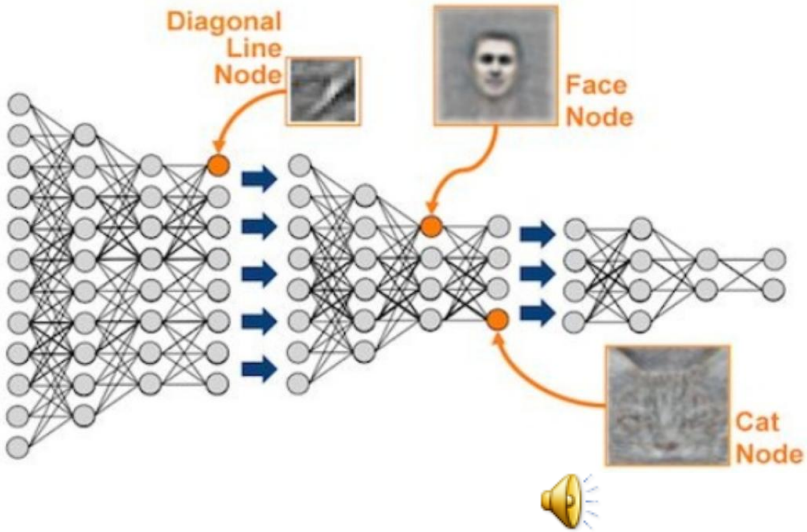
Основы практического использования нейронных сетей.

Лекция 6. Алгоритмы обучения для глубоких НС.

Дмитрий Буряк.
к.ф.-м.н.
dyb04@yandex.ru

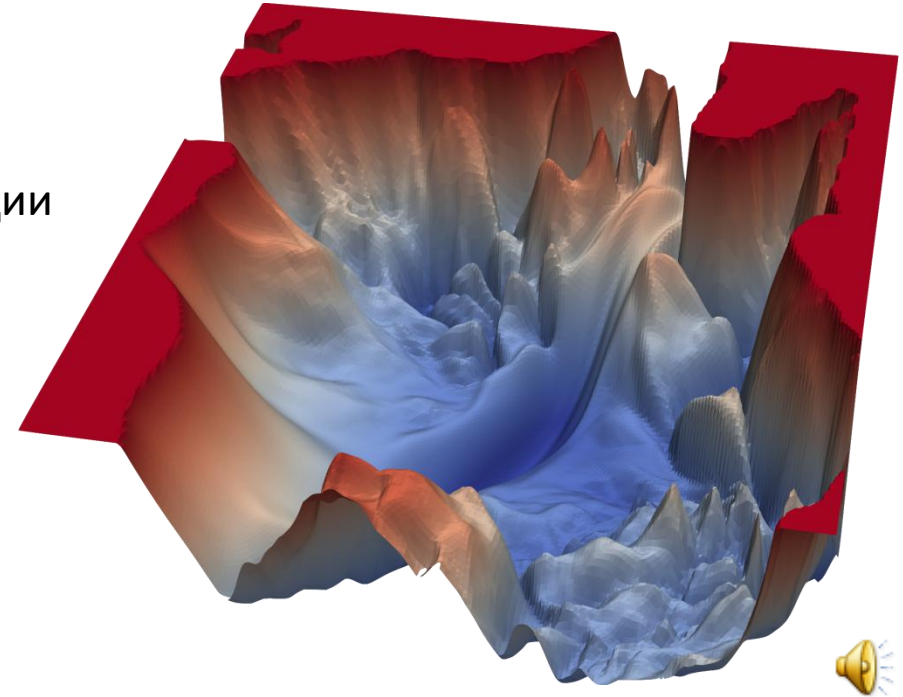
Глубокие НС vs традиционные НС

- Традиционные (shallow) НС:
  - 1-2 внутренних слоев;
  - необходимость предвартельного выделения признаков, предобработка данных;
  - «простота» обучения.
- Глубокие НС:
  - >2 слоев;
  - моделирование абстракций в данных;
  - большое число слоев и нейронов -> большое число весов.
  - адаптирование алгоритмов обучения.
  - высокие требования к вычислительным ресурсам.



Трудности обучения глубоких НС

- ❑ Локальные минимумы функции ошибки
- ❑ Сложный ландшафт целевой функции
- ❑ Неравномерность обновления параметров сети
- ❑ Выбор закона изменения скорости обучения



https://www.cs.umd.edu/~tomg/projects/landscapes/

Стохастический градиентный спуск

- Основной алгоритм для глубокого обучения.
- Базируется на принципах градиентного спуска.
- Вычисление градиента по каждому примеру вычислительно затратно → выбор случайных подмножест из обучающей выборки.

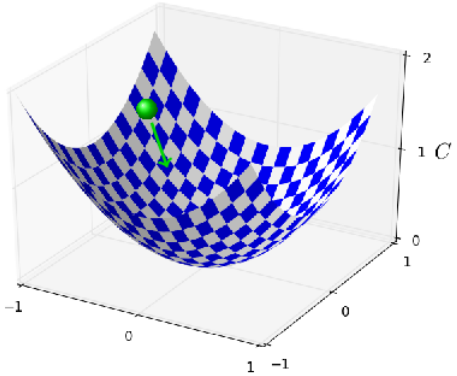
$$w_k \to w'_k = w_k - \eta \frac{\partial C}{\partial w_k}$$

$$b_l \to b'_l = b_l - \eta \frac{\partial C}{\partial b_l}.$$

$$\nabla C \approx \frac{1}{m} \sum_{j=1}^{m} \nabla C_{X_j}$$

$$w_k \to w'_k = w_k - \frac{\eta}{m} \sum_j \frac{\partial C_{X_j}}{\partial w_k}$$

$$b_l \to b'_l = b_l - \frac{\eta}{m} \sum_j \frac{\partial C_{X_j}}{\partial b_l},$$



Методы оптимизации 2-го порядка

$$C(w + \Delta w) \approx C(w) + \nabla C \cdot \Delta w + \frac{1}{2} \Delta w^T H \Delta w$$

$$\Delta w = -H^{-1} \nabla C.$$

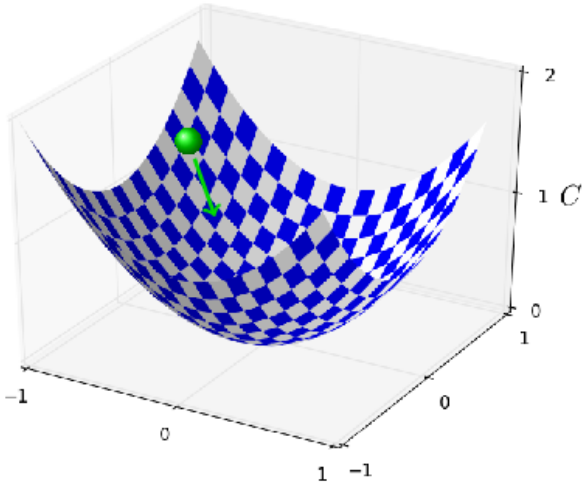
- Оценка Гессиана ~ скорость изменения градиента.
- Требует много памяти: 1000000 весов → 10^{12} элементов Гессиана → >3Tb RAM .
- Методы частичной апроксимации Гессиана (L-BFGS).
- Аппроксимация требует использования всех примеров из обучающей выборки.

Использование момента

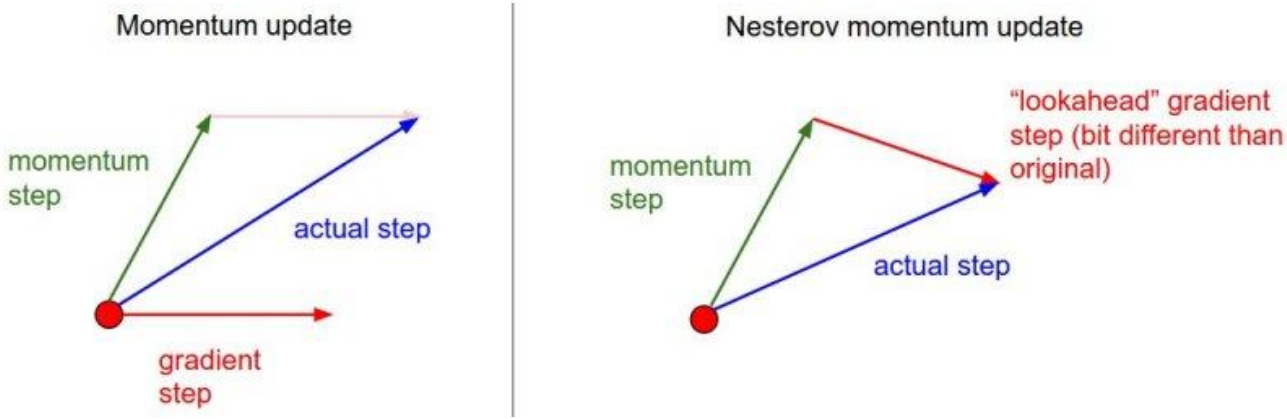
- Идея накопления импульса.
- Оценка «скорости» градиента.
- Применение коэффициента «трения» ~ коэффициент момента.

$$v \to v' = \mu v - \eta \nabla C$$
$$w \to w' = w + v'.$$

- Изменение коэффициента момента в процессе обучения 0.5→0.9



Момент Нестерова



- Оценка градиента в предсказанной позиции функции ошибки

Изменение скорости обучения

- Ступенчатое уменьшение
  - в ***к*** раз каждые ***n*** итераций обучения;
  - уменьшение коэффициента обучения, если ошибка на подтверждающей выборке перестала уменьшаться.

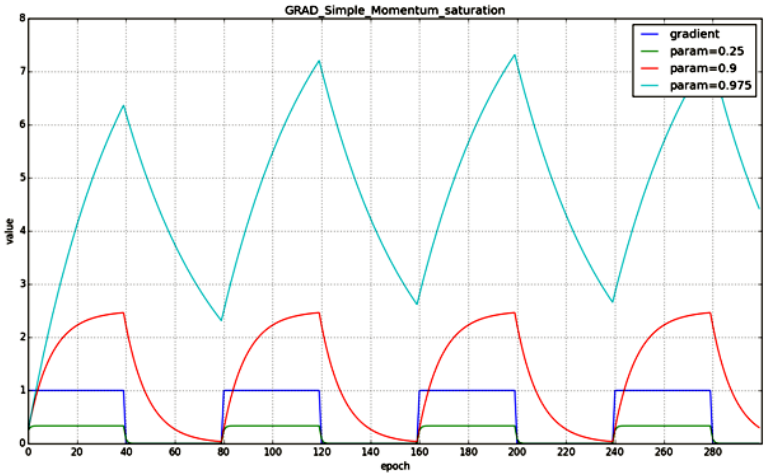
- Экспоненциальное $$\eta = \alpha_0 e^{-\beta t}$$

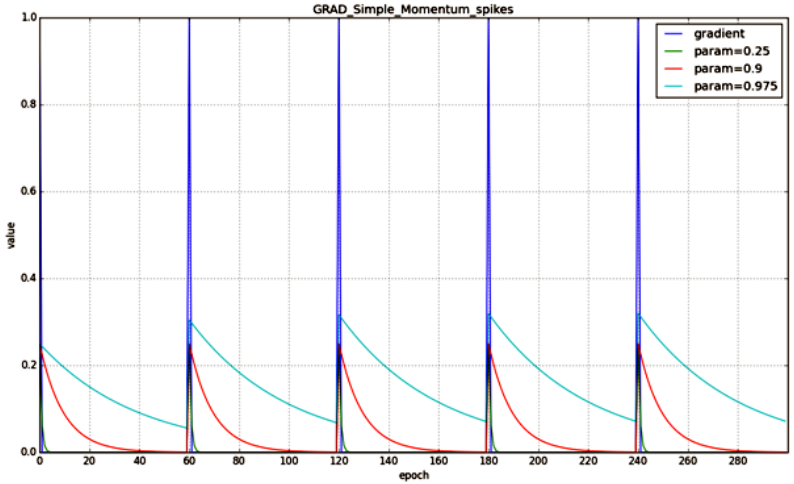
- Гиперболическое $$\eta = \frac{\alpha_0}{1+\beta t}$$

Обзор алгоритмов обучения*. Момент.

$$v_t = \gamma v_{t-1} + \eta g_t \qquad g_t = \nabla_\theta C(\theta)$$

$$\theta_{t+1} = \theta_t - v_t \qquad \gamma = 0.9$$



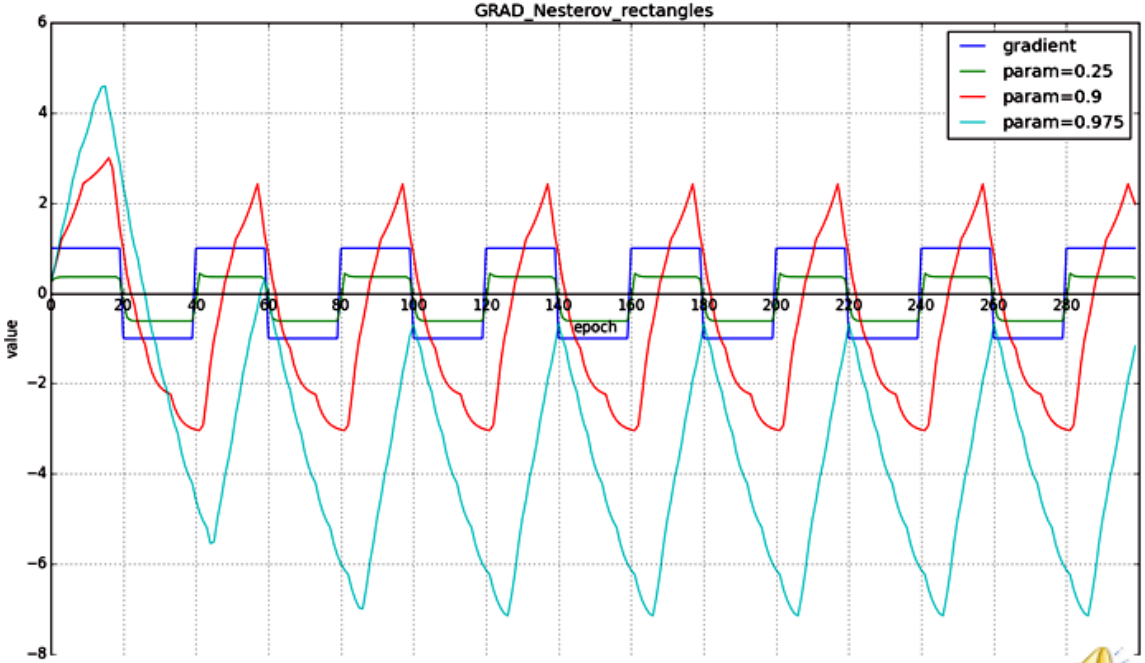


** https://habrahabr.ru/post/318970/*

Обзор алгоритмов обучения. Момент Нестерова.

$$v_t = \gamma v_{t-1} + \eta g_t(\theta - \gamma v_{t-1})$$

$$\theta_{t+1} = \theta_t - v_t$$



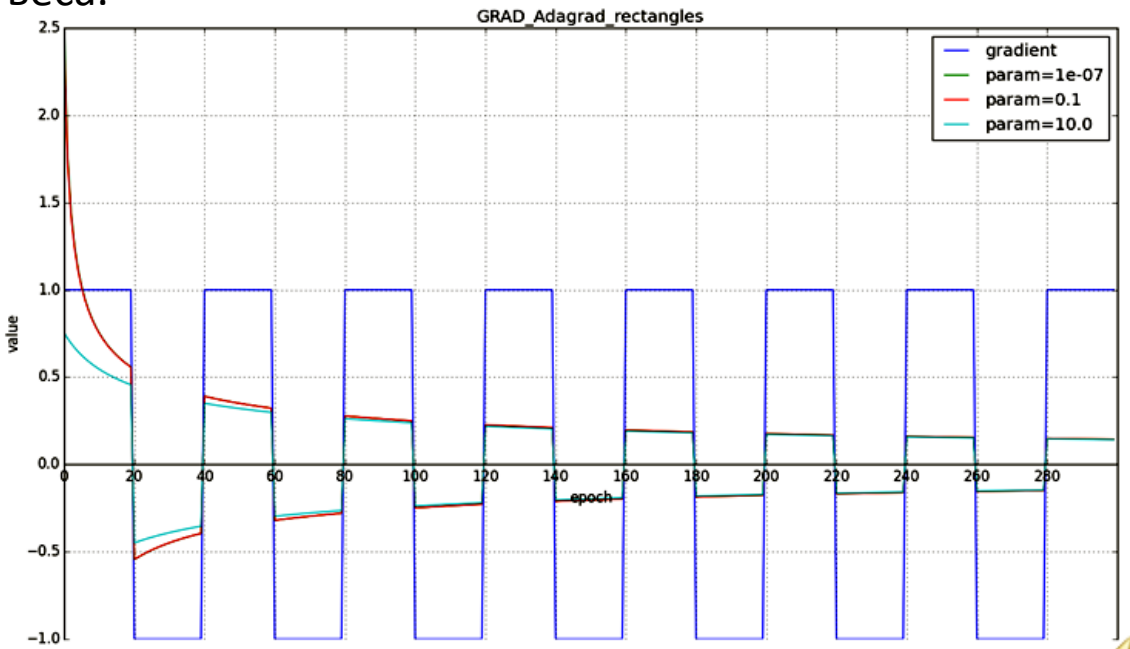
Обзор алгоритмов обучения. Adagrad.

- Учет частоты обновления веса.
- Устойчивость к выбору скорости обучения.

$$G_t = G_t + g_t^2$$

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

ϵ — сглаживающий параметр.

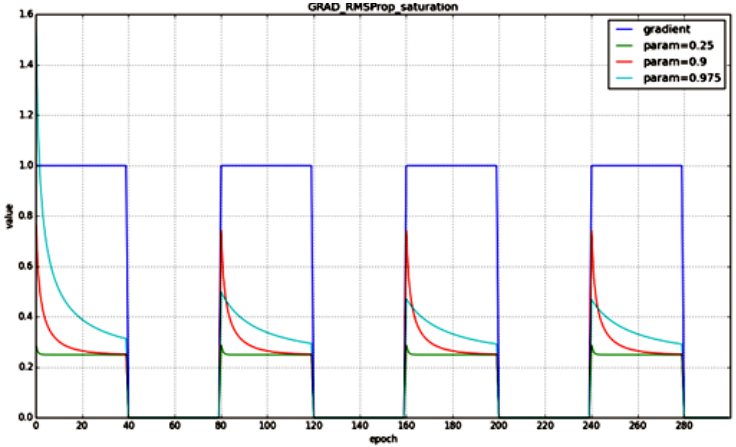


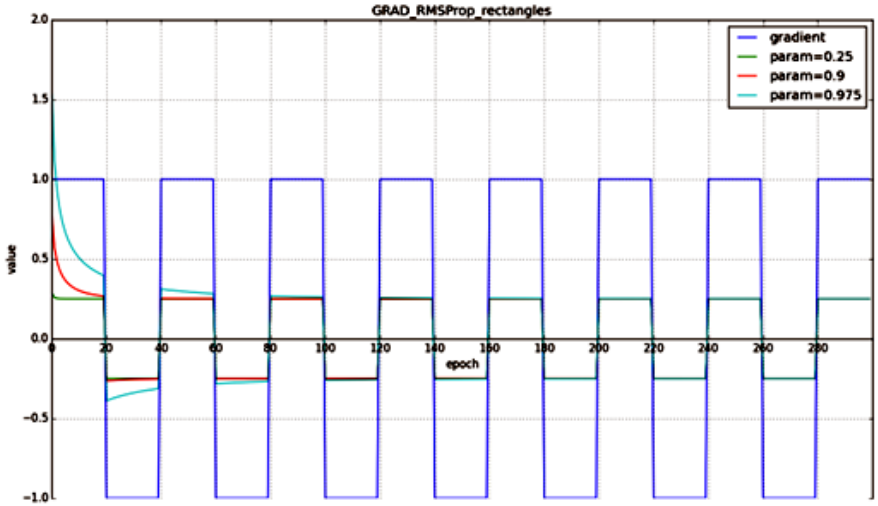
Обзор алгоритмов обучения. RMSProp.

- Учет частоты обновления веса за счет усреднения.

$$E[g^2]_t = \gamma E[g^2]_{t-1} + (1-\gamma) g_t^2$$

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{E[g^2]_t + \epsilon}} g_t$$





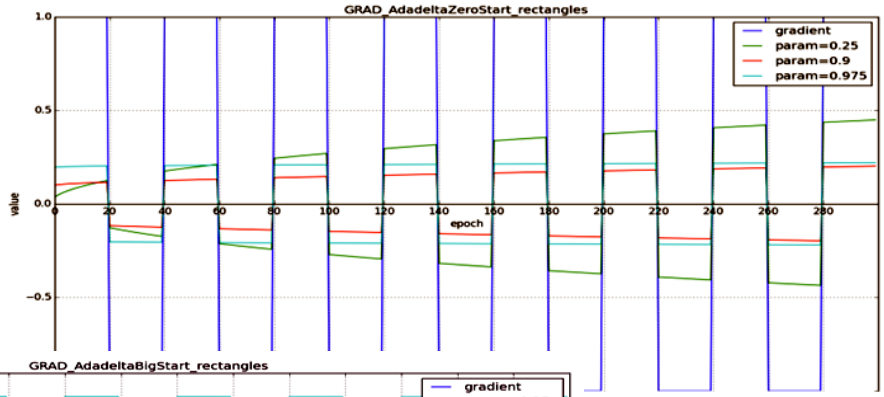
Обзор алгоритмов обучения. Adadelta.

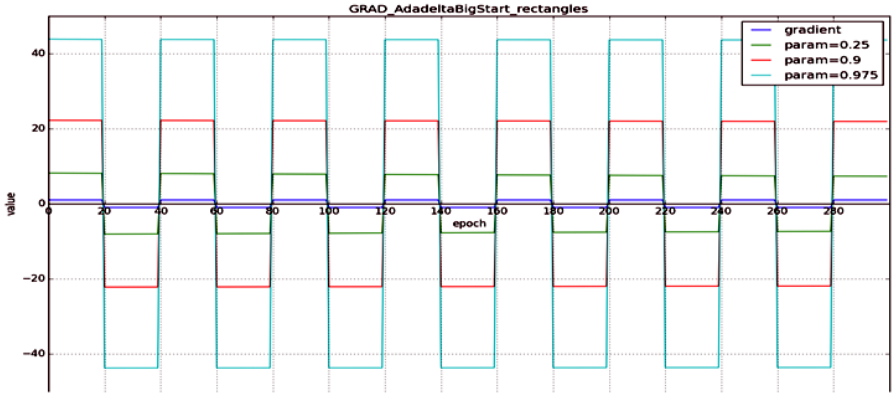
$$\Delta\theta = -\frac{RMS[\Delta\theta]_{t-1}}{RMS[g]_t} g_t$$

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

$$E[\Delta\theta^2]_t = \gamma E[\Delta\theta^2]_{t-1} + (1-\gamma)\Delta\theta_t^2$$

$$RMS[\Delta\theta]_t = \sqrt{E[\Delta\theta^2]_t + \epsilon}$$





Обзор алгоритмов обучения. Adam.

- Момент + учет частоты обновления веса.

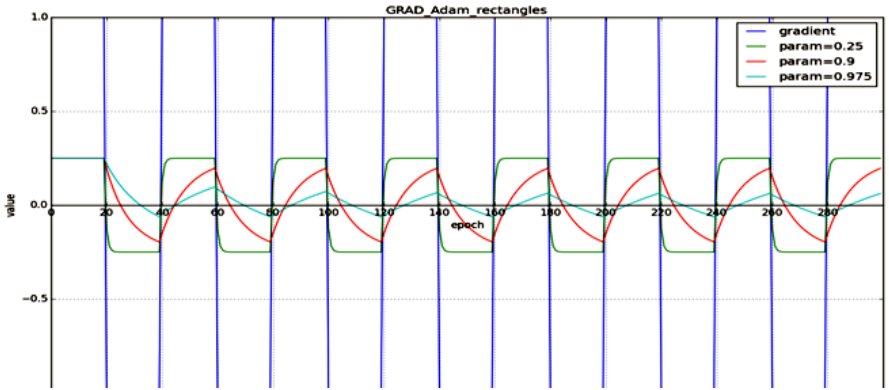
$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t$$

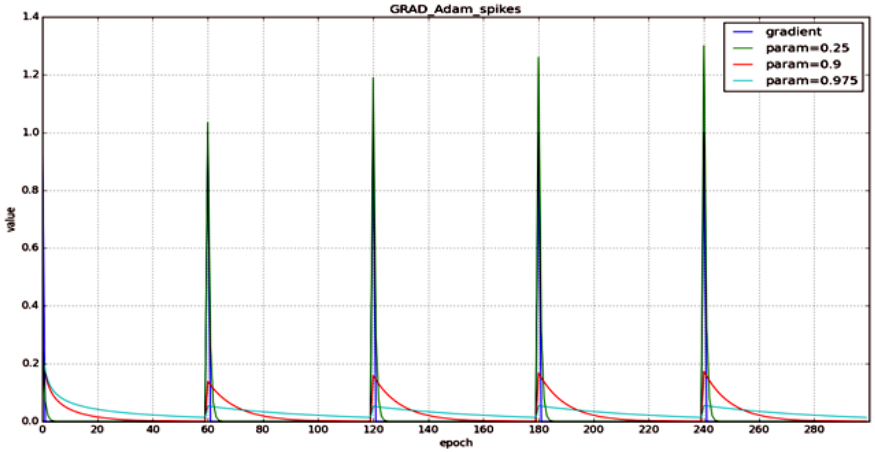
$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2$$

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t}, \ \hat{v}_t = \frac{v_t}{1-\beta_2^t}$$

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

$$\beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 10^{-8}$$





Обзор алгоритмов обучения. Adamax.

$$v_t = \beta_2^p v_{t-1} + (1-\beta_2^p)|g_t|^p$$

$$u_t = \lim_{p\to\infty} v_t^{\frac{1}{p}} = \lim_{p\to\infty}\left[\beta_2^p v_{t-1} + (1-\beta_2^p)|g_t|^p\right] = \lim_{p\to\infty}\left(\sum_{i=1}^{t}\beta_2^{p(t-i)}|g_i|^p\right)^{\frac{1}{p}}$$

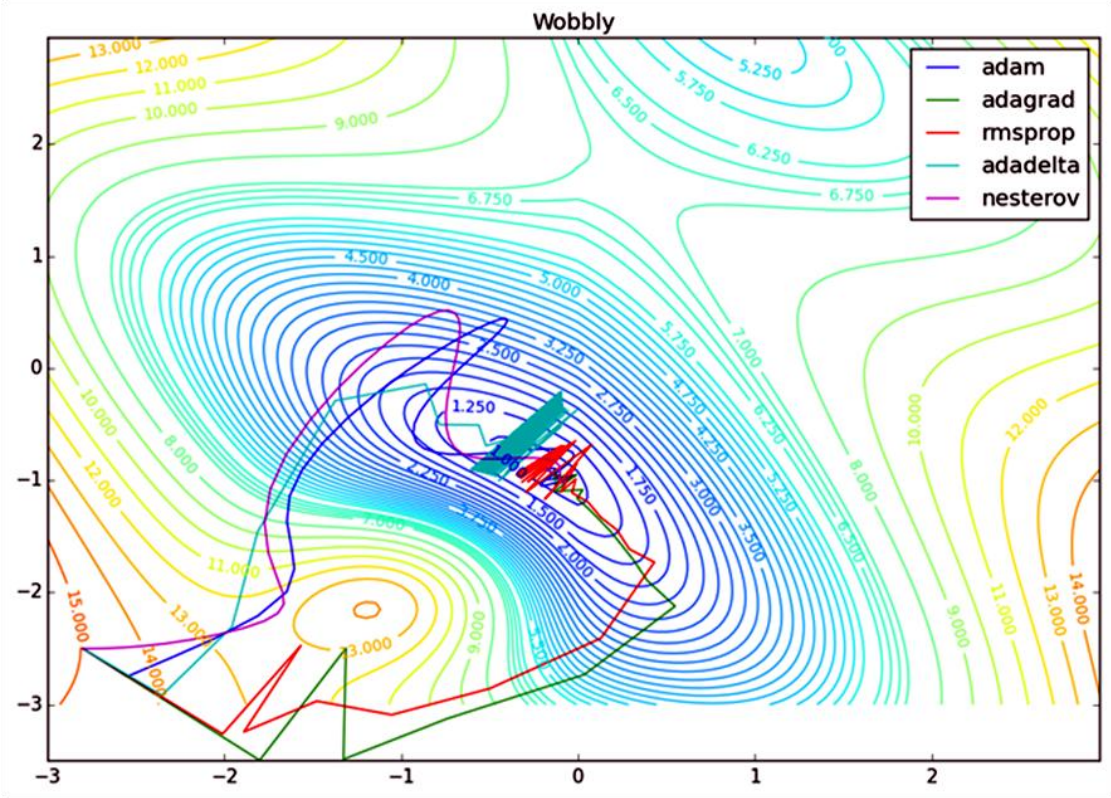
$$= max(\beta_2^{t-1}|g_1|, \beta_2^{t-2}|g_2|, \ldots, \beta_2|g_{t-1}|, |g_t|)$$

$$m_t = \beta_1 m_{t-1} + (1-\beta_1)g_t$$

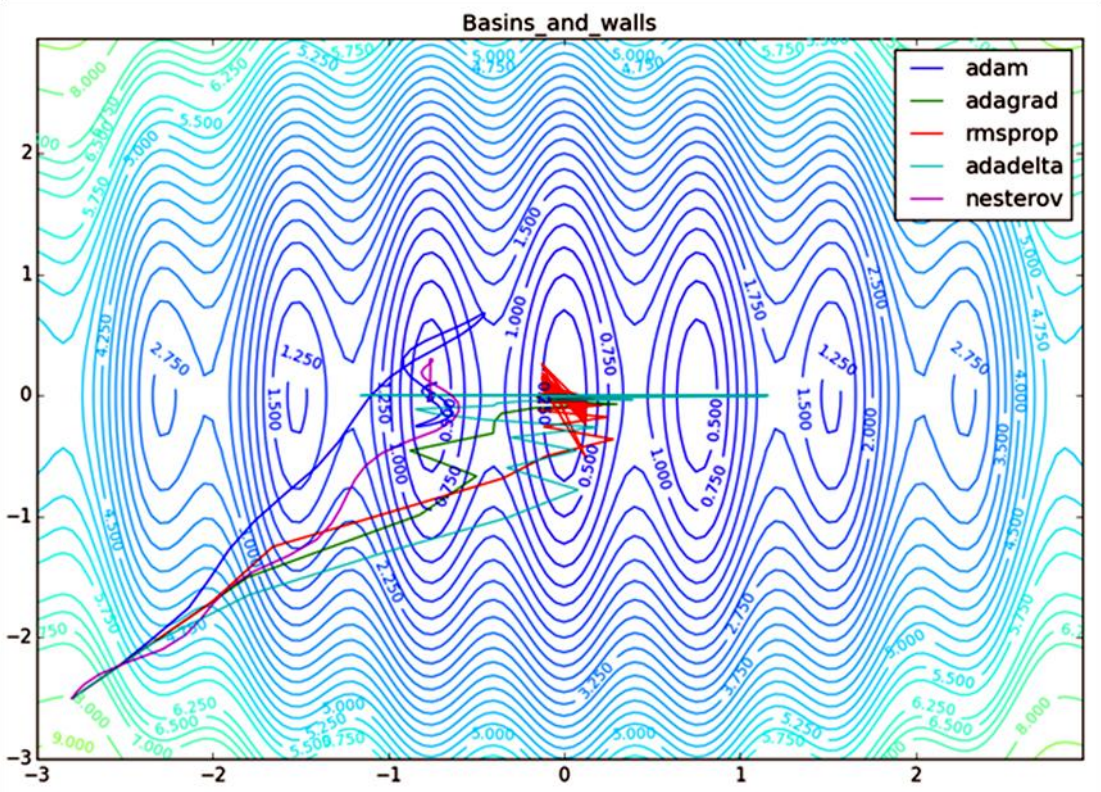
$$\hat{m}_t = \frac{m_t}{1-\beta_1^t}$$

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

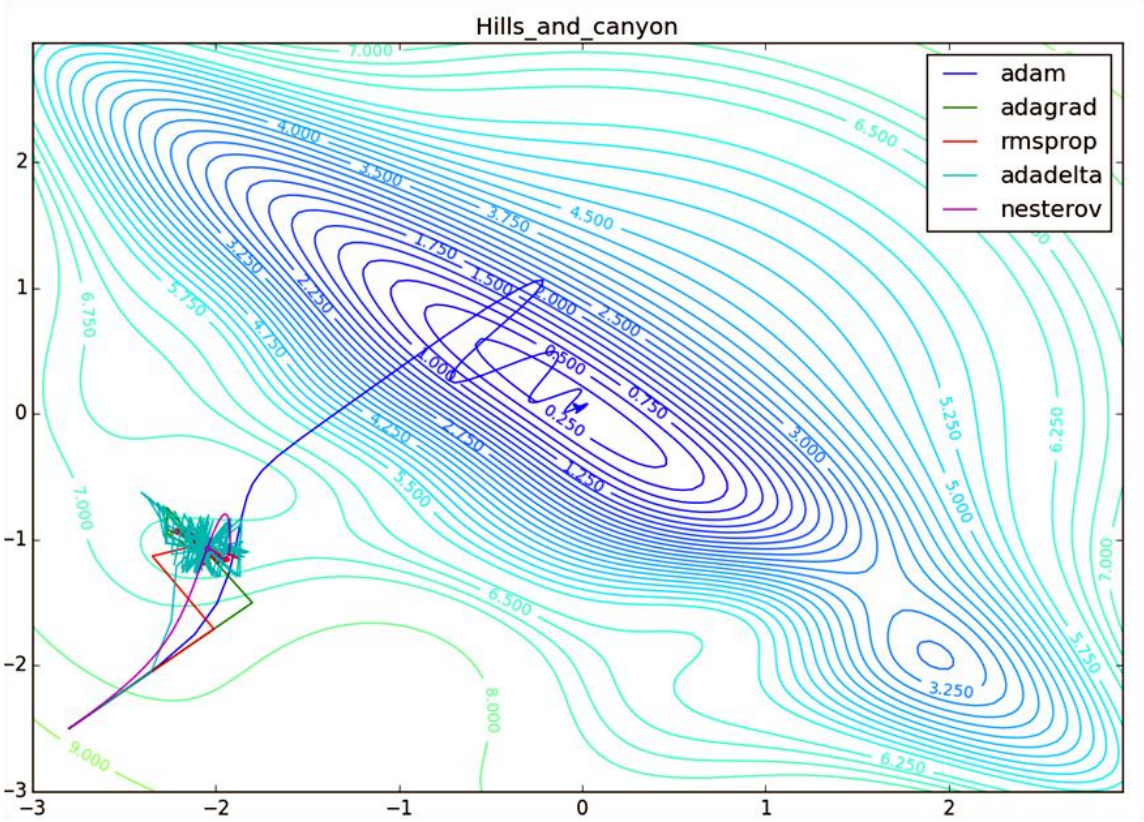
Обзор алгоритмов обучения. Эксперименты (1).



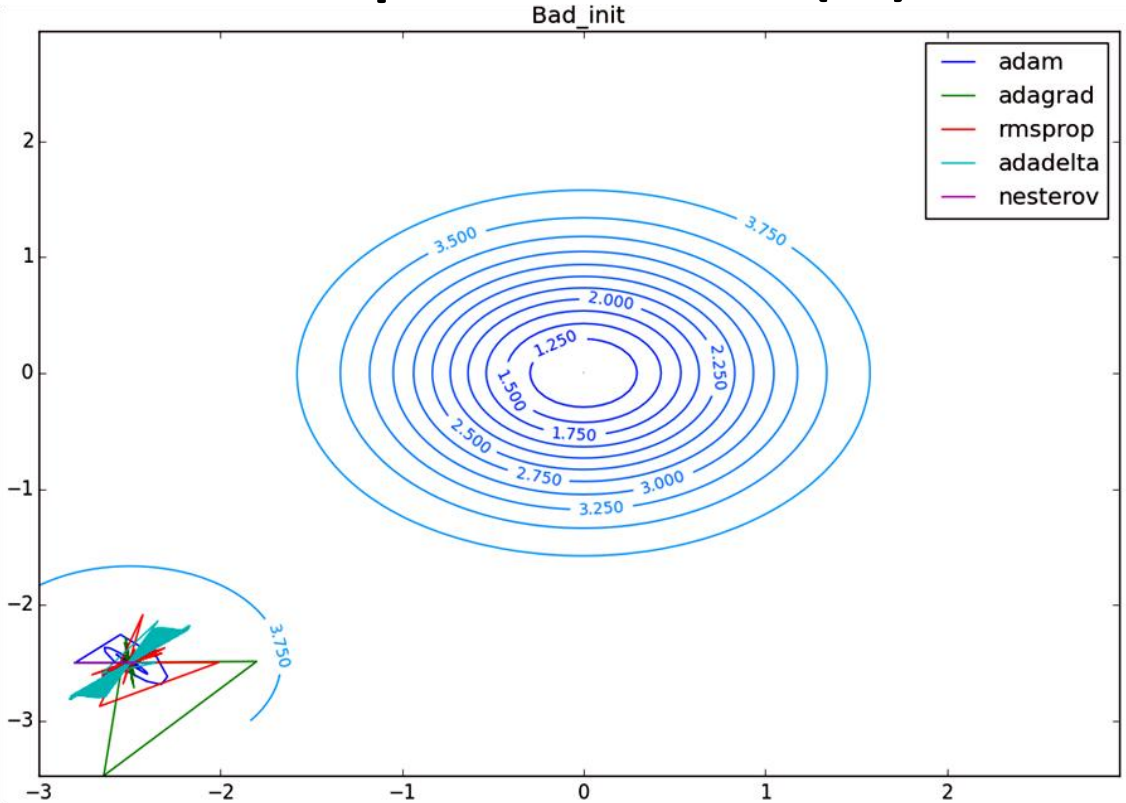
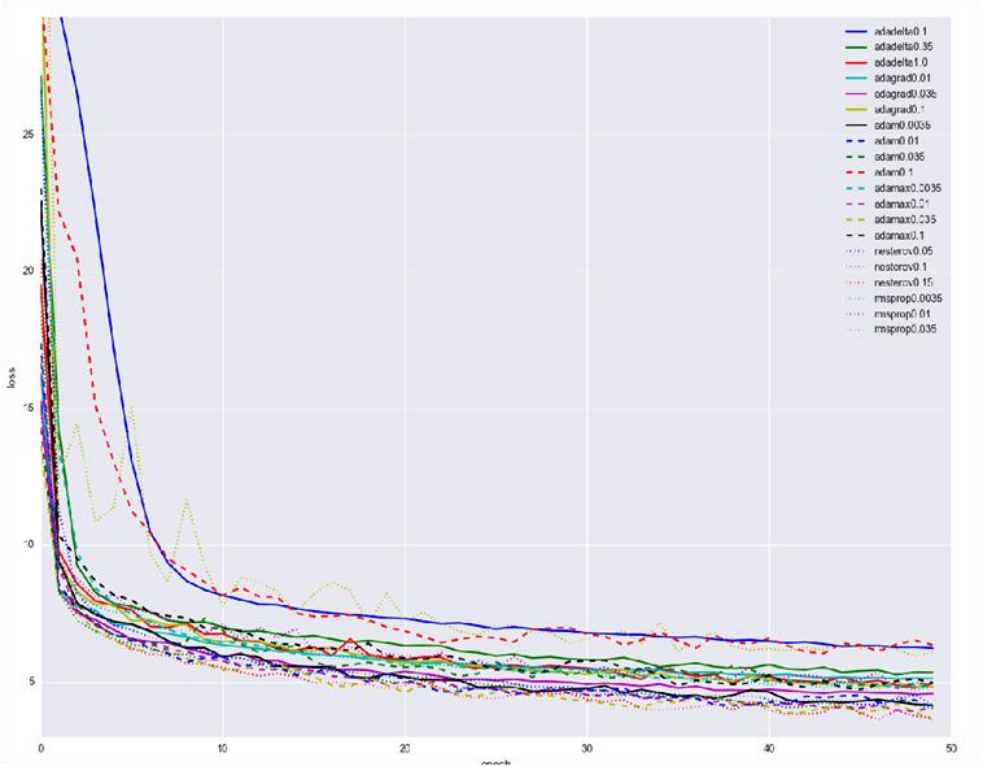
Обзор алгоритмов обучения. Эксперименты (2).



Обзор алгоритмов обучения. Эксперименты (3).



Обзор алгоритмов обучения. Эксперименты (4).



Обзор алгоритмов обучения. Эксперименты (5).