Моментум и ускорение. Оптимальный метод

Методы оптимизации

Александр Безносиков

Московский физико-технический институт

26 сентября 2024

ФПМИ

Вопрос с прошлой лекции

- Была получена верхняя оценка на сходимость градиентного спуска для L-гладких и μ-сильно выпуклых задач. **Вопрос:** сколько итераций/оракульных вызовов нужно сделать, чтобы найти ε-решение?

Вопрос с прошлой лекции

- Была получена верхняя оценка на сходимость градиентного спуска для L-гладких и μ-сильно выпуклых задач. **Вопрос:** сколько итераций/оракульных вызовов нужно сделать, чтобы найти ε-решение?

$$O\left(\frac{L}{\mu}\log\frac{\|x^0 - x^*\|_2}{\varepsilon}\right) \text{ итераций/оракульных вызовов.}$$

Вопрос с прошлой лекции

- Была получена верхняя оценка на сходимость градиентного спуска для L-гладких и μ-сильно выпуклых задач. **Вопрос:** сколько итераций/оракульных вызовов нужно сделать, чтобы найти ε-решение?

$$O\left(\frac{L}{\mu}\log\frac{\|x^0 - x^*\|_2}{\varepsilon}\right) \text{ итераций/оракульных вызовов.}$$

- Вопрос на которым ответим сегодня: а можно ли лучше?

Метод тяжелого шарика

- Б.Т. Поляк в 1964 году предложил метод тяжелого шарика.

Алгоритм 1 Метод тяжелого шарика

Вход: размер шагов $\{\gamma_k\}_{k=0} > 0$, моментумы $\{\tau_k\}_{k=0} \in [0; 1]$, стартовая точка $x^0 = x^{-1} \in \mathbb{R}^d$, количество итераций K

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

2: Вычислить $\nabla f(x^k)$

3: $x^{k+1} = x^k - \gamma_k \nabla f(x^k) + \tau_k (x^k - x^{k-1})$

4: **end for**

Выход: x^K

Метод тяжелого шарика

- Б.Т. Поляк в 1964 году предложил метод тяжелого шарика.

Алгоритм 2 Метод тяжелого шарика

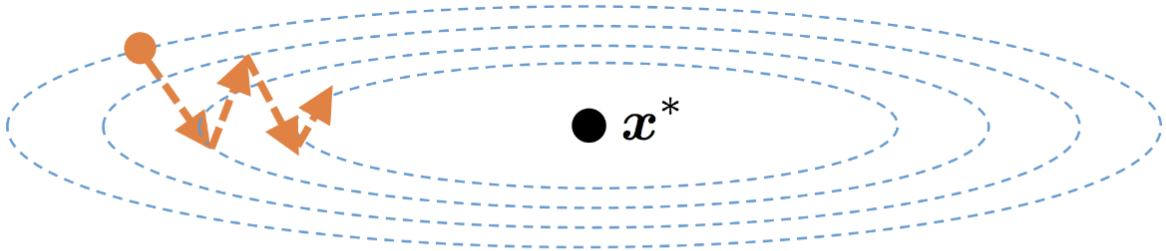
Вход: размер шагов $\{\gamma_k\}_{k=0} > 0$, моментумы $\{\tau_k\}_{k=0} \in [0; 1]$, стартовая точка $x^0 = x^{-1} \in \mathbb{R}^d$, количество итераций K

1: **for** $k = 0, 1, \ldots, K-1$ **do**
2: Вычислить $\nabla f(x^k)$
3: $x^{k+1} = x^k - \gamma_k \nabla f(x^k) + \tau_k (x^k - x^{k-1})$
4: **end for**

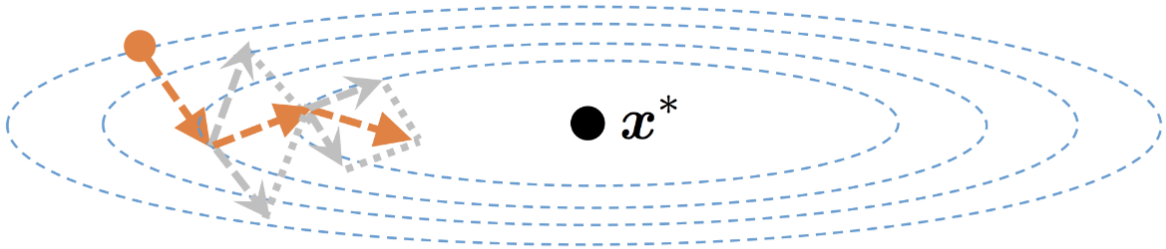
Выход: x^K

- Добавим к градиентному спуску моментумный член — предположим, что у точки, отвечающей за текущее положение значение x^k есть инерция.

Сравнение тяжелого шарика и градиентного спуска

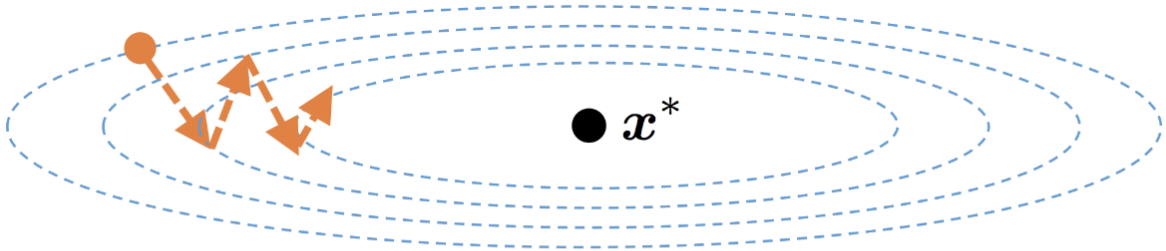


gradient descent

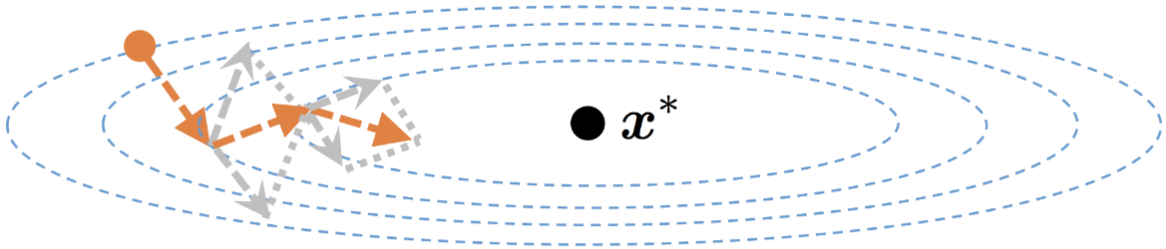


heavy-ball method

Сравнение тяжелого шарика и градиентного спуска



gradient descent



heavy-ball method

Интерактивная иллюстрация доступна по ссылке.

На чем держится ML

- В библиотеке pytorch (основная библиотека Deep Learning) реализован следующий метод:

$$v^{k+1} = \beta v^k + \nabla f(x^k) \quad \beta \in [0; 1)$$

$$x^{k+1} = x^k - \gamma v^{k+1}$$

На чем держится ML

- В библиотеке pytorch (основная библиотека Deep Learning) реализован следующий метод:

$$v^{k+1} = \beta v^k + \nabla f(x^k) \quad \beta \in [0; 1)$$

$$x^{k+1} = x^k - \gamma v^{k+1}$$

Вопрос: как это метод связан с методом тяжелого шарика?

На чем держится ML

- В библиотеке pytorch (основная библиотека Deep Learning) реализован следующий метод:

$$v^{k+1} = \beta v^k + \nabla f(x^k) \quad \beta \in [0; 1)$$

$$x^{k+1} = x^k - \gamma v^{k+1}$$

Вопрос: как это метод связан с методом тяжелого шарика? Это практически он и есть. Поставим первую строку во вторую:

$$x^{k+1} = x^k - \gamma \nabla f(x^k) - \gamma \beta v^k$$

На чем держится ML

- В библиотеке pytorch (основная библиотека Deep Learning) реализован следующий метод:

$$v^{k+1} = \beta v^k + \nabla f(x^k) \quad \beta \in [0; 1)$$

$$x^{k+1} = x^k - \gamma v^{k+1}$$

Вопрос: как это метод связан с методом тяжелого шарика? Это практически он и есть. Поставим первую строку во вторую:

$$x^{k+1} = x^k - \gamma \nabla f(x^k) - \gamma \beta v^k$$

Из второй строки для k шага:

$$-\gamma v^k = x^k - x^{k-1}$$

На чем держится ML

- В библиотеке pytorch (основная библиотека Deep Learning) реализован следующий метод:

$$v^{k+1} = \beta v^k + \nabla f(x^k) \quad \beta \in [0; 1)$$

$$x^{k+1} = x^k - \gamma v^{k+1}$$

Вопрос: как это метод связан с методом тяжелого шарика? Это практически он и есть. Поставим первую строку во вторую:

$$x^{k+1} = x^k - \gamma \nabla f(x^k) - \gamma \beta v^k$$

Из второй строки для k шага:

$$-\gamma v^k = x^k - x^{k-1}$$

Тогда подставим в предыдущие и получим

$$x^{k+1} = x^k - \gamma \nabla f(x^k) + \beta(x^k - x^{k-1})$$

Это показывает еще одну физику метода тяжелого шарика – мы идем по аккумулированному градиенту (старые забываются).

Плюсы и минусы

Вопрос: какие плюсы и минусы видите у методы тяжелого шарика?

Плюсы и минусы

Вопрос: какие плюсы и минусы видите у методы тяжелого шарика?

Плюсы

- Понятная физика и интуиция.
- Легкость в имплантации.
- Дешевизна вычислений.

Минусы

- Нужно подбирать теперь 2 параметра. Мы сейчас умеем только в теории оценивать γ_k. Теперь что-то нужно делать с τ_k... Типично τ_k берут близким к единице или устремляют к единице.
- Мы шли за ускорением градиентного спуска. А оно вообще есть в общем случае?

Плюсы и минусы

Вопрос: какие плюсы и минусы видите у методы тяжелого шарика?

Плюсы

- Понятная физика и интуиция.
- Легкость в имплантации.
- Дешевизна вычислений.

Минусы

- Нужно подбирать теперь 2 параметра. Мы сейчас умеем только в теории оценивать γ_k. Теперь что-то нужно делать с τ_k... Типично τ_k берут близким к единице или устремляют к единице.
- Мы шли за ускорением градиентного спуска. А оно вообще есть в общем случае? Нет...

Ускоренный градиентный метод

- Ю.Е. Нестеров в 1983 году предложил ускоренный градиентный метод.

Алгоритм 3 Ускоренный градиентный метод

Вход: размер шагов $\{\gamma_k\}_{k=0} > 0$, моментумы $\{\tau_k\}_{k=0} \in [0; 1]$, стартовая точка $x^0 = y^0 \in \mathbb{R}^d$, количество итераций K

1: **for** $k = 0, 1, \ldots, K-1$ **do**
2: Вычислить $\nabla f(y^k)$
3: $x^{k+1} = y^k - \gamma_k \nabla f(y^k)$
4: $y^{k+1} = x^{k+1} + \tau_k(x^{k+1} - x^k)$
5: **end for**

Выход: x^K

Ускоренный градиентный метод и тяжелый шарик

- **Вопрос:**В чем ключевое отличие метода Нестерова от тяжелого шарика?
  Тяжелый шарик:

$$x^{k+1} = x^k - \gamma_k \nabla f(x^k) + \tau_k (x^k - x^{k-1})$$

  Ускоренный градиентный метод:

$$x^{k+1} = y^k - \gamma_k \nabla f(y^k)$$

$$y^{k+1} = x^{k+1} + \tau_k (x^{k+1} - x^k)$$

Ускоренный градиентный метод и тяжелый шарик

- **Вопрос:**В чем ключевое отличие метода Нестерова от тяжелого шарика?
  Тяжелый шарик:
  $$x^{k+1} = x^k - \gamma_k \nabla f(x^k) + \tau_k (x^k - x^{k-1})$$
  Ускоренный градиентный метод:
  $$x^{k+1} = y^k - \gamma_k \nabla f(y^k)$$
  $$y^{k+1} = x^{k+1} + \tau_k (x^{k+1} - x^k)$$

- Перепишем ускоренный градиентный метод:
  $$x^{k+1} = x^k + \tau_k (x^k - x^{k-1}) - \gamma_k \nabla f(x^k + \tau_k (x^k - x^{k-1})).$$
  Моментум в точке подсчета градиента/«взгляд вперед»/экстраполяция

Ускоренный градиентный метод и тяжелый шарик

- Сходимость метода Нестерова доказана в пособии.
- Сейчас существуют модификации идеи Нестерова, которые также позволяют добиваться того же результата.

Алгоритм 4 Линейный каплинг: внутренний цикл

Вход: размер шагов $\{\gamma_k\}_{k=0} > 0$ и $\{\eta_k\}_{k=0} > 0$, моментумы $\{\tau_k\}_{k=0} \in [0; 1]$, стартовая точка $x^0 = y^0 = z^0 \in \mathbb{R}^d$, количество итераций K

1: **for** $k = 0, 1, \ldots, K-1$ **do**
2: Вычислить $\nabla f(x^k)$
3: $y^{k+1} = x^k - \eta_k \nabla f(x^k)$
4: $z^{k+1} = z^k - \gamma_k \nabla f(x^k)$
5: $x^{k+1} = \tau_k z^{k+1} + (1 - \tau_k) y^{k+1}$
6: **end for**

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

Нам понадобится

- Сам метод (зафиксировали параметры):

Алгоритм 5 Линейный каплинг: внутренний цикл

Вход: размер шагов $\gamma > 0$ и $\eta > 0$, моментум $\tau \in [0; 1]$, стартовая точка $x^0 = y^0 = z^0 \in \mathbb{R}^d$, количество итераций K

1: **for** $k = 0, 1, \ldots, K-1$ **do**
2: Вычислить $\nabla f(x^k)$
3: $y^{k+1} = x^k - \eta \nabla f(x^k)$
4: $z^{k+1} = z^k - \gamma \nabla f(x^k)$
5: $x^{k+1} = \tau z^{k+1} + (1-\tau) y^{k+1}$
6: **end for**

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

- А также выпуклость и гладкость:

$$\frac{\mu}{2}\|x-y\|_2^2 \leq f(y) - f(x) - \langle \nabla f(x), y - x \rangle \leq \frac{L}{2}\|x-y\|_2^2.$$

Доказательство

Воспользуемся линией 4 Алгоритма 5:

$$\begin{aligned}\|z^{k+1} - x^*\|_2^2 =& \|z^k - \gamma\nabla f(x^k) - x^*\|_2^2 \\ =& \|z^k - x^*\|_2^2 - 2\gamma\langle\nabla f(x^k), z^k - x^*\rangle + \gamma^2\|\nabla f(x^k)\|_2^2 \\ =& \|z^k - x^*\|_2^2 - 2\gamma\langle\nabla f(x^k), x^k - x^*\rangle \\ & - 2\gamma\langle\nabla f(x^k), z^k - x^k\rangle + \gamma^2\|\nabla f(x^k)\|_2^2. \end{aligned} \tag{1}$$

Доказательство

Воспользуемся линией 4 Алгоритма 5:

$$\begin{aligned}\|z^{k+1} - x^*\|_2^2 =& \|z^k - \gamma\nabla f(x^k) - x^*\|_2^2 \\ =& \|z^k - x^*\|_2^2 - 2\gamma\langle\nabla f(x^k), z^k - x^*\rangle + \gamma^2\|\nabla f(x^k)\|_2^2 \\ =& \|z^k - x^*\|_2^2 - 2\gamma\langle\nabla f(x^k), x^k - x^*\rangle \\ & - 2\gamma\langle\nabla f(x^k), z^k - x^k\rangle + \gamma^2\|\nabla f(x^k)\|_2^2. \end{aligned} \tag{1}$$

Оценим $\left[-\langle\nabla f(x^k), z^k - x^k\rangle\right]$ и $\|\nabla f(x^k)\|_2^2$.

Доказательство

Начнем с $\|\nabla f(x^k)\|_2^2$. Для этого применим свойство гладкой.

$$f(y^{k+1}) \leq f(x^k) + \langle \nabla f(x^k), y^{k+1} - x^k \rangle + \frac{L}{2}\|y^{k+1} - x^k\|_2^2.$$

Доказательство

Начнем с $\|\nabla f(x^k)\|_2^2$. Для этого применим свойство гладкой.

$$f(y^{k+1}) \leq f(x^k) + \langle \nabla f(x^k), y^{k+1} - x^k \rangle + \frac{L}{2}\|y^{k+1} - x^k\|_2^2.$$

Подставим итерационный шаг для y^{k+1} (линия 3 Алгоритма 5):

$$\begin{aligned} f(y^{k+1}) \leq & f(x^k) - \eta\|\nabla f(x^k)\|_2^2 + \frac{L\eta^2}{2}\|\nabla f(x^k)\|_2^2. \\ = & f(x^k) - \eta\left(1 - \frac{L\eta}{2}\right)\|\nabla f(x^k)\|_2^2. \end{aligned}$$

Доказательство

Начнем с $\|\nabla f(x^k)\|_2^2$. Для этого применим свойство гладкой.

$$f(y^{k+1}) \leq f(x^k) + \langle \nabla f(x^k), y^{k+1} - x^k \rangle + \frac{L}{2}\|y^{k+1} - x^k\|_2^2.$$

Подставим итерационный шаг для y^{k+1} (линия 3 Алгоритма 5):

$$\begin{aligned} f(y^{k+1}) \leq & f(x^k) - \eta\|\nabla f(x^k)\|_2^2 + \frac{L\eta^2}{2}\|\nabla f(x^k)\|_2^2. \\ = & f(x^k) - \eta\left(1 - \frac{L\eta}{2}\right)\|\nabla f(x^k)\|_2^2. \end{aligned}$$

Выберем $\eta \in \left(0; \frac{2}{L}\right)$, тогда можно гарантировать, что $\left(1 - \frac{L\eta}{2}\right) > 0$, а значит

$$\|\nabla f(x^k)\|_2^2 \leq \frac{2}{\eta(2 - L\eta)}(f(x^k) - f(y^{k+1})). \tag{2}$$

Доказательство

Соединяем (1) и (2):

$$\begin{aligned}\|z^{k+1} - x^*\|_2^2 \leq & \|z^k - x^*\|_2^2 - 2\gamma\langle\nabla f(x^k), x^k - x^*\rangle \\ & + \frac{2\gamma^2}{\eta(2 - L\eta)}(f(x^k) - f(y^{k+1})) \\ & + 2\gamma\langle\nabla f(x^k), x^k - z^k\rangle. \end{aligned} \tag{3}$$

Доказательство

Соединяем (1) и (2):

$$\begin{aligned}\|z^{k+1} - x^*\|_2^2 \leq & \|z^k - x^*\|_2^2 - 2\gamma\langle\nabla f(x^k), x^k - x^*\rangle \\ & + \frac{2\gamma^2}{\eta(2 - L\eta)}(f(x^k) - f(y^{k+1})) \\ & + 2\gamma\langle\nabla f(x^k), x^k - z^k\rangle. \end{aligned} \tag{3}$$

Осталось $\left[-\langle\nabla f(x^k), z^k - x^k\rangle\right]$.

Доказательство

Воспользуемся линией 5 Алгоритма 5:

$$\begin{aligned}\langle \nabla f(x^k), x^k - z^k \rangle =& \langle \nabla f(x^k), x^k - \frac{1}{\tau}(x^k - (1-\tau)y^k) \rangle \\ =& \frac{1-\tau}{\tau} \langle \nabla f(x^k), y^k - x^k \rangle.\end{aligned}$$

Доказательство

Воспользуемся линией 5 Алгоритма 5:

$$\begin{aligned}\langle \nabla f(x^k), x^k - z^k \rangle =& \langle \nabla f(x^k), x^k - \frac{1}{\tau}(x^k - (1-\tau)y^k) \rangle \\ =& \frac{1-\tau}{\tau} \langle \nabla f(x^k), y^k - x^k \rangle.\end{aligned}$$

Далее пользуемся выпуклостью:

$$\begin{aligned}\langle \nabla f(x^k), x^k - z^k \rangle =& \frac{1-\tau}{\tau} \langle \nabla f(x^k), y^k - x^k \rangle \\ \leq& \frac{1-\tau}{\tau}(f(y^k) - f(x^k)). \qquad (4)\end{aligned}$$

Доказательство

Соединяем (3) и (4):

$$\begin{aligned}\|z^{k+1} - x^*\|_2^2 \leq &\|z^k - x^*\|_2^2 - 2\gamma\langle\nabla f(x^k), x^k - x^*\rangle \\ &+ \frac{2\gamma^2}{\eta(2 - L\eta)}(f(x^k) - f(y^{k+1})) \\ &+ 2\gamma \cdot \frac{1-\tau}{\tau}(f(y^k) - f(x^k)).\end{aligned}$$

Подгоним параметры следующим образом $\frac{\gamma}{\eta(2-L\eta)} = \frac{1-\tau}{\tau}$:

$$\begin{aligned}\|z^{k+1} - x^*\|_2^2 \leq &\|z^k - x^*\|_2^2 - 2\gamma\langle\nabla f(x^k), x^k - x^*\rangle \\ &+ \frac{2\gamma^2}{\eta(2 - L\eta)}(f(y^k) - f(y^{k+1})).\end{aligned}$$

Доказательство

Переставляем:

$$2\gamma\langle\nabla f(x^k), x^k - x^*\rangle \leq \|z^k - x^*\|_2^2 - \|z^{k+1} - x^*\|_2^2 + \frac{2\gamma^2}{\eta(2 - L\eta)}(f(y^k) - f(y^{k+1})).$$

Доказательство

Переставляем:

$$2\gamma\langle\nabla f(x^k), x^k - x^*\rangle \leq \|z^k - x^*\|_2^2 - \|z^{k+1} - x^*\|_2^2 + \frac{2\gamma^2}{\eta(2 - L\eta)}(f(y^k) - f(y^{k+1})).$$

Далее выпуклость:

$$2\gamma(f(x^k) - f(x^*)) \leq \|z^k - x^*\|_2^2 - \|z^{k+1} - x^*\|_2^2 + \frac{2\gamma^2}{\eta(2 - L\eta)}(f(y^k) - f(y^{k+1})).$$

Доказательство

Суммируем по k и усредняем:

$$\begin{aligned}\frac{2\gamma}{K}\sum_{k=0}^{K-1}(f(x^k)-f(x^*)) \leq & \frac{1}{K}\sum_{k=0}^{K-1}\left(\|z^k-x^*\|_2^2-\|z^{k+1}-x^*\|_2^2\right)\\ &+\frac{2\gamma^2}{\eta(2-L\eta)K}\sum_{k=0}^{K-1}(f(y^k)-f(y^{k+1}))\\ =&\frac{1}{K}\left(\|z^0-x^*\|_2^2-\|z^K-x^*\|_2^2\right)\\ &+\frac{2\gamma^2}{\eta(2-L\eta)K}(f(y^0)-f(y^K))\\ \leq&\frac{\|x^0-x^*\|_2^2}{K}+\frac{2\gamma^2(f(y^0)-f(x^*))}{\eta(2-L\eta)K}.\end{aligned}$$

Доказательство

Подставим начальные условия: $x^0 = y^0 = z^0$ и применим неравенство Йенсена:

$$2\gamma \left[f\left(\frac{1}{K} \sum_{k=0}^{K-1} x^k \right) - f(x^*) \right] \leq \frac{\|x^0 - x^*\|_2^2}{K} + \frac{2\gamma^2 (f(x^0) - f(x^*))}{\eta(2 - L\eta)K}.$$

Доказательство

Подставим начальные условия: $x^0 = y^0 = z^0$ и применим неравенство Йенсена:

$$2\gamma \left[f\left(\frac{1}{K} \sum_{k=0}^{K-1} x^k \right) - f(x^*) \right] \leq \frac{\|x^0 - x^*\|_2^2}{K} + \frac{2\gamma^2 (f(x^0) - f(x^*))}{\eta(2 - L\eta)K}.$$

Далее μ-сильная выпуклость

$$f\left(\frac{1}{K} \sum_{k=0}^{K-1} x^k \right) - f(x^*) \leq \frac{f(x^0) - f(x^*)}{\mu\gamma K} + \frac{\gamma (f(x^0) - f(x^*))}{\eta(2 - L\eta)K}$$

$$= \left(\frac{1}{\mu\gamma K} + \frac{\gamma}{\eta(2 - L\eta)K} \right) (f(x^0) - f(x^*)).$$

Доказательство

Оптимизируем оценку с помощью выбора $\eta = \frac{1}{L}$:

$$f\left(\frac{1}{K}\sum_{k=0}^{K-1} x^k\right) - f(x^*) \leq \left(\frac{1}{\mu\gamma K} + \frac{\gamma L}{K}\right)(f(x^0) - f(x^*)).$$

Доказательство

Оптимизируем оценку с помощью выбора $\eta = \frac{1}{L}$:

$$f\left(\frac{1}{K}\sum_{k=0}^{K-1} x^k\right) - f(x^*) \leq \left(\frac{1}{\mu\gamma K} + \frac{\gamma L}{K}\right)(f(x^0) - f(x^*)).$$

И еще раз $\gamma = \sqrt{\frac{1}{\mu L}}$:

$$f\left(\frac{1}{K}\sum_{k=0}^{K-1} x^k\right) - f(x^*) \leq \sqrt{\frac{4L}{\mu K^2}}(f(x^0) - f(x^*)).$$

Доказательство

А теперь $K = \sqrt{\frac{16L}{\mu}}$

$$f\left(\frac{1}{K}\sum_{k=0}^{K-1} x^k\right) - f(x^*) \leq \frac{1}{2}(f(x^0) - f(x^*)).$$

Доказательство

А теперь $K = \sqrt{\frac{16L}{\mu}}$

$$f\left(\frac{1}{K}\sum_{k=0}^{K-1} x^k\right) - f(x^*) \leq \frac{1}{2}(f(x^0) - f(x^*)).$$

Вопрос: а зачем?

Доказательство

А теперь $K = \sqrt{\frac{16L}{\mu}}$

$$f\left(\frac{1}{K}\sum_{k=0}^{K-1} x^k\right) - f(x^*) \leq \frac{1}{2}(f(x^0) - f(x^*)).$$

Вопрос: а зачем? за K итераций мы гарантированно «приблизились к решению» в 2 раза. Тогда пусть это одна итерация нашего нового внешнего алгоритма. Т.е. мы запускаем линейный каплинг на K итераций, а потому перезапускаем с новой стартовой точкой $\frac{1}{K}\sum_{k=0}^{K-1} x^k$, взятой из прошлого запуска каплинга. Это называется рестарты.

Доказательство

Тогда, если сделать T рестартов:

$$f\left(x^T\right) - f(x^*) \leq \frac{1}{2^T}(f(x^0) - f(x^*)).$$

Откуда можно сразу же получить оракульную сложность:

$$f\left(x^T\right) - f(x^*) \leq \frac{1}{2^T}(f(x^0) - f(x^*)) \leq \varepsilon.$$

$$T \geq \log_2\left(\frac{f(x^0) - f(x^*)}{\varepsilon}\right)$$

$$K \cdot T = O\left(\sqrt{\frac{L}{\mu}}\log_2\frac{f(x^0) - f(x^*)}{\varepsilon}\right) \quad \text{вызовов оракула.}$$

Сходимость линейного каплинга

О сходимости линейного каплинга

Пусть задача безусловной оптимизации с L-гладкой, μ-сильно выпуклой целевой функцией f решается с помощью реставрированного линейного каплинга. Тогда при $\eta = \frac{1}{L}$, $\gamma = \sqrt{\frac{1}{\mu L}}$ и $K = \sqrt{\frac{16L}{\mu}}$, чтобы добиться точности ε по функции ($f(x) - f(x^*) \leq \varepsilon$), необходимо

$$O\left(\sqrt{\frac{L}{\mu}} \log \frac{f(x^0) - f(x^*)}{\varepsilon}\right)$$ вызовов оракула.

- Ускоренный градиентный метод Нестерова имеет точно такую же оценку на количество вызовов оракула.

Вопросы остаются

- Метод лучше градиентного спуска.
- Но можно ли еще лучше?
- **Вопрос:** как понять, можно ли лучше?

Вопросы остаются

- Метод лучше градиентного спуска.
- Но можно ли еще лучше?
- **Вопрос:** как понять, можно ли лучше? получить нижние оценки.

Вопросы остаются

- Метод лучше градиентного спуска.
- Но можно ли еще лучше?
- **Вопрос:** как понять, можно ли лучше? получить нижние оценки.
- Для получения нижних оценок нужно придумать не метод, а «плохую» функцию, которую будет «долго» оптимизировать любой метод. **Вопрос:** а что здесь значит «любой метод»?

Класс алгоритмов

- Дана начальная точка x^0. Эта начальная точка порождает некоторое множество M_0 – множество всех достигнутых на данный момент точек (на данном шаге k). $M_0 = \{x^0\}$.

Класс алгоритмов

- Дана начальная точка x^0. Эта начальная точка порождает некоторое множество M_0 – множество всех достигнутых на данный момент точек (на данном шаге k). $M_0 = \{x^0\}$.
- На текущем оракульном вызове метод может считать градиент функции в точке x^k: $\nabla f(x^k)$, где $x^k \in M_k$, то есть метод может посчитать градиент во всех точках, которые уже достиг. Изначально можем посчитать градиент только в x^0.

Класс алгоритмов

- Дана начальная точка x^0. Эта начальная точка порождает некоторое множество M_0 – множество всех достигнутых на данный момент точек (на данном шаге k). $M_0 = \{x^0\}$.
- На текущем оракульном вызове метод может считать градиент функции в точке x^k: $\nabla f(x^k)$, где $x^k \in M_k$, то есть метод может посчитать градиент во всех точках, которые уже достиг. Изначально можем посчитать градиент только в x^0.
- $M_{k+1} = \text{span}\{M_k, \nabla f(x')\}$ (линейная оболочка), где $x' \in M_k$.

Класс алгоритмов

- Дана начальная точка x^0. Эта начальная точка порождает некоторое множество M_0 – множество всех достигнутых на данный момент точек (на данном шаге k). $M_0 = \{x^0\}$.
- На текущем оракульном вызове метод может считать градиент функции в точке x^k: $\nabla f(x^k)$, где $x^k \in M_k$, то есть метод может посчитать градиент во всех точках, которые уже достиг. Изначально можем посчитать градиент только в x^0.
- $M_{k+1} = \text{span}\{M_k, \nabla f(x')\}$ (линейная оболочка), где $x' \in M_k$.
- После K вызовов оракула выход метода есть некоторая точка из M_K.

Класс алгоритмов

- Дана начальная точка x^0. Эта начальная точка порождает некоторое множество M_0 – множество всех достигнутых на данный момент точек (на данном шаге k). $M_0 = \{x^0\}$.
- На текущем оракульном вызове метод может считать градиент функции в точке x^k: $\nabla f(x^k)$, где $x^k \in M_k$, то есть метод может посчитать градиент во всех точках, которые уже достиг. Изначально можем посчитать градиент только в x^0.
- $M_{k+1} = \text{span}\{M_k, \nabla f(x')\}$ (линейная оболочка), где $x' \in M_k$.
- После K вызовов оракула выход метода есть некоторая точка из M_K.

Вопрос: подходят ли изученные методы, под такое определение?

Класс алгоритмов

- Дана начальная точка x^0. Эта начальная точка порождает некоторое множество M_0 – множество всех достигнутых на данный момент точек (на данном шаге k). $M_0 = \{x^0\}$.
- На текущем оракульном вызове метод может считать градиент функции в точке x^k: $\nabla f(x^k)$, где $x^k \in M_k$, то есть метод может посчитать градиент во всех точках, которые уже достиг. Изначально можем посчитать градиент только в x^0.
- $M_{k+1} = \text{span}\{M_k, \nabla f(x')\}$ (линейная оболочка), где $x' \in M_k$.
- После K вызовов оракула выход метода есть некоторая точка из M_K.

Вопрос: подходят ли изученные методы, под такое определение? да, градиентный спуск, метод тяжелого шарика, линейный каплинг и ускоренный градиентный метод.

Класс алгоритмов

- Дана начальная точка x^0. Эта начальная точка порождает некоторое множество M_0 – множество всех достигнутых на данный момент точек (на данном шаге k). $M_0 = \{x^0\}$.
- На текущем оракульном вызове метод может считать градиент функции в точке x^k: $\nabla f(x^k)$, где $x^k \in M_k$, то есть метод может посчитать градиент во всех точках, которые уже достиг. Изначально можем посчитать градиент только в x^0.
- $M_{k+1} = \text{span}\{M_k, \nabla f(x')\}$ (линейная оболочка), где $x' \in M_k$.
- После K вызовов оракула выход метода есть некоторая точка из M_K.

Вопрос: подходят ли изученные методы, под такое определение? да, градиентный спуск, метод тяжелого шарика, линейный каплинг и ускоренный градиентный метод.

Вопрос: все ли методы, которые считают градиент здесь учтены?

Класс алгоритмов

- Дана начальная точка x^0. Эта начальная точка порождает некоторое множество M_0 – множество всех достигнутых на данный момент точек (на данном шаге k). $M_0 = \{x^0\}$.
- На текущем оракульном вызове метод может считать градиент функции в точке x^k: $\nabla f(x^k)$, где $x^k \in M_k$, то есть метод может посчитать градиент во всех точках, которые уже достиг. Изначально можем посчитать градиент только в x^0.
- $M_{k+1} = \text{span}\{M_k, \nabla f(x')\}$ (линейная оболочка), где $x' \in M_k$.
- После K вызовов оракула выход метода есть некоторая точка из M_K.

Вопрос: подходят ли изученные методы, под такое определение? да, градиентный спуск, метод тяжелого шарика, линейный каплинг и ускоренный градиентный метод.

Вопрос: все ли методы, которые считают градиент здесь учтены? нет, но это вопрос уже не сегодняшней лекции.

«Плохая» функция

Квадратичная (ее достаточно) функция:

$$f(x) = \frac{L-\mu}{8} x^T A x + \frac{\mu}{2} x^T x - \frac{L-\mu}{4} e_1^T x,$$

где

$$A = \begin{pmatrix} 2 & -1 & & 0 \\ -1 & 2 & -1 & \\ & -1 & 2 & \ddots \\ & & \ddots & \ddots & -1 \\ & 0 & & -1 & \zeta \end{pmatrix}, \quad e_1 = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix},$$

ζ определим позже.

«Плохая» функция

Квадратичная (ее достаточно) функция:

$$f(x) = \frac{L-\mu}{8} x^T A x + \frac{\mu}{2} x^T x - \frac{L-\mu}{4} e_1^T x,$$

где

$$A = \begin{pmatrix} 2 & -1 & & 0 \\ -1 & 2 & -1 & \\ & -1 & 2 & \ddots \\ & & \ddots & \ddots & -1 \\ & 0 & & -1 & \zeta \end{pmatrix}, \quad e_1 = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix},$$

ζ определим позже.

Функция L-гладкая и μ-сильно выпуклая (задача из домашнего задания).

«Плохая» функция: решение

Вопрос: что можем сказать про решение?

«Плохая» функция: решение

Вопрос: что можем сказать про решение? Сильная выпуклая задача — единственное решение.

«Плохая» функция: решение

Вопрос: что можем сказать про решение? Сильная выпуклая задача — единственное решение.

Вопрос: как найти?

«Плохая» функция: решение

Вопрос: что можем сказать про решение? Сильная выпуклая задача — единственное решение.

Вопрос: как найти? Условие оптимальности:

$$\nabla f(x^*) = 0$$

или

$$Ax^* + \frac{4\mu}{L-\mu}x^* - e_1 = 0$$

«Плохая» функция: решение

Вопрос: что можем сказать про решение? Сильная выпуклая задача — единственное решение.

Вопрос: как найти? Условие оптимальности:

$$\nabla f(x^*) = 0$$

или

$$Ax^* + \frac{4\mu}{L-\mu}x^* - e_1 = 0$$

Распишем покомпонентно. Первая компонента:

$$2x_1^* - x_2^* + \frac{4\mu}{L-\mu}x_1^* - 1 = 0 \text{ или } \frac{2(L+\mu)}{L-\mu} \cdot x_1^* - x_2^* = 1$$

«Плохая» функция: решение

Вопрос: что можем сказать про решение? Сильная выпуклая задача — единственное решение.

Вопрос: как найти? Условие оптимальности:

$$\nabla f(x^*) = 0$$

или

$$Ax^* + \frac{4\mu}{L-\mu}x^* - e_1 = 0$$

Распишем покомпонентно. Первая компонента:

$$2x_1^* - x_2^* + \frac{4\mu}{L-\mu}x_1^* - 1 = 0 \text{ или } \frac{2(L+\mu)}{L-\mu} \cdot x_1^* - x_2^* = 1$$

Все координаты (кроме первой и последней):

$$-x_{k-1}^* + \frac{2(L+\mu)}{L-\mu}x_k^* - x_{k+1}^* = 0$$

«Плохая» функция: решение

Последняя координата:

$$-x_{d-1}^* + \zeta x_d^* + \frac{4\mu}{L-\mu} x_d^* = 0 \text{ или } -x_{d-1}^* + \left(\zeta + \frac{4\mu}{L-\mu}\right) x_d^* = 0$$

«Плохая» функция: решение

Последняя координата:

$$-x_{d-1}^* + \zeta x_d^* + \frac{4\mu}{L-\mu} x_d^* = 0 \text{ или } -x_{d-1}^* + \left(\zeta + \frac{4\mu}{L-\mu}\right) x_d^* = 0$$

Можно заметить, что все уравнения (кроме 1го и последнего) просто линейная рекуррента. Решение будет следующим, если правильно подобрать ζ:

$$x_k^* = q^k, \qquad q = \frac{\sqrt{L} - \sqrt{\mu}}{\sqrt{L} + \sqrt{\mu}}$$

«Плохая» функция: продвижение к решению

Более того, возьмём стартовую точку $x^0 = (0 \dots 0)^T$.

«Плохая» функция: продвижение к решению

Более того, возьмём стартовую точку $x^0 = (0 \dots 0)^T$.
Градиент:

$$\nabla f(x) = \frac{L - \mu}{4} Ax + \mu x - \frac{L - \mu}{4} e_1$$

«Плохая» функция: продвижение к решению

Более того, возьмём стартовую точку $x^0 = (0 \dots 0)^T$.
Градиент:

$$\nabla f(x) = \frac{L - \mu}{4} Ax + \mu x - \frac{L - \mu}{4} e_1$$

Заметим, что $\nabla f(x^0) \in \text{span}(e_1)$,

«Плохая» функция: продвижение к решению

Более того, возьмём стартовую точку $x^0 = (0 \dots 0)^T$.
Градиент:

$$\nabla f(x) = \frac{L - \mu}{4} Ax + \mu x - \frac{L - \mu}{4} e_1$$

Заметим, что $\nabla f(x^0) \in \text{span}(e_1)$, поэтому получается, что за первый оракульный вызов только первая координата выхода метода может быть ненулевой

«Плохая» функция: продвижение к решению

Более того, возьмём стартовую точку $x^0 = (0 \dots 0)^T$.
Градиент:

$$\nabla f(x) = \frac{L-\mu}{4} Ax + \mu x - \frac{L-\mu}{4} e_1$$

Заметим, что $\nabla f(x^0) \in \text{span}(e_1)$, поэтому получается, что за первый оракульный вызов только первая координата выхода метода может быть ненулевой
После второго вызова оракула: $\nabla f(x^1) \in \text{span}(e_1, e_2), \quad x^1 \in M_1$, то есть за 2 оракульных вызова максимум 2 первых координаты могут быть ненулевыми.

«Плохая» функция: продвижение к решению

Более того, возьмём стартовую точку $x^0 = (0 \dots 0)^T$.
Градиент:

$$\nabla f(x) = \frac{L-\mu}{4} Ax + \mu x - \frac{L-\mu}{4} e_1$$

Заметим, что $\nabla f(x^0) \in \text{span}(e_1)$, поэтому получается, что за первый оракульный вызов только первая координата выхода метода может быть ненулевой
После второго вызова оракула: $\nabla f(x^1) \in \text{span}(e_1, e_2), \quad x^1 \in M_1$, то есть за 2 оракульных вызова максимум 2 первых координаты могут быть ненулевыми.
После K оракульных вызовов только первые K координат могут быть ненулевые, остальные точно нулевые.

«Плохая» функция: гарантии

Возьмем $d = 2K$, где K - кол-во вызовов оракула. **Вопрос:** зачем?

«Плохая» функция: гарантии

Возьмем $d = 2K$, где K - кол-во вызовов оракула. **Вопрос:** зачем?
Изначальное расстояние до решения:

$$\|x^0 - x^*\|_2^2 = \sum_{i=1}^{2K} q^{2i} = (1 + q^{2K}) \sum_{i=1}^{K} q^{2i}$$

«Плохая» функция: гарантии

Возьмем $d = 2K$, где K - кол-во вызовов оракула. **Вопрос:** зачем? Изначальное расстояние до решения:

$$\|x^0 - x^*\|_2^2 = \sum_{i=1}^{2K} q^{2i} = (1 + q^{2K}) \sum_{i=1}^{K} q^{2i}$$

После K вызовов оракула итоговый вывод можно оценить так (только первые K координат ненулевые):

$$\|x^K - x^*\|^2 \geq \sum_{i=K+1}^{2K} q^{2i} = q^{2K} \sum_{i=1}^{K} q^{2i} = \frac{q^{2K}}{1 + q^{2K}} \|x^0 - x^*\|_2^2$$

$$\geq \frac{q^{2K}}{2} \|x^0 - x^*\|_2^2 = \left(1 - \frac{2\sqrt{\mu}}{\sqrt{L} + \sqrt{\mu}}\right)^{2K} \frac{\|x^0 - x^*\|_2^2}{2}$$

Нижняя оценка на оракульную сложность

Нижняя оценка на оракульную сложность

Для любого метода из класса, описанного выше, существует безусловная задача оптимизации с L-гладкой, μ-сильно выпуклой целевой функцией f такая, что для решения этой задачи методу необходимо

$$\Omega\left(\sqrt{\frac{L}{\mu}}\log\frac{\|x^0 - x^*\|_2}{\varepsilon}\right) \text{ вызовов оракула.}$$

Нижняя оценка на оракульную сложность

Нижняя оценка на оракульную сложность

Для любого метода из класса, описанного выше, существует безусловная задача оптимизации с L-гладкой, μ-сильно выпуклой целевой функцией f такая, что для решения этой задачи методу необходимо

$$\Omega\left(\sqrt{\frac{L}{\mu}}\log\frac{\|x^0 - x^*\|_2}{\varepsilon}\right) \text{ вызовов оракула.}$$

- Линейный каплинг является оптимальным методом с точки зрения оракульных вызовов для L-гладких и μ-сильно выпуклых задач.

Нижняя оценка на оракульную сложность

Нижняя оценка на оракульную сложность

Для любого метода из класса, описанного выше, существует безусловная задача оптимизации с L-гладкой, μ-сильно выпуклой целевой функцией f такая, что для решения этой задачи методу необходимо

$$\Omega\left(\sqrt{\frac{L}{\mu}}\log\frac{\|x^0 - x^*\|_2}{\varepsilon}\right) \text{ вызовов оракула.}$$

- Линейный каплинг является оптимальным методом с точки зрения оракульных вызовов для L-гладких и μ-сильно выпуклых задач.
- Для L-гладких и выпуклых задач тоже.

Нижняя оценка на оракульную сложность

Нижняя оценка на оракульную сложность

Для любого метода из класса, описанного выше, существует безусловная задача оптимизации с L-гладкой, μ-сильно выпуклой целевой функцией f такая, что для решения этой задачи методу необходимо

$$\Omega\left(\sqrt{\frac{L}{\mu}}\log\frac{\|x^0 - x^*\|_2}{\varepsilon}\right) \text{ вызовов оракула.}$$

- Линейный каплинг является оптимальным методом с точки зрения оракульных вызовов для L-гладких и μ-сильно выпуклых задач.
- Для L-гладких и выпуклых задач тоже.
- Для ускоренного градиентного метода результаты такие же.