НЕЙРОННЫЕ СЕТИ В МАШИННОМ ОБУЧЕНИИ

Лекция №4
Методы оптимизации

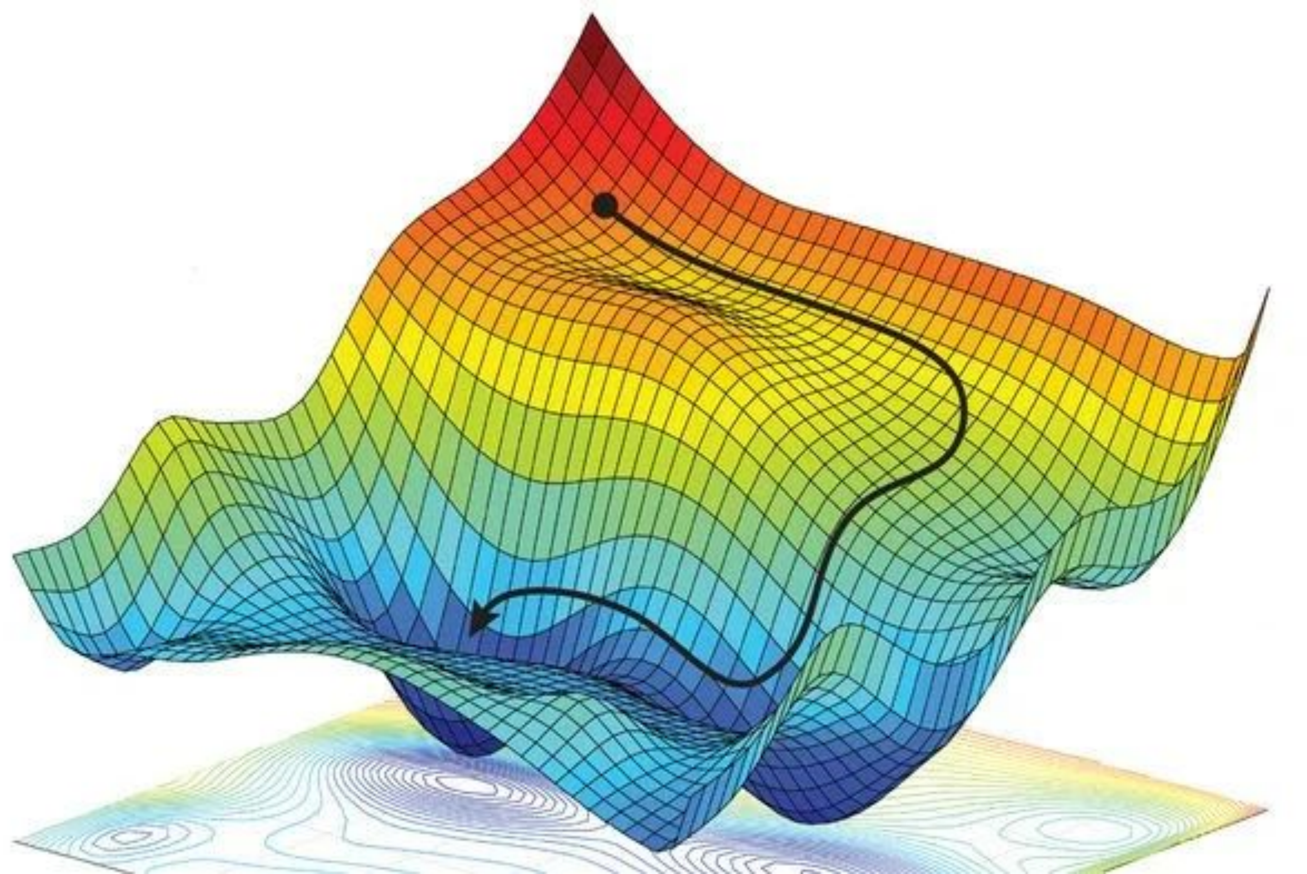
Евгений Ляпустин

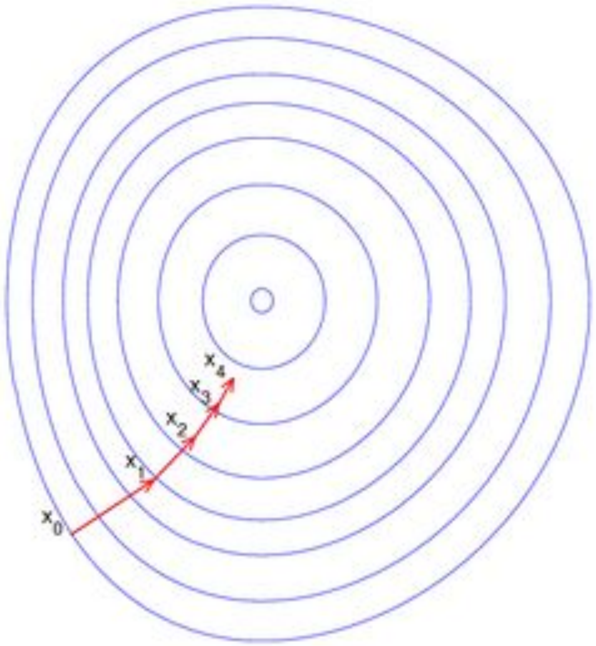
Содержание

1. Постановка задачи
2. Методы
    a. Gradient Descent
    b. Stochastic Gradient Descent (SGD)
    c. SGD+Momentum/Nesterov
    d. AdaGrad
    e. RMSProp
3. Практические проблемы

Постановка задачи

1. Надо найти оптимальные веса $\theta_* = \min_\theta J(\theta)$
2. В любой точке можем вычислить $\nabla_\theta J(\theta)$



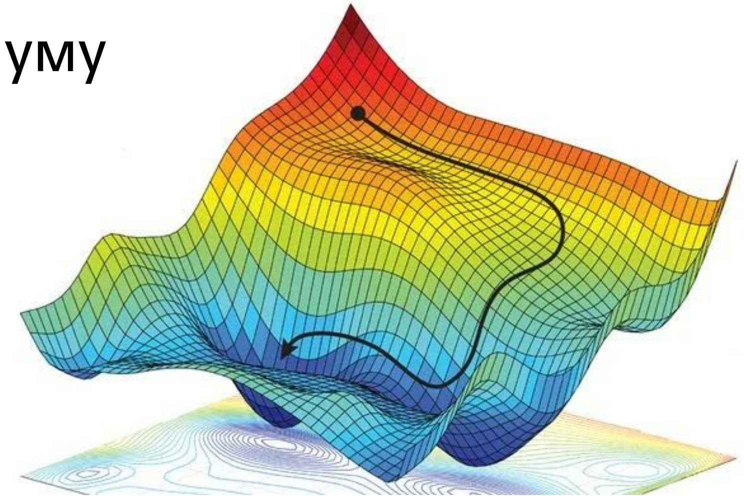


Gradient Descent

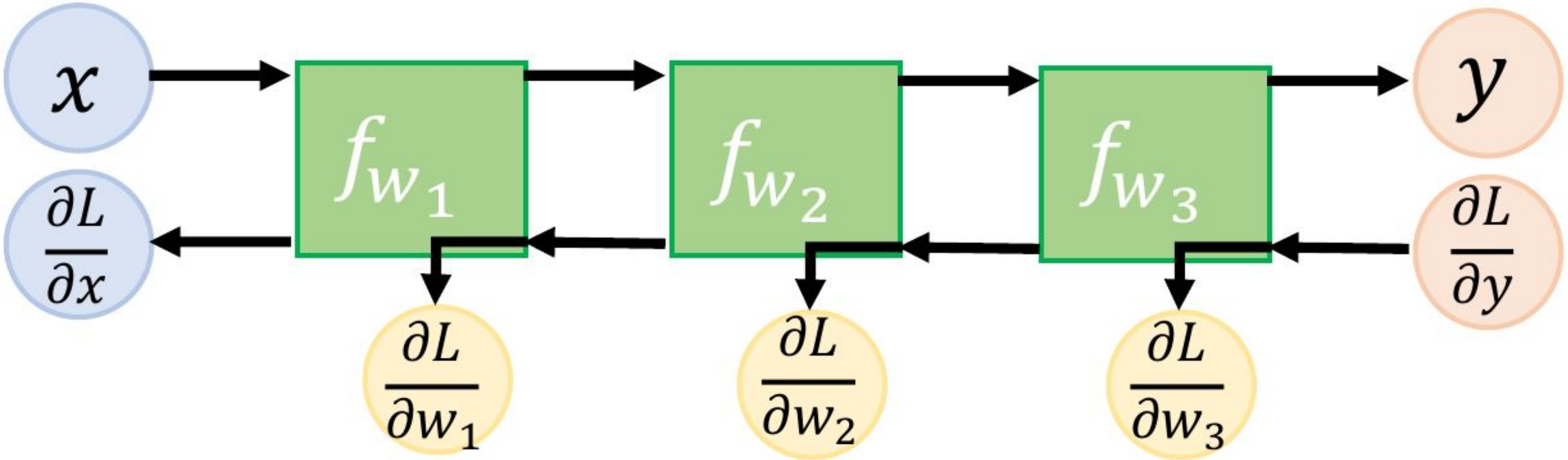
Градиентный спуск:

$$\theta_t = \theta_{t-1} - \eta \sum_{i=1}^{N} \nabla_\theta J_i(\theta_{t-1})$$

- Требуется обработать весь датасет для одного шага
- Нет режима online обучения
- Гарантируется сходимость к (локальному) минимуму при правильном выборе шага



Back propagation



Stochastic Gradient Descent (SGD)

Полная функция потерь:

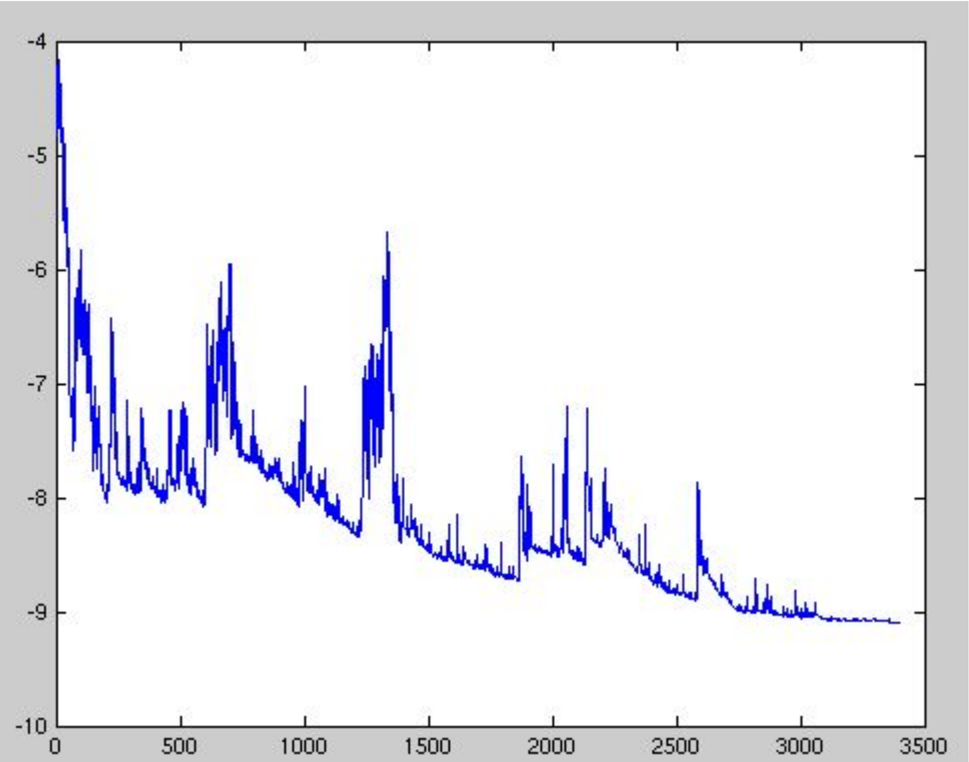
$$J(\theta) = \sum_{i=1}^{N} J_i(\theta)$$

Gradient Descent:

$$\theta_t = \theta_{t-1} - \eta \sum\nolimits_{i=1}^{N} \nabla_\theta J_i(\theta_{t-1})$$

(Mini-Batch) Stochastic Gradient Descent:

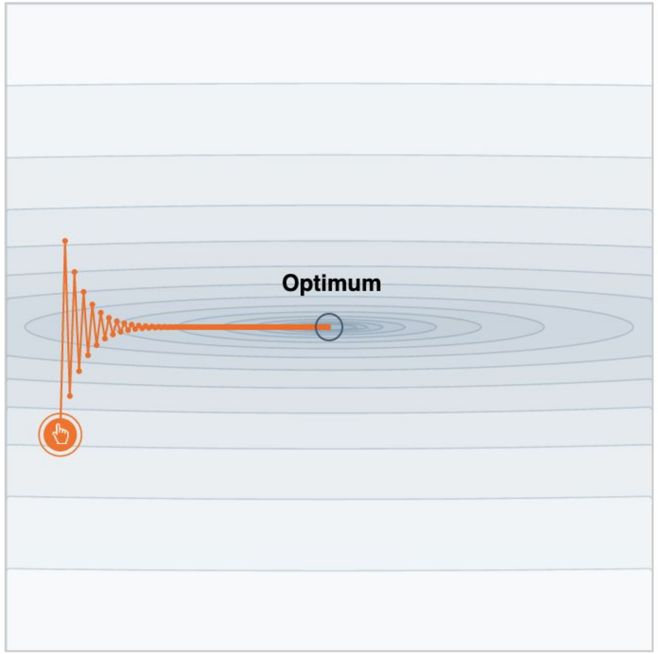
$$\theta_t = \theta_{t-1} - \eta \sum_{i_1,\ldots,i_k} \nabla_\theta J_i(\theta_{t-1})$$



SGD с batch_size=1

https://upload.wikimedia.org/wikipedia/commons/f/f3/Stogra.png

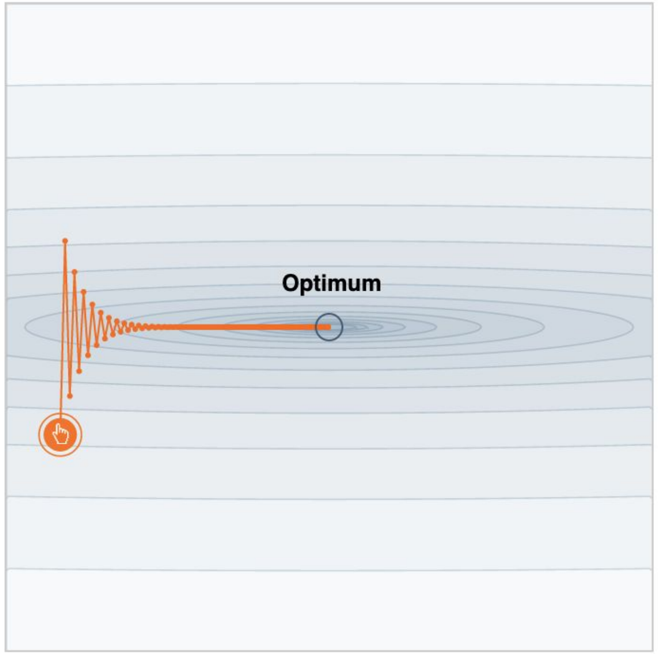
Проблемы SGD



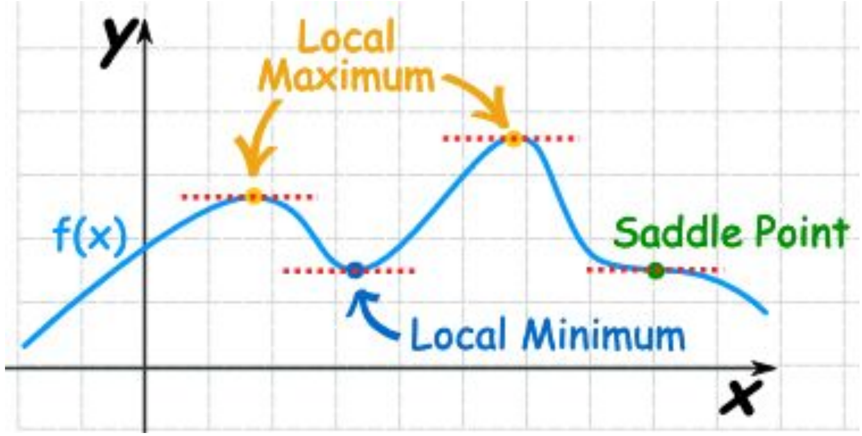
Разное поведение
у весов

https://distill.pub/2017/momentum/

Проблемы SGD



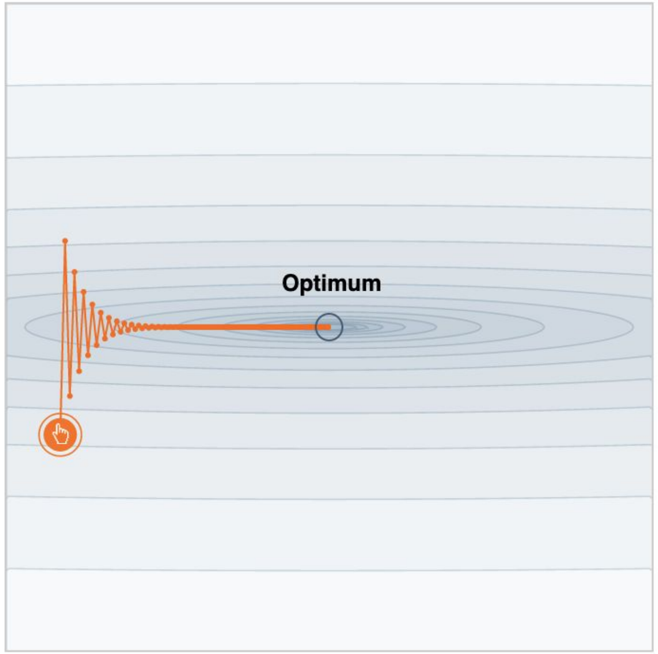
Разное поведение
у весов



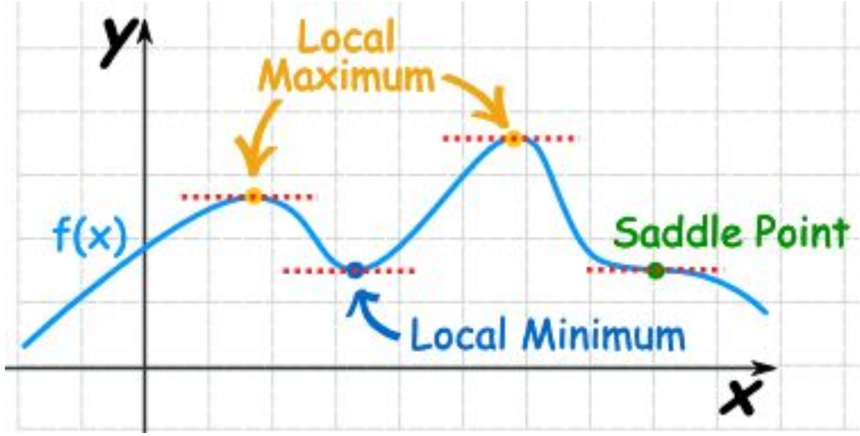
Локальные минимумы и
седловые точки

https://distill.pub/2017/momentum/
https://www.mathsisfun.com/calculus/maxima-minima.html

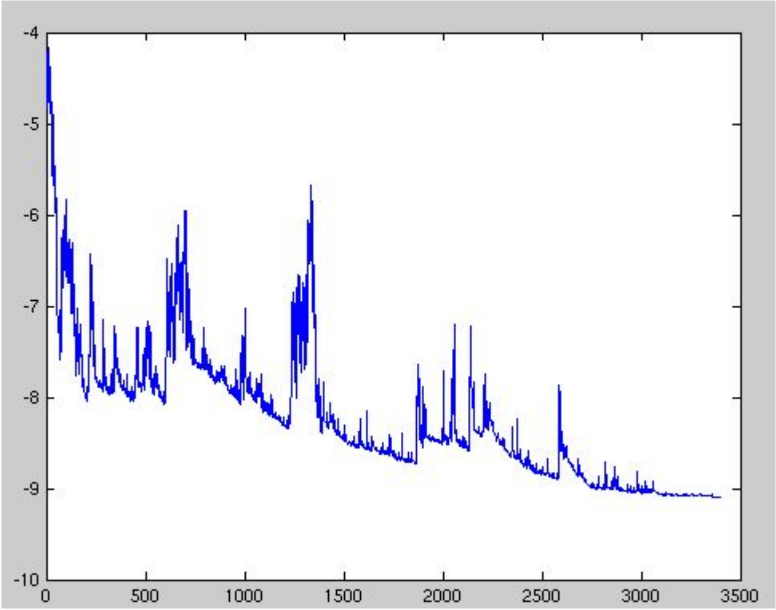
Проблемы SGD



Разное поведение
у весов



Локальные минимумы и
седловые точки



Шумная функция
потерь

https://distill.pub/2017/momentum/
https://www.mathsisfun.com/calculus/maxima-minima.html
https://upload.wikimedia.org/wikipedia/commons/f/f3/Stogra.png

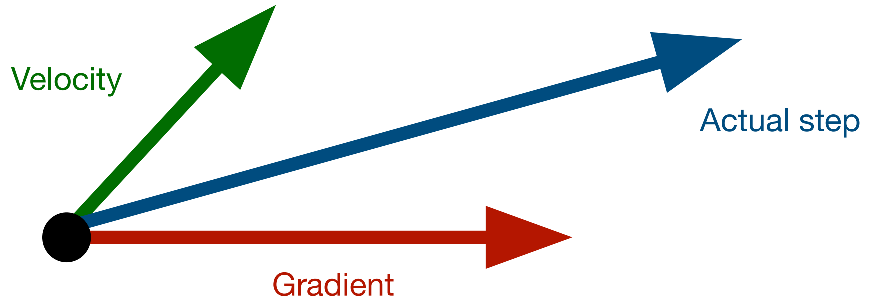
SGD + Momentum

Попробуем сохранять “инерцию”:

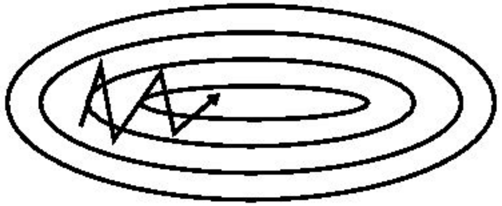
$$\nu_t = \gamma \nu_{t-1} + \eta_t \nabla_\theta J(\theta_{t-1})$$

$$\theta_t = \theta_{t-1} - \nu_t$$

Рекомендовано брать: $\gamma = 0.9$



SGD



SGD with momentum

https://www.ruder.io/optimizing-gradient-descent/

SGD + Momentum

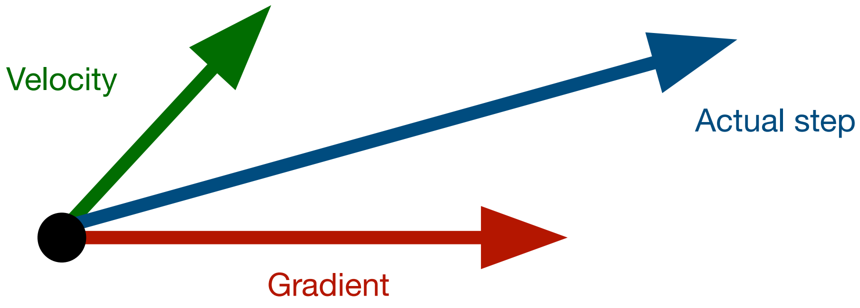
Попробуем сохранять “инерцию”:

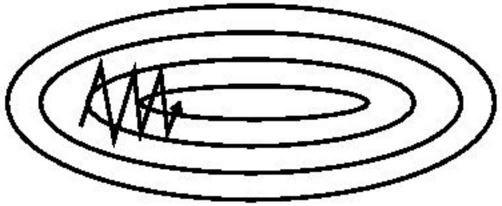
$$\nu_t = \gamma \nu_{t-1} + \eta_t \nabla_\theta J(\theta_{t-1})$$

$$\theta_t = \theta_{t-1} - \nu_t$$

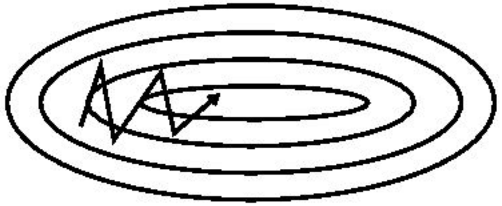
Рекомендовано брать: $\gamma = 0.9$

Какая проблема этого метода?





SGD



SGD with momentum

https://www.ruder.io/optimizing-gradient-descent/

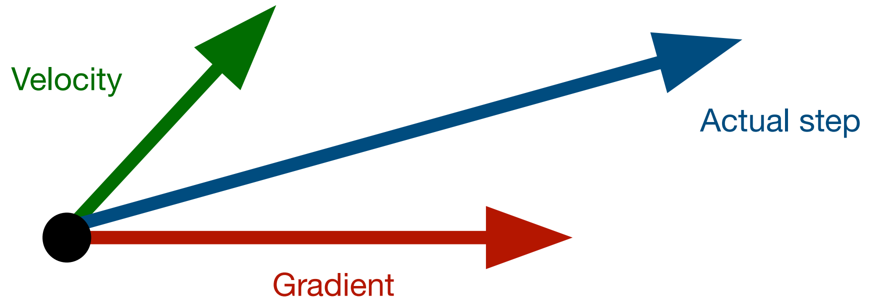
SGD + Momentum

Попробуем сохранять “инерцию”:

$$\nu_t = \gamma \nu_{t-1} + \eta_t \nabla_\theta J(\theta_{t-1})$$

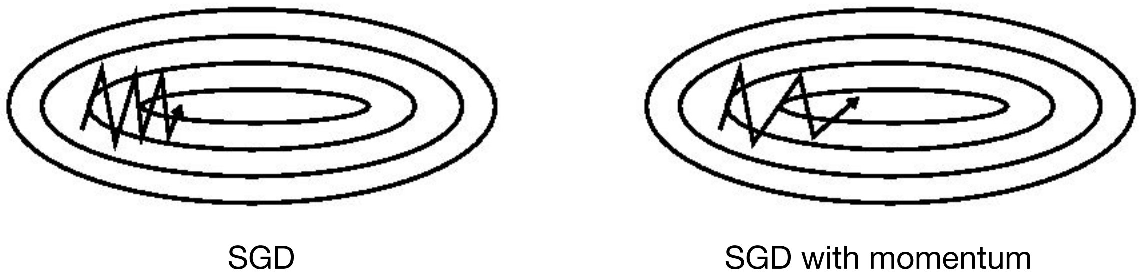
$$\theta_t = \theta_{t-1} - \nu_t$$

Рекомендовано брать: $\gamma = 0.9$



Какая проблема этого метода?

→ Может перескакивать локальные минимумы

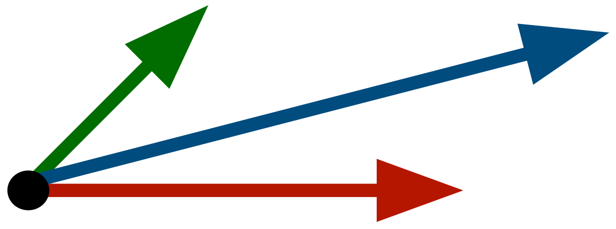


https://www.ruder.io/optimizing-gradient-descent/

Nesterov accelerated gradient

Сдвигаемся на момент до вычисления градиента:

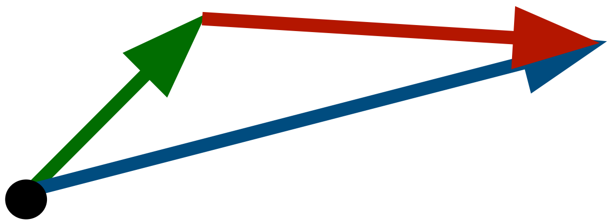
$$\nu_t = \gamma \nu_{t-1} + \eta_t \nabla_\theta J(\theta_{t-1} - \gamma \nu_{t-1})$$



simple momentum

Вычисление градиента в новой точке дает возможность скорректировать направление движения:

$$\theta_t = \theta_{t-1} - \nu_t$$



Nesterov momentum

AdaGrad

Градиент на шаге t:

$$g_t = \nabla_\theta J(\theta)$$

Сумма квадратов градиентов:

$$G_t = \sum_{k=0}^{t} g_t^2$$

Обновление весов:

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{G_t + \epsilon}} \cdot g_t$$

Стандартные значения:

$$\eta = 0.01,\ \epsilon = 10^{-8}$$

AdaGrad

Градиент на шаге t:

$$g_t = \nabla_\theta J(\theta)$$

Сумма квадратов градиентов:

$$G_t = \sum_{k=0}^{t} g_t^2$$

Обновление весов:

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{G_t + \epsilon}} \cdot g_t$$

Стандартные значения:

$$\eta = 0.01,\ \epsilon = 10^{-8}$$

Мотивация: уменьшаем learning rate для активных параметров

Какая проблема этого метода?

AdaGrad

Градиент на шаге t:

$$g_t = \nabla_\theta J(\theta)$$

Сумма квадратов градиентов:

$$G_t = \sum_{k=0}^{t} g_t^2$$

Обновление весов:

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{G_t + \epsilon}} \cdot g_t$$

Стандартные значения:

$$\eta = 0.01,\ \epsilon = 10^{-8}$$

Мотивация: уменьшаем learning rate для активных параметров

Какая проблема этого метода?
→ Не убывает => затухание обновлений

RMSProp

Будем использовать только последние несколько значений для подсчета

Экспоненциальное среднее:

$$G_t = \gamma G_{t-1} + (1-\gamma) g_t^2$$

Обновление весов:

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{G_t + \epsilon}} \cdot g_t$$

https://www.cs.toronto.edu/~hinton/coursera/lecture6/lec6.pdf

Adadelta

RMSProp + избавимся от learning rate

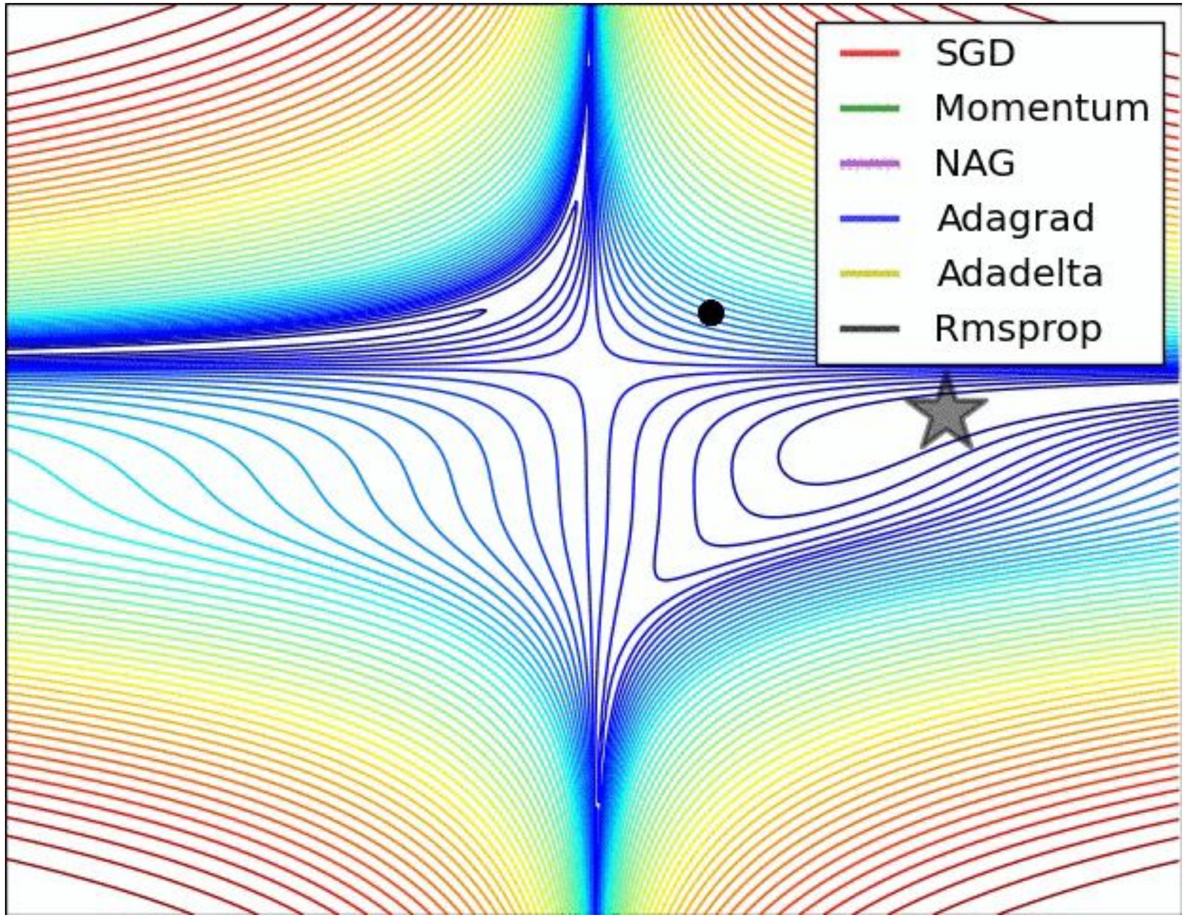
$$\Delta\theta_t = \frac{RMS[\Delta\theta]_{t-1}}{RMS[g]_t} g_t$$

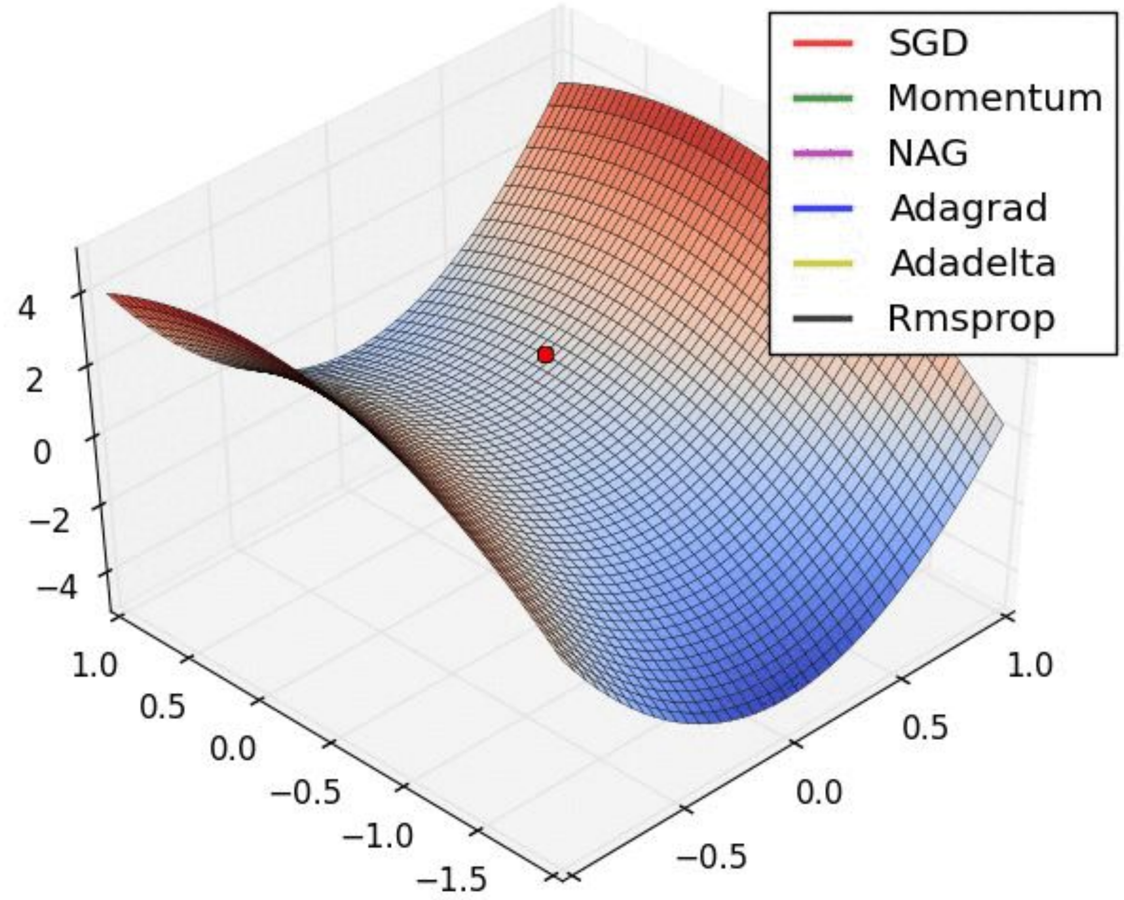
Обновление весов:

$$\theta_t = \theta_{t-1} - \Delta\theta_t$$

https://www.cs.toronto.edu/~hinton/coursera/lecture6/lec6.pdf

Сравнение методов





Adaptive Moment Estimation (Adam)

Объединим лучшие идеи

Момент и затухание:

$$\begin{cases} m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \\ \nu_t = \beta_2 \nu_{t-1} + (1-\beta_2) g_t^2 \end{cases}$$

Обновление весов:

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{\nu}_t + \epsilon}} \hat{m}_t$$

Adaptive Moment Estimation (Adam)

Объединим лучшие идеи

Момент и затухание:

$$\begin{cases} m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \\ \nu_t = \beta_2 \nu_{t-1} + (1-\beta_2) g_t^2 \end{cases}$$

Обновление весов:

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{\nu}_t + \epsilon}} \hat{m}_t$$

Инициализируем средние нулями, поэтому долгий “разгон”

Хотим несмещенность:

$$\mathbb{E}[m_t] = \mathbb{E}[g_t] \; \mathbb{E}[v_t] = \mathbb{E}[g_t^2]$$

Adaptive Moment Estimation (Adam)

Объединим лучшие идеи

Момент и затухание:

$$\begin{cases} m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \\ \nu_t = \beta_2 \nu_{t-1} + (1-\beta_2) g_t^2 \end{cases}$$

Обновление весов:

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{\nu}_t} + \epsilon} \hat{m}_t$$

Инициализируем средние нулями, поэтому долгий “разгон”

Хотим несмещенность:

$$\mathbb{E}[m_t] = \mathbb{E}[g_t] \quad \mathbb{E}[v_t] = \mathbb{E}[g_t^2]$$

Делаем поправку

$$\begin{cases} \hat{m}_t = \frac{m_t}{1-\beta_1^t} \\ \hat{\nu}_t = \frac{\nu_t}{1-\beta_2^t} \end{cases}$$

Критерии остановки

Когда остановить обучение?

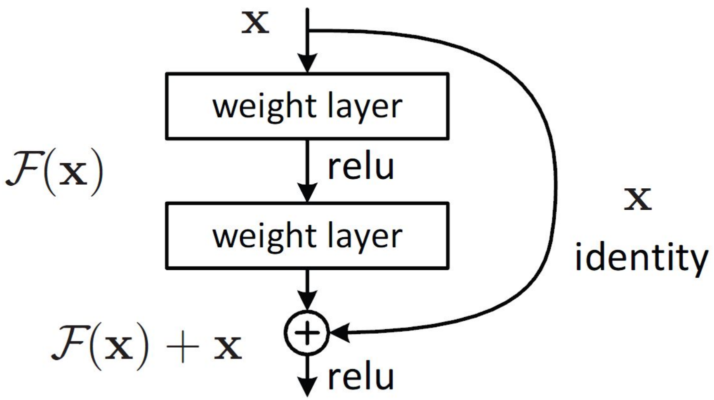
Критерии остановки

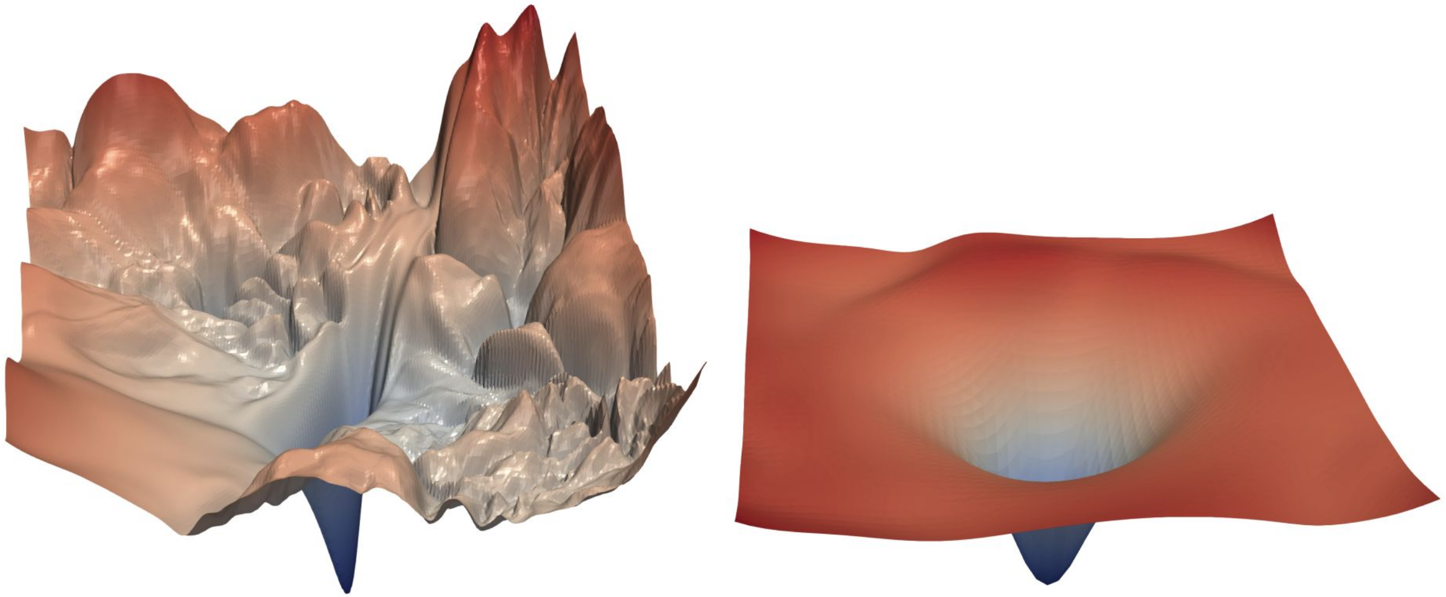
Когда остановить обучение?

- Превышен лимит по числу итераций или времени
- Качество на валидации начало ухудшаться
- $J(\theta_t) - J(\theta_*) \leq \epsilon$
- $J(\theta_t) \leq \epsilon J(\theta_0)$
- $\|\nabla J(\theta_t)\| \leq \epsilon \|\nabla J(\theta_0)\|$

Ландшафт функции потерь

В следующей лекции…





(a) without skip connections

(b) with skip connections

Figure 1: The loss surfaces of ResNet-56 with/without skip connections. The proposed filter normalization scheme is used to enable comparisons of sharpness/flatness between the two figures.

32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada.

[https://arxiv.org/pdf/1712.09913.pdf,](https://arxiv.org/pdf/1712.09913.pdf)
https://www.youtube.com/watch?v=78vq6kgsTa8

Спасибо!

На семинаре будем рисовать vvv