

# 1. Метод градиентного спуска

Градиентные методы довольно часто используются для решения задач многомерной безусловной оптимизации. Пусть стоит задача найти минимум функции:

$$f(\theta) \rightarrow \min_{\theta}, \quad (1)$$

где  $\theta$  – **вектор** переменных. Алгоритм метода градиентного спуска состоит в последовательном движении в направлении наискорейшего спуска, то есть в направлении антиградиента  $-\nabla_{\theta} f(\theta)$ .

## Метод градиентного спуска

Вход: функция  $f(\theta)$ , начальное приближение  $\theta^0$ , шаг градиентного спуска  $\eta$ , число итераций  $n$ ,  $\epsilon$ .

1. Определить  $\nabla_{\theta} f$ .
2. На каждом шаге  $t = 1, 2, \dots, n$ :

$$\theta^{(t+1)} = \theta^{(t)} - \eta \nabla_{\theta} f(\theta^{(t)}).$$

Возможные критерии останова:

- a.  $\|\theta^{(t+1)} - \theta^{(t)}\| \leq \epsilon$ ,
- b.  $\|f(\theta^{(t+1)}) - f(\theta^{(t)})\| \leq \epsilon$ .

Если взять чересчур большим шаг градиентного спуска, то есть риск «перескочить» минимум. Чтобы этого избежать на каждой итерации алгоритма шаг градиентного спуска можно менять:

1. Обратно пропорционально номеру итерации:

$$\begin{aligned} \theta^{(t+1)} &= \theta^{(t)} - \eta^{(t)} \nabla_{\theta} f(\theta^{(t)}) \\ \eta^{(t+1)} &= \frac{\eta^{(0)}}{t}. \end{aligned}$$

2. Недостатком первого способа является то, что с увеличением номера итерации дальнейшие шаги будут чересчур маленькими, и есть риск не дойти до минимума. Чтобы этого избежать, можно использовать экспоненциально затухающий шаг:

$$\eta^{(t+1)} = \eta^{(0)} e^{-\frac{1-t}{t}}.$$

Выбирать шаг градиентного спуска можно таким образом, чтобы значение функции  $f(\theta^{(t+1)} - \eta^{(t+1)} \nabla_{\theta} f(\theta^{(t+1)}))$  было наименьшим. Для нахождения оптимального шага  $\eta^{(t+1)}$  используются любые методы одномерной оптимизации. Полученный алгоритм имеет название **метод наискорейшего спуска**.

---

## 2. Метод импульсов

Приведенные выше методы не учитывают характер и форму целевой функции. Метод импульсов помогает ускорить градиентный спуск в нужном направлении.

Согласно методу импульсов (методу моментов) точка обладает массой, соответствующей текущим значениям вектора переменных, а значит в тот момент, когда точка начнет движение в сторону, противоположную направлению градиента функции, у нее появится ненулевая скорость. Если точка пришла в новое положение с некоторой ненулевой скоростью, то ускорение направлено по градиенту и точка не может резко изменить направление движения. Идея метода состоит в том, чтобы на каждом шаге учитывать направление движения на предыдущем шаге. Обозначим направление движения на предыдущем шаге как  $v^{(t-1)}$ , тогда:

$$\begin{aligned}v^{(t)} &= \gamma v^{(t-1)} + \eta \nabla_{\theta} f(\theta^{(t)}), \\ \theta^{(t+1)} &= \theta^{(t)} - v^{(t)},\end{aligned}$$

где  $\gamma < 1$  – параметр, определяющий скорость изменения направления движения.

---

## 3. Метод Нестерова

Согласно методу Нестерова градиент необходимо считать не в текущей точке, а в той, в которую она придет после шага градиентного спуска. Таким образом алгоритм заглядывает вперед по вектору обновления:

$$\begin{aligned}v^{(t)} &= \gamma v^{(t-1)} + \eta \nabla_{\theta} f(\theta^{(t)} - \gamma v^{(t-1)}), \\ \theta^{(t+1)} &= \theta^{(t)} - v^{(t)}.\end{aligned}$$

Значение параметра  $\gamma$  рекомендуют выбирать равным 0.9.

---

## 4. Метод стохастического градиентного спуска

Отличие метода стохастического градиентного спуска состоит в том, что на каждом шаге градиентного спуска вектор переменных меняется не по всем наблюдениям выборки, а лишь по одному случайно взятому объекту. Таким образом, для сходимости алгоритма потребуется гораздо большее количество итераций, однако нет необходимости на каждом шаге вычислять сумму отклонений всех наблюдений от истинных значений.

### Метод стохастического градиентного спуска

Вход: функция  $f(\theta)$ , начальное приближение  $\theta^0$ , шаг градиентного спуска  $\eta$ , число итераций  $n$ ,  $\epsilon$ .

1. Определить  $\nabla_{\theta} f$ .
2. На каждом шаге  $t = 1, 2, \dots, n$ :

$$x_i \in X :$$

$$\theta^{(t+1)} = \theta^{(t)} - \eta \nabla_{\theta} f(\theta^{(t)}, \{x_i\}).$$

---

## 4\*. Метод стохастического градиентного спуска по мини-батчам

Главное преимущество метода стохастического градиентного спуска состоит в том, что для обучения предсказательной модели нет необходимости хранить в памяти всю выборку. Однако можно заметить, что для сходимости данного алгоритма требуется большое число итераций, кроме того алгоритм чувствителен к выбросам. Для устранения указанных недостатков применяют метод стохастического градиентного спуска по мини-батчам.

### Метод стохастического градиентного спуска по мини-батчам

Вход: функция  $f(\theta)$ , начальное приближение  $\theta^0$ , шаг градиентного спуска  $\eta$ , число итераций  $n$ ,  $\epsilon$ , размер мини-батча.

1. Определить  $\nabla_{\theta} f$ .
2. На каждом шаге  $t = 1, 2, \dots, n$ :

$$X^b \in X :$$

$$\theta^{(t+1)} = \theta^{(t)} - \eta \nabla_{\theta} f(\theta^{(t)}, X^b).$$