

Субградиентный спуск. AdaGrad, Adam, DoG. Регуляризация ℓ_1 , ℓ_2 . Проксимальный оператор

Александр Безносиков

ФПМИ МФТИ

7 октября 2025



Негладкие задачи

Определение выпуклости

Пусть дана непрерывно дифференцируемая на \mathbb{R}^d функция $f: \mathbb{R}^d \rightarrow \mathbb{R}$. Будем говорить, что она является выпуклой, если для любых $x, y \in \mathbb{R}^d$ выполнено $f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle$.

- **Вопрос:** функция $f(w) = |w|$ выпукла?

Негладкие задачи

Определение выпуклости

Пусть дана непрерывно дифференцируемая на \mathbb{R}^d функция $f: \mathbb{R}^d \rightarrow \mathbb{R}$. Будем говорить, что она является выпуклой, если для любых $x, y \in \mathbb{R}^d$ выполнено $f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle$.

- **Вопрос:** функция $f(w) = |w|$ выпукла? Безусловно. А дифференцируемая и гладкая?

Негладкие задачи

Определение выпуклости

Пусть дана непрерывно дифференцируемая на \mathbb{R}^d функция $f: \mathbb{R}^d \rightarrow \mathbb{R}$. Будем говорить, что она является выпуклой, если для любых $x, y \in \mathbb{R}^d$ выполнено $f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle$.

- **Вопрос:** функция $f(w) = |w|$ выпукла? Безусловно. А дифференцируемая и гладкая? Нет.
- Получается, что даже довольно простые выпуклые задачи могут быть негладким. До этого мы смотрели только на гладкие задачи.

Негладкие задачи

Будем рассматривать следующее предположение вместо гладкости (Липшицевости градиента):

Определение M -Липшецевой функции

Пусть дана функция $f : \mathbb{R}^d \rightarrow \mathbb{R}$. Будем говорить, что она является M -Липшицевой, если для любых $x, y \in \mathbb{R}^d$ выполнено

$$|f(x) - f(y)| \leq M \|x - y\|_2.$$

Негладкие задачи

Будем рассматривать следующее предположение вместо гладкости (Липшицевости градиента):

Определение M -Липшецевой функции

Пусть дана функция $f : \mathbb{R}^d \rightarrow \mathbb{R}$. Будем говорить, что она является M -Липшицевой, если для любых $x, y \in \mathbb{R}^d$ выполнено

$$|f(x) - f(y)| \leq M \|x - y\|_2.$$

Вопрос: почему не сильно выпуклые?

Субградиент и субдифференциал

Если функция не дифференцируема в точке, а значит градиента нет.
Что может существовать вместо градиента?

Субградиент и субдифференциал

Если функция не дифференцируема в точке, а значит градиента нет. Что может существовать вместо градиента?

Субградиент и субдифференциал

Пусть дана выпуклая функция $f : \mathbb{R}^d \rightarrow \mathbb{R}$. Вектор g будем называть субградиентом этой функции f в точке $x \in \mathbb{R}^d$, если для любого $y \in \mathbb{R}^d$ выполняется:

$$f(y) \geq f(x) + \langle g, y - x \rangle.$$

Множество $\partial f(x)$ всех субградиентов f в x будем называть субдифференциалом.

Условие оптимальности

Теорема (условие оптимальности)

w^* – минимум выпуклой функции f тогда и только тогда, когда

$$0 \in \partial f(w^*).$$

Свойство M -Липшицевой функции

Лемма (свойство M -Липшицевой функции)

Пусть дана выпуклая функция $f : \mathbb{R}^d \rightarrow \mathbb{R}$. Тогда функция f является M -Липшицевой тогда и только тогда, когда для любого $x \in \mathbb{R}^d$ и $g \in \partial f(x)$ имеем $\|g\|_2 \leq M$.

Субградиентный метод

- Рассматриваем задачу:

$$\min_{w \in \mathbb{R}^d} f(w),$$

где f выпуклая и M -Липшицева.

Субградиентный метод

- Рассматриваем задачу:

$$\min_{w \in \mathbb{R}^d} f(w),$$

где f выпуклая и M -Липшицева.

- Простая идея – вместо градиента использовать какой-то субградиент в текущей точке:

Алгоритм Субградиентный метод

Вход: размеры шага $\gamma > 0$, стартовая точка $w^0 \in \mathbb{R}^d$, количество итераций K

- 1: **for** $k = 0, 1, \dots, K - 1$ **do**
- 2: Вычислить $g^k \in \partial f(w^k)$
- 3: $w^{k+1} = w^k - \gamma g^k$
- 4: **end for**

Выход: $\frac{1}{K} \sum_{k=0}^{K-1} w^k$

Доказательство сходимости

- Ничего сверхъестественного:

$$\begin{aligned}\|w^{k+1} - w^*\|_2^2 &= \|w^k - \gamma g^k - w^*\|_2^2 \\ &= \|w^k - w^*\|_2^2 - 2\gamma \langle g^k, w^k - w^* \rangle + \gamma^2 \|g^k\|_2^2\end{aligned}$$

Доказательство сходимости

- Ничего сверхъестественного:

$$\begin{aligned}\|w^{k+1} - w^*\|_2^2 &= \|w^k - \gamma g^k - w^*\|_2^2 \\ &= \|w^k - w^*\|_2^2 - 2\gamma \langle g^k, w^k - w^* \rangle + \gamma^2 \|g^k\|_2^2\end{aligned}$$

- Из M -Липшицевости f следует, что субградиенты ограничены:

$$\|w^{k+1} - w^*\|_2^2 \leq \|w^k - w^*\|_2^2 - 2\gamma \langle g^k, w^k - w^* \rangle + \gamma^2 M^2$$

Доказательство сходимости

- Ничего сверхъестественного:

$$\begin{aligned}\|w^{k+1} - w^*\|_2^2 &= \|w^k - \gamma g^k - w^*\|_2^2 \\ &= \|w^k - w^*\|_2^2 - 2\gamma \langle g^k, w^k - w^* \rangle + \gamma^2 \|g^k\|_2^2\end{aligned}$$

- Из M -Липшицевости f следует, что субградиенты ограничены:

$$\|w^{k+1} - w^*\|_2^2 \leq \|w^k - w^*\|_2^2 - 2\gamma \langle g^k, w^k - w^* \rangle + \gamma^2 M^2$$

- Из выпуклости и определения субградиента:

$$\|w^{k+1} - w^*\|_2^2 \leq \|w^k - w^*\|_2^2 - 2\gamma (f(w^k) - f(w^*)) + \gamma^2 M^2$$

Доказательство сходимости

- С предыдущего слайда:

$$2\gamma(f(w^k) - f(w^*)) \leq \|w^k - w^*\|_2^2 - \|w^{k+1} - w^*\|_2^2 + \gamma^2 M^2$$

Доказательство сходимости

- С предыдущего слайда:

$$2\gamma(f(w^k) - f(w^*)) \leq \|w^k - w^*\|_2^2 - \|w^{k+1} - w^*\|_2^2 + \gamma^2 M^2$$

- Суммируем по всем k и усредняем:

$$\frac{2\gamma}{K} \sum_{k=0}^{K-1} (f(w^k) - f(w^*)) \leq \frac{\|w^0 - w^*\|_2^2}{K} + \gamma^2 M^2$$

Доказательство сходимости

- С предыдущего слайда:

$$2\gamma(f(w^k) - f(w^*)) \leq \|w^k - w^*\|_2^2 - \|w^{k+1} - w^*\|_2^2 + \gamma^2 M^2$$

- Суммируем по всем k и усредняем:

$$\frac{2\gamma}{K} \sum_{k=0}^{K-1} (f(w^k) - f(w^*)) \leq \frac{\|w^0 - w^*\|_2^2}{K} + \gamma^2 M^2$$

- Откуда

$$\frac{1}{K} \sum_{k=0}^{K-1} (f(w^k) - f(w^*)) \leq \frac{\|w^0 - w^*\|_2^2}{2\gamma K} + \frac{\gamma M^2}{2}$$

Доказательство сходимости

- С предыдущего слайда:

$$\frac{1}{K} \sum_{k=0}^{K-1} (f(w^k) - f(w^*)) \leq \frac{\|w^0 - w^*\|_2^2}{2\gamma K} + \frac{\gamma M^2}{2}$$

- Гладкости нет, поэтому не получится доказать, что $f(w^k) \leq f(w^{k-1})$. Поэтому просто неравенство Йенсена для выпуклой функции:

$$f\left(\frac{1}{K} \sum_{k=0}^{K-1} w^k\right) - f(w^*) \leq \frac{\|w^0 - w^*\|_2^2}{2\gamma K} + \frac{\gamma M^2}{2}$$

Доказательство сходимости

- С предыдущего слайда:

$$f\left(\frac{1}{K} \sum_{k=0}^{K-1} w^k\right) - f(w^*) \leq \frac{\|w^0 - w^*\|_2^2}{2\gamma K} + \frac{\gamma M^2}{2}$$

- Вопрос:** как подобрать шаг?

Доказательство сходимости

- С предыдущего слайда:

$$f\left(\frac{1}{K}\sum_{k=0}^{K-1}w^k\right) - f(w^*) \leq \frac{\|w^0 - w^*\|_2^2}{2\gamma K} + \frac{\gamma M^2}{2}$$

- Вопрос:** как подобрать шаг? минимизировать правую часть по γ :
 $\gamma = \frac{\|w^0 - w^*\|_2^2}{M\sqrt{K}}$. Откуда

$$f\left(\frac{1}{K}\sum_{k=0}^{K-1}w^k\right) - f(w^*) \leq \frac{M\|w^0 - w^*\|_2}{\sqrt{K}}$$

Сходимость

Теорема сходимости субградиентного спуска для M -Липшицевых и выпуклых функций

Пусть задача безусловной оптимизации с M -Липшицевой, выпуклой целевой функцией f решается с помощью субградиентного спуска.

Тогда справедлива следующая оценка сходимости

$$f\left(\frac{1}{K} \sum_{k=0}^{K-1} w^k\right) - f(w^*) \leq \frac{M \|w^0 - w^*\|_2}{\sqrt{K}}$$

Более того, чтобы добиться точности ε по функции, необходимо

$$K = O\left(\frac{M^2 \|w^0 - w^*\|_2^2}{\varepsilon^2}\right) \text{ итераций.}$$

Субградиентный метод: итог

- Обобщение градиентного спуска на негладкие задачи.

Субградиентный метод: итог

- Обобщение градиентного спуска на негладкие задачи.
- Оценки сходимости в выпуклом случае: $\sim \frac{1}{\sqrt{K}}$, в сильно выпуклом случае: $\sim \frac{1}{K}$. **Вопрос:** какие были у градиентного спуска в гладком случае?

Субградиентный метод: итог

- Обобщение градиентного спуска на негладкие задачи.
- Оценки сходимости в выпуклом случае: $\sim \frac{1}{\sqrt{K}}$, в сильно выпуклом случае: $\sim \frac{1}{K}$. **Вопрос:** какие были у градиентного спуска в гладком случае? $\sim \frac{1}{K}$ и линейная соответственно. Сходимость медленнее.

Субградиентный метод: итог

- Обобщение градиентного спуска на негладкие задачи.
- Оценки сходимости в выпуклом случае: $\sim \frac{1}{\sqrt{K}}$, в сильно выпуклом случае: $\sim \frac{1}{K}$. **Вопрос:** какие были у градиентного спуска в гладком случае? $\sim \frac{1}{K}$ и линейная соответственно. Сходимость медленнее.
- Может возможно улучшить результат? Например, улучшить анализ или использовать моментум.

Субградиентный метод: итог

- Обобщение градиентного спуска на негладкие задачи.
- Оценки сходимости в выпуклом случае: $\sim \frac{1}{\sqrt{K}}$, в сильно выпуклом случае: $\sim \frac{1}{K}$. **Вопрос:** какие были у градиентного спуска в гладком случае? $\sim \frac{1}{K}$ и линейная соответственно. Сходимость медленнее.
- Может возможно улучшить результат? Например, улучшить анализ или использовать моментум. В общем случае результат для субградиентного метода является неулучшаемым для выпуклых и сильно-выпуклых задач, т.е. он оптимален.

AdaGradNorm

- Для субградиентного метода был взят шаг $\gamma = \frac{\|w^0 - w^*\|_2}{M\sqrt{K}}$.
- **Вопрос:** Какие есть проблемы в таком подходе?

AdaGradNorm

- Для субградиентного метода был взят шаг $\gamma = \frac{\|w^0 - w^*\|_2}{M\sqrt{K}}$.
- **Вопрос:** Какие есть проблемы в таком подходе?
Зависимость от $K, M, \|w^0 - w^*\|_2$.

AdaGradNorm

- Для субградиентного метода был взят шаг $\gamma = \frac{\|w^0 - w^*\|_2}{M\sqrt{K}}$.
- **Вопрос:** Какие есть проблемы в таком подходе?
Зависимость от $K, M, \|w^0 - w^*\|_2$.
- Можно взять $\gamma_k = \frac{\|w^0 - w^*\|_2}{M\sqrt{k}}$.

AdaGradNorm

- Для субградиентного метода был взят шаг $\gamma = \frac{\|w^0 - w^*\|_2}{M\sqrt{K}}$.
- **Вопрос:** Какие есть проблемы в таком подходе?
Зависимость от $K, M, \|w^0 - w^*\|_2$.
- Можно взять $\gamma_k = \frac{\|w^0 - w^*\|_2}{M\sqrt{k}}$.
- M – ограничение нормы (суб)градиента, тогда можно использовать сам (суб)градиент в качестве этого ограничения, кроме этого $\|w^0 - w^*\|_2 \leq D$:

$$\gamma_k = \frac{D}{\sqrt{\sum_{t=0}^k \|g^t\|_2^2}}.$$

AdaGradNorm

- Получился метод AdaGradNorm. Ada – адаптивность под локальные свойства задачи (в данном случае локальные значения M).

Алгоритм AdaGradNorm

Вход: $D > 0$, стартовая точка $w^0 \in \mathbb{R}^d$, сумма квадратов норм градиентов $G^0 = 0$, количество итераций K

- for** $k = 0, 1, \dots, K - 1$ **do**
- Вычислить $g^k \in \partial f(w^k)$
- Вычислить $G^{k+1} = G^k + \|g^k\|_2^2$
- $w^{k+1} = w^k - \frac{D}{\sqrt{G^{k+1}}} g^k$
- end for**

Выход: $\frac{1}{K} \sum_{k=0}^K w^k$

AdaGrad

- Пойдем дальше и сделаем адаптивность по каждой координате (индивидуальный шаг). Получится AdaGrad:

$$\gamma_{k,i} = \frac{D}{\sqrt{\sum_{t=0}^k (g_i^t)^2}}, \quad \text{где} \quad \sup_i \|w_i - w_i^*\|_2 \leq D.$$

AdaGrad

Алгоритм AdaGrad

Вход: $D > 0$, стартовая точка $w^0 \in \mathbb{R}^d$, сумма квадратов градиентов $G_i^0 = 0$, количество итераций K

- 1: **for** $k = 0, 1, \dots, K - 1$ **do**
- 2: Вычислить $g^k \in \partial f(w^k)$
- 3: Для каждой координаты: $G_i^{k+1} = G_i^k + (g_i^k)^2$
- 4: Для каждой координаты: $w_i^{k+1} = w_i^k - \frac{D}{\sqrt{G_i^{k+1}}} g_i^k$
- 5: **end for**

Выход: $\frac{1}{K} \sum_{k=0}^K w^k$

Сходимость AdaGrad

Теорема сходимость AdaGrad для M -Липшицевых и выпуклых функций

Пусть задача оптимизации с M -Липшицевой, выпуклой целевой функцией f решается с помощью AdaGrad на ограниченном множестве. Тогда справедлива следующая оценка сходимости:

$$f\left(\frac{1}{K} \sum_{k=0}^{K-1} w^k\right) - f(w^*) \leq \frac{3M\tilde{D}}{2\sqrt{K}},$$

где $\tilde{D} = \sum_{i=1}^d D_i$.

Более того, чтобы добиться точности ε по функции, необходимо

$$K = O\left(\frac{9M^2\tilde{D}^2}{4\varepsilon^2}\right) \text{ итераций.}$$

RMSPprop

- Проблема AdaGrad, что старые градиенты в шаге могут быть уже не особо релевантны. **Вопрос:** как можно попробовать их «забывать»?

RMSPprop

- Проблема AdaGrad, что старые градиенты в шаге могут быть уже не особо релевантны. **Вопрос:** как можно попробовать их «забывать»?
- Может помочь техника момента с $\beta_2 \in (0, 1)$ (вспомните, как она работала в случае тяжелого шарика):

$$G_i^{k+1} = \beta_2 G_i^k + (1 - \beta_2)(g_i^k)^2,$$

$$\gamma_{k,i} = \frac{D}{\sqrt{G_i^{k+1}}} g_i^k.$$

Получился метод RMSPprop.

RMSProp

Алгоритм RMSProp

Вход: $D > 0$, моментум $\beta_2 \in (0, 1)$, стартовая точка $w^0 \in \mathbb{R}^d$, сглаженная сумма квадратов градиентов $G_i^0 = 0$, количество итераций K

1: **for** $k = 0, 1, \dots, K - 1$ **do**

2: Вычислить $g^k \in \partial f(w^k)$

3: Для каждой координаты: $G_i^{k+1} = \beta_2 G_i^k + (1 - \beta_2)(g_i^k)^2$

4: Для каждой координаты: $w_i^{k+1} = w_i^k - \frac{D}{\sqrt{G_i^{k+1}}} g_i^k$

5: **end for**

Выход: $\frac{1}{K} \sum_{k=0}^K w^k$

Adam

- Почему бы не добавить и классический моментум вида тяжелого шарика для ускорения? Получится метод Adam:

тяжелый шарик: $v^{k+1} = \beta_1 v^k + (1 - \beta_1) g^k,$

RMSProp: $G_i^{k+1} = \beta_2 G_i^k + (1 - \beta_2) (g_i^k)^2,$

$$w_i^{k+1} = w_i^k - \frac{D}{\sqrt{G_i^{k+1}}} v_i^{k+1}.$$

Adam

- Почему бы не добавить и классический моментум вида тяжелого шарика для ускорения? Получится метод Adam:

тяжелый шарик: $v^{k+1} = \beta_1 v^k + (1 - \beta_1) g^k,$

RMSProp: $G_i^{k+1} = \beta_2 G_i^k + (1 - \beta_2)(g_i^k)^2,$

$$w_i^{k+1} = w_i^k - \frac{D}{\sqrt{G_i^{k+1}}} v_i^{k+1}.$$

- Можно еще доработать – например, менять коэффициенты β_1, β_2 в зависимости от шага: β_1^k, β_2^k

Adam

- Почему бы не добавить и классический моментум вида тяжелого шарика для ускорения? Получится метод Adam:

тяжелый шарик: $v^{k+1} = \beta_1 v^k + (1 - \beta_1) g^k,$

RMSProp: $G_i^{k+1} = \beta_2 G_i^k + (1 - \beta_2)(g_i^k)^2,$

$$w_i^{k+1} = w_i^k - \frac{D}{\sqrt{G_i^{k+1}}} v_i^{k+1}.$$

- Можно еще доработать – например, менять коэффициенты β_1, β_2 в зависимости от шага: β_1^k, β_2^k
- Ещё можно обезопасить себя от деления на 0 с помощью небольшой добавки $\epsilon \sim 10^{-6} - 10^{-8}$:

$$w_i^{k+1} = w_i^k - \frac{D}{\sqrt{G_i^{k+1} + \epsilon}} v_i^{k+1}$$

Adam

Алгоритм Adam

Вход: $D > 0$, моменты $\{\beta_1^k\} \in (0, 1)$ и $\{\beta_2^k\} \in (0, 1)$, стартовая точка $w^0 \in \mathbb{R}^d$, сглаженная сумма квадратов градиентов $G_i^0 = 0$, сглаженная сумма градиентов $v^0 = 0$, добавка $\epsilon > 0$, количество итераций K

- 1: **for** $k = 0, 1, \dots, K - 1$ **do**
- 2: Вычислить $g^k \in \partial f(w^k)$
- 3: Вычислить $v^{k+1} = \beta_1^k v^k + (1 - \beta_1^k) g^k$
- 4: Для каждой координаты: $G_i^{k+1} = \beta_2^k G_i^k + (1 - \beta_2^k)(g_i^k)^2$
- 5: Для каждой координаты: $w_i^{k+1} = w_i^k - \frac{D}{\sqrt{G_i^{k+1} + \epsilon}} v_i^{k+1}$
- 6: **end for**

Выход: $\frac{1}{K} \sum_{k=0}^K w^k$

DoG

- Ситуация в общем случае даже ухудшилась по сравнению с AdaGrad: кроме шагов D_i , нужно подбирать еще моментумы β_1, β_2 .

DoG

- Ситуация в общем случае даже ухудшилась по сравнению с AdaGrad: кроме шагов D_i , нужно подбирать еще моментумы β_1, β_2 .
- Часто рекомендуют брать $\beta_1 = 0.9$, а $\beta_2 = 0.99$, и $D_i = D$ для всех i , но все равно нужно подбирать D .

DoG

- Ситуация в общем случае даже ухудшилась по сравнению с AdaGrad: кроме шагов D_i , нужно подбирать еще моментумы β_1, β_2 .
- Часто рекомендуют брать $\beta_1 = 0.9$, а $\beta_2 = 0.99$, и $D_i = D$ для всех i , но все равно нужно подбирать D .
- Проблема поиска D или D_i исконно пришла от того, что мы не знали $\|w^0 - w^*\|_2$. **Вопрос:** мы знаем w^0 , какая точка (в некотором смысле) является хорошим выбором в качестве приближения w^* ?

DoG

- Ситуация в общем случае даже ухудшилась по сравнению с AdaGrad: кроме шагов D_i , нужно подбирать еще моментумы β_1, β_2 .
- Часто рекомендуют брать $\beta_1 = 0.9$, а $\beta_2 = 0.99$, и $D_i = D$ для всех i , но все равно нужно подбирать D .
- Проблема поиска D или D_i исконно пришла от того, что мы не знали $\|w^0 - w^*\|_2$. **Вопрос:** мы знаем w^0 , какая точка (в некотором смысле) является хорошим выбором в качестве приближения w^* ?

DoG

- Можно предположить, что $w^k \approx w^*$ (метод приближается к решению):

$$\|w^0 - w^*\|_2 \rightarrow \|w^0 - w^k\|_2 \rightarrow d_k = \max_{t \in [k-1]} \|w^0 - w^t\|_2.$$

или

$$|w_i^0 - w_i^*| \rightarrow |w_i^0 - w_i^k| \rightarrow d_{k,i} = \max_{t \in [k-1]} |w_i^0 - w_i^t|.$$

DoG

- Можно предположить, что $w^k \approx w^*$ (метод приближается к решению):

$$\|w^0 - w^*\|_2 \rightarrow \|w^0 - w^k\|_2 \rightarrow d_k = \max_{t \in [k-1]} \|w^0 - w^t\|_2.$$

или

$$|w_i^0 - w_i^*| \rightarrow |w_i^0 - w_i^k| \rightarrow d_{k,i} = \max_{t \in [k-1]} |w_i^0 - w_i^t|.$$

- Имеем метод DoG:

$$\gamma_k = \frac{d_k}{\sqrt{\sum_{t=0}^k \|g^t\|_2^2}} \quad \text{или} \quad \gamma_{k,i} = \frac{d_{k,i}}{\sqrt{\sum_{t=0}^k (g_i^t)^2}}.$$

DoG

Метод DoG:

$$\gamma_k = \frac{d_k}{\sqrt{\sum_{t=0}^k \|g^t\|_2^2}} \quad \text{или} \quad \gamma_{k,i} = \frac{d_{k,i}}{\sqrt{\sum_{t=0}^k (g_i^t)^2}}.$$

Вопрос: Почему DoG?

DoG

Метод DoG:

$$\gamma_k = \frac{d_k}{\sqrt{\sum_{t=0}^k \|g^t\|_2^2}} \quad \text{или} \quad \gamma_{k,i} = \frac{d_{k,i}}{\sqrt{\sum_{t=0}^k (g_i^t)^2}}.$$

Вопрос: Почему DoG? Distance over Gradients:

$$d_k = \max_{t \in [k-1]} \left\| \sum_{\tau=0}^t \gamma_\tau \nabla f(w^\tau) \right\|_2.$$

DoG

Алгоритм DoG v1

Вход: стартовая точка $w^0 \in \mathbb{R}^d$, сумма квадратов норм градиентов $G^0 = 0$, расстояние от точки старта $d_0 = 0$, количество итераций K

- 1: **for** $k = 0, 1, \dots, K - 1$ **do**
- 2: Вычислить $g^k \in \partial f(w^k)$
- 3: Вычислить $G^{k+1} = G^k + \|g^k\|_2^2$
- 4: Вычислить $d_{k+1} = \max\{d_k; \|w^0 - w^k\|\}$
- 5: $w^{k+1} = w^k - \frac{d_{k+1}}{\sqrt{G^{k+1}}} g^k$
- 6: **end for**

Выход: $\frac{1}{K} \sum_{k=0}^K w^k$

DoG

Алгоритм DoG v2

Вход: стартовая точка $w^0 \in \mathbb{R}^d$, сумма квадратов градиентов $G_i^0 = 0$,
координатные расстояния от точки старта $d_{0,i}$, количество итераций
 K

- 1: **for** $k = 0, 1, \dots, K - 1$ **do**
- 2: Вычислить $g^k \in \partial f(w^k)$
- 3: Для каждой координаты: $G_i^{k+1} = G_i^k + (g_i^k)^2$
- 4: Вычислить $d_{k+1,i} = \max\{d_{k,i}; |w_i^0 - w_i^k|\}$
- 5: Для каждой координаты: $w_i^{k+1} = w_i^k - \frac{d_{k,i}}{\sqrt{G_i^{k+1}}} g_i^k$

6: **end for**

Выход: $\frac{1}{K} \sum_{k=0}^K w^k$

Адаптивные методы: итог

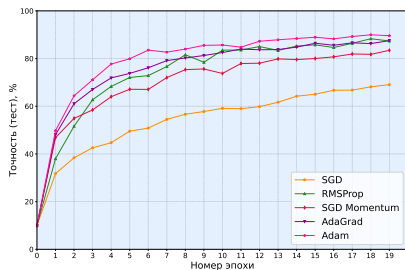
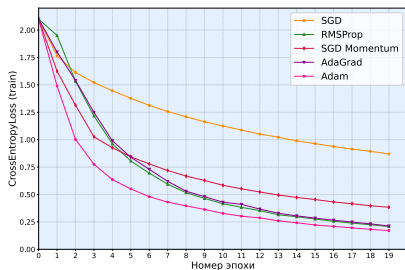
- Суть – подбирать шаг, исходя из локальных свойств задачи, что кажется более эффективным и удобным подходом.
- Adam и его модификации являются самым популярным методами решения задач оптимизации, лежащих в основе обучения нейронных сетей. Но в нем подбор параметров все еще нужен.

Адаптивные методы: итог

- Суть – подбирать шаг, исходя из локальных свойств задачи, что кажется более эффективным и удобным подходом.
- Adam и его модификации являются самым популярным методами решения задач оптимизации, лежащих в основе обучения нейронных сетей. Но в нем подбор параметров все еще нужен.
- Но эта проблема сейчас решена в так называемых parameter-free методах: кроме DoG можно обратить внимание на D-Adaptation/Prodigy.

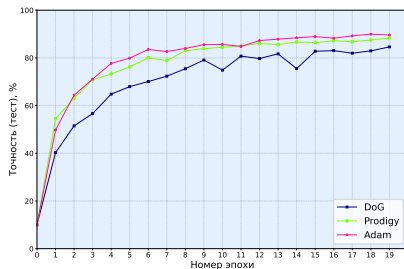
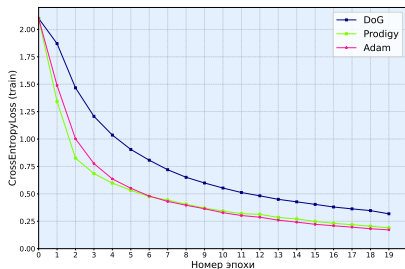
Адаптивные методы: сравнение на практике

- Для наглядности мы поставили эксперименты для задачи классификации на датасете CIFAR10 на модели RESNET18 (популярная модель глубокого обучения).



Parameter-free методы: сравнение на практике

- Аналогично сравним результаты работы DoG, рассмотренного ранее с одним из топовых беспараметрических методов – Prodigy.



Проксимальный оператор

- Поняли, что негладкие задачи «более сложные» по сравнению с гладкими задачами.
- Может быть получится «спрятать под ковер» отсутствие гладкости?

Проксимальный оператор

- Поняли, что негладкие задачи «более сложные» по сравнению с гладкими задачами.
- Может быть получится «спрятать под ковер» отсутствие гладкости?
- Такую возможность дает проксимальный оператор:

Определение проксимального оператора

Для функции $r : \mathbb{R}^d \rightarrow \mathbb{R} \cup \{+\infty\}$ проксимальный оператор определяется следующим образом:

$$\text{prox}_r(w) = \arg \min_{\tilde{w} \in \mathbb{R}^d} \left(r(\tilde{w}) + \frac{1}{2} \|w - \tilde{w}\|^2 \right).$$

Свойства проксимального оператора

Лемма (свойство проксимального оператора)

Пусть $r : \mathbb{R}^d \rightarrow \mathbb{R} \cup \{+\infty\}$ выпуклая функция, для которой определен prox_r . Если существует такая $\hat{w} \in \mathbb{R}^d$, что $r(w) < +\infty$. Тогда проксимальный оператор однозначно определен (т.е. всегда возвращает единственное уникальное значение).

Свойства проксимального оператора

Лемма (свойство проксимального оператора)

Пусть $r : \mathbb{R}^d \rightarrow \mathbb{R} \cup \{+\infty\}$ выпуклая функция, для которой определен prox_r . Если существует такая $\hat{w} \in \mathbb{R}^d$, что $r(w) < +\infty$. Тогда проксимальный оператор однозначно определен (т.е. всегда возвращает единственное уникальное значение).

Доказательство: Проксимальный оператор возвращает минимум некоторой задачи оптимизации. Вопрос: что можно сказать про эту задачу?

Свойства проксимального оператора

Лемма (свойство проксимального оператора)

Пусть $r : \mathbb{R}^d \rightarrow \mathbb{R} \cup \{+\infty\}$ выпуклая функция, для которой определен prox_r . Если существует такая $\hat{w} \in \mathbb{R}^d$, что $r(w) < +\infty$. Тогда проксимальный оператор однозначно определен (т.е. всегда возвращает единственное уникальное значение).

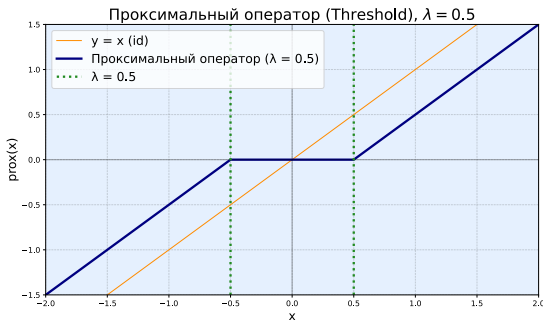
Доказательство: Проксимальный оператор возвращает минимум некоторой задачи оптимизации. Вопрос: что можно сказать про эту задачу? Она сильно выпуклая, а значит имеет строго один уникальный минимум (существование \hat{w} необходимо для того, чтобы $r(\tilde{w}) + \frac{1}{2}\|w - \tilde{w}\|^2$ где-то принимала конечное значение).

Примеры проксимального оператора

- $r(w) = \lambda \|w\|_1$, где $\lambda > 0$. Тогда

$$[\text{prox}_r(w)]_i = [|w_i| - \lambda]_+ \cdot \text{sign}(w_i)$$

Такой проксимальный оператор еще называют трешхолдом.



Примеры проксимального оператора

- $r(w) = \frac{\lambda}{2} \|w\|_2^2$, где $\lambda > 0$. Тогда

$$\text{prox}_r(w) = \frac{w}{1 + \lambda}.$$

Примеры проксимального оператора

- $r(w) = \frac{\lambda}{2} \|w\|_2^2$, где $\lambda > 0$. Тогда

$$\text{prox}_r(w) = \frac{w}{1 + \lambda}.$$

- $r(w) = \mathbb{I}_{\mathcal{W}}(w)$, где \mathcal{W} – выпуклое множество, и

$$\mathbb{I}_{\mathcal{W}}(w) = \begin{cases} 0, & w \in \mathcal{W} \\ +\infty, & w \notin \mathcal{W} \end{cases}.$$

Вопрос: чему равен prox ?

Примеры проксимального оператора

- $r(w) = \frac{\lambda}{2} \|w\|_2^2$, где $\lambda > 0$. Тогда

$$\text{prox}_r(w) = \frac{w}{1 + \lambda}.$$

- $r(w) = \mathbb{I}_{\mathcal{W}}(w)$, где \mathcal{W} – выпуклое множество, и

$$\mathbb{I}_{\mathcal{W}}(w) = \begin{cases} 0, & w \in \mathcal{W} \\ +\infty, & w \notin \mathcal{W} \end{cases}.$$

Вопрос: чему равен prox ?

$$\text{prox}_r(w) = \text{proj}_{\mathcal{W}}(w).$$

Примеры проксимального оператора

- $r(w) = \frac{\lambda}{2} \|w\|_2^2$, где $\lambda > 0$. Тогда

$$\text{prox}_r(w) = \frac{w}{1 + \lambda}.$$

- $r(w) = \mathbb{I}_{\mathcal{W}}(w)$, где \mathcal{W} – выпуклое множество, и

$$\mathbb{I}_{\mathcal{W}}(w) = \begin{cases} 0, & w \in \mathcal{W} \\ +\infty, & w \notin \mathcal{W} \end{cases}.$$

Вопрос: чему равен prox ?

$$\text{prox}_r(w) = \text{proj}_{\mathcal{W}}(w).$$

- И еще множество других примеров и их комбинаций.

Композитная задача

- Рассмотрим следующую задачу:

$$\min_{w \in \mathbb{R}^d} [f(w) + r(w)].$$

Композитная задача

- Рассмотрим следующую задачу:

$$\min_{w \in \mathbb{R}^d} [f(w) + r(w)].$$

- Такая задача называется композитной.
- Предположим, что f является L -гладкой выпуклой функцией, r выпуклой (необязательно гладкой, но) проксимально дружественной функцией.

Композитная задача

- Рассмотрим следующую задачу:

$$\min_{w \in \mathbb{R}^d} [f(w) + r(w)].$$

- Такая задача называется композитной.
- Предположим, что f является L -гладкой выпуклой функцией, r выпуклой (необязательно гладкой, но) проксимально дружественной функцией.
- Получается целевая функция состоит из гладкой и в общем случае негладкой части. Если $r \equiv 0$, то получаем гладкую задачу, которую умеем решать. Если $f \equiv 0$, то получаем негладкую задачу.

Проксимальный градиентный метод

Алгоритм Проксимальный градиентный метод

Вход: размер шага $\gamma > 0$, стартовая точка $w^0 \in \mathbb{R}^d$, количество итераций K

- 1: **for** $k = 0, 1, \dots, K - 1$ **do**
- 2: Вычислить $\nabla f(w^k)$
- 3: $w^{k+1} = \text{prox}_{\gamma r}(w^k - \gamma \nabla f(w^k))$
- 4: **end for**

Выход: w^K

Проксимальный градиентный метод

Алгоритм Проксимальный градиентный метод

Вход: размер шага $\gamma > 0$, стартовая точка $w^0 \in \mathbb{R}^d$, количество итераций K

- 1: **for** $k = 0, 1, \dots, K - 1$ **do**
- 2: Вычислить $\nabla f(w^k)$
- 3: $w^{k+1} = \text{prox}_{\gamma r}(w^k - \gamma \nabla f(w^k))$
- 4: **end for**

Выход: w^K

- Если r непрерывно дифференцируема, то условие оптимальности для подзадачи подсчета проксимального оператора записывается, как:

$$0 = \gamma \nabla r(w^{k+1}) + w^{k+1} - \gamma \nabla f(w^k).$$

Проксимальный градиентный метод

Алгоритм Проксимальный градиентный метод

Вход: размер шага $\gamma > 0$, стартовая точка $w^0 \in \mathbb{R}^d$, количество итераций K

- 1: **for** $k = 0, 1, \dots, K - 1$ **do**
- 2: Вычислить $\nabla f(w^k)$
- 3: $w^{k+1} = \text{prox}_{\gamma r}(w^k - \gamma \nabla f(w^k))$
- 4: **end for**

Выход: w^K

- Если r непрерывно дифференцируема, то условие оптимальности для подзадачи подсчета проксимального оператора записывается, как:

$$0 = \gamma \nabla r(w^{k+1}) + w^{k+1} - \gamma \nabla f(w^k).$$

- Откуда получаем так называемую неявную запись метода:

$$w^{k+1} = w^k - \gamma(\nabla f(w^k) + \nabla r(w^{k+1}))$$

Сходимость проксимального метода

Теорема

Пусть задача композитной оптимизации с L -гладкой, μ -сильно выпуклой целевой функцией f и выпуклой (необязательно гладкой, но) проксимально дружественной функцией r решается с помощью проксимального градиентного спуска. Тогда при $y_k = \frac{1}{L}$ справедлива следующая оценка сходимости:

$$\|w^k - w^*\|_2^2 \leq \left(1 - \frac{\mu}{L}\right)^K \|w^0 - w^*\|_2^2.$$

т.е. для достижения точности ε по норме аргумента ($\|w^K - w^*\| \leq \varepsilon$) необходимо

$$K = \left(\frac{L}{\mu} \log \frac{\|w^0 - w^*\|_2}{\varepsilon} \right) \text{ итераций.}$$