Ускоренные градиентные методы. Идея метода сопряженных градиентов

Даня Меркулов, Петр Остроухов

Оптимизация для всех! ЦУ

Повторение

Результаты сходимости градиентного спуска для гладких функций

$(0.707)^k$

Градиентный спуск: $\min\limits_{x\in\mathbb{R}^n} f(x)$ $\quad x_{k+1} = x_k - \alpha_k \nabla f(x_k)$ $\quad \lambda\left(\nabla^2 f(x)\right) \in [\mu, L], \varkappa = \frac{L}{\mu}$

выпуклая (негладкая)	гладкая (невыпуклая)	гладкая & выпуклая	гладкая & сильно выпуклая
$f(x_k) - f^* = \mathcal{O}\left(\frac{1}{\sqrt{k}}\right)$	$\min\limits_{0\le i\le k} \|\nabla f(x_i)\| = \mathcal{O}\left(\frac{1}{\sqrt{k}}\right)$	$f(x_k) - f^* = \mathcal{O}\left(\frac{1}{k}\right)$	$\|x_k - x^*\|^2 = \mathcal{O}\left(\left(1 - \frac{\mu}{L}\right)^k\right)$
$k_\varepsilon = \mathcal{O}\left(\frac{1}{\varepsilon^2}\right)$	$k_\varepsilon = \mathcal{O}\left(\frac{1}{\varepsilon^2}\right)$	$k_\varepsilon = \mathcal{O}\left(\frac{1}{\varepsilon}\right)$	$k_\varepsilon = \mathcal{O}\left(\varkappa \log \frac{1}{\varepsilon}\right)$

СУБЛИН. ЛИН.

Нижние оценки для произвольных методов I порядка на классе гладких функций

Произвольный метод I порядка: $\min\limits_{x\in\mathbb{R}^n} f(x)$ $\quad x_{k+1} = x_k - \sum\limits_{i=0}^{k} \alpha_i \nabla f(x_i)$ $\quad \lambda\left(\nabla^2 f(x)\right) \in [\mu, L], \varkappa = \frac{L}{\mu}$

выпуклая (негладкая)	гладкая (невыпуклая)[1]	гладкая & выпуклая[2]	гладкая & сильно выпуклая
$f(x_k) - f^* = \Omega\left(\frac{1}{\sqrt{k}}\right)$	$\min\limits_{0\le i\le k} \|\nabla f(x_i)\| = \Omega\left(\frac{1}{\sqrt{k}}\right)$	$f(x_k) - f^* = \Omega\left(\frac{1}{k^2}\right)$	$f(x_k) - f^* = \Omega\left(\left(\frac{\sqrt{\varkappa}-1}{\sqrt{\varkappa}+1}\right)^{2k}\right)$
$k_\varepsilon = \Omega\left(\frac{1}{\varepsilon^2}\right)$	$k_\varepsilon = \Omega\left(\frac{1}{\varepsilon^2}\right)$	$k_\varepsilon = \Omega\left(\frac{1}{\sqrt{\varepsilon}}\right)$	$k_\varepsilon = \Omega\left(\sqrt{\varkappa}\log\frac{1}{\varepsilon}\right)$

ε

$\varepsilon_k \sim \frac{1}{k^2}$

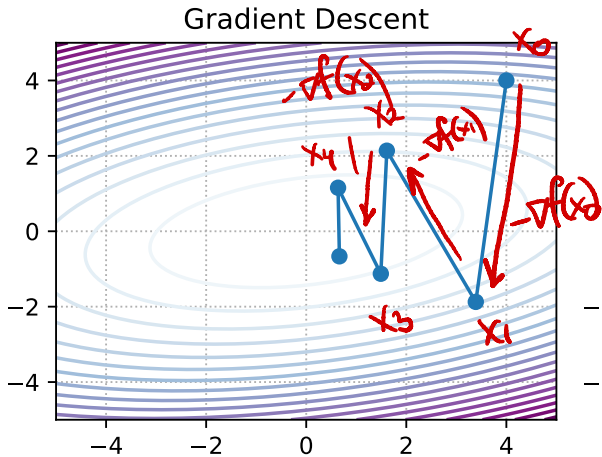
$k \sim \frac{1}{\sqrt{\varepsilon}}$

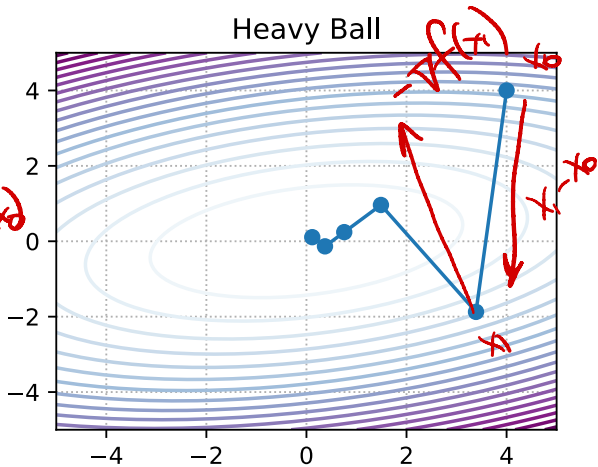
[1] Carmon, Duchi, Hinder, Sidford, 2017
[2] Nemirovski, Yudin, 1979

Метод тяжёлого шарика

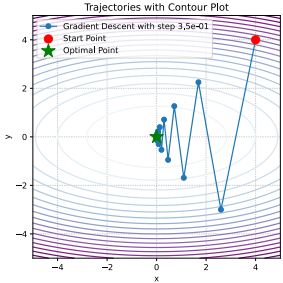
Колебания и ускорение

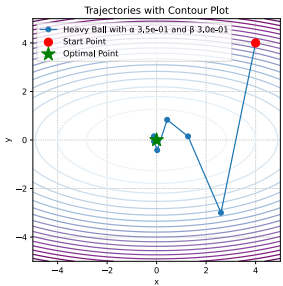
$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}).$$





Метод тяжёлого шарика Поляка





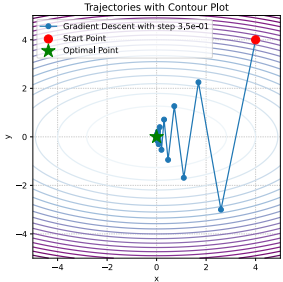
Давайте представим идею моментума (импульса, тяжёлого шарика), предложенную Б.Т. Поляком в 1964 году. Обновление метода тяжёлого шарика имеет вид

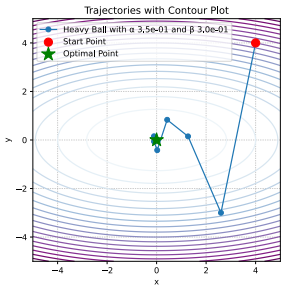
$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}).$$

$$x_k = x_{k-1} - \alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})$$

$$x_k - x_{k-1} = -\alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})$$

Метод тяжёлого шарика Поляка





Давайте представим идею моментума (импульса, тяжёлого шарика), предложенную Б.Т. Поляком в 1964 году. Обновление метода тяжёлого шарика имеет вид

$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}).$$

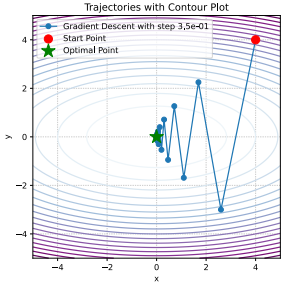
$0 \leq \beta < 1$

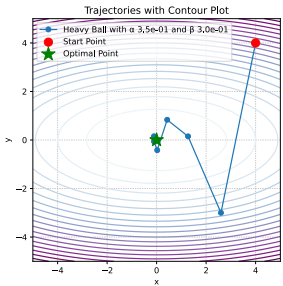
Давайте теперь подставим в итерацию предыдущую итерацию, т.е.
$x_k = x_{k-1} - \alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})$, а также отметим, что
$x_k - x_{k-1} = -\alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})$:

$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta\left[-\alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})\right]$$

$$x_{k+1} = x_k - \alpha\left[\nabla f(x_k) + \beta \nabla f(x_{k-1})\right] + \beta^2(x_{k-1} - x_{k-2})$$

Метод тяжёлого шарика Поляка





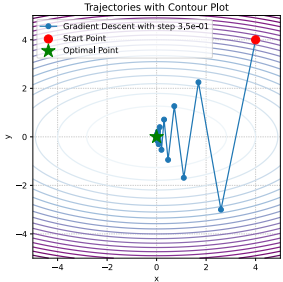
Давайте представим идею моментума (импульса, тяжёлого шарика), предложенную Б.Т. Поляком в 1964 году. Обновление метода тяжёлого шарика имеет вид

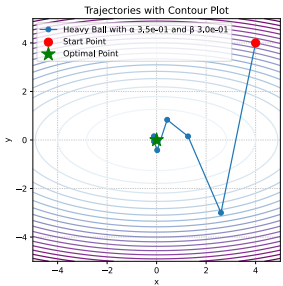
$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}).$$

Давайте теперь подставим в итерацию предыдущую итерацию, т.е. $x_k = x_{k-1} - \alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})$, а так же отметим, что $x_k - x_{k-1} = -\alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})$:

$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1})$$

Метод тяжёлого шарика Поляка





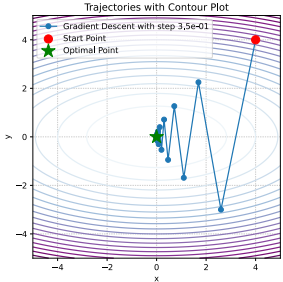
Давайте представим идею моментума (импульса, тяжёлого шарика), предложенную Б.Т. Поляком в 1964 году. Обновление метода тяжёлого шарика имеет вид

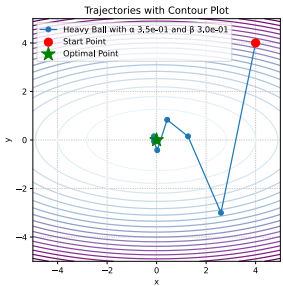
$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}).$$

Давайте теперь подставим в итерацию предыдущую итерацию, т.е. $x_k = x_{k-1} - \alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})$, а так же отметим, что $x_k - x_{k-1} = -\alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})$:

$$\begin{aligned} x_{k+1} &= x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}) \\ &= x_k - \alpha \nabla f(x_k) + \beta\left(-\alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})\right) \end{aligned}$$

Метод тяжёлого шарика Поляка





Давайте представим идею моментума (импульса, тяжёлого шарика), предложенную Б.Т. Поляком в 1964 году. Обновление метода тяжёлого шарика имеет вид

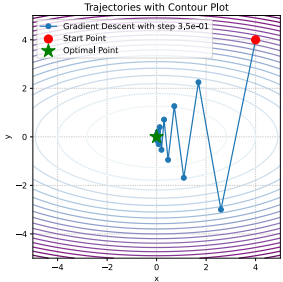
$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}).$$

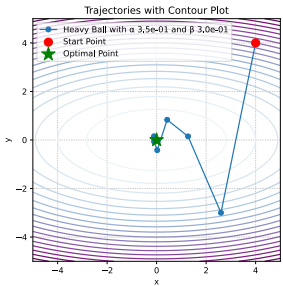
Давайте теперь подставим в итерацию предыдущую итерацию, т.е. $x_k = x_{k-1} - \alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})$, а так же отметим, что $x_k - x_{k-1} = -\alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})$:

$$\begin{aligned} x_{k+1} &= x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}) \\ &= x_k - \alpha \nabla f(x_k) + \beta\left(-\alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})\right) \\ &= x_k - \alpha\left[\nabla f(x_k) + \beta \nabla f(x_{k-1})\right] + \beta^2(x_{k-1} - x_{k-2}) = \end{aligned}$$

$$= x_k - \alpha\left[\nabla f(x_k) + \beta \nabla f(x_{k-1}) + \beta^2 \nabla f(x_{k-2}) + \beta^3 \nabla f(x_{k-3})\right]$$

Метод тяжёлого шарика Поляка





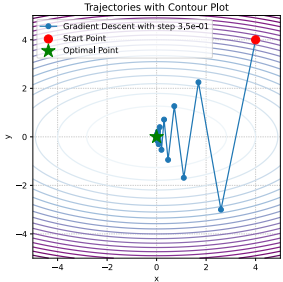
Давайте представим идею моментума (импульса, тяжёлого шарика), предложенную Б.Т. Поляком в 1964 году. Обновление метода тяжёлого шарика имеет вид

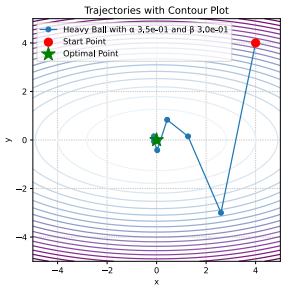
$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}).$$

Давайте теперь подставим в итерацию предыдущую итерацию, т.е. $x_k = x_{k-1} - \alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})$, а так же отметим, что $x_k - x_{k-1} = -\alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})$:

$$\begin{aligned} x_{k+1} &= x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}) \\ &= x_k - \alpha \nabla f(x_k) + \beta\left(-\alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})\right) \\ &= x_k - \alpha\left[\nabla f(x_k) + \beta \nabla f(x_{k-1})\right] + \beta^2(x_{k-1} - x_{k-2}) \\ &= x_k - \alpha\left[\nabla f(x_k) + \beta \nabla f(x_{k-1}) + \beta^2 \nabla f(x_{k-2})\right] + \beta^3(x_{k-2} - x_{k-3}) \end{aligned}$$

Метод тяжёлого шарика Поляка





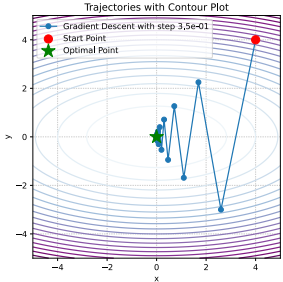
Давайте представим идею моментума (импульса, тяжёлого шарика), предложенную Б.Т. Поляком в 1964 году. Обновление метода тяжёлого шарика имеет вид

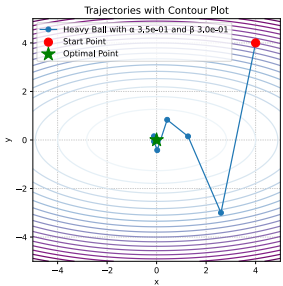
$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}).$$

Давайте теперь подставим в итерацию предыдущую итерацию, т.е. $x_k = x_{k-1} - \alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})$, а так же отметим, что $x_k - x_{k-1} = -\alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})$:

$$\begin{aligned} x_{k+1} &= x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}) \\ &= x_k - \alpha \nabla f(x_k) + \beta\left(-\alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})\right) \\ &= x_k - \alpha\left[\nabla f(x_k) + \beta \nabla f(x_{k-1})\right] + \beta^2(x_{k-1} - x_{k-2}) \\ &= x_k - \alpha\left[\nabla f(x_k) + \beta \nabla f(x_{k-1}) + \beta^2 \nabla f(x_{k-2})\right] + \beta^3(x_{k-2} - x_{k-3}) \\ &\vdots \end{aligned}$$

Метод тяжёлого шарика Поляка





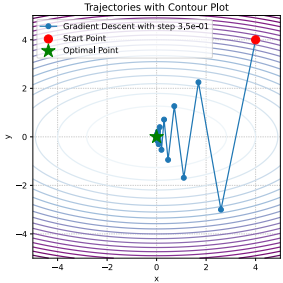
Давайте представим идею моментума (импульса, тяжёлого шарика), предложенную Б.Т. Поляком в 1964 году. Обновление метода тяжёлого шарика имеет вид

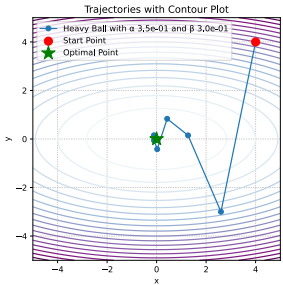
$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}).$$

Давайте теперь подставим в итерацию предыдущую итерацию, т.е. $x_k = x_{k-1} - \alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})$, а так же отметим, что $x_k - x_{k-1} = -\alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})$:

$$\begin{aligned}
x_{k+1} &= x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}) \\
&= x_k - \alpha \nabla f(x_k) + \beta\left(-\alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})\right) \\
&= x_k - \alpha\left[\nabla f(x_k) + \beta \nabla f(x_{k-1})\right] + \beta^2(x_{k-1} - x_{k-2}) \\
&= x_k - \alpha\left[\nabla f(x_k) + \beta \nabla f(x_{k-1}) + \beta^2 \nabla f(x_{k-2})\right] + \beta^3(x_{k-2} - x_{k-3}) \\
&\vdots \\
&= x_k - \alpha\left[\nabla f(x_k) + \beta \nabla f(x_{k-1}) + \beta^2 \nabla f(x_{k-2}) + \cdots + \beta^k \nabla f(x_0)\right]
\end{aligned}$$

Метод тяжёлого шарика Поляка





Давайте представим идею моментума (импульса, тяжёлого шарика), предложенную Б.Т. Поляком в 1964 году. Обновление метода тяжёлого шарика имеет вид

$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}).$$

Давайте теперь подставим в итерацию предыдущую итерацию, т.е. $x_k = x_{k-1} - \alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})$, а так же отметим, что $x_k - x_{k-1} = -\alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})$:

$$\begin{aligned}
x_{k+1} &= x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}) \\
&= x_k - \alpha \nabla f(x_k) + \beta\left(-\alpha \nabla f(x_{k-1}) + \beta(x_{k-1} - x_{k-2})\right) \\
&= x_k - \alpha\left[\nabla f(x_k) + \beta \nabla f(x_{k-1})\right] + \beta^2(x_{k-1} - x_{k-2}) \\
&= x_k - \alpha\left[\nabla f(x_k) + \beta \nabla f(x_{k-1}) + \beta^2 \nabla f(x_{k-2})\right] + \beta^3(x_{k-2} - x_{k-3}) \\
&\vdots \\
&= x_k - \alpha\left[\nabla f(x_k) + \beta \nabla f(x_{k-1}) + \beta^2 \nabla f(x_{k-2}) + \cdots + \beta^k \nabla f(x_0)\right]
\end{aligned}$$

Таким образом, метод тяжёлого шарика учитывает все предудущие градиенты с тем меньшим весом, чем старше итерация ($0 \leq \beta < 1$).

Сходимость метода тяжёлого шарика для квадратичной функции

УСКОРЕННАЯ ЛИН. СХ-ТЬ

Theorem

Предположим, что f является μ-сильно выпуклой и L-гладкой квадратичной функцией. Тогда метод тяжёлого шарика с параметрами

$$\alpha = \frac{4}{(\sqrt{L}+\sqrt{\mu})^2}, \beta = \left(\frac{\sqrt{L}-\sqrt{\mu}}{\sqrt{L}+\sqrt{\mu}}\right)^2$$

сходится линейно:

$$\|x_k - x^*\|_2 \leq \left(\frac{\sqrt{\varkappa}-1}{\sqrt{\varkappa}+1}\right)^k \|x_0 - x^*\|$$

Глобальная сходимость метода тяжёлого шарика [3]

Гарантированная сх-ть вне зависимости от x_0

Theorem

Предположим, что f является гладкой и выпуклой и что

$$\beta \in [0, 1), \quad \alpha \in \left(0, \frac{2(1-\beta)}{L}\right).$$

Тогда последовательность $\{x_k\}$, генерируемая итерациями тяжёлого шарика, удовлетворяет

$$f(\overline{x}_T) - f^\star \leq \begin{cases} \frac{\|x_0 - x^\star\|^2}{2(T+1)} \left(\frac{L\beta}{1-\beta} + \frac{1-\beta}{\alpha}\right), & \text{if } \alpha \in (0, \frac{1-\beta}{L}], \\ \frac{\|x_0 - x^\star\|^2}{2(T+1)(2(1-\beta)-\alpha L)} \left(L\beta + \frac{(1-\beta)^2}{\alpha}\right), & \text{if } \alpha \in [\frac{1-\beta}{L}, \frac{2(1-\beta)}{L}), \end{cases}$$

$\sim \frac{1}{k}$ $\sim \frac{1}{k}$

где $\overline{x}_T$ среднее Чезаро последовательности итераций, т.е.

$$\overline{x}_T = \frac{1}{T+1} \sum_{k=0}^{T} x_k.$$

НЕ УСКОРЕННАЯ СХ-ТЬ

[3] Глобальная сходимость метода тяжёлого шарика для выпуклой оптимизации, Euhanna Ghadimi et al.

Глобальная сходимость метода тяжёлого шарика [4]

Theorem

Предположим, что f является гладкой и сильно выпуклой и что

$$\alpha \in \left(0, \frac{2}{L}\right), \quad 0 \leq \beta < \frac{1}{2}\left(\frac{\mu\alpha}{2} + \sqrt{\frac{\mu^2\alpha^2}{4} + 4(1 - \frac{\alpha L}{2})}\right).$$

Тогда последовательность $\{x_k\}$, генерируемая итерациями метода тяжёлого шарика, сходится линейно к единственному оптимальному решению $x^\star$. В частности,

$$f(x_k) - f^\star \leq q^k(f(x_0) - f^\star),$$

где $q \in [0, 1)$.

[4] Глобальная сходимость метода тяжёлого шарика для выпуклой оптимизации, Euhanna Ghadimi et al.

Итоги по методу тяжёлого шарика

- Обеспечивает ускоренную сходимость для сильно выпуклых квадратичных задач.

Итоги по методу тяжёлого шарика

- Обеспечивает ускоренную сходимость для сильно выпуклых квадратичных задач.
- Локально ускоренная сходимость была доказана в оригинальной статье.

Итоги по методу тяжёлого шарика

- Обеспечивает ускоренную сходимость для сильно выпуклых квадратичных задач.
- Локально ускоренная сходимость была доказана в оригинальной статье.
- Недавно [5] было доказано, что глобального ускорения сходимости для метода не существует.

[5] Provable non-accelerations of the heavy-ball method

Итоги по методу тяжёлого шарика

- Обеспечивает ускоренную сходимость для сильно выпуклых квадратичных задач.
- Локально ускоренная сходимость была доказана в оригинальной статье.
- Недавно [5] было доказано, что глобального ускорения сходимости для метода не существует.
- Метод не был чрезвычайно популярен до ML-бума.

[5] Provable non-accelerations of the heavy-ball method

Итоги по методу тяжёлого шарика

- Обеспечивает ускоренную сходимость для сильно выпуклых квадратичных задач.
- Локально ускоренная сходимость была доказана в оригинальной статье.
- Недавно [5] было доказано, что глобального ускорения сходимости для метода не существует.
- Метод не был чрезвычайно популярен до ML-бума.
- Сейчас он фактически является стандартом для практического ускорения методов градиентного спуска, в том числе для невыпуклых задач (обучение нейронных сетей).

[5] Provable non-accelerations of the heavy-ball method

Ускоренный градиентный метод Нестерова

Концепция ускоренного градиентного метода Нестерова

$$x_{k+1} = x_k - \alpha \nabla f(x_k)$$

GD

$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1})$$

HB

$$\begin{cases} y_{k+1} = x_k + \beta(x_k - x_{k-1}) \\ x_{k+1} = y_{k+1} - \alpha \nabla f(y_{k+1}) \end{cases}$$

NAG

Концепция ускоренного градиентного метода Нестерова

$$x_{k+1} = x_k - \alpha \nabla f(x_k) \qquad x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1})$$

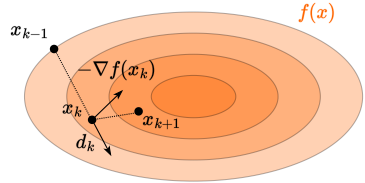
$$\begin{cases} y_{k+1} = x_k + \beta(x_k - x_{k-1}) \\ x_{k+1} = y_{k+1} - \alpha \nabla f(y_{k+1}) \end{cases}$$

Polyak momentum

Давайте определим следующие обозначения

$$x^+ = x - \alpha \nabla f(x) \qquad \text{Градиентный шаг}$$

$$d_k = \beta_k (x_k - x_{k-1}) \qquad \text{Импульс}$$



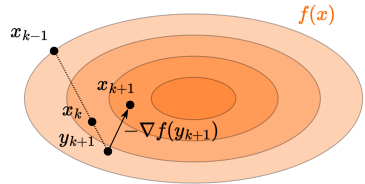
Тогда мы можем записать:

$$x_{k+1} = x_k^+ \qquad \text{Градиентный спуск}$$

$$x_{k+1} = x_k^+ + d_k \qquad \text{Метод тяжёлого шарика}$$

$$x_{k+1} = (x_k + d_k)^+ \qquad \text{Ускоренный градиентный метод Нестерова}$$

Nesterov momentum



Сходимость для выпуклых функций

Theorem

Предположим, что $f : \mathbb{R}^n \to \mathbb{R}$ является выпуклой и L-гладкой. Ускоренный градиентный метод Нестерова (NAG) предназначен для решения задачи минимизации, начиная с начальной точки $x_0 = y_0 \in \mathbb{R}^n$ и $\lambda_0 = 0$. Алгоритм выполняет следующие шаги:

Обновление градиента: $$x_{k+1} = y_k - \frac{1}{L}\nabla f(y_k)$$

Вес экстраполяции: $$\lambda_{k+1} = \frac{1 + \sqrt{1 + 4\lambda_k^2}}{2}$$

$$\gamma_k = \frac{\lambda_k - 1}{\lambda_{k+1}}$$

Экстраполяция: $$y_{k+1} = x_{k+1} + \gamma_k (x_{k+1} - x_k)$$

Последовательность $\{f(x_k)\}_{k\in\mathbb{N}}$, генерируемая алгоритмом, сходится к оптимальному значению f^* со скоростью $\mathcal{O}\left(\frac{1}{k^2}\right)$, в частности:

$$f(x_k) - f^* \leq \frac{2L\|x_0 - x^*\|^2}{k^2}$$

Ускоренная сходимость для сильно выпуклых функций

Theorem

Предположим, что $f : \mathbb{R}^n \to \mathbb{R}$ является μ-сильно выпуклой и L-гладкой. Ускоренный градиентный метод Нестерова (NAG) предназначен для решения задачи минимизации, начиная с начальной точки $x_0 = y_0 \in \mathbb{R}^n$ и $\lambda_0 = 0$. Алгоритм выполняет следующие шаги:

Обновление градиента: $$x_{k+1} = y_k - \frac{1}{L}\nabla f(y_k)$$

Экстраполяция: $$y_{k+1} = x_{k+1} - \gamma\left(x_{k+1} - x_k\right)$$

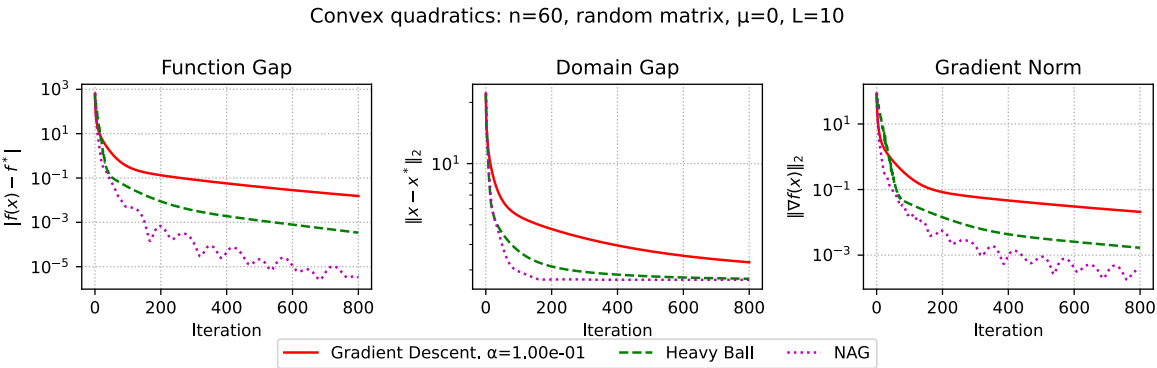
Вес экстраполяции: $$\gamma = \frac{\sqrt{L} - \sqrt{\mu}}{\sqrt{L} + \sqrt{\mu}}$$

Последовательность $\{f(x_k)\}_{k\in\mathbb{N}}$, генерируемая алгоритмом, сходится к оптимальному значению f^* линейно:

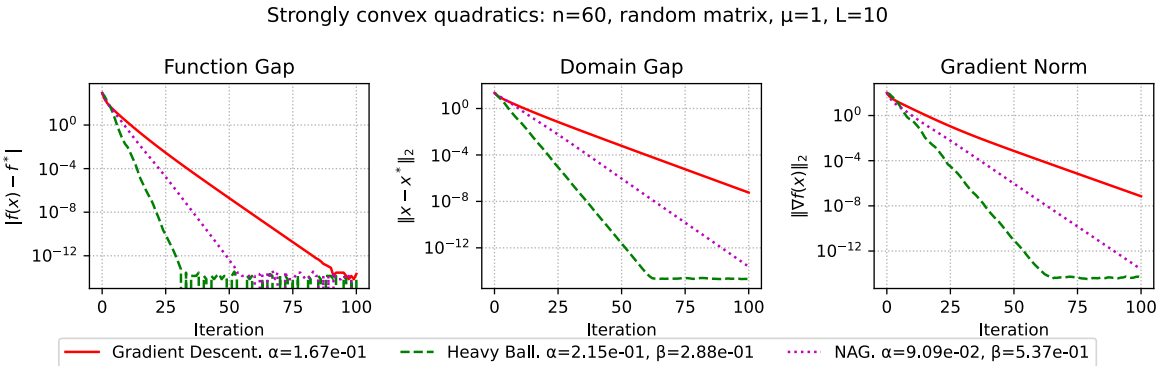
$$f(x_k) - f^* \leq \frac{\mu + L}{2}\|x_0 - x^*\|_2^2 \exp\left(-\frac{k}{\sqrt{\varkappa}}\right)$$

Численные эксперименты

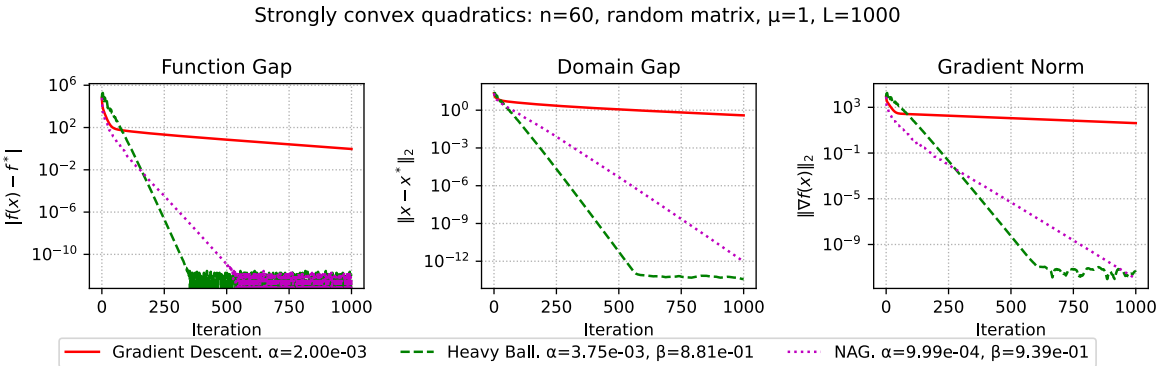
Выпуклая квадратичная задача (линейная регрессия)



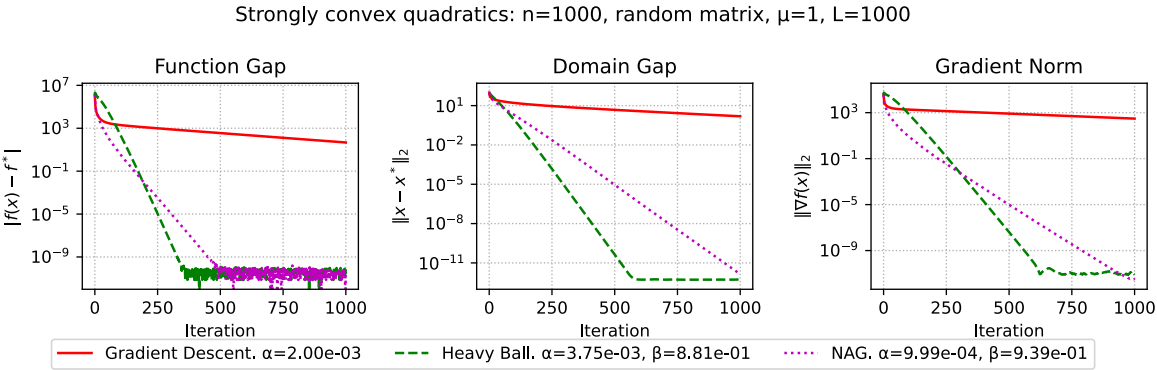
Сильно выпуклая квадратичная задача (регуляризованная линейная регрессия)



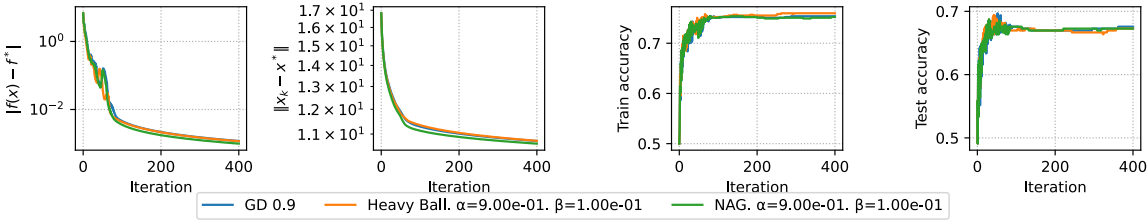
Сильно выпуклая квадратичная задача (регуляризованная линейная регрессия)



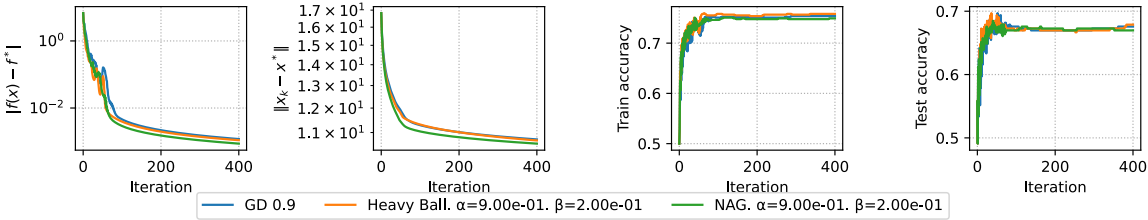
Сильно выпуклая квадратичная задача (регуляризованная линейная регрессия)



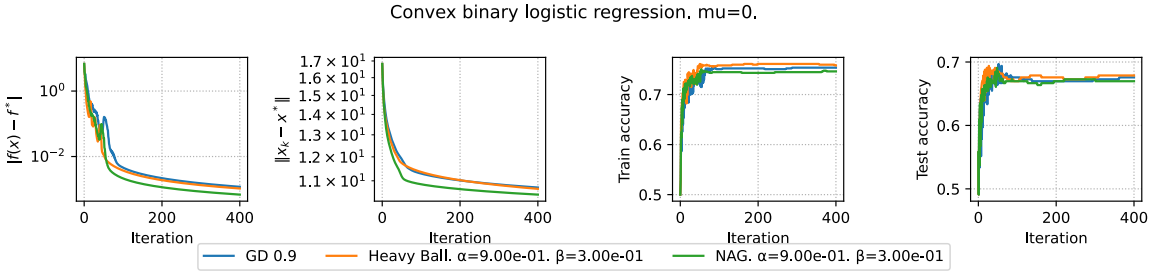
Выпуклая бинарная логистическая регрессия



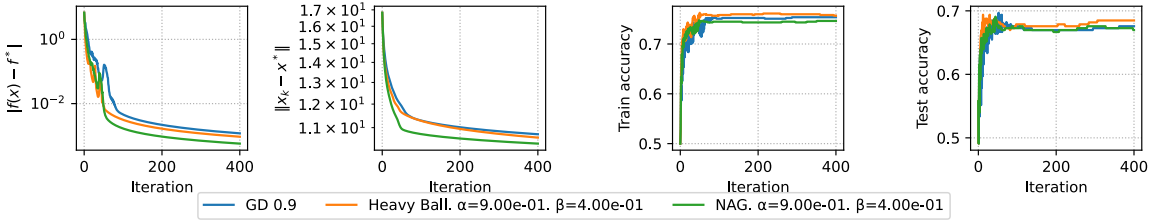
Выпуклая бинарная логистическая регрессия



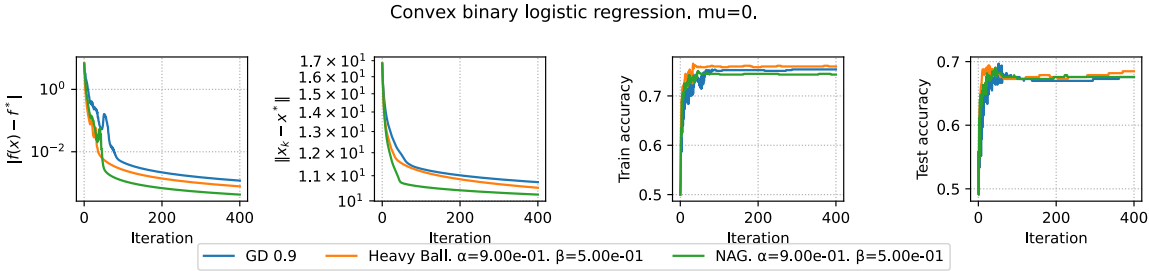
Выпуклая бинарная логистическая регрессия



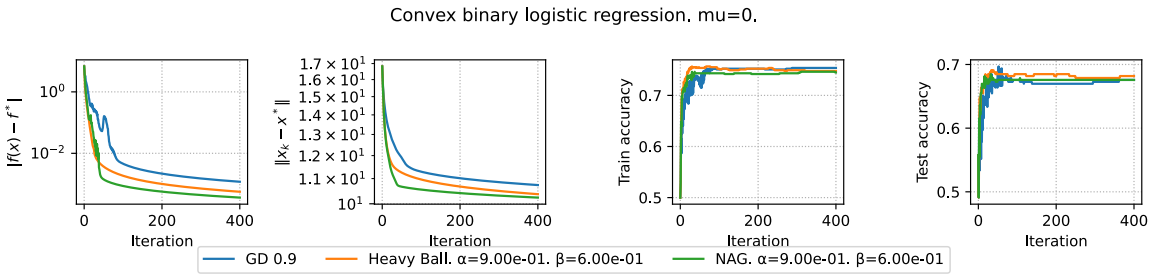
Выпуклая бинарная логистическая регрессия



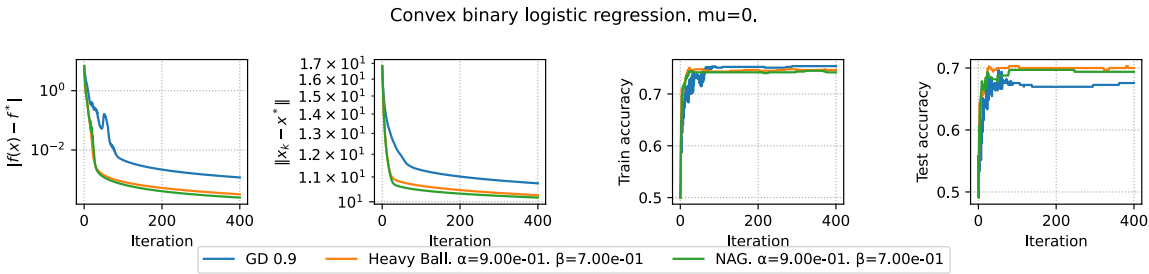
Выпуклая бинарная логистическая регрессия



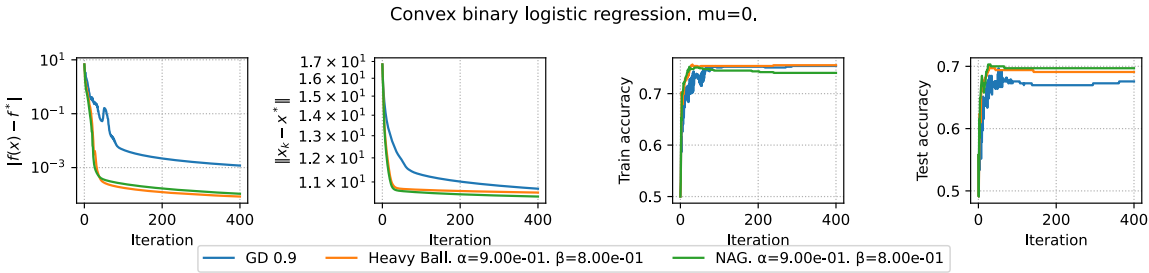
Выпуклая бинарная логистическая регрессия



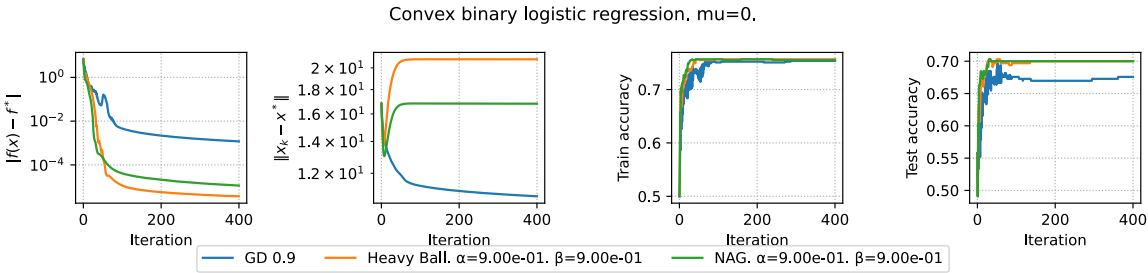
Выпуклая бинарная логистическая регрессия



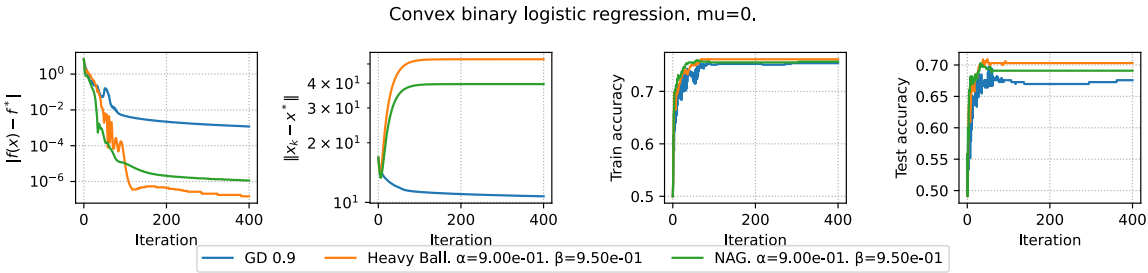
Выпуклая бинарная логистическая регрессия



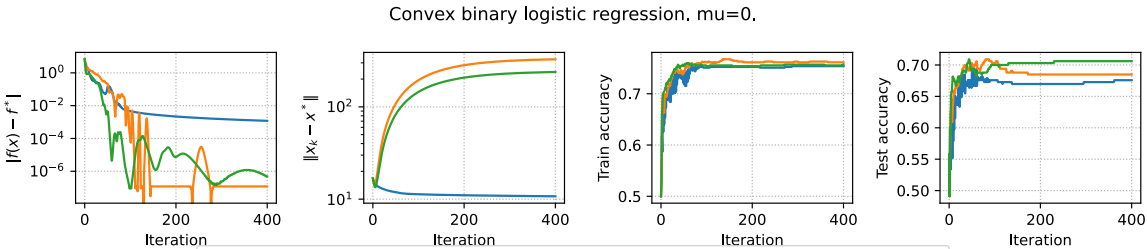
Выпуклая бинарная логистическая регрессия



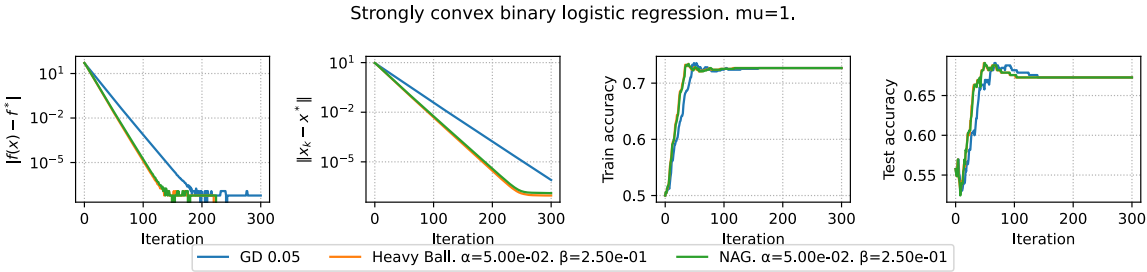
Выпуклая бинарная логистическая регрессия



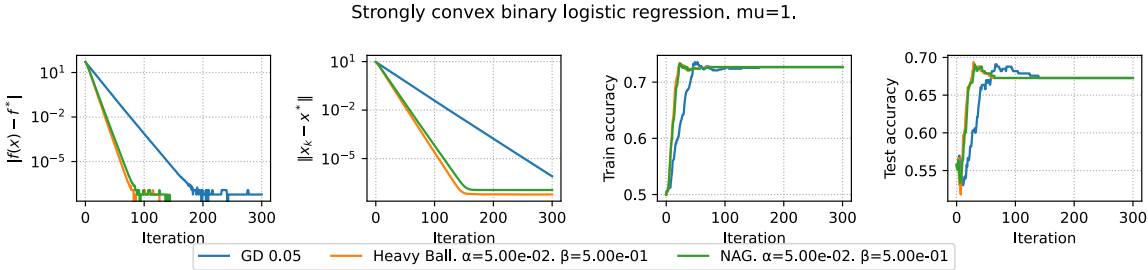
Выпуклая бинарная логистическая регрессия



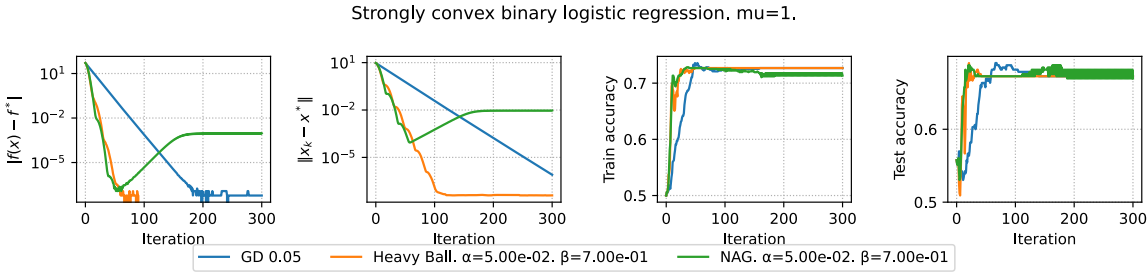
Сильно выпуклая бинарная логистическая регрессия



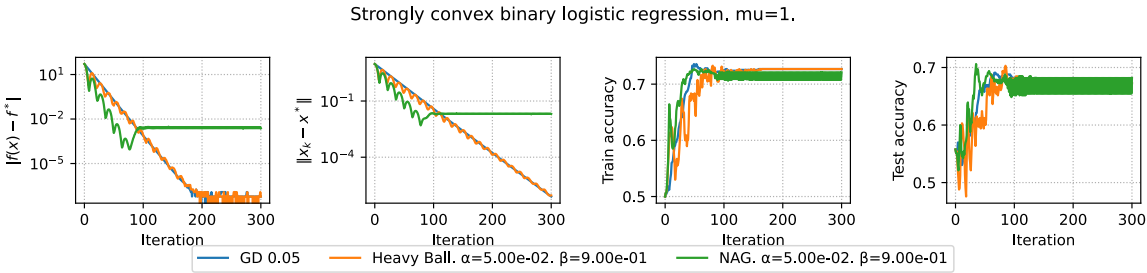
Сильно выпуклая бинарная логистическая регрессия

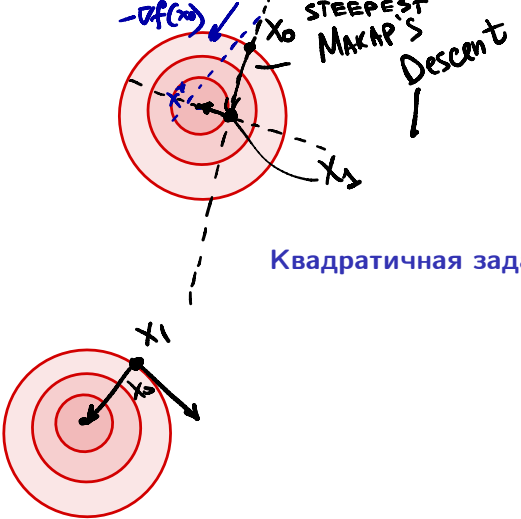


Сильно выпуклая бинарная логистическая регрессия



Сильно выпуклая бинарная логистическая регрессия





Квадратичная задача оптимизации

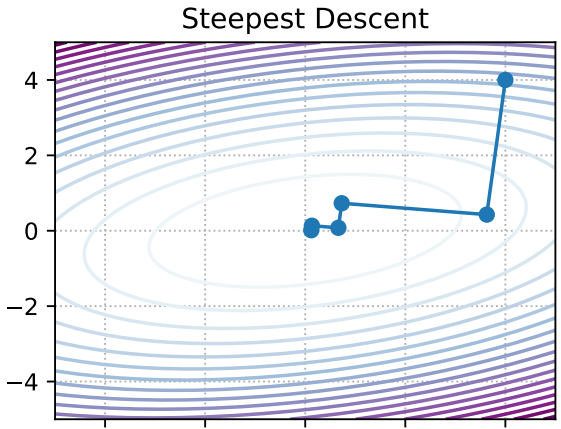
Сильно выпуклая квадратичная функция

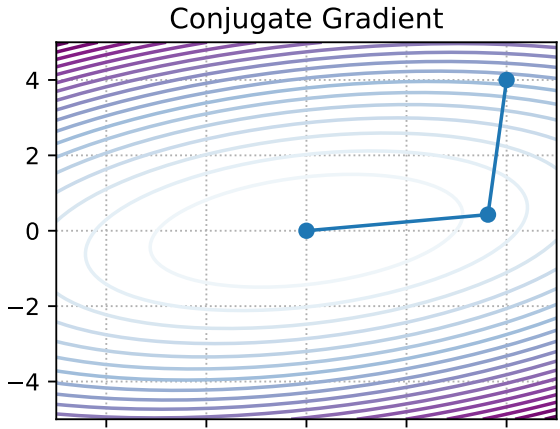
Рассмотрим следующую квадратичную задачу оптимизации:

$$\min_{x \in \mathbb{R}^n} f(x) = \min_{x \in \mathbb{R}^n} \frac{1}{2} x^\top A x - b^\top x + c, \text{ где } A \in \mathbb{S}^n_{++}. \quad (1)$$

Условия оптимальности

$$Ax^* = b$$





Наискорейший спуск aka точный линейный поиск

$$\alpha_k = \arg\min_{\alpha \in \mathbb{R}^+} f(x_{k+1}) = \arg\min_{\alpha \in \mathbb{R}^+} f(x_k - \alpha \nabla f(x_k))$$

Более теоретический, чем практический подход к выбору шага. Он также позволяет анализировать сходимость, но точный линейный поиск может быть численно сложным, если вычисление функции занимает слишком много времени или требует слишком много ресурсов.

Интересное теоретическое свойство этого метода заключается в том, что каждая следующая итерация метода ортогональна предыдущей:

$$\alpha_k = \arg\min_{\alpha \in \mathbb{R}^+} f(x_k - \alpha \nabla f(x_k))$$

Наискорейший спуск aka точный линейный поиск

$$\alpha_k = \arg\min_{\alpha \in \mathbb{R}^+} f(x_{k+1}) = \arg\min_{\alpha \in \mathbb{R}^+} f(x_k - \alpha \nabla f(x_k))$$

Более теоретический, чем практический подход к выбору шага. Он также позволяет анализировать сходимость, но точный линейный поиск может быть численно сложным, если вычисление функции занимает слишком много времени или требует слишком много ресурсов.

Интересное теоретическое свойство этого метода заключается в том, что каждая следующая итерация метода ортогональна предыдущей:

$$\alpha_k = \arg\min_{\alpha \in \mathbb{R}^+} f(x_k - \alpha \nabla f(x_k))$$

Условия оптимальности:

Наискорейший спуск aka точный линейный поиск

$$\alpha_k = \arg\min_{\alpha \in \mathbb{R}^+} f(x_{k+1}) = \arg\min_{\alpha \in \mathbb{R}^+} f(x_k - \alpha \nabla f(x_k))$$

Более теоретический, чем практический подход к выбору шага. Он также позволяет анализировать сходимость, но точный линейный поиск может быть численно сложным, если вычисление функции занимает слишком много времени или требует слишком много ресурсов.

Интересное теоретическое свойство этого метода заключается в том, что каждая следующая итерация метода ортогональна предыдущей:

$$\alpha_k = \arg\min_{\alpha \in \mathbb{R}^+} f(x_k - \alpha \nabla f(x_k))$$

Условия оптимальности:

$$\nabla f(x_k)^T \nabla f(x_{k+1}) = 0$$

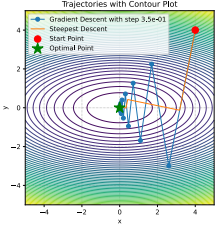


Рис. 1: Наискорейший спуск

Открыть в Colab ♣

🔥 Оптимальное значение для квадратичных функций

$$\nabla f(x_k)^\top A(x_k - \alpha \nabla f(x_k)) - \nabla f(x_k)^\top b = 0 \qquad \alpha_k = \frac{\nabla f(x_k)^T \nabla f(x_k)}{\nabla f(x_k)^T A \nabla f(x_k)}$$

Ортогональность

Сопряженные направления. A-ортогональность.

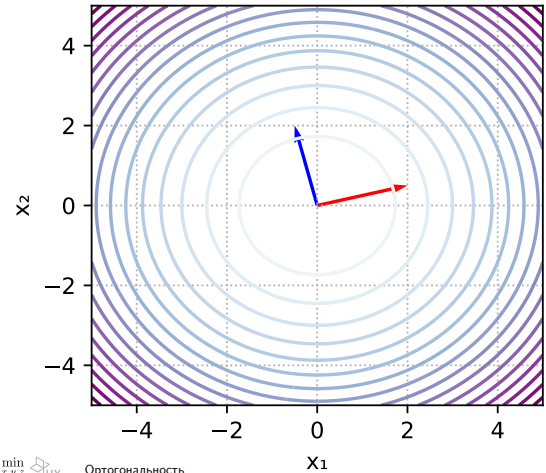
$\tilde{A} = I$

$f(x) = \frac{1}{2} x^T \tilde{A} x$

v_1 and v_2 are orthogonal

$v_1^T v_2 = 0.00$

$v_1^T A v_2 = 1.19$

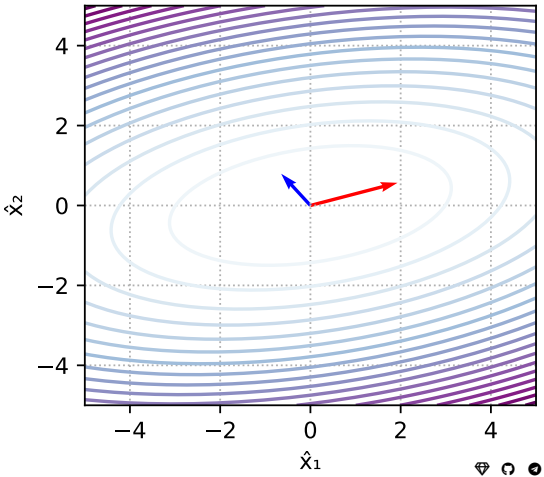


$\hat{v}_1$ and $\hat{v}_2$ are A-orthogonal

$\hat{v}_1^T \hat{v}_2 = -0.80$

$\hat{v}_1^T A \hat{v}_2 = -0.00$

$\tilde{A} = A \neq I$



Сопряженные направления. A-ортогональность.

Предположим, у нас есть две системы координат и квадратичная функция $f(x) = \frac{1}{2}x^T Ix$ выглядит так, как на левой части изображения 2, в то время как в других координатах она выглядит как $f(\hat{x}) = \frac{1}{2}\hat{x}^T A\hat{x}$, где $A \in \mathbb{S}^n_{++}$.

$$\frac{1}{2}x^T Ix \qquad\qquad \frac{1}{2}\hat{x}^T A\hat{x}$$

Поскольку $A = Q\Lambda Q^T$:

$$\frac{1}{2}\hat{x}^T A\hat{x}$$

Сопряженные направления. A-ортогональность.

Предположим, у нас есть две системы координат и квадратичная функция $f(x) = \frac{1}{2}x^T Ix$ выглядит так, как на левой части изображения 2, в то время как в других координатах она выглядит как $f(\hat{x}) = \frac{1}{2}\hat{x}^T A\hat{x}$, где $A \in \mathbb{S}^n_{++}$.

$$\frac{1}{2}x^T Ix \qquad\qquad \frac{1}{2}\hat{x}^T A\hat{x}$$

Поскольку $A = Q\Lambda Q^T$:

$$\frac{1}{2}\hat{x}^T A\hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda Q^T\hat{x}$$

Сопряженные направления. A-ортогональность.

Предположим, у нас есть две системы координат и квадратичная функция $f(x) = \frac{1}{2}x^T Ix$ выглядит так, как на левой части изображения 2, в то время как в других координатах она выглядит как $f(\hat{x}) = \frac{1}{2}\hat{x}^T A\hat{x}$, где $A \in \mathbb{S}^n_{++}$.

$$\frac{1}{2}x^T Ix \qquad\qquad \frac{1}{2}\hat{x}^T A\hat{x}$$

Поскольку $A = Q\Lambda Q^T$:

$$\frac{1}{2}\hat{x}^T A\hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda Q^T\hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda^{\frac{1}{2}}\Lambda^{\frac{1}{2}}Q^T\hat{x}$$

Сопряженные направления. A-ортогональность.

Предположим, у нас есть две системы координат и квадратичная функция $f(x) = \frac{1}{2}x^T Ix$ выглядит так, как на левой части изображения 2, в то время как в других координатах она выглядит как $f(\hat{x}) = \frac{1}{2}\hat{x}^T A\hat{x}$, где $A \in \mathbb{S}^n_{++}$.

$$\frac{1}{2}x^T Ix \qquad\qquad \frac{1}{2}\hat{x}^T A\hat{x}$$

Поскольку $A = Q\Lambda Q^T$:

$$\frac{1}{2}\hat{x}^T A\hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda Q^T\hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda^{\frac{1}{2}}\Lambda^{\frac{1}{2}}Q^T\hat{x} = \frac{1}{2}x^T Ix$$

Сопряженные направления. A-ортогональность.

Предположим, у нас есть две системы координат и квадратичная функция $f(x) = \frac{1}{2}x^T Ix$ выглядит так, как на левой части изображения 2, в то время как в других координатах она выглядит как $f(\hat{x}) = \frac{1}{2}\hat{x}^T A\hat{x}$, где $A \in \mathbb{S}^n_{++}$.

$$\frac{1}{2}x^T Ix \qquad\qquad \frac{1}{2}\hat{x}^T A\hat{x}$$

Поскольку $A = Q\Lambda Q^T$:

$$\frac{1}{2}\hat{x}^T A\hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda Q^T\hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda^{\frac{1}{2}}\Lambda^{\frac{1}{2}}Q^T\hat{x} = \frac{1}{2}x^T Ix \text{ и } \hat{x} = Q\Lambda^{-\frac{1}{2}}x$$

Сопряженные направления. A-ортогональность.

Предположим, у нас есть две системы координат и квадратичная функция $f(x) = \frac{1}{2}x^T Ix$ выглядит так, как на левой части изображения 2, в то время как в других координатах она выглядит как $f(\hat{x}) = \frac{1}{2}\hat{x}^T A\hat{x}$, где $A \in \mathbb{S}^n_{++}$.

$$\frac{1}{2}x^T Ix \qquad\qquad \frac{1}{2}\hat{x}^T A\hat{x}$$

Поскольку $A = Q\Lambda Q^T$:

$$\frac{1}{2}\hat{x}^T A\hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda Q^T\hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda^{\frac{1}{2}}\Lambda^{\frac{1}{2}}Q^T\hat{x} = \frac{1}{2}x^T Ix \text{ и } \hat{x} = Q\Lambda^{-\frac{1}{2}}x$$

Сопряженные направления. A-ортогональность.

Предположим, у нас есть две системы координат и квадратичная функция $f(x) = \frac{1}{2}x^T Ix$ выглядит так, как на левой части изображения 2, в то время как в других координатах она выглядит как $f(\hat{x}) = \frac{1}{2}\hat{x}^T A\hat{x}$, где $A \in \mathbb{S}^n_{++}$.

$$\frac{1}{2}x^T Ix \qquad\qquad \frac{1}{2}\hat{x}^T A\hat{x}$$

Поскольку $A = Q\Lambda Q^T$:

$$\frac{1}{2}\hat{x}^T A\hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda Q^T\hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda^{\frac{1}{2}}\Lambda^{\frac{1}{2}}Q^T\hat{x} = \frac{1}{2}x^T Ix \text{ и } \hat{x} = Q\Lambda^{-\frac{1}{2}}x$$

A-ортогональные векторы

Векторы $x \in \mathbb{R}^n$ и $y \in \mathbb{R}^n$ называются A-ортогональными (или A-сопряженными), если

$$x^T Ay = 0 \qquad \Leftrightarrow \qquad x \perp_A y$$

Когда $A = I$, A-ортогональность превращается в ортогональность.

Процесс Грама-Шмидта

Вход: n линейно независимых векторов $u_0, \dots, u_{n-1}$.

Выход: n линейно независимых попарно ортогональных векторов $d_0, \dots, d_{n-1}$.

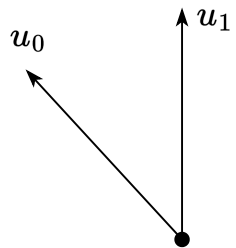


Рис. 3: Иллюстрация процесса Грама-Шмидта

Процесс Грама-Шмидта

Вход: n линейно независимых векторов $u_0, \dots, u_{n-1}$.

Выход: n линейно независимых попарно ортогональных векторов $d_0, \dots, d_{n-1}$.

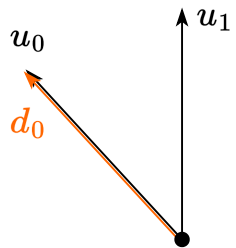


Рис. 4: Иллюстрация процесса Грама-Шмидта

Процесс Грама-Шмидта

Вход: n линейно независимых векторов $u_0, \dots, u_{n-1}$.

Выход: n линейно независимых попарно ортогональных векторов $d_0, \dots, d_{n-1}$.

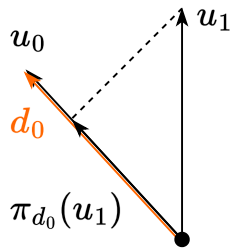


Рис. 5: Иллюстрация процесса Грама-Шмидта

Процесс Грама-Шмидта

Вход: n линейно независимых векторов $u_0, \dots, u_{n-1}$.

Выход: n линейно независимых попарно ортогональных векторов $d_0, \dots, d_{n-1}$.

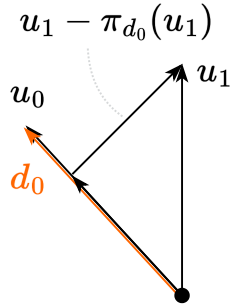


Рис. 6: Иллюстрация процесса Грама-Шмидта

Процесс Грама-Шмидта

Вход: n линейно независимых векторов $u_0, \ldots, u_{n-1}$.

Выход: n линейно независимых попарно ортогональных векторов $d_0, \ldots, d_{n-1}$.

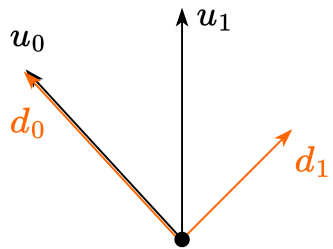
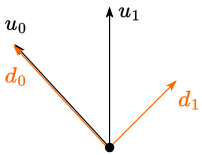
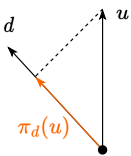


Рис. 7: Иллюстрация процесса Грама-Шмидта

Процесс Грама-Шмидта

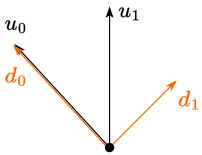
Вход: n линейно независимых векторов $u_0, \dots, u_{n-1}$.

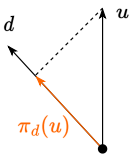




$$\pi_d(u) = \frac{\langle d, u \rangle}{\|d\|_2^2} d$$

Процесс Грама-Шмидта





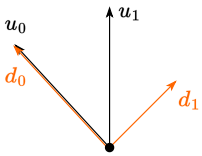
$$\pi_d(u) = \frac{\langle d, u \rangle}{\|d\|_2^2} d$$

Вход: n линейно независимых векторов $u_0, \dots, u_{n-1}$.

Выход: n линейно независимых попарно ортогональных векторов $d_0, \dots, d_{n-1}$.

$$d_0 = u_0$$

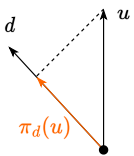
Процесс Грама-Шмидта



Вход: n линейно независимых векторов $u_0, \dots, u_{n-1}$.

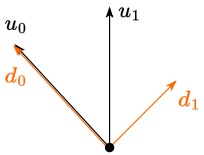
Выход: n линейно независимых попарно ортогональных векторов $d_0, \dots, d_{n-1}$.

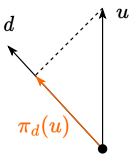
$$d_0 = u_0$$
$$d_1 = u_1 - \pi_{d_0}(u_1)$$



$$\pi_d(u) = \frac{\langle d, u \rangle}{\|d\|_2^2} d$$

Процесс Грама-Шмидта





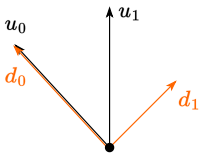
$$\pi_d(u) = \frac{\langle d, u \rangle}{\|d\|_2^2} d$$

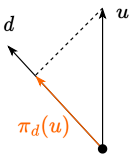
Вход: n линейно независимых векторов $u_0, \dots, u_{n-1}$.

Выход: n линейно независимых попарно ортогональных векторов $d_0, \dots, d_{n-1}$.

$$d_0 = u_0$$
$$d_1 = u_1 - \pi_{d_0}(u_1)$$
$$d_2 = u_2 - \pi_{d_0}(u_2) - \pi_{d_1}(u_2)$$

Процесс Грама-Шмидта





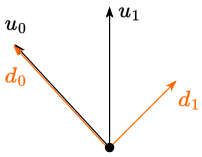
$$\pi_d(u) = \frac{\langle d, u \rangle}{\|d\|_2^2} d$$

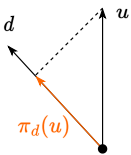
Вход: n линейно независимых векторов $u_0, \dots, u_{n-1}$.

Выход: n линейно независимых попарно ортогональных векторов $d_0, \dots, d_{n-1}$.

$$\begin{aligned} d_0 &= u_0 \\ d_1 &= u_1 - \pi_{d_0}(u_1) \\ d_2 &= u_2 - \pi_{d_0}(u_2) - \pi_{d_1}(u_2) \\ &\vdots \end{aligned}$$

Процесс Грама-Шмидта





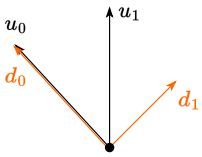
$$\pi_d(u) = \frac{\langle d, u \rangle}{\|d\|_2^2} d$$

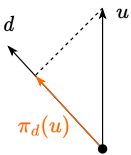
Вход: n линейно независимых векторов $u_0, \dots, u_{n-1}$.

Выход: n линейно независимых попарно ортогональных векторов $d_0, \dots, d_{n-1}$.

$$\begin{aligned}
d_0 &= u_0 \\
d_1 &= u_1 - \pi_{d_0}(u_1) \\
d_2 &= u_2 - \pi_{d_0}(u_2) - \pi_{d_1}(u_2) \\
&\vdots \\
d_k &= u_k - \sum_{i=0}^{k-1} \pi_{d_i}(u_k)
\end{aligned}$$

Процесс Грама-Шмидта





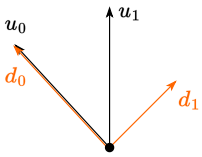
$$\pi_d(u) = \frac{\langle d, u \rangle}{\|d\|_2^2} d$$

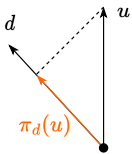
Вход: n линейно независимых векторов $u_0, \dots, u_{n-1}$.

Выход: n линейно независимых попарно ортогональных векторов $d_0, \dots, d_{n-1}$.

$$\begin{aligned}
d_0 &= u_0 \\
d_1 &= u_1 - \pi_{d_0}(u_1) \\
d_2 &= u_2 - \pi_{d_0}(u_2) - \pi_{d_1}(u_2) \\
&\vdots \\
d_k &= u_k - \sum_{i=0}^{k-1} \pi_{d_i}(u_k)
\end{aligned}$$

Процесс Грама-Шмидта





$$\pi_d(u) = \frac{\langle d, u \rangle}{\|d\|_2^2} d$$

Вход: n линейно независимых векторов $u_0, \dots, u_{n-1}$.

Выход: n линейно независимых попарно ортогональных векторов $d_0, \dots, d_{n-1}$.

$$\begin{aligned} d_0 &= u_0 \\ d_1 &= u_1 - \pi_{d_0}(u_1) \\ d_2 &= u_2 - \pi_{d_0}(u_2) - \pi_{d_1}(u_2) \\ &\vdots \\ d_k &= u_k - \sum_{i=0}^{k-1} \pi_{d_i}(u_k) \end{aligned}$$

$$d_k = u_k + \sum_{i=0}^{k-1} \beta_{ik} d_i \qquad \beta_{ik} = -\frac{\langle d_i, u_k \rangle}{\langle d_i, d_i \rangle} \tag{2}$$

Метод сопряженных направлений (CD)

Общая идея

- В изотропном случае $A = I$ метод наискорейшего спуска, запущенный из произвольной точки в n ортогональных линейно независимых направлениях, сойдется за n шагов в точных арифметических вычислениях. Мы пытаемся построить аналогичную процедуру в случае $A \neq I$ с использованием концепции A-ортогональности.

Общая идея

- В изотропном случае $A = I$ метод наискорейшего спуска, запущенный из произвольной точки в n ортогональных линейно независимых направлениях, сойдется за n шагов в точных арифметических вычислениях. Мы пытаемся построить аналогичную процедуру в случае $A \neq I$ с использованием концепции A-ортогональности.
- Предположим, у нас есть набор из n линейно независимых A-ортогональных направлений $d_0, \ldots, d_{n-1}$ (которые будут вычислены с помощью процесса Грама-Шмидта).

Общая идея

- В изотропном случае $A = I$ метод наискорейшего спуска, запущенный из произвольной точки в n ортогональных линейно независимых направлениях, сойдется за n шагов в точных арифметических вычислениях. Мы пытаемся построить аналогичную процедуру в случае $A \neq I$ с использованием концепции A-ортогональности.
- Предположим, у нас есть набор из n линейно независимых A-ортогональных направлений $d_0, \dots, d_{n-1}$ (которые будут вычислены с помощью процесса Грама-Шмидта).
- Мы хотим построить метод, который идет из x_0 в x^* для квадратичной задачи с шагами α_i, который, фактически, является разложением $x^* - x_0$ в некотором базисе:

$$x^* = x_0 + \sum_{i=0}^{n-1} \alpha_i d_i \qquad x^* - x_0 = \sum_{i=0}^{n-1} \alpha_i d_i$$

Общая идея

- В изотропном случае $A = I$ метод наискорейшего спуска, запущенный из произвольной точки в n ортогональных линейно независимых направлениях, сойдется за n шагов в точных арифметических вычислениях. Мы пытаемся построить аналогичную процедуру в случае $A \neq I$ с использованием концепции A-ортогональности.
- Предположим, у нас есть набор из n линейно независимых A-ортогональных направлений $d_0, \dots, d_{n-1}$ (которые будут вычислены с помощью процесса Грама-Шмидта).
- Мы хотим построить метод, который идет из x_0 в x^* для квадратичной задачи с шагами α_i, который, фактически, является разложением $x^* - x_0$ в некотором базисе:

$$x^* = x_0 + \sum_{i=0}^{n-1} \alpha_i d_i \qquad x^* - x_0 = \sum_{i=0}^{n-1} \alpha_i d_i$$

- Мы докажем, что α_i и d_i могут быть построены очень эффективно с вычислительной точки зрения (метод сопряженных градиентов).

Идея метода сопряженных направлений (CD)

Таким образом, мы формулируем алгоритм:

Предположим, что нам заранее известны линейно-независимые векторы $u_0, \dots, u_{n-1}$.

1. $d_0 = -\nabla f(x_0)$.

Идея метода сопряженных направлений (CD)

Таким образом, мы формулируем алгоритм:

Предположим, что нам заранее известны линейно-независимые векторы $u_0, \dots, u_{n-1}$.

1. $d_0 = -\nabla f(x_0)$.
2. С помощью процедуры точного линейного поиска находим оптимальную длину шага. Вычисляем α минимизируя $f(x_k + \alpha_k d_k)$ по формуле

$$\alpha_k = -\frac{d_k^\top (Ax_k - b)}{d_k^\top A d_k} \tag{3}$$

Идея метода сопряженных направлений (CD)

Таким образом, мы формулируем алгоритм:

Предположим, что нам заранее известны линейно-независимые векторы $u_0, \dots, u_{n-1}$.

1. $d_0 = -\nabla f(x_0)$.
2. С помощью процедуры точного линейного поиска находим оптимальную длину шага. Вычисляем α минимизируя $f(x_k + \alpha_k d_k)$ по формуле

$$\alpha_k = -\frac{d_k^\top (Ax_k - b)}{d_k^\top A d_k} \tag{3}$$

3. Выполняем шаг алгоритма:

$$x_{k+1} = x_k + \alpha_k d_k$$

Идея метода сопряженных направлений (CD)

Таким образом, мы формулируем алгоритм:

Предположим, что нам заранее известны линейно-независимые векторы $u_0, \dots, u_{n-1}$.

1. $d_0 = -\nabla f(x_0)$.
2. С помощью процедуры точного линейного поиска находим оптимальную длину шага. Вычисляем α минимизируя $f(x_k + \alpha_k d_k)$ по формуле
$$\alpha_k = -\frac{d_k^\top (Ax_k - b)}{d_k^\top A d_k} \tag{3}$$
3. Выполняем шаг алгоритма:
$$x_{k+1} = x_k + \alpha_k d_k$$
4. Обновляем направление: d_{k+1} получаем из u_{k+1} с помощью модифицированной процедуры Грама–Шмидта в скалярном произведении $\langle v, w \rangle_A = v^\top A w$ относительно уже построенных $d_0, \dots, d_k$:
$$d_{k+1} = u_{k+1} - \sum_{i=0}^{k} \beta_{k+1,i}\, d_i, \quad \beta_{k+1,i} = \frac{u_{k+1}^\top A d_i}{d_i^\top A d_i}$$
что обеспечивает $d_{k+1} \perp_A d_j$ для всех $j \le k$.

Идея метода сопряженных направлений (CD)

Таким образом, мы формулируем алгоритм:

Предположим, что нам заранее известны линейно-независимые векторы $u_0, \dots, u_{n-1}$.

1. $d_0 = -\nabla f(x_0)$.
2. С помощью процедуры точного линейного поиска находим оптимальную длину шага. Вычисляем α минимизируя $f(x_k + \alpha_k d_k)$ по формуле
$$\alpha_k = -\frac{d_k^\top (Ax_k - b)}{d_k^\top A d_k} \tag{3}$$
3. Выполняем шаг алгоритма:
$$x_{k+1} = x_k + \alpha_k d_k$$
4. Обновляем направление: d_{k+1} получаем из u_{k+1} с помощью модифицированной процедуры Грама–Шмидта в скалярном произведении $\langle v, w \rangle_A = v^\top A w$ относительно уже построенных $d_0, \dots, d_k$:
$$d_{k+1} = u_{k+1} - \sum_{i=0}^{k} \beta_{k+1,i}\, d_i, \quad \beta_{k+1,i} = \frac{u_{k+1}^\top A d_i}{d_i^\top A d_i}$$
что обеспечивает $d_{k+1} \perp_A d_j$ для всех $j \le k$.
5. Повторяем шаги 2–4, пока не построим n направлений, где n — размерность пространства (x).

Метод сопряженных направлений (CD)

Лемма 1. Линейная независимость A-ортогональных векторов.

Если множество векторов $d_1, \dots, d_n$ - попарно A-ортогональны (каждая пара векторов A-ортогональна), то эти векторы линейно независимы. $A \in \mathbb{S}^n_{++}$.

Метод сопряженных направлений (CD)

Лемма 1. Линейная независимость A-ортогональных векторов.

Если множество векторов $d_1, \dots, d_n$ - попарно A-ортогональны (каждая пара векторов A-ортогональна), то эти векторы линейно независимы. $A \in \mathbb{S}^n_{++}$.

Доказательство

Покажем, что если $\sum_{i=1}^{n} \alpha_i d_i = 0$, то все коэффициенты должны быть равны нулю:

Метод сопряженных направлений (CD)

Лемма 1. Линейная независимость A-ортогональных векторов.

Если множество векторов $d_1, \dots, d_n$ - попарно A-ортогональны (каждая пара векторов A-ортогональна), то эти векторы линейно независимы. $A \in \mathbb{S}^n_{++}$.

Доказательство

Покажем, что если $\sum_{i=1}^{n} \alpha_i d_i = 0$, то все коэффициенты должны быть равны нулю:

$$0 = \sum_{i=1}^{n} \alpha_i d_i$$

Метод сопряженных направлений (CD)

Лемма 1. Линейная независимость A-ортогональных векторов.

Если множество векторов $d_1, \dots, d_n$ - попарно A-ортогональны (каждая пара векторов A-ортогональна), то эти векторы линейно независимы. $A \in \mathbb{S}^n_{++}$.

Доказательство

Покажем, что если $\sum_{i=1}^{n} \alpha_i d_i = 0$, то все коэффициенты должны быть равны нулю:

$$
\begin{aligned}
0 &= \sum_{i=1}^{n} \alpha_i d_i \\
\text{(Умножаем на } d_j^T A\text{)} \qquad &= d_j^\top A \left(\sum_{i=1}^{n} \alpha_i d_i \right)
\end{aligned}
$$

Метод сопряженных направлений (CD)

Лемма 1. Линейная независимость A-ортогональных векторов.

Если множество векторов $d_1, \dots, d_n$ - попарно A-ортогональны (каждая пара векторов A-ортогональна), то эти векторы линейно независимы. $A \in \mathbb{S}^n_{++}$.

Доказательство

Покажем, что если $\sum_{i=1}^{n} \alpha_i d_i = 0$, то все коэффициенты должны быть равны нулю:

$$0 = \sum_{i=1}^{n} \alpha_i d_i$$

(Умножаем на $d_j^T A$)
$$= d_j^\top A \left(\sum_{i=1}^{n} \alpha_i d_i \right) = \sum_{i=1}^{n} \alpha_i d_j^\top A d_i$$

Метод сопряженных направлений (CD)

Лемма 1. Линейная независимость A-ортогональных векторов.

Если множество векторов $d_1, \dots, d_n$ - попарно A-ортогональны (каждая пара векторов A-ортогональна), то эти векторы линейно независимы. $A \in \mathbb{S}^n_{++}$.

Доказательство

Покажем, что если $\sum_{i=1}^{n} \alpha_i d_i = 0$, то все коэффициенты должны быть равны нулю:

$$
\begin{aligned}
0 &= \sum_{i=1}^{n} \alpha_i d_i \\
\text{(Умножаем на } d_j^T A\text{)} \qquad &= d_j^\top A \left(\sum_{i=1}^{n} \alpha_i d_i \right) = \sum_{i=1}^{n} \alpha_i d_j^\top A d_i \\
&= \alpha_j d_j^\top A d_j + 0 + \dots + 0
\end{aligned}
$$

Метод сопряженных направлений (CD)

Лемма 1. Линейная независимость A-ортогональных векторов.

Если множество векторов $d_1, \dots, d_n$ - попарно A-ортогональны (каждая пара векторов A-ортогональна), то эти векторы линейно независимы. $A \in \mathbb{S}^n_{++}$.

Доказательство

Покажем, что если $\sum_{i=1}^{n} \alpha_i d_i = 0$, то все коэффициенты должны быть равны нулю:

$$
\begin{aligned}
0 &= \sum_{i=1}^{n} \alpha_i d_i \\
\text{(Умножаем на } d_j^T A\text{)} \qquad &= d_j^\top A \left(\sum_{i=1}^{n} \alpha_i d_i \right) = \sum_{i=1}^{n} \alpha_i d_j^\top A d_i \\
&= \alpha_j d_j^\top A d_j + 0 + \dots + 0
\end{aligned}
$$

Метод сопряженных направлений (CD)

Лемма 1. Линейная независимость A-ортогональных векторов.

Если множество векторов $d_1, \dots, d_n$ - попарно A-ортогональны (каждая пара векторов A-ортогональна), то эти векторы линейно независимы. $A \in \mathbb{S}^n_{++}$.

Доказательство

Покажем, что если $\sum_{i=1}^{n} \alpha_i d_i = 0$, то все коэффициенты должны быть равны нулю:

$$
\begin{aligned}
0 &= \sum_{i=1}^{n} \alpha_i d_i \\
\text{(Умножаем на } d_j^T A\text{)} \qquad &= d_j^\top A \left(\sum_{i=1}^{n} \alpha_i d_i \right) = \sum_{i=1}^{n} \alpha_i d_j^\top A d_i \\
&= \alpha_j d_j^\top A d_j + 0 + \dots + 0
\end{aligned}
$$

Таким образом, $\alpha_j = 0$, для всех остальных индексов нужно проделать тот же процесс

Доказательство сходимости

Введем следующие обозначения:

- $r_k = b - Ax_k$ - невязка

Доказательство сходимости

Введем следующие обозначения:

- $r_k = b - Ax_k$ - невязка
- $e_k = x_k - x^*$ - ошибка

Доказательство сходимости

Введем следующие обозначения:

- $r_k = b - Ax_k$ - невязка
- $e_k = x_k - x^*$ - ошибка
- Поскольку $Ax^* = b$, имеем $r_k = b - Ax_k = Ax^* - Ax_k = -A(x_k - x^*)$

$$r_k = -Ae_k. \tag{4}$$

Доказательство сходимости

Введем следующие обозначения:

- $r_k = b - Ax_k$ - невязка
- $e_k = x_k - x^*$ - ошибка
- Поскольку $Ax^* = b$, имеем $r_k = b - Ax_k = Ax^* - Ax_k = -A(x_k - x^*)$

$$r_k = -Ae_k. \tag{4}$$

- Также заметим, что поскольку $x_{k+1} = x_0 + \sum\limits_{i=0}^{k} \alpha_i d_i$, имеем

$$e_{k+1} = e_0 + \sum_{i=0}^{k} \alpha_i d_i. \tag{5}$$

Доказательство сходимости

Лемма 2. Сходимость метода сопряженных направлений.

Предположим, мы решаем n-мерную квадратичную сильно выпуклую задачу оптимизации (1). Метод сопряженных направлений

$$x_{k+1} = x_0 + \sum_{i=0}^{k} \alpha_i d_i$$

с $\alpha_i = \frac{\langle d_i, r_i \rangle}{\langle d_i, A d_i \rangle}$ взятым из точного линейного поиска, сходится за не более n шагов алгоритма.

Доказательство сходимости

Лемма 2. Сходимость метода сопряженных направлений.

Предположим, мы решаем n-мерную квадратичную сильно выпуклую задачу оптимизации (1). Метод сопряженных направлений

$$x_{k+1} = x_0 + \sum_{i=0}^{k} \alpha_i d_i$$

с $\alpha_i = \frac{\langle d_i, r_i \rangle}{\langle d_i, A d_i \rangle}$ взятым из точного линейного поиска, сходится за не более n шагов алгоритма.

Доказательство Пусть

$$e_0 = x_0 - x^* = \sum_{i=0}^{n-1} \delta_i d_i$$

Докажем, что $\delta_i = -\alpha_i$:

Доказательство сходимости

Лемма 2. Сходимость метода сопряженных направлений.

Предположим, мы решаем n-мерную квадратичную сильно выпуклую задачу оптимизации (1). Метод сопряженных направлений

$$x_{k+1} = x_0 + \sum_{i=0}^{k} \alpha_i d_i$$

с $\alpha_i = \frac{\langle d_i, r_i \rangle}{\langle d_i, A d_i \rangle}$ взятым из точного линейного поиска, сходится за не более n шагов алгоритма.

Доказательство Пусть

$$e_0 = x_0 - x^* = \sum_{i=0}^{n-1} \delta_i d_i$$

Умножаем обе части слева на $d_k^T A$:

$$d_k^T A e_0 = \sum_{i=0}^{n-1} \delta_i d_k^T A d_i$$

Докажем, что $\delta_i = -\alpha_i$:

Доказательство сходимости

Лемма 2. Сходимость метода сопряженных направлений.

Предположим, мы решаем n-мерную квадратичную сильно выпуклую задачу оптимизации (1). Метод сопряженных направлений

$$x_{k+1} = x_0 + \sum_{i=0}^{k} \alpha_i d_i$$

с $\alpha_i = \frac{\langle d_i, r_i \rangle}{\langle d_i, A d_i \rangle}$ взятым из точного линейного поиска, сходится за не более n шагов алгоритма.

Доказательство Пусть

$$e_0 = x_0 - x^* = \sum_{i=0}^{n-1} \delta_i d_i$$

Умножаем обе части слева на $d_k^T A$:

$$d_k^T A e_0 = \sum_{i=0}^{n-1} \delta_i d_k^T A d_i = \delta_k d_k^T A d_k$$

Докажем, что $\delta_i = -\alpha_i$:

Доказательство сходимости

Лемма 2. Сходимость метода сопряженных направлений.

Предположим, мы решаем n-мерную квадратичную сильно выпуклую задачу оптимизации (1). Метод сопряженных направлений

$$x_{k+1} = x_0 + \sum_{i=0}^{k} \alpha_i d_i$$

с $\alpha_i = \frac{\langle d_i, r_i \rangle}{\langle d_i, A d_i \rangle}$ взятым из точного линейного поиска, сходится за не более n шагов алгоритма.

Доказательство Пусть

$$e_0 = x_0 - x^* = \sum_{i=0}^{n-1} \delta_i d_i$$

Умножаем обе части слева на $d_k^T A$:

$$d_k^T A e_0 = \sum_{i=0}^{n-1} \delta_i d_k^T A d_i = \delta_k d_k^T A d_k$$

Докажем, что $\delta_i = -\alpha_i$:

$$d_k^T A e_k$$

Доказательство сходимости

Лемма 2. Сходимость метода сопряженных направлений.

Предположим, мы решаем n-мерную квадратичную сильно выпуклую задачу оптимизации (1). Метод сопряженных направлений

$$x_{k+1} = x_0 + \sum_{i=0}^{k} \alpha_i d_i$$

с $\alpha_i = \frac{\langle d_i, r_i \rangle}{\langle d_i, A d_i \rangle}$ взятым из точного линейного поиска, сходится за не более n шагов алгоритма.

Доказательство Пусть

$$e_0 = x_0 - x^* = \sum_{i=0}^{n-1} \delta_i d_i$$

Докажем, что $\delta_i = -\alpha_i$:

Умножаем обе части слева на $d_k^T A$:

$$d_k^T A e_0 = \sum_{i=0}^{n-1} \delta_i d_k^T A d_i = \delta_k d_k^T A d_k$$

$$d_k^T A e_k = d_k^T A \left(e_0 + \sum_{i=0}^{k-1} \alpha_i d_i \right)$$

Доказательство сходимости

Лемма 2. Сходимость метода сопряженных направлений.

Предположим, мы решаем n-мерную квадратичную сильно выпуклую задачу оптимизации (1). Метод сопряженных направлений

$$x_{k+1} = x_0 + \sum_{i=0}^{k} \alpha_i d_i$$

с $\alpha_i = \frac{\langle d_i, r_i \rangle}{\langle d_i, A d_i \rangle}$ взятым из точного линейного поиска, сходится за не более n шагов алгоритма.

Доказательство Пусть

$$e_0 = x_0 - x^* = \sum_{i=0}^{n-1} \delta_i d_i$$

Докажем, что $\delta_i = -\alpha_i$:

Умножаем обе части слева на $d_k^T A$:

$$d_k^T A e_0 = \sum_{i=0}^{n-1} \delta_i d_k^T A d_i = \delta_k d_k^T A d_k$$

$$d_k^T A e_k = d_k^T A \left(e_0 + \sum_{i=0}^{k-1} \alpha_i d_i \right) \overset{\perp_A}{=} \delta_k d_k^T A d_k$$

Доказательство сходимости

Лемма 2. Сходимость метода сопряженных направлений.

Предположим, мы решаем n-мерную квадратичную сильно выпуклую задачу оптимизации (1). Метод сопряженных направлений

$$x_{k+1} = x_0 + \sum_{i=0}^{k} \alpha_i d_i$$

с $\alpha_i = \frac{\langle d_i, r_i \rangle}{\langle d_i, A d_i \rangle}$ взятым из точного линейного поиска, сходится за не более n шагов алгоритма.

Доказательство Пусть

$$e_0 = x_0 - x^* = \sum_{i=0}^{n-1} \delta_i d_i$$

Докажем, что $\delta_i = -\alpha_i$:

Умножаем обе части слева на $d_k^T A$:

$$d_k^T A e_0 = \sum_{i=0}^{n-1} \delta_i d_k^T A d_i = \delta_k d_k^T A d_k$$

$$d_k^T A e_k = d_k^T A \left(e_0 + \sum_{i=0}^{k-1} \alpha_i d_i \right) \overset{\perp_A}{=} \delta_k d_k^T A d_k$$

$$\delta_k = \frac{d_k^T A e_k}{d_k^T A d_k}$$

Доказательство сходимости

Лемма 2. Сходимость метода сопряженных направлений.

Предположим, мы решаем n-мерную квадратичную сильно выпуклую задачу оптимизации (1). Метод сопряженных направлений

$$x_{k+1} = x_0 + \sum_{i=0}^{k} \alpha_i d_i$$

с $\alpha_i = \frac{\langle d_i, r_i \rangle}{\langle d_i, A d_i \rangle}$ взятым из точного линейного поиска, сходится за не более n шагов алгоритма.

Доказательство Пусть

$$e_0 = x_0 - x^* = \sum_{i=0}^{n-1} \delta_i d_i$$

Докажем, что $\delta_i = -\alpha_i$:

Умножаем обе части слева на $d_k^T A$:

$$d_k^T A e_0 = \sum_{i=0}^{n-1} \delta_i d_k^T A d_i = \delta_k d_k^T A d_k$$

$$d_k^T A e_k = d_k^T A \left(e_0 + \sum_{i=0}^{k-1} \alpha_i d_i \right) \overset{\perp_A}{=} \delta_k d_k^T A d_k$$

$$\delta_k = \frac{d_k^T A e_k}{d_k^T A d_k} = -\frac{d_k^T r_k}{d_k^T A d_k}$$

Доказательство сходимости

Лемма 2. Сходимость метода сопряженных направлений.

Предположим, мы решаем n-мерную квадратичную сильно выпуклую задачу оптимизации (1). Метод сопряженных направлений

$$x_{k+1} = x_0 + \sum_{i=0}^{k} \alpha_i d_i$$

с $\alpha_i = \frac{\langle d_i, r_i \rangle}{\langle d_i, Ad_i \rangle}$ взятым из точного линейного поиска, сходится за не более n шагов алгоритма.

Доказательство Пусть

$$e_0 = x_0 - x^* = \sum_{i=0}^{n-1} \delta_i d_i$$

Докажем, что $\delta_i = -\alpha_i$:

Умножаем обе части слева на $d_k^T A$:

$$d_k^T A e_0 = \sum_{i=0}^{n-1} \delta_i d_k^T A d_i = \delta_k d_k^T A d_k$$

$$d_k^T A e_k = d_k^T A \left(e_0 + \sum_{i=0}^{k-1} \alpha_i d_i \right) \overset{\perp_A}{=} \delta_k d_k^T A d_k$$

$$\delta_k = \frac{d_k^T A e_k}{d_k^T A d_k} = -\frac{d_k^T r_k}{d_k^T A d_k} \Leftrightarrow \delta_k = -\alpha_k$$

Метод сопряженных градиентов (CG)

Идея метода сопряженных градиентов (CG)

- Это буквально метод сопряженных направлений, в котором мы выбираем специальный набор $d_0, \dots, d_{n-1}$, позволяющий значительно ускорить процесс Грама-Шмидта.

Идея метода сопряженных градиентов (CG)

- Это буквально метод сопряженных направлений, в котором мы выбираем специальный набор $d_0, \ldots, d_{n-1}$, позволяющий значительно ускорить процесс Грама-Шмидта.
- Используется процесс Грама-Шмидта с A-ортогональностью вместо Евклидовой ортогональности, чтобы получить их из набора начальных векторов.

Идея метода сопряженных градиентов (CG)

- Это буквально метод сопряженных направлений, в котором мы выбираем специальный набор $d_0, \dots, d_{n-1}$, позволяющий значительно ускорить процесс Грама-Шмидта.
- Используется процесс Грама-Шмидта с A-ортогональностью вместо Евклидовой ортогональности, чтобы получить их из набора начальных векторов.
- На каждой итерации $r_0, \dots, r_{n-1}$ используются в качестве начальных линейно-независимых векторов для процесса Грама-Шмидта.

Идея метода сопряженных градиентов (CG)

- Это буквально метод сопряженных направлений, в котором мы выбираем специальный набор $d_0, \ldots, d_{n-1}$, позволяющий значительно ускорить процесс Грама-Шмидта.
- Используется процесс Грама-Шмидта с A-ортогональностью вместо Евклидовой ортогональности, чтобы получить их из набора начальных векторов.
- На каждой итерации $r_0, \ldots, r_{n-1}$ используются в качестве начальных линейно-независимых векторов для процесса Грама-Шмидта.
- Основная идея заключается в том, что для произвольного метода CD процесс Грама-Шмидта вычислительно дорогой и требует квадратичного числа операций сложения векторов и скалярных произведений $\mathcal{O}\left(n^2\right)$, в то время как в случае CG мы покажем, что сложность этой процедуры может быть уменьшена до линейной $\mathcal{O}\left(n\right)$.

Идея метода сопряженных градиентов (CG)

- Это буквально метод сопряженных направлений, в котором мы выбираем специальный набор $d_0, \ldots, d_{n-1}$, позволяющий значительно ускорить процесс Грама-Шмидта.
- Используется процесс Грама-Шмидта с A-ортогональностью вместо Евклидовой ортогональности, чтобы получить их из набора начальных векторов.
- На каждой итерации $r_0, \ldots, r_{n-1}$ используются в качестве начальных линейно-независимых векторов для процесса Грама-Шмидта.
- Основная идея заключается в том, что для произвольного метода CD процесс Грама-Шмидта вычислительно дорогой и требует квадратичного числа операций сложения векторов и скалярных произведений $\mathcal{O}\left(n^2\right)$, в то время как в случае CG мы покажем, что сложность этой процедуры может быть уменьшена до линейной $\mathcal{O}\left(n\right)$.

Идея метода сопряженных градиентов (CG)

- Это буквально метод сопряженных направлений, в котором мы выбираем специальный набор $d_0, \ldots, d_{n-1}$, позволяющий значительно ускорить процесс Грама-Шмидта.
- Используется процесс Грама-Шмидта с A-ортогональностью вместо Евклидовой ортогональности, чтобы получить их из набора начальных векторов.
- На каждой итерации $r_0, \ldots, r_{n-1}$ используются в качестве начальных линейно-независимых векторов для процесса Грама-Шмидта.
- Основная идея заключается в том, что для произвольного метода CD процесс Грама-Шмидта вычислительно дорогой и требует квадратичного числа операций сложения векторов и скалярных произведений $\mathcal{O}(n^2)$, в то время как в случае CG мы покажем, что сложность этой процедуры может быть уменьшена до линейной $\mathcal{O}(n)$.

CG = CD + $r_0, \ldots, r_{n-1}$ как начальные векторы для процесса Грама-Шмидта + A-ортогональность.

Леммы для сходимости

Лемма 5. Невязки ортогональны друг другу в методе CG

Все невязки в методе CG ортогональны друг другу:

$$r_i^T r_k = 0 \qquad \forall i \neq k \tag{6}$$

Лемма 7. Коэффициенты для процесса Грама-Шмидта для CG

В процессе Грама-Шмидта для CG

$$\beta_{ji} = \frac{r_i^\top r_i}{r_{i-1}^\top r_{i-1}}, \; i = j + 1.$$

Все остальные коэффициенты равны нулю кроме $i = j$, но этот случай нам неинтересен.

Метод сопряженных градиентов (CG)

$r_0 := b - Ax_0$

если r_0 достаточно мал, то вернуть x_0 как результат

$d_0 := r_0$

$k := 0$

повторять

$$\alpha_k := \frac{r_k^\mathsf{T} r_k}{d_k^\mathsf{T} A d_k}$$

$x_{k+1} := x_k + \alpha_k d_k$

$r_{k+1} := r_k - \alpha_k A d_k$

если r_{k+1} достаточно мал, то выйти из цикла

$$\beta_k := \frac{r_{k+1}^\mathsf{T} r_{k+1}}{r_k^\mathsf{T} r_k}$$

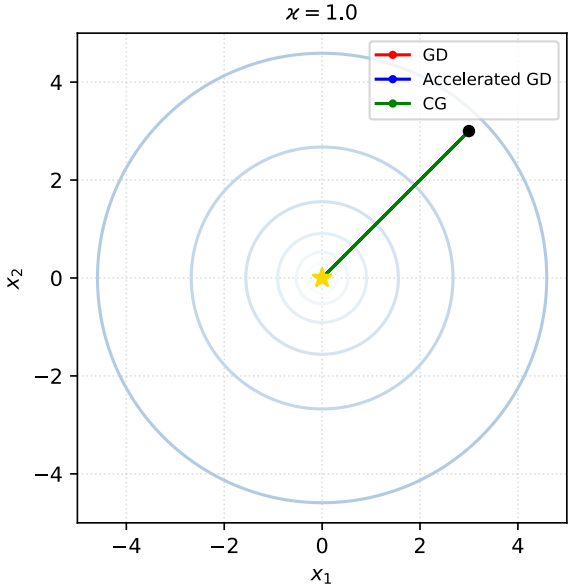
$d_{k+1} := r_{k+1} + \beta_k d_k$

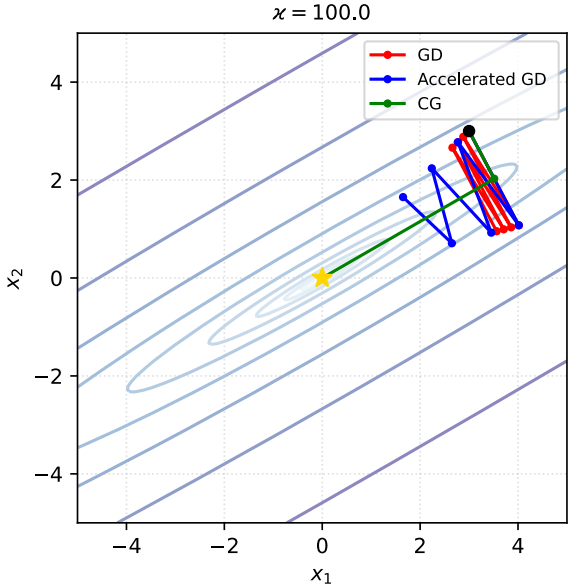
$k := k + 1$

конец повторения

вернуть x_{k+1} как результат

Закрываем квадратичный вопрос





Сходимость

Теорема 1. Если матрица A имеет только r различных собственных значений, то метод сопряженных градиентов сходится за r итераций.

Теорема 2. Следующая оценка сходимости выполняется для метода сопряженных градиентов, как для итерационного метода в сильно выпуклой задаче:

$$\|x_k - x^*\|_A \leq 2\left(\frac{\sqrt{\kappa(A)}-1}{\sqrt{\kappa(A)}+1}\right)^k \|x_0 - x^*\|_A,$$

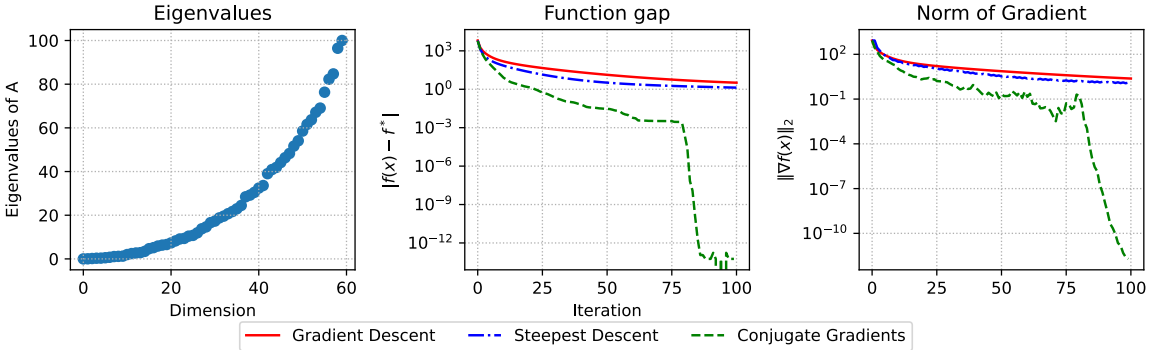
где $\|x\|_A^2 = x^\top Ax$ и $\varkappa(A) = \frac{\lambda_1(A)}{\lambda_n(A)}$ - это число обусловленности матрицы A, $\lambda_1(A) \geq ... \geq \lambda_n(A)$ - собственные значения матрицы A

Примечание: Сравните коэффициент геометрической прогрессии с его аналогом в методе градиентного спуска.

Численные эксперименты

$$f(x) = \frac{1}{2}x^T Ax - b^T x \to \min_{x \in \mathbb{R}^n}$$

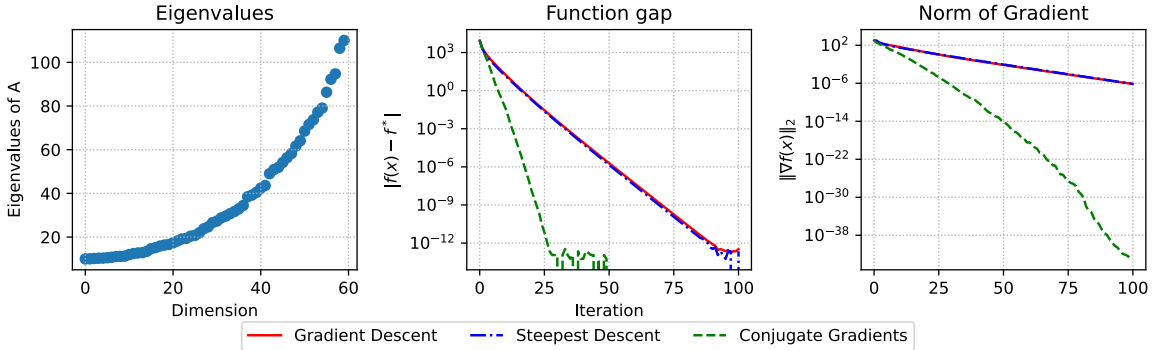
Convex quadratics. n=60, random matrix.



Численные эксперименты

$$f(x) = \frac{1}{2}x^T Ax - b^T x \to \min_{x \in \mathbb{R}^n}$$

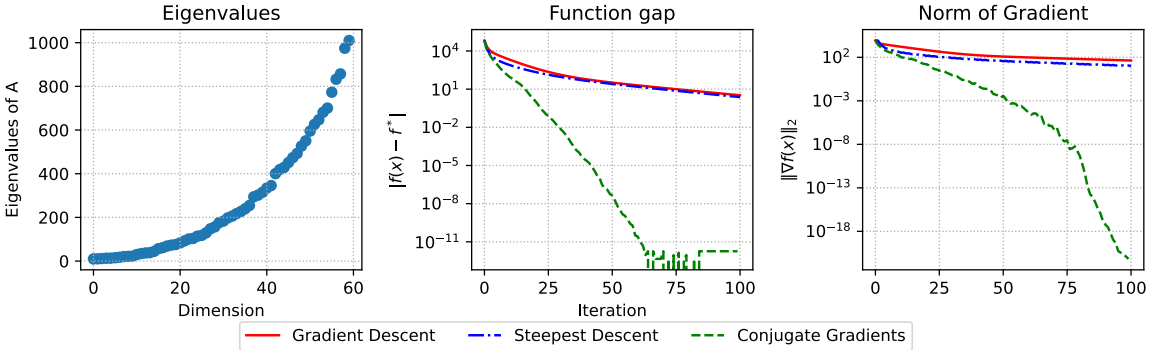
Strongly convex quadratics. n=60, random matrix.



Численные эксперименты

$$f(x) = \frac{1}{2}x^T Ax - b^T x \to \min_{x \in \mathbb{R}^n}$$

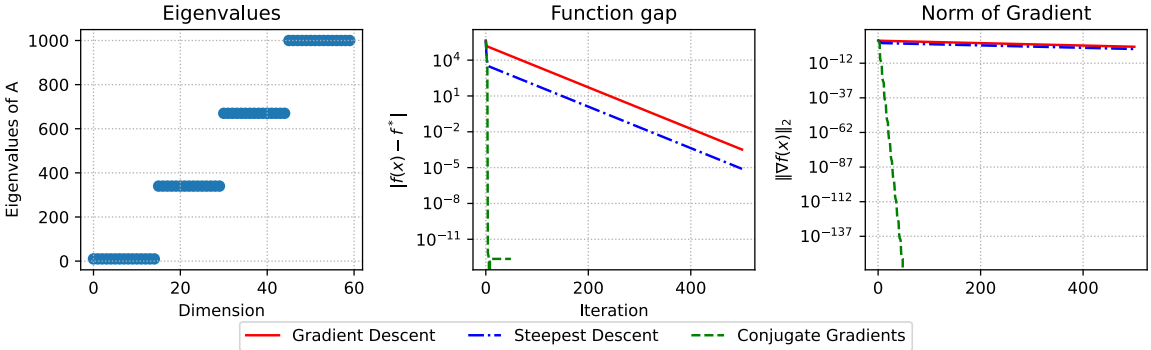
Strongly convex quadratics. n=60, random matrix.



Численные эксперименты

$$f(x) = \frac{1}{2} x^T A x - b^T x \to \min_{x \in \mathbb{R}^n}$$

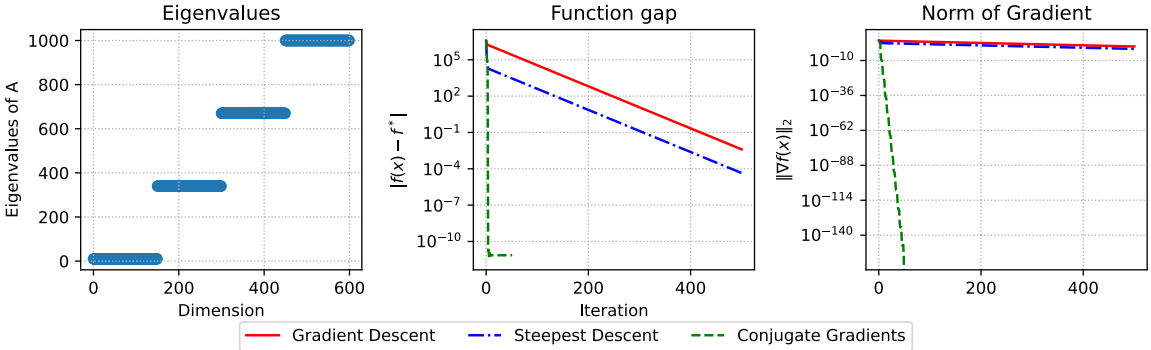
Strongly convex quadratics. n=60, clustered matrix.



Численные эксперименты

$$f(x) = \frac{1}{2}x^T Ax - b^T x \to \min_{x \in \mathbb{R}^n}$$

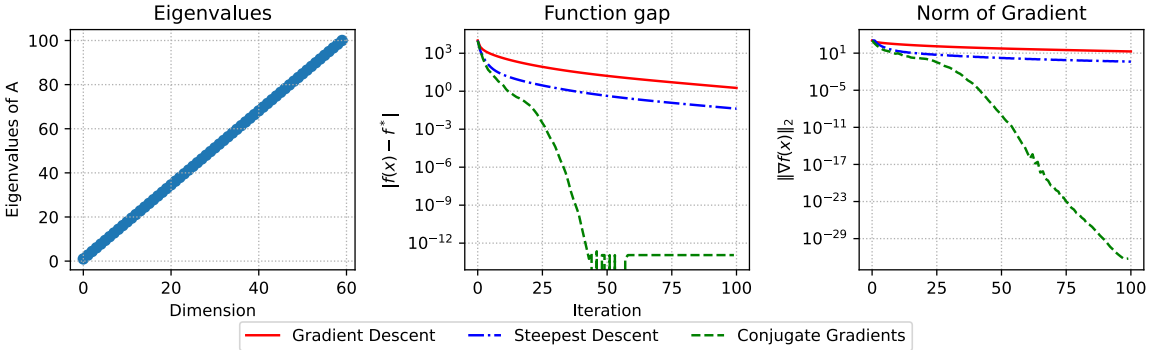
Strongly convex quadratics. n=600, clustered matrix.



Численные эксперименты

$$f(x) = \frac{1}{2}x^T Ax - b^T x \to \min_{x \in \mathbb{R}^n}$$

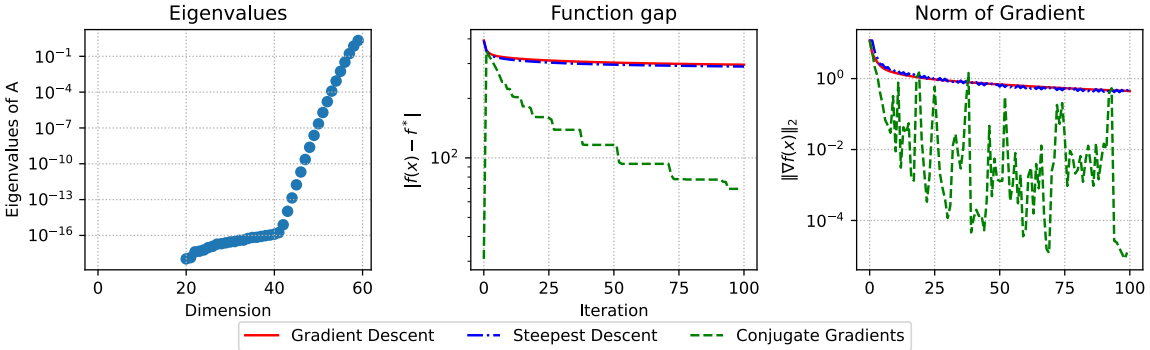
Strongly convex quadratics. n=60, uniform spectrum matrix.



Численные эксперименты

$$f(x) = \frac{1}{2}x^T Ax - b^T x \to \min_{x \in \mathbb{R}^n}$$

Strongly convex quadratics. n=60, Hilbert matrix.



Численные эксперименты

Посмотрим на видео с экспериментами в VS Code.

Метод сопряженных градиентов для неквадратичных задач (Non-linear CG)

Метод сопряженных градиентов для неквадратичных задач (Non-linear CG)

В случае, когда нет аналитического выражения для функции или ее градиента, мы, скорее всего, не сможем решить одномерную задачу минимизации аналитически. Поэтому α_k подбирается обычной процедурой линейного поиска. Но для выбора β_k есть следующий математический трюк:

Для двух итераций справедливо:

$$x_{k+1} - x_k = cd_k,$$

Метод сопряженных градиентов для неквадратичных задач (Non-linear CG)

В случае, когда нет аналитического выражения для функции или ее градиента, мы, скорее всего, не сможем решить одномерную задачу минимизации аналитически. Поэтому α_k подбирается обычной процедурой линейного поиска. Но для выбора β_k есть следующий математический трюк:

Для двух итераций справедливо:

$$x_{k+1} - x_k = cd_k,$$

где c - некоторая константа. Тогда для квадратичного случая мы имеем:

$$\nabla f(x_{k+1}) - \nabla f(x_k) = (Ax_{k+1} - b) - (Ax_k - b) = A(x_{k+1} - x_k) = cAd_k$$

Метод сопряженных градиентов для неквадратичных задач (Non-linear CG)

В случае, когда нет аналитического выражения для функции или ее градиента, мы, скорее всего, не сможем решить одномерную задачу минимизации аналитически. Поэтому α_k подбирается обычной процедурой линейного поиска. Но для выбора β_k есть следующий математический трюк:

Для двух итераций справедливо:

$$x_{k+1} - x_k = cd_k,$$

где c - некоторая константа. Тогда для квадратичного случая мы имеем:

$$\nabla f(x_{k+1}) - \nabla f(x_k) = (Ax_{k+1} - b) - (Ax_k - b) = A(x_{k+1} - x_k) = cAd_k$$

Выражая из этого уравнения величину $Ad_k = \frac{1}{c}\left(\nabla f(x_{k+1}) - \nabla f(x_k)\right)$, мы избавляемся от знания функции в определении β_k, тогда пункт 4 будет переписан как:

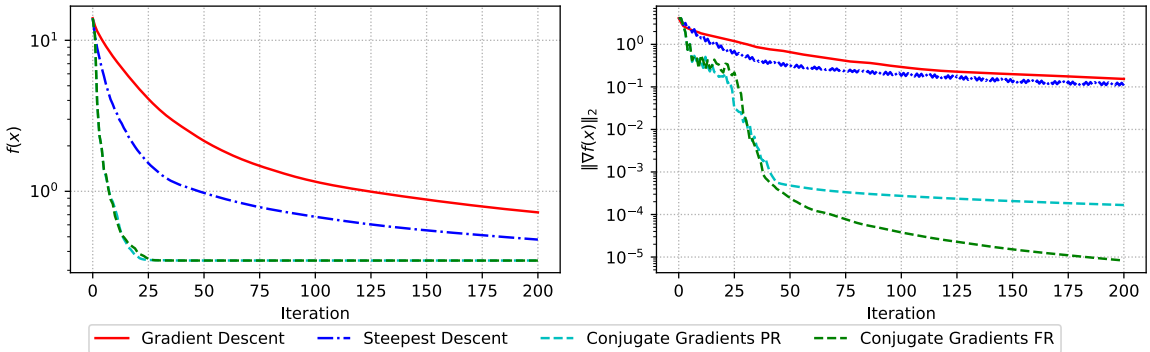
$$\beta_k = \frac{\nabla f(x_{k+1})^\top (\nabla f(x_{k+1}) - \nabla f(x_k))}{d_k^\top (\nabla f(x_{k+1}) - \nabla f(x_k))}.$$

Этот метод называется методом Полака-Рибьера.

Численные эксперименты

$$f(x) = \frac{\mu}{2}\|x\|_2^2 + \frac{1}{m}\sum_{i=1}^{m}\log(1+\exp(-y_i\langle a_i, x\rangle)) \to \min_{x\in\mathbb{R}^n}$$

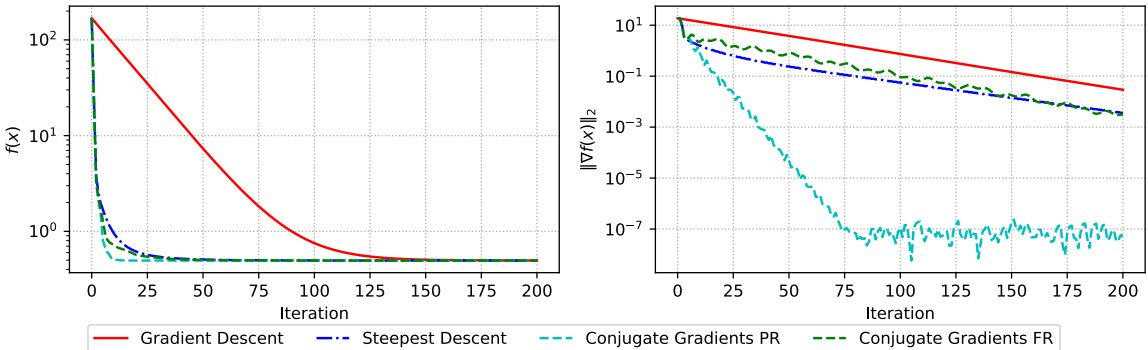
Regularized binary logistic regression. n=300. m=1000. μ=0



Численные эксперименты

$$f(x) = \frac{\mu}{2}\|x\|_2^2 + \frac{1}{m}\sum_{i=1}^{m}\log(1+\exp(-y_i\langle a_i, x\rangle)) \to \min_{x\in\mathbb{R}^n}$$

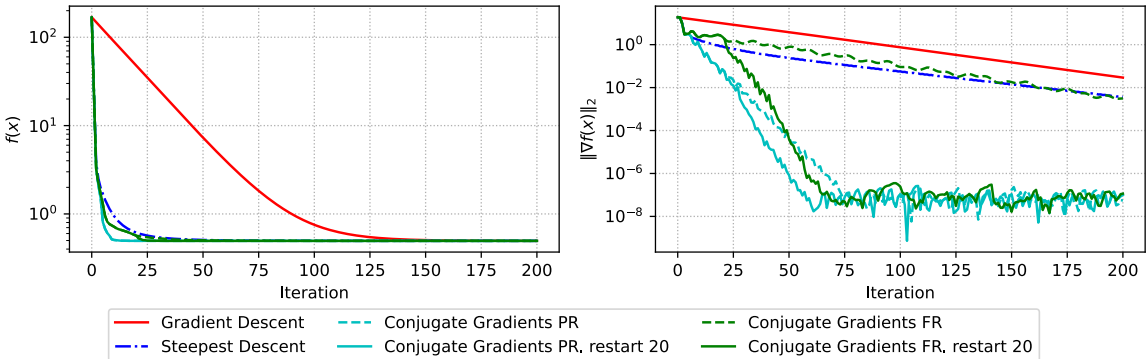
Regularized binary logistic regression. n=300. m=1000. μ=1



Численные эксперименты

$$f(x) = \frac{\mu}{2}\|x\|_2^2 + \frac{1}{m}\sum_{i=1}^{m}\log(1+\exp(-y_i\langle a_i, x\rangle)) \to \min_{x\in\mathbb{R}^n}$$

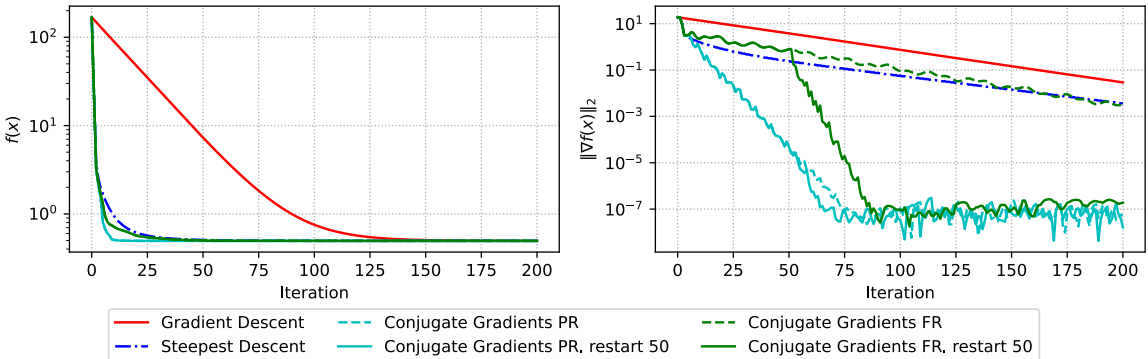
Regularized binary logistic regression. n=300. m=1000. μ=1



Численные эксперименты

$$f(x) = \frac{\mu}{2}\|x\|_2^2 + \frac{1}{m}\sum_{i=1}^{m}\log(1+\exp(-y_i\langle a_i, x\rangle)) \to \min_{x\in\mathbb{R}^n}$$

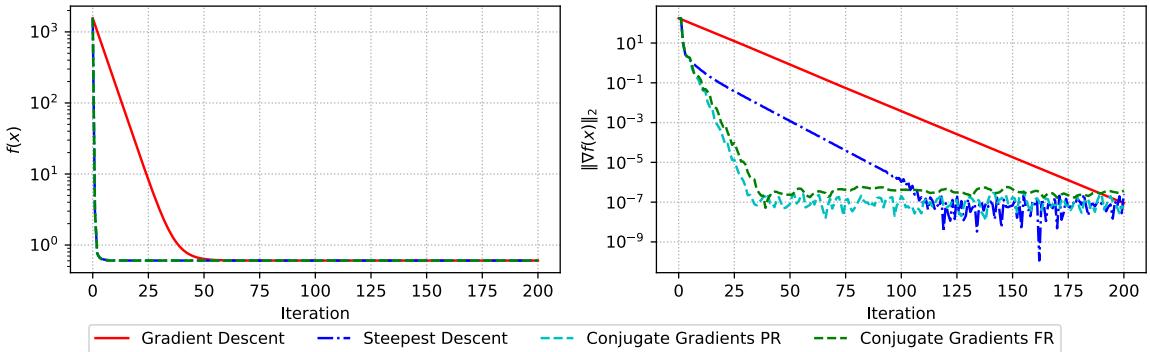
Regularized binary logistic regression. n=300. m=1000. μ=1



Численные эксперименты

$$f(x) = \frac{\mu}{2}\|x\|_2^2 + \frac{1}{m}\sum_{i=1}^{m}\log(1+\exp(-y_i\langle a_i, x\rangle)) \to \min_{x\in\mathbb{R}^n}$$

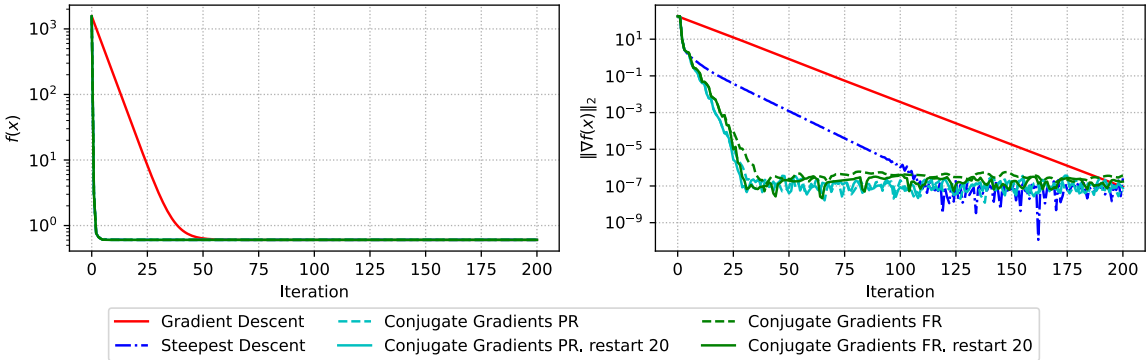
Regularized binary logistic regression. n=300. m=1000. μ=10



Численные эксперименты

$$f(x) = \frac{\mu}{2}\|x\|_2^2 + \frac{1}{m}\sum_{i=1}^{m}\log(1+\exp(-y_i\langle a_i, x\rangle)) \to \min_{x\in\mathbb{R}^n}$$

Regularized binary logistic regression. n=300. m=1000. μ=10



Бонус: дополнительные технические леммы и доказательства

Леммы для сходимости

Лемма 3. Разложение ошибки.

$$e_i = \sum_{j=i}^{n-1} -\alpha_j d_j \tag{7}$$

Леммы для сходимости

Лемма 3. Разложение ошибки.

$$e_i = \sum_{j=i}^{n-1} -\alpha_j d_j \tag{7}$$

Доказательство

По определению

$$e_i = e_0 + \sum_{j=0}^{i-1} \alpha_j d_j$$

Леммы для сходимости

Лемма 3. Разложение ошибки.

$$e_i = \sum_{j=i}^{n-1} -\alpha_j d_j \tag{7}$$

Доказательство

По определению

$$e_i = e_0 + \sum_{j=0}^{i-1} \alpha_j d_j = x_0 - x^* + \sum_{j=0}^{i-1} \alpha_j d_j$$

Леммы для сходимости

Лемма 3. Разложение ошибки.

$$e_i = \sum_{j=i}^{n-1} -\alpha_j d_j \tag{7}$$

Доказательство

По определению

$$e_i = e_0 + \sum_{j=0}^{i-1} \alpha_j d_j = x_0 - x^* + \sum_{j=0}^{i-1} \alpha_j d_j = -\sum_{j=0}^{n-1} \alpha_j d_j + \sum_{j=0}^{i-1} \alpha_j d_j$$

Леммы для сходимости

Лемма 3. Разложение ошибки.

$$e_i = \sum_{j=i}^{n-1} -\alpha_j d_j \tag{7}$$

Доказательство

По определению

$$e_i = e_0 + \sum_{j=0}^{i-1} \alpha_j d_j = x_0 - x^* + \sum_{j=0}^{i-1} \alpha_j d_j = -\sum_{j=0}^{n-1} \alpha_j d_j + \sum_{j=0}^{i-1} \alpha_j d_j = \sum_{j=i}^{n-1} -\alpha_j d_j$$

Леммы для сходимости

Лемма 4. Невязка ортогональна всем предыдущим направлениям для CD.

Рассмотрим невязку метода сопряженных направлений на k итерации r_k, тогда для любого $i < k$:

$$d_i^T r_k = 0 \tag{8}$$

Леммы для сходимости

Лемма 4. Невязка ортогональна всем предыдущим направлениям для CD.

Рассмотрим невязку метода сопряженных направлений на k итерации r_k, тогда для любого $i < k$:

$$d_i^T r_k = 0 \tag{8}$$

Доказательство

Запишем (7) для некоторого фиксированного индекса k:

Леммы для сходимости

Лемма 4. Невязка ортогональна всем предыдущим направлениям для CD.

Рассмотрим невязку метода сопряженных направлений на k итерации r_k, тогда для любого $i < k$:

$$d_i^T r_k = 0 \tag{8}$$

Доказательство

Запишем (7) для некоторого фиксированного индекса k:

$$e_k = \sum_{j=k}^{n-1} -\alpha_j d_j$$

Леммы для сходимости

Лемма 4. Невязка ортогональна всем предыдущим направлениям для CD.

Рассмотрим невязку метода сопряженных направлений на k итерации r_k, тогда для любого $i < k$:

$$d_i^T r_k = 0 \tag{8}$$

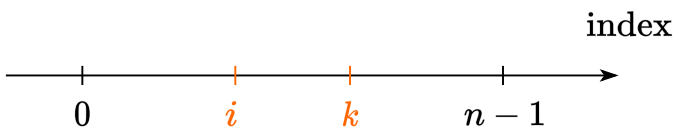
Доказательство

Запишем (7) для некоторого фиксированного индекса k:

$$e_k = \sum_{j=k}^{n-1} -\alpha_j d_j$$

Умножаем обе части на $-d_i^T A$·

$$-d_i^T A e_k = \sum_{j=k}^{n-1} \alpha_j d_i^T A d_j = 0$$



Таким образом, $d_i^T r_k = 0$ и невязка r_k ортогональна всем предыдущим направлениям d_i для метода CD.

Леммы для сходимости

Лемма 5. Невязки ортогональны друг другу в методе CG

Все невязки в методе CG ортогональны друг другу:

$$r_i^T r_k = 0 \qquad \forall i \neq k \tag{9}$$

Леммы для сходимости

Лемма 5. Невязки ортогональны друг другу в методе CG

Все невязки в методе CG ортогональны друг другу:

$$r_i^T r_k = 0 \qquad \forall i \neq k \tag{9}$$

Доказательство

Запишем процесс Грама-Шмидта (2) с $\langle \cdot, \cdot \rangle$ замененным на $\langle \cdot, \cdot \rangle_A = x^T A y$

Леммы для сходимости

Лемма 5. Невязки ортогональны друг другу в методе CG

Все невязки в методе CG ортогональны друг другу:

$$r_i^T r_k = 0 \qquad \forall i \neq k \tag{9}$$

Доказательство

Запишем процесс Грама-Шмидта (2) с $\langle \cdot, \cdot \rangle$ замененным на $\langle \cdot, \cdot \rangle_A = x^T A y$

$$d_i = u_i + \sum_{j=0}^{i-1} \beta_{ji} d_j \;\; \beta_{ji} = -\frac{\langle d_j, u_i \rangle_A}{\langle d_j, d_j \rangle_A} \tag{10}$$

Леммы для сходимости

Лемма 5. Невязки ортогональны друг другу в методе CG

Все невязки в методе CG ортогональны друг другу:

$$r_i^T r_k = 0 \qquad \forall i \neq k \tag{9}$$

Доказательство

Запишем процесс Грама-Шмидта (2) с $\langle \cdot, \cdot \rangle$ замененным на $\langle \cdot, \cdot \rangle_A = x^T A y$

$$d_i = u_i + \sum_{j=0}^{i-1} \beta_{ji} d_j \quad \beta_{ji} = -\frac{\langle d_j, u_i \rangle_A}{\langle d_j, d_j \rangle_A} \tag{10}$$

Тогда, мы используем невязки в качестве начальных векторов для процесса и $u_i = r_i$.

Леммы для сходимости

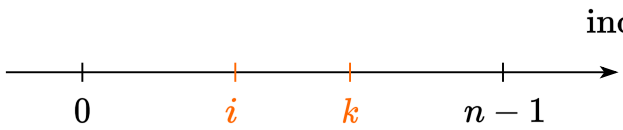
Лемма 5. Невязки ортогональны друг другу в методе CG

Все невязки в методе CG ортогональны друг другу:

$$r_i^T r_k = 0 \qquad \forall i \neq k \tag{9}$$

Доказательство

Запишем процесс Грама-Шмидта (2) с $\langle \cdot, \cdot \rangle$ замененным на $\langle \cdot, \cdot \rangle_A = x^T A y$



$$d_i = u_i + \sum_{j=0}^{i-1} \beta_{ji} d_j \quad \beta_{ji} = -\frac{\langle d_j, u_i \rangle_A}{\langle d_j, d_j \rangle_A} \tag{10}$$

Умножаем обе части (10) на $r_k^T \cdot$ для некоторого индекса k:

$$r_k^T d_i = r_k^T u_i + \sum_{j=0}^{i-1} \beta_{ji} r_k^T d_j$$

Тогда, мы используем невязки в качестве начальных векторов для процесса и $u_i = r_i$.

$$d_i = r_i + \sum_{j=0}^{i-1} \beta_{ji} d_j \quad \beta_{ji} = -\frac{\langle d_j, r_i \rangle_A}{\langle d_j, d_j \rangle_A} \tag{11}$$

Леммы для сходимости

Лемма 5. Невязки ортогональны друг другу в методе CG

Все невязки в методе CG ортогональны друг другу:

$$r_i^T r_k = 0 \qquad \forall i \neq k \tag{9}$$

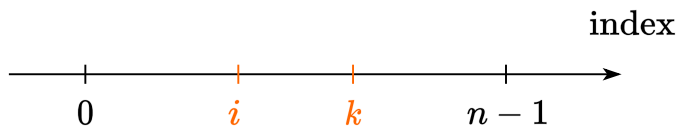
Доказательство

Запишем процесс Грама-Шмидта (2) с $\langle \cdot, \cdot \rangle$ замененным на $\langle \cdot, \cdot \rangle_A = x^T A y$

$$d_i = u_i + \sum_{j=0}^{i-1} \beta_{ji} d_j \quad \beta_{ji} = -\frac{\langle d_j, u_i \rangle_A}{\langle d_j, d_j \rangle_A} \tag{10}$$

Тогда, мы используем невязки в качестве начальных векторов для процесса и $u_i = r_i$.

$$d_i = r_i + \sum_{j=0}^{i-1} \beta_{ji} d_j \quad \beta_{ji} = -\frac{\langle d_j, r_i \rangle_A}{\langle d_j, d_j \rangle_A} \tag{11}$$



Умножаем обе части (10) на $r_k^T \cdot$ для некоторого индекса k:

$$r_k^T d_i = r_k^T u_i + \sum_{j=0}^{i-1} \beta_{ji} r_k^T d_j$$

Если $j < i < k$, то имеем лемму 4 с $d_i^T r_k = 0$ и $d_j^T r_k = 0$. Имеем:

$r_k^T u_i = 0$ для CD $r_k^T r_i = 0$ для CG

Леммы для сходимости

Более того, если $k = i$:

$$r_k^T d_k = r_k^T u_k + \sum_{j=0}^{k-1} \beta_{jk} r_k^T d_j$$

Леммы для сходимости

Более того, если $k = i$:

$$r_k^T d_k = r_k^T u_k + \sum_{j=0}^{k-1} \beta_{jk} r_k^T d_j = r_k^T u_k + 0,$$

Леммы для сходимости

Более того, если $k = i$:

$$r_k^T d_k = r_k^T u_k + \sum_{j=0}^{k-1} \beta_{jk} r_k^T d_j = r_k^T u_k + 0,$$

Леммы для сходимости

Более того, если $k = i$:

$$r_k^T d_k = r_k^T u_k + \sum_{j=0}^{k-1} \beta_{jk} r_k^T d_j = r_k^T u_k + 0,$$

и мы имеем для любого k (из-за произвольного выбора i):

$$r_k^T d_k = r_k^T u_k. \tag{12}$$

Леммы для сходимости

Более того, если $k = i$:

$$r_k^T d_k = r_k^T u_k + \sum_{j=0}^{k-1} \beta_{jk} r_k^T d_j = r_k^T u_k + 0,$$

и мы имеем для любого k (из-за произвольного выбора i):

$$r_k^T d_k = r_k^T u_k. \tag{12}$$

Леммы для сходимости

Лемма 6. Пересчет невязки

$$r_{k+1} = r_k - \alpha_k A d_k \tag{13}$$

Леммы для сходимости

Лемма 6. Пересчет невязки

$$r_{k+1} = r_k - \alpha_k A d_k \tag{13}$$

$$r_{k+1} = -Ae_{k+1} = -A\left(e_k + \alpha_k d_k\right) = -Ae_k - \alpha_k A d_k = r_k - \alpha_k A d_k$$

Наконец, все эти вышеуказанные леммы достаточны для доказательства, что $\beta_{ji} = 0$ для всех i, j, кроме соседних.

Леммы для сходимости

Лемма 7. Коэффициенты для процесса Грама-Шмидта для CG

В процессе Грама-Шмидта для CG

$$\beta_{ji} = \frac{\langle r_i, r_i \rangle}{r_{i-1}, r_{i-1}}, \; i = j + 1.$$

Все остальные коэффициенты равны нулю кроме $i = j$, но этот случай нам неинтересен.

Рассмотрим процесс Грам-Шмидта в методе CG

$$\beta_{ji} = -\frac{\langle d_j, u_i \rangle_A}{\langle d_j, d_j \rangle_A}$$

Леммы для сходимости

Лемма 7. Коэффициенты для процесса Грама-Шмидта для CG

В процессе Грама-Шмидта для CG

$$\beta_{ji} = \frac{\langle r_i, r_i \rangle}{r_{i-1}, r_{i-1}}, \; i = j + 1.$$

Все остальные коэффициенты равны нулю кроме $i = j$, но этот случай нам неинтересен.

Рассмотрим процесс Грам-Шмидта в методе CG

$$\beta_{ji} = -\frac{\langle d_j, u_i \rangle_A}{\langle d_j, d_j \rangle_A} = -\frac{d_j^T A u_i}{d_j^T A d_j}$$

Леммы для сходимости

Лемма 7. Коэффициенты для процесса Грама-Шмидта для CG

В процессе Грама-Шмидта для CG

$$\beta_{ji} = \frac{\langle r_i, r_i \rangle}{r_{i-1}, r_{i-1}}, \; i = j + 1.$$

Все остальные коэффициенты равны нулю кроме $i = j$, но этот случай нам неинтересен.

Рассмотрим процесс Грам-Шмидта в методе CG

$$\beta_{ji} = -\frac{\langle d_j, u_i \rangle_A}{\langle d_j, d_j \rangle_A} = -\frac{d_j^T A u_i}{d_j^T A d_j} = -\frac{d_j^T A r_i}{d_j^T A d_j}$$

Леммы для сходимости

Лемма 7. Коэффициенты для процесса Грама-Шмидта для CG

В процессе Грама-Шмидта для CG

$$\beta_{ji} = \frac{\langle r_i, r_i \rangle}{r_{i-1}, r_{i-1}}, \; i = j + 1.$$

Все остальные коэффициенты равны нулю кроме $i = j$, но этот случай нам неинтересен.

Рассмотрим процесс Грам-Шмидта в методе CG

$$\beta_{ji} = -\frac{\langle d_j, u_i \rangle_A}{\langle d_j, d_j \rangle_A} = -\frac{d_j^T A u_i}{d_j^T A d_j} = -\frac{d_j^T A r_i}{d_j^T A d_j} = -\frac{r_i^T A d_j}{d_j^T A d_j}.$$

Леммы для сходимости

Лемма 7. Коэффициенты для процесса Грама-Шмидта для CG

В процессе Грама-Шмидта для CG

$$\beta_{ji} = \frac{\langle r_i, r_i \rangle}{r_{i-1}, r_{i-1}}, \; i = j + 1.$$

Все остальные коэффициенты равны нулю кроме $i = j$, но этот случай нам неинтересен.

Рассмотрим процесс Грам-Шмидта в методе CG

$$\beta_{ji} = -\frac{\langle d_j, u_i \rangle_A}{\langle d_j, d_j \rangle_A} = -\frac{d_j^T A u_i}{d_j^T A d_j} = -\frac{d_j^T A r_i}{d_j^T A d_j} = -\frac{r_i^T A d_j}{d_j^T A d_j}.$$

Леммы для сходимости

Лемма 7. Коэффициенты для процесса Грама-Шмидта для CG

В процессе Грама-Шмидта для CG

$$\beta_{ji} = \frac{\langle r_i, r_i \rangle}{r_{i-1}, r_{i-1}}, \; i = j + 1.$$

Все остальные коэффициенты равны нулю кроме $i = j$, но этот случай нам неинтересен.

Рассмотрим процесс Грам-Шмидта в методе CG

$$\beta_{ji} = -\frac{\langle d_j, u_i \rangle_A}{\langle d_j, d_j \rangle_A} = -\frac{d_j^T A u_i}{d_j^T A d_j} = -\frac{d_j^T A r_i}{d_j^T A d_j} = -\frac{r_i^T A d_j}{d_j^T A d_j}.$$

Рассмотрим скалярное произведение $\langle r_i, r_{j+1} \rangle$ используя (13):

$$\langle r_i, r_{j+1} \rangle$$

Леммы для сходимости

Лемма 7. Коэффициенты для процесса Грама-Шмидта для CG

В процессе Грама-Шмидта для CG

$$\beta_{ji} = \frac{\langle r_i, r_i \rangle}{r_{i-1}, r_{i-1}}, \; i = j + 1.$$

Все остальные коэффициенты равны нулю кроме $i = j$, но этот случай нам неинтересен.

Рассмотрим процесс Грам-Шмидта в методе CG

$$\beta_{ji} = -\frac{\langle d_j, u_i \rangle_A}{\langle d_j, d_j \rangle_A} = -\frac{d_j^T A u_i}{d_j^T A d_j} = -\frac{d_j^T A r_i}{d_j^T A d_j} = -\frac{r_i^T A d_j}{d_j^T A d_j}.$$

Рассмотрим скалярное произведение $\langle r_i, r_{j+1} \rangle$ используя (13):

$$\langle r_i, r_{j+1} \rangle = \langle r_i, r_j - \alpha_j A d_j \rangle$$

Леммы для сходимости

Лемма 7. Коэффициенты для процесса Грама-Шмидта для CG

В процессе Грама-Шмидта для CG

$$\beta_{ji} = \frac{\langle r_i, r_i \rangle}{r_{i-1}, r_{i-1}}, \; i = j + 1.$$

Все остальные коэффициенты равны нулю кроме $i = j$, но этот случай нам неинтересен.

Рассмотрим процесс Грам-Шмидта в методе CG

$$\beta_{ji} = -\frac{\langle d_j, u_i \rangle_A}{\langle d_j, d_j \rangle_A} = -\frac{d_j^T A u_i}{d_j^T A d_j} = -\frac{d_j^T A r_i}{d_j^T A d_j} = -\frac{r_i^T A d_j}{d_j^T A d_j}.$$

Рассмотрим скалярное произведение $\langle r_i, r_{j+1} \rangle$ используя (13):

$$\langle r_i, r_{j+1} \rangle = \langle r_i, r_j - \alpha_j A d_j \rangle = \langle r_i, r_j \rangle - \alpha_j \langle r_i, A d_j \rangle$$

Леммы для сходимости

Лемма 7. Коэффициенты для процесса Грама-Шмидта для CG

В процессе Грама-Шмидта для CG

$$\beta_{ji} = \frac{\langle r_i, r_i \rangle}{r_{i-1}, r_{i-1}}, \; i = j + 1.$$

Все остальные коэффициенты равны нулю кроме $i = j$, но этот случай нам неинтересен.

Рассмотрим процесс Грам-Шмидта в методе CG

$$\beta_{ji} = -\frac{\langle d_j, u_i \rangle_A}{\langle d_j, d_j \rangle_A} = -\frac{d_j^T A u_i}{d_j^T A d_j} = -\frac{d_j^T A r_i}{d_j^T A d_j} = -\frac{r_i^T A d_j}{d_j^T A d_j}.$$

Рассмотрим скалярное произведение $\langle r_i, r_{j+1} \rangle$ используя (13):

$$\langle r_i, r_{j+1} \rangle = \langle r_i, r_j - \alpha_j A d_j \rangle = \langle r_i, r_j \rangle - \alpha_j \langle r_i, A d_j \rangle$$

$\alpha_j \langle r_i, A d_j \rangle$

Леммы для сходимости

Лемма 7. Коэффициенты для процесса Грама-Шмидта для CG

В процессе Грама-Шмидта для CG

$$\beta_{ji} = \frac{\langle r_i, r_i \rangle}{r_{i-1}, r_{i-1}}, \; i = j + 1.$$

Все остальные коэффициенты равны нулю кроме $i = j$, но этот случай нам неинтересен.

Рассмотрим процесс Грам-Шмидта в методе CG

$$\beta_{ji} = -\frac{\langle d_j, u_i \rangle_A}{\langle d_j, d_j \rangle_A} = -\frac{d_j^T A u_i}{d_j^T A d_j} = -\frac{d_j^T A r_i}{d_j^T A d_j} = -\frac{r_i^T A d_j}{d_j^T A d_j}.$$

Рассмотрим скалярное произведение $\langle r_i, r_{j+1} \rangle$ используя (13):

$$\langle r_i, r_{j+1} \rangle = \langle r_i, r_j - \alpha_j A d_j \rangle = \langle r_i, r_j \rangle - \alpha_j \langle r_i, A d_j \rangle$$
$$\alpha_j \langle r_i, A d_j \rangle = \langle r_i, r_j \rangle - \langle r_i, r_{j+1} \rangle$$

Леммы для сходимости

1. Если $i = j$: $\alpha_i \langle r_i, Ad_i \rangle = \langle r_i, r_i \rangle - \langle r_i, r_{i+1} \rangle = \langle r_i, r_i \rangle$. Этот случай не интересен по построению процесса Грам-Шмидта.

Леммы для сходимости

1. Если $i = j$: $\alpha_i \langle r_i, Ad_i \rangle = \langle r_i, r_i \rangle - \langle r_i, r_{i+1} \rangle = \langle r_i, r_i \rangle$. Этот случай не интересен по построению процесса Грам-Шмидта.
2. Соседний случай $i = j + 1$: $\alpha_j \langle r_i, Ad_j \rangle = \langle r_i, r_{i-1} \rangle - \langle r_i, r_i \rangle = -\langle r_i, r_i \rangle$

Леммы для сходимости

1. Если $i = j$: $\alpha_i \langle r_i, Ad_i \rangle = \langle r_i, r_i \rangle - \langle r_i, r_{i+1} \rangle = \langle r_i, r_i \rangle$. Этот случай не интересен по построению процесса Грам-Шмидта.
2. Соседний случай $i = j + 1$: $\alpha_j \langle r_i, Ad_j \rangle = \langle r_i, r_{i-1} \rangle - \langle r_i, r_i \rangle = -\langle r_i, r_i \rangle$
3. Для любого другого случая: $\alpha_j \langle r_i, Ad_j \rangle = 0$, потому что все невязки ортогональны друг другу.

Леммы для сходимости

1. Если $i = j$: $\alpha_i \langle r_i, Ad_i \rangle = \langle r_i, r_i \rangle - \langle r_i, r_{i+1} \rangle = \langle r_i, r_i \rangle$. Этот случай не интересен по построению процесса Грам-Шмидта.
2. Соседний случай $i = j + 1$: $\alpha_j \langle r_i, Ad_j \rangle = \langle r_i, r_{i-1} \rangle - \langle r_i, r_i \rangle = -\langle r_i, r_i \rangle$
3. Для любого другого случая: $\alpha_j \langle r_i, Ad_j \rangle = 0$, потому что все невязки ортогональны друг другу.

Леммы для сходимости

1. Если $i = j$: $\alpha_i \langle r_i, Ad_i \rangle = \langle r_i, r_i \rangle - \langle r_i, r_{i+1} \rangle = \langle r_i, r_i \rangle$. Этот случай не интересен по построению процесса Грам-Шмидта.
2. Соседний случай $i = j + 1$: $\alpha_j \langle r_i, Ad_j \rangle = \langle r_i, r_{i-1} \rangle - \langle r_i, r_i \rangle = -\langle r_i, r_i \rangle$
3. Для любого другого случая: $\alpha_j \langle r_i, Ad_j \rangle = 0$, потому что все невязки ортогональны друг другу.

Наконец, мы имеем формулу для $i = j + 1$:

$$\beta_{ji} = -\frac{r_i^T A d_j}{d_j^T A d_j}$$

Леммы для сходимости

1. Если $i = j$: $\alpha_i \langle r_i, Ad_i \rangle = \langle r_i, r_i \rangle - \langle r_i, r_{i+1} \rangle = \langle r_i, r_i \rangle$. Этот случай не интересен по построению процесса Грам-Шмидта.
2. Соседний случай $i = j + 1$: $\alpha_j \langle r_i, Ad_j \rangle = \langle r_i, r_{i-1} \rangle - \langle r_i, r_i \rangle = -\langle r_i, r_i \rangle$
3. Для любого другого случая: $\alpha_j \langle r_i, Ad_j \rangle = 0$, потому что все невязки ортогональны друг другу.

Наконец, мы имеем формулу для $i = j + 1$:

$$\beta_{ji} = -\frac{r_i^T A d_j}{d_j^T A d_j} = \frac{1}{\alpha_j} \frac{\langle r_i, r_i \rangle}{d_j^T A d_j}$$

Леммы для сходимости

1. Если $i = j$: $\alpha_i \langle r_i, Ad_i \rangle = \langle r_i, r_i \rangle - \langle r_i, r_{i+1} \rangle = \langle r_i, r_i \rangle$. Этот случай не интересен по построению процесса Грам-Шмидта.
2. Соседний случай $i = j + 1$: $\alpha_j \langle r_i, Ad_j \rangle = \langle r_i, r_{i-1} \rangle - \langle r_i, r_i \rangle = -\langle r_i, r_i \rangle$
3. Для любого другого случая: $\alpha_j \langle r_i, Ad_j \rangle = 0$, потому что все невязки ортогональны друг другу.

Наконец, мы имеем формулу для $i = j + 1$:

$$\beta_{ji} = -\frac{r_i^T A d_j}{d_j^T A d_j} = \frac{1}{\alpha_j} \frac{\langle r_i, r_i \rangle}{d_j^T A d_j} = \frac{d_j^T A d_j}{d_j^T r_j} \frac{\langle r_i, r_i \rangle}{d_j^T A d_j}$$

Леммы для сходимости

1. Если $i = j$: $\alpha_i \langle r_i, Ad_i \rangle = \langle r_i, r_i \rangle - \langle r_i, r_{i+1} \rangle = \langle r_i, r_i \rangle$. Этот случай не интересен по построению процесса Грам-Шмидта.
2. Соседний случай $i = j + 1$: $\alpha_j \langle r_i, Ad_j \rangle = \langle r_i, r_{i-1} \rangle - \langle r_i, r_i \rangle = -\langle r_i, r_i \rangle$
3. Для любого другого случая: $\alpha_j \langle r_i, Ad_j \rangle = 0$, потому что все невязки ортогональны друг другу.

Наконец, мы имеем формулу для $i = j + 1$:

$$\beta_{ji} = -\frac{r_i^T A d_j}{d_j^T A d_j} = \frac{1}{\alpha_j} \frac{\langle r_i, r_i \rangle}{d_j^T A d_j} = \frac{d_j^T A d_j}{d_j^T r_j} \frac{\langle r_i, r_i \rangle}{d_j^T A d_j} = \frac{\langle r_i, r_i \rangle}{\langle r_j, r_j \rangle}$$

Леммы для сходимости

1. Если $i = j$: $\alpha_i \langle r_i, Ad_i \rangle = \langle r_i, r_i \rangle - \langle r_i, r_{i+1} \rangle = \langle r_i, r_i \rangle$. Этот случай не интересен по построению процесса Грам-Шмидта.
2. Соседний случай $i = j + 1$: $\alpha_j \langle r_i, Ad_j \rangle = \langle r_i, r_{i-1} \rangle - \langle r_i, r_i \rangle = -\langle r_i, r_i \rangle$
3. Для любого другого случая: $\alpha_j \langle r_i, Ad_j \rangle = 0$, потому что все невязки ортогональны друг другу.

Наконец, мы имеем формулу для $i = j + 1$:

$$\beta_{ji} = -\frac{r_i^T A d_j}{d_j^T A d_j} = \frac{1}{\alpha_j} \frac{\langle r_i, r_i \rangle}{d_j^T A d_j} = \frac{d_j^T A d_j}{d_j^T r_j} \frac{\langle r_i, r_i \rangle}{d_j^T A d_j} = \frac{\langle r_i, r_i \rangle}{\langle r_j, r_j \rangle} = \frac{\langle r_i, r_i \rangle}{\langle r_{i-1}, r_{i-1} \rangle}$$

Леммы для сходимости

1. Если $i = j$: $\alpha_i \langle r_i, Ad_i \rangle = \langle r_i, r_i \rangle - \langle r_i, r_{i+1} \rangle = \langle r_i, r_i \rangle$. Этот случай не интересен по построению процесса Грам-Шмидта.
2. Соседний случай $i = j + 1$: $\alpha_j \langle r_i, Ad_j \rangle = \langle r_i, r_{i-1} \rangle - \langle r_i, r_i \rangle = -\langle r_i, r_i \rangle$
3. Для любого другого случая: $\alpha_j \langle r_i, Ad_j \rangle = 0$, потому что все невязки ортогональны друг другу.

Наконец, мы имеем формулу для $i = j + 1$:

$$\beta_{ji} = -\frac{r_i^T A d_j}{d_j^T A d_j} = \frac{1}{\alpha_j} \frac{\langle r_i, r_i \rangle}{d_j^T A d_j} = \frac{d_j^T A d_j}{d_j^T r_j} \frac{\langle r_i, r_i \rangle}{d_j^T A d_j} = \frac{\langle r_i, r_i \rangle}{\langle r_j, r_j \rangle} = \frac{\langle r_i, r_i \rangle}{\langle r_{i-1}, r_{i-1} \rangle}$$

Леммы для сходимости

1. Если $i = j$: $\alpha_i \langle r_i, Ad_i \rangle = \langle r_i, r_i \rangle - \langle r_i, r_{i+1} \rangle = \langle r_i, r_i \rangle$. Этот случай не интересен по построению процесса Грам-Шмидта.
2. Соседний случай $i = j + 1$: $\alpha_j \langle r_i, Ad_j \rangle = \langle r_i, r_{i-1} \rangle - \langle r_i, r_i \rangle = -\langle r_i, r_i \rangle$
3. Для любого другого случая: $\alpha_j \langle r_i, Ad_j \rangle = 0$, потому что все невязки ортогональны друг другу.

Наконец, мы имеем формулу для $i = j + 1$:

$$\beta_{ji} = -\frac{r_i^T A d_j}{d_j^T A d_j} = \frac{1}{\alpha_j} \frac{\langle r_i, r_i \rangle}{d_j^T A d_j} = \frac{d_j^T A d_j}{d_j^T r_j} \frac{\langle r_i, r_i \rangle}{d_j^T A d_j} = \frac{\langle r_i, r_i \rangle}{\langle r_j, r_j \rangle} = \frac{\langle r_i, r_i \rangle}{\langle r_{i-1}, r_{i-1} \rangle}$$

И для направления $d_{k+1} = r_{k+1} + \beta_{k,k+1} d_k, \qquad \beta_{k,k+1} = \beta_k = \dfrac{\langle r_{k+1}, r_{k+1} \rangle}{\langle r_k, r_k \rangle}.$

Бонус: нижние оценки для методов I порядка (🔗источник)

Тип задачи	Критерий	Нижняя оценка	Верхняя оценка	Ссылка (Ниж.)	Ссылка (Верх.)
L-гладкая выпуклая	Зазор оптимальности	$\Omega\left(\sqrt{L}\,\varepsilon^{-1}\right)$	✔(точное совпадение)	[1], Теорема 2.1.7	[1], Теорема 2.2.2
L-гладкая μ-сильно выпуклая	Зазор оптимальности	$\Omega\left(\sqrt{\varkappa}\,\log\frac{1}{\varepsilon}\right)$	✔	[1], Теорема 2.1.13	[1], Теорема 2.2.2
Негладкая G-липшицева выпуклая	Зазор оптимальности	$\Omega\left(G^2\,\varepsilon^{-2}\right)$	✔(точное совпадение)	[1], Теорема 3.2.1	[1], Теорема 3.2.2
Негладкая G-липшицева μ-сильно выпуклая	Зазор оптимальности	$\Omega\left(G^2\,(\mu\varepsilon)^{-1}\right)$	✔	[1], Теорема 3.2.5	[3], Теорема 3.9
L-гладкая выпуклая (сходимость по функции)	Стационарность	$\Omega\left(\sqrt{\Delta L}\,\varepsilon^{-1}\right)$	✔(с точностью до логарифмического множителя)	[2], Теорема 1	[2], Приложение A.1
L-гладкая выпуклая (сходимость по аргументу)	Стационарность	$\Omega\left(\sqrt{DL}\,\varepsilon^{-1/2}\right)$	✔	[2], Теорема 1	[6], Раздел 6.5
L-гладкая невыпуклая	Стационарность	$\Omega\left(\Delta L\,\varepsilon^{-2}\right)$	✔	[5], Теорема 1	[7], Теорема 10.15
Негладкая G-липшицева ρ-слабо выпуклая (WC)	Квази-стационарность	Неизвестно	$\mathcal{O}\left(\varepsilon^{-4}\right)$	/	[8], Следствие 2.2
L-гладкая μ-PL	Зазор оптимальности	$\Omega\left(\varkappa\log\frac{1}{\varepsilon}\right)$	✔	[9], Теорема 3	[10], Теорема 1

Источники:

- [1] - Lectures on Convex Optimization, Y. Nesterov.
- [2] - Lower bounds for finding stationary points II: first-order methods, Y. Carmon, J.C. Duchi, O. Hinder, A. Sidford.
- [3] - Convex optimization: Algorithms and complexity, S. Bubeck, others.
- [4] - Optimizing the efficiency of first-order methods for decreasing the gradient of smooth convex functions D. Kim, J.A. Fessler.
- [5] - Lower bounds for finding stationary points I, Y. Carmon, J.C. Duchi, O. Hinder, A. Sidford.
- [6] - Optimizing the efficiency of first-order methods for decreasing the gradient of smooth convex functions, D. Kim, J.A. Fessler.
- [7] - First-order methods in optimization, A. Beck. SIAM. 2017.
- [8] - Stochastic subgradient method converges at the rate $ O (k^{-1/4}) $ on weakly convex functions, D. Davis, D. Drusvyatskiy.
- [9] - On the lower bound of minimizing Polyak-Łojasiewicz functions, P. Yue, C. Fang, Z. Lin.
- [10] - Linear convergence of gradient and proximal-gradient methods under the Polyak-Łojasiewicz condition, H. Karimi, J. Nutini, M. Schmidt.

Обозначения:

- Зазор оптимальности: $f(x_k) - f^* \leq \varepsilon$
- Стационарность: $\|\nabla f(x_k)\| \leq \varepsilon$
- Квази-стационарность: $\|\nabla f_\lambda(x_k)\| \leq \varepsilon$, где $f_\lambda(x) = \inf\limits_{y\in\mathbb{R}^n}\left(f(y) + \frac{1}{2\lambda}\|y-x\|^2\right)$
- Липшицевость функции: $|f(x) - f(y)| \leq G\|x-y\| \forall x, y \in \mathbb{R}^n$
- Липшицевость градиента (L-гладкость): $\|\nabla f(x) - \nabla f(y)\| \leq L\|x-y\| \forall x, y \in \mathbb{R}^n$
- μ-сильная выпуклость: $f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda)f(y) - \frac{\mu}{2}\lambda(1-\lambda)\|x-y\|^2$
- ρ-слабо выпуклая функция: $f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda)f(y) + \rho\lambda(1-\lambda)\|x-y\|^2 \forall x, y \in \mathbb{R}^n$
- Число обусловленности: $\varkappa = \frac{L}{\mu}$
- Зазор в начальной точке: $f(x_0) - f^* \leq \Delta$
- Зазор по аргументу: $D = \|x_0 - x^*\|$

Бонус: нижние оценки для градиентных методов

Чёрный ящик

Итерация градиентного спуска:

$$x_{k+1} = x_k - \alpha_k \nabla f(x_k)$$

Чёрный ящик

Итерация градиентного спуска:

$$\begin{aligned} x_{k+1} &= x_k - \alpha_k \nabla f(x_k) \\ &= x_{k-1} - \alpha_{k-1} \nabla f(x_{k-1}) - \alpha_k \nabla f(x_k) \end{aligned}$$

Чёрный ящик

Итерация градиентного спуска:

$$\begin{aligned} x_{k+1} &= x_k - \alpha_k \nabla f(x_k) \\ &= x_{k-1} - \alpha_{k-1} \nabla f(x_{k-1}) - \alpha_k \nabla f(x_k) \\ &\vdots \end{aligned}$$

Чёрный ящик

Итерация градиентного спуска:

$$
\begin{aligned}
x_{k+1} &= x_k - \alpha_k \nabla f(x_k) \\
&= x_{k-1} - \alpha_{k-1} \nabla f(x_{k-1}) - \alpha_k \nabla f(x_k) \\
&\vdots \\
&= x_0 - \sum_{i=0}^{k} \alpha_{k-i} \nabla f(x_{k-i})
\end{aligned}
$$

Чёрный ящик

Итерация градиентного спуска:

$$\begin{aligned} x_{k+1} &= x_k - \alpha_k \nabla f(x_k) \\ &= x_{k-1} - \alpha_{k-1}\nabla f(x_{k-1}) - \alpha_k \nabla f(x_k) \\ &\vdots \\ &= x_0 - \sum_{i=0}^{k} \alpha_{k-i} \nabla f(x_{k-i}) \end{aligned}$$

Рассмотрим семейство методов первого порядка, где

$$\begin{aligned} &x_{k+1} \in x_0 + \mathsf{Lin}\left\{\nabla f(x_0), \nabla f(x_1), \dots, \nabla f(x_k)\right\} && f \text{ — гладкая} \\ &x_{k+1} \in x_0 + \mathsf{Lin}\left\{g_0, g_1, \dots, g_k\right\}, \text{ где } g_i \in \partial f(x_i) && f \text{ — негладкая} \end{aligned} \tag{14}$$

Чёрный ящик

Итерация градиентного спуска:

$$\begin{aligned} x_{k+1} &= x_k - \alpha_k \nabla f(x_k) \\ &= x_{k-1} - \alpha_{k-1}\nabla f(x_{k-1}) - \alpha_k \nabla f(x_k) \\ &\vdots \\ &= x_0 - \sum_{i=0}^{k} \alpha_{k-i} \nabla f(x_{k-i}) \end{aligned}$$

Рассмотрим семейство методов первого порядка, где

$$\begin{aligned} &x_{k+1} \in x_0 + \mathsf{Lin}\left\{\nabla f(x_0), \nabla f(x_1), \dots, \nabla f(x_k)\right\} && f \text{ — гладкая} \\ &x_{k+1} \in x_0 + \mathsf{Lin}\left\{g_0, g_1, \dots, g_k\right\}, \text{ где } g_i \in \partial f(x_i) && f \text{ — негладкая} \end{aligned} \tag{14}$$

Чтобы построить нижнюю оценку, нам нужно найти функцию f из соответствующего класса, такую, что любой метод из семейства (14) будет работать не быстрее этой нижней оценки.

Гладкий случай

Theorem

Существует L-гладкая и выпуклая функция f, такая, что любой метод (14) для всех k, $1 \leq k \leq \frac{n-1}{2}$, удовлетворяет:

$$f(x_k) - f^* \geq \frac{3L\|x_0 - x^*\|_2^2}{32(k+1)^2}$$

Гладкий случай

Theorem

Существует L-гладкая и выпуклая функция f, такая, что любой метод (14) для всех k, $1 \leq k \leq \frac{n-1}{2}$, удовлетворяет:

$$f(x_k) - f^* \geq \frac{3L\|x_0 - x^*\|_2^2}{32(k+1)^2}$$

- Какой бы метод из семейства методов первого порядка вы ни использовали, найдётся функция f, на которой скорость сходимости не лучше $\mathcal{O}\left(\frac{1}{k^2}\right)$.

Гладкий случай

Theorem

Существует L-гладкая и выпуклая функция f, такая, что любой метод (14) для всех k, $1 \leq k \leq \frac{n-1}{2}$, удовлетворяет:

$$f(x_k) - f^* \geq \frac{3L\|x_0 - x^*\|_2^2}{32(k+1)^2}$$

- Какой бы метод из семейства методов первого порядка вы ни использовали, найдётся функция f, на которой скорость сходимости не лучше $\mathcal{O}\left(\frac{1}{k^2}\right)$.
- Ключом к доказательству является явное построение специальной функции f.

Гладкий случай

Theorem

Существует L-гладкая и выпуклая функция f, такая, что любой метод (14) для всех k, $1 \leq k \leq \frac{n-1}{2}$, удовлетворяет:

$$f(x_k) - f^* \geq \frac{3L\|x_0 - x^*\|_2^2}{32(k+1)^2}$$

- Какой бы метод из семейства методов первого порядка вы ни использовали, найдётся функция f, на которой скорость сходимости не лучше $\mathcal{O}\left(\frac{1}{k^2}\right)$.
- Ключом к доказательству является явное построение специальной функции f.
- Обратите внимание, что эта граница $\mathcal{O}\left(\frac{1}{k^2}\right)$ не соответствует скорости градиентного спуска $\mathcal{O}\left(\frac{1}{k}\right)$. Два возможных варианта:

Гладкий случай

Theorem

Существует L-гладкая и выпуклая функция f, такая, что любой метод (14) для всех k, $1 \leq k \leq \frac{n-1}{2}$, удовлетворяет:

$$f(x_k) - f^* \geq \frac{3L\|x_0 - x^*\|_2^2}{32(k+1)^2}$$

- Какой бы метод из семейства методов первого порядка вы ни использовали, найдётся функция f, на которой скорость сходимости не лучше $\mathcal{O}\left(\frac{1}{k^2}\right)$.
- Ключом к доказательству является явное построение специальной функции f.
- Обратите внимание, что эта граница $\mathcal{O}\left(\frac{1}{k^2}\right)$ не соответствует скорости градиентного спуска $\mathcal{O}\left(\frac{1}{k}\right)$. Два возможных варианта:
  a. Нижняя оценка не является точной.

Гладкий случай

Theorem

Существует L-гладкая и выпуклая функция f, такая, что любой метод (14) для всех k, $1 \leq k \leq \frac{n-1}{2}$, удовлетворяет:

$$f(x_k) - f^* \geq \frac{3L\|x_0 - x^*\|_2^2}{32(k+1)^2}$$

- Какой бы метод из семейства методов первого порядка вы ни использовали, найдётся функция f, на которой скорость сходимости не лучше $\mathcal{O}\left(\frac{1}{k^2}\right)$.
- Ключом к доказательству является явное построение специальной функции f.
- Обратите внимание, что эта граница $\mathcal{O}\left(\frac{1}{k^2}\right)$ не соответствует скорости градиентного спуска $\mathcal{O}\left(\frac{1}{k}\right)$. Два возможных варианта:
  a. Нижняя оценка не является точной.
  b. Метод градиентного спуска не является оптимальным для этой задачи.

Гладкий случай

Theorem

Существует L-гладкая и выпуклая функция f, такая, что любой метод (14) для всех k, $1 \leq k \leq \frac{n-1}{2}$, удовлетворяет:

$$f(x_k) - f^* \geq \frac{3L\|x_0 - x^*\|_2^2}{32(k+1)^2}$$

- Какой бы метод из семейства методов первого порядка вы ни использовали, найдётся функция f, на которой скорость сходимости не лучше $\mathcal{O}\left(\frac{1}{k^2}\right)$.
- Ключом к доказательству является явное построение специальной функции f.
- Обратите внимание, что эта граница $\mathcal{O}\left(\frac{1}{k^2}\right)$ не соответствует скорости градиентного спуска $\mathcal{O}\left(\frac{1}{k}\right)$. Два возможных варианта:
  a. Нижняя оценка не является точной.
  b. **Метод градиентного спуска не является оптимальным для этой задачи.**

Наихудшая функция Нестерова

- Пусть $n = 2k + 1$ и $A \in \mathbb{R}^{n \times n}$.

$$A = \begin{bmatrix} 2 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 \\ 0 & 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 2 \end{bmatrix}$$

Наихудшая функция Нестерова

- Пусть $n = 2k + 1$ и $A \in \mathbb{R}^{n \times n}$.

$$A = \begin{bmatrix} 2 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 \\ 0 & 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 2 \end{bmatrix}$$

- Обратите внимание, что

$$x^T A x = x_1^2 + x_n^2 + \sum_{i=1}^{n-1} (x_i - x_{i+1})^2,$$

Следовательно, $x^T A x \geq 0$. Также легко увидеть, что $0 \preceq A \preceq 4I$.

Наихудшая функция Нестерова

- Пусть $n = 2k + 1$ и $A \in \mathbb{R}^{n \times n}$.

$$A = \begin{bmatrix} 2 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 \\ 0 & 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 2 \end{bmatrix}$$

- Обратите внимание, что

$$x^T A x = x_1^2 + x_n^2 + \sum_{i=1}^{n-1} (x_i - x_{i+1})^2,$$

Следовательно, $x^T A x \geq 0$. Также легко увидеть, что $0 \preceq A \preceq 4I$.

Наихудшая функция Нестерова

- Пусть $n = 2k + 1$ и $A \in \mathbb{R}^{n \times n}$.

$$A = \begin{bmatrix} 2 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 \\ 0 & 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 2 \end{bmatrix}$$

- Обратите внимание, что

$$x^T A x = x_1^2 + x_n^2 + \sum_{i=1}^{n-1} (x_i - x_{i+1})^2,$$

Следовательно, $x^T A x \geq 0$. Также легко увидеть, что $0 \preceq A \preceq 4I$.

Пример, когда $n = 3$:

$$A = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$$

Наихудшая функция Нестерова

- Пусть $n = 2k + 1$ и $A \in \mathbb{R}^{n \times n}$.

$$A = \begin{bmatrix} 2 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 \\ 0 & 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 2 \end{bmatrix}$$

- Обратите внимание, что

$$x^T A x = x_1^2 + x_n^2 + \sum_{i=1}^{n-1} (x_i - x_{i+1})^2,$$

Следовательно, $x^T A x \geq 0$. Также легко увидеть, что $0 \preceq A \preceq 4I$.

Пример, когда $n = 3$:

$$A = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$$

Нижняя оценка:

$$\begin{aligned} x^T A x &= 2x_1^2 + 2x_2^2 + 2x_3^2 - 2x_1 x_2 - 2x_2 x_3 \\ &= x_1^2 + x_1^2 - 2x_1 x_2 + x_2^2 + x_2^2 - 2x_2 x_3 + x_3^2 + x_3^2 \\ &= x_1^2 + (x_1 - x_2)^2 + (x_2 - x_3)^2 + x_3^2 \geq 0 \end{aligned}$$

Наихудшая функция Нестерова

- Пусть $n = 2k + 1$ и $A \in \mathbb{R}^{n \times n}$.

$$A = \begin{bmatrix} 2 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 \\ 0 & 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 2 \end{bmatrix}$$

- Обратите внимание, что

$$x^T A x = x_1^2 + x_n^2 + \sum_{i=1}^{n-1} (x_i - x_{i+1})^2,$$

Следовательно, $x^T A x \geq 0$. Также легко увидеть, что $0 \preceq A \preceq 4I$.

Пример, когда $n = 3$:

$$A = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$$

Нижняя оценка:

$$\begin{aligned} x^T A x &= 2x_1^2 + 2x_2^2 + 2x_3^2 - 2x_1 x_2 - 2x_2 x_3 \\ &= x_1^2 + x_1^2 - 2x_1 x_2 + x_2^2 + x_2^2 - 2x_2 x_3 + x_3^2 + x_3^2 \\ &= x_1^2 + (x_1 - x_2)^2 + (x_2 - x_3)^2 + x_3^2 \geq 0 \end{aligned}$$

Верхняя оценка

$$x^T A x = 2x_1^2 + 2x_2^2 + 2x_3^2 - 2x_1 x_2 - 2x_2 x_3$$

Наихудшая функция Нестерова

- Пусть $n = 2k + 1$ и $A \in \mathbb{R}^{n \times n}$.

$$A = \begin{bmatrix} 2 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 \\ 0 & 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 2 \end{bmatrix}$$

- Обратите внимание, что

$$x^T A x = x_1^2 + x_n^2 + \sum_{i=1}^{n-1} (x_i - x_{i+1})^2,$$

Следовательно, $x^T A x \geq 0$. Также легко увидеть, что $0 \preceq A \preceq 4I$.

Пример, когда $n = 3$:

$$A = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$$

Нижняя оценка:

$$\begin{aligned} x^T A x &= 2x_1^2 + 2x_2^2 + 2x_3^2 - 2x_1 x_2 - 2x_2 x_3 \\ &= x_1^2 + x_1^2 - 2x_1 x_2 + x_2^2 + x_2^2 - 2x_2 x_3 + x_3^2 + x_3^2 \\ &= x_1^2 + (x_1 - x_2)^2 + (x_2 - x_3)^2 + x_3^2 \geq 0 \end{aligned}$$

Верхняя оценка

$$\begin{aligned} x^T A x &= 2x_1^2 + 2x_2^2 + 2x_3^2 - 2x_1 x_2 - 2x_2 x_3 \\ &\leq 4(x_1^2 + x_2^2 + x_3^2) \end{aligned}$$

Наихудшая функция Нестерова

- Пусть $n = 2k + 1$ и $A \in \mathbb{R}^{n \times n}$.

$$A = \begin{bmatrix} 2 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 \\ 0 & 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 2 \end{bmatrix}$$

- Обратите внимание, что

$$x^T A x = x_1^2 + x_n^2 + \sum_{i=1}^{n-1} (x_i - x_{i+1})^2,$$

Следовательно, $x^T A x \geq 0$. Также легко увидеть, что $0 \preceq A \preceq 4I$.

Пример, когда $n = 3$:

$$A = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$$

Нижняя оценка:

$$\begin{aligned} x^T A x &= 2x_1^2 + 2x_2^2 + 2x_3^2 - 2x_1 x_2 - 2x_2 x_3 \\ &= x_1^2 + x_1^2 - 2x_1 x_2 + x_2^2 + x_2^2 - 2x_2 x_3 + x_3^2 + x_3^2 \\ &= x_1^2 + (x_1 - x_2)^2 + (x_2 - x_3)^2 + x_3^2 \geq 0 \end{aligned}$$

Верхняя оценка

$$\begin{aligned} x^T A x &= 2x_1^2 + 2x_2^2 + 2x_3^2 - 2x_1 x_2 - 2x_2 x_3 \\ &\leq 4(x_1^2 + x_2^2 + x_3^2) \\ 0 &\leq 2x_1^2 + 2x_2^2 + 2x_3^2 + 2x_1 x_2 + 2x_2 x_3 \end{aligned}$$

Наихудшая функция Нестерова

- Пусть $n = 2k + 1$ и $A \in \mathbb{R}^{n \times n}$.

$$A = \begin{bmatrix} 2 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 \\ 0 & 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 2 \end{bmatrix}$$

- Обратите внимание, что

$$x^T A x = x_1^2 + x_n^2 + \sum_{i=1}^{n-1} (x_i - x_{i+1})^2,$$

Следовательно, $x^T A x \geq 0$. Также легко увидеть, что $0 \preceq A \preceq 4I$.

Пример, когда $n = 3$:

$$A = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$$

Нижняя оценка:

$$\begin{aligned} x^T A x &= 2x_1^2 + 2x_2^2 + 2x_3^2 - 2x_1 x_2 - 2x_2 x_3 \\ &= x_1^2 + x_1^2 - 2x_1 x_2 + x_2^2 + x_2^2 - 2x_2 x_3 + x_3^2 + x_3^2 \\ &= x_1^2 + (x_1 - x_2)^2 + (x_2 - x_3)^2 + x_3^2 \geq 0 \end{aligned}$$

Верхняя оценка

$$\begin{aligned} x^T A x &= 2x_1^2 + 2x_2^2 + 2x_3^2 - 2x_1 x_2 - 2x_2 x_3 \\ &\leq 4(x_1^2 + x_2^2 + x_3^2) \\ 0 &\leq 2x_1^2 + 2x_2^2 + 2x_3^2 + 2x_1 x_2 + 2x_2 x_3 \\ 0 &\leq x_1^2 + x_1^2 + 2x_1 x_2 + x_2^2 + x_2^2 + 2x_2 x_3 + x_3^2 + x_3^2 \end{aligned}$$

Наихудшая функция Нестерова

- Пусть $n = 2k + 1$ и $A \in \mathbb{R}^{n \times n}$.

$$A = \begin{bmatrix} 2 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 \\ 0 & 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 2 \end{bmatrix}$$

- Обратите внимание, что

$$x^T A x = x_1^2 + x_n^2 + \sum_{i=1}^{n-1} (x_i - x_{i+1})^2,$$

Следовательно, $x^T A x \geq 0$. Также легко увидеть, что $0 \preceq A \preceq 4I$.

Пример, когда $n = 3$:

$$A = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$$

Нижняя оценка:

$$\begin{aligned} x^T A x &= 2x_1^2 + 2x_2^2 + 2x_3^2 - 2x_1 x_2 - 2x_2 x_3 \\ &= x_1^2 + x_1^2 - 2x_1 x_2 + x_2^2 + x_2^2 - 2x_2 x_3 + x_3^2 + x_3^2 \\ &= x_1^2 + (x_1 - x_2)^2 + (x_2 - x_3)^2 + x_3^2 \geq 0 \end{aligned}$$

Верхняя оценка

$$\begin{aligned} x^T A x &= 2x_1^2 + 2x_2^2 + 2x_3^2 - 2x_1 x_2 - 2x_2 x_3 \\ &\leq 4(x_1^2 + x_2^2 + x_3^2) \\ 0 &\leq 2x_1^2 + 2x_2^2 + 2x_3^2 + 2x_1 x_2 + 2x_2 x_3 \\ 0 &\leq x_1^2 + x_1^2 + 2x_1 x_2 + x_2^2 + x_2^2 + 2x_2 x_3 + x_3^2 + x_3^2 \\ 0 &\leq x_1^2 + (x_1 + x_2)^2 + (x_2 + x_3)^2 + x_3^2 \end{aligned}$$

Наихудшая функция Нестерова

- Определим следующую L-гладкую выпуклую функцию: $f(x) = \frac{L}{4}\left(\frac{1}{2}x^TAx - e_1^Tx\right) = \frac{L}{8}x^TAx - \frac{L}{4}e_1^Tx$.

Наихудшая функция Нестерова

- Определим следующую L-гладкую выпуклую функцию: $f(x) = \frac{L}{4}\left(\frac{1}{2}x^TAx - e_1^Tx\right) = \frac{L}{8}x^TAx - \frac{L}{4}e_1^Tx.$
- Оптимальное решение x^* удовлетворяет $Ax^* = e_1$, и решение этой системы уравнений дает:

$$\begin{bmatrix} 2 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 \\ 0 & 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 2 \end{bmatrix} \begin{bmatrix} x_1^* \\ x_2^* \\ x_3^* \\ \vdots \\ x_n^* \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad \begin{cases} 2x_1^* - x_2^* = 1 \\ -x_{i-1}^* + 2x_i^* - x_{i+1}^* = 0, \ i = 2, \ldots, n-1 \\ -x_{n-1}^* + 2x_n^* = 0 \end{cases}$$

Наихудшая функция Нестерова

- Определим следующую L-гладкую выпуклую функцию: $f(x) = \frac{L}{4}\left(\frac{1}{2}x^T Ax - e_1^T x\right) = \frac{L}{8}x^T Ax - \frac{L}{4}e_1^T x$.
- Оптимальное решение x^* удовлетворяет $Ax^* = e_1$, и решение этой системы уравнений дает:

$$\begin{bmatrix} 2 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 \\ 0 & 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 2 \end{bmatrix} \begin{bmatrix} x_1^* \\ x_2^* \\ x_3^* \\ \vdots \\ x_n^* \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad \begin{cases} 2x_1^* - x_2^* = 1 \\ -x_{i-1}^* + 2x_i^* - x_{i+1}^* = 0, \ i = 2, \ldots, n-1 \\ -x_{n-1}^* + 2x_n^* = 0 \end{cases}$$

- Гипотеза: $x_i^* = a + bi$ (вдохновлённая физикой). Проверьте, что выполнено второе уравнение, в то время как a и b вычисляются из первого и последнего уравнений.

Наихудшая функция Нестерова

- Определим следующую L-гладкую выпуклую функцию: $f(x) = \frac{L}{4}\left(\frac{1}{2}x^TAx - e_1^Tx\right) = \frac{L}{8}x^TAx - \frac{L}{4}e_1^Tx$.
- Оптимальное решение x^* удовлетворяет $Ax^* = e_1$, и решение этой системы уравнений дает:

$$\begin{bmatrix} 2 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 \\ 0 & 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 2 \end{bmatrix} \begin{bmatrix} x_1^* \\ x_2^* \\ x_3^* \\ \vdots \\ x_n^* \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad \begin{cases} 2x_1^* - x_2^* = 1 \\ -x_{i-1}^* + 2x_i^* - x_{i+1}^* = 0, \ i = 2, \ldots, n-1 \\ -x_{n-1}^* + 2x_n^* = 0 \end{cases}$$

- Гипотеза: $x_i^* = a + bi$ (вдохновлённая физикой). Проверьте, что выполнено второе уравнение, в то время как a и b вычисляются из первого и последнего уравнений.
- Решение:

$$x_i^* = 1 - \frac{i}{n+1},$$

Наихудшая функция Нестерова

- Определим следующую L-гладкую выпуклую функцию: $f(x) = \frac{L}{4}\left(\frac{1}{2}x^TAx - e_1^Tx\right) = \frac{L}{8}x^TAx - \frac{L}{4}e_1^Tx$.
- Оптимальное решение x^* удовлетворяет $Ax^* = e_1$, и решение этой системы уравнений дает:

$$\begin{bmatrix} 2 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 \\ 0 & 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 2 \end{bmatrix} \begin{bmatrix} x_1^* \\ x_2^* \\ x_3^* \\ \vdots \\ x_n^* \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad \begin{cases} 2x_1^* - x_2^* = 1 \\ -x_{i-1}^* + 2x_i^* - x_{i+1}^* = 0, \ i = 2, \ldots, n-1 \\ -x_{n-1}^* + 2x_n^* = 0 \end{cases}$$

- Гипотеза: $x_i^* = a + bi$ (вдохновлённая физикой). Проверьте, что выполнено второе уравнение, в то время как a и b вычисляются из первого и последнего уравнений.
- Решение:

$$x_i^* = 1 - \frac{i}{n+1},$$

- И значение функции равно

$$f(x^*) = \frac{L}{8}x^{*T}Ax^* - \frac{L}{4}\langle x^*, e_1\rangle = -\frac{L}{8}\langle x^*, e_1\rangle = -\frac{L}{8}\left(1 - \frac{1}{n+1}\right).$$

Гладкий случай (доказательство)

- Предположим, что мы начинаем с $x_0 = 0$. Запросив у оракула градиент, мы получаем $g_0 = -\frac{L}{4} e_1$. Тогда, x_1 должен лежать на линии, генерируемой e_1. В этой точке все компоненты x_1 равны нулю, кроме первой, поэтому

$$x_1 = \begin{bmatrix} \bullet \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Гладкий случай (доказательство)

- Предположим, что мы начинаем с $x_0 = 0$. Запросив у оракула градиент, мы получаем $g_0 = -\frac{L}{4} e_1$. Тогда, x_1 должен лежать на линии, генерируемой e_1. В этой точке все компоненты x_1 равны нулю, кроме первой, поэтому

$$x_1 = \begin{bmatrix} \bullet \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

- На второй итерации оракул возвращает градиент $g_1 = \frac{L}{4}(Ax_1 - e_1)$. Тогда, x_2 должен лежать на линии, генерируемой e_1 и $Ax_1 - e_1$. Все компоненты x_2 равны нулю, кроме первых двух, поэтому

$$\begin{bmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \cdots & 0 \\ 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 2 \end{bmatrix} \begin{bmatrix} \bullet \\ 0 \\ \vdots \\ 0 \end{bmatrix} \Rightarrow x_2 = \begin{bmatrix} \bullet \\ \bullet \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Гладкий случай (доказательство)

- Предположим, что мы начинаем с $x_0 = 0$. Запросив у оракула градиент, мы получаем $g_0 = -\frac{L}{4} e_1$. Тогда, x_1 должен лежать на линии, генерируемой e_1. В этой точке все компоненты x_1 равны нулю, кроме первой, поэтому

$$x_1 = \begin{bmatrix} \bullet \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

- На второй итерации оракул возвращает градиент $g_1 = \frac{L}{4}(Ax_1 - e_1)$. Тогда, x_2 должен лежать на линии, генерируемой e_1 и $Ax_1 - e_1$. Все компоненты x_2 равны нулю, кроме первых двух, поэтому

$$\begin{bmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \cdots & 0 \\ 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 2 \end{bmatrix} \begin{bmatrix} \bullet \\ 0 \\ \vdots \\ 0 \end{bmatrix} \Rightarrow x_2 = \begin{bmatrix} \bullet \\ \bullet \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

- Из-за структуры матрицы A можно показать, что после k итераций все последние $n - k$ компоненты x_k равны нулю.

$$x_k = \begin{bmatrix} \bullet & 1 \\ \bullet & 2 \\ \vdots & \vdots \\ \bullet & k \\ 0 & k+1 \\ \vdots & \vdots \\ 0 & n \end{bmatrix}$$

Гладкий случай (доказательство)

- Предположим, что мы начинаем с $x_0 = 0$. Запросив у оракула градиент, мы получаем $g_0 = -\frac{L}{4} e_1$. Тогда, x_1 должен лежать на линии, генерируемой e_1. В этой точке все компоненты x_1 равны нулю, кроме первой, поэтому

$$x_1 = \begin{bmatrix} \bullet \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

- На второй итерации оракул возвращает градиент $g_1 = \frac{L}{4}(Ax_1 - e_1)$. Тогда, x_2 должен лежать на линии, генерируемой e_1 и $Ax_1 - e_1$. Все компоненты x_2 равны нулю, кроме первых двух, поэтому

$$\begin{bmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \cdots & 0 \\ 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 2 \end{bmatrix} \begin{bmatrix} \bullet \\ 0 \\ \vdots \\ 0 \end{bmatrix} \Rightarrow x_2 = \begin{bmatrix} \bullet \\ \bullet \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

- Из-за структуры матрицы A можно показать, что после k итераций все последние $n - k$ компоненты x_k равны нулю.

$$x_k = \begin{bmatrix} \bullet & 1 \\ \bullet & 2 \\ \vdots & \vdots \\ \bullet & k \\ 0 & k+1 \\ \vdots & \vdots \\ 0 & n \end{bmatrix}$$

- Однако, поскольку каждая итерация x_k, произведенная нашим методом, лежит в $S_k = \mathrm{Lin}\{e_1, e_2, \ldots, e_k\}$ (т.е. имеет нули в координатах $k+1, \ldots, n$), она не может "достичь" полного оптимального вектора x^*. Другими словами, даже если бы мы выбрали лучший возможный вектор из S_k, обозначаемый

$$\tilde{x}_k = \arg\min_{x \in S_k} f(x),$$

значение функции в нём $f(\tilde{x}_k)$ будет выше, чем $f(x^*)$.

Гладкий случай (доказательство)

- Поскольку $x_k \in S_k = \text{Lin}\{e_1, e_2, \ldots, e_k\}$ и $\tilde{x}_k$ является лучшим возможным приближением к x^* в S_k, мы имеем

$$f(x_k) \geq f(\tilde{x}_k).$$

Гладкий случай (доказательство)

- Поскольку $x_k \in S_k = \text{Lin}\{e_1, e_2, \ldots, e_k\}$ и $\tilde{x}_k$ является лучшим возможным приближением к x^* в S_k, мы имеем
$$f(x_k) \geq f(\tilde{x}_k).$$
- Следовательно,
$$f(x_k) - f(x^*) \geq f(\tilde{x}_k) - f(x^*).$$

Гладкий случай (доказательство)

- Поскольку $x_k \in S_k = \text{Lin}\{e_1, e_2, \dots, e_k\}$ и $\widetilde{x}_k$ является лучшим возможным приближением к x^* в S_k, мы имеем
$$f(x_k) \geq f(\widetilde{x}_k).$$
- Следовательно,
$$f(x_k) - f(x^*) \geq f(\widetilde{x}_k) - f(x^*).$$
- Аналогично, для оптимума исходной функции, мы имеем $\widetilde{x}_{k_{(i)}} = 1 - \frac{i}{k+1}$ и $f(\widetilde{x}_k) = -\frac{L}{8}\left(1 - \frac{1}{k+1}\right)$.

Гладкий случай (доказательство)

- Поскольку $x_k \in S_k = \text{Lin}\{e_1, e_2, \dots, e_k\}$ и $\tilde{x}_k$ является лучшим возможным приближением к x^* в S_k, мы имеем
$$f(x_k) \geq f(\tilde{x}_k).$$
- Следовательно,
$$f(x_k) - f(x^*) \geq f(\tilde{x}_k) - f(x^*).$$
- Аналогично, для оптимума исходной функции, мы имеем $\tilde{x}_{k_{(i)}} = 1 - \frac{i}{k+1}$ и $f(\tilde{x}_k) = -\frac{L}{8}\left(1 - \frac{1}{k+1}\right)$.
- Теперь мы имеем:
$$f(x_k) - f(x^*) \geq f(\tilde{x}_k) - f(x^*)$$

(15)

Гладкий случай (доказательство)

- Поскольку $x_k \in S_k = \text{Lin}\{e_1, e_2, \ldots, e_k\}$ и $\tilde{x}_k$ является лучшим возможным приближением к x^* в S_k, мы имеем
$$f(x_k) \geq f(\tilde{x}_k).$$
- Следовательно,
$$f(x_k) - f(x^*) \geq f(\tilde{x}_k) - f(x^*).$$
- Аналогично, для оптимума исходной функции, мы имеем $\tilde{x}_{k_{(i)}} = 1 - \frac{i}{k+1}$ и $f(\tilde{x}_k) = -\frac{L}{8}\left(1 - \frac{1}{k+1}\right)$.
- Теперь мы имеем:
$$\begin{aligned} f(x_k) - f(x^*) &\geq f(\tilde{x}_k) - f(x^*) \\ &= -\frac{L}{8}\left(1 - \frac{1}{k+1}\right) - \left(-\frac{L}{8}\left(1 - \frac{1}{n+1}\right)\right) \end{aligned} \tag{15}$$

Гладкий случай (доказательство)

- Поскольку $x_k \in S_k = \text{Lin}\{e_1, e_2, \ldots, e_k\}$ и $\tilde{x}_k$ является лучшим возможным приближением к x^* в S_k, мы имеем
$$f(x_k) \geq f(\tilde{x}_k).$$
- Следовательно,
$$f(x_k) - f(x^*) \geq f(\tilde{x}_k) - f(x^*).$$
- Аналогично, для оптимума исходной функции, мы имеем $\tilde{x}_{k_{(i)}} = 1 - \frac{i}{k+1}$ и $f(\tilde{x}_k) = -\frac{L}{8}\left(1 - \frac{1}{k+1}\right)$.
- Теперь мы имеем:
$$\begin{aligned} f(x_k) - f(x^*) &\geq f(\tilde{x}_k) - f(x^*) \\ &= -\frac{L}{8}\left(1 - \frac{1}{k+1}\right) - \left(-\frac{L}{8}\left(1 - \frac{1}{n+1}\right)\right) \\ &= \frac{L}{8}\left(\frac{1}{k+1} - \frac{1}{n+1}\right) = \frac{L}{8}\left(\frac{n-k}{(k+1)(n+1)}\right) \end{aligned} \tag{15}$$

Гладкий случай (доказательство)

- Поскольку $x_k \in S_k = \text{Lin}\{e_1, e_2, \ldots, e_k\}$ и $\tilde{x}_k$ является лучшим возможным приближением к x^* в S_k, мы имеем
$$f(x_k) \geq f(\tilde{x}_k).$$
- Следовательно,
$$f(x_k) - f(x^*) \geq f(\tilde{x}_k) - f(x^*).$$
- Аналогично, для оптимума исходной функции, мы имеем $\tilde{x}_{k_{(i)}} = 1 - \frac{i}{k+1}$ и $f(\tilde{x}_k) = -\frac{L}{8}\left(1 - \frac{1}{k+1}\right)$.
- Теперь мы имеем:
$$\begin{aligned} f(x_k) - f(x^*) &\geq f(\tilde{x}_k) - f(x^*) \\ &= -\frac{L}{8}\left(1 - \frac{1}{k+1}\right) - \left(-\frac{L}{8}\left(1 - \frac{1}{n+1}\right)\right) \\ &= \frac{L}{8}\left(\frac{1}{k+1} - \frac{1}{n+1}\right) = \frac{L}{8}\left(\frac{n-k}{(k+1)(n+1)}\right) \\ &\overset{n=2k+1}{=} \frac{L}{16(k+1)} \end{aligned} \tag{15}$$

Гладкий случай (доказательство)

- Теперь мы ограничиваем $R = \|x_0 - x^*\|_2$:

$$\|x_0 - x^*\|_2^2 = \|0 - x^*\|_2^2 = \|x^*\|_2^2 = \sum_{i=1}^{n} \left(1 - \frac{i}{n+1}\right)^2$$

Гладкий случай (доказательство)

- Теперь мы ограничиваем $R = \|x_0 - x^*\|_2$:

$$\|x_0 - x^*\|_2^2 = \|0 - x^*\|_2^2 = \|x^*\|_2^2 = \sum_{i=1}^{n} \left(1 - \frac{i}{n+1}\right)^2$$

$$= n - \frac{2}{n+1} \sum_{i=1}^{n} i + \frac{1}{(n+1)^2} \sum_{i=1}^{n} i^2$$

Гладкий случай (доказательство)

- Теперь мы ограничиваем $R = \|x_0 - x^*\|_2$:

$$\begin{aligned}\|x_0 - x^*\|_2^2 = \|0 - x^*\|_2^2 = \|x^*\|_2^2 &= \sum_{i=1}^{n} \left(1 - \frac{i}{n+1}\right)^2 \\ &= n - \frac{2}{n+1}\sum_{i=1}^{n} i + \frac{1}{(n+1)^2}\sum_{i=1}^{n} i^2 \\ &\leq n - \frac{2}{n+1} \cdot \frac{n(n+1)}{2} + \frac{1}{(n+1)^2} \cdot \frac{(n+1)^3}{3}\end{aligned}$$

Гладкий случай (доказательство)

- Теперь мы ограничиваем $R = \|x_0 - x^*\|_2$:

$$\begin{aligned}
\|x_0 - x^*\|_2^2 = \|0 - x^*\|_2^2 = \|x^*\|_2^2 &= \sum_{i=1}^{n} \left(1 - \frac{i}{n+1}\right)^2 \\
&= n - \frac{2}{n+1} \sum_{i=1}^{n} i + \frac{1}{(n+1)^2} \sum_{i=1}^{n} i^2 \\
&\leq n - \frac{2}{n+1} \cdot \frac{n(n+1)}{2} + \frac{1}{(n+1)^2} \cdot \frac{(n+1)^3}{3} \\
&= \frac{n+1}{3} \overset{n=2k+1}{=} \frac{2(k+1)}{3}.
\end{aligned}$$

Гладкий случай (доказательство)

- Теперь мы ограничиваем $R = \|x_0 - x^*\|_2$:

$$\begin{aligned}\|x_0 - x^*\|_2^2 = \|0 - x^*\|_2^2 = \|x^*\|_2^2 &= \sum_{i=1}^{n} \left(1 - \frac{i}{n+1}\right)^2 \\ &= n - \frac{2}{n+1}\sum_{i=1}^{n} i + \frac{1}{(n+1)^2}\sum_{i=1}^{n} i^2 \\ &\leq n - \frac{2}{n+1} \cdot \frac{n(n+1)}{2} + \frac{1}{(n+1)^2} \cdot \frac{(n+1)^3}{3} \\ &= \frac{n+1}{3} \overset{n=2k+1}{=} \frac{2(k+1)}{3}.\end{aligned}$$

- Следовательно,

$$k + 1 \geq \frac{3}{2}\|x_0 - x^*\|_2^2 = \frac{3}{2}R^2 \tag{16}$$

Гладкий случай (доказательство)

- Теперь мы ограничиваем $R = \|x_0 - x^*\|_2$:

$$\begin{aligned}\|x_0 - x^*\|_2^2 = \|0 - x^*\|_2^2 = \|x^*\|_2^2 &= \sum_{i=1}^{n} \left(1 - \frac{i}{n+1}\right)^2 \\ &= n - \frac{2}{n+1}\sum_{i=1}^{n} i + \frac{1}{(n+1)^2}\sum_{i=1}^{n} i^2 \\ &\leq n - \frac{2}{n+1} \cdot \frac{n(n+1)}{2} + \frac{1}{(n+1)^2} \cdot \frac{(n+1)^3}{3} \\ &= \frac{n+1}{3} \overset{n=2k+1}{=} \frac{2(k+1)}{3}.\end{aligned}$$

- Следовательно,

$$k + 1 \geq \frac{3}{2}\|x_0 - x^*\|_2^2 = \frac{3}{2}R^2 \tag{16}$$

Гладкий случай (доказательство)

- Теперь мы ограничиваем $R = \|x_0 - x^*\|_2$:

$$\begin{aligned}\|x_0 - x^*\|_2^2 = \|0 - x^*\|_2^2 = \|x^*\|_2^2 &= \sum_{i=1}^{n} \left(1 - \frac{i}{n+1}\right)^2 \\ &= n - \frac{2}{n+1}\sum_{i=1}^{n} i + \frac{1}{(n+1)^2}\sum_{i=1}^{n} i^2 \\ &\leq n - \frac{2}{n+1} \cdot \frac{n(n+1)}{2} + \frac{1}{(n+1)^2} \cdot \frac{(n+1)^3}{3} \\ &= \frac{n+1}{3} \overset{n=2k+1}{=} \frac{2(k+1)}{3}.\end{aligned}$$

Заметим, что

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$$

$$\sum_{i=1}^{n} i^2 = \frac{n(n+1)(2n+1)}{6} \leq \frac{(n+1)^3}{3}$$

- Следовательно,

$$k + 1 \geq \frac{3}{2}\|x_0 - x^*\|_2^2 = \frac{3}{2}R^2 \tag{16}$$

Гладкий случай (доказательство)

Наконец, используя (15) и (16), мы получаем:

$$f(x_k) - f(x^*) \geq \frac{L}{16(k+1)} = \frac{L(k+1)}{16(k+1)^2}$$

Гладкий случай (доказательство)

Наконец, используя (15) и (16), мы получаем:

$$f(x_k) - f(x^*) \geq \frac{L}{16(k+1)} = \frac{L(k+1)}{16(k+1)^2}$$
$$\geq \frac{L}{16(k+1)^2}\frac{3}{2}R^2$$

Гладкий случай (доказательство)

Наконец, используя (15) и (16), мы получаем:

$$
\begin{aligned}
f(x_k) - f(x^*) &\geq \frac{L}{16(k+1)} = \frac{L(k+1)}{16(k+1)^2} \\
&\geq \frac{L}{16(k+1)^2}\frac{3}{2}R^2 \\
&= \frac{3LR^2}{32(k+1)^2}
\end{aligned}
$$

Гладкий случай (доказательство)

Наконец, используя (15) и (16), мы получаем:

$$\begin{aligned} f(x_k) - f(x^*) &\geq \frac{L}{16(k+1)} = \frac{L(k+1)}{16(k+1)^2} \\ &\geq \frac{L}{16(k+1)^2}\frac{3}{2}R^2 \\ &= \frac{3LR^2}{32(k+1)^2} \end{aligned}$$

Это завершает доказательство с желаемой скоростью $\mathcal{O}\left(\frac{1}{k^2}\right)$.

Нижние оценки для гладкого случая

Гладкий выпуклый случай

Существует L-гладкая выпуклая функция f, такая, что любой метод в форме 14 для всех k, $1 \leq k \leq \frac{n-1}{2}$, удовлетворяет:

$$f(x_k) - f^* \geq \frac{3L\|x_0 - x^*\|_2^2}{32(k+1)^2}$$

Гладкий сильно выпуклый случай

Для любого x_0 и любого $\mu > 0$, $\varkappa = \frac{L}{\mu} > 1$, существует L-гладкая и μ-сильно выпуклая функция f, такая, что для любого метода в форме 14 выполняются неравенства:

$$\|x_k - x^*\|_2 \geq \left(\frac{\sqrt{\varkappa}-1}{\sqrt{\varkappa}+1}\right)^k \|x_0 - x^*\|_2$$

$$f(x_k) - f^* \geq \frac{\mu}{2}\left(\frac{\sqrt{\varkappa}-1}{\sqrt{\varkappa}+1}\right)^{2k} \|x_0 - x^*\|_2^2$$

Бонус: ускорение для квадратичных функций

Результат сходимости для квадратичных функций

Предположим, что мы решаем задачу минимизации сильно выпуклой квадратичной функции, с помощью метода градиентного спуска:

$$f(x) = \frac{1}{2}x^T Ax - b^T x \qquad x_{k+1} = x_k - \alpha_k \nabla f(x_k).$$

Результат сходимости для квадратичных функций

Предположим, что мы решаем задачу минимизации сильно выпуклой квадратичной функции, с помощью метода градиентного спуска:

$$f(x) = \frac{1}{2}x^T Ax - b^T x \qquad x_{k+1} = x_k - \alpha_k \nabla f(x_k).$$

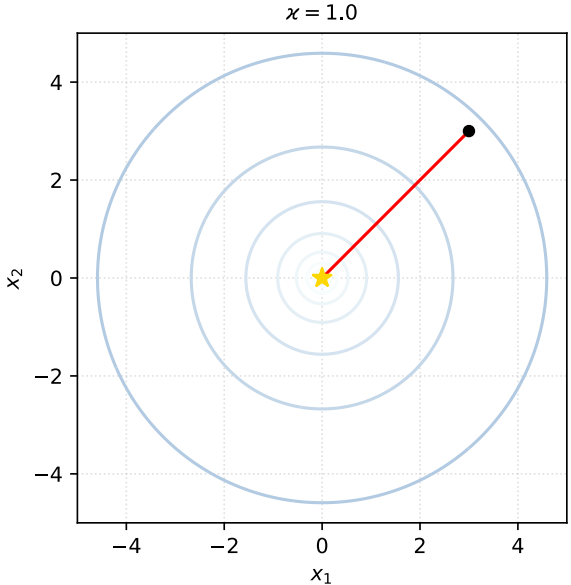
Theorem

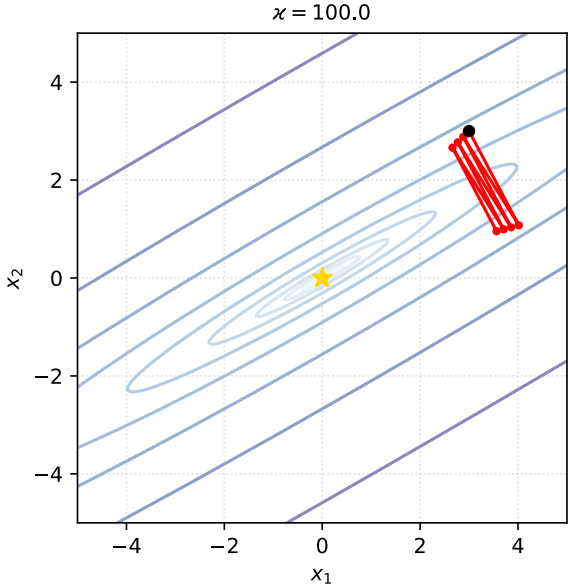
Градиентный спуск с шагом $\alpha_k = \frac{2}{\mu + L}$ сходится к оптимальному решению x^* со следующей гарантией:

$$\|x_{k+1} - x^*\|_2 \leq \left(\frac{\varkappa - 1}{\varkappa + 1}\right)^k \|x_0 - x^*\|_2 \qquad f(x_{k+1}) - f(x^*) \leq \left(\frac{\varkappa - 1}{\varkappa + 1}\right)^{2k} (f(x_0) - f(x^*))$$

где $\varkappa = \frac{L}{\mu}$ является числом обусловленности A.

Число обусловленности $\varkappa$





Ускорение из первых принципов

$$f(x) = \frac{1}{2}x^T Ax - b^T x \qquad x_{k+1} = x_k - \alpha_k \nabla f(x_k).$$

Пусть x^* будет единственным решением системы линейных уравнений $Ax = b$ и пусть $e_k = x_k - x^*$, где $x_{k+1} = x_k - \alpha_k(Ax_k - b)$ определяется рекурсивно, начиная с некоторого x_0, а α_k — шаг, который мы определим позже.

$$e_{k+1} = (I - \alpha_k A)e_k.$$

Ускорение из первых принципов

$$f(x) = \frac{1}{2}x^T Ax - b^T x \qquad x_{k+1} = x_k - \alpha_k \nabla f(x_k).$$

Пусть x^* будет единственным решением системы линейных уравнений $Ax = b$ и пусть $e_k = x_k - x^*$, где $x_{k+1} = x_k - \alpha_k(Ax_k - b)$ определяется рекурсивно, начиная с некоторого x_0, а α_k — шаг, который мы определим позже.

$$e_{k+1} = (I - \alpha_k A)e_k.$$

Полиномы

Вышеуказанный расчет дает нам $e_k = p_k(A)e_0$,
где p_k является полиномом

$$p_k(a) = \prod_{i=1}^{k}(1 - \alpha_i a).$$

Ускорение из первых принципов

$$f(x) = \frac{1}{2}x^T Ax - b^T x \qquad x_{k+1} = x_k - \alpha_k \nabla f(x_k).$$

Пусть x^* будет единственным решением системы линейных уравнений $Ax = b$ и пусть $e_k = x_k - x^*$, где $x_{k+1} = x_k - \alpha_k(Ax_k - b)$ определяется рекурсивно, начиная с некоторого x_0, а α_k — шаг, который мы определим позже.

$$e_{k+1} = (I - \alpha_k A)e_k.$$

Полиномы

Вышеуказанный расчет дает нам $e_k = p_k(A)e_0$,
где p_k является полиномом

$$p_k(a) = \prod_{i=1}^{k}(1 - \alpha_i a).$$

Мы можем ограничить норму ошибки как

$$\|e_k\| \leq \|p_k(A)\| \cdot \|e_0\|.$$

Ускорение из первых принципов

$$f(x) = \frac{1}{2}x^T Ax - b^T x \qquad x_{k+1} = x_k - \alpha_k \nabla f(x_k).$$

Пусть x^* будет единственным решением системы линейных уравнений $Ax = b$ и пусть $e_k = x_k - x^*$, где $x_{k+1} = x_k - \alpha_k(Ax_k - b)$ определяется рекурсивно, начиная с некоторого x_0, а α_k — шаг, который мы определим позже.

$$e_{k+1} = (I - \alpha_k A)e_k.$$

Полиномы

Вышеуказанный расчет дает нам $e_k = p_k(A)e_0$, где p_k является полиномом

$$p_k(a) = \prod_{i=1}^{k}(1 - \alpha_i a).$$

Мы можем ограничить норму ошибки как

$$\|e_k\| \leq \|p_k(A)\| \cdot \|e_0\| .$$

Поскольку A является симметричной матрицей с собственными значениями в $[\mu, L]$,:

$$\|p_k(A)\| \leq \max_{\mu \leq a \leq L} |p_k(a)| .$$

Это приводит к интересной постановке задачи: среди всех полиномов, удовлетворяющих $p_k(0) = 1$, мы ищем полином, значение которого как можно меньше отклоняется от нуля на интервале $[\mu, L]$.

Наивное полиномиальное решение

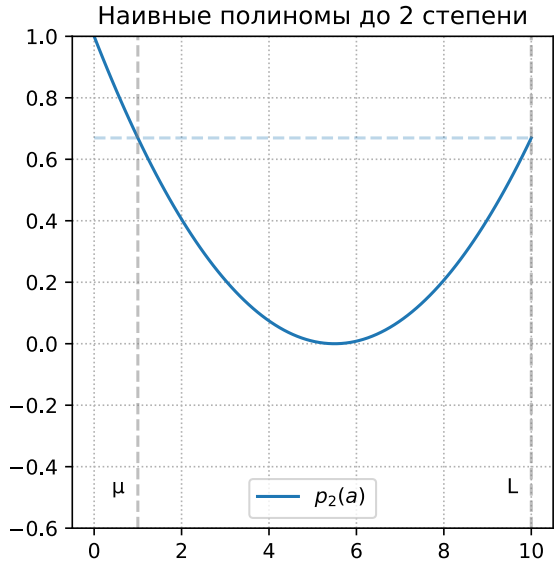
Наивное решение состоит в том, чтобы выбрать равномерный шаг $\alpha_k = \frac{2}{\mu+L}$. Благодаря этому $|p_k(\mu)| = |p_k(L)|$.

$$\|e_k\| \leq \left(\frac{\varkappa - 1}{\varkappa + 1}\right)^k \|e_0\|$$

Это точно та же скорость, которую мы доказали в предыдущей лекции для любой гладкой и сильно выпуклой функции.

Давайте посмотрим на этот полином поближе. На правом рисунке мы выбираем $\mu = 1$ и $L = 10$ так, что $\varkappa = 10$. Следовательно, соответствующий интервал равен $[1, 10]$.

Можем ли мы сделать лучше? Ответ — да.



Наивное полиномиальное решение

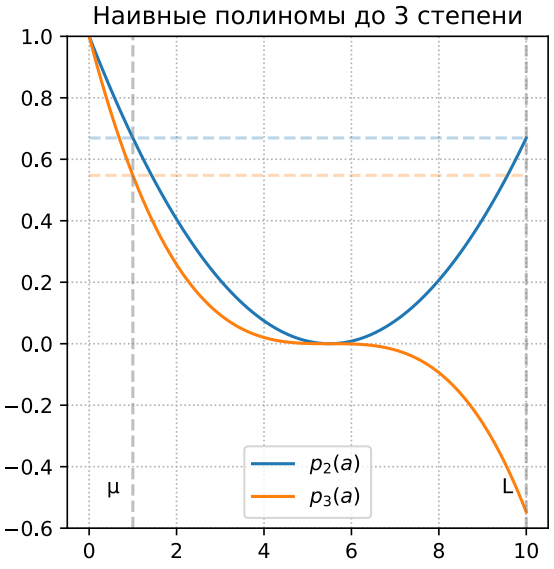
Наивное решение состоит в том, чтобы выбрать равномерный шаг $\alpha_k = \frac{2}{\mu+L}$. Благодаря этому $|p_k(\mu)| = |p_k(L)|$.

$$\|e_k\| \leq \left(\frac{\varkappa - 1}{\varkappa + 1}\right)^k \|e_0\|$$

Это точно та же скорость, которую мы доказали в предыдущей лекции для любой гладкой и сильно выпуклой функции.

Давайте посмотрим на этот полином поближе. На правом рисунке мы выбираем $\mu = 1$ и $L = 10$ так, что $\varkappa = 10$. Следовательно, соответствующий интервал равен $[1, 10]$.

Можем ли мы сделать лучше? Ответ — да.



Наивное полиномиальное решение

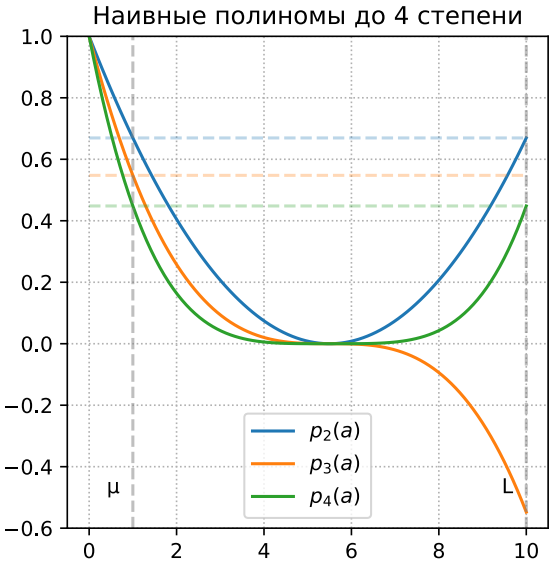
Наивное решение состоит в том, чтобы выбрать равномерный шаг $\alpha_k = \frac{2}{\mu+L}$. Благодаря этому $|p_k(\mu)| = |p_k(L)|$.

$$\|e_k\| \leq \left(\frac{\varkappa - 1}{\varkappa + 1}\right)^k \|e_0\|$$

Это точно та же скорость, которую мы доказали в предыдущей лекции для любой гладкой и сильно выпуклой функции.

Давайте посмотрим на этот полином поближе. На правом рисунке мы выбираем $\mu = 1$ и $L = 10$ так, что $\varkappa = 10$. Следовательно, соответствующий интервал равен $[1, 10]$.

Можем ли мы сделать лучше? Ответ — да.



Наивное полиномиальное решение

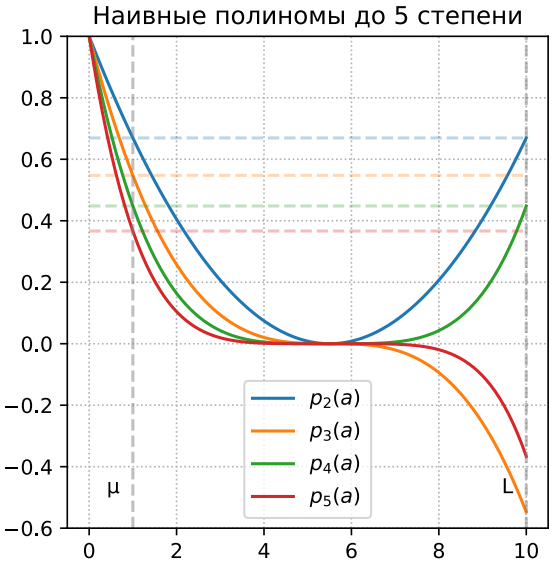
Наивное решение состоит в том, чтобы выбрать равномерный шаг $\alpha_k = \frac{2}{\mu+L}$. Благодаря этому $|p_k(\mu)| = |p_k(L)|$.

$$\|e_k\| \leq \left(\frac{\varkappa - 1}{\varkappa + 1}\right)^k \|e_0\|$$

Это точно та же скорость, которую мы доказали в предыдущей лекции для любой гладкой и сильно выпуклой функции.

Давайте посмотрим на этот полином поближе. На правом рисунке мы выбираем $\mu = 1$ и $L = 10$ так, что $\varkappa = 10$. Следовательно, соответствующий интервал равен $[1, 10]$.

Можем ли мы сделать лучше? Ответ — да.



Наивное полиномиальное решение

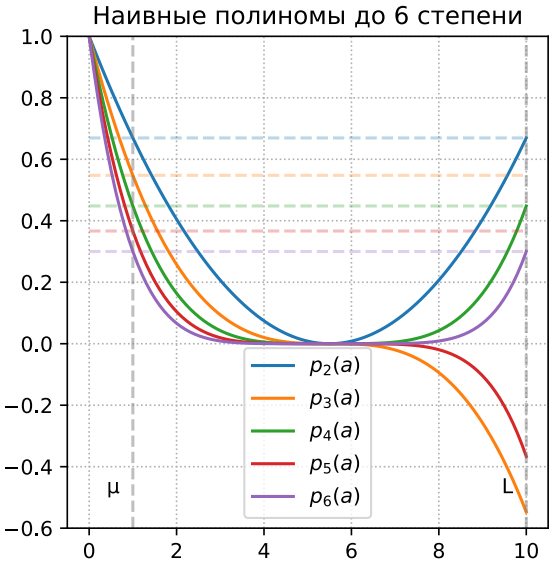
Наивное решение состоит в том, чтобы выбрать равномерный шаг $\alpha_k = \frac{2}{\mu+L}$. Благодаря этому $|p_k(\mu)| = |p_k(L)|$.

$$\|e_k\| \leq \left(\frac{\varkappa - 1}{\varkappa + 1}\right)^k \|e_0\|$$

Это точно та же скорость, которую мы доказали в предыдущей лекции для любой гладкой и сильно выпуклой функции.

Давайте посмотрим на этот полином поближе. На правом рисунке мы выбираем $\mu = 1$ и $L = 10$ так, что $\varkappa = 10$. Следовательно, соответствующий интервал равен $[1, 10]$.

Можем ли мы сделать лучше? Ответ — да.

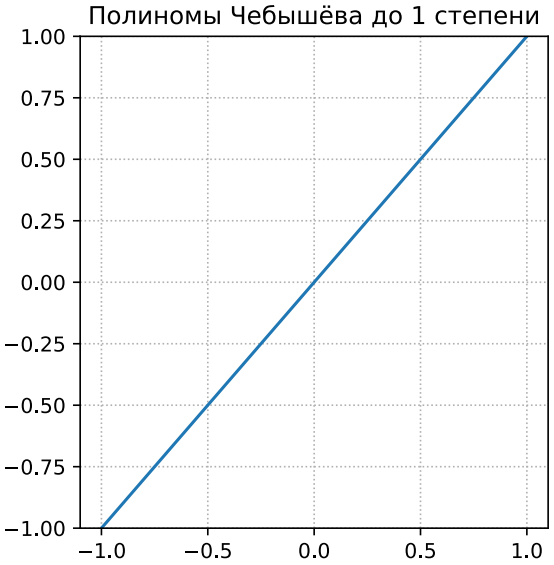


Полиномы Чебышева

Полиномы Чебышёва дают оптимальный ответ на поставленный вопрос. При соответствующем шкалировании они минимизируют абсолютное значение на заданном интервале $[\mu, L]$, одновременно удовлетворяя нормировочному условию $p(0) = 1$.

$$
\begin{aligned}
T_0(x) &= 1 \\
T_1(x) &= x \\
T_k(x) &= 2xT_{k-1}(x) - T_{k-2}(x), \qquad k \geq 2.
\end{aligned}
$$

Давайте построим стандартные полиномы Чебышёва (без масштабирования):

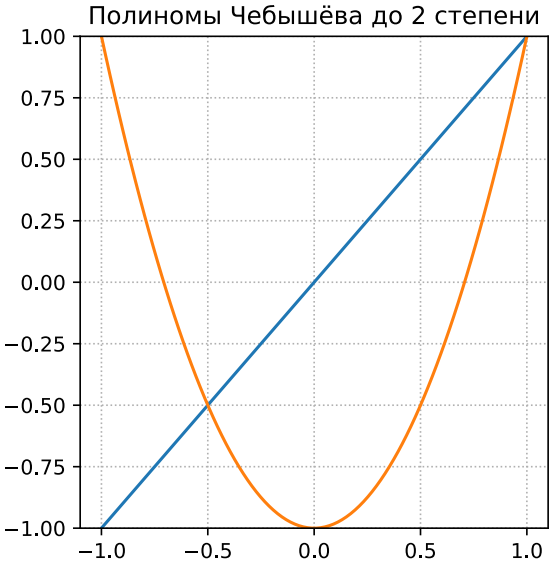


Полиномы Чебышева

Полиномы Чебышёва дают оптимальный ответ на поставленный вопрос. При соответствующем шкалировании они минимизируют абсолютное значение на заданном интервале $[\mu, L]$, одновременно удовлетворяя нормировочному условию $p(0) = 1$.

$$
\begin{aligned}
T_0(x) &= 1 \\
T_1(x) &= x \\
T_k(x) &= 2xT_{k-1}(x) - T_{k-2}(x), \qquad k \geq 2.
\end{aligned}
$$

Давайте построим стандартные полиномы Чебышёва (без масштабирования):

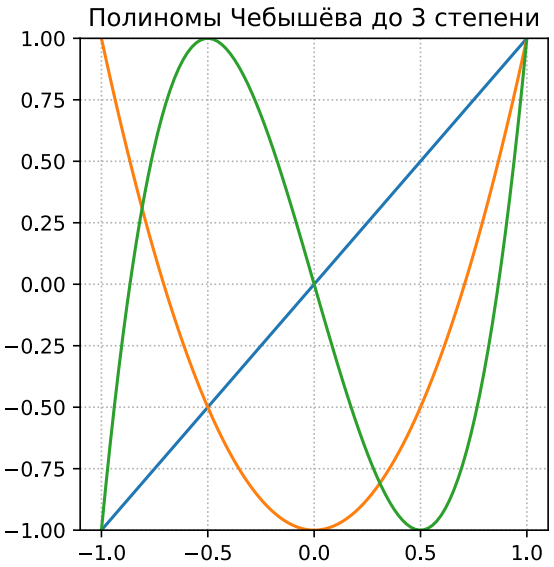


Полиномы Чебышева

Полиномы Чебышёва дают оптимальный ответ на поставленный вопрос. При соответствующем шкалировании они минимизируют абсолютное значение на заданном интервале $[\mu, L]$, одновременно удовлетворяя нормировочному условию $p(0) = 1$.

$$
\begin{aligned}
T_0(x) &= 1 \\
T_1(x) &= x \\
T_k(x) &= 2xT_{k-1}(x) - T_{k-2}(x), \qquad k \geq 2.
\end{aligned}
$$

Давайте построим стандартные полиномы Чебышёва (без масштабирования):

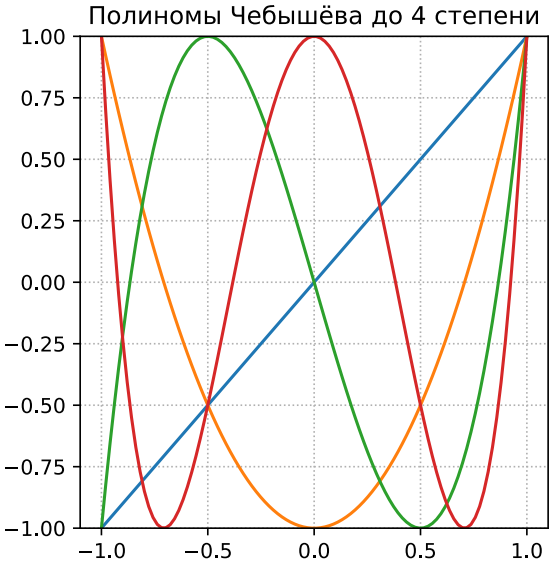


Полиномы Чебышева

Полиномы Чебышёва дают оптимальный ответ на поставленный вопрос. При соответствующем шкалировании они минимизируют абсолютное значение на заданном интервале $[\mu, L]$, одновременно удовлетворяя нормировочному условию $p(0) = 1$.

$$
\begin{aligned}
T_0(x) &= 1 \\
T_1(x) &= x \\
T_k(x) &= 2xT_{k-1}(x) - T_{k-2}(x), \qquad k \geq 2.
\end{aligned}
$$

Давайте построим стандартные полиномы Чебышёва (без масштабирования):

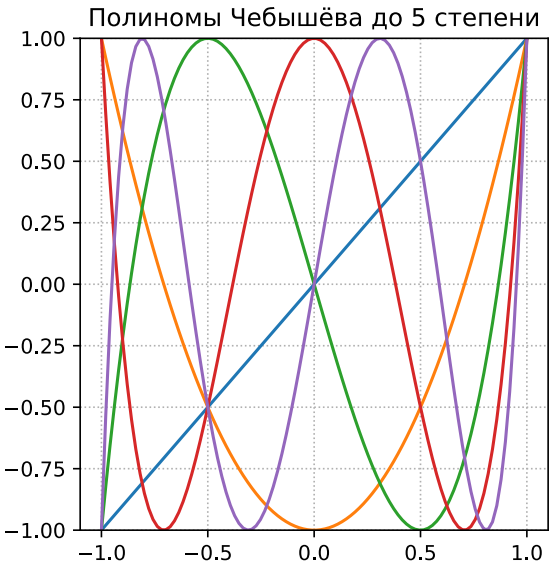


Полиномы Чебышева

Полиномы Чебышёва дают оптимальный ответ на поставленный вопрос. При соответствующем шкалировании они минимизируют абсолютное значение на заданном интервале $[\mu, L]$, одновременно удовлетворяя нормировочному условию $p(0) = 1$.

$$
\begin{aligned}
T_0(x) &= 1 \\
T_1(x) &= x \\
T_k(x) &= 2xT_{k-1}(x) - T_{k-2}(x), \qquad k \geq 2.
\end{aligned}
$$

Давайте построим стандартные полиномы Чебышёва (без масштабирования):



Отшкалированные полиномы Чебышёва

Оригинальные полиномы Чебышёва определены на интервале $[-1, 1]$. Чтобы использовать их для наших целей, мы должны отшкалировать их на интервал $[\mu, L]$.

Отшкалированные полиномы Чебышёва

Оригинальные полиномы Чебышёва определены на интервале $[-1, 1]$. Чтобы использовать их для наших целей, мы должны отшкалировать их на интервал $[\mu, L]$.

Мы будем использовать следующее аффинное преобразование:

$$x = \frac{L + \mu - 2a}{L - \mu}, \quad a \in [\mu, L], \quad x \in [-1, 1].$$

Обратите внимание, что $x = 1$ соответствует $a = \mu$, $x = -1$ соответствует $a = L$ и $x = 0$ соответствует $a = \frac{\mu+L}{2}$. Это преобразование гарантирует, что поведение полинома Чебышёва на интервале $[-1, 1]$ транслируется на интервал $[\mu, L]$.

Отшкалированные полиномы Чебышёва

Оригинальные полиномы Чебышёва определены на интервале $[-1, 1]$. Чтобы использовать их для наших целей, мы должны отшкалировать их на интервал $[\mu, L]$.

Мы будем использовать следующее аффинное преобразование:

$$x = \frac{L + \mu - 2a}{L - \mu}, \quad a \in [\mu, L], \quad x \in [-1, 1].$$

Обратите внимание, что $x = 1$ соответствует $a = \mu$, $x = -1$ соответствует $a = L$ и $x = 0$ соответствует $a = \frac{\mu+L}{2}$. Это преобразование гарантирует, что поведение полинома Чебышёва на интервале $[-1, 1]$ транслируется на интервал $[\mu, L]$.

В нашем анализе ошибок мы требуем, чтобы полином был равен 1 в 0 (т.е. $p_k(0) = 1$). После применения преобразования значение T_k в точке, соответствующей $a = 0$, может не быть 1. Следовательно, мы умножаем на обратную величину T_k в точке

$$\frac{L + \mu}{L - \mu}, \qquad \text{что обеспечивает} \qquad P_k(0) = T_k\left(\frac{L + \mu - 0}{L - \mu}\right) \cdot T_k\left(\frac{L + \mu}{L - \mu}\right)^{-1} = 1.$$

Отшкалированные полиномы Чебышёва

Оригинальные полиномы Чебышёва определены на интервале $[-1, 1]$. Чтобы использовать их для наших целей, мы должны отшкалировать их на интервал $[\mu, L]$.

Мы будем использовать следующее аффинное преобразование:

$$x = \frac{L + \mu - 2a}{L - \mu}, \quad a \in [\mu, L], \quad x \in [-1, 1].$$

Обратите внимание, что $x = 1$ соответствует $a = \mu$, $x = -1$ соответствует $a = L$ и $x = 0$ соответствует $a = \frac{\mu + L}{2}$. Это преобразование гарантирует, что поведение полинома Чебышёва на интервале $[-1, 1]$ транслируется на интервал $[\mu, L]$.

В нашем анализе ошибок мы требуем, чтобы полином был равен 1 в 0 (т.е. $p_k(0) = 1$). После применения преобразования значение T_k в точке, соответствующей $a = 0$, может не быть 1. Следовательно, мы умножаем на обратную величину T_k в точке

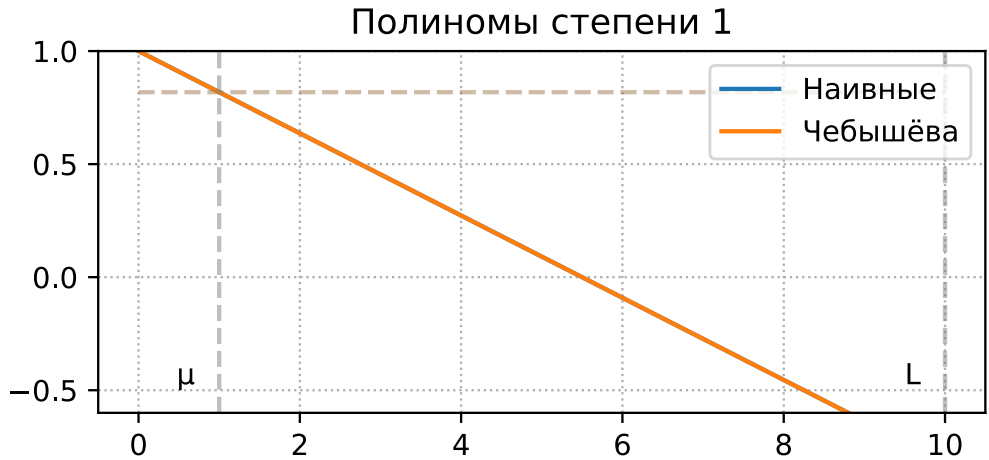
$$\frac{L + \mu}{L - \mu}, \qquad \text{что обеспечивает} \qquad P_k(0) = T_k\left(\frac{L + \mu - 0}{L - \mu}\right) \cdot T_k\left(\frac{L + \mu}{L - \mu}\right)^{-1} = 1.$$

Построим отшкалированные полиномы Чебышёва

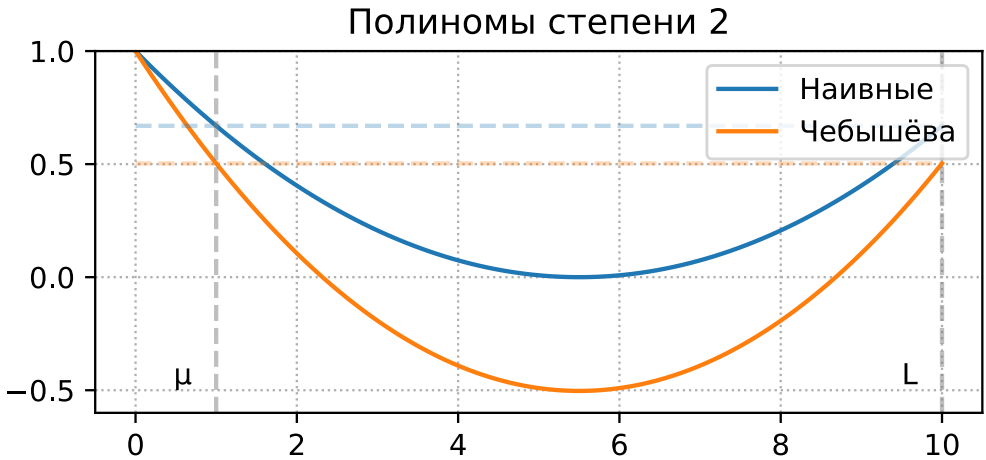
$$P_k(a) = T_k\left(\frac{L + \mu - 2a}{L - \mu}\right) \cdot T_k\left(\frac{L + \mu}{L - \mu}\right)^{-1}$$

и увидим, что они больше подходят для нашей задачи, чем наивные полиномы на интервале $[\mu, L]$.

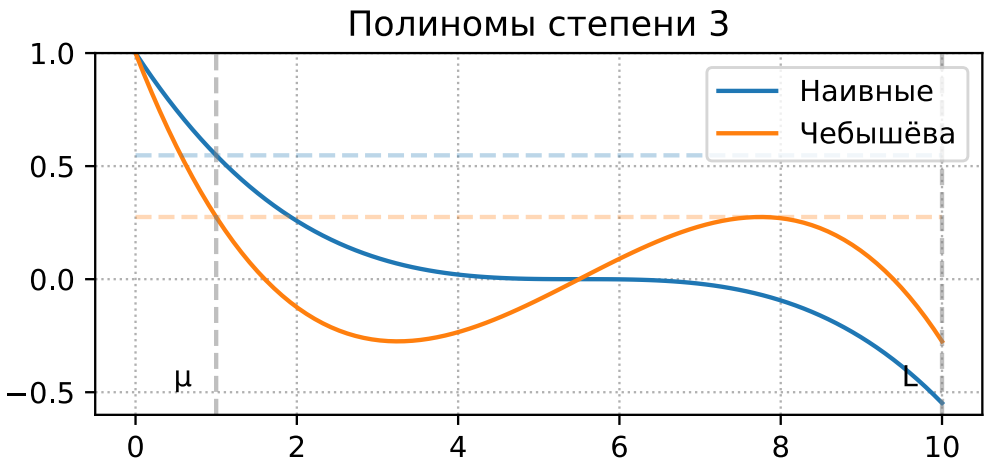
Отшкалированные полиномы Чебышёва



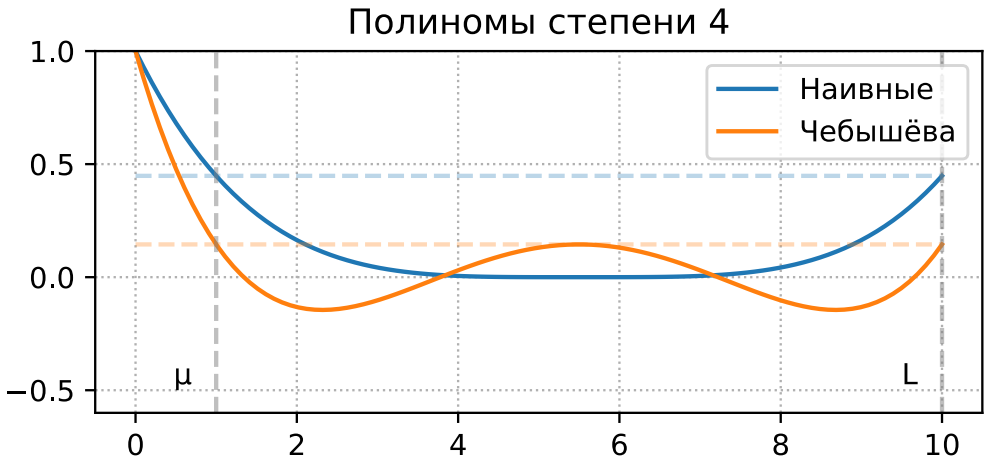
Отшкалированные полиномы Чебышёва



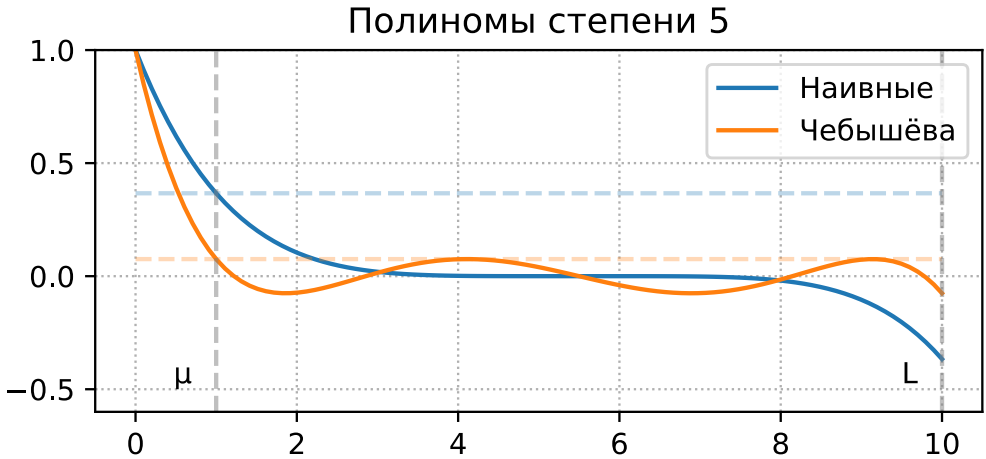
Отшкалированные полиномы Чебышёва



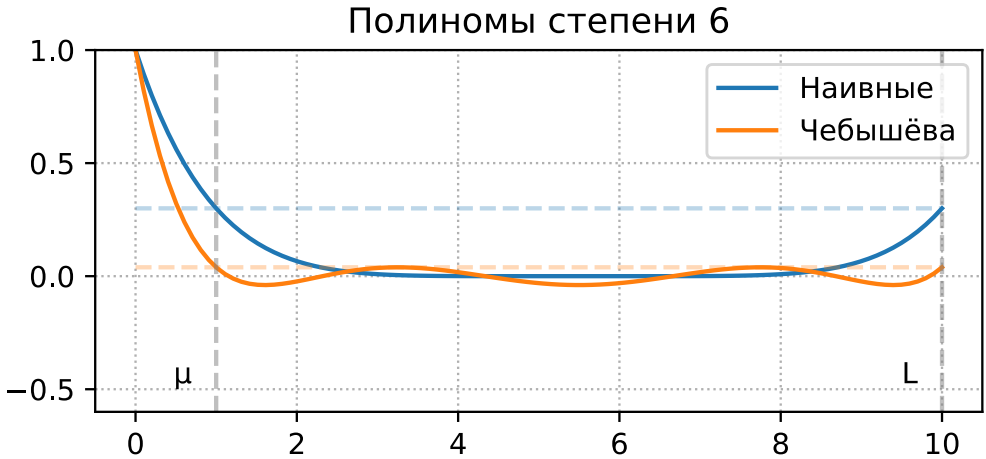
Отшкалированные полиномы Чебышёва



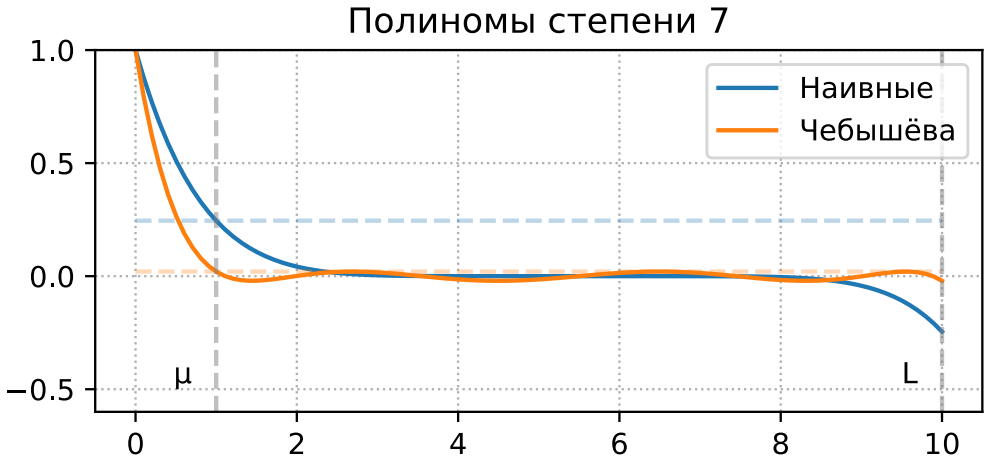
Отшкалированные полиномы Чебышёва



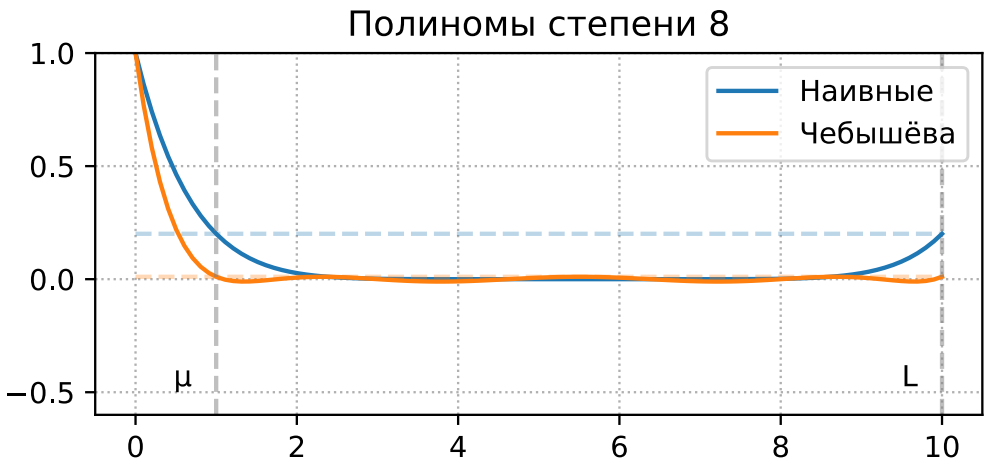
Отшкалированные полиномы Чебышёва



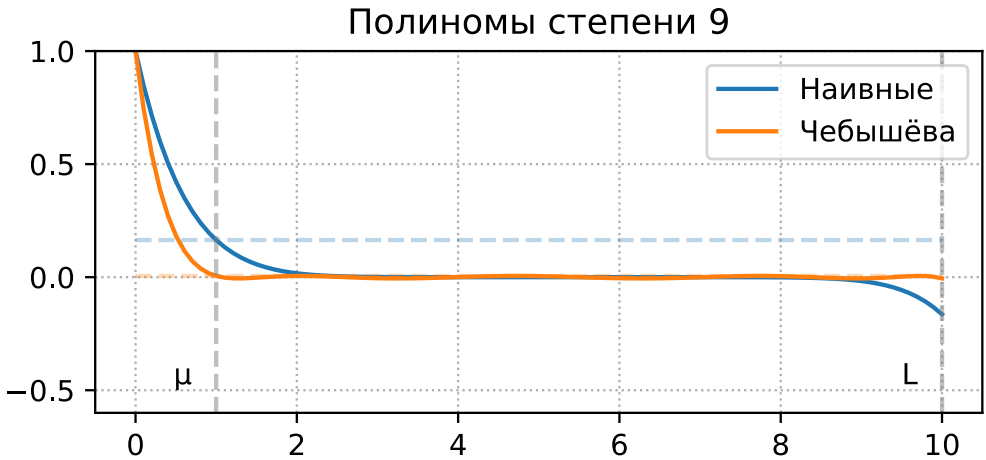
Отшкалированные полиномы Чебышёва



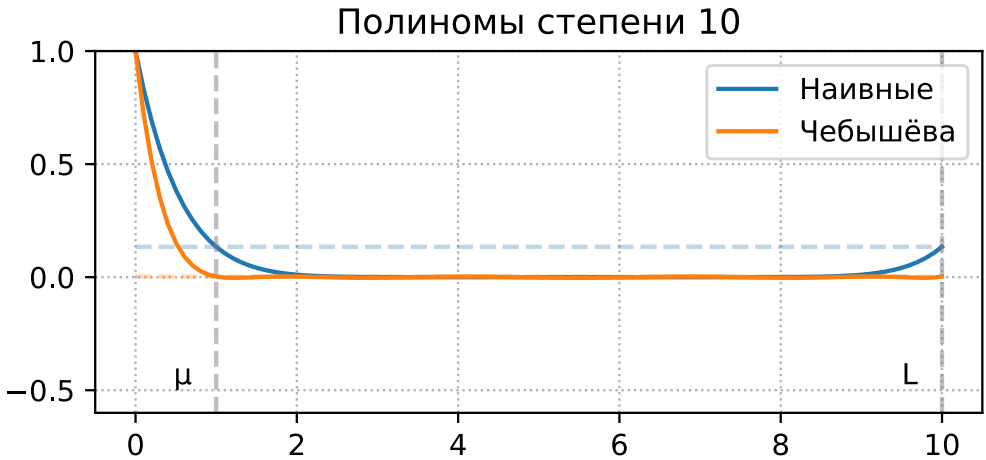
Отшкалированные полиномы Чебышёва



Отшкалированные полиномы Чебышёва



Отшкалированные полиномы Чебышёва



Верхняя оценка для полиномов Чебышёва

Мы можем видеть, что максимальное значение полинома Чебышёва на интервале $[\mu, L]$ достигается на концах отрезка в точках $a = \mu$ и $a = L$. Следовательно, мы можем использовать следующую верхнюю оценку:

$$\|P_k(A)\|_2 \leq P_k(\mu) = T_k\left(\frac{L+\mu-2\mu}{L-\mu}\right) \cdot T_k\left(\frac{L+\mu}{L-\mu}\right)^{-1} = T_k(1) \cdot T_k\left(\frac{L+\mu}{L-\mu}\right)^{-1} = T_k\left(\frac{L+\mu}{L-\mu}\right)^{-1}$$

Верхняя оценка для полиномов Чебышёва

Мы можем видеть, что максимальное значение полинома Чебышёва на интервале $[\mu, L]$ достигается на концах отрезка в точках $a = \mu$ и $a = L$. Следовательно, мы можем использовать следующую верхнюю оценку:

$$\|P_k(A)\|_2 \leq P_k(\mu) = T_k\left(\frac{L+\mu-2\mu}{L-\mu}\right) \cdot T_k\left(\frac{L+\mu}{L-\mu}\right)^{-1} = T_k(1) \cdot T_k\left(\frac{L+\mu}{L-\mu}\right)^{-1} = T_k\left(\frac{L+\mu}{L-\mu}\right)^{-1}$$

Используя определение числа обусловленности $\varkappa = \frac{L}{\mu}$, мы получаем:

$$\|P_k(A)\|_2 \leq T_k\left(\frac{\varkappa+1}{\varkappa-1}\right)^{-1} = T_k\left(1+\frac{2}{\varkappa-1}\right)^{-1} = T_k(1+\epsilon)^{-1}, \quad \epsilon = \frac{2}{\varkappa-1}.$$

Верхняя оценка для полиномов Чебышёва

Мы можем видеть, что максимальное значение полинома Чебышёва на интервале $[\mu, L]$ достигается на концах отрезка в точках $a = \mu$ и $a = L$. Следовательно, мы можем использовать следующую верхнюю оценку:

$$\|P_k(A)\|_2 \leq P_k(\mu) = T_k\left(\frac{L+\mu-2\mu}{L-\mu}\right) \cdot T_k\left(\frac{L+\mu}{L-\mu}\right)^{-1} = T_k(1) \cdot T_k\left(\frac{L+\mu}{L-\mu}\right)^{-1} = T_k\left(\frac{L+\mu}{L-\mu}\right)^{-1}$$

Используя определение числа обусловленности $\varkappa = \frac{L}{\mu}$, мы получаем:

$$\|P_k(A)\|_2 \leq T_k\left(\frac{\varkappa+1}{\varkappa-1}\right)^{-1} = T_k\left(1+\frac{2}{\varkappa-1}\right)^{-1} = T_k(1+\epsilon)^{-1}, \quad \epsilon = \frac{2}{\varkappa-1}.$$

Именно в этот момент явно возникнет ускорение. Мы ограничим значение $\|P_k(A)\|_2$ сверху величиной $\left(\frac{1}{1+\sqrt{\epsilon}}\right)^k$. Для этого детально изучим величину $|T_k(1+\epsilon)|$.

Верхняя оценка для полиномов Чебышёва

Чтобы ограничить $|P_k|$ сверху, мы должны ограничить $|T_k(1+\epsilon)|$ снизу.

Верхняя оценка для полиномов Чебышёва

Чтобы ограничить $|P_k|$ сверху, мы должны ограничить $|T_k(1+\epsilon)|$ снизу.

1. Для любого $x \geq 1$, полиномы Чебышёва первого рода могут быть записаны как

$$T_k(x) = \cosh\left(k \operatorname{arccosh}(x)\right)$$
$$T_k(1+\epsilon) = \cosh\left(k \operatorname{arccosh}(1+\epsilon)\right).$$

Верхняя оценка для полиномов Чебышёва

Чтобы ограничить $|P_k|$ сверху, мы должны ограничить $|T_k(1+\epsilon)|$ снизу.

1. Для любого $x \geq 1$, полиномы Чебышёва первого рода могут быть записаны как

$$T_k(x) = \cosh\left(k \operatorname{arccosh}(x)\right)$$
$$T_k(1+\epsilon) = \cosh\left(k \operatorname{arccosh}(1+\epsilon)\right).$$

2. Помните, что:

$$\cosh(x) = \frac{e^x + e^{-x}}{2} \quad \operatorname{arccosh}(x) = \ln(x + \sqrt{x^2 - 1}).$$

Верхняя оценка для полиномов Чебышёва

Чтобы ограничить $|P_k|$ сверху, мы должны ограничить $|T_k(1+\epsilon)|$ снизу.

1. Для любого $x \geq 1$, полиномы Чебышёва первого рода могут быть записаны как

$$T_k(x) = \cosh\left(k \operatorname{arccosh}(x)\right)$$
$$T_k(1+\epsilon) = \cosh\left(k \operatorname{arccosh}(1+\epsilon)\right).$$

2. Помните, что:

$$\cosh(x) = \frac{e^x + e^{-x}}{2} \quad \operatorname{arccosh}(x) = \ln(x + \sqrt{x^2 - 1}).$$

3. Теперь, пусть $\phi = \operatorname{arccosh}(1+\epsilon)$,

$$e^{\phi} = 1 + \epsilon + \sqrt{2\epsilon + \epsilon^2} \geq 1 + \sqrt{\epsilon}.$$

Верхняя оценка для полиномов Чебышёва

Чтобы ограничить $|P_k|$ сверху, мы должны ограничить $|T_k(1+\epsilon)|$ снизу.

1. Для любого $x \geq 1$, полиномы Чебышёва первого рода могут быть записаны как

$$T_k(x) = \cosh\left(k \operatorname{arccosh}(x)\right)$$
$$T_k(1+\epsilon) = \cosh\left(k \operatorname{arccosh}(1+\epsilon)\right).$$

2. Помните, что:

$$\cosh(x) = \frac{e^x + e^{-x}}{2} \quad \operatorname{arccosh}(x) = \ln(x + \sqrt{x^2-1}).$$

3. Теперь, пусть $\phi = \operatorname{arccosh}(1+\epsilon)$,

$$e^{\phi} = 1 + \epsilon + \sqrt{2\epsilon + \epsilon^2} \geq 1 + \sqrt{\epsilon}.$$

4. Следовательно,

$$\begin{aligned} T_k(1+\epsilon) &= \cosh\left(k \operatorname{arccosh}(1+\epsilon)\right) \\ &= \cosh\left(k\phi\right) \\ &= \frac{e^{k\phi} + e^{-k\phi}}{2} \geq \frac{e^{k\phi}}{2} \\ &= \frac{\left(1+\sqrt{\epsilon}\right)^k}{2}. \end{aligned}$$

Верхняя оценка для полиномов Чебышёва

Чтобы ограничить $|P_k|$ сверху, мы должны ограничить $|T_k(1+\epsilon)|$ снизу.

1. Для любого $x \geq 1$, полиномы Чебышёва первого рода могут быть записаны как

$$T_k(x) = \cosh\left(k \operatorname{arccosh}(x)\right)$$
$$T_k(1+\epsilon) = \cosh\left(k \operatorname{arccosh}(1+\epsilon)\right).$$

2. Помните, что:

$$\cosh(x) = \frac{e^x + e^{-x}}{2} \quad \operatorname{arccosh}(x) = \ln(x + \sqrt{x^2-1}).$$

3. Теперь, пусть $\phi = \operatorname{arccosh}(1+\epsilon)$,

$$e^{\phi} = 1 + \epsilon + \sqrt{2\epsilon + \epsilon^2} \geq 1 + \sqrt{\epsilon}.$$

4. Следовательно,

$$\begin{aligned} T_k(1+\epsilon) &= \cosh\left(k \operatorname{arccosh}(1+\epsilon)\right) \\ &= \cosh\left(k\phi\right) \\ &= \frac{e^{k\phi} + e^{-k\phi}}{2} \geq \frac{e^{k\phi}}{2} \\ &= \frac{\left(1+\sqrt{\epsilon}\right)^k}{2}. \end{aligned}$$

5. Наконец, мы получаем:

$$\begin{aligned} \|e_k\| &\leq \|P_k(A)\|\|e_0\| \leq \frac{2}{\left(1+\sqrt{\epsilon}\right)^k}\|e_0\| \\ &\leq 2\left(1 + \sqrt{\frac{2}{\varkappa - 1}}\right)^{-k}\|e_0\| \\ &\leq 2\exp\left(-\sqrt{\frac{2}{\varkappa-1}}k\right)\|e_0\| \end{aligned}$$

Ускоренный метод [1/2]

Из-за рекурсивного определения полиномов Чебышёва мы непосредственно получаем итерационную схему ускоренного алгоритма. Переформулируя рекурсию в терминах наших отшкалированных полиномов Чебышёва, мы получаем:

$$T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x)$$

Ускоренный метод [1/2]

Из-за рекурсивного определения полиномов Чебышёва мы непосредственно получаем итерационную схему ускоренного алгоритма. Переформулируя рекурсию в терминах наших отшкалированных полиномов Чебышёва, мы получаем:

$$T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x)$$

Принимая во внимание, что $x = \frac{L+\mu-2a}{L-\mu}$, и:

$$P_k(a) = T_k\left(\frac{L+\mu-2a}{L-\mu}\right) T_k\left(\frac{L+\mu}{L-\mu}\right)^{-1}$$

Ускоренный метод [1/2]

Из-за рекурсивного определения полиномов Чебышёва мы непосредственно получаем итерационную схему ускоренного алгоритма. Переформулируя рекурсию в терминах наших отшкалированных полиномов Чебышёва, мы получаем:

$$T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x)$$

Принимая во внимание, что $x = \frac{L+\mu-2a}{L-\mu}$, и:

$$P_k(a) = T_k\left(\frac{L+\mu-2a}{L-\mu}\right) T_k\left(\frac{L+\mu}{L-\mu}\right)^{-1}$$

$$T_k\left(\frac{L+\mu-2a}{L-\mu}\right) = P_k(a) T_k\left(\frac{L+\mu}{L-\mu}\right)$$

Ускоренный метод [1/2]

Из-за рекурсивного определения полиномов Чебышёва мы непосредственно получаем итерационную схему ускоренного алгоритма. Переформулируя рекурсию в терминах наших отшкалированных полиномов Чебышёва, мы получаем:

$$T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x)$$

Принимая во внимание, что $x = \frac{L+\mu-2a}{L-\mu}$, и:

$$P_k(a) = T_k\left(\frac{L+\mu-2a}{L-\mu}\right) T_k\left(\frac{L+\mu}{L-\mu}\right)^{-1}$$

$$T_k\left(\frac{L+\mu-2a}{L-\mu}\right) = P_k(a) T_k\left(\frac{L+\mu}{L-\mu}\right)$$

$$T_{k-1}\left(\frac{L+\mu-2a}{L-\mu}\right) = P_{k-1}(a) T_{k-1}\left(\frac{L+\mu}{L-\mu}\right)$$

$$T_{k+1}\left(\frac{L+\mu-2a}{L-\mu}\right) = P_{k+1}(a) T_{k+1}\left(\frac{L+\mu}{L-\mu}\right)$$

Ускоренный метод [1/2]

Из-за рекурсивного определения полиномов Чебышёва мы непосредственно получаем итерационную схему ускоренного алгоритма. Переформулируя рекурсию в терминах наших отшкалированных полиномов Чебышёва, мы получаем:

$$T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x)$$

Принимая во внимание, что $x = \frac{L+\mu-2a}{L-\mu}$, и:

$$P_k(a) = T_k\left(\frac{L+\mu-2a}{L-\mu}\right) T_k\left(\frac{L+\mu}{L-\mu}\right)^{-1}$$

$$T_k\left(\frac{L+\mu-2a}{L-\mu}\right) = P_k(a)T_k\left(\frac{L+\mu}{L-\mu}\right)$$

$$T_{k-1}\left(\frac{L+\mu-2a}{L-\mu}\right) = P_{k-1}(a)T_{k-1}\left(\frac{L+\mu}{L-\mu}\right)$$

$$T_{k+1}\left(\frac{L+\mu-2a}{L-\mu}\right) = P_{k+1}(a)T_{k+1}\left(\frac{L+\mu}{L-\mu}\right)$$

$$P_{k+1}(a)t_{k+1} = 2\frac{L+\mu-2a}{L-\mu}P_k(a)t_k - P_{k-1}(a)t_{k-1}, \text{ где } t_k = T_k\left(\frac{L+\mu}{L-\mu}\right)$$

Ускоренный метод [1/2]

Из-за рекурсивного определения полиномов Чебышёва мы непосредственно получаем итерационную схему ускоренного алгоритма. Переформулируя рекурсию в терминах наших отшкалированных полиномов Чебышёва, мы получаем:

$$T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x)$$

Принимая во внимание, что $x = \frac{L+\mu-2a}{L-\mu}$, и:

$$P_k(a) = T_k\left(\frac{L+\mu-2a}{L-\mu}\right) T_k\left(\frac{L+\mu}{L-\mu}\right)^{-1}$$

$$T_k\left(\frac{L+\mu-2a}{L-\mu}\right) = P_k(a) T_k\left(\frac{L+\mu}{L-\mu}\right)$$

$$T_{k-1}\left(\frac{L+\mu-2a}{L-\mu}\right) = P_{k-1}(a) T_{k-1}\left(\frac{L+\mu}{L-\mu}\right)$$

$$T_{k+1}\left(\frac{L+\mu-2a}{L-\mu}\right) = P_{k+1}(a) T_{k+1}\left(\frac{L+\mu}{L-\mu}\right)$$

$$P_{k+1}(a)t_{k+1} = 2\frac{L+\mu-2a}{L-\mu}P_k(a)t_k - P_{k-1}(a)t_{k-1}, \text{ где } t_k = T_k\left(\frac{L+\mu}{L-\mu}\right)$$

$$P_{k+1}(a) = 2\frac{L+\mu-2a}{L-\mu}P_k(a)\frac{t_k}{t_{k+1}} - P_{k-1}(a)\frac{t_{k-1}}{t_{k+1}}$$

Ускоренный метод [1/2]

Из-за рекурсивного определения полиномов Чебышёва мы непосредственно получаем итерационную схему ускоренного алгоритма. Переформулируя рекурсию в терминах наших отшкалированных полиномов Чебышёва, мы получаем:

$$T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x)$$

Принимая во внимание, что $x = \frac{L+\mu-2a}{L-\mu}$, и:

$$P_k(a) = T_k\left(\frac{L+\mu-2a}{L-\mu}\right) T_k\left(\frac{L+\mu}{L-\mu}\right)^{-1}$$

$$T_k\left(\frac{L+\mu-2a}{L-\mu}\right) = P_k(a) T_k\left(\frac{L+\mu}{L-\mu}\right)$$

$$T_{k-1}\left(\frac{L+\mu-2a}{L-\mu}\right) = P_{k-1}(a) T_{k-1}\left(\frac{L+\mu}{L-\mu}\right)$$

$$T_{k+1}\left(\frac{L+\mu-2a}{L-\mu}\right) = P_{k+1}(a) T_{k+1}\left(\frac{L+\mu}{L-\mu}\right)$$

$$P_{k+1}(a)t_{k+1} = 2\frac{L+\mu-2a}{L-\mu}P_k(a)t_k - P_{k-1}(a)t_{k-1}, \text{ где } t_k = T_k\left(\frac{L+\mu}{L-\mu}\right)$$

$$P_{k+1}(a) = 2\frac{L+\mu-2a}{L-\mu}P_k(a)\frac{t_k}{t_{k+1}} - P_{k-1}(a)\frac{t_{k-1}}{t_{k+1}}$$

Поскольку мы имеем $P_{k+1}(0) = P_k(0) = P_{k-1}(0) = 1$, получаем рекуррентную формулу вида:

$$P_{k+1}(a) = (1 - \alpha_k a)P_k(a) + \beta_k \left(P_k(a) - P_{k-1}(a)\right).$$

Ускоренный метод [2/2]

Перегруппируя члены, мы получаем:

$$P_{k+1}(a) = (1 + \beta_k)P_k(a) - \alpha_k a P_k(a) - \beta_k P_{k-1}(a),$$

Ускоренный метод [2/2]

Перегруппируя члены, мы получаем:

$$P_{k+1}(a) = (1+\beta_k)P_k(a) - \alpha_k a P_k(a) - \beta_k P_{k-1}(a),$$

$$P_{k+1}(a) = 2\frac{L+\mu}{L-\mu}\frac{t_k}{t_{k+1}}P_k(a) - \frac{4a}{L-\mu}\frac{t_k}{t_{k+1}}P_k(a) - \frac{t_{k-1}}{t_{k+1}}P_{k-1}(a)$$

Ускоренный метод [2/2]

Перегруппируя члены, мы получаем:

$$P_{k+1}(a) = (1+\beta_k)P_k(a) - \alpha_k a P_k(a) - \beta_k P_{k-1}(a),$$

$$P_{k+1}(a) = 2\frac{L+\mu}{L-\mu}\frac{t_k}{t_{k+1}}P_k(a) - \frac{4a}{L-\mu}\frac{t_k}{t_{k+1}}P_k(a) - \frac{t_{k-1}}{t_{k+1}}P_{k-1}(a)$$

$$\begin{cases} \beta_k = \dfrac{t_{k-1}}{t_{k+1}}, \\ \alpha_k = \dfrac{4}{L-\mu}\dfrac{t_k}{t_{k+1}}, \\ 1+\beta_k = 2\dfrac{L+\mu}{L-\mu}\dfrac{t_k}{t_{k+1}} \end{cases}$$

Ускоренный метод [2/2]

Перегруппируя члены, мы получаем:

$$P_{k+1}(a) = (1+\beta_k)P_k(a) - \alpha_k a P_k(a) - \beta_k P_{k-1}(a),$$

$$P_{k+1}(a) = 2\frac{L+\mu}{L-\mu}\frac{t_k}{t_{k+1}}P_k(a) - \frac{4a}{L-\mu}\frac{t_k}{t_{k+1}}P_k(a) - \frac{t_{k-1}}{t_{k+1}}P_{k-1}(a)$$

$$\begin{cases} \beta_k = \dfrac{t_{k-1}}{t_{k+1}}, \\ \alpha_k = \dfrac{4}{L-\mu}\dfrac{t_k}{t_{k+1}}, \\ 1+\beta_k = 2\dfrac{L+\mu}{L-\mu}\dfrac{t_k}{t_{k+1}} \end{cases}$$

Мы почти закончили :) Помним, что $e_{k+1} = P_{k+1}(A)e_0$. Также обратим внимание, что мы работаем с квадратичной задачей, поэтому мы можем предположить $x^* = 0$ без ограничения общности. В этом случае $e_0 = x_0$ и $e_{k+1} = x_{k+1}$.

Ускоренный метод [2/2]

Перегруппируя члены, мы получаем:

$$P_{k+1}(a) = (1+\beta_k)P_k(a) - \alpha_k a P_k(a) - \beta_k P_{k-1}(a),$$

$$P_{k+1}(a) = 2\frac{L+\mu}{L-\mu}\frac{t_k}{t_{k+1}}P_k(a) - \frac{4a}{L-\mu}\frac{t_k}{t_{k+1}}P_k(a) - \frac{t_{k-1}}{t_{k+1}}P_{k-1}(a)$$

$$\begin{cases} \beta_k = \dfrac{t_{k-1}}{t_{k+1}}, \\ \alpha_k = \dfrac{4}{L-\mu}\dfrac{t_k}{t_{k+1}}, \\ 1+\beta_k = 2\dfrac{L+\mu}{L-\mu}\dfrac{t_k}{t_{k+1}} \end{cases}$$

Мы почти закончили :) Помним, что $e_{k+1} = P_{k+1}(A)e_0$. Также обратим внимание, что мы работаем с квадратичной задачей, поэтому мы можем предположить $x^* = 0$ без ограничения общности. В этом случае $e_0 = x_0$ и $e_{k+1} = x_{k+1}$.

$$x_{k+1} = P_{k+1}(A)x_0$$

Ускоренный метод [2/2]

Перегруппируя члены, мы получаем:

$$P_{k+1}(a) = (1+\beta_k)P_k(a) - \alpha_k a P_k(a) - \beta_k P_{k-1}(a),$$

$$P_{k+1}(a) = 2\frac{L+\mu}{L-\mu}\frac{t_k}{t_{k+1}}P_k(a) - \frac{4a}{L-\mu}\frac{t_k}{t_{k+1}}P_k(a) - \frac{t_{k-1}}{t_{k+1}}P_{k-1}(a)$$

$$\begin{cases} \beta_k = \dfrac{t_{k-1}}{t_{k+1}}, \\ \alpha_k = \dfrac{4}{L-\mu}\dfrac{t_k}{t_{k+1}}, \\ 1+\beta_k = 2\dfrac{L+\mu}{L-\mu}\dfrac{t_k}{t_{k+1}} \end{cases}$$

Мы почти закончили :) Помним, что $e_{k+1} = P_{k+1}(A)e_0$. Также обратим внимание, что мы работаем с квадратичной задачей, поэтому мы можем предположить $x^* = 0$ без ограничения общности. В этом случае $e_0 = x_0$ и $e_{k+1} = x_{k+1}$.

$$x_{k+1} = P_{k+1}(A)x_0 = (I - \alpha_k A)P_k(A)x_0 + \beta_k\left(P_k(A) - P_{k-1}(A)\right)x_0$$

Ускоренный метод [2/2]

Перегруппируя члены, мы получаем:

$$P_{k+1}(a) = (1+\beta_k)P_k(a) - \alpha_k a P_k(a) - \beta_k P_{k-1}(a),$$
$$P_{k+1}(a) = 2\frac{L+\mu}{L-\mu}\frac{t_k}{t_{k+1}}P_k(a) - \frac{4a}{L-\mu}\frac{t_k}{t_{k+1}}P_k(a) - \frac{t_{k-1}}{t_{k+1}}P_{k-1}(a)$$

$$\begin{cases} \beta_k = \dfrac{t_{k-1}}{t_{k+1}}, \\ \alpha_k = \dfrac{4}{L-\mu}\dfrac{t_k}{t_{k+1}}, \\ 1+\beta_k = 2\dfrac{L+\mu}{L-\mu}\dfrac{t_k}{t_{k+1}} \end{cases}$$

Мы почти закончили :) Помним, что $e_{k+1} = P_{k+1}(A)e_0$. Также обратим внимание, что мы работаем с квадратичной задачей, поэтому мы можем предположить $x^* = 0$ без ограничения общности. В этом случае $e_0 = x_0$ и $e_{k+1} = x_{k+1}$.

$$\begin{aligned} x_{k+1} &= P_{k+1}(A)x_0 = (I - \alpha_k A)P_k(A)x_0 + \beta_k\left(P_k(A) - P_{k-1}(A)\right)x_0 \\ &= (I - \alpha_k A)x_k + \beta_k(x_k - x_{k-1}) \end{aligned}$$

Ускоренный метод [2/2]

Перегруппируя члены, мы получаем:

$$P_{k+1}(a) = (1+\beta_k)P_k(a) - \alpha_k a P_k(a) - \beta_k P_{k-1}(a),$$
$$P_{k+1}(a) = 2\frac{L+\mu}{L-\mu}\frac{t_k}{t_{k+1}}P_k(a) - \frac{4a}{L-\mu}\frac{t_k}{t_{k+1}}P_k(a) - \frac{t_{k-1}}{t_{k+1}}P_{k-1}(a)$$

$$\begin{cases} \beta_k = \dfrac{t_{k-1}}{t_{k+1}}, \\ \alpha_k = \dfrac{4}{L-\mu}\dfrac{t_k}{t_{k+1}}, \\ 1+\beta_k = 2\dfrac{L+\mu}{L-\mu}\dfrac{t_k}{t_{k+1}} \end{cases}$$

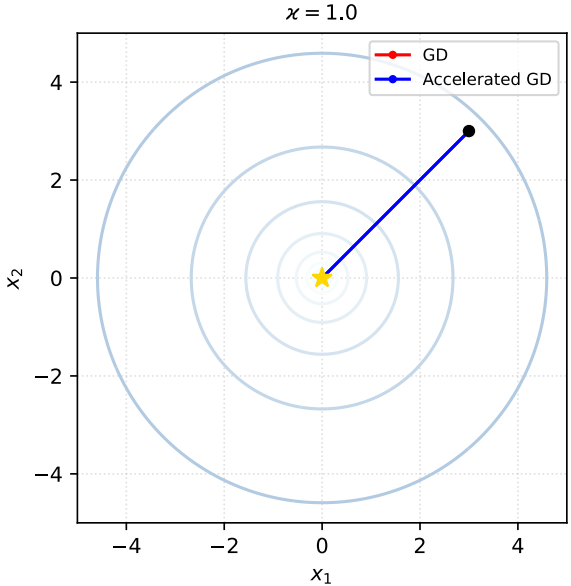
Мы почти закончили :) Помним, что $e_{k+1} = P_{k+1}(A)e_0$. Также обратим внимание, что мы работаем с квадратичной задачей, поэтому мы можем предположить $x^* = 0$ без ограничения общности. В этом случае $e_0 = x_0$ и $e_{k+1} = x_{k+1}$.

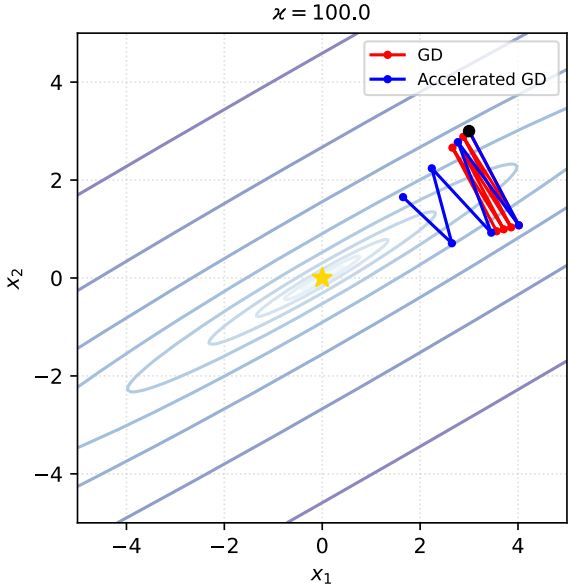
$$\begin{aligned} x_{k+1} &= P_{k+1}(A)x_0 = (I - \alpha_k A)P_k(A)x_0 + \beta_k\left(P_k(A) - P_{k-1}(A)\right)x_0 \\ &= (I - \alpha_k A)x_k + \beta_k(x_k - x_{k-1}) \end{aligned}$$

Для квадратичной задачи мы имеем $\nabla f(x_k) = Ax_k$, поэтому мы можем переписать обновление как:

$$\boxed{x_{k+1} = x_k - \alpha_k \nabla f(x_k) + \beta_k(x_k - x_{k-1})}$$

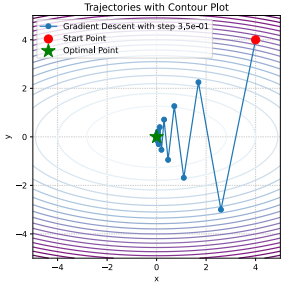
Ускорение из первых принципов

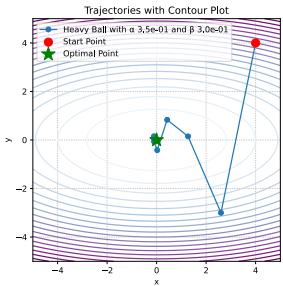




Бонус: анализ сходимости метода тяжёлого шарика

Метод тяжёлого шарика Поляка

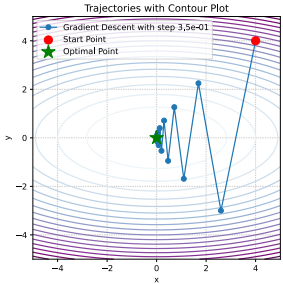


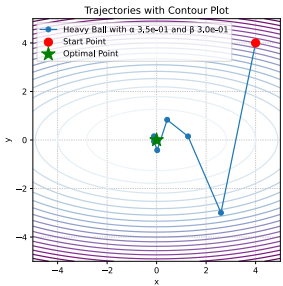


Давайте представим идею моментума (импульса, тяжёлого шарика), предложенную Б.Т. Поляком в 1964 году. Обновление метода тяжёлого шарика имеет вид

$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}).$$

Метод тяжёлого шарика Поляка





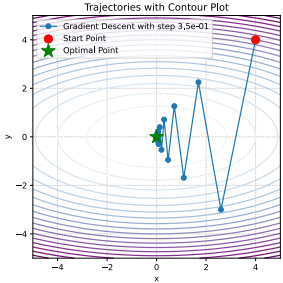
Давайте представим идею моментума (импульса, тяжёлого шарика), предложенную Б.Т. Поляком в 1964 году. Обновление метода тяжёлого шарика имеет вид

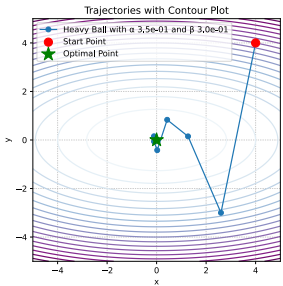
$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}).$$

В нашем (квадратичном) случае это

$$\hat{x}_{k+1} = \hat{x}_k - \alpha \Lambda \hat{x}_k + \beta(\hat{x}_k - \hat{x}_{k-1}) = (I - \alpha\Lambda + \beta I)\hat{x}_k - \beta \hat{x}_{k-1}$$

Метод тяжёлого шарика Поляка





Давайте представим идею моментума (импульса, тяжёлого шарика), предложенную Б.Т. Поляком в 1964 году. Обновление метода тяжёлого шарика имеет вид

$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}).$$

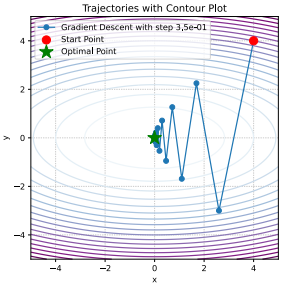
В нашем (квадратичном) случае это

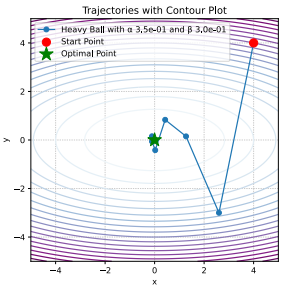
$$\hat{x}_{k+1} = \hat{x}_k - \alpha\Lambda\hat{x}_k + \beta(\hat{x}_k - \hat{x}_{k-1}) = (I - \alpha\Lambda + \beta I)\hat{x}_k - \beta\hat{x}_{k-1}$$

Это можно переписать как

$$\begin{aligned}\hat{x}_{k+1} &= (I - \alpha\Lambda + \beta I)\hat{x}_k - \beta\hat{x}_{k-1},\\ \hat{x}_k &= \hat{x}_k.\end{aligned}$$

Метод тяжёлого шарика Поляка





Давайте представим идею моментума (импульса, тяжёлого шарика), предложенную Б.Т. Поляком в 1964 году. Обновление метода тяжёлого шарика имеет вид

$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}).$$

В нашем (квадратичном) случае это

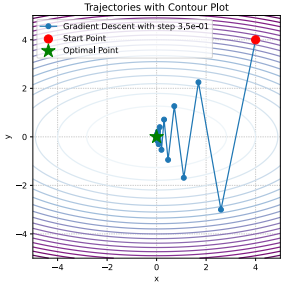
$$\hat{x}_{k+1} = \hat{x}_k - \alpha\Lambda\hat{x}_k + \beta(\hat{x}_k - \hat{x}_{k-1}) = (I - \alpha\Lambda + \beta I)\hat{x}_k - \beta\hat{x}_{k-1}$$

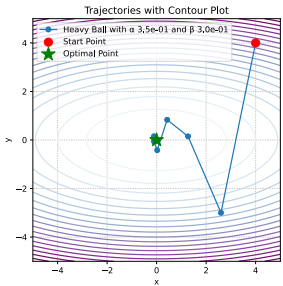
Это можно переписать как

$$\begin{aligned}\hat{x}_{k+1} &= (I - \alpha\Lambda + \beta I)\hat{x}_k - \beta\hat{x}_{k-1},\\ \hat{x}_k &= \hat{x}_k.\end{aligned}$$

Давайте введем следующее обозначение: $\hat{z}_k = \begin{bmatrix}\hat{x}_{k+1}\\ \hat{x}_k\end{bmatrix}$. Следовательно, $\hat{z}_{k+1} = M\hat{z}_k$, где матрица итерации M имеет вид:

Метод тяжёлого шарика Поляка





Давайте представим идею моментума (импульса, тяжёлого шарика), предложенную Б.Т. Поляком в 1964 году. Обновление метода тяжёлого шарика имеет вид

$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}).$$

В нашем (квадратичном) случае это

$$\hat{x}_{k+1} = \hat{x}_k - \alpha\Lambda\hat{x}_k + \beta(\hat{x}_k - \hat{x}_{k-1}) = (I - \alpha\Lambda + \beta I)\hat{x}_k - \beta\hat{x}_{k-1}$$

Это можно переписать как

$$\begin{aligned}\hat{x}_{k+1} &= (I - \alpha\Lambda + \beta I)\hat{x}_k - \beta\hat{x}_{k-1},\\ \hat{x}_k &= \hat{x}_k.\end{aligned}$$

Давайте введем следующее обозначение: $\hat{z}_k = \begin{bmatrix}\hat{x}_{k+1}\\ \hat{x}_k\end{bmatrix}$. Следовательно, $\hat{z}_{k+1} = M\hat{z}_k$, где матрица итерации M имеет вид:

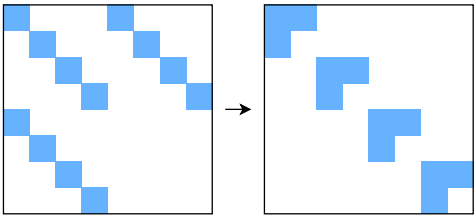
$$M = \begin{bmatrix} I - \alpha\Lambda + \beta I & -\beta I \\ I & 0_d \end{bmatrix}.$$

Сведение к скалярному случаю

Обратим внимание, что M является матрицей $2d \times 2d$ с четырьмя блочно-диагональными матрицами размера $d \times d$ внутри. Это означает, что мы можем изменить порядок координат, чтобы сделать M блочно-диагональной. Обратите внимание, что в уравнении ниже матрица M обозначает то же самое, что и в обозначении выше, за исключением описанной перестановки строк и столбцов. Мы используем эту небольшую перегрузку обозначений для простоты.

Сведение к скалярному случаю

Обратим внимание, что M является матрицей $2d \times 2d$ с четырьмя блочно-диагональными матрицами размера $d \times d$ внутри. Это означает, что мы можем изменить порядок координат, чтобы сделать M блочно-диагональной. Обратите внимание, что в уравнении ниже матрица M обозначает то же самое, что и в обозначении выше, за исключением описанной перестановки строк и столбцов. Мы используем эту небольшую перегрузку обозначений для простоты.



$$\begin{bmatrix} \hat{x}_k^{(1)} \\ \vdots \\ \hat{x}_k^{(d)} \\ \hat{x}_{k-1}^{(1)} \\ \vdots \\ \hat{x}_{k-1}^{(d)} \end{bmatrix} \to \begin{bmatrix} \hat{x}_k^{(1)} \\ \hat{x}_{k-1}^{(1)} \\ \vdots \\ \hat{x}_k^{(d)} \\ \hat{x}_{k-1}^{(d)} \end{bmatrix} \qquad M = \begin{bmatrix} M_1 & & & \\ & M_2 & & \\ & & \ddots & \\ & & & M_d \end{bmatrix}$$

Рис. 8: Иллюстрация перестановки матрицы M

где $\hat{x}_k^{(i)}$ является i-й координатой вектора $\hat{x}_k \in \mathbb{R}^d$ и M_i обозначает 2×2 матрицу. Переупорядочение позволяет нам исследовать динамику метода независимо от размерности. Асимптотическая скорость сходимости $2d$-мерной последовательности векторов $\hat{z}_k$ определяется наихудшей скоростью сходимости среди его блока координат. Следовательно, достаточно исследовать оптимизацию в одномерном случае.

Сведение к скалярному случаю

Для i-й координаты, где λ_i — i-е собственное значение матрицы A, имеем:

$$M_i = \begin{bmatrix} 1 - \alpha\lambda_i + \beta & -\beta \\ 1 & 0 \end{bmatrix}.$$

Сведение к скалярному случаю

Для i-й координаты, где λ_i — i-е собственное значение матрицы A, имеем:

$$M_i = \begin{bmatrix} 1 - \alpha\lambda_i + \beta & -\beta \\ 1 & 0 \end{bmatrix}.$$

Метод будет сходиться, если $\rho(M) < 1$, и оптимальные параметры могут быть вычислены путем оптимизации спектрального радиуса

$$\alpha^*, \beta^* = \arg\min_{\alpha,\beta} \max_i \rho(M_i), \quad \alpha^* = \frac{4}{(\sqrt{L} + \sqrt{\mu})^2}, \quad \beta^* = \left(\frac{\sqrt{L} - \sqrt{\mu}}{\sqrt{L} + \sqrt{\mu}} \right)^2.$$

Сведение к скалярному случаю

Для i-й координаты, где λ_i — i-е собственное значение матрицы A, имеем:

$$M_i = \begin{bmatrix} 1 - \alpha\lambda_i + \beta & -\beta \\ 1 & 0 \end{bmatrix}.$$

Метод будет сходиться, если $\rho(M) < 1$, и оптимальные параметры могут быть вычислены путем оптимизации спектрального радиуса

$$\alpha^*, \beta^* = \arg\min_{\alpha,\beta} \max_i \rho(M_i), \quad \alpha^* = \frac{4}{(\sqrt{L} + \sqrt{\mu})^2}, \quad \beta^* = \left(\frac{\sqrt{L} - \sqrt{\mu}}{\sqrt{L} + \sqrt{\mu}} \right)^2.$$

Можно показать, что для таких параметров матрица M имеет комплексные собственные значения, которые образуют комплексно-сопряжённую пару, поэтому расстояние до оптимума (в этом случае $\|z_k\|$) обычно не убывает монотонно.

Сходимость метода тяжёлого шарика для квадратичной функции

Мы можем явно вычислить собственные значения M_i:

$$\lambda_1^M, \lambda_2^M = \lambda\left(\begin{bmatrix} 1 - \alpha\lambda_i + \beta & -\beta \\ 1 & 0 \end{bmatrix}\right) = \frac{1 + \beta - \alpha\lambda_i \pm \sqrt{(1 + \beta - \alpha\lambda_i)^2 - 4\beta}}{2}.$$

Сходимость метода тяжёлого шарика для квадратичной функции

Мы можем явно вычислить собственные значения M_i:

$$\lambda_1^M, \lambda_2^M = \lambda\left(\begin{bmatrix} 1-\alpha\lambda_i+\beta & -\beta \\ 1 & 0 \end{bmatrix}\right) = \frac{1+\beta-\alpha\lambda_i \pm \sqrt{(1+\beta-\alpha\lambda_i)^2-4\beta}}{2}.$$

Когда α и β оптимальны (α^*, β^*), собственные значения являются комплексно-сопряженной парой $(1+\beta-\alpha\lambda_i)^2 - 4\beta \leq 0$, т.е. $\beta \geq (1-\sqrt{\alpha\lambda_i})^2$.

Сходимость метода тяжёлого шарика для квадратичной функции

Мы можем явно вычислить собственные значения M_i:

$$\lambda_1^M, \lambda_2^M = \lambda\left(\begin{bmatrix} 1 - \alpha\lambda_i + \beta & -\beta \\ 1 & 0 \end{bmatrix}\right) = \frac{1 + \beta - \alpha\lambda_i \pm \sqrt{(1 + \beta - \alpha\lambda_i)^2 - 4\beta}}{2}.$$

Когда α и β оптимальны (α^*, β^*), собственные значения являются комплексно-сопряженной парой $(1 + \beta - \alpha\lambda_i)^2 - 4\beta \leq 0$, т.е. $\beta \geq (1 - \sqrt{\alpha\lambda_i})^2$.

$$\text{Re}(\lambda^M) = \frac{L + \mu - 2\lambda_i}{(\sqrt{L} + \sqrt{\mu})^2}, \quad \text{Im}(\lambda^M) = \frac{\pm 2\sqrt{(L - \lambda_i)(\lambda_i - \mu)}}{(\sqrt{L} + \sqrt{\mu})^2}, \quad |\lambda^M| = \frac{L - \mu}{(\sqrt{L} + \sqrt{\mu})^2}.$$

Сходимость метода тяжёлого шарика для квадратичной функции

Мы можем явно вычислить собственные значения M_i:

$$\lambda_1^M, \lambda_2^M = \lambda\left(\begin{bmatrix} 1 - \alpha\lambda_i + \beta & -\beta \\ 1 & 0 \end{bmatrix}\right) = \frac{1 + \beta - \alpha\lambda_i \pm \sqrt{(1 + \beta - \alpha\lambda_i)^2 - 4\beta}}{2}.$$

Когда α и β оптимальны (α^*, β^*), собственные значения являются комплексно-сопряженной парой $(1 + \beta - \alpha\lambda_i)^2 - 4\beta \leq 0$, т.е. $\beta \geq (1 - \sqrt{\alpha\lambda_i})^2$.

$$\text{Re}(\lambda^M) = \frac{L + \mu - 2\lambda_i}{(\sqrt{L} + \sqrt{\mu})^2}, \quad \text{Im}(\lambda^M) = \frac{\pm 2\sqrt{(L - \lambda_i)(\lambda_i - \mu)}}{(\sqrt{L} + \sqrt{\mu})^2}, \quad |\lambda^M| = \frac{L - \mu}{(\sqrt{L} + \sqrt{\mu})^2}.$$

И скорость сходимости не зависит от шага и равна $\sqrt{\beta^*}$.