Теория к зачёту по методам оптимизации

Иванов Вячеслав, группа 699

21 декабря 2018 г.

Оглавление

1 Безусловные методы первого порядка

1.1 Градиентный спуск

Основная формула[1]:

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

Теорема 1.1 (*Выпуклая функция, оценка сверху*).
Пусть целевая функция f выпуклая с липшицевым градиентом, а $\alpha = \frac{1}{L}$. Тогда:

$$f(x_{k+1}) - f^* \leqslant \frac{2L\|x - x_0\|_2^2}{k+4} = \mathcal{O}\left(\frac{1}{k}\right)$$

Т.е. сходимость сублинейная в смысле$\|x_{k+1} - x^*\|_2 \leqslant Ck^{\alpha},\ \alpha < 0$.
Т.е. для достижения точности ε нужно $\mathcal{O}\left(\frac{1}{\varepsilon}\right)$ итераций.

Теорема 1.2 (*Сильно выпуклая функция, оценка сверху*).
Пусть f сильно выпукла с липшицевым градиентом, а $\alpha = \frac{2}{L+\mu}$. Тогда:

$$f(x_k) - f^* \leqslant \frac{L}{2}\left(\frac{\kappa - 1}{\kappa + 1}\right)^{2k} \|x_0 - x^*\|_2^2,\ \kappa = \frac{L}{\mu}$$

Т.е. сходимость линейная в смысле определения $\|x_{k+1} - x^*\|_2 \leqslant Cq^k$ с $q = \frac{\kappa-1}{\kappa+1}$.
Т.е. для достижения точности ε нужно $\mathcal{O}\left(\kappa \ln \frac{1}{\varepsilon}\right)$ итераций.

Теорема 1.3 (*Выпуклая функция, оценка снизу*).

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

т.е. показатель сублинейной сходимости $\alpha \in (-1,\ -0.5)$.
Более того, оценка верна для всех методов первого порядка.

Теорема 1.4 (*Сильно выпуклая функция, оценка снизу*).

$$f(x_{k+1}) - f^* \geqslant \frac{\mu}{2}\left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}\right)^{2k} \|x_0 - x^*\|_2^2$$

т.е. знаменатель прогрессии в линейной сходимости $q \in \left(\frac{\kappa-1}{\kappa+1}; \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)$.

1.2 Метод тяжёлого шарика

Основная формула[2]:

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

Теорема 1.5 (*Сильно выпуклая функция, оценка сверху*).
Пусть f — сильно выпуклая функция с липшицевым градиентом. Тогда:

$$\alpha_k = \frac{4}{(\sqrt{L} + \sqrt{\mu})^2},\ \beta_k = \max\{|1 - \sqrt{\alpha_k L}|,\ |1 - \sqrt{\alpha_k \mu}|\}^2 \implies$$

$$\implies \left\|\begin{bmatrix} x_{k+1} - x^* \\ x_k - x^* \end{bmatrix}\right\|_2 \leqslant \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}\right)^k \left\|\begin{bmatrix} x_1 - x^* \\ x_0 - x^* \end{bmatrix}\right\|_2$$

Т.е. сходимость линейная в смысле определения $\|x_{k+1} - x^*\|_2 \leqslant Cq^k$ с $q = \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}$.
Т.е. для достижения точности ε нужно $\mathcal{O}\left(\sqrt{\kappa} \ln \frac{1}{\varepsilon}\right)$ итераций.

[1] Дискретизация ОДУ $\dot{x}\,(t) = -\nabla f(x)$
[2] Дискретизация ОДУ второго порядка — затухающих колебаний в вязкой среде

1.3 Метод Нестерова

Основные формулы:

$$x_{k+1} = y_k - \alpha_k \nabla f(y_k)$$
$$y_{k+1} = x_{k+1} + \frac{k}{k+3}(x_{k+1} - x_k)$$

Теорема 1.6 (*Оценка сверху*).

$$f(x_{k+1}) - f^* \leqslant \frac{2L\|x_0 - x^*\|_2^2}{(k+2)^2}$$

Т.е. сходимость сублинейная в смысле$\|x_{k+1} - x^*\|_2 \leqslant Ck^{\alpha},\ \alpha < 0.$
Т.е. для достижения точности ε нужно $\mathcal{O}\left(\frac{1}{\sqrt{\varepsilon}}\right)$ итераций.

1.4 Метод сопряжённых градиентов

$$\min \frac{1}{2}x^\top Ax - b^\top x,\ A \in S^n_{++}$$

Идея: каждым шагом устранять отличие x_k от x^* в одной координате (как минимум).
Факт: для этого не обязательно знать x^*. По сути, обобщается идея предобуславливания.

Определение (*A-ортогональность*).

$$x, y:\ x^\top Ay = 0,\ A \in S^n_{++}$$

Эквивалентно ортогональности относительно скалярного произведения с матрицей Грама A.

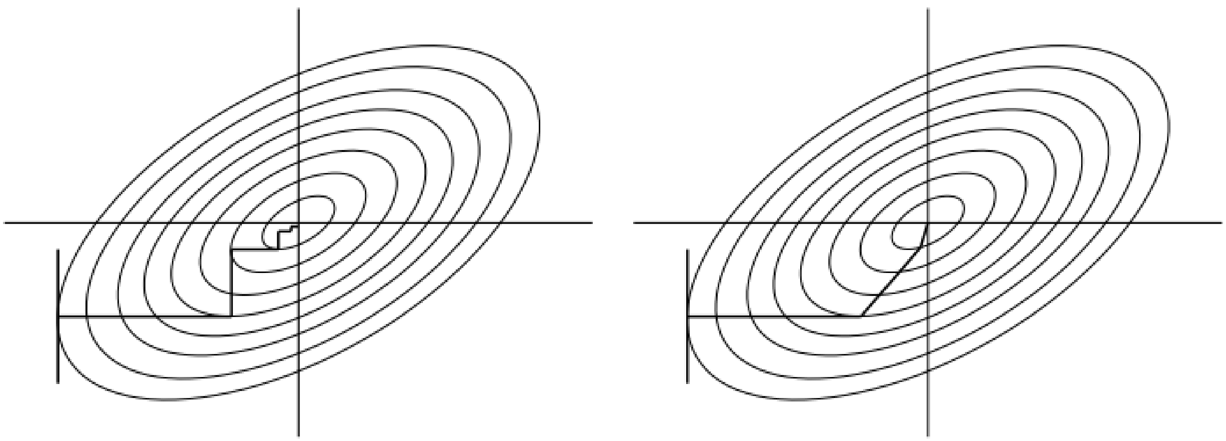

Figure 6.14: Steepest descent vs. conjugate gradient.

В алгоритме выстраивается последовательность A-ортогональных векторов p_j, через которые пересчитываются направления x_k. Формулы пересчёта имеют вид:

$$p_0 = -r_0 = Ax - b$$
$$x_{k+1} = x_k + \alpha_k p_k$$
$$r_{k+1} = r_k + \alpha_k Ap_k$$
$$p_{k+1} = -r_{k+1} + \beta_{k+1}p_k$$
$$\alpha_{k+1} = -\frac{r_k^\top p_k}{p_k^\top Ap_k},\ \beta_{k+1} = \frac{p_k^\top Ar_{k+1}}{p_k^\top Ap_k}$$

Где формула для β_{k+1} выводится из условия A-ортогональности векторов p_{k+1} и p_k, а для α_{k+1} — из условия $r_k \perp p_i,\ i = 1, \dots, k-1$. Тогда $x_k = \underset{x\ \in P}{\arg\min} f(x),\ P = x_0 + \operatorname{span}(p_0, \dots, p_{k-1})$

Формулы пересчёта коэффициентов упрощаются:

$$
\begin{aligned}
\alpha_k &= -\frac{r_k^\top p_k}{p_k^\top A p_k} = \\
&= -\frac{r_k^\top(-r_k + \beta_k p_{k-1})}{p_k^\top A p_k} = \\
&= \frac{\|r_k\|_2^2}{p_k^\top A p_k} + \frac{r_k^\top p_{k-1}}{p_k^\top A p_k} = [r_k \perp p_{k-1}] = \\
&= \frac{\|r_k\|_2^2}{p_k^\top A p_k} \\
\beta_{k+1} &= \frac{r_{k+1}^\top A p_k}{p_k^\top A p_k} = [\alpha_k A p_k = r_{k+1} - r_k] = \\
&= \frac{r_{k+1}^\top (r_{k+1} - r_k)}{(-r_k + \beta_k p_{k-1})^\top (r_{k+1} - r_k)} = \\
&= \frac{\|r_{k+1}\|_2^2}{\|r_k\|_2^2}
\end{aligned}
$$

Теорема 1.7.

- $r_k \perp r_i,\ i < k$
- $\operatorname{span}(r_0, \dots, r_k) = \operatorname{span}(p_0, \dots, p_k) = \operatorname{span}(r_0, Ar_0, \dots, A^k r_0)$
- $p_k^\top A p_k = 0,\ i < k$
- Метод сопряжённых градиентов оптимален для квадратичной целевой функции. Он сходится за $|\operatorname{spec}(A)| \leqslant n$ итераций, где $\operatorname{spec}(A)$ — спектр оператора A.
- Справедлива оценка:

$$f_k - f^* \leqslant C \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^k$$

1.5 Метод Флетчера-Ривза

Обобщение метода сопряжённый градиентов на неквадратичную целевую функцию.

- α_k подбирается адаптивно (замкнутой формы нет)
- β_k ищется с помощью градиентов $\nabla f'(x_{k-1}), \nabla f'(x_{k-2})$
- r_k в формулах заменяется на $\nabla f'(x_k)$ и пересчитывается в лоб

При этом:

- С ростом числа итераций направления p_k становятся всё более коллинеарными.
- Не при любом способе выбора α_k получается направление убывания.

Первая проблема решается рестартами, вторая разобрана не была.

2 Проксимальные методы

2.1 PGM и его частные случаи

Определение (*Проксимальный оператор*).

$$\operatorname*{prox}_{\alpha f}(x) = \arg\min_u \left(f(u) + \frac{1}{2\alpha} \|x - u\|_2^2 \right)$$

Если f выпукла, то проксимальный оператор — сильно выпуклая функция.

Теорема 2.1. Минимум функции будет неподвижной точкой проксимального оператора.

Утверждение 2.1. Для проксимального оператора выполнено некоторое подобие линейности:

$$f(x) = \sum_{i=1}^{n} f_i(x_i) \implies \underset{f}{prox}(v)_i = \underset{f_i}{prox}(v_i)$$

Определение (*Проксимальный градиентный метод*).
Пусть выпуклая функция f представима в виде $f = g + h$, где g выпукла и может принимать бесконечные значения, а h дифференцируемая и выпуклая. Тогда положим:

$$\begin{aligned}
x_{k+1} &= \operatorname*{prox}_{\alpha_k g}(x_k - \alpha_k \nabla h(x_k)) = \\
&= \arg\min_x \left(g(x) + \frac{1}{2\alpha_k} \|x - (x_k - \alpha_k \nabla h(x_k))\|_2^2 \right) = \\
&= \arg\min_x \left(g(x) + h(x_k) + \langle \nabla h(x_k), x - x_k \rangle + \frac{1}{2\alpha_k} \|x - x_k\|_2^2 \right)
\end{aligned}$$

Такой метод сходится как $\mathcal{O}\left(\frac{1}{k}\right)$ для $\alpha_k \equiv \text{const} \in \left(0, \frac{1}{L}\right]$, где L — константа Липшица для ∇h.
Видно, что:

1. Если $g(x) := \begin{cases} x, & x \in G \\ +\infty, & \text{otherwise} \end{cases}$ — индикатор выпуклого множества G, — то:

$$\begin{aligned}
&\arg\min_x \left(g(x) + h(x_k) + \langle \nabla h(x_k), x - x_k \rangle + \frac{1}{2\alpha_k} \|x - x_k\|_2^2 \right) = \\
&= \arg\min_{x \in G} \left(h(x_k) + \langle \nabla h(x_k), x - x_k \rangle + \frac{1}{2\alpha_k} \|x - x_k\|_2^2 \right)
\end{aligned}$$

   — т.н. *метод проекции градиента.*

2. Если $h(x) \equiv 0$, то получаем *проксимальный метод.*

3. Если $g(x) \equiv 0$, то получаем градиентный спуск (при $\alpha_k = \frac{1}{L}$).

Сходимость каждого метода аналогична оной у градиентного спуска (с поправкой на то, что в методе проекции градиента нужно учесть сложность вычисления проекции точки на множество).

Теорема 2.2 (*Метод проекции градиента, оценка сверху*).
Если f выпуклая с липшицевым градиентом на замкнутом выпуклом мн-ве P, то при $\alpha \in \left(0, \frac{2}{L}\right]$:

- ◇ $x_k \to x^*$ сублинейно, а в случае сильной выпуклости — линейно со знаменателем q
- ◇ Если при этом $\exists l > 0 \forall x \in P : \nabla^2 f(x) \succeq l\mathbf{I}$, то $q = \max\{|1 - \alpha l|, |1 - \alpha L|\}$

Определение (*FISTA*[3])**.** Аналог метода Нестерова для проксимального градиентного метода:

$$\begin{aligned} y_1 &:= x_0,\ t_1 := 1,\ k = 1 \\ x_k &:= \underset{\alpha_k g}{\operatorname{prox}}(y_k - \alpha_k \nabla h(y_k)) \\ t_{k+1} &:= \frac{1 + \sqrt{1 + 4t_k^2}}{2} \\ y_{k+1} &:= x_k + \frac{t_k - 1}{t_{k+1}}(x_k - x_{k-1}) \end{aligned}$$

Теорема 2.3 (*О сходимости FISTA*)**.**

	шаг	скорость сходимости	число итераций
h выпуклая с липшицевым градиентом	$\alpha_k \equiv \frac{1}{k}$	$\mathcal{O}\left(\frac{1}{k^2}\right)$	$\mathcal{O}\left(\frac{1}{\sqrt{\varepsilon}}\right)$
h сильно выпуклая с липшицевым градиентом	$\alpha_k \equiv \frac{1}{k}$	$\mathcal{O}\left(\left(1 - \frac{1}{\sqrt{\kappa}}\right)^k\right)$	$\mathcal{O}(\sqrt{\kappa} \log \frac{1}{\varepsilon})$

Pro:

- Часто проекцию можно вычислить аналитически
- При этом сходимость аналогична градиентому спуску в безусловной оптимизации
- Обобщается на негладкий случай

Contra:

- Проекция не всегда вычисляется за $\mathcal{O}(1)$ (пример — проекция на политоп)
- При обновлении градиента может меняться структура задачи (из-за свойств множества)

3 Условные методы первого порядка

3.1 Метод Франка-Вольфе (условного градиента)

Идея: в методе градиентного спуска подбирать направление и длину шага так, чтобы не выходить за пределы множества. Это возможно, если ограничения задают выпуклое замкнутое множество.

$$f(x_k + s_k) \approx f(x_k) + \langle \nabla f(x_k), s_k \rangle \to \min_{s_k \in P}$$

Направление $s_k - x_k$ называется *условным градиентом* функции f в точке x_k на мн-ве P. В силу выпуклости, $f(s) \geqslant f(x_k) + \langle \nabla f(x_k), s - x_k \rangle$, откуда получаем аналог зазора двойственности:

$$f(x) - f(x^*) \leqslant -\min_{s \in P} \langle \nabla f(x_k), x - s \rangle = \max_{s \in P} \langle \nabla f(x_k), x - s \rangle = g(x)$$

Теорема 3.1 (*Метод условного градиента, оценка сверху*)**.**
Пусть f выпуклая с липшицевым градиентом на выпуклом компакте X. Тогда при $\alpha_k = \frac{2}{k+1}$:

$$f(x_k) - f(x^*) \leqslant \frac{2d^2 L}{k+2},\ k \geqslant 1,\ d = \operatorname{diam}(X)$$

Полученная сублинейная скорость сходимости не зависит от размерности. Тем не менее, она неулучшаема даже для сильно выпуклых функций, а метод не обобщается на негладкий случай.

[3] Fast Iterative Shrinkage-Thresholding Algorithm

4 Методы второго порядка

4.1 Метод Ньютона

Основная формула:

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

Выводится из квадратичной аппроксимации при $\nabla^2 f(x) \succ 0$:

$$\begin{aligned} f(x+h) \approx\ & f(x) + \nabla f(x)^\top h + \frac{1}{2} h^\top \nabla^2 f(x) h \ \Big| \ \frac{\partial}{\partial h} \\ 0 &= \nabla f(x) + \nabla^2 f(x) h \\ h &= -(\nabla^2 f(x))^{-1} \nabla f(x) \end{aligned}$$

Аналогичный метод применяется для решения систем нелинейных уравнений:

$$\begin{gathered} G(x) = 0, \quad G : \mathbb{R}^n \to \mathbb{R}^n \\ G(x_k + \Delta x) \approx G(x_k) + \nabla G(x_k) \Delta x = 0 \\ \Delta x = -(\nabla G(x_k))^{-1} G(x_k) \\ x_{k+1} = x_k - (\nabla G(x_k))^{-1} G(x_k) \end{gathered}$$

Можно заметить, что необходимость решать такие системы возникает из условий оптимальности:

$$f'(x^*) = G(x) = 0$$

Определение (*Демпфированный метод Ньютона*)**.**

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

Адаптивный выбор размера шага расширяет область сходимости.

Теорема 4.1 (*Характер сходимости в методе У а*)**.**
Пусть

1. $f(x)$ локально сильно выпукла с константой μ: $\exists x^* : \nabla^2 f(x^*) \succeq \mu I$
2. Её гессиан липшицев с константой M: $\|\nabla^2 f(x) - \nabla^2 f(y)\| \leqslant M\|x - y\|$
3. Начальная точка достаточно близка к оптимальной: $\|x_0 - x^*\| \leqslant \frac{2\mu}{3M}$

Тогда метод Ньютона сходится квадратично:

$$\|x_{k+1} - x^*\| \leqslant \frac{M\|x_k - x^*\|^2}{2(\mu - M\|x_k - x^*\|)}$$

Т.е. для достижения точности ε нужно $\mathcal{O}\left(\log\log \frac{1}{\varepsilon}\right)$ итераций.

4.2 Квазиньютоновские методы

Мотивированы тем, что в методе Ньютона нужно явно хранить гессиан ($\mathcal{O}(n^2)$ памяти) и обращать его ($\mathcal{O}(n^3)$ операций) на каждой итерации. Кроме того, на очередной итерации он может оказаться вырожденным. Этих недостатков можно избавиться, если заменить гессиан на его приближение и придумать разумные формулы пересчёта. Посмотрим на две крайности:

- **Градиентный спуск:**

$$\begin{aligned} f(x+h) &\leqslant f(x) + \langle \nabla f(x), h \rangle + \frac{1}{2\alpha} h^\top I h, \ \alpha \in (0, L^{-1}] \\ \min_h f_g(h) &\implies h^* = -\alpha \nabla f(x) \\ x_{k+1} &= x_k - \alpha_k \nabla f(x_k) \end{aligned}$$

- **Метод Ньютона:**

$$\begin{aligned} f(x+h) &\approx f(x) + \langle \nabla f(x), h \rangle + \frac{1}{2\alpha} h^\top \nabla^2 f(x) h, \ \nabla^2 f(x) \succ 0 \\ \min_h f_N(h) &\implies h^* = -(\nabla^2 f(x))^{-1} \nabla f(x) \\ x_{k+1} &= x_k - (\nabla^2 f(x_k))^{-1} \nabla f(x_k) \end{aligned}$$

Все квазиньютоновские методы находятся "между ними":

$$\begin{aligned} f_q(h) := f(x+h) &\approx f(x) + \langle \nabla f(x), h \rangle + \frac{1}{2\alpha} h^\top B_k h, \ B_k \succ 0 \\ \min_h f_q(h) &\implies h^* = -B_k^{-1} \nabla f(x) \\ x_{k+1} &= x_k - B_k^{-1} \nabla f(x_k) = \\ &= x_k - H_k \nabla f(x_k) \end{aligned}$$

К B_k, кроме близости к гессиану по матричной норме, предъявляются следующие требования:

- Быстрый пересчёт $B_k \to B_{k+1}$, когда доступны только градиенты.
- Быстрый поиск направления h_k.
- Возможность компактного хранения B_k.
- Сверхлинейная сходимость.

Для обновления B_k есть следующие 2 правила:

1. **Правило двух градиентов:**

$$\begin{aligned} f_q'(-\alpha_k h_k) = f'(x_k) &\implies \nabla f(x_{k+1}) - \alpha_k B_{k+1} h_k = \nabla f(x_k) \\ f_q'(0) = f'(x_{k+1}) & \end{aligned}$$

2. **Secant equation:**

$$\begin{aligned} s_k &:= x_{k+1} - x_k \\ y_k &:= \nabla f(x_{k+1}) - \nabla f(x_k) \\ B_{k+1} s_k &= y_k \end{aligned}$$

4.3 Barzilai-Borwein

Аппроксимация гессиана диагональной матрицей:

$$\alpha_k \nabla f(x_k) = \alpha_k I \nabla f(x_k) = \left(\frac{1}{\alpha_k} I \right)^{-1} \nabla f(x_k) \approx (\nabla^2 f(x))^{-1} \nabla f(x)$$

Квазиньютоновское уравнение принимает вид:

$$\alpha_k^{-1} s_{k-1} \approx y_{k-1}$$

Отсюда естественным образом возникает задача наименьших квадратов:

$$\min_{\alpha_k} \frac{1}{2} \|s_{k-1} - \alpha_k y_{k-1}\|_2^2 \implies \alpha_k = \frac{s_{k-1}^\top y_{k-1}}{y_{k-1}^\top y_{k-1}}$$

Хотя её можно поставить иначе (скажем, в другой норме) и получить другие методы.

4.4 DFP

Поиск B_{k+1} сам по себе записывается в форме задачи оптимизации:

$$\begin{aligned} B_{k+1} =& \arg\min_B \|B_k - B\|_2^2 \\ & B^\top = B \\ \text{s.t. } & Bs_k = y_k \\ & B \succeq 0 \end{aligned}$$

Замкнутый вид решения, откуда H_{k+1} получается по формуле ШМВ:

$$B_{k+1} = (I - \rho_k y_k s_k^\top) B_k (I - \rho_k s_k y_k^\top) + \rho_k y_k y_k^\top, \ \rho_k = \frac{1}{y_k^\top s_k}$$

$$B_{k+1}^{-1} = H_{k+1} = H_k - \frac{H_k y_k y_k^\top H_k}{y_k^\top H_k y_k} + \frac{s_k s_k^\top}{y_k^\top s_k}$$

4.5 (L-)BFGS

Аналогичная задача оптимизации ставится сразу для H_{k+1}:

$$\begin{aligned} H_{k+1} =& \arg\min_H \|H_k - H\|_2^2 \\ & H^\top = H \\ \text{s.t. } & Hy_k = s_k \\ & H \succeq 0 \end{aligned}$$

Замкнутый вид решения, откуда H_{k+1} получается по формуле ШМВ:

$$H_{k+1} = (I - \rho_k s_k y_k^\top) H_k (I - \rho_k y_k s_k^\top) + \rho_k s_k s_k^\top, \ \rho_k = \frac{1}{y_k^\top s_k}$$

Теорема 4.2. Если f сильно выпуклая с липшицевым гессианом, то при некоторых дополнительных технических допущениях BFGS сходится сверхлинейно.

Если размерность велика, хранить эти матрицы целиком становится непрактично.
Т.е. нужна процедура эффективного умножения на вектор $\nabla f(x)$.
Также есть проблема, что значения s_k, y_k на первых итерациях могут портить оценку B_k, H_k на более поздних итерациях. Модификация L-BFGS позволяет рекурсивно вычислять $H_{k+1}\nabla f(x_k)$ за счёт хранения только последних $m \ll n$ значений (s_k, y_k) и обновления $H_{m,0}$.

- BFGS обновляет H рекурсивно

$$H_{k+1} = V_k^\top H_k V_k + \rho_k s_k s_k^\top, \quad V_k = I - \rho_k y_k s_k^\top$$

- Развернём m шагов рекурсии

$$\begin{aligned}
H_{k+1} &= V_k^\top H_k V_k + \rho_k s_k s_k^\top \\
&= V_k^\top V_{k-1}^\top H_{k-1} V_{k-1} V_k + \rho_{k-1} V_k^\top V_{k-1}^\top s_{k-1} s_{k-1}^\top V_{k-1} V_k + \rho_k s_k s_k^\top \\
&= V_k^\top \dots V_{k-m+1}^\top H_{m,0} V_{k-m+1} \dots V_k \\
&+ \rho_{k-m+1} V_k^\top \dots V_{k-m+2}^\top s_{k-m+1} s_{k-m+1}^\top V_{k-m+2} \dots V_k \\
&+ \dots + \rho_k s_k s_k^\top
\end{aligned}$$

Отсюда получаем $\mathcal{O}(n^2)$ операций на одну итерацию и линейную сложность по памяти. Тем не менее, у метода есть недостатки:

- Он не рандомизируется.
- Нужно подбирать B_0 или H_0.
- Для него нет разработанной теории сходимости и точных оценок.
- Не любой способ выбора шага гарантирует, что $y_k^\top s_k > 0$.

5 Линейное программирование (LP)

$$\begin{aligned}
&\min_x c^\top x \\
&\text{s.t. } Ax \preceq 0, \\
&\quad\; x \succeq 0
\end{aligned}$$

Определение (*Угловая точка*)**.**

1. Точка из допустимого множества называется *вершиной многоугольника*, если она не лежит на отрезке между двумя другими точками многоугольника.
2. Точка x называется *угловой точкой* многоугольника, если:
   - Она лежит в множестве.
   - Существует такое множество $\mathcal{B} \subset \{1, \dots, n\}$, что
     - $|\mathcal{B}| = m$
     - $i \notin \mathcal{B} \Rightarrow x_i = 0$
     - матрица $B = [a_i]_{i \in \mathcal{B}}$ невырождена, где a_i - i-ый столбец матрицы A.

   Матрица B называется *матрицей базиса*.

Теорема 5.1. Все угловые точки являются вершинами многоугольника $Ax \preceq b$.

Теорема 5.2 (*Основная теорема линейного программирования*).

- ◇ Если в задаче линейного програмирования допустимое множество непусто, тогда оно имеет как минимум одну угловую точку.
- ◇ Если задача линейного программирования имеет решения, тогда хотя бы одно из них является угловой точкой.
- ◇ Если задача линейного программирования ограничена и допустимое множество непусто, тогда она имеет оптимальное решение.

Основной алгоритм решения задачи линейного программирования называется *симплекс-методом*:

1. Начальная точка — произвольная вершина.
2. Перебрать смежные вершины и пойти в ту, в которой значение целевой функции меньше.

Шаги алгоритма:
Дана угловая точка x, соответствующая ей матрица базиса B и множество индексов $\mathcal{B}$.

1. Вычислить *оценки замещения* (reduced costs) $\overline{c}_j = c_j - c_{\mathcal{B}}^\top B^{-1} a_j$ для всех $j \notin \mathcal{B}$.
   (a) если $\overline{c}_j \geq 0$ для всех j, то текущее значение является оптимальным и уменьшить целевую функцию нельзя
   (b) иначе выбрать индекс j^*, для которого $\overline{c}_{j^*} < 0$
2. Вычислить $u = B^{-1} a_{j^*}$
   (a) если все компоненты u неположительны, то задача неограничена, оптимальное значение равно $-\infty$
   (b) если есть положительные компоненты, то
   $$\theta^* := \min_{\{i | u_i > 0\}} \frac{x_{\mathcal{B}(i)}}{u_i}$$
   По смыслу — выбрать столбец в симплекс-таблице на основе оценок замещения.
3. Пусть ℓ такой индекс, что
   $$\theta^* = \frac{x_{\mathcal{B}(\ell)}}{u_\ell}.$$
   Сформировать новую матрицу базиса $\hat{B}$ с помощью замены столбца $a_{\mathcal{B}(\ell)}$ на столбец a_{j^*}. Новая угловая точка $\hat{x}$, соответствующая матрице базиса $\hat{B}$, определяется так:
   $$\begin{aligned} &\hat{x}_{j^*} = \theta^* \\ &\hat{x}_{\mathcal{B}(k)} = x_{\mathcal{B}(k)} - \theta^* u_k, \text{если } k \neq \ell \end{aligned}$$
   По смыслу — выбрать строку в симплекс-таблице и провести итерацию метода Гаусса с ведущим элементом на пересечении выбранных строки и столбца.

Чтобы избежать зацикливания, которое может произойти, если угловая точка содержит больше $(m - n)$ нулевых координат, на каждом шаге нужно выбирать лекс. минимальные индексы.

- ◇ Было показано, что в худшем случае время работы симплекс-метода **экспоненциально** зависит от размерности задачи!
- ◇ Однако на практике сложность чаще всего пропорциональна количеству ограничений и симплекс-метод сходится быстро.

- ◇ Почему это так, неясно до сих пор.

Занимательный факт: в невырожденных задачах линейного программирования всегда выполнена сильная двойственность. При этом в двойственной задаче число переменных совпадает с числом ограничений в прямой задаче, что может сильно понизить размерность. Но неясно, как по двойственным переменным восстановить ответ прямой задачи, в нашем курсе это не обсуждалось.

5.1 Двухфазный симплекс-метод

Проблема: выбор начальной точки неочевиден.
Решение: вспомогательная. задача оптимизации (при условии $b \succeq 0$)[4]:

$$
\begin{aligned}
&\min_{z,y} y_1 + \ldots + y_m \\
\text{s.t. } & Az + y = b \\
& z \geq 0,\ y \geq 0
\end{aligned}
$$

Начальная точка для такой задачи очевидна — $z = 0,\ y = b$.

- ◇ Если оптимальное значение функции в этой задаче **не равно** 0, то допустимое множество исходной задачи пусто.
- ◇ Если оптимальное значение функции в этой задаче **равно** 0, то $y^* = 0$ и $x_0 = z^*$.

5.2 М-метод

Идея: объединить двухфазный симплекс-метод в однофазный

$$
\begin{aligned}
&\min_{z,y} c^\top z + M(y_1 + \ldots + y_m) \\
\text{s.t. } & Az + y = b \\
& z \geq 0,\ y \geq 0
\end{aligned}
$$

M - произвольное большое положительное число.[5]

6 Полуопределённое программирование (SDP)

Постановка задачи полуопределённого (semidefinite) программирования:

$$
\begin{aligned}
&\min\ c^\top x \\
&\text{s.t. } F_0 + \sum_{i=1}^m x_i F_i \succeq 0 \\
&F_i \in \mathbb{R}^{n \times n}
\end{aligned}
$$

К такой постановке сводятся задачи LP:

$$
\begin{aligned}
&\min\ c^\top x \\
&\text{s.t. } Ax + b \succeq 0
\end{aligned}
\quad \longrightarrow \quad
\begin{aligned}
&\min\ c^\top x \\
&\text{s.t. } \mathrm{diag}(b) + \sum_{i=1}^n x_i \mathrm{diag}(A_{:,i}) \succeq 0
\end{aligned}
$$

[4] А если это не так? Что тогда? Всегда ли есть сведение к такому виду?
[5] Подозрительно похоже на регуляризацию.

Тем не менее, SDP более выразительно, чем LP:

$$
\begin{array}{l} \min\limits_x \dfrac{\langle c, x\rangle^2}{\langle d, x\rangle} \\ \text{s.t. } Ax + b \succeq 0 \end{array}
\quad\longrightarrow\quad
\begin{array}{c} \min\limits_x t \\ \text{s.t. } Ax + b \succeq 0, \\ \dfrac{\langle c, x\rangle^2}{\langle d, x\rangle} \leqslant t, \\ t \geqslant 0 \end{array}
\quad\longrightarrow\quad
\begin{array}{l} \min\limits_x t \\ \text{s.t. } \begin{bmatrix} \operatorname{diag}(Ax - b) & 0 & 0 \\ 0 & t & c^\top x \\ 0 & c^\top x & d^\top x \end{bmatrix} \succeq 0 \end{array}
$$

Такую задачу в форме LP уже не записать.
Важным инструментом при сведении задачи к такой форме является дополнение по Шуру:

Утверждение 6.1 (*Критерий положительной полуопределённости*).
Пусть дана блочно-диагональная матрица M:

$$M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$$

Дополнением по Шуру блока D называется матрица $A - BD^{-1}C$.
Аналогично для блока A: $D - CA^{-1}B$.
Тогда верно, что: симметричная матрица $M \succeq 0 \iff D \succeq 0,\ A - BD^{-1}B^\top \succeq 0$.

6.1 Двойственность в SDP-задачах

Утверждение 6.2 (*О виде двойственной задачи*).

$$
\begin{array}{l} \min c^\top x \\ \text{s.t. } F_0 + \sum\limits_{i=1}^m x_i F_i \succeq 0 \end{array}
\quad\longrightarrow\quad
\begin{array}{c} \max\ -\operatorname{tr}(F_0 Z) \\ \text{s.t. } \operatorname{tr}(F_i Z) = c_i \\ Z \succeq 0 \end{array}
$$

Доказательство. Вид Лагранжиана следует из условий ККТ:

$$
\begin{aligned}
L(x, Z) &= c^\top x - \operatorname{tr}\left(Z^\top \left(F_0 + \sum_{i=1}^m x_i F_i\right)\right) = \\
&= c^\top x - \operatorname{tr}(Z^\top F_0) - \sum_{i=1}^m x_i \cdot \operatorname{tr}\left(Z^\top F_i\right) \\
\nabla_x L(x, Z) &= c - \begin{bmatrix} \operatorname{tr}\left(Z^\top F_1\right) & \dots & \operatorname{tr}\left(Z^\top F_m\right) \end{bmatrix}^\top = 0 \\
&\iff \operatorname{tr}\left(Z^\top F_i\right) = c_i
\end{aligned}
$$

Отсюда:

$$
\begin{aligned}
L(x^*, Z) &= \inf_x \left(c^\top x - \operatorname{tr}(Z^\top F_0) - \sum_{i=1}^m x_i \cdot \operatorname{tr}\left(Z^\top F_i\right)\right) = \\
&= -\operatorname{tr}(Z^\top F_0) =: g(Z),\ Z \succeq 0
\end{aligned}
$$

Итоговый вид двойственной задачи:

$$
\begin{array}{c} \max\ -\operatorname{tr}(F_0 W) \\ \text{s.t. } \operatorname{tr}(F_i W) = c_i \\ W \succeq 0 \end{array}
$$

где $W := Z^\top$. □

Сильной двойственности может не быть (хуже, чем LP), но зазор двойственности всегда конечен.

Теорема 6.1 (*О сильной двойственности в SDP*)**.**
Для SDP-задачи выполнена сильная двойственность, если хотя бы одно из условий выполнено:

- ◇ Прямая задача строго допустима, т.е. $\exists x : F(x) \succ 0$.
- ◇ Двойственная задача строго допустима, т.е. $\exists Z \in S^n_{++} \forall i = 1, \dots, m : \operatorname{tr}(ZF_i) = c_i$.

6.2 Минимизация спектрального радиуса

Пусть $A(x) = A_0 + \sum_{i=1}^m x_i A_i,\ \forall j > 0 : A_j = A_j^\top$ и поставлена задача минимизации:

$$\min_x \lambda_{\max}(A(x))$$

может быть переписана в SDP-виде:

$$\begin{aligned} &\min_{t,x} t \\ &\text{s.t. } tI - A(x) \succeq 0 \end{aligned}$$

Если матрица несимметрична/неквадратна, то минимизируется макс. сингулярное значение:

$$\min_x \sigma_{\max}(A(x))$$

что в SDP-терминах записывается как:

$$\begin{aligned} &\min_{t,x} t \\ &\text{s.t. } \begin{bmatrix} tI & A(x) \\ A^\top(x) & tI \end{bmatrix} \succeq 0 \end{aligned}$$

6.3 Поиск описанного эллипсоида минимального объёма

Пусть дан набор точек в $\mathbb{R}^d$ и нужно найти описанный вокруг них эллипс минимальной площади. Здесь нужно воспользоваться тем, что эллипс можно задать аффинным преобразованием:

$$\{x \mid \|x\|_2^2 \leqslant 1\} \to \{u \mid \|u\|_2^2 \leqslant 1,\ u = Ax + b\}$$

Площадь при этом увеличивается в $\det(A^{-1})$ раз, откуда возникает задача минимизации:

$$\begin{aligned} &\min_{A,b} -\log\det(A) \\ &\text{s.t. } \|Ax_i + b\|_2^2 \leqslant 1, \\ &\qquad A \succ 0 \end{aligned}$$

А уже ограничение на норму переписывается в SDP-виде:

$$\|Ax_i + b\|_2^2 \leqslant 1 \iff \begin{bmatrix} I & Ax_i + b \\ (Ax_i + b)^\top & 1 \end{bmatrix} \succeq 0$$

6.4 Приближённое решение невыпуклых задач

В курсе рассматривалась только задача вида:

$$\begin{aligned}&\min f_0(x)\\&\text{s.t. } f_i(x) \leqslant 0\end{aligned}$$

где $f_i(x) \leqslant x^\top A_i x + 2b_i^\top x + c_i$, но матрицы A_i неопределены.

1. Перепишем ограничения в матричном виде:

$$f_i(x) = \begin{bmatrix} x & 1 \end{bmatrix}^\top \begin{bmatrix} A_i & b_i \\ b_i^\top & c_i \end{bmatrix} \begin{bmatrix} x \\ 1 \end{bmatrix} \leqslant 0$$

2. Пусть x допустимая точка в исходной задаче, а $t, \tau_1, \ldots, \tau_M$ допустимые точки для введённой задаче, тогда можно выписать SDP-задачу:

$$\begin{aligned}&\max t\\ \text{s.t. } &\tau_i \geq 0\\ &\begin{bmatrix} A_0 & b_0 \\ b_0^\top & c_0 - t \end{bmatrix} + \tau_1 \begin{bmatrix} A_1 & b_1 \\ b_1^\top & c_1 \end{bmatrix} + \ldots + \tau_M \begin{bmatrix} A_M & b_M \\ b_M^\top & c_M \end{bmatrix} \succeq 0\end{aligned}$$

$$\begin{aligned}&0 \leq \begin{bmatrix} x \\ 1 \end{bmatrix}^\top \left(\begin{bmatrix} A_0 & b_0 \\ b_0 & c_0 - t \end{bmatrix} + \tau_1 \begin{bmatrix} A_1 & b_1 \\ b_1^\top & c_1 \end{bmatrix} \ldots + \tau_M \begin{bmatrix} A_M & b_M \\ b_M^\top & c_M \end{bmatrix} \right) \begin{bmatrix} x \\ 1 \end{bmatrix} =\\ &0 \leq f_0(x) - t + \tau_1 f_1(x) + \ldots + \tau_M f_M(x) \leq f_0(x) - t\end{aligned}$$

Отсюда $f_0(x) \geqslant t$, т.е. $\max t = \min f_0(x)$, задачи эквивалентны.
В частности, такой подход позволяет решать задачу о пересечении эллипсоидов в $\mathbb{R}^n$.

6.5 Реласкация задачи MAXCUT

Задача: найти макс.разрез в неориентированном графе $G = (V, E)$ с весами рёбер w_{ij}.

$$\begin{aligned}&\max \frac{1}{4} \sum_{i,j=1}^{n} (1 - x_i x_j) w_{ij}\\&\text{s.t. } x_i \in \{-1, 1\}\end{aligned}$$

У этой задачи есть SDP-релаксация:

$$X = xx^\top \to X \succeq xx^\top,\ X_{ii} = 1$$

$$\begin{aligned}&\max \frac{1}{4} \sum_{i,j=1}^{n} (1 - X_{ij}) w_{ij}\\&\text{s.t. } X_{ii} = 1\\&X \succeq 0\end{aligned}$$

Пусть $X = V^\top V$, а r — случайный вектор, равномерно распределённый на сфере.
Если $S = \{i \mid r^\top v_i \geqslant 0\}$, то

$$\mathbb{E}(\text{разрез}) \geqslant 0.876 \cdot MAXCUT$$

В целом, SDP часто применяется в комбинаторной оптимизации.

7 Методы внутренней точки

7.1 Задача выпуклой оптимизации с ограничениями типа равенств

$$\begin{aligned}
&\min f(x)\\
&\text{s.t. } Ax = b
\end{aligned}$$

где f — выпукла и дважды дифф-ма, $A \in \mathbb{R}^{p\times n}$, $\operatorname{rk}(A) = p < n$.
Двойственная задача:

$$\begin{aligned}
g(\mu) &= -b^\top \mu + \inf_x (f(x) + \mu^\top Ax) =\\
&= -b^\top \mu - \sup_x ((-A^\top \mu)^\top x - f(x)) =\\
&= -b^\top \mu - f^*(-A^\top \mu)
\end{aligned}$$

$$\begin{aligned}
&\max_\mu \; -b^\top \mu - f^*(-A^\top \mu)\\
&\text{s.t. } \mu \succeq 0
\end{aligned}$$

Условия оптимальности прямой задачи при этом имеют вид:

$$Ax^* = b, \quad \nabla f(x^*) + A^\top \mu^* = 0$$

$$\begin{bmatrix} \nabla f & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} x^* \\ \mu^* \end{bmatrix} = \begin{bmatrix} 0 \\ b \end{bmatrix}$$

Такую систему можно решать методом Ньютона, если f дважды дифференцируема с липшицевым гессианом и ограничена снизу. Причём часто минимизируют квадратичную аппроксимацию:

$$\begin{aligned}
&\min_v \; f(x) + \nabla f(x)^\top v + \frac{1}{2} v^\top \nabla^2 f(x) v\\
&\text{s.t. } A(x+v) = b
\end{aligned}$$

Условия оптимальности для такой задачи выгодно линеаризуются:

$$\begin{aligned}
A(x+v^*) = b &\iff Av^* = 0\\
\nabla f(x+v^*) + A^\top w &\approx \nabla f(x) + \nabla^2 f(x)^\top v^* + A^\top w = 0\\
&\implies \nabla^2 f(x)^\top v^* + A^\top w = -\nabla f(x)
\end{aligned}$$

Отсюда получаем систему:

$$\begin{bmatrix} \nabla^2 f(x) & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} = \begin{bmatrix} -\nabla f(x) \\ 0 \end{bmatrix}$$

Как понять, что пора прекращать итерации? По сути, мы ищем оптимальное квадратичное приближение на допустимом множестве. Остаток запишется так:

$$f(x) - \inf_v (\widehat{f}(x+v) \mid A(x+v) = b)$$

где $\widehat{f}$ — квадратичная аппроксимация функции f. Преобразуем одно из условий оптимальности. Пусть h — направление убывания в точке x. Тогда:

$$\begin{aligned}
\nabla^2 f(x)h + A^\top w &= -\nabla f(x) \mid h^\top\\
h^\top \nabla^2 f(x)h + (Ah)\top w &= -h^\top \nabla f(x) \; [Ah = 0]\\
h^\top \nabla^2 f(x) h &= -\nabla f(x)^\top h
\end{aligned}$$

$$\inf_v(\widehat{f}(x+v) \mid A(x+v)=b) = f(x) - \frac{1}{2}h^\top \nabla^2 f(x) h$$

Т.е. $h^\top \nabla^2 f(x) h$ — наиболее разумный критерий остановки метода Ньютона в такой задаче.

Теорема 7.1 (*О скорости сходимости*).
Скорость сходимости аналогична сходимости метода Ньютона в безусловной оптимизации.
Пусть выполнены следующие условия:

- ◇ Множество уровней $S = \{x \in D(f) \mid f(x) \leqslant f(x_0), Ax = b\}$ замкнуто, а $x_0 \in D(f),\ Ax_0 = b$.
- ◇ $\nabla^2 f$ липшицев на S.
- ◇ Норма обратной матрицы ККТ системы ограничена сверху.

Тогда метод Ньютона сходится к паре (x^*, μ^*) глобально линейно, но локально квадратично.

Если же начальная точка не лежит в допустимом множестве. Тогда ККТ можно записать так:

$$\begin{bmatrix} \nabla^2 f(x) & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} = - \begin{bmatrix} \nabla f(x) \\ Ax - b \end{bmatrix}$$

При таком подходе достаточно, чтобы $x_0 \in D(f)$, а такую точку найти сильно проще.

7.2 Общая задача выпуклой оптимизации. Барьерные методы

$$\begin{aligned} &\min_{x \in \mathbb{R}^n} f_0(x) \\ &\text{s.t. } f_i(x) \leqslant 0 \quad i = 1, \dots, m \\ &\qquad Ax = b \end{aligned}$$

где ограничения выпуклые, дважды непрерывно дифференцируемые, а $A \in \mathbb{R}^{p \times n},\ \operatorname{rk}(A) = p < n$.
Будем предполагать, что условия Слейтера выполнены, т.е. имеется сильная двойственность.
Условия ККТ:

$$\begin{aligned} &Ax^* = b,\ f_i(x^*) \leqslant 0 \\ &\lambda_i^* f_i(x^*) = 0 \\ &\lambda^* \succeq 0 \\ &\nabla_x \mathcal{L}(x^*, \mu^*, \lambda^*) = \nabla f_0(x^*) + \sum_{i=1}^m \lambda_i^* \nabla f_i(x^*) + A^\top \mu^* = 0 \end{aligned}$$

Ограничения типа неравенств можно перенести в целевую функцию, используя индикаторы:

$$I_-(u) = \begin{cases} 0, & u \leqslant 0 \\ +\infty, & u > 0 \end{cases}$$

Целевая функция при этом перестаёт быть дифференцируемой, в связи с чем индикатор приближают барьерами — последовательностью выпуклых дважды дифференцируемых функций $g(t, x)$, которая в пределе (при $t \to 0$) сходится к индикатору.

$$\begin{aligned} &\min_x f(x) + \sum_{i=1}^m -t \log(-f_i(x)) \\ &\text{s.t. } Ax = b \end{aligned}$$

Эту задачу уже можно решать, как описано в предыдущем разделе.
Тем не менее, барьерный метод требует допустимой начальной точки. Отсюда возникает естественное разделение алгоритма на две фазы: в первой ищется начальное приближение, а во второй оно используется на итерации барьерного метода.
Первую фазу можно реализовать, например, так:

$$\begin{gathered}\min_{x,s} \sum_{i=1}^{m} s_i \\ \text{s.t. } f_i(x) \leqslant s_i \\ Ax = b\end{gathered}$$