Быстрый градиентный метод

Родоманов А. О.   Кропотов Д. А.

Факультет ВМК
МГУ им. М. В. Ломоносова

15 апреля 2014 г.

Спецсеминар «Байесовские методы машинного обучения»

Обзор

1. Введение
   - Задачи и методы оптимизации
   - Основные теоретические сведения
2. Метод градиентного спуска
   - Стандартный метод для гладких функций
   - Метод для составных функций и автоматический подбор L
3. Быстрый градиентный метод
   - Метод для составных функций
   - Рестарт
   - Модификация подбора L и выбор L_0
4. Эксперименты и выводы
   - Эксперименты
   - Выводы

План

1. Введение
   - Задачи и методы оптимизации
   - Основные теоретические сведения
2. Метод градиентного спуска
   - Стандартный метод для гладких функций
   - Метод для составных функций и автоматический подбор L
3. Быстрый градиентный метод
   - Метод для составных функций
   - Рестарт
   - Модификация подбора L и выбор L_0
4. Эксперименты и выводы
   - Эксперименты
   - Выводы

Общая постановка непрерывной задачи оптимизации

Непрерывная задача оптимизации

$$\begin{aligned} \min_{\mathbf{x}} \quad & f_0(\mathbf{x}) \\ s.t. \quad & g_i(\mathbf{x}) \leqslant 0, \quad i = 1, \ldots, m, \\ & h_j(\mathbf{x}) = 0, \quad j = 1, \ldots, k, \\ & \mathbf{x} \in S, \end{aligned}$$

где

- $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n$;
- $S \subseteq \mathbb{R}^n$;
- $f_0(\mathbf{x}), g_1(\mathbf{x}), \ldots, g_m(\mathbf{x}), h_1(\mathbf{x}), \ldots, h_k(\mathbf{x})$ — непрерывные вещественные функции.

Итеративная природа методов оптимизации

1. Обычно метод не способен найти точное решение за конечное число шагов.
2. Метод генерирует бесконечную последовательность «приблизительных» решений $\{\mathbf{x}_k\}_{k=0}^{\infty}$.
3. Каждая следующая точка $\mathbf{x}_{k+1}$ формируется по некоторым правилам на основе *локальной* информации, собранной на предыдущих итерациях.

Итеративная природа методов оптимизации – 2

Общая итеративная схема метода оптимизации

Вход: начальное приближение $\mathbf{x}_0 \in \mathbb{R}^n$ и параметр точности $\varepsilon > 0$;
Выход: решение $\bar{\mathbf{x}} \in \mathbb{R}^n$ в пределах заданной точности;

$\mathcal{I}_{-1} := \varnothing$;
для $k = 0, 1, 2, \ldots$
    {вычислить локальную информацию $\mathcal{O}(\mathbf{x}_k)$ в точке $\mathbf{x}_k$};
    $\mathcal{I}_k := \mathcal{I}_{k-1} \cup (\mathbf{x}_k, \mathcal{O}(\mathbf{x}_k))$; // обновить собранную информацию
    {применить правила метода к $\mathcal{I}_k$ для формирования $\mathbf{x}_{k+1}$};
    {проверить критерий остановки для заданной точности ε};
    если *{критерий остановки выполняется}* **то**
        {сформировать итоговый ответ $\bar{\mathbf{x}}$};
        выход;

Сходимость и сложность методов оптимизации

1. Метод оптимизации должен *сходиться*.
2. *Скорость сходимости* метода определяет количество итераций, необходимых и достаточных для решения оптимизационной задачи.
3. Арифметическая сложность метода складывается из его скорости сходимости и *арифметической сложности одной итерации*.

Классификация методов оптимизации

- Методы *0-го порядка*
  Требуют: только значения функций
- Методы *1-го порядка*
  Требуют: значения функций и первые производные
- Методы *2-го порядка*
  Требуют: значения функций, первые и вторые производные

Метод какого порядка использовать?

С ростом порядка метода

1. скорость сходимости возрастает;
2. арифметическая сложность одной итерации тоже возрастает.

Пример (логистическая регрессия)

$$Q(\mathbf{w}) = \frac{1}{m}\sum_{i=1}^{m} \ln(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i)) \to \min_{\mathbf{w}\in\mathbb{R}^n}$$

Объект вычисления	Арифметическая сложность
Значение функции	$O(mn)$
Первые производные	$O(mn + n^2)$
Вторые производные	$O(mn + n^3)$

Вывод: при значениях $n \geqslant 500$ слишком дорого считать вторые производные на каждой итерации.

План

1. Введение
   - Задачи и методы оптимизации
   - Основные теоретические сведения
2. Метод градиентного спуска
   - Стандартный метод для гладких функций
   - Метод для составных функций и автоматический подбор L
3. Быстрый градиентный метод
   - Метод для составных функций
   - Рестарт
   - Модификация подбора L и выбор L_0
4. Эксперименты и выводы
   - Эксперименты
   - Выводы

Предварительные замечания

В дальнейшем мы будем рассматривать:

- методы оптимизации 1-го порядка;
- оптимизационные задачи без ограничений, т. е. задачи вида
$$f(\mathbf{x}) \to \min_{\mathbf{x}\in\mathbb{R}^n},$$
где $f(\mathbf{x})$ — некоторая непрерывная функция.

Гладкие и негладкие функции

Определение

Функция $f(\mathbf{x})$ называется *гладкой*, если все ее частные производные существуют и являются непрерывными функциями.

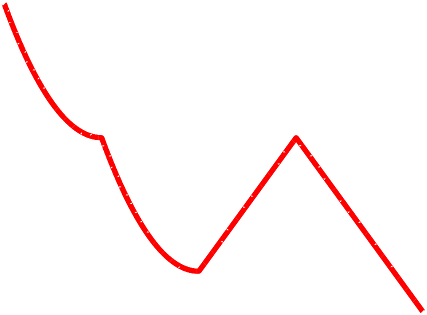
Пример (гладкая функция: логистическая регрессия)

$$f(\mathbf{w}) = \frac{1}{m}\sum_{i=1}^{m} \ln(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i))$$

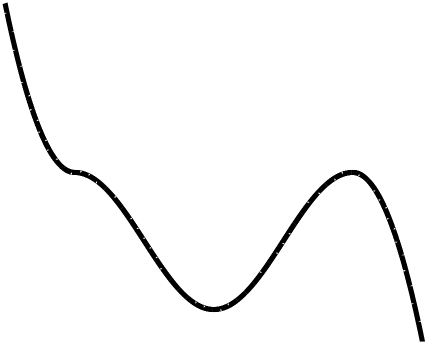
Пример (негладкая функция: L_1-регуляризатор)

$$g(\mathbf{w}) = \tau\|\mathbf{w}\|_1 = \tau\sum_{j=1}^{n} |w^{(j)}|, \quad \tau \geqslant 0$$

Гладкие и негладкие функции: иллюстрация



Негладкая функция



Гладкая функция

Градиент и гессиан функции

Определение

Градиентом гладкой функции $f(\mathbf{x})$ в точке $\mathbf{x}$ называется вектор $\nabla f(\mathbf{x})$, составленный из частных производных функции в этой точке:

$$\nabla f(\mathbf{x}) = \left(\frac{\partial f(\mathbf{x})}{\partial x^{(j)}} \right)_{j=1}^{n}$$

Определение

Гессианом дважды дифференцируемой функции $f(\mathbf{x})$ в точке $\mathbf{x}$ называется матрица $\nabla^2 f(\mathbf{x})$, составленная из вторых частных производных функции в этой точке:

$$\nabla^2 f(\mathbf{x}) = \left(\frac{\partial^2 f(\mathbf{x})}{\partial x^{(j)} \partial x^{(k)}} \right)_{j,\,k=1}^{n}$$

Формула Тейлора

Градиент и гессиан позволяют локально аппроксимировать гладкую функцию.

Пусть $f(\mathbf{x})$ является гладкой функцией. Тогда $\forall\, \mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ справедливы следующие формулы:

Линейная аппроксимация

$$f(\mathbf{x}) = f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle + o(\|\mathbf{x} - \mathbf{y}\|)$$

Квадратичная аппроксимация

$$f(\mathbf{x}) = f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle + \langle \nabla^2 f(\mathbf{y})(\mathbf{x} - \mathbf{y}), \mathbf{x} - \mathbf{y} \rangle + o(\|\mathbf{x} - \mathbf{y}\|^2)$$

Класс функций $\mathcal{F}_L^1$

Определение

Говорят, что гладкая функция $f(\mathbf{x})$ обладает *липшицевым градиентом с константой* $L > 0$, если $\forall\, \mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ справедливо

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leqslant L\|\mathbf{x} - \mathbf{y}\|.$$

Определение

Функция $f(\mathbf{x})$ называется *выпуклой*, если $\forall\, \mathbf{x}, \mathbf{y} \in \mathbb{R}^n, \forall \alpha \in [0, 1]$ справедливо

$$f(\alpha\mathbf{x} + (1-\alpha)\mathbf{y}) \leqslant \alpha f(\mathbf{x}) + (1-\alpha)f(\mathbf{y}).$$

Определение

Говорят, что $f(\mathbf{x}) \in \mathcal{F}_L^1$, если $f(\mathbf{x})$ является выпуклой функцией и обладает липшицевым градиентом с константой L.

Класс функций $\mathcal{S}^1_{\mu,\,L}$

Определение

Функция $f(\mathbf{x})$ называется *строго выпуклой*, если $\exists\,\mu > 0 : \forall\,\mathbf{y} \in \mathbb{R}^n$ функция $F(\mathbf{x}) = f(\mathbf{x}) - \frac{\mu}{2}\|\mathbf{x} - \mathbf{y}\|^2$ является выпуклой.
Число μ называется *параметром выпуклости* функции $f(\mathbf{x})$.

Замечание

При $\mu = 0$ получаем обычное определение выпуклости.

Определение

Говорят, что $f(\mathbf{x}) \in \mathcal{S}^1_{\mu,\,L}$, если $f(\mathbf{x})$ является строго выпуклой функцией с параметром выпуклости μ и обладает липшицевым градиентом с константой L.

Классы функций $\mathcal{F}_L^1$ и $\mathcal{S}_{\mu,L}^1$: свойства

Утверждение

$f(\mathbf{x}) \in \mathcal{F}_L^1$ *тогда и только тогда, когда* $\forall\, \mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ *справедливы оценки*

$$f(\mathbf{x}) \leqslant f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle + \frac{L}{2}\|\mathbf{x} - \mathbf{y}\|^2$$

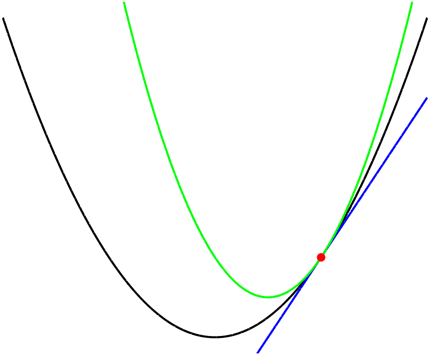
$$f(\mathbf{x}) \geqslant f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle$$

Утверждение

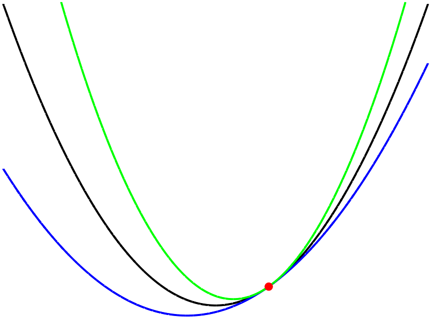
$f(\mathbf{x}) \in \mathcal{S}_{\mu,L}^1$ *тогда и только тогда, когда* $\forall\, \mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ *справедливы оценки*

$$f(\mathbf{x}) \leqslant f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle + \frac{L}{2}\|\mathbf{x} - \mathbf{y}\|^2$$

$$f(\mathbf{x}) \geqslant f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle + \frac{\mu}{2}\|\mathbf{x} - \mathbf{y}\|^2$$



$f(\mathbf{x}) \in \mathcal{F}_L^1$



$f(\mathbf{x}) \in \mathcal{S}_{\mu,L}^1$

Классы функций $\mathcal{F}_L^1$ и $\mathcal{S}_{\mu,L}^1$: критерий принадлежности

Утверждение

Пусть $f(\mathbf{x})$ *дважды непрерывно дифференцируемая функция. Тогда* $f(\mathbf{x}) \in \mathcal{F}_L^1$ *тогда и только тогда, когда* $\forall\, \mathbf{x} \in \mathbb{R}^n$ *выполнено*

$$\lambda_{min}(\nabla^2 f(\mathbf{x})) \geqslant 0,$$

$$\lambda_{max}(\nabla^2 f(\mathbf{x})) \leqslant L.$$

Утверждение

Пусть $f(\mathbf{x})$ *дважды непрерывно дифференцируемая функция. Тогда* $f(\mathbf{x}) \in \mathcal{S}_{\mu,L}^1$ *тогда и только тогда, когда* $\forall\, \mathbf{x} \in \mathbb{R}^n$ *выполнено*

$$\lambda_{min}(\nabla^2 f(\mathbf{x})) \geqslant \mu,$$

$$\lambda_{max}(\nabla^2 f(\mathbf{x})) \leqslant L.$$

(Здесь $\lambda_{min}(A)$ и $\lambda_{max}(A)$ — наименьшее и наибольшее собственные значения матрицы $A \succeq 0$.)

Классы функций $\mathcal{F}_L^1$ и $\mathcal{S}_{\mu,\,L}^1$: пример

Пример (квадратичная функция)

Рассмотрим квадратичную функцию

$$f(\mathbf{x}) = \frac{1}{2}\langle A\mathbf{x}, \mathbf{x}\rangle + \langle \mathbf{b}, \mathbf{x}\rangle + c, \quad A = A^T \succeq 0.$$

Гессиан этой функции в каждой точке $\mathbf{x} \in \mathbb{R}^n$ одинаков и равен

$$\nabla^2 f(\mathbf{x}) = A \succeq 0.$$

Таким образом, $f(\mathbf{x}) \in \mathcal{F}_L^1$, где $L = \lambda_{max}(A)$.
Более того, если $\lambda_{min}(A) > 0$, то $f(\mathbf{x}) \in \mathcal{S}_{\mu,\,L}^1$, где $\mu = \lambda_{min}(A)$, $L = \lambda_{max}(A)$.

План

1. Введение
   - Задачи и методы оптимизации
   - Основные теоретические сведения
2. Метод градиентного спуска
   - Стандартный метод для гладких функций
   - Метод для составных функций и автоматический подбор L
3. Быстрый градиентный метод
   - Метод для составных функций
   - Рестарт
   - Модификация подбора L и выбор L_0
4. Эксперименты и выводы
   - Эксперименты
   - Выводы

Метод градиентного спуска

Градиентный спуск

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \nabla f(\mathbf{x}_k), \quad k \geqslant 0$$

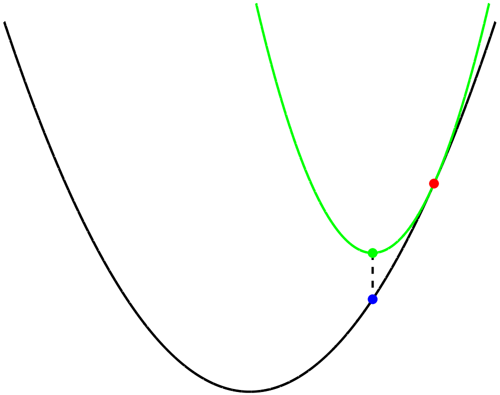
Стратегии выбора длины шага:

1. **Наискорейший спуск**:
$$\alpha_k = \min_{\alpha_k \in \mathbb{R}} f(\mathbf{x}_k - \alpha_k \nabla f(\mathbf{x}_k))$$
2. **Правило Гольдштейна**: найти α_k, такое что
$$f(\mathbf{x}_{k+1}) \leqslant f(\mathbf{x}_k) - \gamma \alpha_k \|\nabla f(\mathbf{x}_k)\|^2,$$
$$f(\mathbf{x}_{k+1}) \geqslant f(\mathbf{x}_k) - (1-\gamma) \alpha_k \|\nabla f(\mathbf{x}_k)\|^2,$$
где $0 < \gamma < 0.5$ — некоторая константа.
3. **Константный шаг**:
   - $\alpha_k = \frac{1}{L}$ для $f(\mathbf{x}) \in \mathcal{F}_L^1$;
   - $\alpha_k = \frac{2}{\mu + L}$ или $\alpha_k = \frac{1}{L}$ для $f(\mathbf{x}) \in \mathcal{S}_{\mu,\, L}^1$.

Метод градиентного спуска: иллюстрация

На самом деле, шаг градиентного метода есть минимизация простой квадратичной функции:

$$\mathbf{x}_{k+1} = \operatorname*{arg\,min}_{\mathbf{x}\in\mathbb{R}^n} \left[f(\mathbf{x}_k) + \langle \nabla f(\mathbf{x}_k), \mathbf{x} - \mathbf{x}_k \rangle + \frac{1}{2\alpha_k} \|\mathbf{x} - \mathbf{x}_k\|^2 \right]$$



Скорость сходимости: константный шаг

Теорема

Пусть $f(\mathbf{x}) \in \mathcal{F}_L^1$. *Тогда метод градиентного спуска с константным шагом* $\alpha_k \equiv \frac{1}{L}$ *имеет следующую скорость сходимости* ($k \geqslant 1$)*:*

$$f(\mathbf{x}_k) - f^* \leqslant \frac{2L}{k+4} \|\mathbf{x}_0 - \mathbf{x}^*\|^2.$$

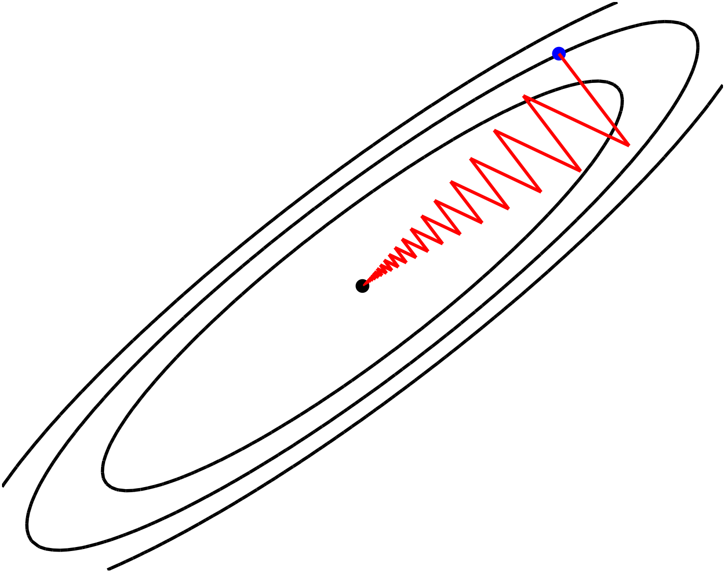
Теорема

Пусть $f(\mathbf{x}) \in \mathcal{S}_{\mu,L}^1$. *Тогда метод градиентного спуска с константным шагом* $\alpha_k \equiv \frac{2}{\mu+L}$ *имеет следующую скорость сходимости* ($k \geqslant 1$)*:*

$$f(\mathbf{x}_k) - f^* \leqslant \frac{L}{2} \left(1 - \frac{2}{Q+1}\right)^{2k} \|\mathbf{x}_0 - \mathbf{x}^*\|^2,$$

где $Q = \frac{L}{\mu} \geqslant 1$ *— число обусловленности функции* $f(\mathbf{x})$*.*

Скорость сходимости для класса $\mathcal{S}_{\mu, L}$: иллюстрация



Скорость сходимости: другие стратегии выбора шага

Для других стратегий выбора шага ситуация сильно не меняется.

Пример (наискорейший спуск для квадратичной задачи)

$$f(\mathbf{x}) = \frac{1}{2}\langle A\mathbf{x}, \mathbf{x}\rangle, \quad A = A^T \succeq 0.$$

Стратегия наискорейшего спуска:

$$\alpha_k = \min_{\alpha \in \mathbb{R}^n} f(\mathbf{x}_k - \alpha \nabla f(\mathbf{x}_k)).$$

Указанный минимум можно найти аналитически:

$$\alpha_k = \frac{\langle A^2\mathbf{x}_k, \mathbf{x}_k\rangle}{\langle A^3\mathbf{x}_k, \mathbf{x}_k\rangle}.$$

Можно показать, что

$$f(\mathbf{x}_k) - f^* \leqslant \left(1 - \frac{2}{Q+1}\right)^{2k} [f(\mathbf{x}_0) - f^*].$$

Скорость сходимости принципиально не изменилась!

План

1. Введение
   - Задачи и методы оптимизации
   - Основные теоретические сведения
2. Метод градиентного спуска
   - Стандартный метод для гладких функций
   - Метод для составных функций и автоматический подбор L
3. Быстрый градиентный метод
   - Метод для составных функций
   - Рестарт
   - Модификация подбора L и выбор L_0
4. Эксперименты и выводы
   - Эксперименты
   - Выводы

Составные функции

Определение

Составными функциями будем называть функции вида
$$F(\mathbf{x}) = f(\mathbf{x}) + h(\mathbf{x}),$$
где $f(\mathbf{x}) \in \mathcal{F}_L^1$, а $h(\mathbf{x})$ — некоторая простая (негладкая) выпуклая функция.

(Что значит «простая» станет понятно позже.)

Пример (логистическая регрессия с L_1-регуляризатором)

$$F(\mathbf{w}) = \underbrace{\frac{1}{m}\sum_{i=1}^{m} \ln(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i))}_{f(\mathbf{w})} + \underbrace{\tau\|\mathbf{w}\|_1}_{h(\mathbf{w})}, \quad \tau > 0$$

Как минимизировать составные функции?

1. Составные функции уже *не* являются гладкими.
2. Стандартный метод градиентного спуска уже *не* применим.
3. Можно использовать субградиентный спуск, однако в этом случае получаем скорость сходимости $O\left(\frac{1}{\sqrt{k}}\right)$, что существенно хуже, чем в гладком случае.
4. Мы увидим, что если правильно учесть структуру задачи и слегка модифицировать метод градиентного спуска, то получится метод со скоростью сходимости $O\left(\frac{1}{k}\right)$.

Градиентное отображение

Напомним, что в случае гладкой функции $f(\mathbf{x})$ метод градиентного спуска (с $\alpha_k \equiv 1/L$) делал итерации вида

$$\mathbf{x}_{k+1} = \operatorname*{arg\,min}_{\mathbf{x}\in\mathbb{R}^n} \left[f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle + \frac{L}{2} \|\mathbf{x} - \mathbf{y}\|^2 \right],$$

т. е. на каждой итерации минимизировал верхнюю квадратичную оценку на $f(\mathbf{x})$.

Построим аналог такой итерации для составных функций.

Определение

Градиентным отображением $T_L(\mathbf{y})$ точки $\mathbf{y} \in \mathbb{R}^n$ называется точка

$$T_L(\mathbf{y}) = \operatorname*{arg\,min}_{\mathbf{x}\in\mathbb{R}^n} \left[f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle + \frac{L}{2} \|\mathbf{x} - \mathbf{y}\|^2 + h(\mathbf{x}) \right].$$

Градиентное отображение: пример

Пример (L_1-регуляризатор)

Пусть $g(\mathbf{x}) = \tau\|\mathbf{x}\|_1$, $\tau > 0$.
Тогда

$$T_L(\mathbf{y}) = \arg\min_{\mathbf{x}\in\mathbb{R}^n}\left[f(\mathbf{y}) + \langle\nabla f(\mathbf{y}), \mathbf{x}-\mathbf{y}\rangle + \frac{L}{2}\|\mathbf{x}-\mathbf{y}\|^2 + \tau\|\mathbf{x}\|_1\right].$$

Данный минимум можно найти аналитически:

$$T_L(\mathbf{y}) = \mathcal{V}_{\tau/L}\left(\mathbf{y} - \frac{1}{L}\nabla f(\mathbf{y})\right),$$

где $\mathcal{V}_\alpha^{(j)} = \max\left(|x^{(j)}| - \alpha, 0\right)\operatorname{sgn}\left(x^{(j)}\right)$, $j = 1, \ldots, n$ — *сжимающий* оператор.

Градиентный спуск для составных функций

Схема метода

Вход: $\mathbf{x}_0 \in \mathbb{R}^n$, $L_f > 0$;

для $k = 0, 1, 2, \ldots$

  {вычислить $f(\mathbf{x}_k)$, $\nabla f(\mathbf{x}_k)$};

  $\mathbf{x}_{k+1} := T_{L_f}(\mathbf{x}_k)$; // вместо $\mathbf{x}_{k+1} := \mathbf{x}_k - \frac{1}{L}\nabla f(\mathbf{x}_k)$

Обсуждение

1. Что если константа Липшица L_f для функции $f(\mathbf{x})$ нам неизвестна?
2. Более того, что если функция в разных областях имеет сильно разные константы Липшица? В этом случае глобальная константа L_f локально будет плохо аппроксимировать изгиб функции. В результате шаги будут существенно меньше, чем они могли бы быть.
3. Какому основному условию должна удовлетворять локальная константа Липшица L_k, чтобы метод по-прежнему хорошо работал? Оказывается, достаточно потребовать всего лишь знакомой верхней оценки:

$$f(\mathbf{x}_{k+1}) \leqslant f(\mathbf{x}_k) + \langle \nabla f(\mathbf{x}_k), \mathbf{x}_{k+1} - \mathbf{x}_k \rangle + \frac{L_k}{2}\|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2$$

Градиентный спуск с автоматическим подбором L_k

Вход: $\mathbf{x}_0 \in \mathbb{R}^n$, $L_0 > 0$, $\gamma_u > 1$, $\gamma_d \geqslant 1$;

для $k = 0, 1, 2, \ldots$
- {вычислить $f(\mathbf{x}_k)$, $\nabla f(\mathbf{x}_k)$};
- $L := L_k$;
- **повторять**
  - $\mathbf{T} := T_L(\mathbf{x}_k)$;
  - {вычислить $f(\mathbf{T})$};
  - **если** $f(\mathbf{T}) > f(\mathbf{x}_k) + \langle \nabla f(\mathbf{x}_k), \mathbf{T} - \mathbf{x}_k \rangle + \frac{L}{2}\|\mathbf{T} - \mathbf{x}_k\|^2$ **то**
    - $L := \gamma_u L$;
- **пока неверно, что** $f(\mathbf{T}) \leqslant f(\mathbf{x}_k) + \langle \nabla f(\mathbf{x}_k), \mathbf{T} - \mathbf{x}_k \rangle + \frac{L}{2}\|\mathbf{T} - \mathbf{x}_k\|^2$;
- $\mathbf{x}_{k+1} := \mathbf{T}$;
- $L_{k+1} := \max\left(L_0, \frac{L}{\gamma_d}\right)$;

Количество вычислений функции

Теорема

Пусть N_k — количество вычислений функции $f(\mathbf{x})$ за первые k итераций градиентного спуска. Тогда

$$N_k \leqslant \left(1 + \frac{\ln \gamma_d}{\ln \gamma_u}\right)(k+1) + \frac{1}{\ln \gamma_u} \max\left(\ln \frac{\gamma_u L_f}{\gamma_d L_0},\, 0\right)$$

Например, если $\gamma_d = 1, \gamma_u = 2$, то

$$N_k \leqslant (k+1) + \log_2 \frac{2L_f}{L_0},$$

т. е. среднее число вычислений функции за одну итерацию равно 1.

На практике хорошие значения $\gamma_u = 2$, $\gamma_d = 1.1$.

Скорость сходимости

Теорема

Градиентный спуск для составных функций имеет следующую скорость сходимости:

$$F(\mathbf{x}_k) - F^* \leqslant \frac{2\gamma_u L_f}{k+2} \|\mathbf{x}_0 - \mathbf{x}^*\|^2$$

Более того, если $F(\mathbf{x})$ является строго выпуклой с параметром выпуклости μ_F, то гарантируется следующая скорость:

$$F(\mathbf{x}_k) - F^* \leqslant \left(1 - \frac{\mu_F}{4\gamma_u L_f}\right)^k [F(\mathbf{x}_0) - F^*].$$

Замечание: Интересной особенностью является то, что методу не нужно заранее знать константу μ_F, чтобы гарантировать последнее неравенство.

План

1 Введение
- Задачи и методы оптимизации
- Основные теоретические сведения

2 Метод градиентного спуска
- Стандартный метод для гладких функций
- Метод для составных функций и автоматический подбор L

3 Быстрый градиентный метод
- Метод для составных функций
- Рестарт
- Модификация подбора L и выбор L_0

4 Эксперименты и выводы
- Эксперименты
- Выводы

В дальнейшем мы будем рассматривать задачи вида

$$\min_{\mathbf{x} \in \mathbb{R}^n} F(\mathbf{x}),$$

где $F(\mathbf{x})$ — составная функция, т. е.

$$F(\mathbf{x}) = f(\mathbf{x}) + h(\mathbf{x}),$$

где $f(\mathbf{x}) \in \mathcal{F}^1_{L_f}$ и $h(\mathbf{x})$ — некоторая простая (негладкая) выпуклая функция.

Под «простотой» функции $h(\mathbf{x})$ подразумевается, что мы легко можем вычислить градиентное отображение

$$T_L(\mathbf{y}) = \arg\min_{\mathbf{x} \in \mathbb{R}^n} \left[f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle + \frac{L}{2} \|\mathbf{x} - \mathbf{y}\|^2 + h(\mathbf{x}) \right].$$

Оценочная последовательность

Определение

Оценочной последовательностью для функции $F(\mathbf{x})$ называется тройка, состоящая из

- *минимизирующей последовательности* $\{\mathbf{x}_k\}_{k=0}^{\infty}$,
- последовательности *масштабирующих коэффициентов* $\{A_k\}_{k=0}^{\infty}$,
- последовательности *оценочных функций* $\{\psi_k(\mathbf{x})\}_{k=0}^{\infty}$,

обеспечивающих выполнение следующих двух отношений $\forall\, k \geqslant 0$:

$$\mathcal{R}_k^1: \quad A_k F(\mathbf{x}_k) \leqslant \psi_k^* \equiv \min_{\mathbf{x}\in\mathbb{R}^n} \psi_k(\mathbf{x}),$$

$$\mathcal{R}_k^2: \quad \psi_k(\mathbf{x}) \leqslant A_k F(\mathbf{x}) + \psi_0(\mathbf{x}), \quad \forall\, \mathbf{x} \in \mathbb{R}^n.$$

(Подразумевается, что $A_k > 0$ для $k \geqslant 1$.)

Функция $\psi_0(\mathbf{x})$ называется *проксимальной* функцией.

Оценочная последовательность: скорость сходимости

Если отношения $\mathcal{R}_k^1$ и $\mathcal{R}_k^2$ выполняются $\forall\, k \geqslant 0$, то получаем оценку скорости сходимости:

$$F(\mathbf{x}_k) - F^* \leqslant \frac{1}{A_k}\psi_0(\mathbf{x}^*), \quad k \geqslant 1.$$

Например, если выбрать $\psi_0(\mathbf{x}) = \frac{1}{2}\|\mathbf{x} - \mathbf{x}_0\|^2$, то

$$F(\mathbf{x}_k) - F^* \leqslant \frac{1}{2A_k}\|\mathbf{x}^* - \mathbf{x}_0\|^2, \quad k \geqslant 1.$$

Таким образом, скорость сходимости определяется тем, насколько быстро растут масштабирующие коэффициенты A_k.
Далее мы построим такую оценочную последовательность, что масштабирующие коэффициенты A_k будут расти как $O(k^2)$.

Оценочная последовательность: построение – 1

Будем искать оценочные функции $\psi_k(\mathbf{x})$ и масштабирующие коэффициенты A_k в виде ($k \geqslant 1$)

$$\psi_k(\mathbf{x}) = \psi_{k-1}(\mathbf{x}) + a_k \underbrace{[f(\mathbf{x}_k) + \langle \nabla f(\mathbf{x}_k), \mathbf{x} - \mathbf{x}_k \rangle + h(\mathbf{x})]}_{\text{нижняя оценка на функцию } F(\mathbf{x})},$$

$$A_k = A_{k-1} + a_k,$$

где $A_0 = 0$; коэффициенты $a_k > 0$, $\forall\, k \geqslant 1$ контролируют скорость роста масштабирующих коэффициентов.

Такой выбор сразу же обеспечивает выполнение отношения $\mathcal{R}_k^2$ (доказывается по индукции).

Оценочная последовательность: построение – 2

1. Зафиксировав конкретный вид оценочных функций $\psi_k(\mathbf{x})$, мы обеспечили выполнение отношения $\mathcal{R}_k^2$, $\forall\, k \geqslant 0$.
2. Свободные параметры оценочной последовательности:
   - минимизирующая последовательность $\{\mathbf{x}_k\}_{k=0}^{\infty}$;
   - коэффициенты роста масштабирующих коэффициентов $\{a_k\}_{k=1}^{\infty}$;
   - проксимальная функция $\psi_0(\mathbf{x})$.
3. Далее мы обеспечим выполнение оставшегося отношения $\mathcal{R}_k^1$, задействовав все свободные на текущий момент параметры, и получим конкретную схему быстрого градиентного метода.

Оценочная последовательность: построение – 3

Выберем проксимальную функцию $\psi_0(\mathbf{x}) = \frac{1}{2}\|\mathbf{x} - \mathbf{x}_0\|^2$.
Тогда

$$\psi_k(\mathbf{x}) = \underbrace{\frac{1}{2}\|\mathbf{x} - \mathbf{x}_0\|^2}_{\text{строго выпуклая}} + \sum_{i=1}^{k} a_i \underbrace{[f(\mathbf{x}_i) + \langle \nabla f(\mathbf{x}_i), \mathbf{x} - \mathbf{x}_i \rangle + h(\mathbf{x})]}_{\text{выпуклая}},$$

т. е. $\psi_k(\mathbf{x})$ является строго выпуклой функцией с параметром выпуклости 1.
Тогда

$$\psi_k(\mathbf{x}) \geqslant \psi_k^* + \frac{1}{2}\|\mathbf{x} - \mathbf{v}_k\|^2,$$

где $\mathbf{v}_k = \arg\min_{\mathbf{x} \in \mathbb{R}^n} \psi_k(\mathbf{x})$.

Оценочная последовательность: построение – 4

Оценочная функция:

$$\psi_{k+1}(\mathbf{x}) = \psi_k(\mathbf{x}) + a_{k+1}\left[f(\mathbf{x}_{k+1}) + \langle \nabla f(\mathbf{x}_{k+1}), \mathbf{x} - \mathbf{x}_{k+1}\rangle + h(\mathbf{x})\right].$$

Пусть выполняется отношение $\mathcal{R}_k^1$. Можно показать, что в этом случае

$$\underbrace{\psi_{k+1}^* \geqslant A_{k+1} F(\mathbf{x}_{k+1})}_{\text{отношение } \mathcal{R}_{k+1}^1} + \\ + A_{k+1}\left[\langle F'(\mathbf{x}_{k+1}), \mathbf{y}_k - \mathbf{x}_{k+1}\rangle - \frac{a_{k+1}^2}{2A_{k+1}}\|F'(\mathbf{x}_{k+1})\|^2\right],$$

где

$$\mathbf{y}_k = \frac{A_k \mathbf{x}_k + a_{k+1}\mathbf{v}_k}{A_k + a_{k+1}},$$

а $F'(\mathbf{x}_{k+1})$ — любой субградиент функции $F(\mathbf{x})$ в точке $\mathbf{x}_{k+1}$.

Оценочная последовательность: построение – 5

Осталось выбором a_{k+1} и $\mathbf{x}_{k+1}$ добиться выполнения неравенства

$$\langle F'(\mathbf{x}_{k+1}), \mathbf{y}_k - \mathbf{x}_{k+1} \rangle \geqslant \frac{a_{k+1}^2}{2A_{k+1}} \|F'(\mathbf{x}_{k+1})\|^2.$$

Это можно сделать с помощью градиентного шага.

Лемма

$\forall\, L \geqslant L_f$ и $\mathbf{T} = T_L(\mathbf{y})$ верно следующее неравенство:

$$\langle F'(\mathbf{T}), \mathbf{y} - \mathbf{T} \rangle \geqslant \frac{1}{L} \|F'(\mathbf{T})\|^2,$$

где $F'(\mathbf{T}) = L(\mathbf{y} - \mathbf{T}) - [\nabla f(\mathbf{y}) - \nabla f(\mathbf{T})]$ — конкретный субградиент в точке $\mathbf{T} = T_L(\mathbf{y})$.

Напомним, что

$$T_L(\mathbf{y}) = \arg\min_{\mathbf{x} \in \mathbb{R}^n} \left[f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle + \frac{L}{2} \|\mathbf{x} - \mathbf{y}\|^2 + h(\mathbf{x}) \right].$$

Быстрый градиентный метод для составных функций

Вход: $\mathbf{x}_0 \in \mathbb{R}^n$, $0 < L_0 \leqslant L_f$, $\gamma_u > 1$, $\gamma_d \geqslant 1$;

$A_0 := 0$;

для $k = 0, 1, 2, \ldots$

  $L := L_k$;

  повторять

    {найти a из уравнения $\frac{a^2}{2(A_k+a)} = \frac{1}{L}$};

    {вычислить $\mathbf{v}_k = \arg\min_{\mathbf{x}\in\mathbb{R}^n} \psi_k(\mathbf{x})$};

    $\mathbf{y} := \frac{A_k \mathbf{x}_k + a\mathbf{v}_k}{A_k + a}$; {вычислить $\nabla f(\mathbf{y})$};

    $\mathbf{T} := T_L(\mathbf{y})$; {вычислить $\nabla f(\mathbf{T})$};

    $s := L(\mathbf{y} - \mathbf{T}) - [\nabla f(\mathbf{y}) - \nabla f(\mathbf{T})]$;

    если $\langle s, \mathbf{y} - \mathbf{T}\rangle < \frac{1}{L}\|s\|^2$ **то** $L := \gamma_u L$;

  пока неверно, что $\langle s, \mathbf{y} - \mathbf{T}\rangle \geqslant \frac{1}{L}\|s\|^2$;

  $\mathbf{x}_{k+1} := \mathbf{T}$; $A_{k+1} := A_k + a$; $L_{k+1} := \max\left(L_0, \frac{L}{\gamma_d}\right)$;

Быстрый градиентный метод: простая схема

Если оптимизируемая функция $F(\mathbf{x})$ гладкая (т. е. $F(\mathbf{x}) \equiv f(\mathbf{x})$) с известной константой Липшица для градиента L_f, то схема упрощается.

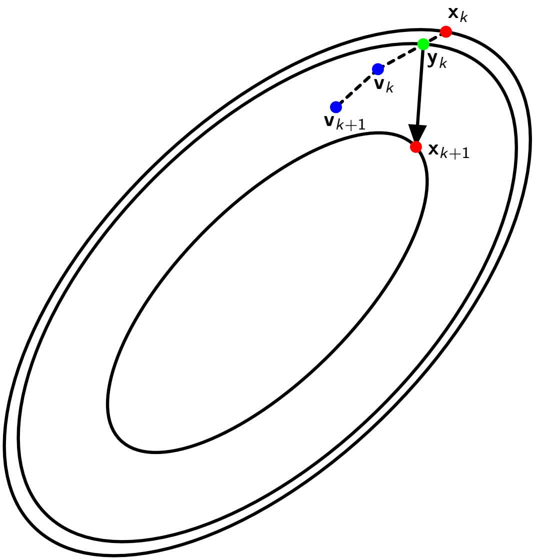
Вход: $\mathbf{x}_0 \in \mathbb{R}^n$;

$\mathbf{v}_0 := \mathbf{x}_0$; $A_0 := 0$;

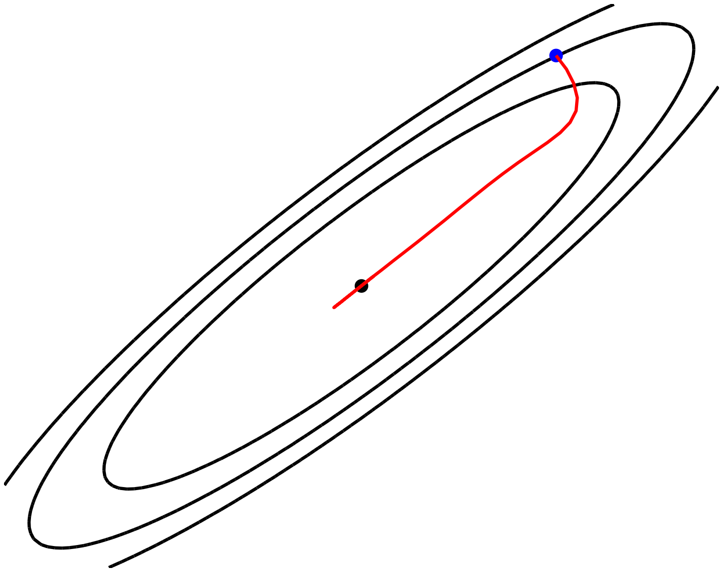
для $k = 0, 1, 2, \ldots$

- {найти a из уравнения $\frac{a^2}{2(A_k+a)} = \frac{1}{L_f}$};
- $\mathbf{y} := \frac{A_k \mathbf{x}_k + a\mathbf{v}_k}{A_k+a}$; {вычислить $\nabla f(\mathbf{y})$};
- $\mathbf{x}_{k+1} := \mathbf{y}_k - \frac{1}{L_f}\nabla f(\mathbf{y}_k)$; {вычислить $\nabla f(\mathbf{x}_{k+1})$};
- $\mathbf{v}_{k+1} := \mathbf{v}_k - a\nabla f(\mathbf{x}_{k+1})$;
- $A_{k+1} := A_k + a$;

Одна итерация метода: иллюстрация



Траектория метода



Скорость сходимости – 1

Напомним, что параметры роста a_k масштабирующих коэффициентов A_k находятся из уравнения

$$\frac{a_k^2}{2(A_k + a_k)} = \frac{1}{L}.$$

Из этого уравнения и того, что $L \leqslant \gamma_u L_f$ получаем следующую оценку на скорость роста масштабирующих коэффициентов.

Лемма

Для масштабирующих коэффициентов A_k справедлива оценка

$$A_k \geqslant \frac{k^2}{2\gamma_u L_f}, \quad k \geqslant 0.$$

Скорость сходимости – 2

Теорема

Быстрый градиентный метод для составных функций имеет следующую скорость сходимости:

$$F(\mathbf{x}_k) - F^* \leqslant \frac{\gamma_u L_f}{k^2} \|\mathbf{x}_0 - \mathbf{x}^*\|^2$$

Напомним, что скорость сходимости обычного градиентного спуска на порядок хуже:

$$F(\mathbf{x}_k) - F^* \leqslant \frac{2\gamma_u L_f}{k+2} \|\mathbf{x}_0 - \mathbf{x}^*\|^2.$$

План

1. Введение
   - Задачи и методы оптимизации
   - Основные теоретические сведения
2. Метод градиентного спуска
   - Стандартный метод для гладких функций
   - Метод для составных функций и автоматический подбор L
3. Быстрый градиентный метод
   - Метод для составных функций
   - Рестарт
   - Модификация подбора L и выбор L_0
4. Эксперименты и выводы
   - Эксперименты
   - Выводы

Что насчет строго выпуклых функций?

1. Для строго выпуклой функции $F(\mathbf{x})$ с параметром выпуклости μ_F обычный градиентный спуск имеет следующую скорость сходимости:
$$F(\mathbf{x}_k) - F^* \leqslant \left(1 - \frac{\mu_F}{4\gamma_u L_f}\right)^k [F(\mathbf{x}_0) - F^*] .$$
При этом методу совсем не требуется знание константы μ_F.
2. Можно ли гарантировать подобный результат для быстрого градиентного метода? Оказывается, что можно. Но для этого необходима модификация метода.
3. Далее мы рассмотрим технику т. н. *рестарта*, которая потребует знания константы μ_F. Затем мы откажемся от этого требования благодаря стратегии *адаптивного рестарта*.

Рестарт

Пусть функция $F(\mathbf{x})$ является строго выпуклой с параметром выпуклости μ_F.

Вспомним, что для быстрого градиентного метода справедлива следующая оценка скорости сходимости ($k \geqslant 0$):

$$F(\mathbf{x}_k) - F^* \leqslant \frac{\gamma_u L_f}{k^2} \|\mathbf{x}_0 - \mathbf{x}^*\|^2.$$

Из данной оценки и строгой выпуклости $F(\mathbf{x})$ вытекает, что $\forall k \geqslant 0$

$$F(\mathbf{x}_k) - F^* \leqslant \frac{2\gamma_u L_f}{\mu_F k^2} [F(\mathbf{x}_0) - F^*].$$

Если положить $k = \left\lceil 2\sqrt{\frac{\gamma_u L_f}{\mu_F}} \right\rceil \equiv N$, то получим

$$F(\mathbf{x}_N) - F^* \leqslant \frac{1}{2} [F(\mathbf{x}_0) - F^*].$$

Рестарт – 2

Мы получили, что для $N = \left\lceil 2\sqrt{\frac{\gamma_u L_f}{\mu_F}} \right\rceil$ справедлива оценка

$$F(\mathbf{x}_N) - F^* \leqslant \frac{1}{2}\left[F(\mathbf{x}_0) - F^*\right],$$

т. е. за N итераций быстрый градиентный метод уменьшает невязку по значению функции как минимум вдвое.

Если после N итераций метода положить $\mathbf{x}_0 = \mathbf{x}_N$, т. е. *перезапустить* метод из новой начальной точки $\mathbf{x}_N$, то получим что

$$F(\mathbf{x}_{2N}) - F^* \leqslant \frac{1}{2}\left[F(\mathbf{x}_N) - F^*\right] \leqslant \frac{1}{4}\left[F(\bar{\mathbf{x}}_0) - F^*\right],$$

где $\bar{\mathbf{x}}_0$ — исходная начальная точка.
Невязка уменьшилась уже в четыре раза!

Получаем геометрическую прогрессию:

$$F(\mathbf{x}_{KN}) - F^* \leqslant \frac{1}{2^K}\left[F(\bar{\mathbf{x}}_0) - F^*\right].$$

Рестарт: алгоритм

Будем обозначать $\mathcal{A}_N(\mathbf{u})$ точку, полученную быстрым градиентным методом за N итераций из начального приближения $\mathbf{u}$.

Для реализации рестарта можно использовать следующую двухуровневую схему.

Процедура рестарта

Вход: $\mathbf{x}_0 \in \mathbb{R}^n$, N — частота рестартов;

1 $\mathbf{u}_0 = \mathbf{x}_0$;

2 **для** $k = 0, 1, 2, \ldots$

3 $\quad \mathbf{u}_{k+1} := \mathcal{A}_N(\mathbf{u}_k)$;

Вместо двухуровневой схемы можно интегрировать рестарт внутрь метода (следующий слайд).

Быстрый градиентный метод с рестартом

Вход: $\mathbf{x}_0 \in \mathbb{R}^n$, $0 < L_0 \leqslant L_f$, $\gamma_u > 1$, $\gamma_d \geqslant 1$, N — частота рестартов;
$A_0 := 0$;
для $k = 0, 1, 2, \ldots$
  $L := L_k$;
  повторять
    {найти a из уравнения $\frac{a^2}{2(A_k+a)} = \frac{1}{L}$};
    {вычислить $\mathbf{v}_k = \arg\min_{\mathbf{x}\in\mathbb{R}^n} \psi_k(\mathbf{x})$};
    $\mathbf{y} := \frac{A_k\mathbf{x}_k + a\mathbf{v}_k}{A_k+a}$; {вычислить $\nabla f(\mathbf{y})$};
    $\mathbf{T} := T_L(\mathbf{y})$; {вычислить $\nabla f(\mathbf{T})$};
    $s := L(\mathbf{y} - \mathbf{T}) - [\nabla f(\mathbf{y}) - \nabla f(\mathbf{T})]$;
    если $\langle s, \mathbf{y} - \mathbf{T}\rangle < \frac{1}{L}\|s\|^2$ **то** $L := \gamma_u L$;
  пока неверно, что $\langle s, \mathbf{y} - \mathbf{T}\rangle \geqslant \frac{1}{L}\|s\|^2$;
  $\mathbf{x}_{k+1} := \mathbf{T}$; $A_{k+1} := A_k + a$; $L_{k+1} := \max\left(L_0, \frac{L}{\gamma_d}\right)$;
  если $N \mid k$ **то** $\mathbf{x}_{k+1} := \mathbf{x}_k$; $A_{k+1} := 0$;

Рестарт: скорость сходимости

Утверждение

Для генерации ε-решения по значению функции быстрому градиентному методу с рестартом достаточно $O\left(\sqrt{\frac{L_f}{\mu_F}} \ln \frac{1}{\varepsilon}\right)$ *итераций.*

Заметим, что гарантия обычного градиентного спуска составляет $O\left(\frac{L_f}{\mu_F} \ln \frac{1}{\varepsilon}\right)$ итераций, что значительно хуже.

Константы L_f и μ_F неизвестны

1. Рестарт гарантирует хорошую скорость сходимости для строго выпуклых функций, однако он требует знания констант L_f и μ_F, которые на практике редко известны.
2. Как отказаться от требования знания констант L_f и μ_F?
3. Можно перезапускать метод, например, каждые $N = 100$, $N = 200$ или $N = 500$ итераций. Главный недостаток такой стратегии состоит в том, что нет никаких правил для выбора N.
4. Обязательно ли делать рестарт с одинаковым периодом N? Почему бы не заменить условие рестарта «$N \mid k$» на какое-нибудь другое?
5. Далее мы увидим, что можно делать т. н. *адаптивный рестарт*, который не требует знания констант L_f и μ_F.

Предлагается в качестве условия рестарта использовать следующее.

Градиентное условие рестарта

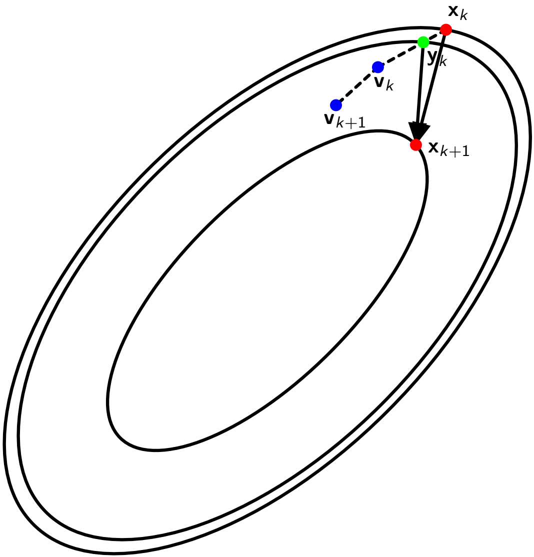
$$\langle g_L(\mathbf{y}_k), \mathbf{x}_{k+1} - \mathbf{x}_k \rangle > 0,$$

где $g_L(\mathbf{y}) = L(\mathbf{y} - T_L(\mathbf{y}))$ — аналог градиента для составных функций.

Таким образом, если переход от точки $\mathbf{x}_k$ к точке $\mathbf{x}_{k+1}$ произошел по направлению убывания в точке $\mathbf{y}_k$, то все хорошо. Иначе рестарт.

Замечание: Данное условие является скорее эвристикой, чем фундаментальной научной идеей. Возможны и другие (похожие) условия рестарта.

Адаптивный рестарт: иллюстрация



Алгоритм

Вход: $\mathbf{x}_0 \in \mathbb{R}^n$, $0 < L_0 \leqslant L_f$, $\gamma_u > 1$, $\gamma_d \geqslant 1$;

$A_0 := 0$;

для $k = 0, 1, 2, \ldots$

  $L := L_k$;

  повторять

    {найти a из уравнения $\frac{a^2}{2(A_k+a)} = \frac{1}{L}$};

    {вычислить $\mathbf{v}_k = \arg\min_{\mathbf{x}\in\mathbb{R}^n} \psi_k(\mathbf{x})$};

    $\mathbf{y} := \frac{A_k\mathbf{x}_k + a\mathbf{v}_k}{A_k+a}$; {вычислить $\nabla f(\mathbf{y})$};

    $\mathbf{T} := T_L(\mathbf{y})$; {вычислить $\nabla f(\mathbf{T})$};

    $s := L(\mathbf{y} - \mathbf{T}) - [\nabla f(\mathbf{y}) - \nabla f(\mathbf{T})]$;

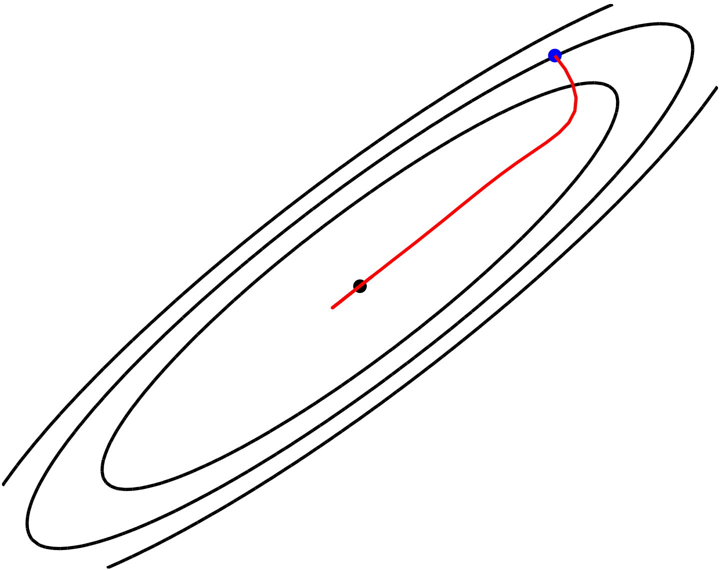
    если $\langle s, \mathbf{y} - \mathbf{T}\rangle < \frac{1}{L}\|s\|^2$ **то** $L := \gamma_u L$;

  пока неверно, что $\langle s, \mathbf{y} - \mathbf{T}\rangle \geqslant \frac{1}{L}\|s\|^2$;

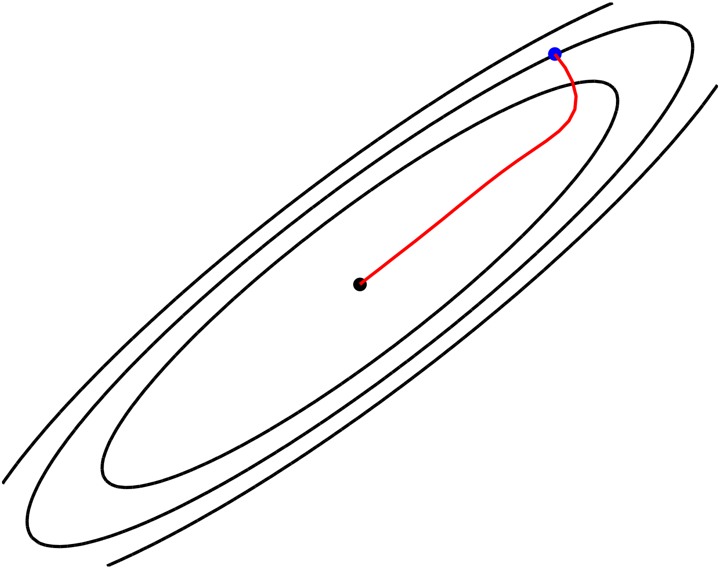
  $\mathbf{x}_{k+1} := \mathbf{T}$; $A_{k+1} := A_k + a$; $L_{k+1} := \max\left(L_0, \frac{L}{\gamma_d}\right)$;

  если $\langle \mathbf{y} - \mathbf{x}_{k+1}, \mathbf{x}_{k+1} - \mathbf{x}_k\rangle > 0$ **то** $\mathbf{x}_{k+1} := \mathbf{x}_k$; $A_{k+1} := 0$;

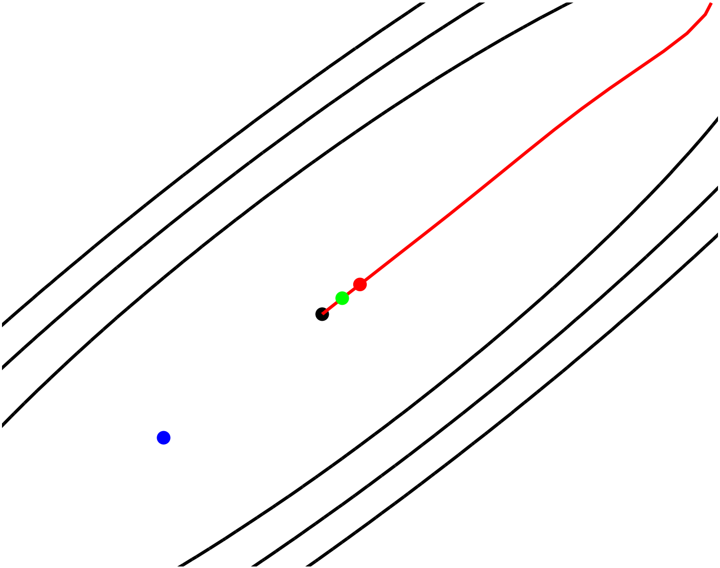
Траектория метода без рестарта



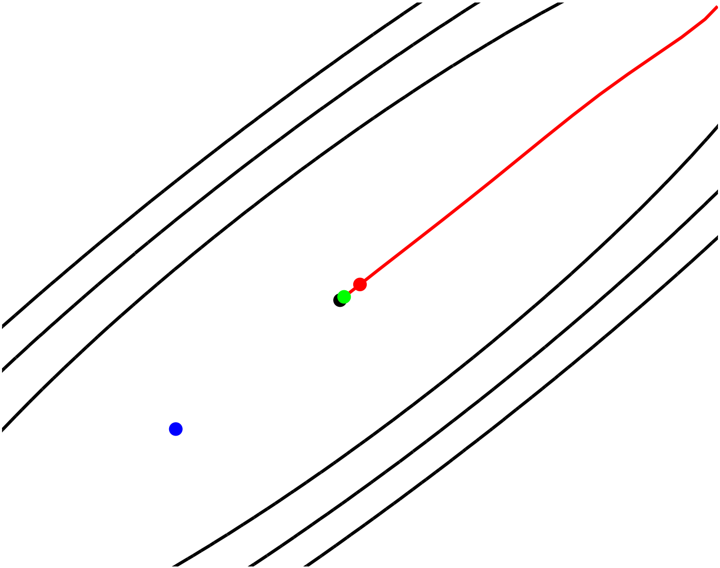
Траектория метода с рестартом



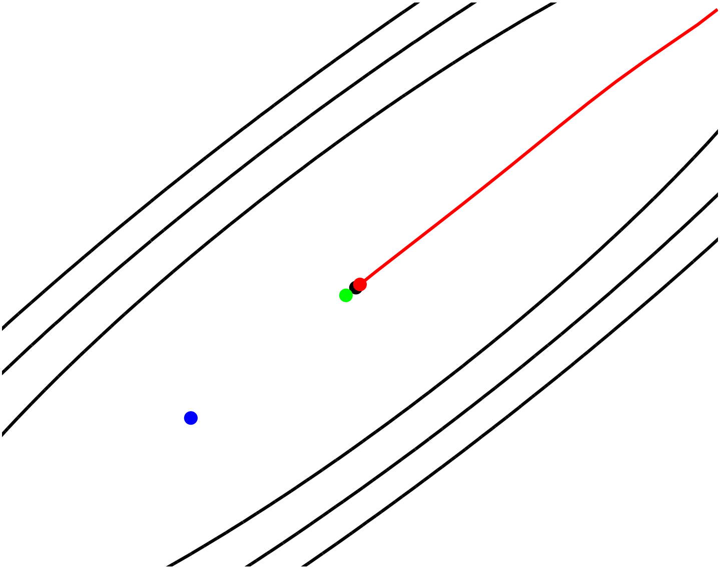
Траектория метода с рестартом ($k = 20$)



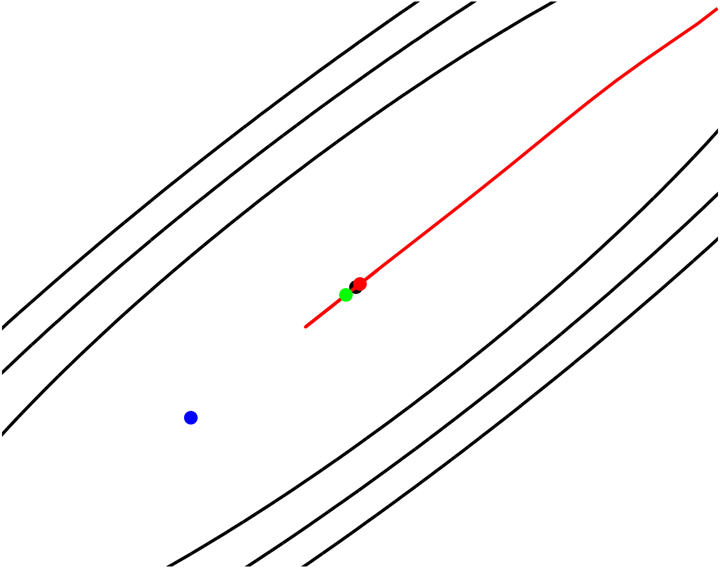
Траектория метода с рестартом ($k = 21$)



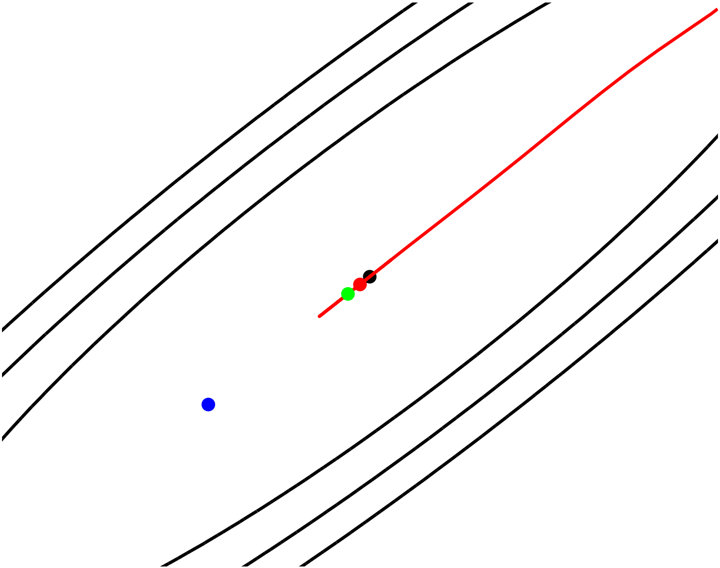
Траектория метода с рестартом ($k = 22$): рестарт



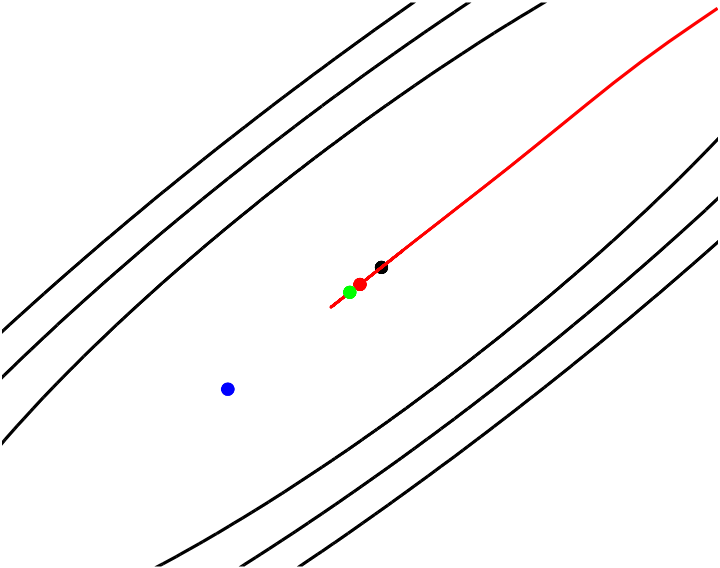
Траектория метода без рестарта ($k = 22$)



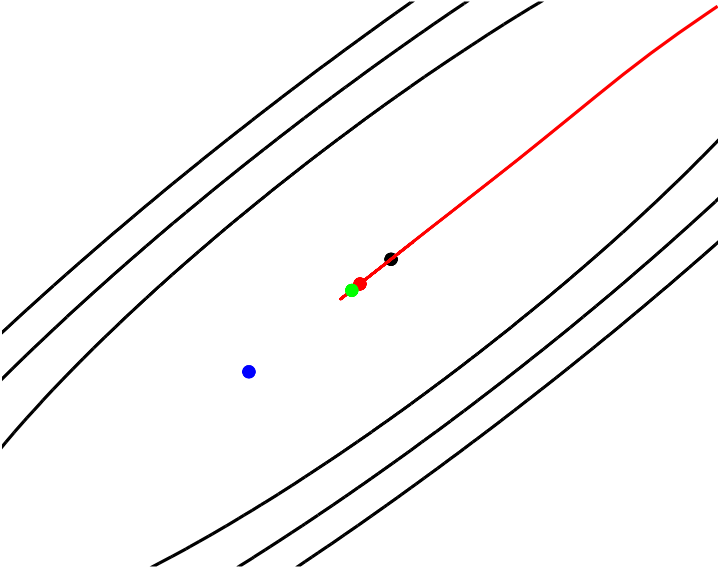
Траектория метода без рестарта ($k = 23$): мимо



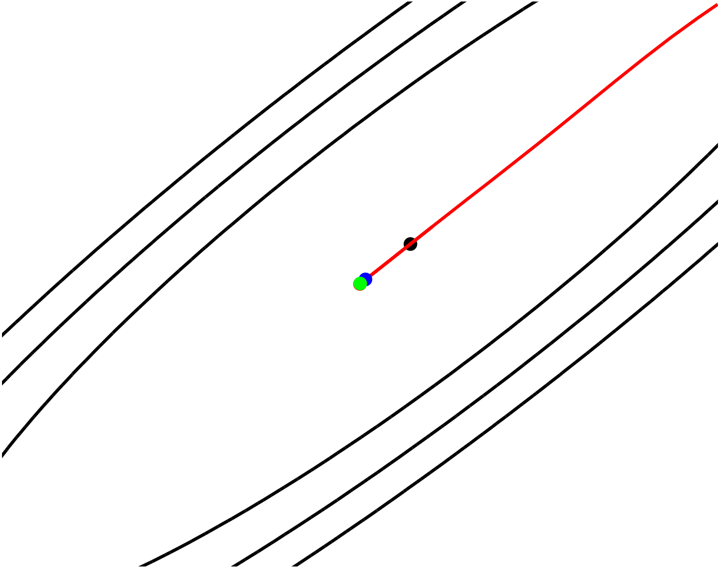
Траектория метода без рестарта ($k = 24$)



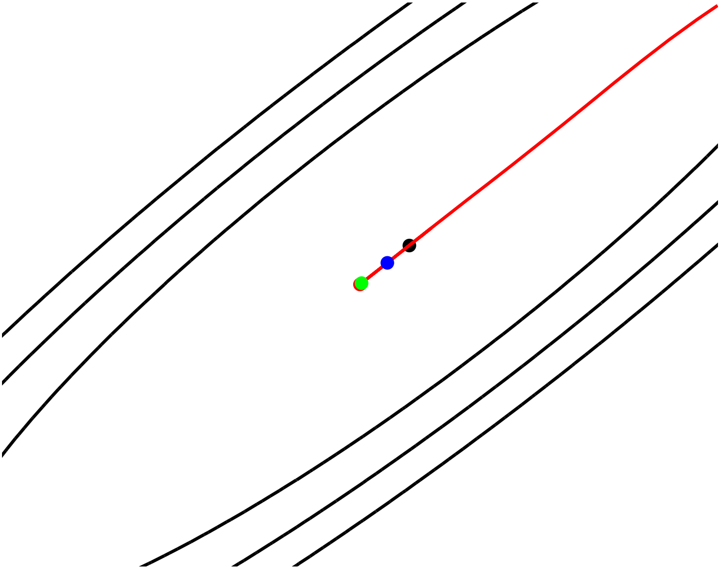
Траектория метода без рестарта ($k = 25$)



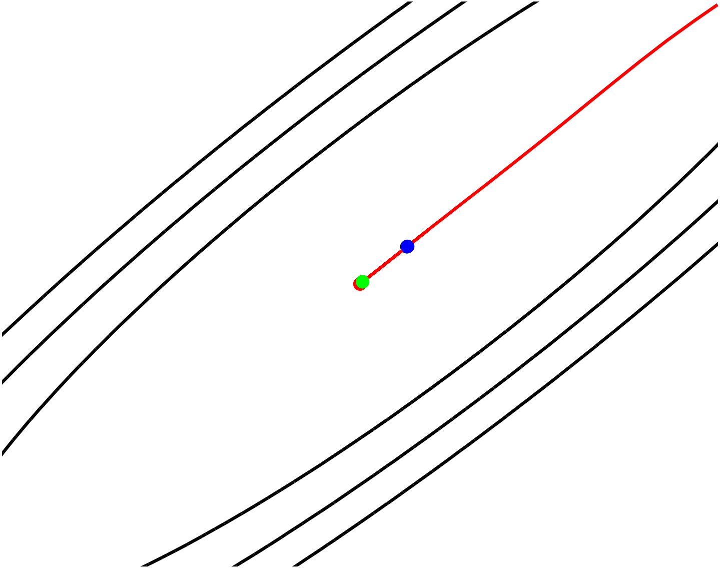
Траектория метода без рестарта ($k = 30$): разворот



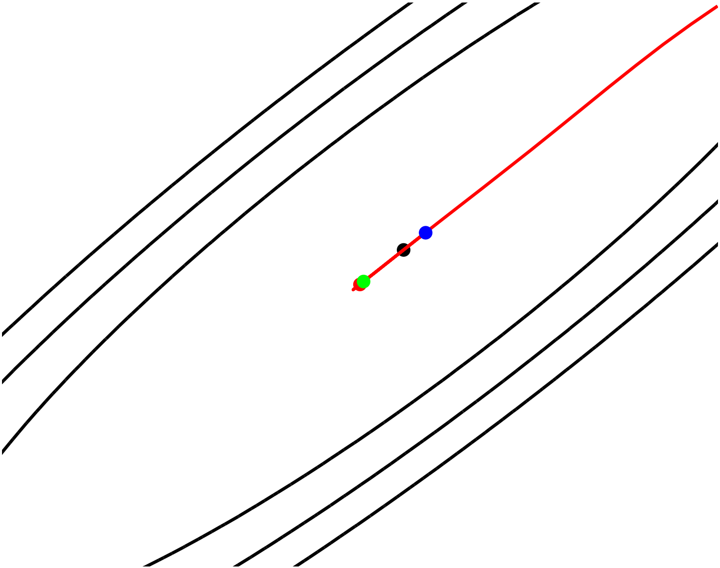
Траектория метода без рестарта ($k = 31$): обратно



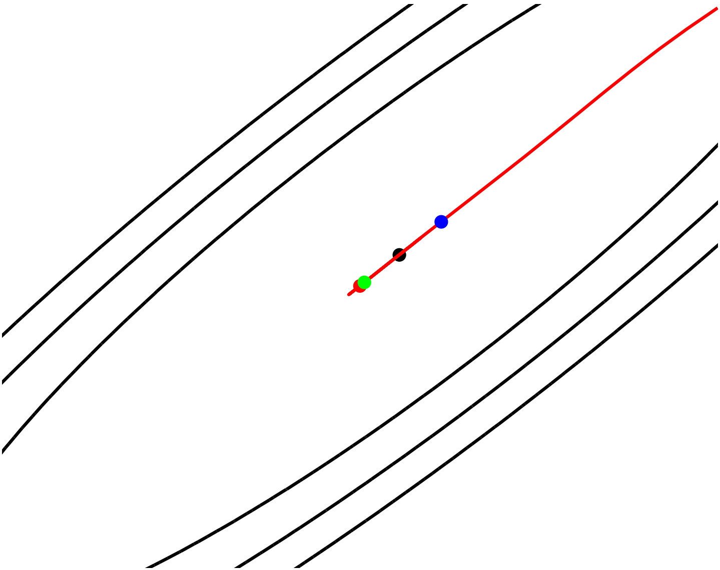
Траектория метода без рестарта ($k = 32$): обратно



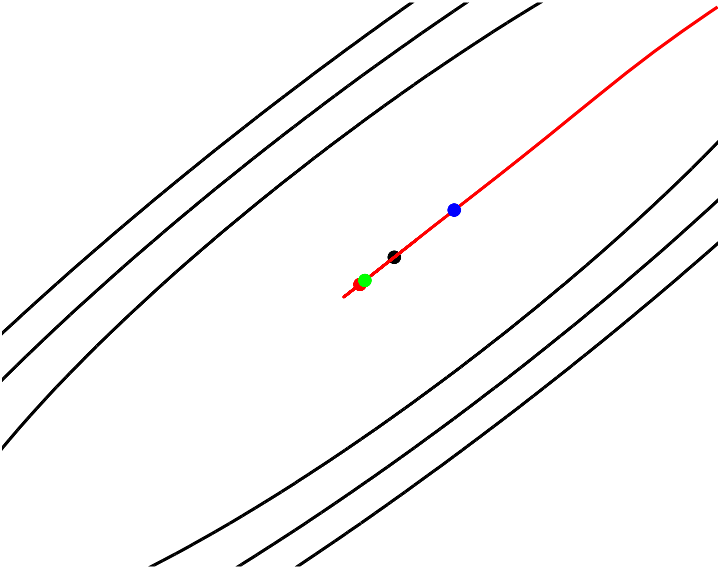
Траектория метода без рестарта ($k = 33$): обратно



Траектория метода без рестарта ($k = 34$): обратно



Траектория метода без рестарта ($k = 35$): обратно



Замечания

1. Корректность данной схемы доказана лишь в случае квадратичной функции. Корректность для неквадратичных функций является открытым вопросом.
2. На практике схема адаптивного рестарта почти всегда дает ускорение в сходимости и работает гораздо лучше схемы рестарта с фиксированным периодом.

План

1. Введение
   - Задачи и методы оптимизации
   - Основные теоретические сведения
2. Метод градиентного спуска
   - Стандартный метод для гладких функций
   - Метод для составных функций и автоматический подбор L
3. Быстрый градиентный метод
   - Метод для составных функций
   - Рестарт
   - Модификация подбора L и выбор L_0
4. Эксперименты и выводы
   - Эксперименты
   - Выводы

Схема автоматического подбора L: модификация

Текущая схема подбора L:

$L := L_k$;

повторять

  $\{$вычислить точки $\mathbf{y}$ и $\mathbf{T} = T_L(\mathbf{y})\}$;

  $s := L(\mathbf{y} - \mathbf{T}) - [\nabla f(\mathbf{y}) - \nabla f(\mathbf{T})]$;

  если $\langle s, \mathbf{y} - \mathbf{T} \rangle < \frac{1}{L}\|s\|^2$ **то** $L := \gamma_u L$;

пока неверно, что $\langle s, \mathbf{y} - \mathbf{T} \rangle \geqslant \frac{1}{L}\|s\|^2$;

$L_{k+1} := \max\left(L_0, \frac{L}{\gamma_d}\right)$;

Предлагается вместо условия выхода из цикла $\langle s, \mathbf{y} - \mathbf{T} \rangle \geqslant \frac{1}{L}\|s\|^2$ использовать следующее стандартное условие для L:

$$f(\mathbf{T}) \leqslant f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{T} - \mathbf{y} \rangle + \frac{L}{2}\|\mathbf{T} - \mathbf{y}\|^2.$$

(Как в обычном градиентом спуске.)

Модифицированная схема автоматического подбора L

$L := L_k$;
повторять
  {вычислить точки $\mathbf{y}$ и $\mathbf{T} = T_L(\mathbf{y})$};
  если $f(\mathbf{T}) > f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{T} - \mathbf{y} \rangle + \frac{L}{2}\|\mathbf{T} - \mathbf{y}\|^2$ **то**
    $L := \gamma_u L$;
пока неверно, что $f(\mathbf{T}) \leqslant f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{T} - \mathbf{y} \rangle + \frac{L}{2}\|\mathbf{T} - \mathbf{y}\|^2$;
$L_{k+1} := \max\left(L_0, \frac{L}{\gamma_d}\right)$;

На практике эта схема работает лучше, чем предыдущая.

Начальное приближение L_0

1. Как выбирать начальное приближение L_0?
2. Текущая схема требует только одно условие: $L_0 < L_f$.
3. Чем сильнее L_0 недооценивает L в области, в которой находится x_0, тем больше схема «разогревается» и «простраивает» вначале.

Пример

Пусть $L_0 = 1$, а L в области, в которой находится x_0, равно 100; $\gamma_u = 2$. Тогда процедура подбора L на первом шаге метода совершит 7 итераций внутреннего цикла (14 вызовов функции) прежде, чем будет получена точка x_1.

4. На практике имеет смысл выбрать L_0, которое будет переоценивать локальное L, чтобы метод сразу же начал «шагать». Чтобы это было возможно, потребуется небольшая модификация схемы.

Модификация, позволяющая выбрать большое L_0

$L := L_k$;
повторять
 {вычислить точки $\mathbf{y}$ и $\mathbf{T} = T_L(\mathbf{y})$};
 если $f(\mathbf{T}) > f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{T} - \mathbf{y} \rangle + \frac{L}{2} \|\mathbf{T} - \mathbf{y}\|^2$ **то**
  $L := \gamma_u L$;
пока неверно, что $f(\mathbf{T}) \leqslant f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{T} - \mathbf{y} \rangle + \frac{L}{2} \|\mathbf{T} - \mathbf{y}\|^2$;
$L_{k+1} := \frac{L}{\gamma_d}$;

На практике

1. $L_0 = L_f$, если глобальная константа Липшица L_f известна;
2. $L_0 = 500$, если L_f неизвестно.

Итоговый алгоритм быстрого градиентного метода

Вход: $\mathbf{x}_0 \in \mathbb{R}^n$, $L_0 > 0$, $\gamma_u > 1$, $\gamma_d \geqslant 1$;

$A_0 := 0$;

для $k = 0, 1, 2, \ldots$

  $L := L_k$;

  повторять

    {найти a из уравнения $\frac{a^2}{2(A_k+a)} = \frac{1}{L}$};

    {вычислить $\mathbf{v}_k = \arg\min_{\mathbf{x}\in\mathbb{R}^n} \psi_k(\mathbf{x})$};

    $\mathbf{y} := \frac{A_k \mathbf{x}_k + a\mathbf{v}_k}{A_k + a}$; {вычислить $\nabla f(\mathbf{y})$};

    $\mathbf{T} := T_L(\mathbf{y})$; {вычислить $\nabla f(\mathbf{T})$};

    если $f(\mathbf{T}) > f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{T} - \mathbf{y} \rangle + \frac{L}{2}\|\mathbf{T} - \mathbf{y}\|^2$ **то**

      $L := \gamma_u L$;

  пока неверно, что $f(\mathbf{T}) \leqslant f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{T} - \mathbf{y} \rangle + \frac{L}{2}\|\mathbf{T} - \mathbf{y}\|^2$;

  $\mathbf{x}_{k+1} := \mathbf{T}$; $A_{k+1} := A_k + a$; $L_{k+1} := \frac{L}{\gamma_d}$;

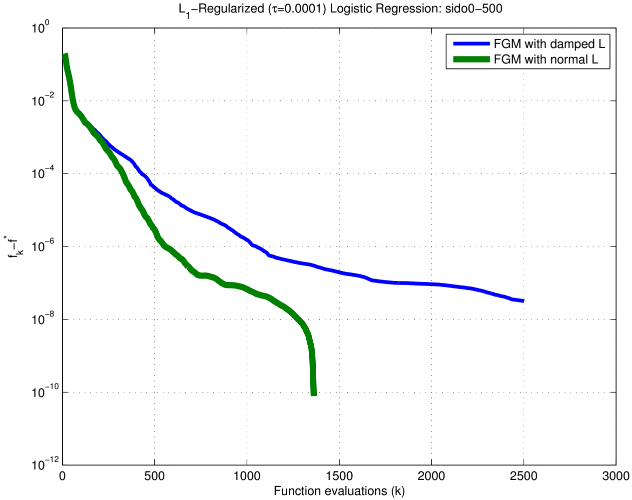
  если $\langle \mathbf{y} - \mathbf{x}_{k+1}, \mathbf{x}_{k+1} - \mathbf{x}_k \rangle > 0$ **то** $\mathbf{x}_{k+1} := \mathbf{x}_k$; $A_{k+1} := 0$;

Рекомендуемые параметры: $L_0 = 500$ (или L_f), $\gamma_u = 2$, $\gamma_d = 1.1$.

План

1. Введение
   - Задачи и методы оптимизации
   - Основные теоретические сведения
2. Метод градиентного спуска
   - Стандартный метод для гладких функций
   - Метод для составных функций и автоматический подбор L
3. Быстрый градиентный метод
   - Метод для составных функций
   - Рестарт
   - Модификация подбора L и выбор L_0
4. Эксперименты и выводы
   - Эксперименты
   - Выводы

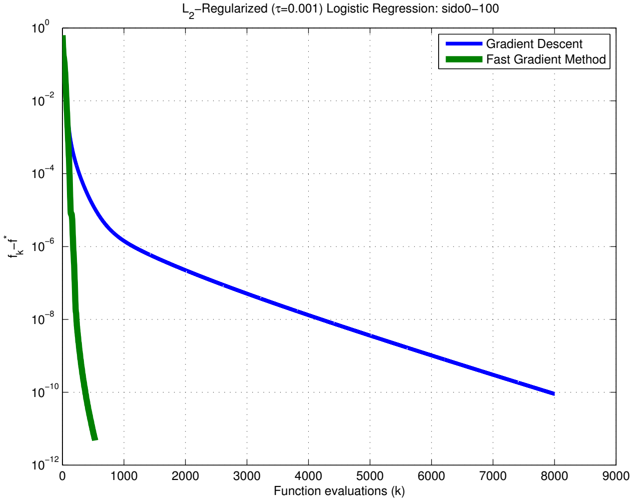
Схемы автоматического подбора L



Логистическая регрессия с L_1-регуляризатором,

$\tau = 0.0001$, 500 объектов, 4 933 признаков.

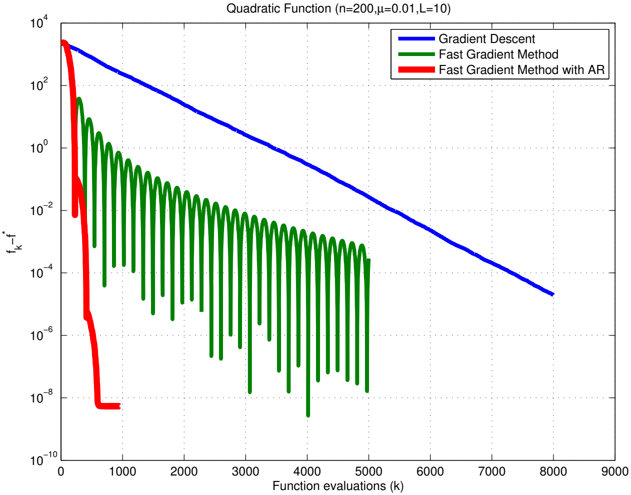
Градиентный спуск и быстрый градиентный метод



Логистическая регрессия с L_2-регуляризатором,

$\tau = 0.001$, 100 объектов, 4 933 признаков.

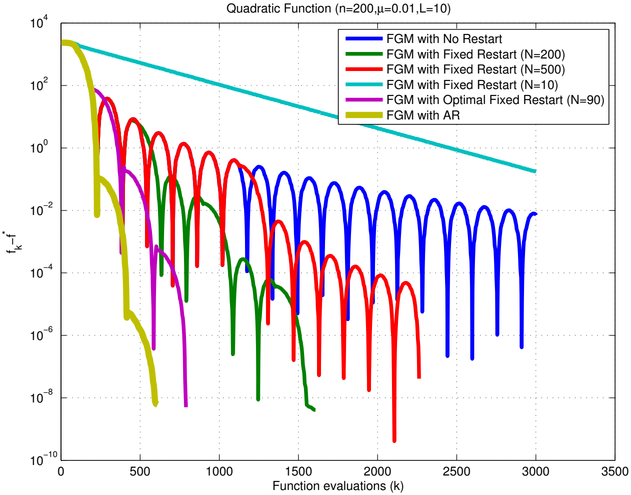
Градиентный спуск и быстрый градиентный метод – 2



Квадратичная функция,

$n = 200$, $\mu = 0.01$, $L = 10$.

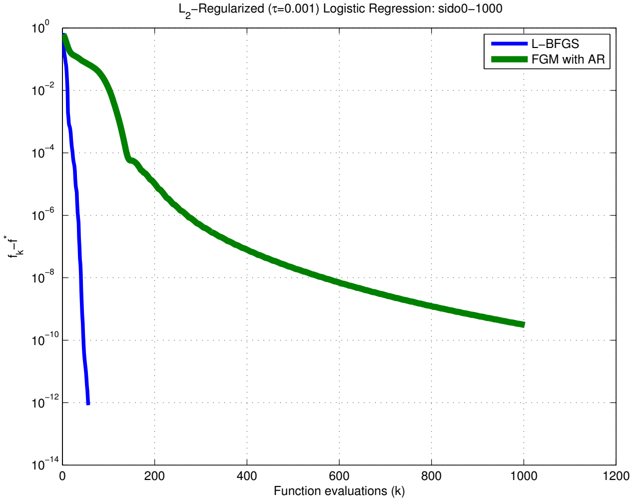
Стратегии рестарта



Квадратичная функция,

$n = 200, \; \mu = 0.01, \; L = 10.$

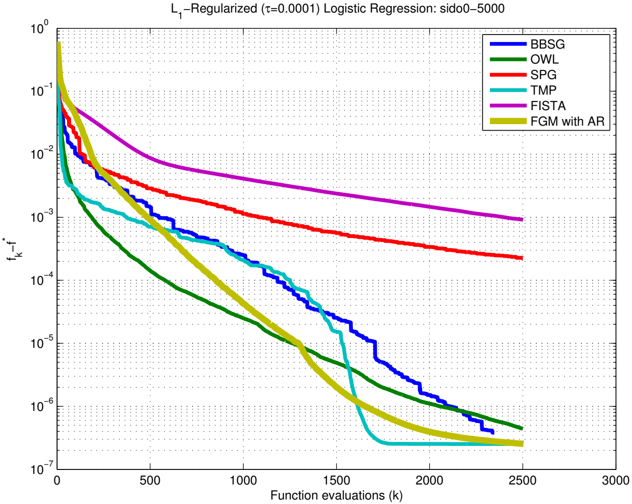
Быстрый градиентный метод и L-BFGS



Логистическая регрессия с L_2-регуляризатором,

$\tau = 0.001$, 1 000 объектов, 4 933 признаков.

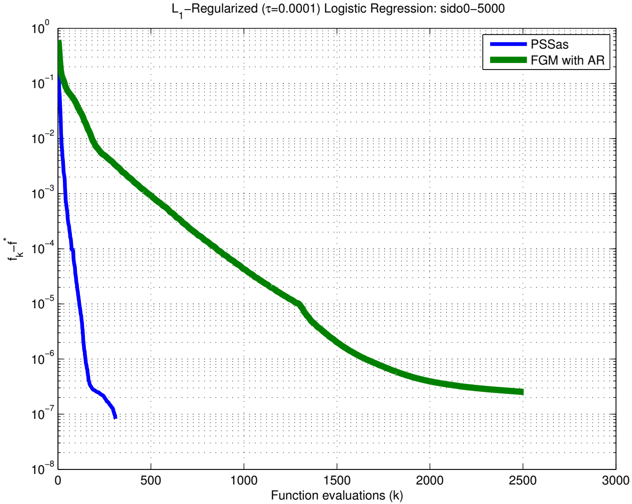
Быстрый градиентный метод и другие методы



Логистическая регрессия с L_1-регуляризатором,

$\tau = 0.0001$, 5 000 объектов, 4 933 признаков.

Быстрый градиентный метод и PSSas



Логистическая регрессия с L_1-регуляризатором,

$\tau = 0.0001$, 5 000 объектов, 4 933 признаков.

План

1. Введение
   - Задачи и методы оптимизации
   - Основные теоретические сведения
2. Метод градиентного спуска
   - Стандартный метод для гладких функций
   - Метод для составных функций и автоматический подбор L
3. Быстрый градиентный метод
   - Метод для составных функций
   - Рестарт
   - Модификация подбора L и выбор L_0
4. Эксперименты и выводы
   - Эксперименты
   - Выводы

Заключение

1. Быстрый градиентный метод работает на порядок быстрее обычного градиентного метода.
2. Как и обычный градиентный спуск, быстрый градиентный метод легко обобщается на случай составных функций, что позволяет решать гладкие задачи с простой негладкой добавкой.
3. Метод можно обобщить на случай выпуклых задач с гладкими ограничениями. Однако итоговая схема получится существенно сложнее и уже будет иметь некоторую другую скорость сходимости.
4. Несмотря на то, что методы типа L-BFGS не имеют хороших гарантий сходимости, в отличие от быстрого градиентного метода, на практике они работают существенно быстрее последнего.

Открытые вопросы

1. Ускорение текущей версии метода за счет
   - модификации градиентного шага: более хорошая схема подбора L, шаг типа L-BFGS и т. п.;
   - модификации оценочных функций: например, оценка с помощью пучка (англ. bundle).
2. Обобщение метода для невыпуклых функций.
3. Обобщение метода на стохастический случай.