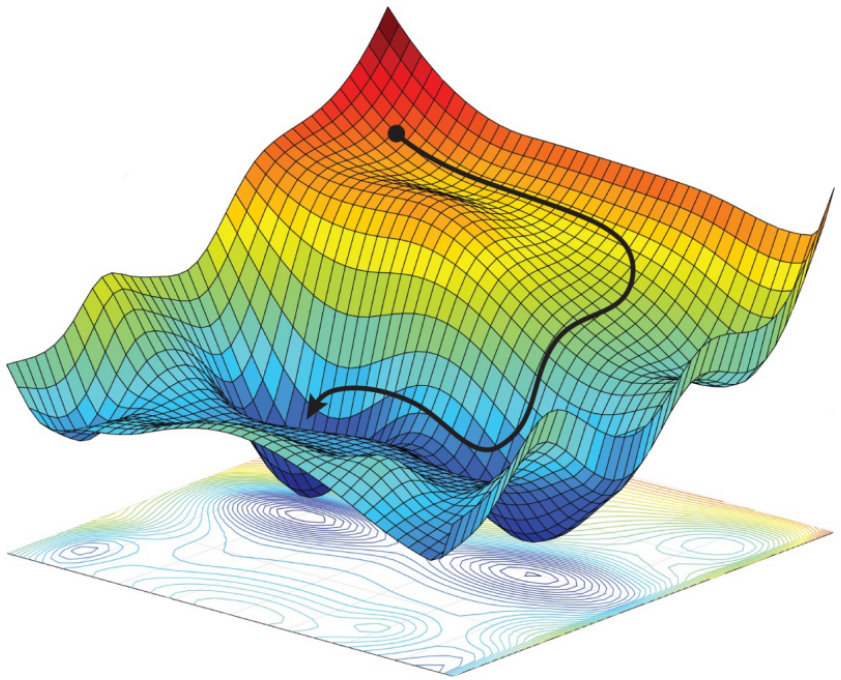
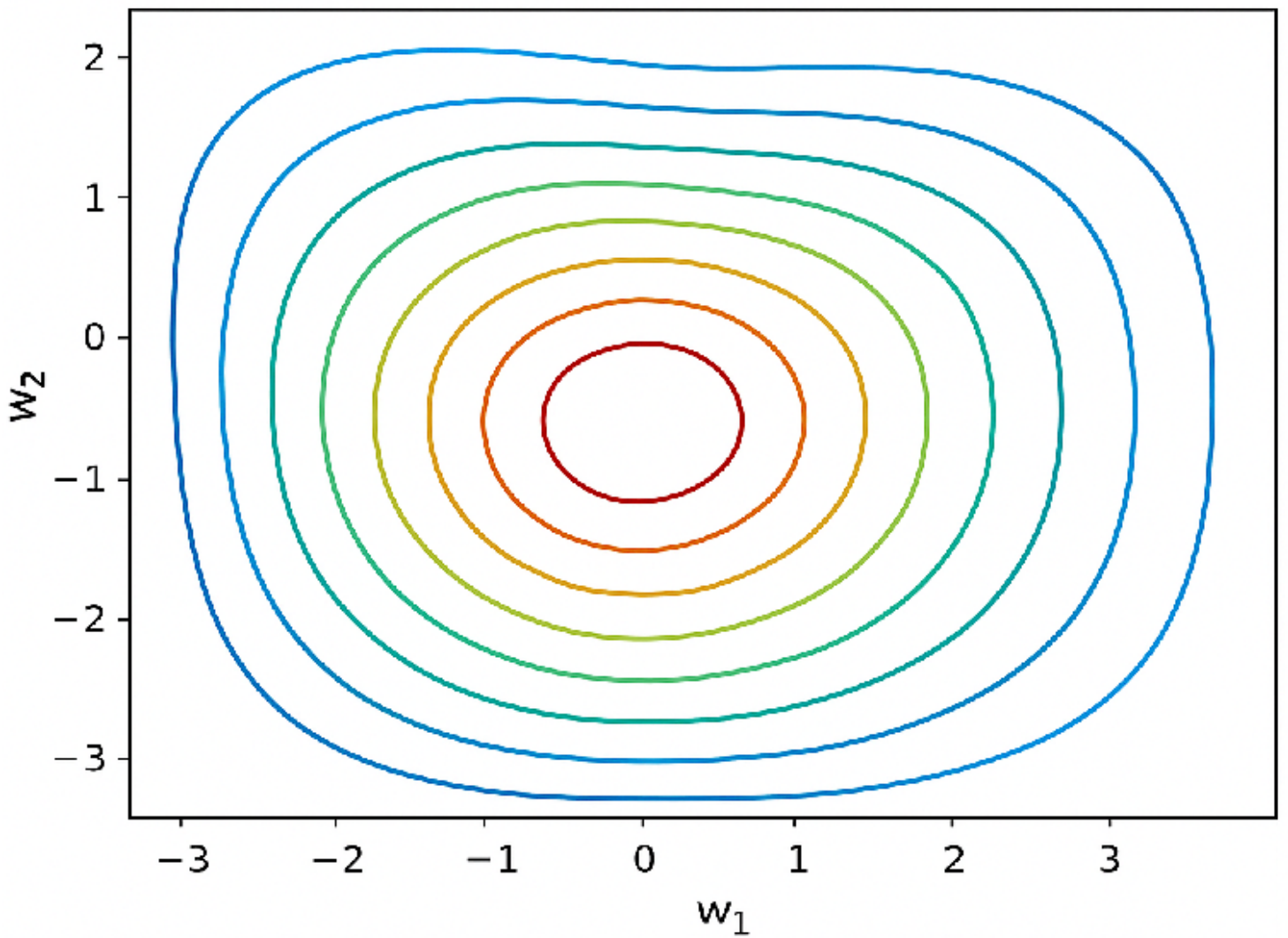
Особенности настройки глубоких нейросетей

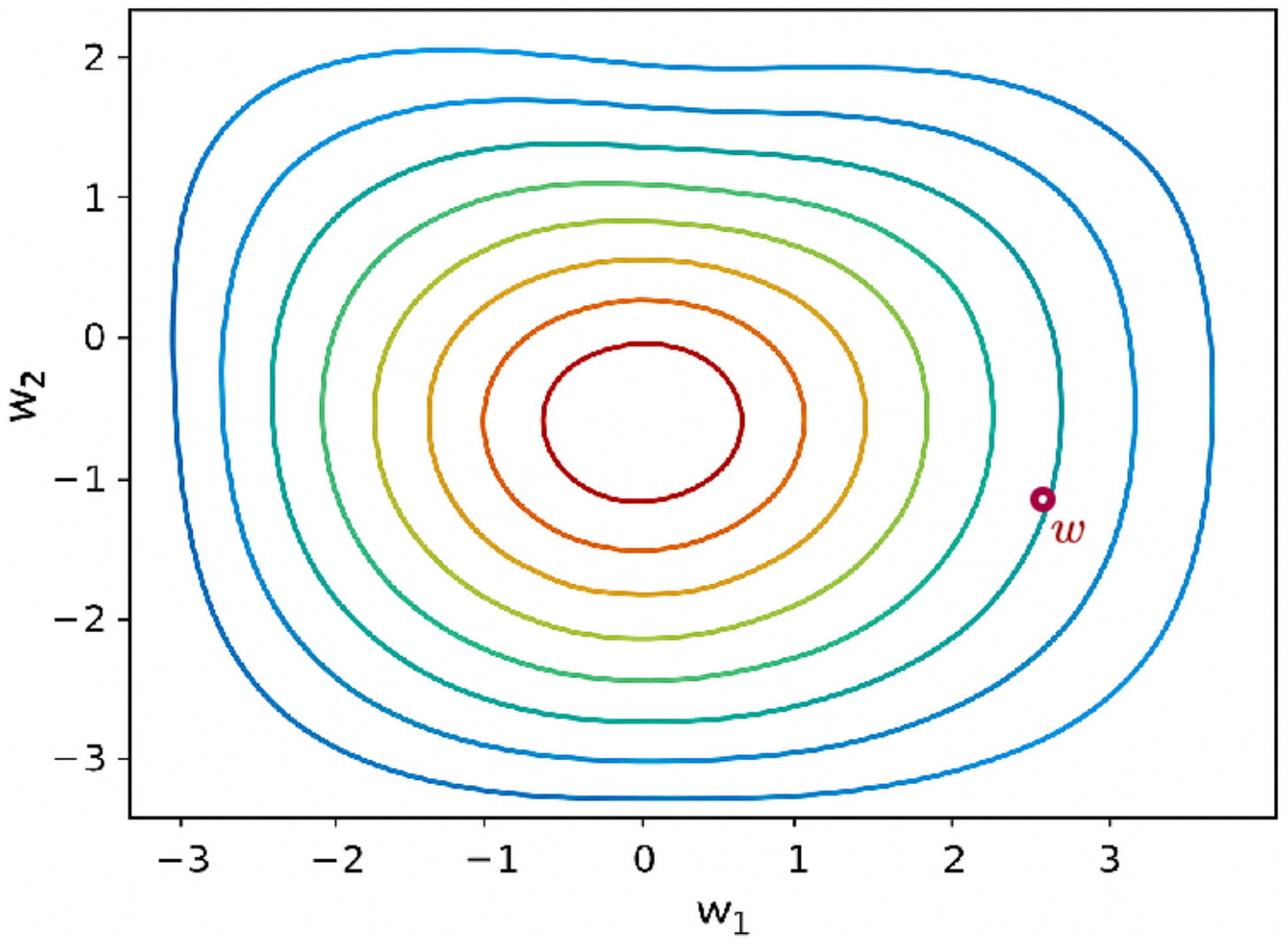
Виктор Китов
victorkitov.github.io



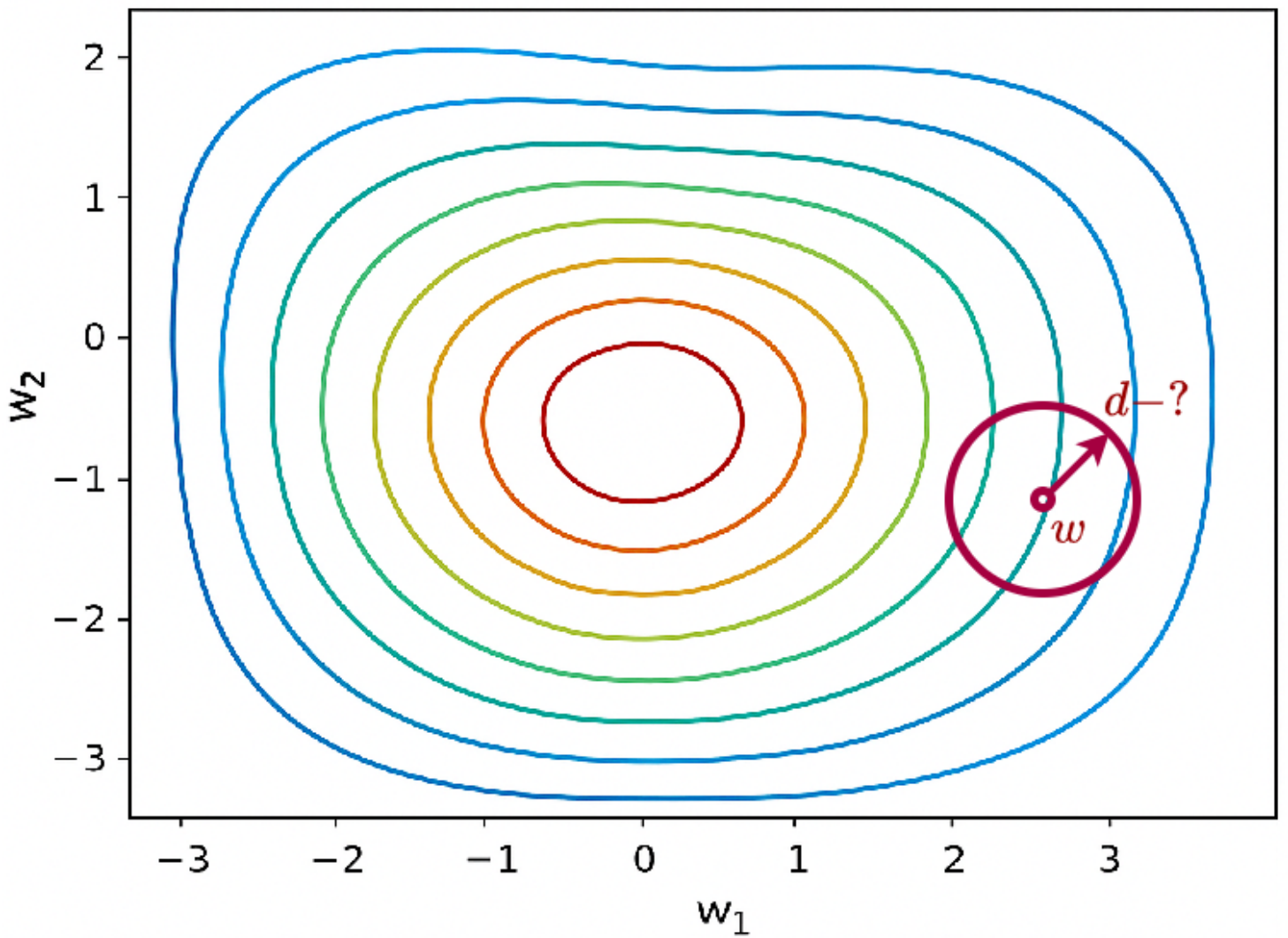
Минимизируемая функция



Текущая оценка весов



Направление максимального изменения, $\|\mathbf{d}\| = 1$



Направление максимальных изменений

Разложение Тейлора 1-го порядка

$$L(\mathbf{w}+\lambda\mathbf{d}) = L(\mathbf{w}) + \nabla L(\mathbf{w})^T(\lambda\mathbf{d}) + o(\lambda), \quad \lambda > 0$$

$$\nabla L(\mathbf{w}) = \left[\frac{\partial L(\mathbf{w})}{\partial w_1}; \frac{\partial L(\mathbf{w})}{\partial w_2}; \ ... \ \frac{\partial L(\mathbf{w})}{\partial w_K}\right]$$

Направление максимальных изменений

Разложение Тейлора 1-го порядка

$$L(\mathbf{w}+\lambda\mathbf{d}) = L(\mathbf{w}) + \nabla L(\mathbf{w})^T(\lambda\mathbf{d}) + o(\lambda), \quad \lambda > 0$$

$$\nabla L(\mathbf{w}) = \left[\frac{\partial L(\mathbf{w})}{\partial w_1}; \frac{\partial L(\mathbf{w})}{\partial w_2}; \ \ldots \ \frac{\partial L(\mathbf{w})}{\partial w_K}\right]$$

Из неравенства Коши-Буняковского при $\|\mathbf{d}\| = 1$:

$$\left|\nabla L(\mathbf{w})^T\mathbf{d}\right| \leq \|\nabla L(\mathbf{w})\|\,\|\mathbf{d}\| = \|\nabla L(\mathbf{w})\|$$

Направление максимальных изменений

Разложение Тейлора 1-го порядка

$$L(\mathbf{w}+\lambda\mathbf{d}) = L(\mathbf{w}) + \nabla L(\mathbf{w})^T(\lambda\mathbf{d}) + o(\lambda), \quad \lambda > 0$$

$$\nabla L(\mathbf{w}) = \left[\frac{\partial L(\mathbf{w})}{\partial w_1}; \frac{\partial L(\mathbf{w})}{\partial w_2}; \ \ldots \ \frac{\partial L(\mathbf{w})}{\partial w_K}\right]$$

Из неравенства Коши-Буняковского при $\|\mathbf{d}\| = 1$:

$$\left|\nabla L(\mathbf{w})^T\mathbf{d}\right| \leq \|\nabla L(\mathbf{w})\|\,\|\mathbf{d}\| = \|\nabla L(\mathbf{w})\|$$

Равенство при $\mathbf{d} \uparrow\downarrow \nabla L(\mathbf{w})$

Направление максимальных изменений

Разложение Тейлора 1-го порядка

$$L(\mathbf{w}+\lambda\mathbf{d}) = L(\mathbf{w}) + \nabla L(\mathbf{w})^T(\lambda\mathbf{d}) + o(\lambda), \quad \lambda > 0$$

$$\nabla L(\mathbf{w}) = \left[\frac{\partial L(\mathbf{w})}{\partial w_1}; \frac{\partial L(\mathbf{w})}{\partial w_2}; \ \ldots \ \frac{\partial L(\mathbf{w})}{\partial w_K}\right]$$

Из неравенства Коши-Буняковского при $\|\mathbf{d}\| = 1$:

$$\left|\nabla L(\mathbf{w})^T\mathbf{d}\right| \leq \|\nabla L(\mathbf{w})\| \, \|\mathbf{d}\| = \|\nabla L(\mathbf{w})\|$$

Равенство при $\mathbf{d} \uparrow\downarrow \nabla L(\mathbf{w})$

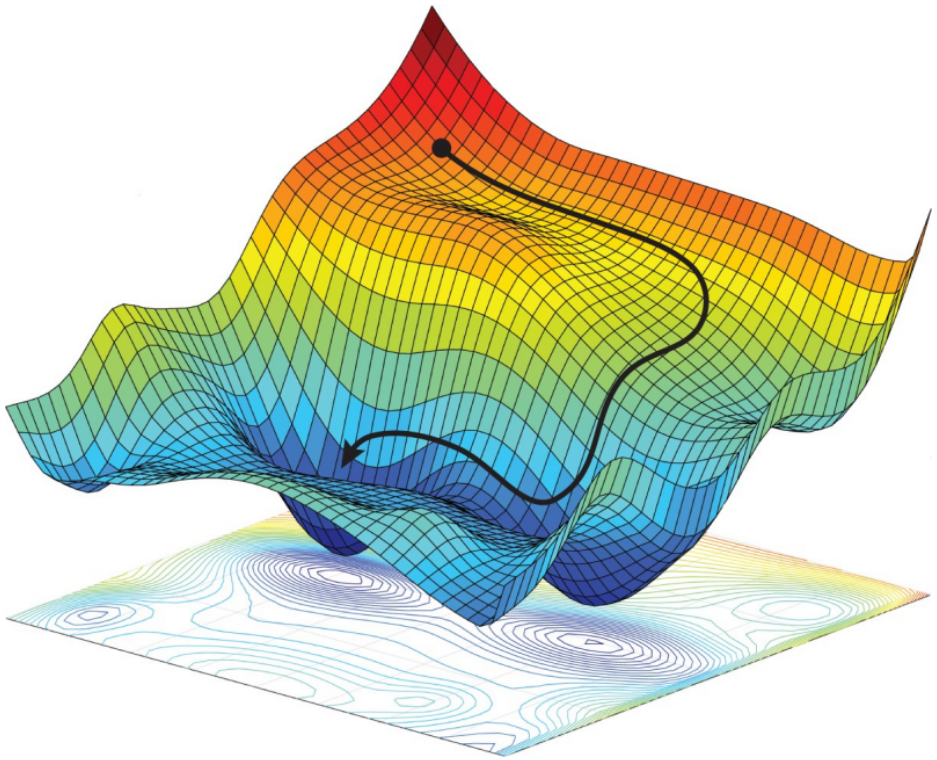
- Максимальное $\uparrow L(\mathbf{w})$ при $\mathbf{d} = \nabla L(\mathbf{w}) / \|\nabla L(\mathbf{w})\|$
- Максимальное $\downarrow L(\mathbf{w})$ при $\mathbf{d} = -\nabla L(\mathbf{w}) / \|\nabla L(\mathbf{w})\|$
- $-\nabla L(\mathbf{w})$ называется антиградиентом

Градиентный спуск: идея

Метод градиентного спуска (gradient descent) - итеративное смещение в направлении антиградиента:

$$\mathbf{w}_{t+1} := \mathbf{w}_t - \varepsilon \nabla_{\mathbf{w}} L\left(\mathbf{w}_t\right)$$

$\varepsilon > 0$ - шаг спуска (learning rate)



Метод Ньютона (квадратичная аппроксимация)

- Рассмотрим минимизацию $L(\mathbf{w}) \to \min_{\mathbf{w}}$.
- $\mathbf{w}^* = \arg\min_w L(\mathbf{w})$, $\nabla L(\mathbf{w}^*) = \mathbf{0}$.

Метод Ньютона (квадратичная аппроксимация)

- Рассмотрим минимизацию $L(\mathbf{w}) \to \min_{\mathbf{w}}$.
- $\mathbf{w}^* = \arg\min_w L(\mathbf{w})$, $\nabla L(\mathbf{w}^*) = \mathbf{0}$.
- Разложение Тейлора $L(\mathbf{w}^*)$ относительно $\mathbf{w}$:

$$\nabla L(\mathbf{w}^*) = 0 = \nabla L(\mathbf{w}) + \nabla^2 L(\mathbf{w})(\mathbf{w}^* - \mathbf{w}) + o(\|\mathbf{w} - \mathbf{w}^*\|)$$

Метод Ньютона (квадратичная аппроксимация)

- Рассмотрим минимизацию $L(\mathbf{w}) \to \min_{\mathbf{w}}$.
- $\mathbf{w}^* = \arg\min_w L(\mathbf{w})$, $\nabla L(\mathbf{w}^*) = \mathbf{0}$.
- Разложение Тейлора $L(\mathbf{w}^*)$ относительно $\mathbf{w}$:

$$\nabla L(\mathbf{w}^*) = 0 = \nabla L(\mathbf{w}) + \nabla^2 L(\mathbf{w})(\mathbf{w}^* - \mathbf{w}) + o(\|\mathbf{w} - \mathbf{w}^*\|)$$

Получаем $\Delta\mathbf{w}$ для перехода в оптимум:

$$\mathbf{w}^* - \mathbf{w} = -\left[\nabla^2 L(\mathbf{w})\right]^{-1} \nabla L(\mathbf{w}) + o(\|\mathbf{w} - \mathbf{w}^*\|)$$

Метод Ньютона (квадратичная аппроксимация)

- Рассмотрим минимизацию $L(\mathbf{w}) \to \min_{\mathbf{w}}$.
- $\mathbf{w}^* = \arg\min_w L(\mathbf{w})$, $\nabla L(\mathbf{w}^*) = \mathbf{0}$.
- Разложение Тейлора $L(\mathbf{w}^*)$ относительно $\mathbf{w}$:

$$\nabla L(\mathbf{w}^*) = 0 = \nabla L(\mathbf{w}) + \nabla^2 L(\mathbf{w})(\mathbf{w}^* - \mathbf{w}) + o(\|\mathbf{w} - \mathbf{w}^*\|)$$

Получаем $\Delta\mathbf{w}$ для перехода в оптимум:

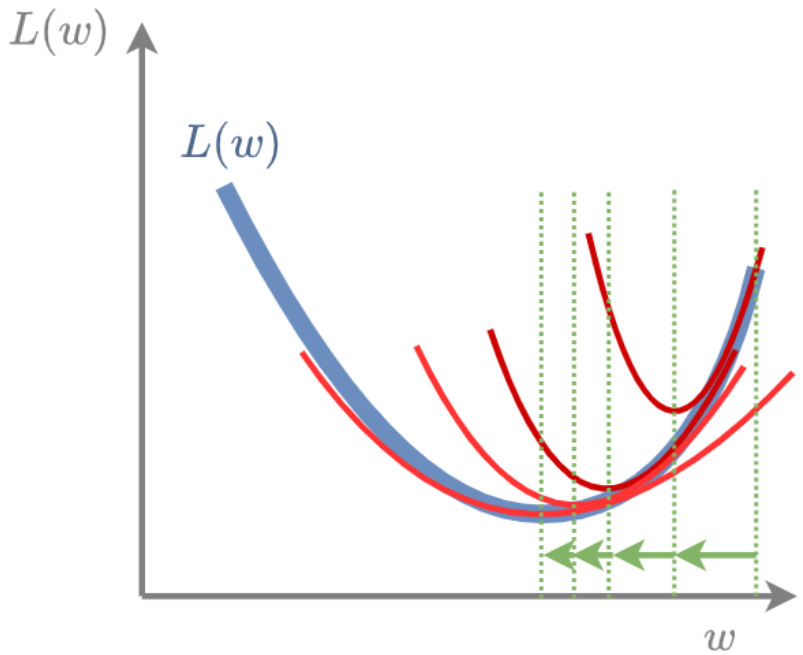
$$\mathbf{w}^* - \mathbf{w} = -\left[\nabla^2 L(\mathbf{w})\right]^{-1} \nabla L(\mathbf{w}) + o(\|\mathbf{w} - \mathbf{w}^*\|)$$

- Отбрасывая $o(\|\mathbf{w} - \mathbf{w}^*\|)$, получаем приближённое правило обновления весов:

$$\mathbf{w}_{t+1} := \mathbf{w}_t - \left[\nabla^2 L(\mathbf{w}_t)\right]^{-1} \nabla L(\mathbf{w}_t)$$

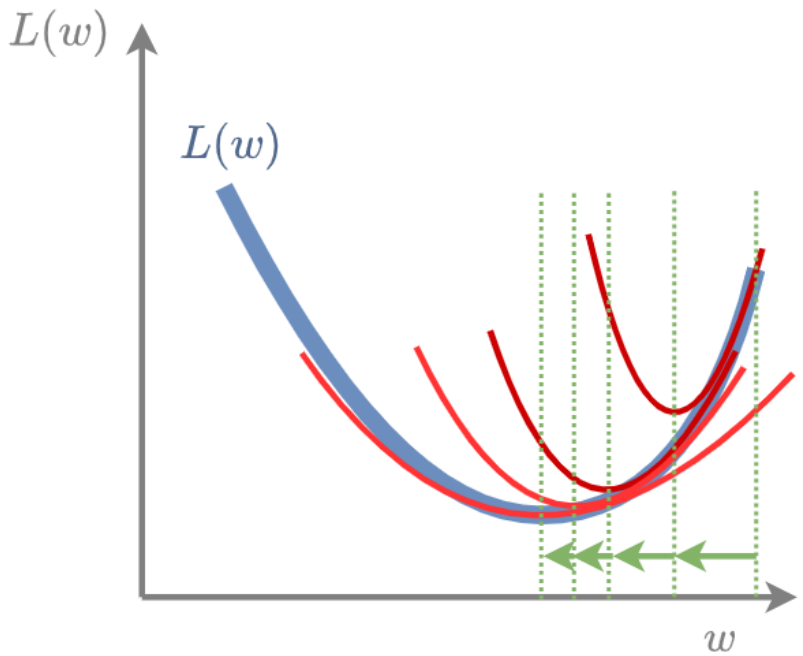
- Это линейно-преобразованный шаг GD.

Геометрическая интерпретация



- Используется **локальная квадратичная аппроксимация**
  - для квадратичного функционала сходится за 1 итерацию

Геометрическая интерпретация



- Используется **локальная квадратичная аппроксимация**
  - для квадратичного функционала сходится за 1 итерацию
- В настройке нейросетей не исп-ся из-за $\left[\nabla^2 L(w)\right]^{-1}$.
  - но для малых и средних моделей применяются аппроксимации, например диагональный Гессиан, BFGS / L-BFGS.

Градиентный спуск: алгоритм

$$\mathbf{w}_{t+1} := \mathbf{w}_t - \varepsilon \nabla_{\mathbf{w}} L\left(\mathbf{w}_t\right)$$

ВХОД:
* $\varepsilon > 0$: шаг одной итерации, контролирующий скорость сходимости
* правило остановки

АЛГОРИТМ:
инициализировать $t = 0$, а w_0 случайно.
ПОКА правило остановки не выполнено:
$w_{t+1} := w_t - \varepsilon \nabla_w L(w_t)$
$t := t + 1$

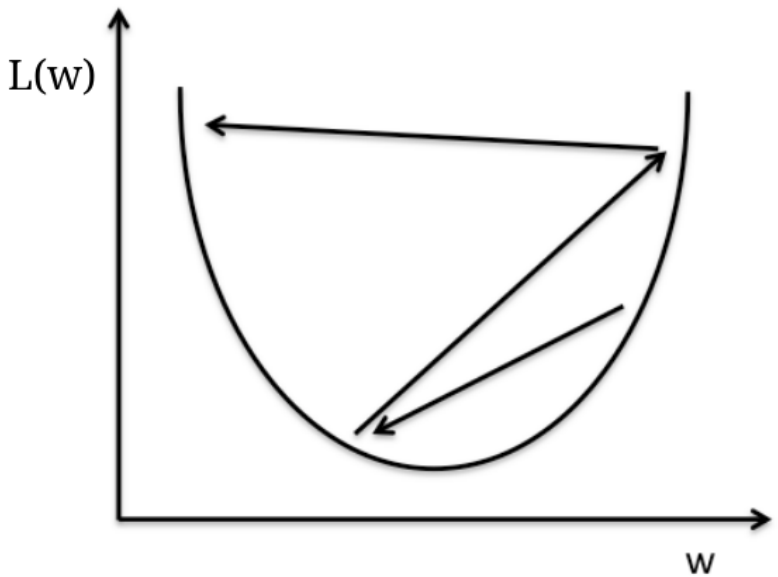
ВЕРНУТЬ w_n

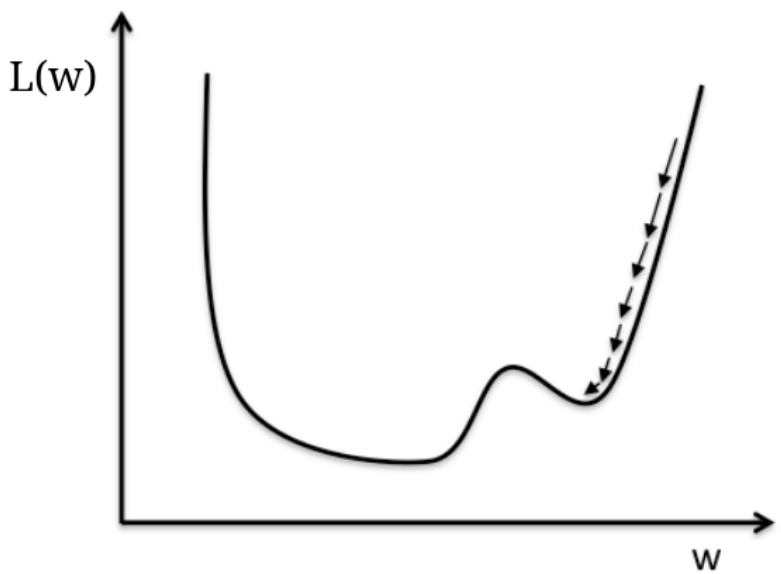
Возможные правила остановки:

- $|L(w_{t+1}) - L(w_t)| < H_1$, $\|w_{t+1} - w_t\| < H_2$, $t > H$.

Выбор шага градиентного спуска

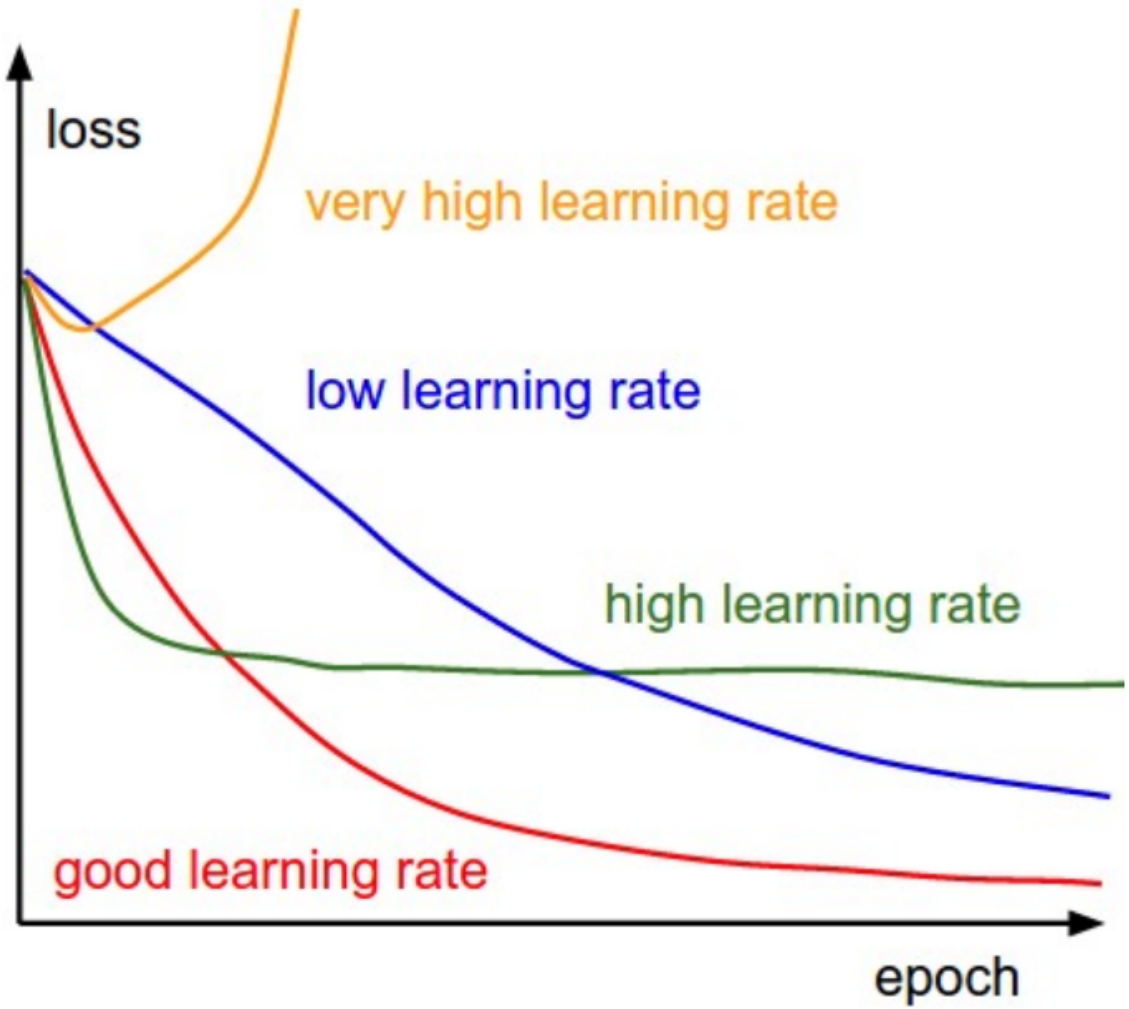
- Большое ε => расходимость
- Малое ε => медленная сходимость





Подбор шага обучения

- Можно начать с большого ε, потом пробовать $\downarrow$



Вычислительная проблема GD

ВХОД:
* $\varepsilon > 0$: шаг одной итерации, контролирующий скорость сходимости
* правило остановки

АЛГОРИТМ:
инициализировать $t = 0$, а w_0 случайно.
ПОКА правило остановки не выполнено:
$w_{t+1} := w_t - \varepsilon \nabla_w L(w_t)$
$t := t + 1$

ВЕРНУТЬ w_n

Вычислительная проблема GD

ВХОД:
* $\varepsilon > 0$: шаг одной итерации, контролирующий скорость сходимости
* правило остановки

АЛГОРИТМ:
инициализировать $t = 0$, а w_0 случайно.
ПОКА правило остановки не выполнено:
$w_{t+1} := w_t - \varepsilon \nabla_w L(w_t)$
$t := t + 1$

ВЕРНУТЬ w_n

$$L(\mathbf{w}) = \frac{1}{N} \sum_{n=1}^{N} \mathcal{L}\left(f_{\mathbf{w}}\left(x_n\right), y_n\right)$$ - вычисляется за $O(N)$

Вычислительная проблема GD

ВХОД:
* $\varepsilon > 0$: шаг одной итерации, контролирующий скорость сходимости
* правило остановки

АЛГОРИТМ:
инициализировать $t = 0$, а w_0 случайно.
ПОКА правило остановки не выполнено:
$w_{t+1} := w_t - \varepsilon \nabla_w L(w_t)$
$t := t + 1$

ВЕРНУТЬ w_n

$$L(\mathbf{w}) = \frac{1}{N}\sum_{n=1}^{N}\mathcal{L}\left(f_{\mathbf{w}}\left(x_n\right), y_n\right) \quad \text{- вычисляется за } O(N)$$

$$\frac{1}{N}\sum_{n=1}^{N}\mathcal{L}\left(f_{\mathbf{w}}\left(x_n\right), y_n\right) \approx \frac{1}{K}\sum_{n\in I}\mathcal{L}\left(f_{\mathbf{w}}\left(x_n\right), y_n\right) \quad \text{- сложность } O(K)$$

Стохастический градиентный спуск

ВХОД:
* $\varepsilon_t > 0$: динамика уменьшения шага
* правило остановки

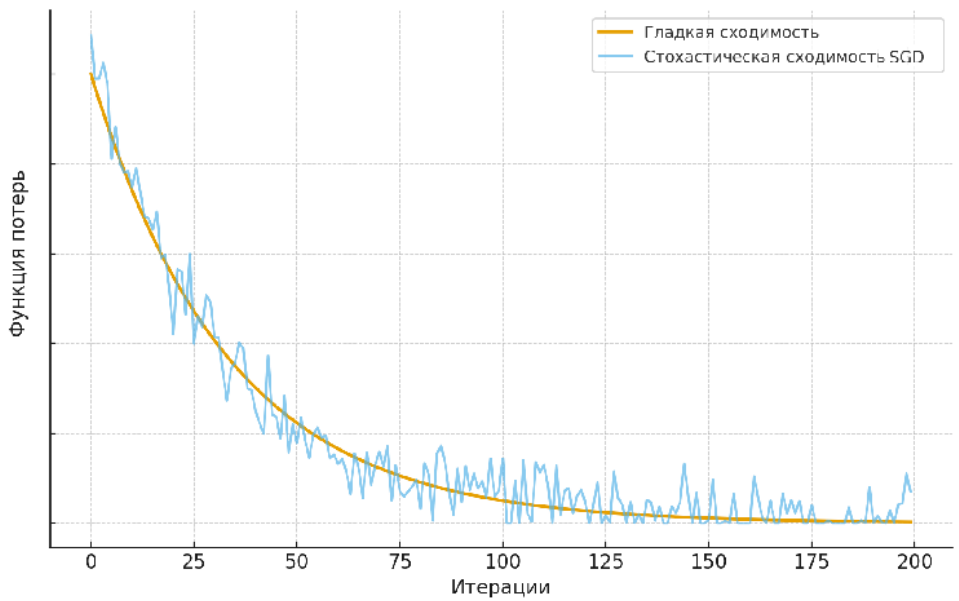
АЛГОРИТМ:
инициализировать $t = 0$, а w_0 случайно
ПОКА не выполнено правило остановки:
    случайно выбрать K объектов $I = \{n_1, ...n_K\}$ из $\{1, 2, ...N\}$
    $w_{t+1} := w_t - \varepsilon_t \frac{1}{K} \sum_{n \in I} \nabla_w \mathcal{L}(f_{w_t}(x_n), y_n)$
    $t := t + 1$

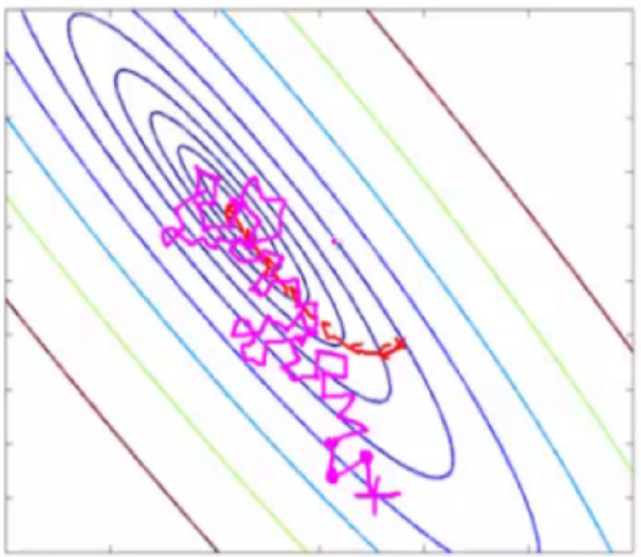
ВЕРНУТЬ w_t

- Минибатч случайный $\Rightarrow$ используется вся выборка.
- Выборку случайно перемешивают на перед каждой эпохой, потом идут последовательно.
- Размер минибатча - максимальный, при котором достигается параллельная обработка.

Стохастическая сходимость

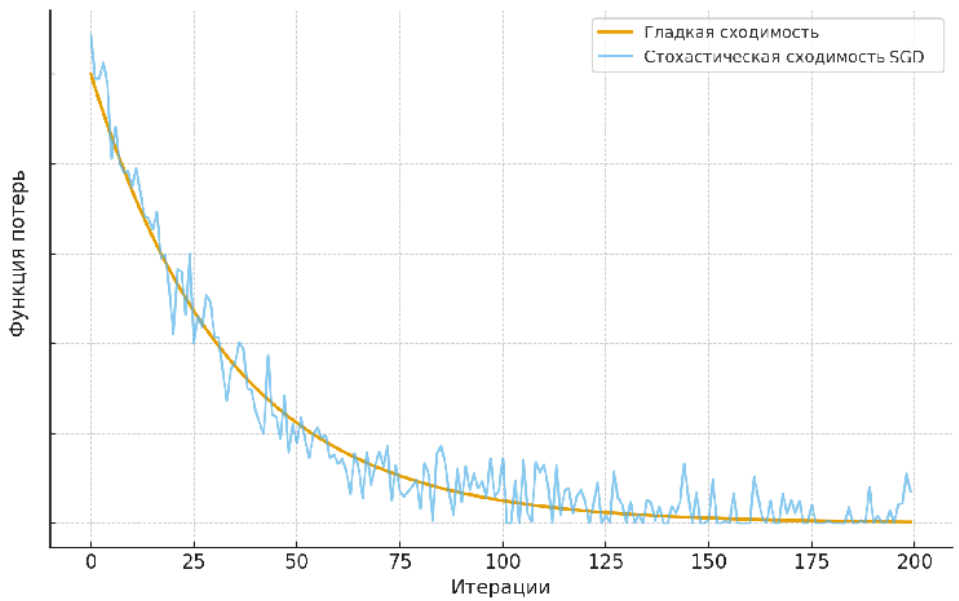
Сходимость уже будет стохастической:

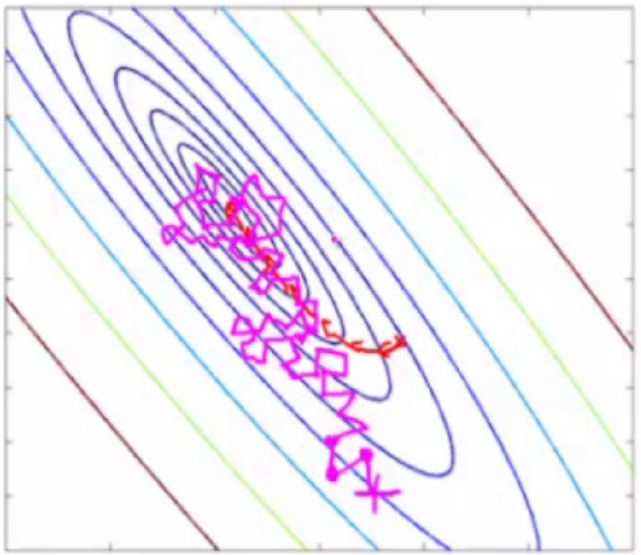




Стохастическая сходимость

Сходимость уже будет стохастической:





Поэтому шаг обучения необходимо **постепенно уменьшать**!

Условия сходимости

- Условия сходимости к оптимуму[1]:

$$\sum_t \varepsilon_t = +\infty \qquad \text{достигаем произвольной точки}$$

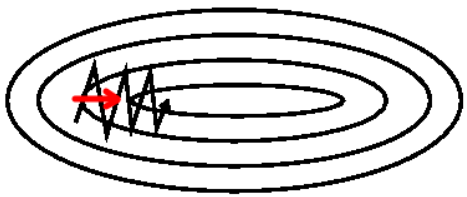
$$\sum_t \varepsilon_t^2 < +\infty \qquad \text{сумма ошибок аппр-ции не уводит в сторону}$$

- Нужно задать расписание $\downarrow \varepsilon_t$ (learning rate schedule).
- Динамическое расписание - $\downarrow \varepsilon_t$ после выхода на плато (reduce on plateu).

[1] Приведите пример ε_t, для которого эти условия выполнены.

SGD с инерцией

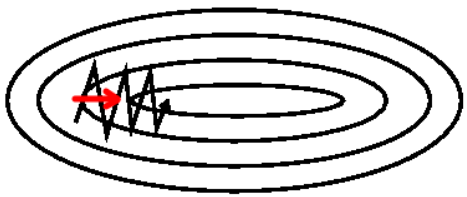
- Хотим $\uparrow$ стабильность оценок $\nabla L(\mathbf{w})$ без $\uparrow$ размера минибатча:



- Тогда мы могли бы $\uparrow \varepsilon_t$ и ускорить сходимость.

SGD с инерцией

- Хотим $\uparrow$ стабильность оценок $\nabla L(\mathbf{w})$ без $\uparrow$ размера минибатча:



- Тогда мы могли бы $\uparrow \varepsilon_t$ и ускорить сходимость.
- SGD с инерцией (momentum):

$$v_t := \gamma v_{t-1} + \varepsilon \nabla L(w_t)$$
$$w_{t+1} := w_t - v_t$$

SGD с инерцией

- Хотим ↑ стабильность оценок $\nabla L(\mathbf{w})$ без ↑ размера минибатча:

- Тогда мы могли бы ↑ ε_t и ускорить сходимость.
- SGD с инерцией (momentum):

$$v_t := \gamma v_{t-1} + \varepsilon \nabla L(w_t)$$
$$w_{t+1} := w_t - v_t$$

- Устойчивее градиент: усредняем с градиентами прошлых итераций.

SGD с инерцией

- Участвуют градиенты со всех предыдущих итераций с эксп. $\downarrow$ весами:

$$v_t := \gamma v_{t-1} + \varepsilon \nabla L(w_t) =$$
$$= \gamma(\gamma v_{t-2} + \varepsilon \nabla L(w_{t-1})) + \varepsilon \nabla L(w_t)$$
$$= \gamma^2 v_{t-2} + \varepsilon\gamma \nabla L(w_{t-1}) + \varepsilon \nabla L(w_t)$$
$$= \gamma^2(\gamma v_{t-3} + \varepsilon \nabla L(w_{t-2})) + \varepsilon\gamma \nabla L(w_{t-1}) + \varepsilon \nabla L(w_t)$$
$$= \gamma^3 v_{t-3} + \varepsilon\gamma^2 \nabla L(w_{t-2}) + \varepsilon\gamma \nabla L(w_{t-1}) + \varepsilon \nabla L(w_t)$$
$$\dots$$

- Аналогия: "мяч катится с горы".
- γ управляет противоречием между стабильностью и актуальностью градиентов.
  - по умолчанию $\gamma = 0.9$.

Инерция Нестерова

- Хотим повысить актуальность весов в

$$v_t := \gamma v_{t-1} + \varepsilon \nabla L(w_t)$$
$$w_{t+1} := w_t - v_t$$

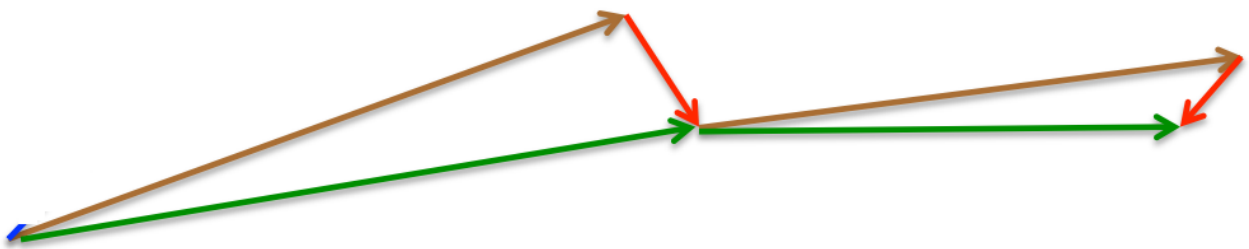
Инерция Нестерова

- Хотим повысить актуальность весов в

$$v_t := \gamma v_{t-1} + \varepsilon \nabla L(w_t)$$
$$w_{t+1} := w_t - v_t$$

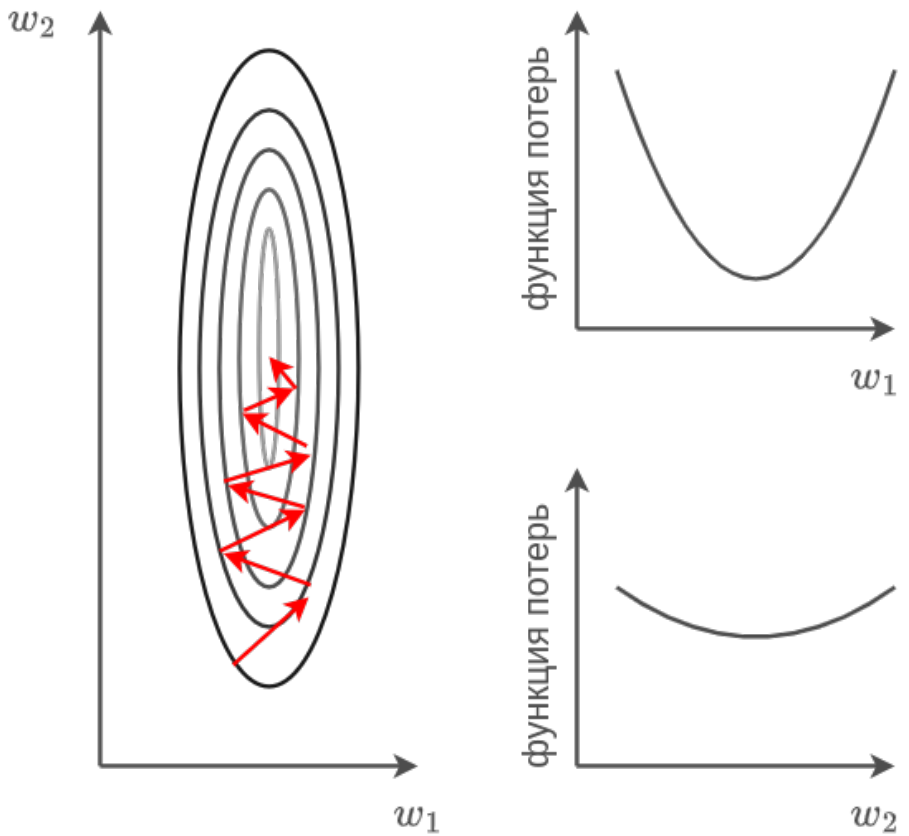
- Nesterov Accelerated Gradient (Nesterov Momentum)

$$v_t := \gamma v_{t-1} + \varepsilon \nabla L(w_t - \gamma v_{t-1})$$
$$w_{t+1} := w_t - v_t$$



Проблема методов

- Все рассмотренные методы предполагают одинаковость ε_t вдоль разных осей.
- При этом оптимальная скорость сходимости для каждой оси будет своя.



AdaGrad, RMSprop

- **AdaGrad**:

$$G_t := G_t + \nabla_w L(w_t)^2$$
$$w_{t+1} := w_t - \frac{\varepsilon}{\sqrt{G_t + s}} \cdot \nabla_w L(w_t)$$

- Операции над векторами поэлементные.
- $s = \text{const} = 10^{-6}$ чтобы не делить на ноль.
- Проблема?

AdaGrad, RMSprop

- **AdaGrad**:

$$G_t := G_t + \nabla_w L(w_t)^2$$
$$w_{t+1} := w_t - \frac{\varepsilon}{\sqrt{G_t + s}} \cdot \nabla_w L(w_t)$$

- Операции над векторами поэлементные.
- $s = \text{const} = 10^{-6}$ чтобы не делить на ноль.
- Проблема?
- Помним большие $\nabla_w L(w_t)^2$, которые были давно.
- **RMSprop**:

$$G_t := \gamma G_{t-1} + (1-\gamma)\nabla_w L(w_t)^2$$
$$w_{t+1} := w_t - \frac{\varepsilon}{\sqrt{G_t + s}} \cdot \nabla_w L(w_t)$$

Adam (упрощённый)

- Adam=RMSprop+инерция:

$$m_t := \beta_1 m_{t-1} + (1-\beta_1)\nabla_w L(w_t)$$
$$G_t := \beta_2 G_{t-1} + (1-\beta_2)\nabla_w L(w_t)^2$$
$$w_{t+1} := w_t - \frac{\varepsilon}{\sqrt{G_t}+s}\cdot m_t$$

- Значения по умолчанию: $\beta_1 = 0.9,\ \beta_2 = 0.999,\ s = 10^{-8}$.

Adam (реальный)

- Реальный Adam чуть сложнее:

$$m_t := \beta_1 m_{t-1} + (1-\beta_1)\nabla_w L(w_t)$$
$$G_t := \beta_2 G_{t-1} + (1-\beta_2)\nabla_w L(w_t)^2$$
$$\widehat{m}_t = \frac{m_t}{1-\beta_1^t}$$
$$\widehat{G}_t = \frac{G_t}{1-\beta_2^t}$$
$$w_{t+1} := w_t - \frac{\varepsilon}{\sqrt{\widehat{G}_t + s}} \cdot \widehat{m}_t$$

- $m_t \to \widehat{m}_t,\ G_t \to \widehat{G}_t$, чтобы избавиться от смещения, вызванного $m_0 = G_0 = 0$, при $t > 30$: $m_t \approx \widehat{m}_t,\ G_t \approx \widehat{G}_t$.
- Самый популярный способ оптимизации!
- Nadam: Adam+Nesterov Accelerated Gradient.

AdamW[2]

- При L2-регуляризации $\tilde{L}(w) = L(w) + \lambda \|w\|^2$сильно влияющие веса регуляризуются слабее, чем слабо влияющие, из-за разного шага обучения.
- Хотим **равномерность регуляризации**.

[2] https://arxiv.org/pdf/1711.05101

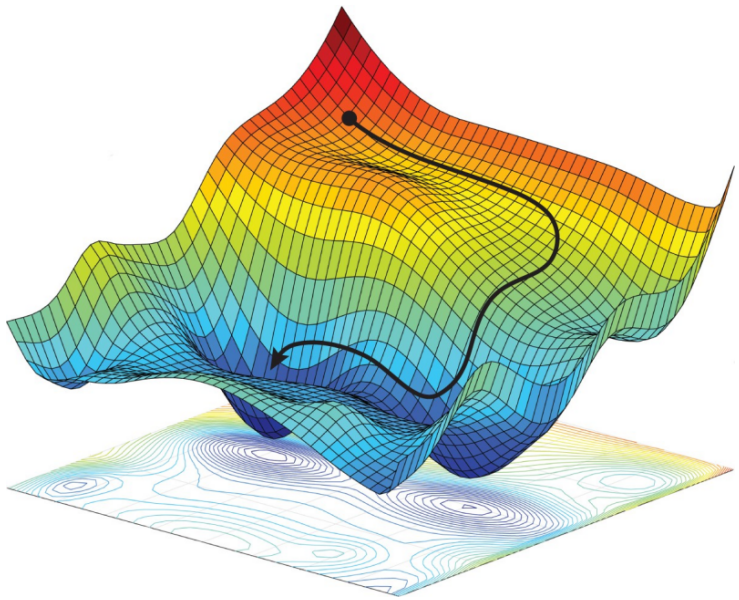
AdamW[2]

- При L2-регуляризации $\tilde{L}(w) = L(w) + \lambda \|w\|^2$сильно влияющие веса регуляризуются слабее, чем слабо влияющие, из-за разного шага обучения.
- Хотим **равномерность регуляризации**.
- AdamW - Adam (по $L(w)$)+независимый учёт рег-ции:

$$m_t := \beta_1 m_{t-1} + (1-\beta_1)\nabla_w L(w_t)$$
$$G_t := \beta_2 G_{t-1} + (1-\beta_2)\nabla_w L(w_t)^2$$
$$\widehat{m}_t = \frac{m_t}{1-\beta_1^t}$$
$$\widehat{G}_t = \frac{G_t}{1-\beta_2^t}$$
$$w_{t+1} := w_t - \frac{\varepsilon}{\sqrt{\widehat{G}_t + s}} \cdot \widehat{m}_t - \varepsilon\,(2\lambda w_t)$$

[2] https://arxiv.org/pdf/1711.05101

Множественность оптимумов



- ML: линейные методы на выпуклых $\mathcal{L}(\mathbf{w})$ => $L(\mathbf{w})$-выпуклая => лок. минимум является глобальным
- DL: $\mathcal{L}(\mathbf{w})$ не выпуклая, сходимся в **локальный** оптимум
  - зависит от $\mathbf{w}_0$, ε, метода оптимизации, его настроек
  - можно перезапустить несколько раз и
    - выбрать лучшее решение
    - сделать ансамбль

Заключение

- Нейросети настраиваются градиентными методами оптимизации 1го порядка.
  - для 2го порядка высокие накладные расходы
- GD не используется из-за сложности $O(N)$ при вычислении $\nabla L(w)$.
- Используются модификации GD:
  - SGD, SGD+momentum, SGD+Nesterov momentum.
- Ускорить сходимость за счёт адаптации ε_t вдоль осей позволяют:
  - AdaGrad, RMSprop, Adam, Nadam, AdamW.
- При настройке есть риск найти неоптимальный локальный минимум.