ТЕХНОСФЕРА

Лекция 7
Методы оптимизации

Байгушев Данила

19 марта 2021 г.

Постановка задачи

- $\theta_* = min_\theta J(\theta)$
- В любой точке можем вычислить $\nabla_\theta J(\theta)$

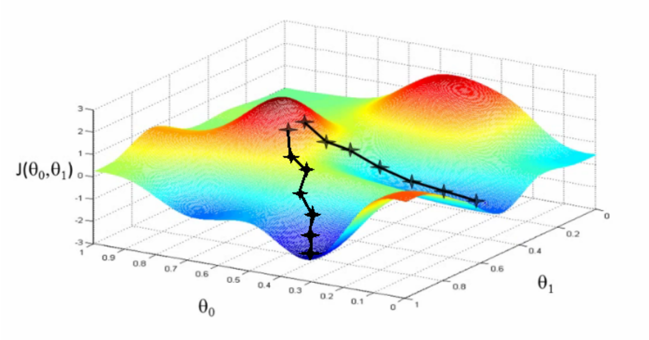


Figure: Пример функции для оптимизации

Batch Gradient Descend

Формула пересчета:

$$\theta_t = \theta_{t-1} - \eta_t \nabla_\theta J(\theta_{t-1})$$

- Требуется обработать все объекты для одного шага
- Нет режима online обучения

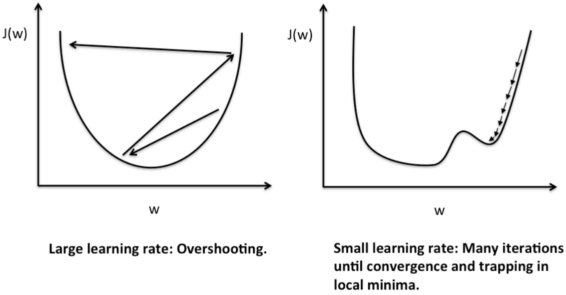


Figure: Выбор темпа обучения

SGD / Mini-batch SGD

- Какие функции оптимизируем?

SGD / Mini-batch SGD

- Какие функции оптимизируем?
- Большие суммы функций: $J(\theta) = \sum_{i=1}^{N} J_i(\theta)$
- Формула пересчета: $\theta_t = \theta_{t-1} - \eta_t \nabla_\theta J_i(\theta_{t-1})$
- Mini-batch SGD: $\theta_t = \theta_{t-1} - \eta_t \sum_{i \in \{i_1, i_2, \ldots, i_k\}} \nabla_\theta J_i(\theta_{t-1})$
- Легко попасть в регион неопределенности, тяжело найти общий оптимум

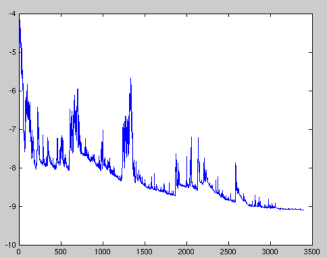


Figure: Измененние значения J во время обучения

SGD / Mini-batch SGD

Для выпуклых функций гарантируется сходимость, если:

$$\eta_t \xrightarrow{t \to \infty} 0$$

$$\sum_{t=0}^{\infty} \eta_t = \infty$$

$$\sum_{t=0}^{\infty} \eta_t^2 < \infty$$

Ландшафт функции потерь[1]

Почему мы вообще используем градиентный спуск?

[1] https://habr.com/ru/post/351924/

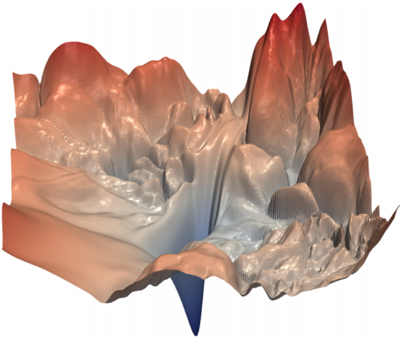
Ландшафт функции потерь[1]

Почему мы вообще используем градиентный спуск?

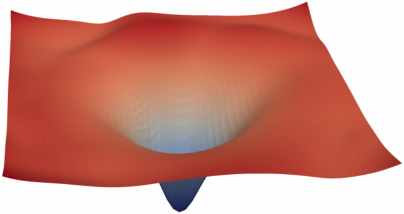
- Большинство локальных минимумов целевой функции сконцентрированы в сравнительно небольшом подпространстве весов. Соответствующие этим минимумам сети дают примерно одинаковый loss на тестовом датасете.
- Сложность ландшафта увеличивается по приближении к глобальным минимумам. Почти во всём объёме пространства весов подавляющая часть седловых точек имеет большое количество направлений, по которым из них можно сбежать. Чем ближе к центру кластера минимумов, тем меньше «направлений побега» у встреченных на пути седловых точек.
- В глубоких нейронных сетях основным препятствием для обучения являются седловые точки, а не локальные минимумы, как считалось ранее.

[1] https://habr.com/ru/post/351924/

Ландшафт функции потерь[2]



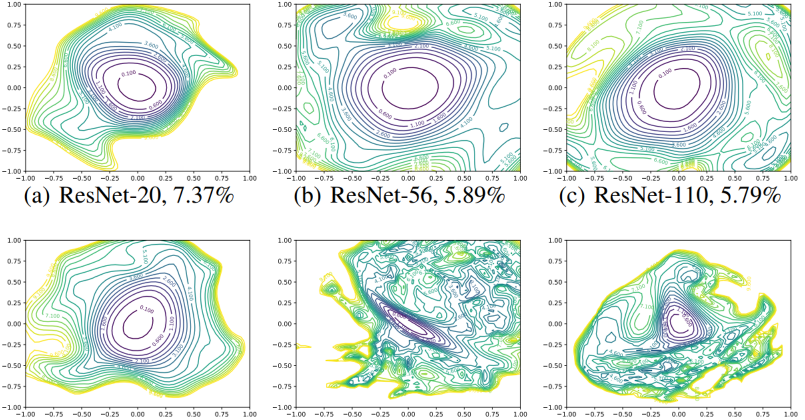
(a) without skip connections



(b) with skip connections

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

Ландшафт функции потерь[3]



(a) ResNet-20, 7.37% (b) ResNet-56, 5.89% (c) ResNet-110, 5.79%

(d) ResNet-20-NS, 8.18% (e) ResNet-56-NS, 13.31% (f) ResNet-110-NS, 16.44%

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

Momentum

- $\nu_t = \gamma \nu_{t-1} + \eta_t \nabla_\theta J(\theta_{t-1}) \leftarrow$ “инерция”
- $\theta_t = \theta_{t-1} - \nu_t$
- Рекомендовано брать $\gamma = 0.9$
- Проблема: метод приводит к перескокам через локальный минимум

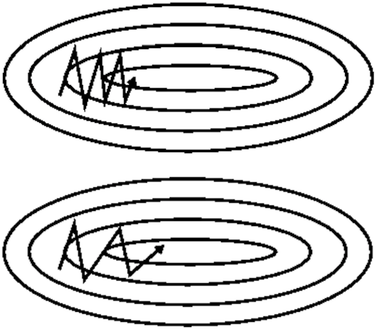


Figure: Слева: без моментума, справа: с моментумом

Nesterov accelerated gradient

- Следующая позиция приближенно равна $\theta_{t-1} - \gamma\nu_{t-1}$
- Вычисление градиента в новой точке дает возможность скорректировать направление движения
- $\nu_t = \gamma\nu_{t-1} + \eta_t \nabla_\theta J(\theta_{t-1} - \gamma\nu_{t-1})$
- $\theta_t = \theta_{t-1} - \nu_t$

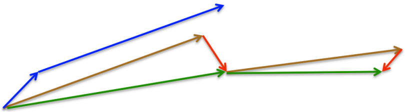


Figure: NAG[4]. brown = jump; red = correction, green = accumulated gradient; blue vectors = standard momentum

- Сначала делаем шаг в направлении накопленного градиента
- Затем вычисляем градиент там и делаем поправку

[4] http://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf

Методы

- SGD $\nu_t = \eta_t \nabla_\theta J_i(\theta_{t-1})$
  Momentum $\nu_t = \gamma \nu_{t-1} + \eta_t \nabla_\theta J(\theta)$
  NAG $\nu_t = \gamma \nu_{t-1} + \eta_t \nabla_\theta J(\theta - \gamma \nu_{t-1})$
- $\theta = \theta - \nu_t$
- Общая проблема: одинаковый шаг для всех параметров
- Трудно подобрать η_t
- Примеры расписаний: $\eta_t = \gamma^t \eta_0$, $\eta_t = \begin{cases} \alpha_1 & t \leq A \\ \alpha_2 & t > A \end{cases}$

Adagrad

- $g_{t,i} = \nabla_{\theta_i} J(\theta)$
- $\theta_{t+1,i} = \theta_{t,i} - \frac{\eta}{\sqrt{G_{t,ii}+\epsilon}} \cdot g_{t,i}$
- $G_{t,ii}$ – сумма квадратов значений $g_{t,i}$ вплоть до текущего
- Стандартные значения: $\eta = 0.01$, $\epsilon = 10^{-8}$
- Мотивация: маленькие обновления для часто встречающихся параметров, большие для редких

? Какова проблема этого метода?

Adagrad

- $g_{t,i} = \nabla_{\theta_i} J(\theta)$
- $\theta_{t+1,i} = \theta_{t,i} - \frac{\eta}{\sqrt{G_{t,ii} + \epsilon}} \cdot g_{t,i}$
- $G_{t,ii}$ – сумма квадратов значений $g_{t,i}$ вплоть до текущего
- Стандартные значения: $\eta = 0.01$, $\epsilon = 10^{-8}$
- Мотивация: маленькие обновления для часто встречающихся параметров, большие для редких

? Какова проблема этого метода?
$G_{t,ii}$ не убывает $\Rightarrow$ затухание обновлений

RMSProp / Adadelta

- Будем использовать последние несколько значений g_t^2 для подсчета G_t
- Экспоненциальное среднее: $E[g^2]_t = \gamma E[g^2]_{t-1} + (1-\gamma) g_t^2$, $\gamma = 0.9$
- $\theta_t = \theta_{t-1} - \Delta\theta_t$
- $\Delta\theta_t = \frac{\eta}{\sqrt{E[g^2]_t + \epsilon}} g_t = \frac{\eta}{RMS[g]_t} g_t \leftarrow$ RMSprop
- Adadelta: избавимся от η
- $\Delta\theta_t = \frac{RMS[\Delta\theta]_{t-1}}{RMS[g]_t} g_t$

Adadelta: интуиция

- Метод Ньютона: $\Delta\theta_t = (\nabla^2 J)^{-1} \cdot \nabla J$
- Диагональная аппроксимация: $\nabla^2 J \approx \mathrm{diag}(\frac{\partial^2 J}{\partial \theta_{t,i}^2})$
- $\Delta\theta_{t,i} = (\frac{\partial^2 J}{\partial \theta_{t,i}^2})^{-1} \frac{\partial J}{\partial \theta_{t,i}}$
- $\frac{\partial^2 J}{\partial \theta_{t,i}^2} = \frac{\frac{\partial J}{\partial \theta_{t,i}}}{\Delta\theta_{t,i}}$

Adadelta: интуиция

- Метод Ньютона: $\Delta\theta_t = (\nabla^2 J)^{-1} \cdot \nabla J$
- Диагональная аппроксимация: $\nabla^2 J \approx \text{diag}(\frac{\partial^2 J}{\partial \theta_{t,i}^2})$
- $\Delta\theta_{t,i} = (\frac{\partial^2 J}{\partial \theta_{t,i}^2})^{-1} \frac{\partial J}{\partial \theta_{t,i}}$
- $\frac{\partial^2 J}{\partial \theta_{t,i}^2} = \frac{\frac{\partial J}{\partial \theta_{t,i}}}{\Delta\theta_{t,i}}$
- Не знаем числитель, но можем оценить при помощи RMS: $\frac{\partial^2 J}{\partial \theta_{t,i}^2} \approx \frac{RMS[g]_t}{RMS[\Delta\theta]_{t-1,i}}$

Adam (Adaptive Moment Estimation)

- $\begin{cases} m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \\ \nu_t = \beta_2 \nu_{t-1} + (1-\beta_2) g_t^2 \end{cases}$
- m_t, ν_t инициализируются нулями, поэтому долгий "разгон" $\Rightarrow$ надо уменьшить инерцию в начале обучения
- Надо обеспечить несмещенность: $\mathbb{E}[m_t] = \mathbb{E}[g_t]$ и $\mathbb{E}[v_t] = \mathbb{E}[g_t^2]$
- Поправка: $\begin{cases} \hat{m}_t = \frac{m_t}{1-\beta_1^t} \\ \hat{\nu}_t = \frac{\nu_t}{1-\beta_2^t} \end{cases}$
- $\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{\nu}_t} + \epsilon} \hat{m}_t$
- $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$

Критерии остановки

Когда остановить обучение?

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

Визуализация

- 2D визуализация (gif)
- Седловая точка (gif)

Вопросы