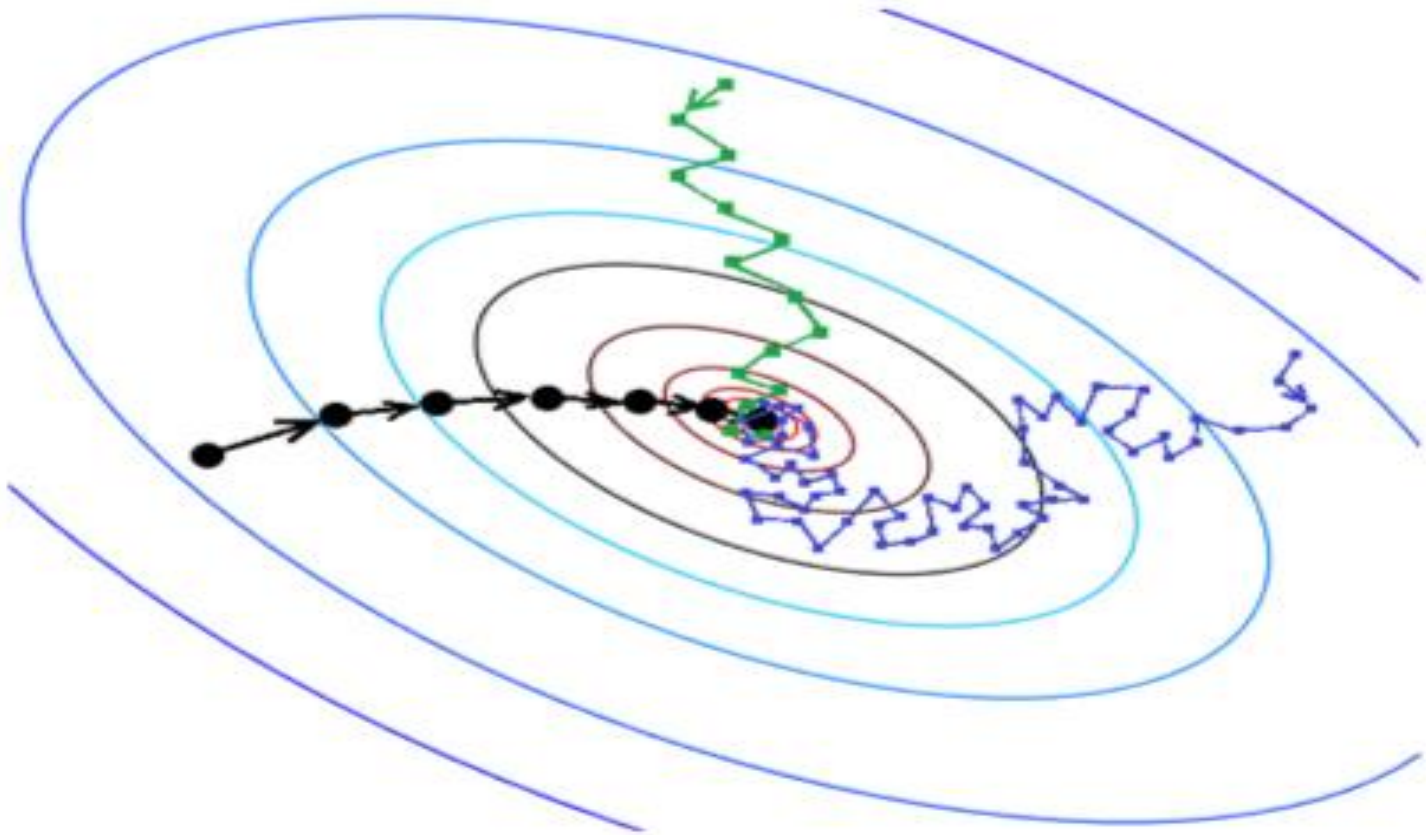
Промежуточное решение между классическим градиентным спуском и стохастическим вариантом.

- Выбираем batch size (например, 32, 64 и т.д.). Разбиваем все пары объект-ответ на группы размера batch size.
- На i-й итерации градиентного спуска вычисляем $\nabla Q(w)$ только по объектам i-го батча:

$$w^{(k)} = w^{(k-1)} - \eta_k \cdot \nabla Q_i\left(w^{(k-1)}\right),$$

где $\nabla Q_i\left(w^{(k-1)}\right)$ - градиент функции потерь, вычисленный по объектам из i-го батча.

ВАРИАНТЫ ГРАДИЕНТНОГО СПУСКА



Batch GD
- Slowest
- Perfect gradient

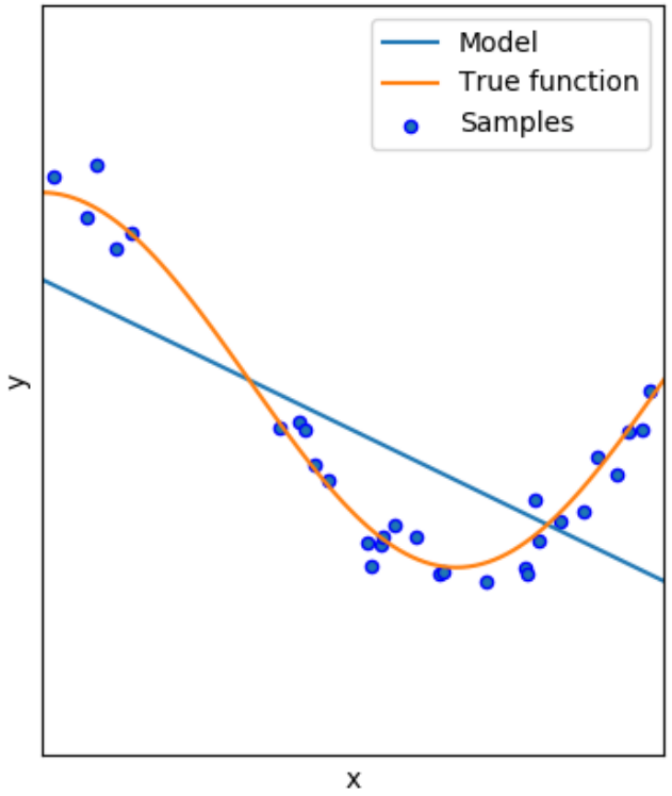
Stochastic GD
- Fastest
- Rough-estimate grad

Mini-batch GD
- Compromise

Переобучение и регуляризация линейных моделей

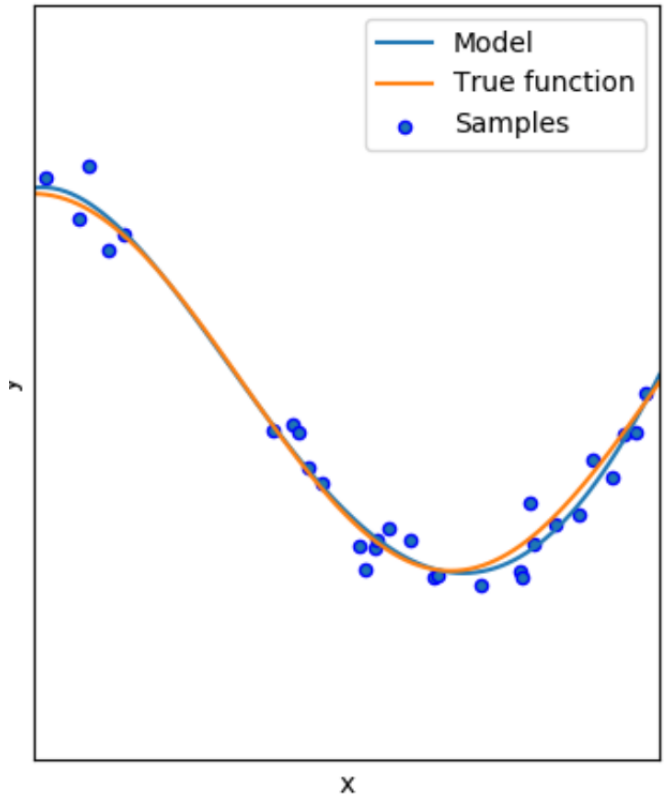
Нелинейная задача

$$a(x) = w_0 + w_1 x$$



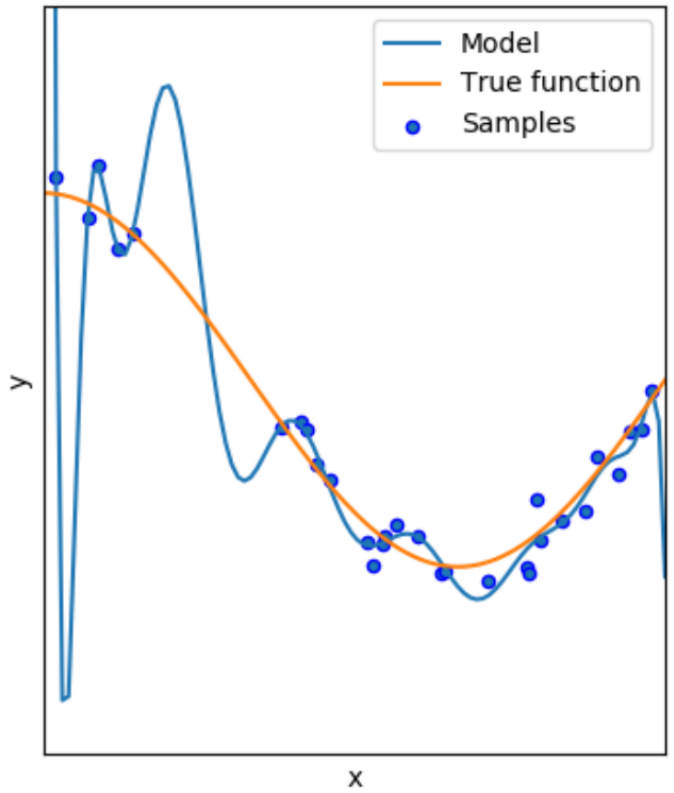
Нелинейная задача

$$a(x) = w_0 + w_1 x + w_2 x^2 + w_3 x^3 + w_4 x^4$$



Нелинейная задача

$$a(x) = w_0 + w_1 x + w_2 x^2 + w_3 x^3 + w_4 x^4 + \cdots + w_{15} x^{15}$$



Симптом переобучения

$$a(x) = 0.5 + 13458922x - 43983740x^2 + \cdots$$

- Большие коэффициенты — симптом переобучения
- Эмпирическое наблюдение

Симптом переобучения

- Большие коэффициенты в линейной модели — это плохо
- Пример: предсказание роста по весу

$$a(x) = 698x - 41714$$

- Изменение веса на 0.01 кг приведет к изменению роста на 7 см
- Не похоже не правильную зависимость

Регуляризация

- Будем штрафовать за большие веса!
- Пример функционала:

$$Q(a, X) = \frac{1}{\ell} \sum_{i=1}^{\ell} (\langle w, x_i \rangle - y_i)^2$$

- Регуляризатор:

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

Регуляризация

- Регуляризованный функционал

$$\frac{1}{\ell}\sum_{i=1}^{\ell}(\langle w, x_i\rangle - y_i)^2 + \lambda\|w\|^2 \to \min_w$$

- λ — коэффициент регуляризации

Регуляризация

- Регуляризованный функционал

$$\frac{1}{\ell}\sum_{i=1}^{\ell}(\langle w, x_i\rangle - y_i)^2 + \lambda\|w\|^2 \to \min_w$$

- Аналитическое решение:

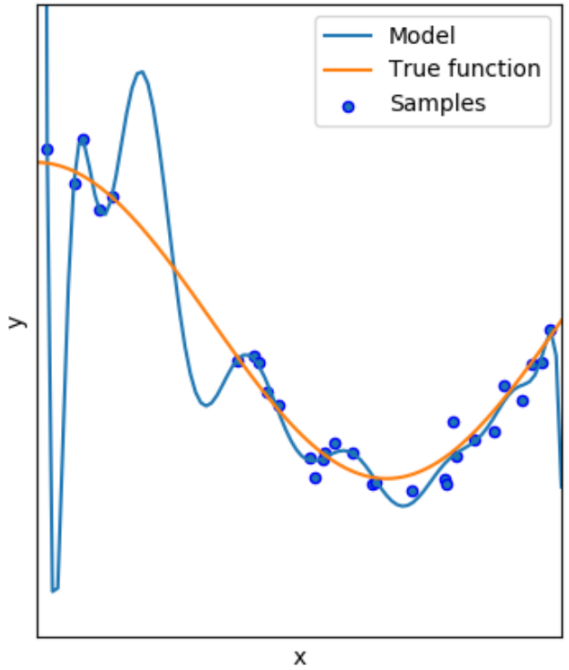
$$w = (X^T X + \lambda I)^{-1} X^T y$$

- Гребневая регрессия (Ridge regression)

Эффект регуляризации

$$a(x) = w_0 + w_1 x + w_2 x^2 + w_3 x^3 + w_4 x^4 + \cdots + w_{15} x^{15}$$

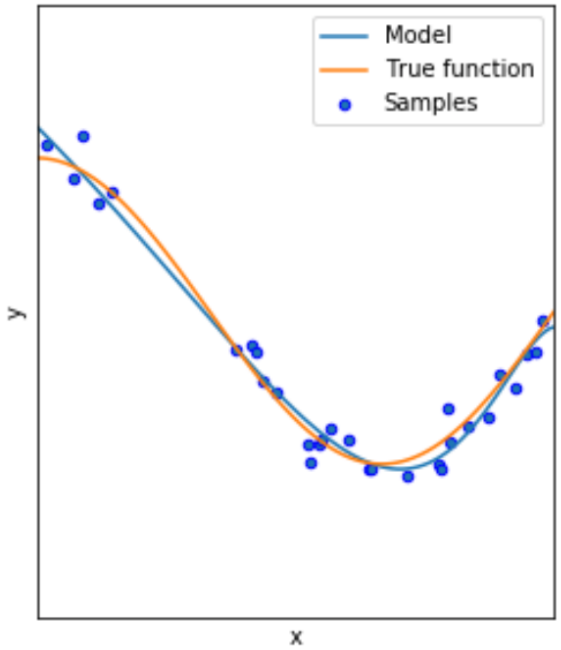
$$\frac{1}{\ell} \sum_{i=1}^{\ell} (a(x_i) - y_i)^2 \to \min_w$$



Эффект регуляризации

$$a(x) = w_0 + w_1 x + w_2 x^2 + w_3 x^3 + w_4 x^4 + \cdots + w_{15} x^{15}$$

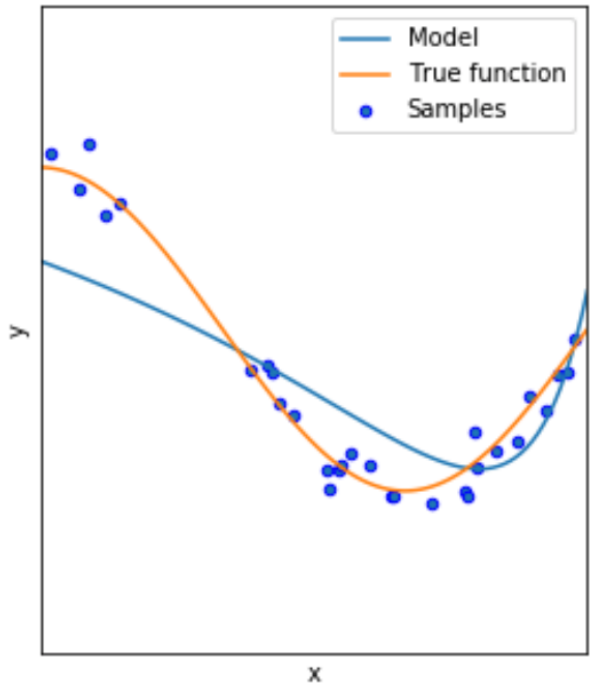
$$\frac{1}{\ell} \sum_{i=1}^{\ell} (a(x_i) - y_i)^2 + 0.01 \, \|w\|^2 \to \min_w$$



Эффект регуляризации

$$a(x) = w_0 + w_1 x + w_2 x^2 + w_3 x^3 + w_4 x^4 + \cdots + w_{15} x^{15}$$

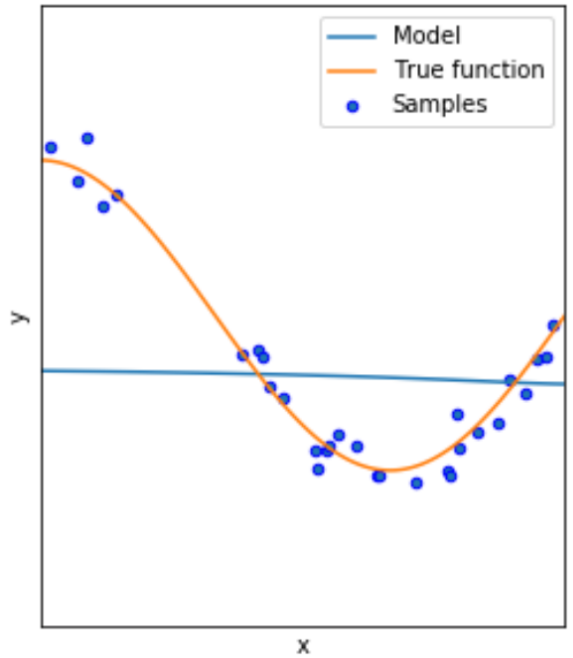
$$\frac{1}{\ell} \sum_{i=1}^{\ell} (a(x_i) - y_i)^2 + 1 \, \|w\|^2 \to \min_w$$



Эффект регуляризации

$$a(x) = w_0 + w_1 x + w_2 x^2 + w_3 x^3 + w_4 x^4 + \cdots + w_{15} x^{15}$$

$$\frac{1}{\ell} \sum_{i=1}^{\ell} (a(x_i) - y_i)^2 + 100 \, \|w\|^2 \to \min_w$$



Лассо

- Регуляризованный функционал

$$\frac{1}{\ell}\sum_{i=1}^{\ell}(\langle w, x_i\rangle - y_i)^2 + \lambda\sum_{j=1}^{d}|w_j| \to \min_w$$

- LASSO (Least Absolute Shrinkage and Selection Operator)
- Некоторые веса зануляются
- Приводит к отбору признаков

Регуляризаторы

- $\|z\|_2 = \sqrt{\sum_{j=1}^{d} z_j^2}$ — L_2-норма
- $\|z\|_1 = \sum_{j=1}^{d} |z_j|$ — L_1-норма

Интерпретация линейных моделей

Предсказание стоимости квартиры

$$a(x) = 100.000 * (\text{площадь})$$
$$+ 500.000 * (\text{число магазинов рядом})$$
$$+ 100 * (\text{средний доход жильцов дома})$$

Предсказание стоимости квартиры

$$a(x) = 100.000 * (\text{площадь})$$
$$+ 500.000 * (\text{число магазинов рядом})$$
$$+ 100 * (\text{средний доход жильцов дома})$$

- Чем больше вес, тем важнее признак?

Предсказание стоимости квартиры

$$a(x) = 100.000 * (\text{площадь в кв. м.})$$
$$+ 500.000 * (\text{число магазинов рядом})$$
$$+ 100 * (\text{средний доход жильцов дома})$$

- Чем больше вес, тем важнее признак?

Предсказание стоимости квартиры

$$a(x) = 10 * (\text{площадь в кв. см.})$$
$$+ 500.000 * (\text{число магазинов рядом})$$
$$+ 100 * (\text{средний доход жильцов дома})$$

- Чем больше вес, тем важнее признак?

Предсказание стоимости квартиры

$$a(x) = 100.000 * (\text{площадь в кв. м.})$$
$$+ 500.000 * (\text{число магазинов рядом})$$
$$+ 100 * (\text{средний доход жильцов дома})$$

- Чем больше вес, тем важнее признак?

Предсказание стоимости квартиры

$$a(x) = 100.000 * (\text{площадь в кв. м.})$$
$$+ 500.000 * (\text{число магазинов рядом})$$
$$+ 100 * (\text{средний доход жильцов дома})$$

- Чем больше вес, тем важнее признак?
- Только если признаки масштабированы!

Масштабирование признаков

- Отмасштабируем j-й признак
- Вычисляем среднее и стандартное отклонение признака на обучающей выборке:

$$\mu_j = \frac{1}{\ell}\sum_{i=1}^{\ell} x_i^j$$

$$\sigma_j = \sqrt{\frac{1}{\ell}\sum_{i=1}^{\ell}\left(x_i^j - \mu_j\right)^2}$$

Масштабирование признаков

- Вычтем из каждого значения признака среднее и поделим на стандартное отклонение:

$$x_i^j := \frac{x_i^j - \mu_j}{\sigma_j}$$

Регуляризация

- Если модель переобучается, то веса используются для запоминания обучающей выборки
- Правильнее масштабировать признаки и регуляризовать модель перед изучением весов

Пример

- 1000 объектов
- Два признака
- Первый принимает значения от 0 до 1
- Второй равен единице на 10 объектах и нулю на 990 объектах
- $y = x_1 + 2x_2$

Пример

```
[0.3175037 , 1.        ],
[0.59558502, 1.        ],
[0.48660609, 1.        ],
[0.69255463, 1.        ],
[0.81968981, 1.        ],
[0.48844247, 1.        ],
[0.13426702, 1.        ],
[0.850628  , 1.        ],
[0.57499033, 1.        ],
[0.73993748, 1.        ],
[0.70466465, 0.        ],
[0.96821177, 0.        ],
[0.29530732, 0.        ],
[0.70530677, 0.        ],
[0.36567633, 0.        ],
[0.39541072, 0.        ],
[0.23059464, 0.        ],
[0.34401018, 0.        ],
[0.94829675, 0.        ],
[0.29257085, 0.        ],
[0.24599061, 0.        ],
[0.58313798, 0.        ],
```

Пример

$$a(x) = x_1 + 2x_2$$

- Удаляем первый признак, получаем MSE = 0.08
- Удаляем второй признак, получаем MSE = 0.04

- Правильнее удалить признак и посмотреть, как сильно растёт ошибка без него