Машинное обучение

Лекция 4
Линейная регрессия

Андрей Нарцев
andrei.nartsev@gmail.com
anartsev@hse.ru

НИУ ВШЭ, 2025

Организационное

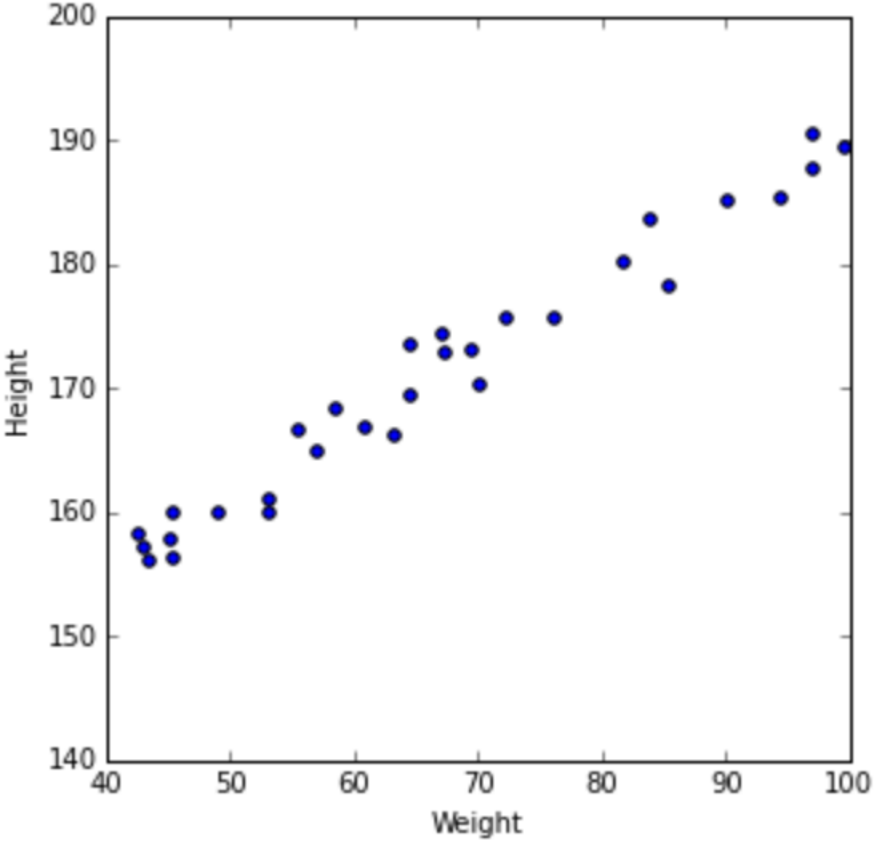
- Скоро появится первое домашнее задание – будет объявление в канале
- Будет дополнительная лекция – день и время выберем голосованием

План лекции

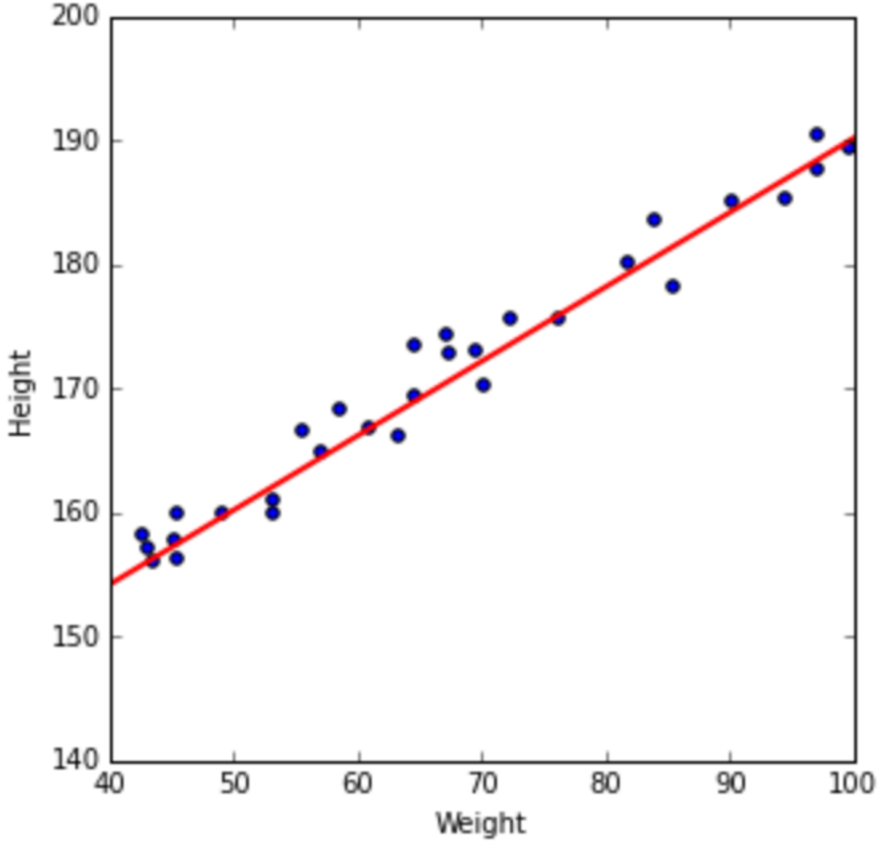
- Основные понятия
- Особенности применения
- Обоснование минимизации среднеквадратичной ошибки
- Аналитическое решение в матричном виде
- Градиентный спуск

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

Много признаков

- Общий случай: d признаков
- Модель

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

- Количество параметров: $d + 1$

Много признаков

Запишем через скалярное произведение:

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

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

Будем считать, что есть признак, всегда равный единице:

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

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

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

Применимость линейной регрессии

Модель линейной регрессии

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

- 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

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

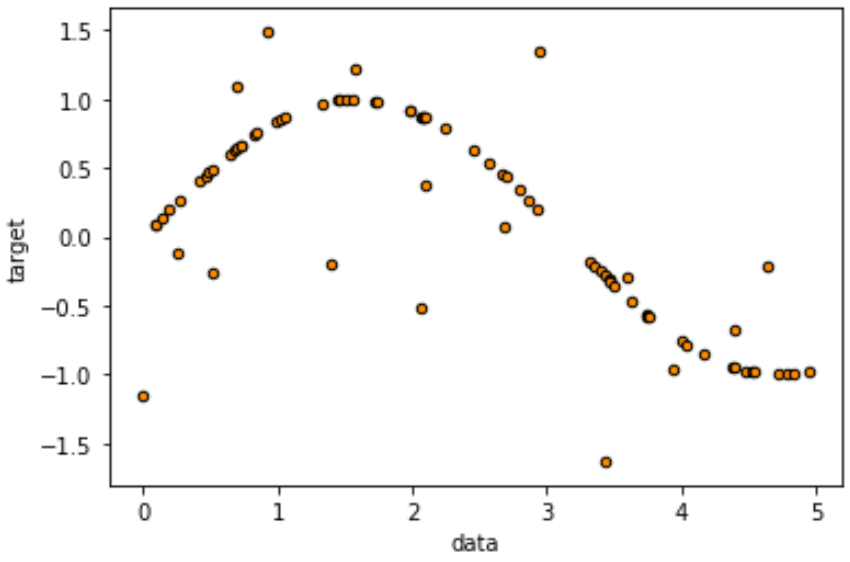
$$+ w_2 * (\text{район})$$

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

- 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

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

$$+ w_2 * (\text{квартира в ЦАО?})$$

$$+ w_3 * (\text{квартира в ЮАО?})$$

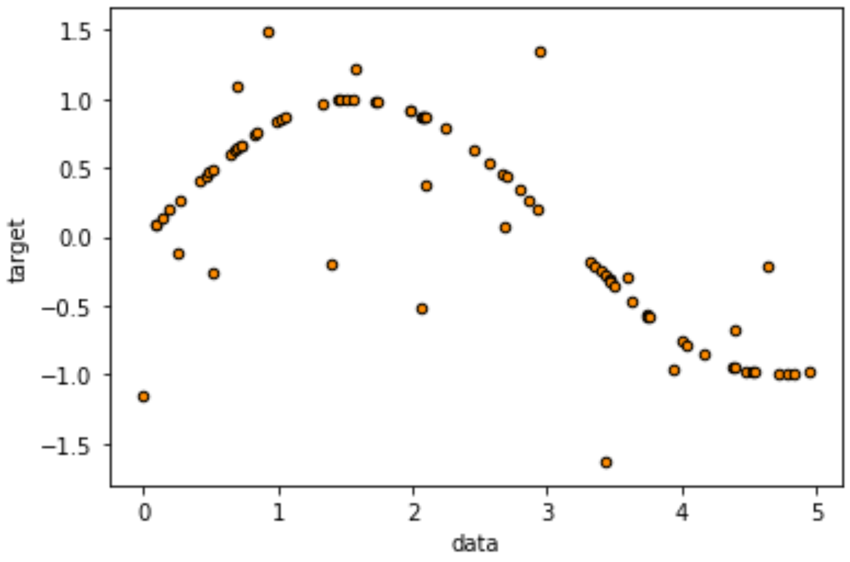
$$+ w_4 * (\text{квартира в САО?})$$

Предсказание стоимости квартиры

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

$$+ w_2 * (\text{район})$$

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

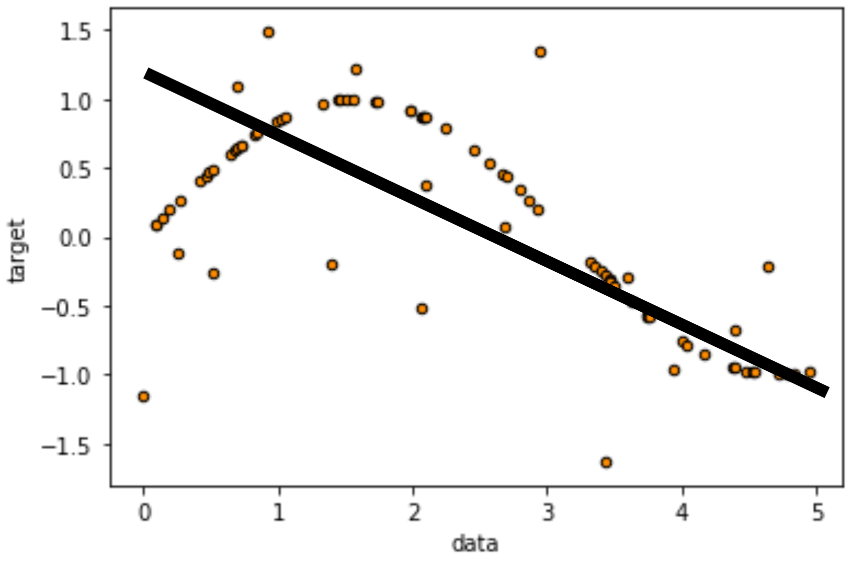


Предсказание стоимости квартиры

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

$$+ w_2 * (\text{район})$$

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

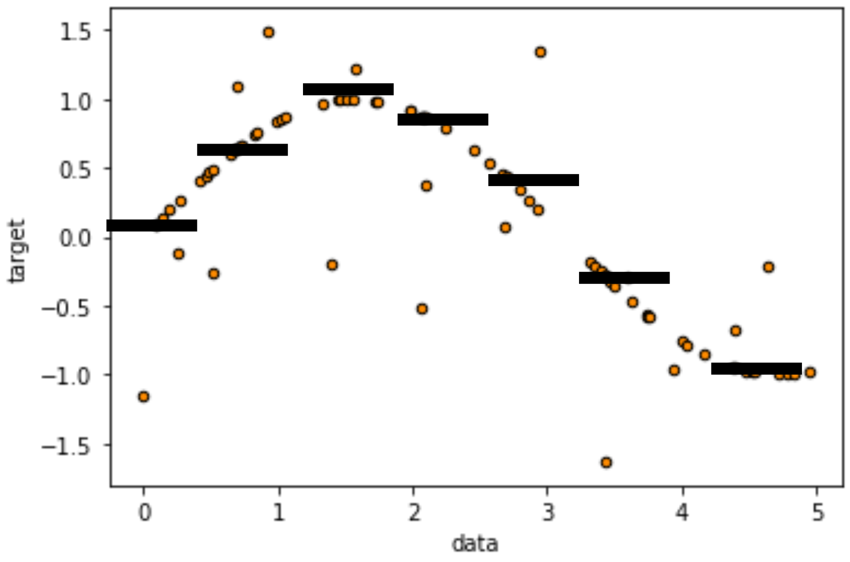


Предсказание стоимости квартиры

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

$$+ w_2 * (\text{район})$$

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

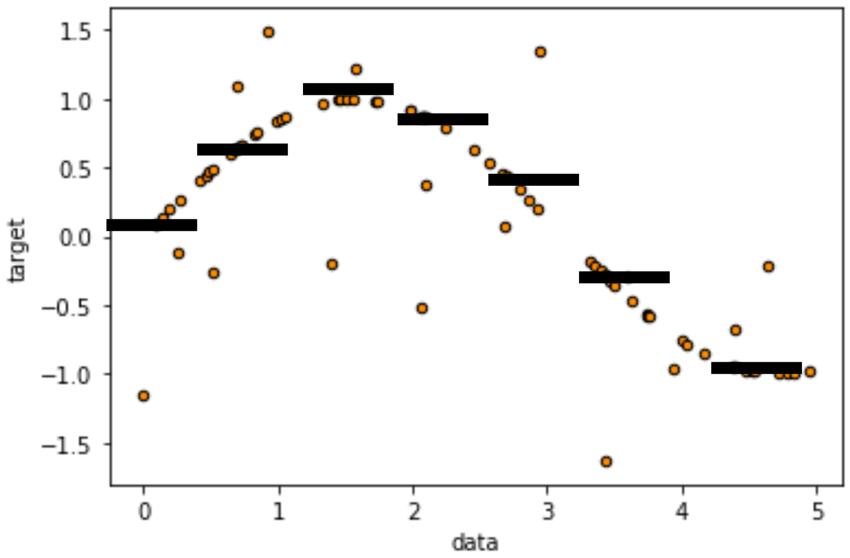


Предсказание стоимости квартиры

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

$$+ w_2 * (\text{район})$$

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



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

Линейная регрессия:

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

Обучение линейной регрессии - минимизация среднеквадратичной ошибки:

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

(здесь l – количество объектов)

ПОЧЕМУ MSE?

ВЕРОЯТНОСТНАЯ ПОСТАНОВКА

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

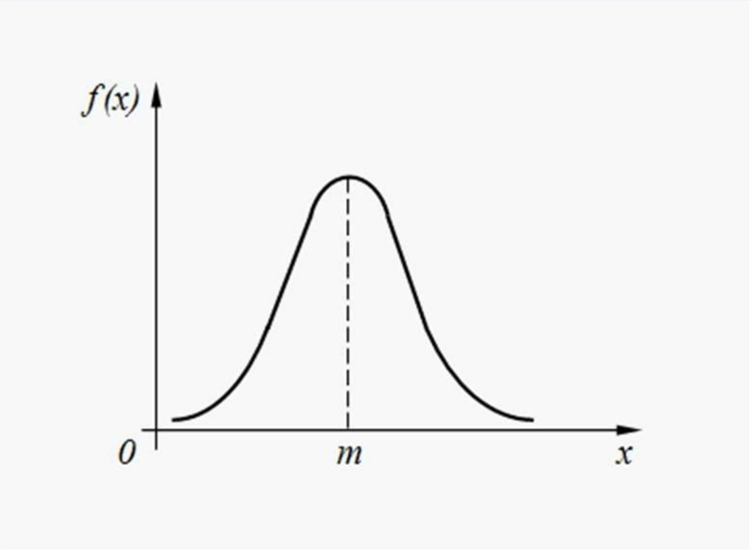
- Шум в данных обычно имеет некоторое распределение. В большинстве реальных задач считается, что

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

- Отсюда получаем, что

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

График плотности нормального распределения



ВЕРОЯТНОСТНАЯ ПОСТАНОВКА

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

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):

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

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

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:

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

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 $\boldsymbol{w}$, на которых достигается минимум функции ошибки.
- В простейшем случае, если ошибка среднеквадратичная, то её график – это парабола.
- Идея метода градиентного спуска:

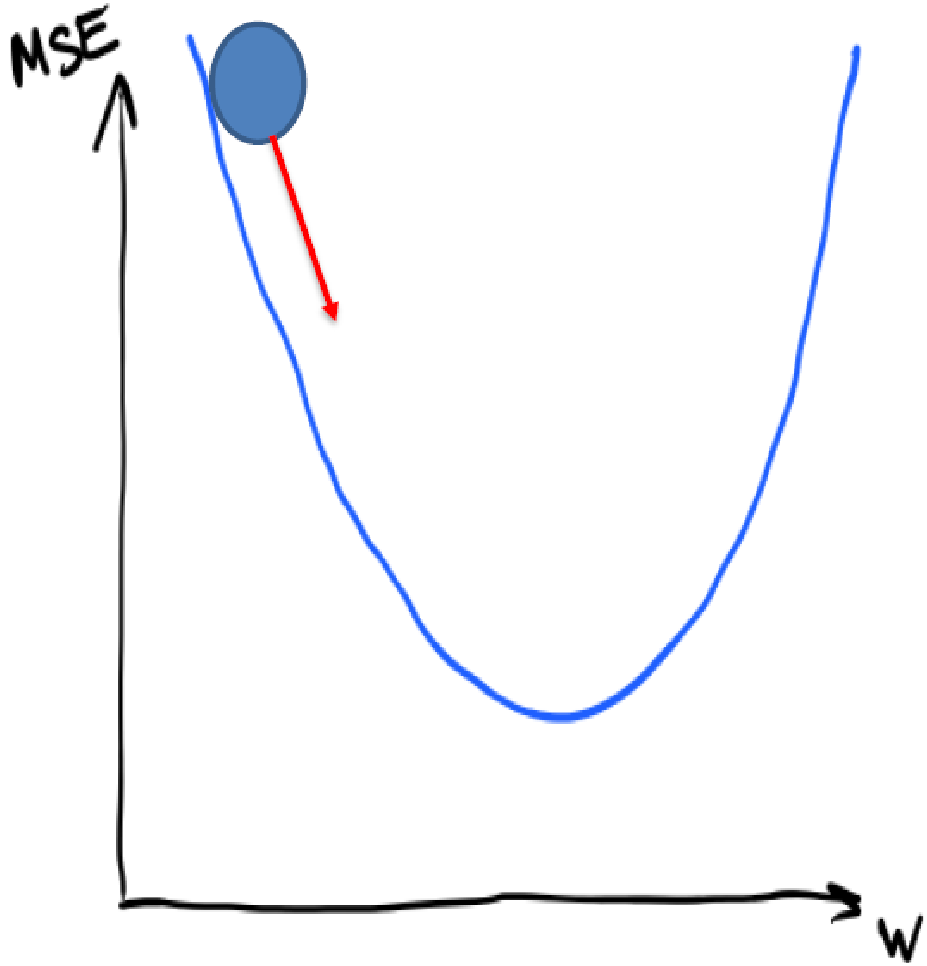
На каждом шаге (на каждой итерации метода) движемся в сторону антиградиента функции потерь!

То есть на каждом шаге движемся в направлении уменьшения ошибки.

Вектор градиента функции потерь обозначают $\boldsymbol{grad\ Q}$ *или* $\boldsymbol{\nabla Q}$.

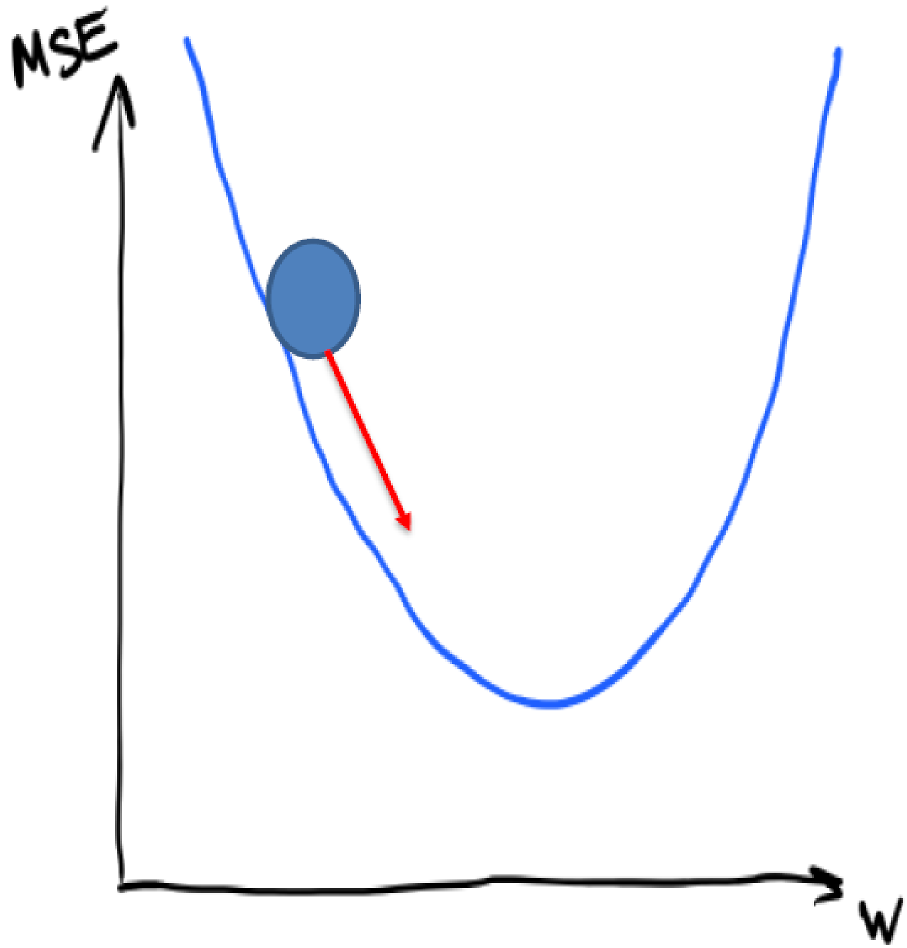
МЕТОД ГРАДИЕНТНОГО СПУСКА

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



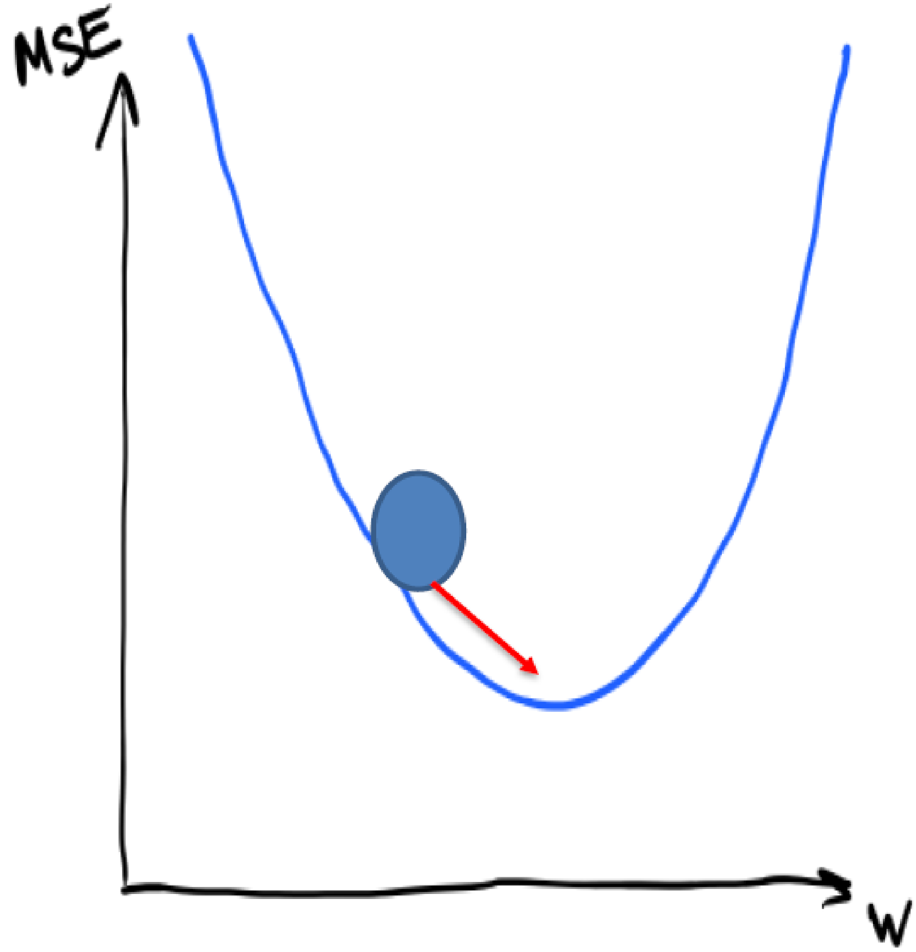
МЕТОД ГРАДИЕНТНОГО СПУСКА

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



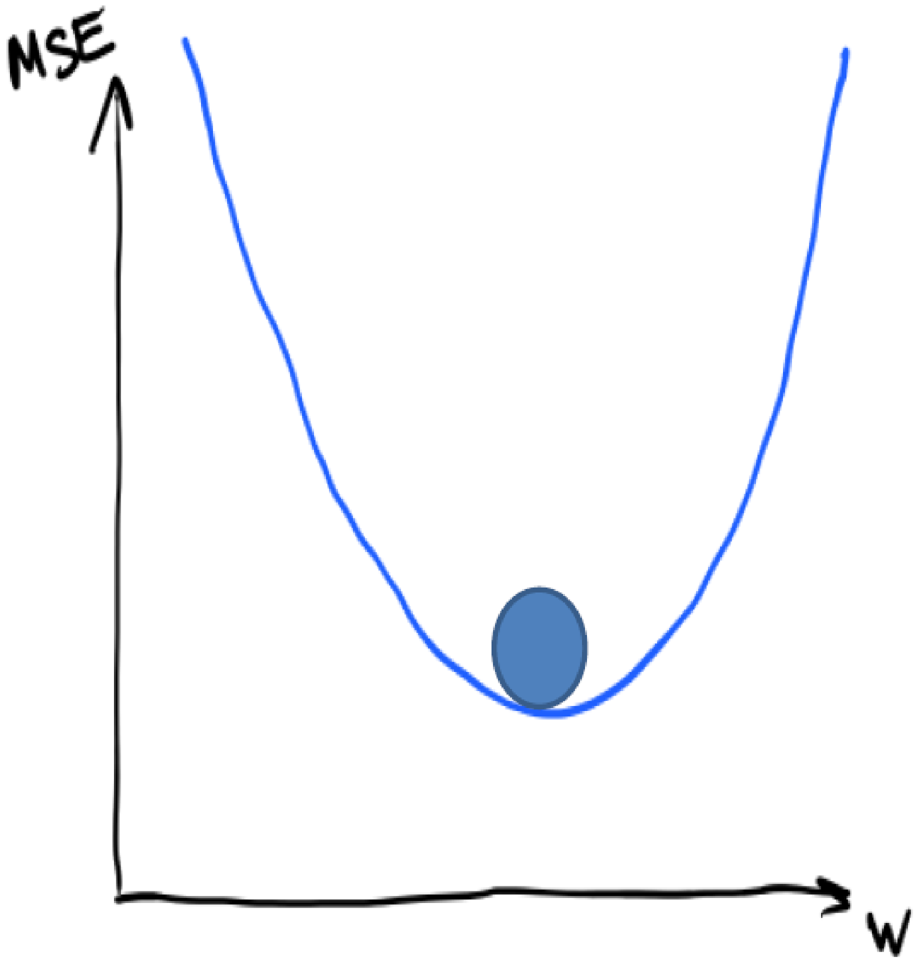
МЕТОД ГРАДИЕНТНОГО СПУСКА

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):

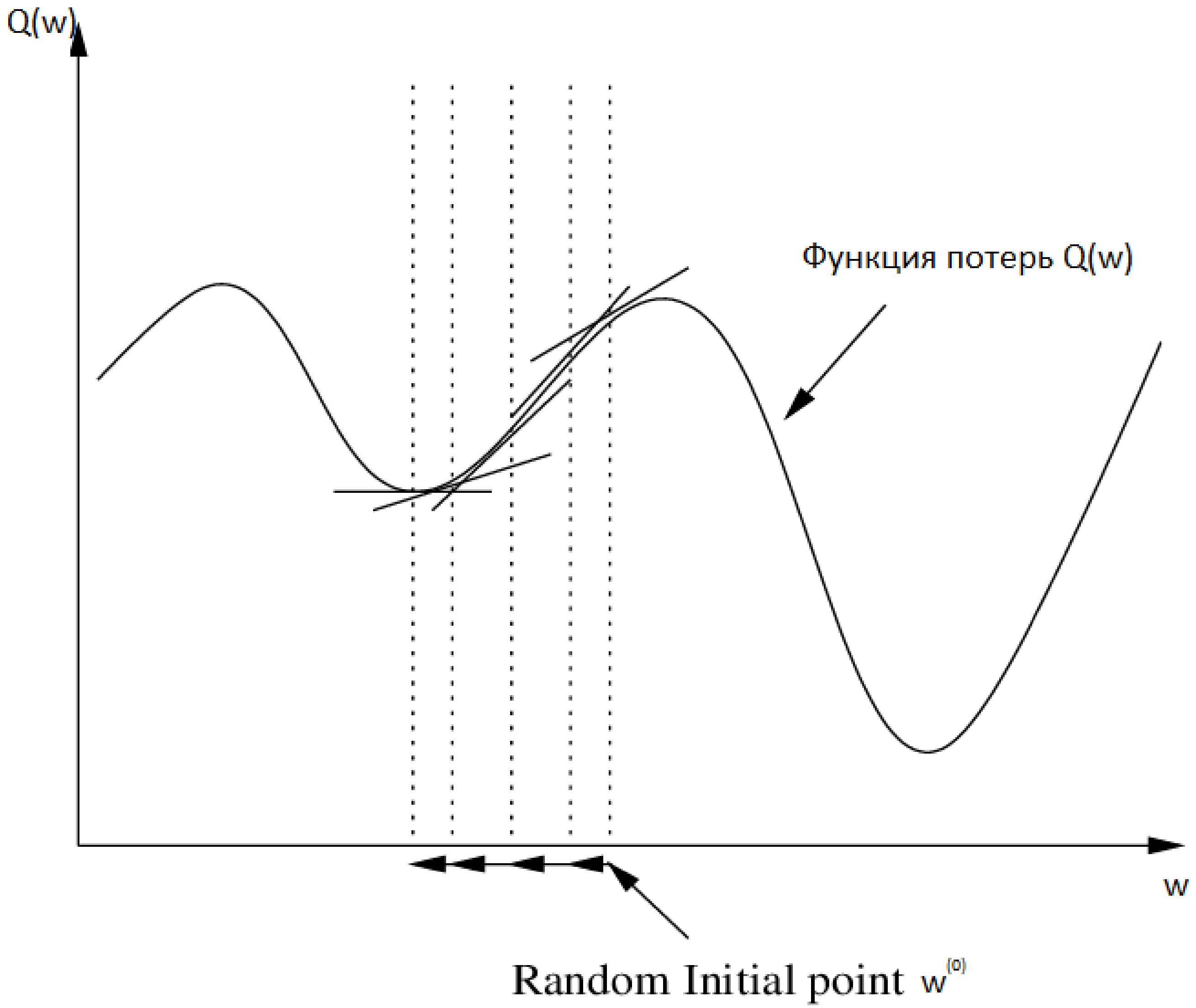
Пусть у нас только один вес - w.

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

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

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:

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

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:

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

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

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

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

- $w_j = 0, j = 1, \dots, n$
- 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

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

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

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

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

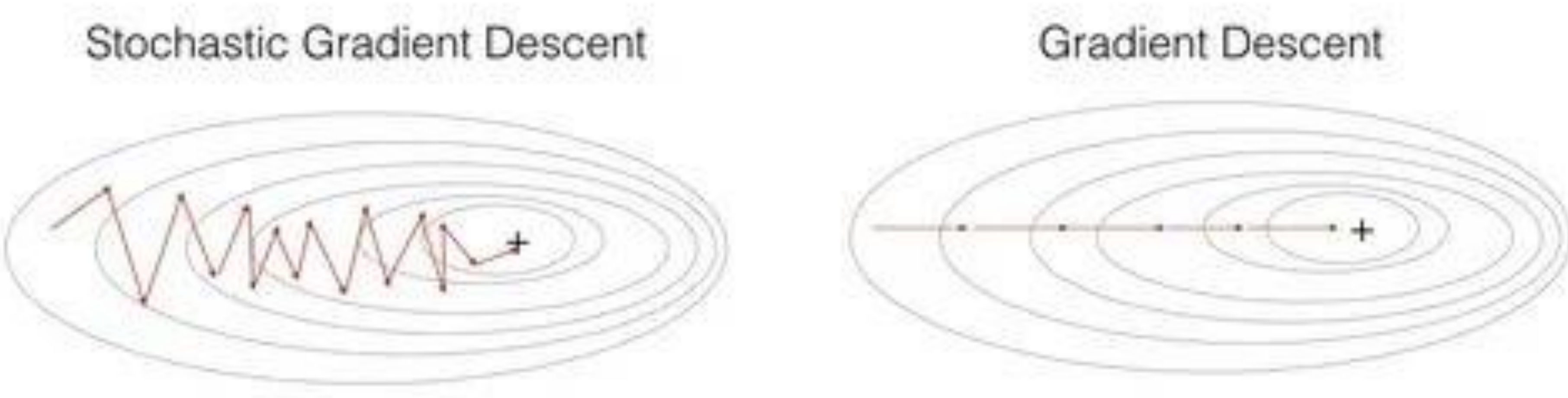
Stochastic gradient descent (SGD):

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

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

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

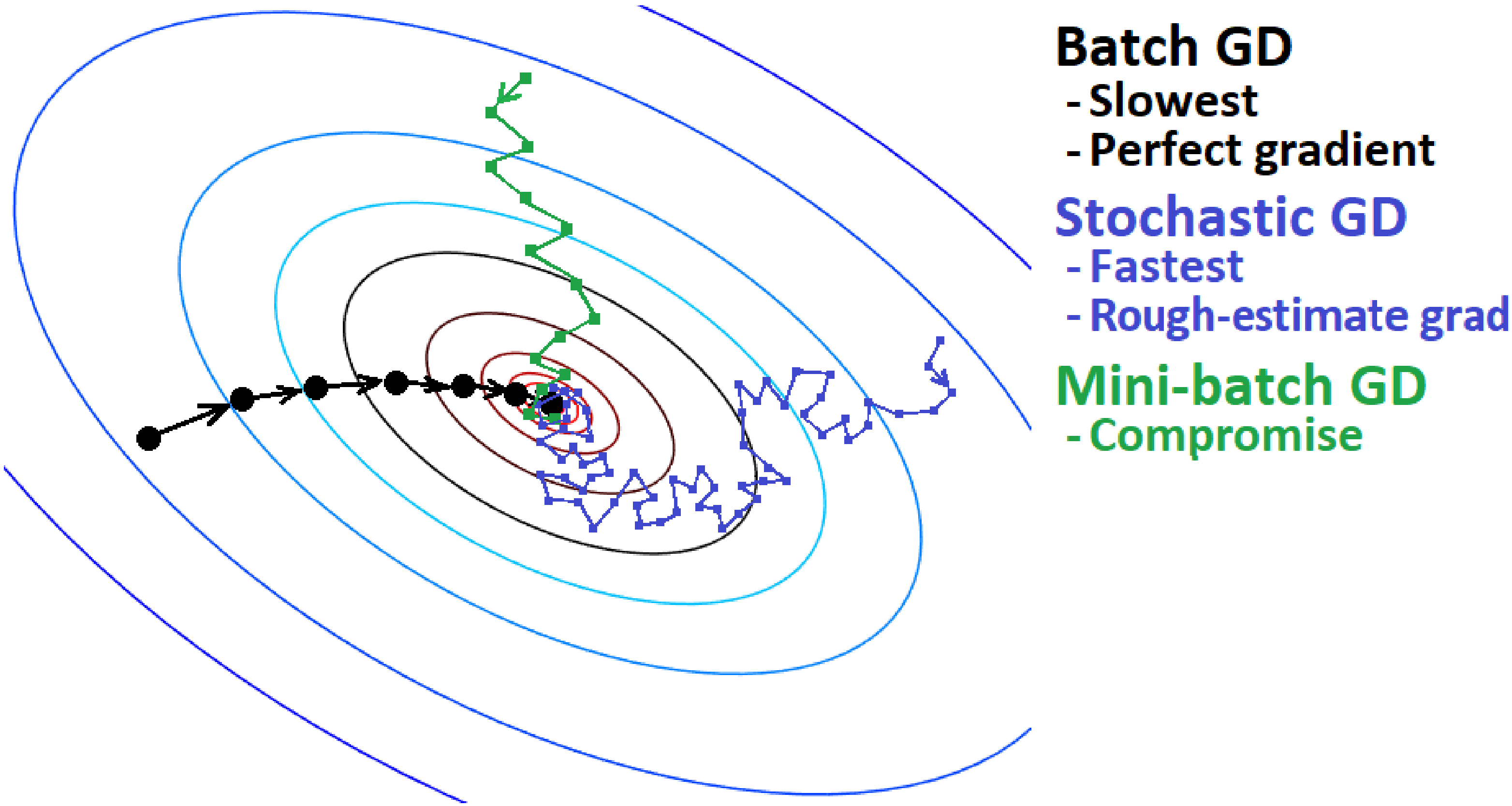
Промежуточное решение между классическим градиентным спуском и стохастическим вариантом.

- 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:

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

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

ВАРИАНТЫ ГРАДИЕНТНОГО СПУСКА



В следующих сериях

Линейная регрессия:

- Вывод аналитического решения в матричном виде для MSE
- Модификации градиентного спуска
- Метрики в задачах регрессии
- Регуляризация линейных моделей для борьбы с переобучением

Линейная классификация