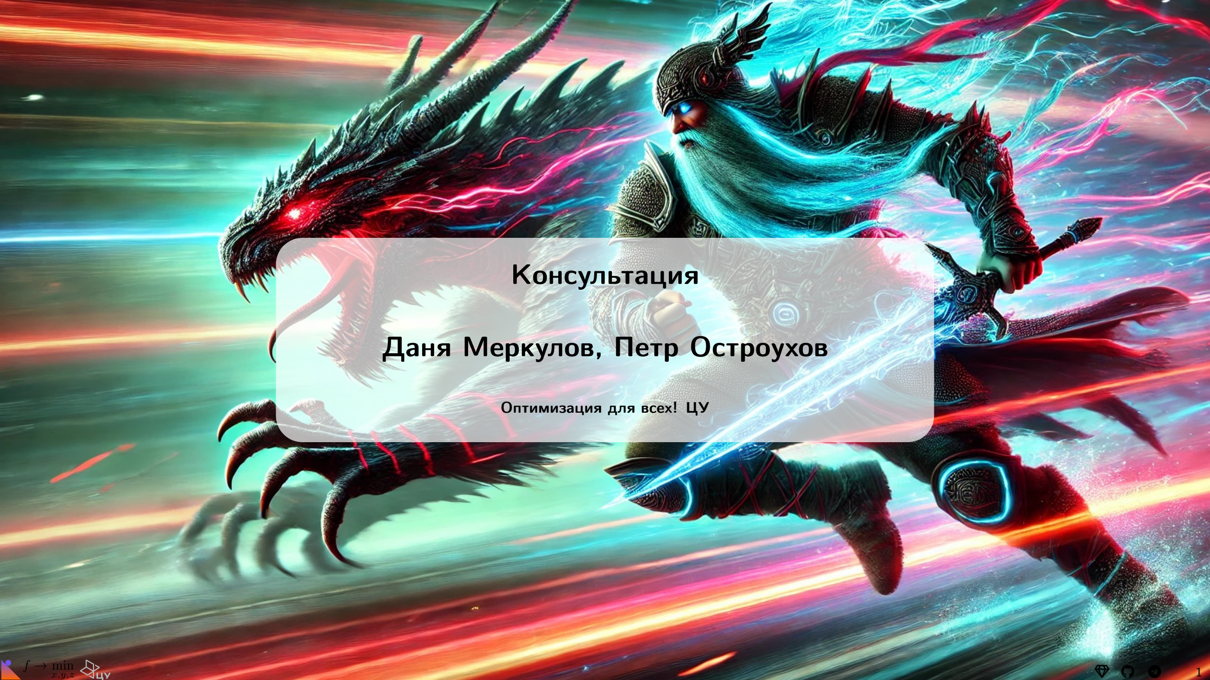


Консультация

Даня Меркулов, Петр Остроухов

Оптимизация для всех! ЦУ

Лекция 1. Вспоминаем линал, скорости сходимости.

Скорость сходимости

Линейная сходимость последовательности r_k определяется следующим образом:

Definition

Последовательность $\{r_k\}_{k=m}^\infty$ сходится линейно с параметром $0 < q < 1$, если существует константа $C > 0$ такая, что:

$$r_k \leq Cq^k, \quad \text{for all } k \geq m.$$

Если такое q существует, то последовательность называется линейно сходящейся. **Точная нижняя граница** всех q, удовлетворяющих неравенству, называется **константой линейной сходимости** последовательности.

Definition

Если последовательность r_k сходится к нулю, но не имеет линейной сходимости, то сходимость называется сублинейной. Иногда мы можем рассмотреть следующий частный случай сублинейной сходимости:

$$\|x_{k+1} - x^*\|_2 \leq Ck^q,$$

где $q < 0$ и $0 < C < \infty$. Интуитивно, сублинейная сходимость означает, что последовательность сходится медленнее любой геометрической прогрессии.

Скорость сходимости

Сходимость последовательности $\{r_k\}_{k=m}^{\infty}$ называется **сверхлинейной**, если она сходится к нулю быстрее любой линейно сходящейся последовательности.

Definition

- Последовательность $\{r_k\}_{k=m}^{\infty}$ сходится **сверхлинейно**, если она сходится линейно с параметром $q = 0$.

Скорость сходимости

Сходимость последовательности $\{r_k\}_{k=m}^{\infty}$ называется **сверхлинейной**, если она сходится к нулю быстрее любой линейно сходящейся последовательности.

Definition

- Последовательность $\{r_k\}_{k=m}^{\infty}$ сходится **сверхлинейно**, если она сходится линейно с параметром $q = 0$.
- Для $p > 1$, последовательность $\{r_k\}_{k=m}^{\infty}$ сходится **сверхлинейно порядка** p, если существует $C > 0$ и $0 < q < 1$ такая, что:

$$r_k \leq Cq^{k^p}, \quad \text{for all } k \geq m.$$

Последовательность называется **сходящейся квадратично**, если

$$r_k \leq Cq^{2^k}, \quad \text{for all } k \geq m,$$

Скорость сходимости

Сходимость последовательности $\{r_k\}_{k=m}^\infty$ называется **сверхлинейной**, если она сходится к нулю быстрее любой линейно сходящейся последовательности.

Definition

- Последовательность $\{r_k\}_{k=m}^\infty$ сходится **сверхлинейно**, если она сходится линейно с параметром $q = 0$.
- Для $p > 1$, последовательность $\{r_k\}_{k=m}^\infty$ сходится **сверхлинейно порядка** p, если существует $C > 0$ и $0 < q < 1$ такая, что:

  $$r_k \leq Cq^{k^p}, \quad \text{for all } k \geq m.$$

  Последовательность называется **сходящейся квадратично**, если

  $$r_k \leq Cq^{2^k}, \quad \text{for all } k \geq m,$$

- Для $p > 1$, последовательность $\{r_k\}_{k=m}^\infty$ сходится **сверхлинейно порядка** p, если существует $C > 0$ такая, что:

  $$r_k \leq Cr_{k-1}^p, \quad \text{for all } k \geq m.$$

  Когда $p = 2$, это называется **квадратичной сходимостью**.

Важный пример

Предположим, что $x^* = 1.23456789$ (истинное решение), и $x = 1.234$ $r_k = \|x - x^*\| = 0.00056789 \leq 10^{-3}$.

1. После первой итерации:

$$r_{k+1} \approx r_k^2 \leq (10^{-3})^2 = 10^{-6}.$$

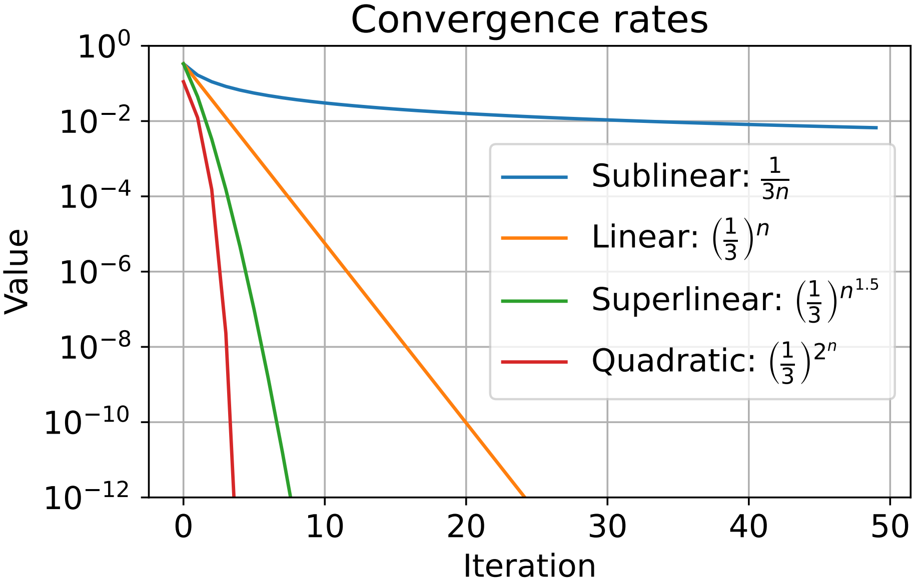
   Теперь ошибка равна 10^{-6}, и мы имеем 6 правильных значащих цифр ($x = 1.23456$).

Важный пример

Предположим, что $x^* = 1.23456789$ (истинное решение), и $x = 1.234$ $r_k = \|x - x^*\| = 0.00056789 \leq 10^{-3}$.

1. После первой итерации:
$$r_{k+1} \approx r_k^2 \leq (10^{-3})^2 = 10^{-6}.$$
Теперь ошибка равна 10^{-6}, и мы имеем 6 правильных значащих цифр ($x = 1.23456$).
2. После второй итерации:
$$r_{k+2} \approx r_{k+1}^2 = (10^{-6})^2 = 10^{-12}.$$
Теперь ошибка равна 10^{-12}, и мы имеем 12 правильных значащих цифр (1.234567890123).

Скорость сходимости



Задача. Знайте свое скалярное произведение.

Упростите следующее выражение:

$$\sum_{i=1}^{n} \langle S^{-1} a_i, a_i \rangle,$$

где $S = \sum\limits_{i=1}^{n} a_i a_i^T, a_i \in \mathbb{R}^n, \det(S) \neq 0$

Задача. Простая, но важная идея о матричных вычислениях.

Предположим, у вас есть следующее выражение

$$b = A_1 A_2 A_3 x,$$

где $A_1, A_2, A_3 \in \mathbb{R}^{3\times 3}$ - случайные квадратные плотные матрицы, и $x \in \mathbb{R}^n$ - вектор. Вам нужно вычислить b.

Какой способ лучше всего использовать?

1. $A_1 A_2 A_3 x$ (слева направо)

Проверьте простой код после вашего интуитивного ответа.

Задача. Простая, но важная идея о матричных вычислениях.

Предположим, у вас есть следующее выражение

$$b = A_1 A_2 A_3 x,$$

где $A_1, A_2, A_3 \in \mathbb{R}^{3 \times 3}$ - случайные квадратные плотные матрицы, и $x \in \mathbb{R}^n$ - вектор. Вам нужно вычислить b.

Какой способ лучше всего использовать?

1. $A_1 A_2 A_3 x$ (слева направо)
2. $\left(A_1 \left(A_2 \left(A_3 x\right)\right)\right)$ (справа налево)

Проверьте простой 🐍 код после вашего интуитивного ответа.

Задача. Простая, но важная идея о матричных вычислениях.

Предположим, у вас есть следующее выражение

$$b = A_1 A_2 A_3 x,$$

где $A_1, A_2, A_3 \in \mathbb{R}^{3 \times 3}$ - случайные квадратные плотные матрицы, и $x \in \mathbb{R}^n$ - вектор. Вам нужно вычислить b.

Какой способ лучше всего использовать?

1. $A_1 A_2 A_3 x$ (слева направо)
2. $\left(A_1 \left(A_2 \left(A_3 x\right)\right)\right)$ (справа налево)
3. Не имеет значения

Проверьте простой код после вашего интуитивного ответа.

Задача. Простая, но важная идея о матричных вычислениях.

Предположим, у вас есть следующее выражение

$$b = A_1 A_2 A_3 x,$$

где $A_1, A_2, A_3 \in \mathbb{R}^{3 \times 3}$ - случайные квадратные плотные матрицы, и $x \in \mathbb{R}^n$ - вектор. Вам нужно вычислить b.

Какой способ лучше всего использовать?

1. $A_1 A_2 A_3 x$ (слева направо)
2. $\left(A_1 \left(A_2 \left(A_3 x\right)\right)\right)$ (справа налево)
3. Не имеет значения
4. Результаты первых двух вариантов не будут одинаковыми.

Проверьте простой 🐍 код после вашего интуитивного ответа.

Лекция 2. Одномерная оптимизация. Градиент. Гессиан. Матрично-векторное дифференцирование.

Градиент, Гессиан

Пусть $f(x) : \mathbb{R}^n \to \mathbb{R}$, тогда вектор, который содержит все первые частные производные:

$$\nabla f(x) = \dfrac{df}{dx} = \begin{pmatrix} \dfrac{\partial f}{\partial x_1} \\ \dfrac{\partial f}{\partial x_2} \\ \vdots \\ \dfrac{\partial f}{\partial x_n} \end{pmatrix}$$

называется градиентом функции $f(x)$. Этот вектор указывает направление наискорейшего возрастания.

Пусть $f(x) : \mathbb{R}^n \to \mathbb{R}$, тогда матрица, содержащая все вторые частные производные:

$$f''(x) = \nabla^2 f(x) = \dfrac{\partial^2 f}{\partial x_i \partial x_j} = \begin{pmatrix} \dfrac{\partial^2 f}{\partial x_1 \partial x_1} & \dfrac{\partial^2 f}{\partial x_1 \partial x_2} & \dots & \dfrac{\partial^2 f}{\partial x_1 \partial x_n} \\ \dfrac{\partial^2 f}{\partial x_2 \partial x_1} & \dfrac{\partial^2 f}{\partial x_2 \partial x_2} & \dots & \dfrac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \dfrac{\partial^2 f}{\partial x_n \partial x_1} & \dfrac{\partial^2 f}{\partial x_n \partial x_2} & \dots & \dfrac{\partial^2 f}{\partial x_n \partial x_n} \end{pmatrix}$$

называется Гессианом.

Якобиан

Обобщением понятия градиента на случай многомерной функции $f(x) : \mathbb{R}^n \to \mathbb{R}^m$ является следующая матрица:

$$
J_f = f'(x) = \frac{df}{dx^T} = \begin{pmatrix}
\frac{\partial f_1}{\partial x_1} & \frac{\partial f_2}{\partial x_1} & \dots & \frac{\partial f_m}{\partial x_1} \\
\frac{\partial f_1}{\partial x_2} & \frac{\partial f_2}{\partial x_2} & \dots & \frac{\partial f_m}{\partial x_2} \\
\vdots & \vdots & \ddots & \vdots \\
\frac{\partial f_1}{\partial x_n} & \frac{\partial f_2}{\partial x_n} & \dots & \frac{\partial f_m}{\partial x_n}
\end{pmatrix}
$$

Она содержит информацию о скорости изменения функции по отношению к ее входу.

Итог

$$
f(x) : X \to Y; \qquad \frac{\partial f(x)}{\partial x} \in G
$$

X	Y	G	Name
$\mathbb{R}$	$\mathbb{R}$	$\mathbb{R}$	$f'(x)$ (производная)
$\mathbb{R}^n$	$\mathbb{R}$	$\mathbb{R}^n$	$\dfrac{\partial f}{\partial x_i}$ (градиент)
$\mathbb{R}^n$	$\mathbb{R}^m$	$\mathbb{R}^{n \times m}$	$\dfrac{\partial f_i}{\partial x_j}$ (якобиан)

Аппроксимации Тейлора

- Если $f : \mathbb{R}^n \to \mathbb{R}$ - дифференцируемая функция, то ее аппроксимация первого порядка задается следующим образом:

$$f_{x_0}^I(x) = f(x_0) + \nabla f(x_0)^T (x - x_0)$$

где: $f(x_0)$ - значение функции в точке x_0. $\nabla f(x_0)$ - градиент функции в точке x_0.

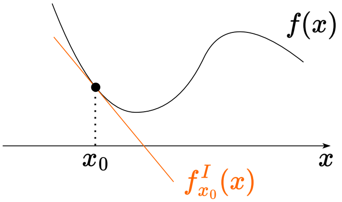
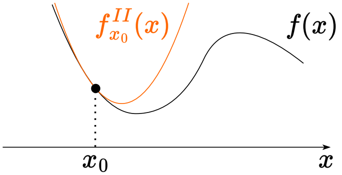


Рис. 2: Аппроксимация Тейлора первого порядка в окрестности точки x_0



Аппроксимации Тейлора

- Если $f : \mathbb{R}^n \to \mathbb{R}$ - дифференцируемая функция, то ее аппроксимация первого порядка задается следующим образом:

$$f_{x_0}^I(x) = f(x_0) + \nabla f(x_0)^T (x - x_0)$$

где: $f(x_0)$ - значение функции в точке x_0. $\nabla f(x_0)$ - градиент функции в точке x_0.

- Для дважды дифференцируемой функции $f : \mathbb{R}^n \to \mathbb{R}$, ее аппроксимация второго порядка, строящаяся вблизи некоторой точки x_0, задается следующим образом:

$$f_{x_0}^{II}(x) = f(x_0) + \nabla f(x_0)^T (x - x_0) + \frac{1}{2}(x - x_0)^T \nabla^2 f(x_0)(x - x_0)$$

где $\nabla^2 f(x_0)$ - гессиан функции f в точке x_0.

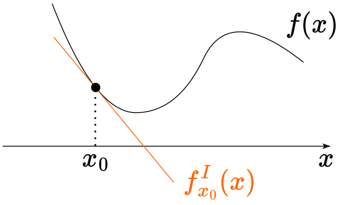
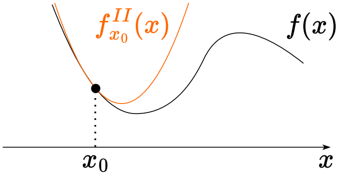


Рис. 2: Аппроксимация Тейлора первого порядка в окрестности точки x_0



Матрично-векторное дифференцирование через дифференциал

После получения дифференциальной записи df мы можем получить градиент, используя следующую формулу:

$$df(x) = \langle \nabla f(x), dx \rangle$$

Матрично-векторное дифференцирование через дифференциал

После получения дифференциальной записи df мы можем получить градиент, используя следующую формулу:

$$df(x) = \langle \nabla f(x), dx \rangle$$

Далее, если у нас есть дифференциал в такой форме и мы хотим вычислить вторую производную матричной/векторной функции, мы рассматриваем "старый" dx как константу dx_1, затем вычисляем $d(df) = d^2 f(x)$

$$d^2 f(x) = \langle \nabla^2 f(x) dx_1, dx \rangle = \langle H_f(x) dx_1, dx \rangle$$

Пример

Example

Найти $df, \nabla f(x)$, если $f(x) = \ln\langle x, Ax\rangle$.

Пример

Example

Найти $df, \nabla f(x)$, если $f(x) = \ln\langle x, Ax\rangle$.

1. Заметим, что A должна быть положительно определенной, потому что $\langle x, Ax\rangle$ аргумент логарифма и для любого x формула должна быть положительной. Таким образом, $A \in \mathbb{S}^n_{++}$ Давайте сначала найдем дифференциал:

$$
df = d\left(\ln\langle x, Ax\rangle\right) = \frac{d\left(\langle x, Ax\rangle\right)}{\langle x, Ax\rangle} = \frac{\langle dx, Ax\rangle + \langle x, d(Ax)\rangle}{\langle x, Ax\rangle} =
$$

$$
= \frac{\langle Ax, dx\rangle + \langle x, Adx\rangle}{\langle x, Ax\rangle} = \frac{\langle Ax, dx\rangle + \langle A^T x, dx\rangle}{\langle x, Ax\rangle} = \frac{\langle (A + A^T)x, dx\rangle}{\langle x, Ax\rangle}
$$

Пример

Example

Найти $df, \nabla f(x)$, если $f(x) = \ln\langle x, Ax\rangle$.

1. Заметим, что A должна быть положительно определенной, потому что $\langle x, Ax\rangle$ аргумент логарифма и для любого x формула должна быть положительной. Таким образом, $A \in \mathbb{S}^n_{++}$ Давайте сначала найдем дифференциал:

$$
df = d\left(\ln\langle x, Ax\rangle\right) = \frac{d\left(\langle x, Ax\rangle\right)}{\langle x, Ax\rangle} = \frac{\langle dx, Ax\rangle + \langle x, d(Ax)\rangle}{\langle x, Ax\rangle} =
$$

$$
= \frac{\langle Ax, dx\rangle + \langle x, Adx\rangle}{\langle x, Ax\rangle} = \frac{\langle Ax, dx\rangle + \langle A^T x, dx\rangle}{\langle x, Ax\rangle} = \frac{\langle (A + A^T)x, dx\rangle}{\langle x, Ax\rangle}
$$

2. Наша основная цель - получить форму $df = \langle \cdot, dx\rangle$

$$
df = \left\langle \frac{2Ax}{\langle x, Ax\rangle}, dx\right\rangle
$$

Таким образом, градиент равен $\nabla f(x) = \dfrac{2Ax}{\langle x, Ax\rangle}$

Линейный поиск: унимодальная функция

- Задача одномерного линейного поиска: $\min_{\alpha \in \mathbb{R}} f(\alpha)$ (пример - поиск оптимального размера шага градиентного спуска).

Линейный поиск: унимодальная функция

- Задача одномерного линейного поиска: $\min_{\alpha \in \mathbb{R}} f(\alpha)$ (пример - поиск оптимального размера шага градиентного спуска).
- Функция $f(\alpha)$ предполагается унимодальной на выбранном интервале, то есть имеет единственный локальный (и глобальный) минимум.

Линейный поиск: унимодальная функция

- Задача одномерного линейного поиска: $\min_{\alpha \in \mathbb{R}} f(\alpha)$ (пример - поиск оптимального размера шага градиентного спуска).
- Функция $f(\alpha)$ предполагается унимодальной на выбранном интервале, то есть имеет единственный локальный (и глобальный) минимум.
- Метод дихотомии сужает интервал $[a, b]$, пока его длина не станет меньше заданной точности, требуя два новых значения функции на шаг.

Линейный поиск: унимодальная функция

- Задача одномерного линейного поиска: $\min_{\alpha \in \mathbb{R}} f(\alpha)$ (пример - поиск оптимального размера шага градиентного спуска).
- Функция $f(\alpha)$ предполагается унимодальной на выбранном интервале, то есть имеет единственный локальный (и глобальный) минимум.
- Метод дихотомии сужает интервал $[a, b]$, пока его длина не станет меньше заданной точности, требуя два новых значения функции на шаг.
- Метод золотого сечения переиспользует одно из предыдущих значений, сокращая интервал в 0.618 раз на итерацию и снижая число вычислений f.

Неточный линейный поиск

Нам не всегда нужно точно решать задачу минимизации. Иногда, достаточно найти приближенное решение. Такое часто встречается в задаче выбора шага метода оптимизации

$$
\begin{aligned}
x_{k+1} &= x_k - \alpha \nabla f(x_k) \\
\alpha &= \text{argmin } f(x_{k+1})
\end{aligned}
$$

Неточный линейный поиск

Нам не всегда нужно точно решать задачу минимизации. Иногда, достаточно найти приближенное решение. Такое часто встречается в задаче выбора шага метода оптимизации

$$
\begin{aligned}
x_{k+1} &= x_k - \alpha \nabla f(x_k) \\
\alpha &= \text{argmin } f(x_{k+1})
\end{aligned}
$$

Рассмотрим скалярную функцию $\phi(\alpha)$ в точке x_k:

$$
\phi(\alpha) = f(x_k - \alpha \nabla f(x_k)), \alpha \geq 0
$$

Неточный линейный поиск

Нам не всегда нужно точно решать задачу минимизации. Иногда, достаточно найти приближенное решение. Такое часто встречается в задаче выбора шага метода оптимизации

$$
\begin{aligned}
x_{k+1} &= x_k - \alpha \nabla f(x_k) \\
\alpha &= \text{argmin } f(x_{k+1})
\end{aligned}
$$

Рассмотрим скалярную функцию $\phi(\alpha)$ в точке x_k:

$$
\phi(\alpha) = f(x_k - \alpha \nabla f(x_k)), \alpha \geq 0
$$

Первое приближение $\phi(\alpha)$ в окрестности $\alpha = 0$ равно:

$$
\phi(\alpha) \approx \phi_0^I(\alpha) = f(x_k) - \alpha \nabla f(x_k)^T \nabla f(x_k)
$$

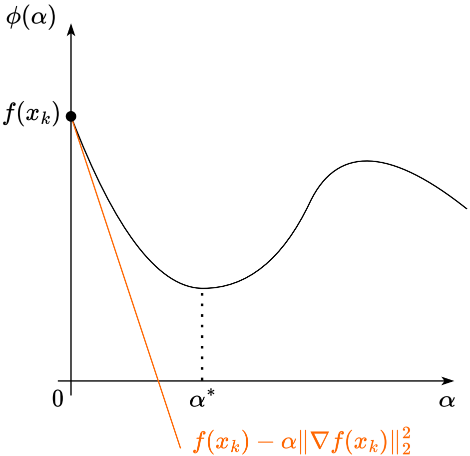


Рис. 4: иллюстрация аппроксимации тейлора $\phi_0^i(\alpha)$

Неточный линейный поиск. Условия Гольдштейна

Рассмотрим две линейные скалярные функции $\phi_1(\alpha)$ и $\phi_2(\alpha)$:

$$
\phi_1(\alpha) = f(x_k) - c_1\alpha\|\nabla f(x_k)\|^2
$$

$$
\phi_2(\alpha) = f(x_k) - c_2\alpha\|\nabla f(x_k)\|^2
$$

Неточный линейный поиск. Условия Гольдштейна

Рассмотрим две линейные скалярные функции $\phi_1(\alpha)$ и $\phi_2(\alpha)$:

$$\phi_1(\alpha) = f(x_k) - c_1\alpha\|\nabla f(x_k)\|^2$$

$$\phi_2(\alpha) = f(x_k) - c_2\alpha\|\nabla f(x_k)\|^2$$

Условия Гольдштейна-Армихо находят функцию $\phi(\alpha)$ между $\phi_1(\alpha)$ и $\phi_2(\alpha)$. Обычно $c_1 = \rho$ и $c_2 = 1 - \rho$, с $\rho \in (0, 0.5)$.

Ограничение только сверху задает условие Армихо (достаточного убывания).

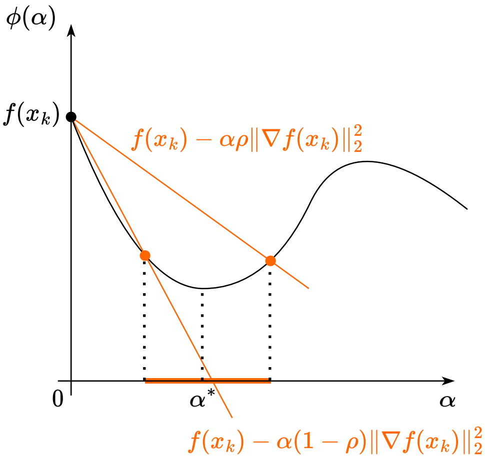


Рис. 5: Иллюстрация условий Гольдштейна

Неточный линейный поиск. Условие ограничения на кривизну

Чтобы избежать слишком коротких шагов, мы вводим второй критерий:

$$-\nabla f(x_k - \alpha \nabla f(x_k))^T \nabla f(x_k) \geq c_2 \nabla f(x_k)^T (-\nabla f(x_k))$$

Неточный линейный поиск. Условие ограничения на кривизну

Чтобы избежать слишком коротких шагов, мы вводим второй критерий:

$$-\nabla f(x_k - \alpha \nabla f(x_k))^T \nabla f(x_k) \geq c_2 \nabla f(x_k)^T (-\nabla f(x_k))$$

для некоторого $c_2 \in (c_1, 1)$. Здесь c_1 из условия Армихо.
Левая часть является производной $\nabla_\alpha \phi(\alpha)$, гарантирующей, что наклон $\phi(\alpha)$ в целевой точке не менее чем в c_2 раз больше начального наклона $\nabla_\alpha \phi(\alpha)(0)$.

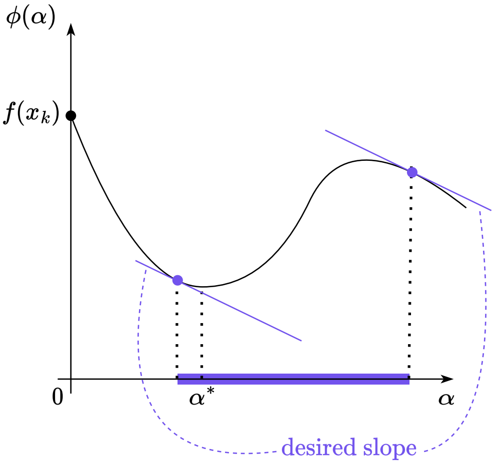


Рис. 6: Иллюстрация условия ограничения на кривизну

Неточный линейный поиск. Условия Вульфа

$$
f(x_k - \alpha \nabla f(x_k)) \leq f(x_k) - c_1 \cdot \alpha \nabla f(x_k)^T \nabla f(x_k),
$$
$$
-\nabla f(x_k - \alpha \nabla f(x_k))^T \nabla f(x_k) \geq c_2 \nabla f(x_k)^T (-\nabla f(x_k))
$$

Вместе, условие Армихо и ограничение на кривизну образуют условия Вульфа.

Theorem

Пусть

1. $f : \mathbb{R}^n \to \mathbb{R}$ непрерывно дифференцируема,
   $\phi(\alpha) = f(x_k - \alpha \nabla f(x_k))$.

Тогда для $0 < c_1 < c_2 < 1$, существуют интервалы шагов, удовлетворяющие условиям Вульфа.

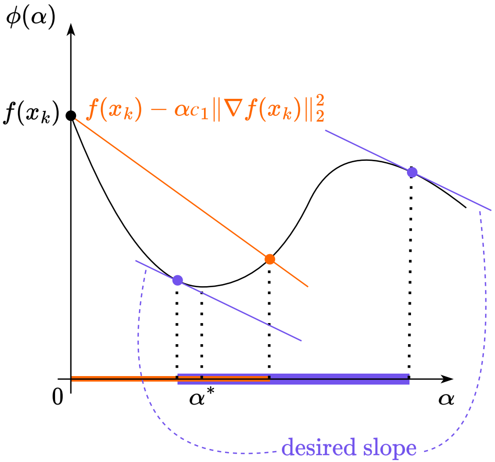


Рис. 7: Иллюстрация условий Вульфа

Неточный линейный поиск. Условия Вульфа

$$
f(x_k - \alpha \nabla f(x_k)) \leq f(x_k) - c_1 \cdot \alpha \nabla f(x_k)^T \nabla f(x_k),
$$
$$
-\nabla f(x_k - \alpha \nabla f(x_k))^T \nabla f(x_k) \geq c_2 \nabla f(x_k)^T (-\nabla f(x_k))
$$

Вместе, условие Армихо и ограничение на кривизну образуют условия Вульфа.

Theorem

Пусть

1. $f : \mathbb{R}^n \to \mathbb{R}$ непрерывно дифференцируема, $\phi(\alpha) = f(x_k - \alpha \nabla f(x_k))$.
2. $\nabla f(x_k)^T p_k < 0$, где $p_k = -\nabla f(x_k)$, делая p_k направлением спуска.

Тогда для $0 < c_1 < c_2 < 1$, существуют интервалы шагов, удовлетворяющие условиям Вульфа.

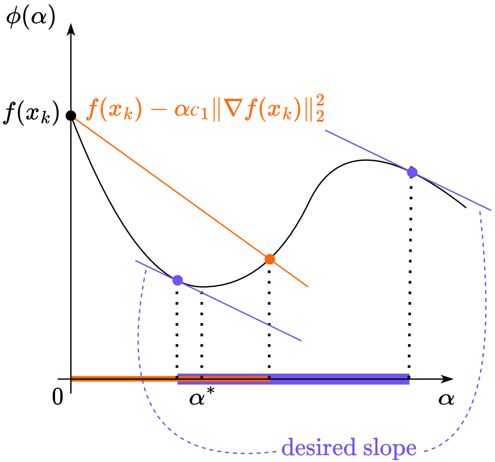


Рис. 7: Иллюстрация условий Вульфа

Неточный линейный поиск. Условия Вульфа

$$f(x_k - \alpha \nabla f(x_k)) \leq f(x_k) - c_1 \cdot \alpha \nabla f(x_k)^T \nabla f(x_k),$$
$$-\nabla f(x_k - \alpha \nabla f(x_k))^T \nabla f(x_k) \geq c_2 \nabla f(x_k)^T (-\nabla f(x_k))$$

Вместе, условие Армихо и ограничение на кривизну образуют условия Вульфа.

Theorem

Пусть
1. $f: \mathbb{R}^n \to \mathbb{R}$ непрерывно дифференцируема, $\phi(\alpha) = f(x_k - \alpha \nabla f(x_k))$.
2. $\nabla f(x_k)^T p_k < 0$, где $p_k = -\nabla f(x_k)$, делая p_k направлением спуска.
3. f ограничена снизу вдоль луча $\{x_k + \alpha p_k \mid \alpha > 0\}$

Тогда для $0 < c_1 < c_2 < 1$, существуют интервалы шагов, удовлетворяющие условиям Вульфа.

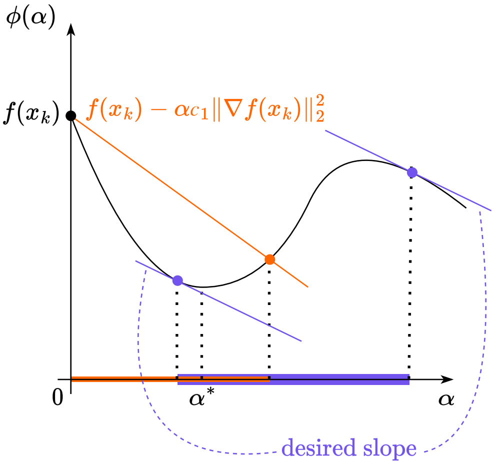


Рис. 7: Иллюстрация условий Вульфа

Неточный линейный поиск: бэктрекинг

- Алгоритм стартует с некоторого большого α_0 и параметров $\beta \in (0,1)$, $c_1 \in (0,1)$.

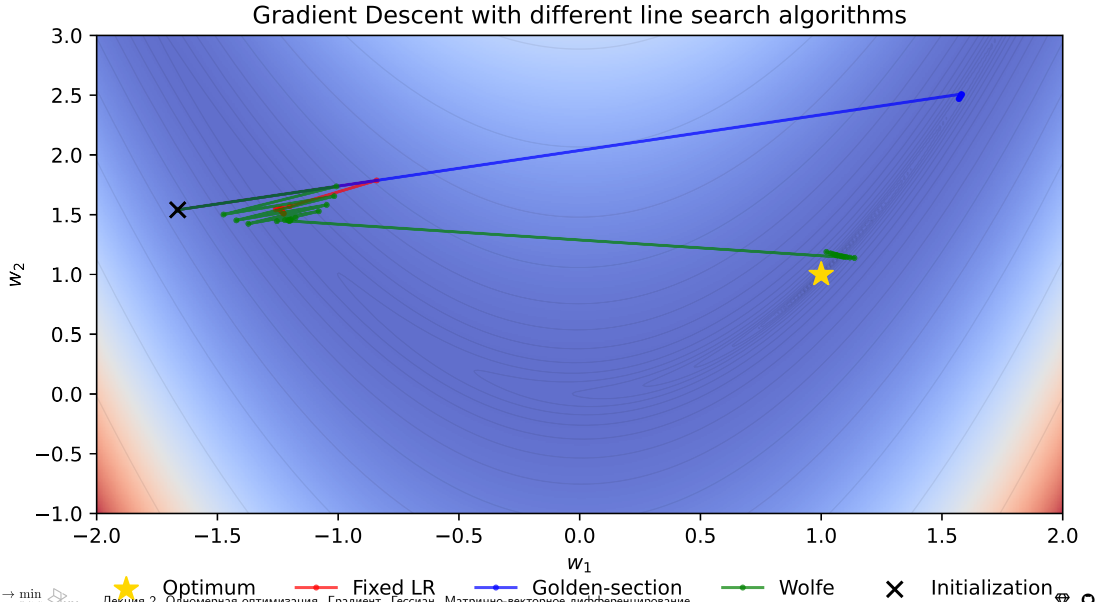
Неточный линейный поиск: бэктрекинг

- Алгоритм стартует с некоторого большого α_0 и параметров $\beta \in (0, 1)$, $c_1 \in (0, 1)$.
- Пока условие (например, Армихо) нарушено, заменяем $\alpha \leftarrow \beta\alpha$ и пересчитываем $f(x_k + \alpha p_k)$.

Неточный линейный поиск: бэктрекинг

- Алгоритм стартует с некоторого большого α_0 и параметров $\beta \in (0, 1)$, $c_1 \in (0, 1)$.
- Пока условие (например, Армихо) нарушено, заменяем $\alpha \leftarrow \beta\alpha$ и пересчитываем $f(x_k + \alpha p_k)$.
- Метод экономичен, так как требует по одному вычислению f на итерацию и быстро подбирает подходящий шаг.

Градиентный спуск с линейным поиском



Линейный поиск. Пример 1: Сравнение методов (Colab ♣)

$$f_1(x) = x(x-2)(x+2)^2 + 10$$
$$[a, b] = [-3, 2]$$

Случайный поиск: 72 вызова функции. 36 итераций. $f_1^* = 0.09$
Метод дихотомии: 23 вызова функции. 13 итераций. $f_1^* = 10.00$
Золотое сечение: 19 вызова функции. 18 итераций. $f_1^* = 10.00$
Параболический поиск: 20 вызова функции. 17 итераций.
$f_1^* = 10.00$

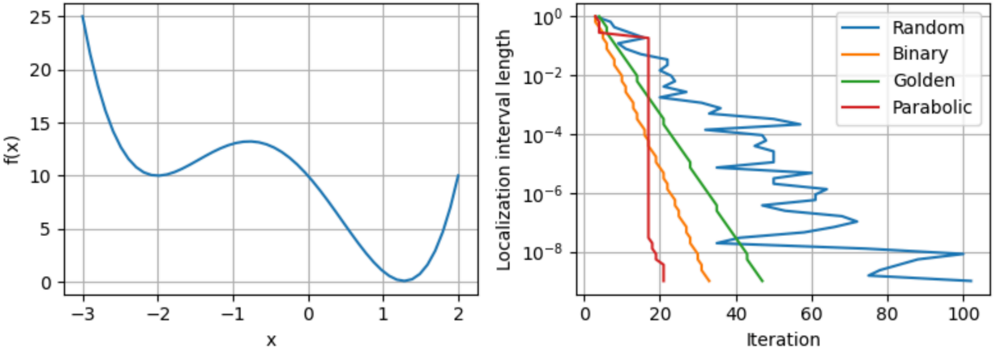


Рис. 8: Сравнение различных методов линейного поиска с f_1

Линейный поиск. Пример 2: Сравнение методов (Colab ♣)

$$f_2(x) = -\sqrt{\frac{2}{\pi}}\frac{x^2 e^{-\frac{x^2}{8}}}{8}$$

$$[a, b] = [0, 6]$$

Случайный поиск: 68 вызова функции. 34 итераций. $f_2^* = 0.71$
Метод дихотомии: 23 вызова функции. 13 итераций. $f_2^* = 0.71$
Золотое сечение: 20 вызова функции. 19 итераций. $f_2^* = 0.71$
Параболический поиск: 17 вызова функции. 14 итераций.
$f_2^* = 0.71$

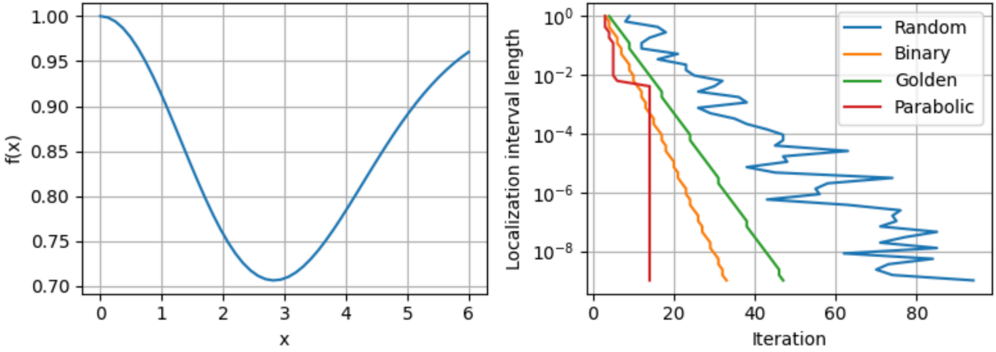


Рис. 9: Сравнение различных методов линейного поиска с f_2

Линейный поиск. Пример 3: Сравнение методов (Colab ♣)

$$f_3(x) = \sin\left(\sin\left(\sin\left(\sqrt{\frac{x}{2}}\right)\right)\right)$$

$$[a, b] = [5, 70]$$

Случайный поиск: 66 вызовов функции. 33 итерации. $f_3^* = 0.25$
Метод дихотомии: 32 вызова функции. 17 итераций. $f_3^* = 0.25$
Золотое сечение: 25 вызова функции. 24 итераций. $f_3^* = 0.25$
Параболический поиск: 103 вызова функции. 100 итераций. $f_3^* = 0.25$

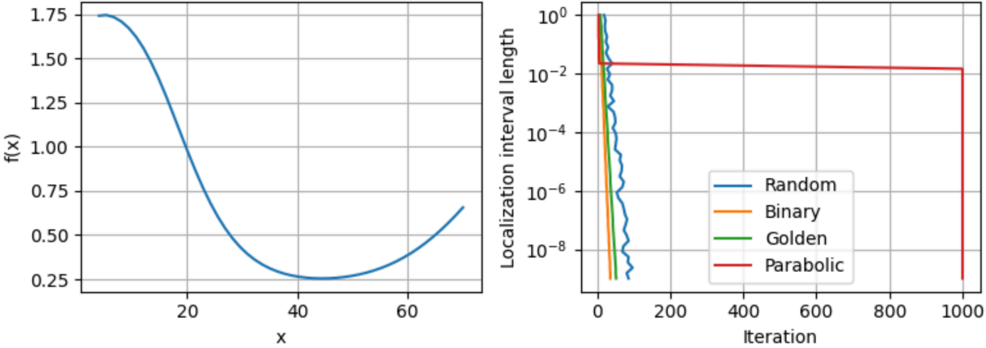


Рис. 10: Сравнение различных методов линейного поиска с f_3

Лекция 3. Автоматическое дифференцирование

Почему нужны градиенты

- Хотим решить

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

где d достигает миллиардов и каждая оценка $f(x)$ дорога.

Почему нужны градиенты

- Хотим решить
$$\min_{w \in \mathbb{R}^d} f(x)$$
где d достигает миллиардов и каждая оценка $f(x)$ дорога.
- Методы нулевого порядка, опирающиеся только на значения функции, в таких задачах быстро упираются в «проклятие размерности»: их скорость деградирует до $\mathcal{O}(n/k)$ против $\mathcal{O}(1/k)$ у градиентных методов.

Почему нужны градиенты

- Хотим решить
$$\min_{w \in \mathbb{R}^d} f(x)$$
где d достигает миллиардов и каждая оценка $f(x)$ дорога.
- Методы нулевого порядка, опирающиеся только на значения функции, в таких задачах быстро упираются в «проклятие размерности»: их скорость деградирует до $\mathcal{O}(n/k)$ против $\mathcal{O}(1/k)$ у градиентных методов.
- В сильно выпуклом случае сходимость безградиентных схем замедляется с $\left(1 - \frac{\mu}{L}\right)^k$ до $\left(1 - \frac{\mu}{nL}\right)^k$, а даже лучшие двухточечные оценки не дают зависимости лучше, чем $\sqrt{n}$.

Почему нужны градиенты

- Хотим решить
$$\min_{w \in \mathbb{R}^d} f(x)$$
где d достигает миллиардов и каждая оценка $f(x)$ дорога.
- Методы нулевого порядка, опирающиеся только на значения функции, в таких задачах быстро упираются в «проклятие размерности»: их скорость деградирует до $\mathcal{O}(n/k)$ против $\mathcal{O}(1/k)$ у градиентных методов.
- В сильно выпуклом случае сходимость безградиентных схем замедляется с $\left(1 - \frac{\mu}{L}\right)^k$ до $\left(1 - \frac{\mu}{nL}\right)^k$, а даже лучшие двухточечные оценки не дают зависимости лучше, чем $\sqrt{n}$.
- Отсюда мотивация: научиться вычислять полный градиент $\nabla_w L$ и переходить к методам первого порядка, которые масштабируются лучше в больших задачах.

Прямой режим автоматического дифференцирования

Чтобы глубже понять идею автоматического дифференцирования, рассмотрим простую функцию для вычисления производных:

$$L(w_1, w_2) = w_2 \log w_1 + \sqrt{w_2 \log w_1}$$

Прямой режим автоматического дифференцирования

Чтобы глубже понять идею автоматического дифференцирования, рассмотрим простую функцию для вычисления производных:

$$L(w_1, w_2) = w_2 \log w_1 + \sqrt{w_2 \log w_1}$$

Давайте нарисуем *вычислительный граф* этой функции:

$$L(w_1, w_2) = w_2 \log w_1 + \sqrt{w_2 \log w_1}$$

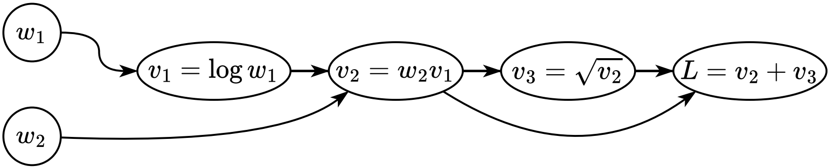


Рис. 11: Иллюстрация вычислительного графа для функции $L(w_1, w_2)$

Прямой режим автоматического дифференцирования

Чтобы глубже понять идею автоматического дифференцирования, рассмотрим простую функцию для вычисления производных:

$$L(w_1, w_2) = w_2 \log w_1 + \sqrt{w_2 \log w_1}$$

Давайте нарисуем *вычислительный граф* этой функции:

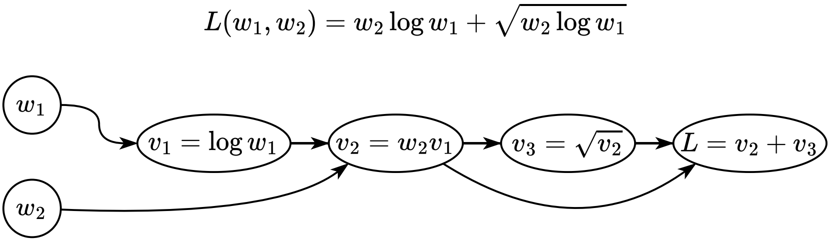


Рис. 11: Иллюстрация вычислительного графа для функции $L(w_1, w_2)$

Давайте пойдем от начала графа к концу и вычислим производную $\dfrac{\partial L}{\partial w_1}$.

Прямой режим автоматического дифференцирования

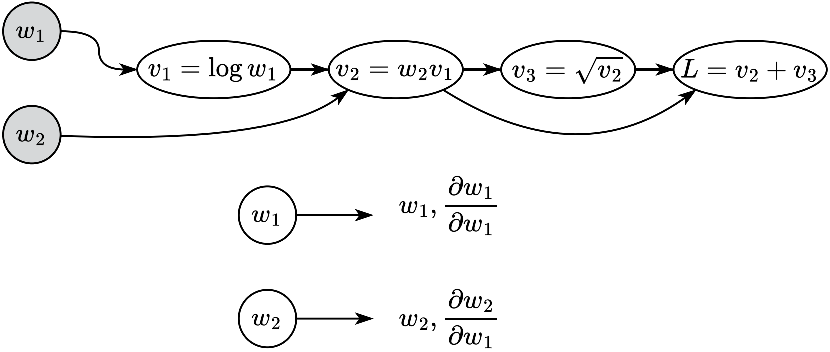


Рис. 12: Иллюстрация прямого режима автоматического дифференцирования

Функция

$w_1 = w_1, w_2 = w_2$

Прямой режим автоматического дифференцирования

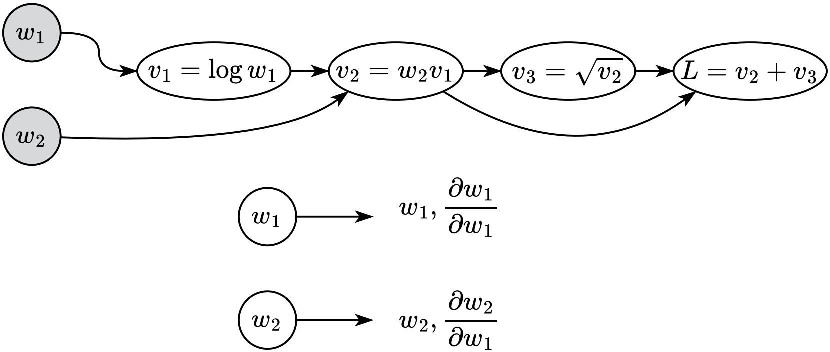


Рис. 12: Иллюстрация прямого режима автоматического дифференцирования

Функция

$w_1 = w_1, w_2 = w_2$

Производная

$$\frac{\partial w_1}{\partial w_1} = 1, \frac{\partial w_2}{\partial w_1} = 0$$

Прямой режим автоматического дифференцирования

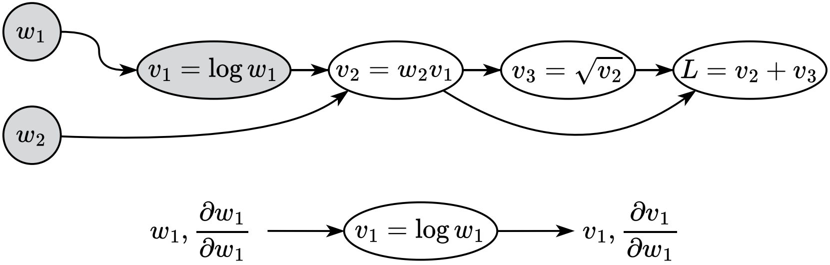


Рис. 13: Иллюстрация прямого режима автоматического дифференцирования

Прямой режим автоматического дифференцирования

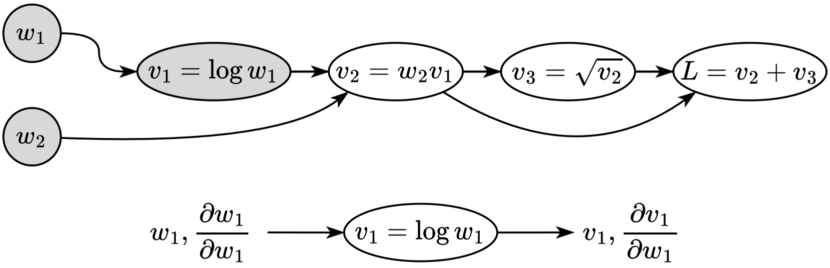


Рис. 13: Иллюстрация прямого режима автоматического дифференцирования

Функция

$v_1 = \log w_1$

Прямой режим автоматического дифференцирования

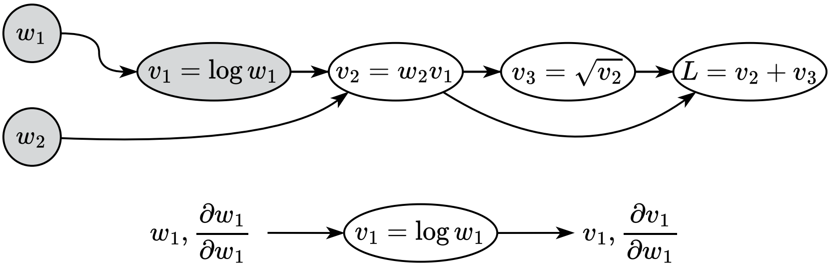


Рис. 13: Иллюстрация прямого режима автоматического дифференцирования

Функция

$v_1 = \log w_1$

Производная

$$\frac{\partial v_1}{\partial w_1} = \frac{\partial v_1}{\partial w_1}\frac{\partial w_1}{\partial w_1} = \frac{1}{w_1}1$$

Прямой режим автоматического дифференцирования

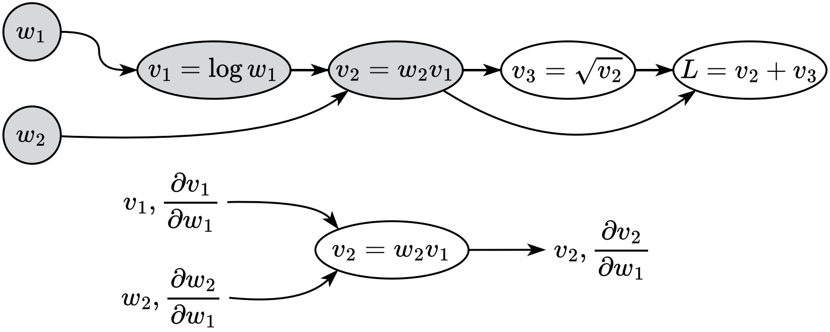


Рис. 14: Иллюстрация прямого режима автоматического дифференцирования

Прямой режим автоматического дифференцирования

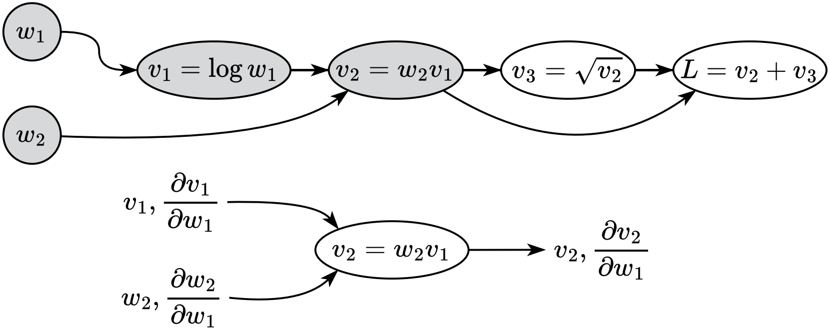


Рис. 14: Иллюстрация прямого режима автоматического дифференцирования

Функция

$v_2 = w_2 v_1$

Прямой режим автоматического дифференцирования

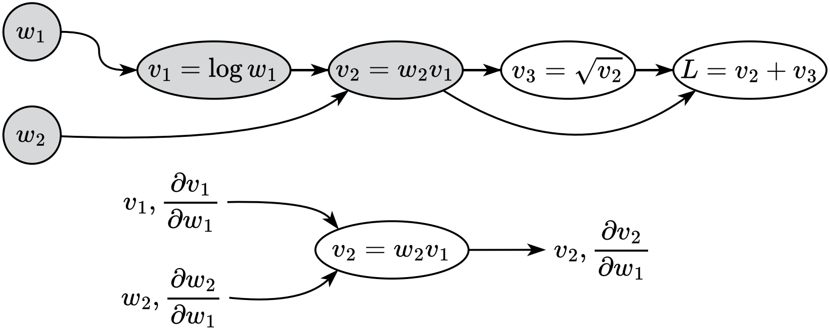


Рис. 14: Иллюстрация прямого режима автоматического дифференцирования

Функция

$$v_2 = w_2 v_1$$

Производная

$$\frac{\partial v_2}{\partial w_1} = \frac{\partial v_2}{\partial v_1}\frac{\partial v_1}{\partial w_1} + \frac{\partial v_2}{\partial w_2}\frac{\partial w_2}{\partial w_1} = w_2 \frac{\partial v_1}{\partial w_1} + v_1 \frac{\partial w_2}{\partial w_1}$$

Прямой режим автоматического дифференцирования

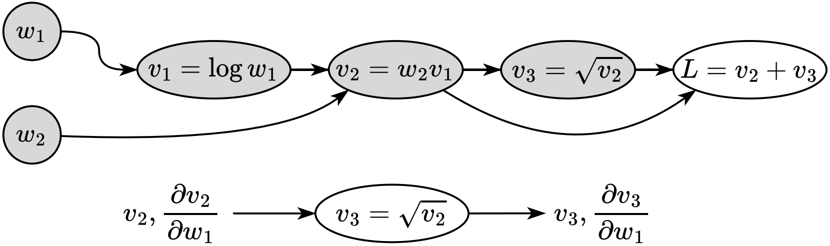


Рис. 15: Иллюстрация прямого режима автоматического дифференцирования

Прямой режим автоматического дифференцирования

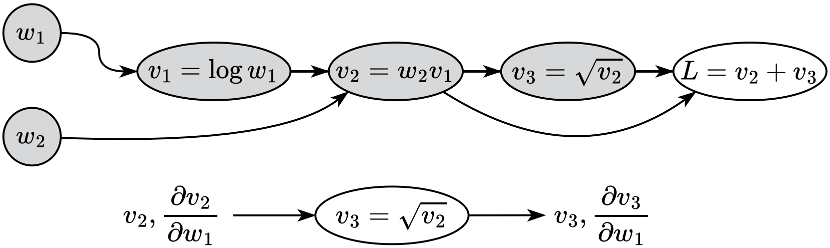


Рис. 15: Иллюстрация прямого режима автоматического дифференцирования

Функция

$v_3 = \sqrt{v_2}$

Прямой режим автоматического дифференцирования

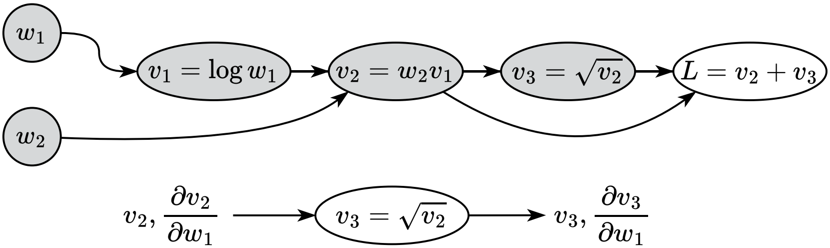


Рис. 15: Иллюстрация прямого режима автоматического дифференцирования

Функция

$$v_3 = \sqrt{v_2}$$

Производная

$$\frac{\partial v_3}{\partial w_1} = \frac{\partial v_3}{\partial v_2}\frac{\partial v_2}{\partial w_1} = \frac{1}{2\sqrt{v_2}}\frac{\partial v_2}{\partial w_1}$$

Прямой режим автоматического дифференцирования

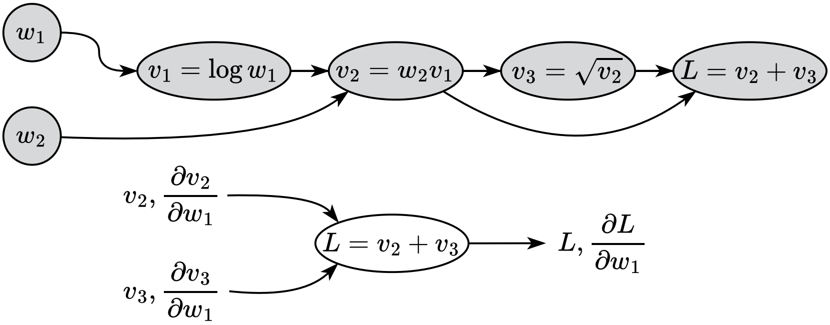
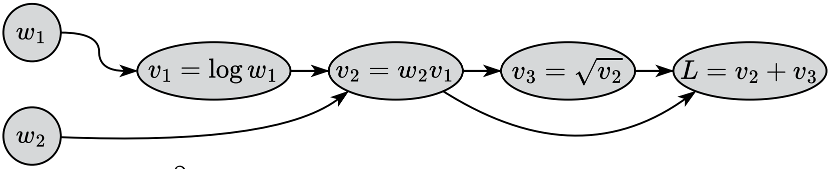


Рис. 16: Иллюстрация прямого режима автоматического дифференцирования

Прямой режим автоматического дифференцирования



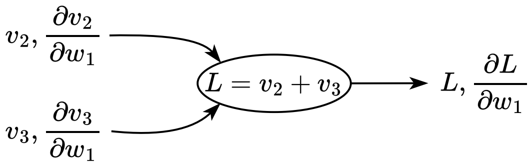


Рис. 16: Иллюстрация прямого режима автоматического дифференцирования

Функция

$L = v_2 + v_3$

Прямой режим автоматического дифференцирования

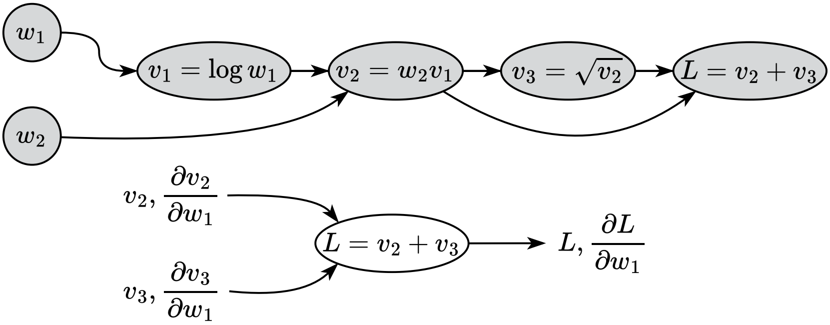


Рис. 16: Иллюстрация прямого режима автоматического дифференцирования

Функция

$$L = v_2 + v_3$$

Производная

$$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial v_2}\frac{\partial v_2}{\partial w_1} + \frac{\partial L}{\partial v_3}\frac{\partial v_3}{\partial w_1} = 1\frac{\partial v_2}{\partial w_1} + 1\frac{\partial v_3}{\partial w_1}$$

Обратный режим автоматического дифференцирования

Мы рассмотрим ту же функцию с вычислительным графом:

$$
L(w_1, w_2) = w_2 \log w_1 + \sqrt{w_2 \log w_1}
$$

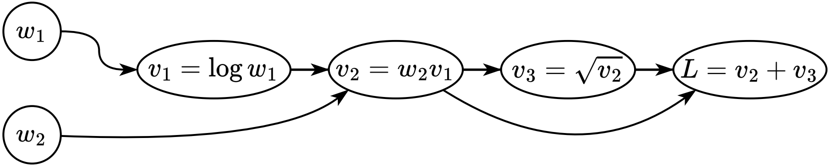


Рис. 17: Иллюстрация вычислительного графа для функции $L(w_1, w_2)$

Обратный режим автоматического дифференцирования

Мы рассмотрим ту же функцию с вычислительным графом:

$$L(w_1, w_2) = w_2 \log w_1 + \sqrt{w_2 \log w_1}$$

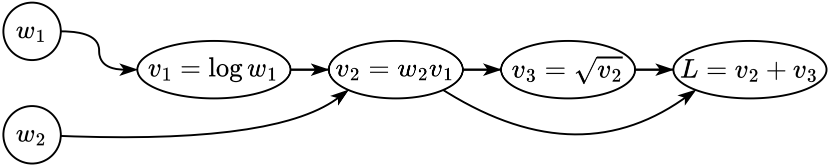


Рис. 17: Иллюстрация вычислительного графа для функции $L(w_1, w_2)$

Предположим, что у нас есть некоторые значения параметров w_1, w_2 и мы уже выполнили прямой проход (т.е. вычисление значений всех промежуточных узлов вычислительного графа). Предположим также, что мы как-то сохранили все промежуточные значения v_i. Давайте пойдем от конца графа к началу и вычислим производные $\dfrac{\partial L}{\partial w_1}, \dfrac{\partial L}{\partial w_2}$:

Пример обратного режима автоматического дифференцирования

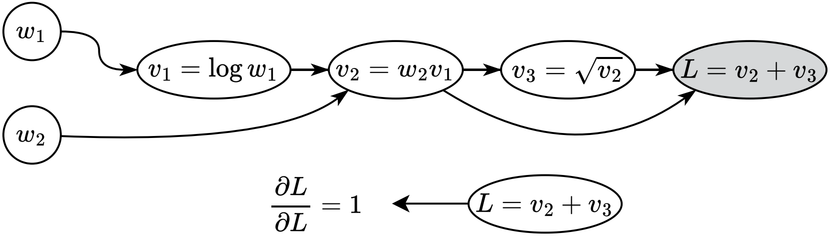


Рис. 18: Иллюстрация обратного режима автоматического дифференцирования

Пример обратного режима автоматического дифференцирования

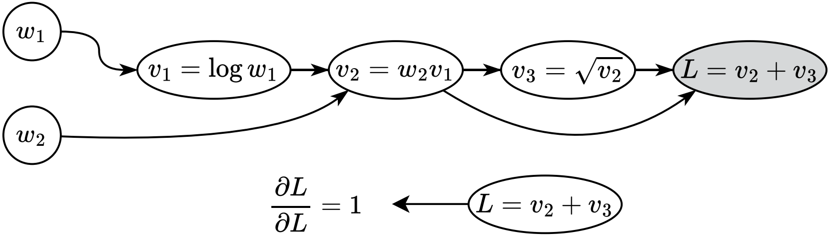


Рис. 18: Иллюстрация обратного режима автоматического дифференцирования

Производные

Пример обратного режима автоматического дифференцирования

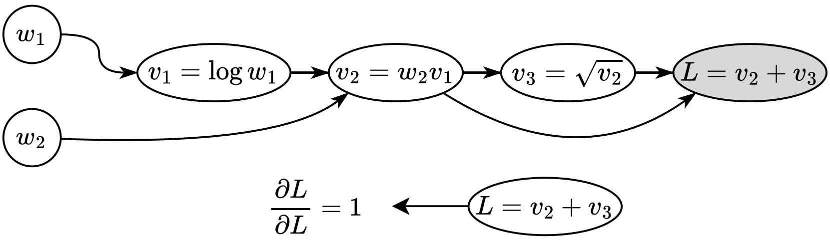


Рис. 18: Иллюстрация обратного режима автоматического дифференцирования

Производные

$$
\frac{\partial L}{\partial L} = 1
$$

Пример обратного режима автоматического дифференцирования

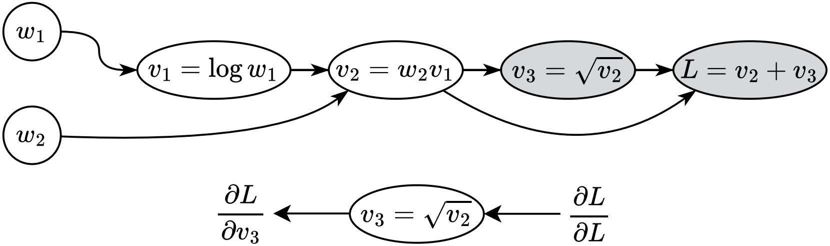


Рис. 19: Иллюстрация обратного режима автоматического дифференцирования

Пример обратного режима автоматического дифференцирования

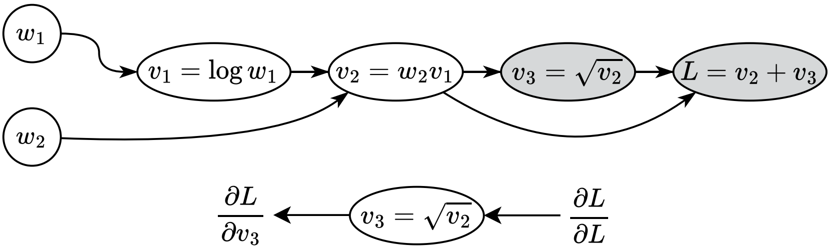
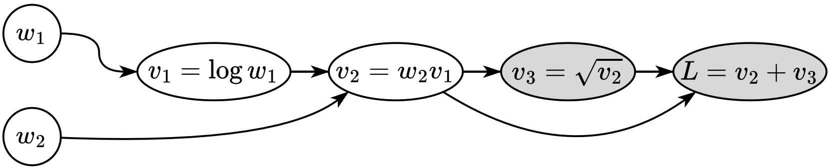


Рис. 19: Иллюстрация обратного режима автоматического дифференцирования

Производные

Пример обратного режима автоматического дифференцирования



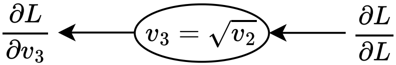


Рис. 19: Иллюстрация обратного режима автоматического дифференцирования

Производные

$$\frac{\partial L}{\partial v_3} = \frac{\partial L}{\partial L}\frac{\partial L}{\partial v_3} \quad = \frac{\partial L}{\partial L} 1$$

Пример обратного режима автоматического дифференцирования

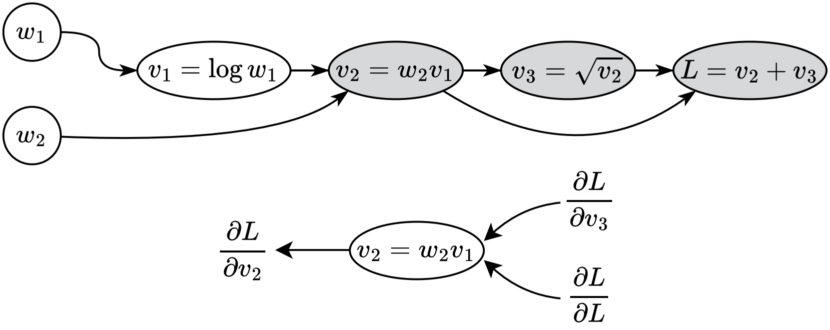


Рис. 20: Иллюстрация обратного режима автоматического дифференцирования

Пример обратного режима автоматического дифференцирования

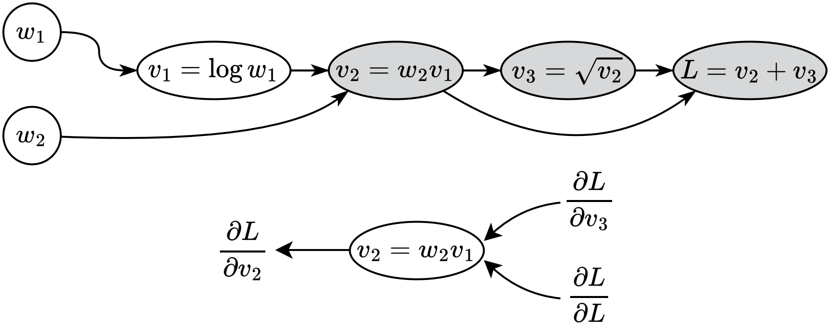


Рис. 20: Иллюстрация обратного режима автоматического дифференцирования

Производные

Пример обратного режима автоматического дифференцирования

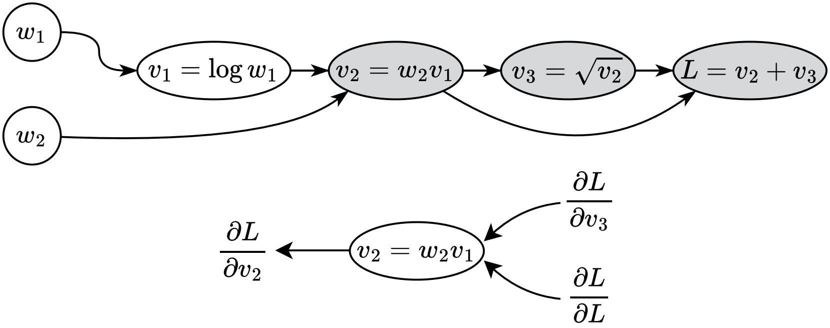


Рис. 20: Иллюстрация обратного режима автоматического дифференцирования

Производные

$$
\frac{\partial L}{\partial v_2} = \frac{\partial L}{\partial v_3}\frac{\partial v_3}{\partial v_2} + \frac{\partial L}{\partial L}\frac{\partial L}{\partial v_2} \quad = \frac{\partial L}{\partial v_3}\frac{1}{2\sqrt{v_2}} + \frac{\partial L}{\partial L}1
$$

Пример обратного режима автоматического дифференцирования

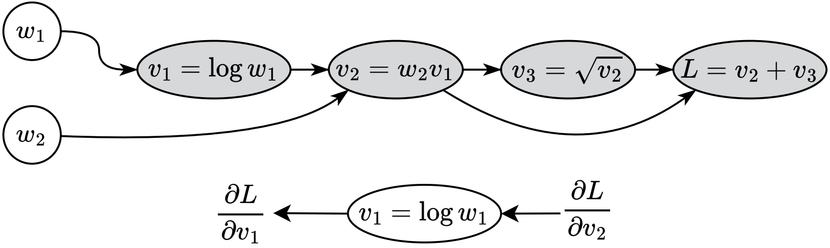


Рис. 21: Иллюстрация обратного режима автоматического дифференцирования

Пример обратного режима автоматического дифференцирования

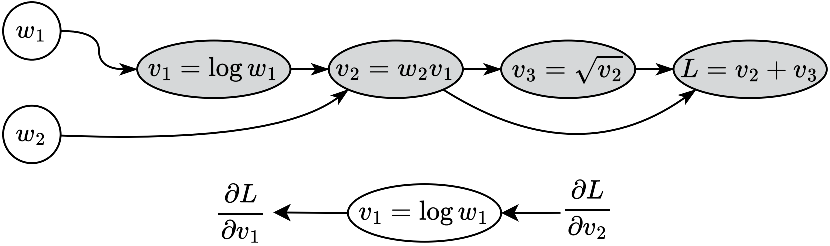


Рис. 21: Иллюстрация обратного режима автоматического дифференцирования

Производные

Пример обратного режима автоматического дифференцирования

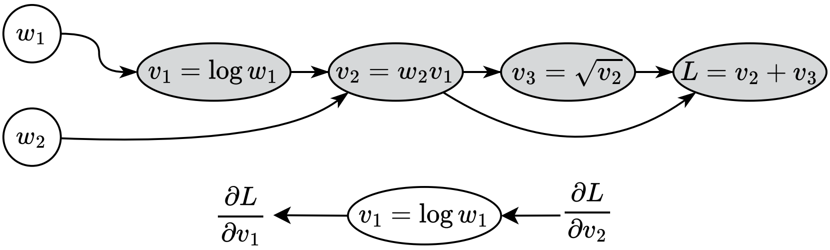


Рис. 21: Иллюстрация обратного режима автоматического дифференцирования

Производные

$$\frac{\partial L}{\partial v_1} = \frac{\partial L}{\partial v_2}\frac{\partial v_2}{\partial v_1} \quad = \frac{\partial L}{\partial v_2} w_2$$

Пример обратного режима автоматического дифференцирования

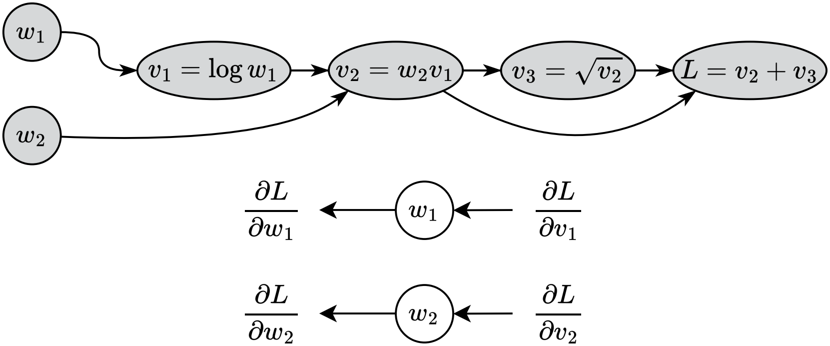


Рис. 22: Иллюстрация обратного режима автоматического дифференцирования

Пример обратного режима автоматического дифференцирования

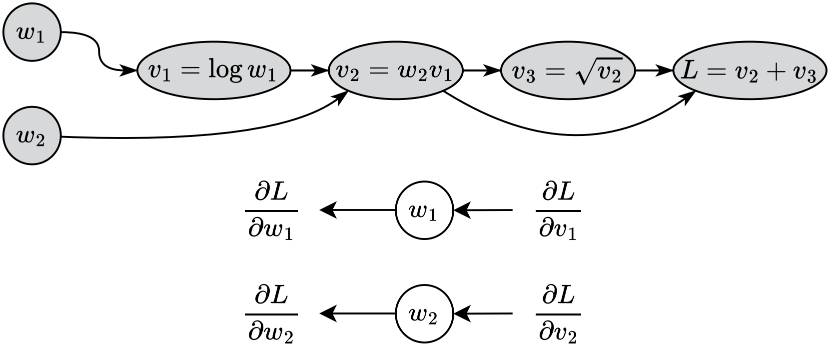


Рис. 22: Иллюстрация обратного режима автоматического дифференцирования

Производные

Пример обратного режима автоматического дифференцирования

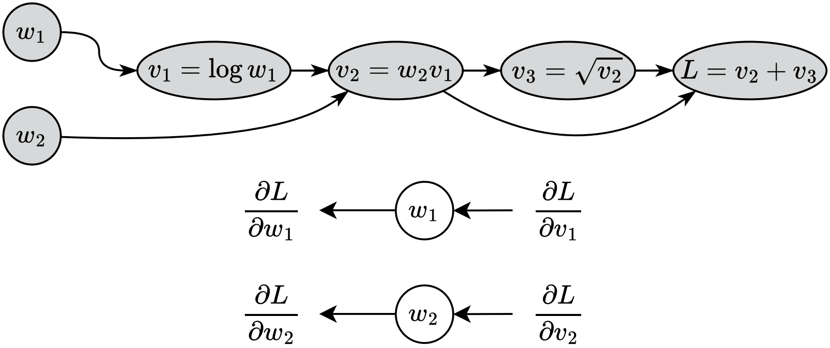


Рис. 22: Иллюстрация обратного режима автоматического дифференцирования

Производные

$$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial v_1}\frac{\partial v_1}{\partial w_1} = \frac{\partial L}{\partial v_1}\frac{1}{w_1} \qquad \frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial v_2}\frac{\partial v_2}{\partial w_2} = \frac{\partial L}{\partial v_1} v_1$$

Обратный режим автоматического дифференцирования

Question

Обратите внимание, что для того же количества вычислений, что и в прямом режиме, мы получаем полный вектор градиента $\nabla_w L$. Какова стоимость ускорения?

Обратный режим автоматического дифференцирования

Question

Обратите внимание, что для того же количества вычислений, что и в прямом режиме, мы получаем полный вектор градиента $\nabla_w L$. Какова стоимость ускорения?
Ответ Обратите внимание, что для использования обратного режима AD вам нужно хранить все промежуточные вычисления из прямого прохода. Эта проблема может быть частично решена с помощью чекпоинтинга, при котором мы сохраняем только часть промежуточных значений, а остальные пересчитываем заново по мере необходимости. Это позволяет значительно уменьшить объём требуемой памяти при обучении больших моделей машинного обучения.

Choose your fighter

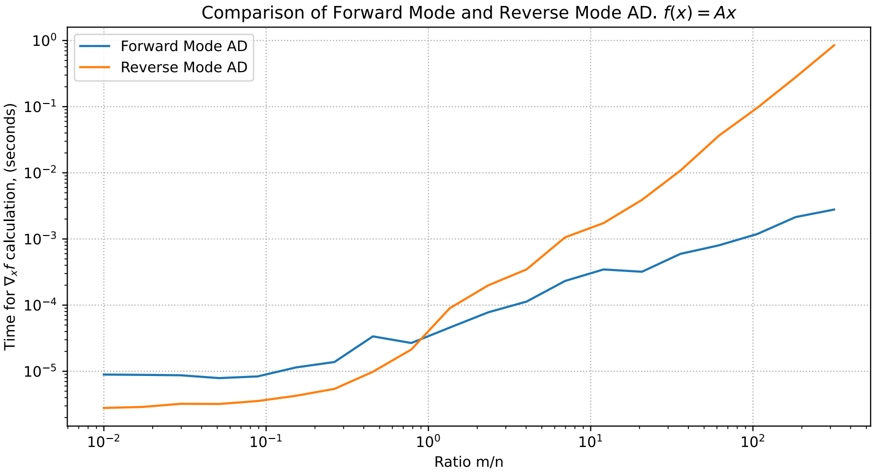


Рис. 23: ♣ График иллюстрирует идею выбора между режимами автоматического дифференцирования. Размерность входа $n = 100$ фиксирована, измерено время вычисления якобиана в зависимости от соотношения размерностей выхода и входа для разных размерностей выхода m.

Чекпоинтинг

Анимация вышеуказанных подходов

Пример использования контрольных точек градиента

[1] ZeRO: Memory Optimizations Toward Training Trillion Parameter Models

Чекпоинтинг

Анимация вышеуказанных подходов

Пример использования контрольных точек градиента

В качестве примера рассмотрим обучение **GPT-2**[1]:

- Активации в простом режиме могут занимать гораздо больше памяти: для последовательности длиной 1K и размера батча 32, 60 GB нужно для хранения всех промежуточных активаций.

[1] ZeRO: Memory Optimizations Toward Training Trillion Parameter Models

Чекпоинтинг

Анимация вышеуказанных подходов

Пример использования контрольных точек градиента

В качестве примера рассмотрим обучение **GPT-2**[1]:

- Активации в простом режиме могут занимать гораздо больше памяти: для последовательности длиной 1K и размера батча 32, 60 GB нужно для хранения всех промежуточных активаций.
- Чекпоинтинг может снизить потребление до 8 GB, пересчитывая их (33% дополнительных вычислений)

[1] ZeRO: Memory Optimizations Toward Training Trillion Parameter Models

Лекция 4. Выпуклость. Выпуклые множества. Выпуклые функции. Сильно выпуклые функции. Условие Поляка - Лоясиевича.

Аффинное множество, конус, выпуклый конус, выпуклое множество, выпуклая комбинация, оболочка

Definition

- **Аффинное множество:** $\forall x_1, x_2 \in S, \ \forall \theta \in \mathbb{R} \Rightarrow \theta x_1 + (1-\theta)x_2 \in S$ (прямая, проходящая через две точки принадлежит A)

Аффинное множество, конус, выпуклый конус, выпуклое множество, выпуклая комбинация, оболочка

Definition

- **Аффинное множество:** $\forall x_1, x_2 \in S,\ \forall \theta \in \mathbb{R} \Rightarrow \theta x_1 + (1-\theta)x_2 \in S$ (прямая, проходящая через две точки принадлежит A)
- **Выпуклое множество:** $\forall x_1, x_2 \in S,\ \forall \theta \in [0,1] \Rightarrow \theta x_1 + (1-\theta)x_2 \in S$ (для любых двух точек из множества отрезок между ними тоже принадлежит этому множеству)

Аффинное множество, конус, выпуклый конус, выпуклое множество, выпуклая комбинация, оболочка

Definition

- **Аффинное множество:** $\forall x_1, x_2 \in S, \ \forall \theta \in \mathbb{R} \Rightarrow \theta x_1 + (1-\theta)x_2 \in S$ (прямая, проходящая через две точки принадлежит A)
- **Выпуклое множество:** $\forall x_1, x_2 \in S, \ \forall \theta \in [0,1] \Rightarrow \theta x_1 + (1-\theta)x_2 \in S$ (для любых двух точек из множества отрезок между ними тоже принадлежит этому множеству)
- **Конус:** $\forall x \in S, \ \forall \alpha \geq 0 \Rightarrow \alpha x \in S$ (луч, проходящий через точку из начала координат, принадлежит множеству)

Аффинное множество, конус, выпуклый конус, выпуклое множество, выпуклая комбинация, оболочка

Definition

- **Аффинное множество:** $\forall x_1, x_2 \in S,\ \forall \theta \in \mathbb{R} \Rightarrow \theta x_1 + (1-\theta)x_2 \in S$ (прямая, проходящая через две точки принадлежит A)
- **Выпуклое множество:** $\forall x_1, x_2 \in S,\ \forall \theta \in [0,1] \Rightarrow \theta x_1 + (1-\theta)x_2 \in S$ (для любых двух точек из множества отрезок между ними тоже принадлежит этому множеству)
- **Конус:** $\forall x \in S,\ \forall \alpha \geq 0 \Rightarrow \alpha x \in S$ (луч, проходящий через точку из начала координат, принадлежит множеству)
- **Выпуклый конус:** $\forall x_1, x_2 \in S,\ \forall \theta_1, \theta_2 \geq 0 \Rightarrow \theta_1 x_1 + \theta_2 x_2 \in S$ (конус, который еще и выпуклый)

Аффинное множество, конус, выпуклый конус, выпуклое множество, выпуклая комбинация, оболочка

Definition

- **Аффинное множество:** $\forall x_1, x_2 \in S,\ \forall \theta \in \mathbb{R} \Rightarrow \theta x_1 + (1-\theta)x_2 \in S$ (прямая, проходящая через две точки принадлежит A)
- **Выпуклое множество:** $\forall x_1, x_2 \in S,\ \forall \theta \in [0,1] \Rightarrow \theta x_1 + (1-\theta)x_2 \in S$ (для любых двух точек из множества отрезок между ними тоже принадлежит этому множеству)
- **Конус:** $\forall x \in S,\ \forall \alpha \geq 0 \Rightarrow \alpha x \in S$ (луч, проходящий через точку из начала координат, принадлежит множеству)
- **Выпуклый конус:** $\forall x_1, x_2 \in S,\ \forall \theta_1, \theta_2 \geq 0 \Rightarrow \theta_1 x_1 + \theta_2 x_2 \in S$ (конус, который еще и выпуклый)
- **Выпуклая комбинация:** $\sum_i \theta_i x_i$, где $\theta_i \geq 0,\ \sum_i \theta_i = 1$.

Аффинное множество, конус, выпуклый конус, выпуклое множество, выпуклая комбинация, оболочка

Definition

- **Аффинное множество:** $\forall x_1, x_2 \in S, \ \forall \theta \in \mathbb{R} \Rightarrow \theta x_1 + (1-\theta)x_2 \in S$ (прямая, проходящая через две точки принадлежит A)
- **Выпуклое множество:** $\forall x_1, x_2 \in S, \ \forall \theta \in [0,1] \Rightarrow \theta x_1 + (1-\theta)x_2 \in S$ (для любых двух точек из множества отрезок между ними тоже принадлежит этому множеству)
- **Конус:** $\forall x \in S, \ \forall \alpha \geq 0 \Rightarrow \alpha x \in S$ (луч, проходящий через точку из начала координат, принадлежит множеству)
- **Выпуклый конус:** $\forall x_1, x_2 \in S, \ \forall \theta_1, \theta_2 \geq 0 \Rightarrow \theta_1 x_1 + \theta_2 x_2 \in S$ (конус, который еще и выпуклый)
- **Выпуклая комбинация:** $\sum_i \theta_i x_i$, где $\theta_i \geq 0, \ \sum_i \theta_i = 1$.
- **Выпуклая оболочка:** $\text{conv}(S) = \{\sum_i \theta_i x_i | x_i \in S, \ \theta_i \geq 0, \ \sum_i \theta_i = 1\}$ (обвыпукливание множества: для любых двух точек из множества включить отрезок между ними).

Проверка выпуклости множества и сохраняющие операции

- **Как проверить выпуклость:**

Проверка выпуклости множества и сохраняющие операции

- **Как проверить выпуклость:**
    - По определению

Проверка выпуклости множества и сохраняющие операции

- **Как проверить выпуклость:**
    - По определению
    - Операции, сохраняющие выпуклость:

Проверка выпуклости множества и сохраняющие операции

- **Как проверить выпуклость:**
    - По определению
    - Операции, сохраняющие выпуклость:
        1. Пусть $S_x,\ S_y$ выпуклы, тогда $S = \left\{ s \mid s = c_1 x + c_2 y,\ x \in S_x,\ y \in S_y,\ c_1, c_2 \in \mathbb{R} \right\}$ выпукло

Проверка выпуклости множества и сохраняющие операции

- **Как проверить выпуклость:**
  - По определению
  - Операции, сохраняющие выпуклость:
    1. Пусть $S_x, \ S_y$ выпуклы, тогда $S = \left\{s \mid s = c_1 x + c_2 y, \ x \in S_x, \ y \in S_y, \ c_1, c_2 \in \mathbb{R}\right\}$ выпукло
    2. Пересечение любого числа выпуклых множеств

Проверка выпуклости множества и сохраняющие операции

- **Как проверить выпуклость:**
  - По определению
  - Операции, сохраняющие выпуклость:
    1. Пусть S_x, S_y выпуклы, тогда $S = \left\{ s \mid s = c_1 x + c_2 y, \ x \in S_x, \ y \in S_y, \ c_1, c_2 \in \mathbb{R} \right\}$ выпукло
    2. Пересечение любого числа выпуклых множеств
    3. $S \subseteq \mathbb{R}^n$ выпукло $\quad \to \quad f(S) = \{f(x) \mid x \in S\}$ выпукло $\quad (f(x) = \mathbf{A}x + \mathbf{b})$

Выпуклые функции, неравенство Йенсена, надграфик

Definition

- **Выпуклая функция:** $\forall x, y \in S \subseteq \mathbb{R}^n$, где S выпукло, $\theta \in [0,1] \Rightarrow f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta) f(y)$.

Неравенство Йенсена: если f выпукла, $\theta_i \geq 0$, $\sum_i \theta_i = 1$, то

$$f\left(\sum_i \theta_i x_i\right) \leq \sum_i \theta_i f(x_i).$$

Definition

Функция называется **строго выпуклой**, если в определении выпуклой функции неравенство строгое. Пример: $f(x) = x^4$.

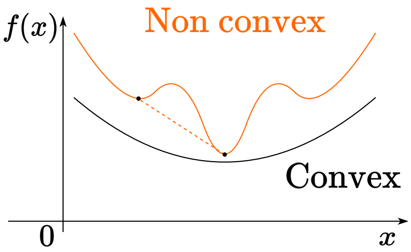


Рис. 24: Разница между выпуклой и невыпуклой функцией

Выпуклые функции, неравенство Йенсена, надграфик

Definition

- **Выпуклая функция:** $\forall x, y \in S \subseteq \mathbb{R}^n$, где S выпукло, $\theta \in [0, 1] \Rightarrow f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta) f(y)$.
- **Надграфик (эпиграф):** $\text{epi}(f) = \{(x, t) : f(x) \leq t\}$; f выпукла $\Leftrightarrow$ $\text{epi}(f)$ выпукл.

Неравенство Йенсена: если f выпукла, $\theta_i \geq 0$, $\sum_i \theta_i = 1$, то

$$
f\left(\sum_i \theta_i x_i\right) \leq \sum_i \theta_i f(x_i).
$$

Definition

Функция называется **строго выпуклой**, если в определении выпуклой функции неравенство строгое. Пример: $f(x) = x^4$.

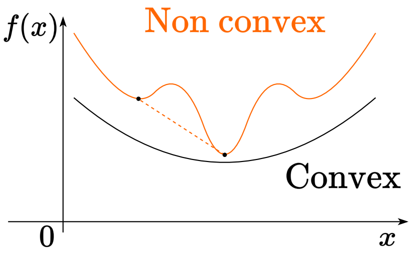


Рис. 24: Разница между выпуклой и невыпуклой функцией

Пример. Функция максимального собственного значения матрицы является выпуклой.

Example

Покажите, что $f(A) = \lambda_{max}(A)$ - выпукла, если $A \in S^n_+$.

Дифференциальные критерии выпуклости (1-й и 2-й порядок)

Definition

- **Критерий 1-го порядка:** Пусть $f : S \to \mathbb{R}$ дифференцируема, где S выпукла. Тогда

$$f(x) \text{ выпукла } \Leftrightarrow f(y) \geq f(x) + \nabla f(x)^\top (y - x) \quad \forall x, y.$$

Дифференциальные критерии выпуклости (1-й и 2-й порядок)

Definition

- **Критерий 1-го порядка:** Пусть $f : S \to \mathbb{R}$ дифференцируема, где S выпукла. Тогда

$$f(x) \text{ выпукла } \Leftrightarrow f(y) \geq f(x) + \nabla f(x)^\top (y - x) \quad \forall x, y.$$

- **Критерий 2-го порядка:** Пусть $f : S \to \mathbb{R}$ дважды дифференцируема, где S выпукло. Тогда

$$f(x) \text{ выпукла } \Leftrightarrow \nabla^2 f(x) \succeq 0 \quad \forall x.$$

Сильная выпуклость

Definition

$f(x)$, **определенная на выпуклом множестве** $S \subseteq \mathbb{R}^n$, называется μ-сильно выпуклой (сильно выпуклой) на S, если:

$$f(\lambda x_1 + (1-\lambda)x_2) \leq \lambda f(x_1) + (1-\lambda)f(x_2) - \frac{\mu}{2}\lambda(1-\lambda)\|x_1 - x_2\|^2$$

для любых $x_1, x_2 \in S$ и $0 \leq \lambda \leq 1$ для некоторого $\mu > 0$.

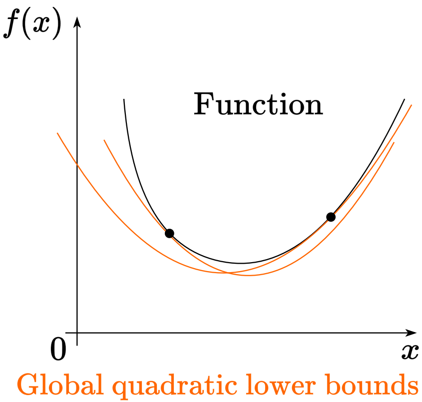


Рис. 25: Сильно выпуклая функция не меньше некоторой параболы в любой точке

Дифференциальные критерии сильной выпуклости (1-й и 2-й порядок)

Definition

- **Критерий 1-го порядка:** Пусть $f : S \to \mathbb{R}$ дифференцируема, где S выпукло. Тогда для некоторого $\mu > 0$

$$f(x) \ \mu\text{-сильно выпукла} \Leftrightarrow f(y) \geq f(x) + \nabla f(x)^\top (y - x) + \frac{\mu}{2}\|x - y\|^2 \quad \forall x, y.$$

Дифференциальные критерии сильной выпуклости (1-й и 2-й порядок)

Definition

- **Критерий 1-го порядка:** Пусть $f : S \to \mathbb{R}$ дифференцируема, где S выпукло. Тогда для некоторого $\mu > 0$
$$f(x) \ \mu\text{-сильно выпукла} \Leftrightarrow f(y) \geq f(x) + \nabla f(x)^\top (y - x) + \frac{\mu}{2} \|x - y\|^2 \quad \forall x, y.$$
- **Критерий 2-го порядка:** Пусть $f : S \to \mathbb{R}$ дважды дифференцируема, где S выпукло. Тогда для некоторого $\mu > 0$
$$f(x) \ \mu\text{-сильно выпукла} \Leftrightarrow \nabla^2 f(x) \succeq \mu I \quad \forall x.$$

Пример. Квадратичная функция.

Example

Покажите, что $f(x) = x^\top A x$, где $A \succeq 0$ - выпукла на $\mathbb{R}^n$. Является ли она сильно выпуклой?

Условие Поляка-Лоясиевича. Линейная сходимость градиентного спуска без выпуклости

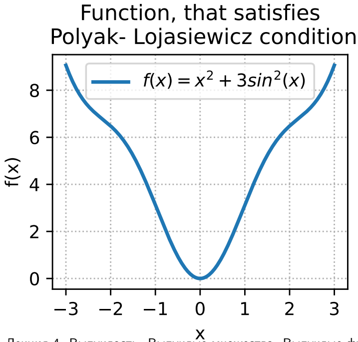
Неравенство PL выполняется, если выполняется следующее условие для некоторого $\mu > 0$,

$$\|\nabla f(x)\|^2 \geq 2\mu(f(x) - f^*) \forall x$$

При выполнении условия PL алгоритм градиентного спуска имеет линейную сходимость.

Следующие функции удовлетворяют условию PL, но не являются выпуклыми. Ссылка на код

$$f(x) = x^2 + 3\sin^2(x)$$



Условие Поляка-Лоясиевича. Линейная сходимость градиентного спуска без выпуклости

Неравенство PL выполняется, если выполняется следующее условие для некоторого $\mu > 0$,

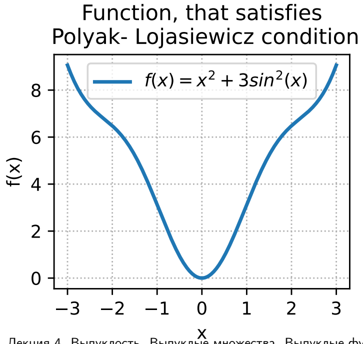
$$\|\nabla f(x)\|^2 \geq 2\mu(f(x) - f^*)\forall x$$

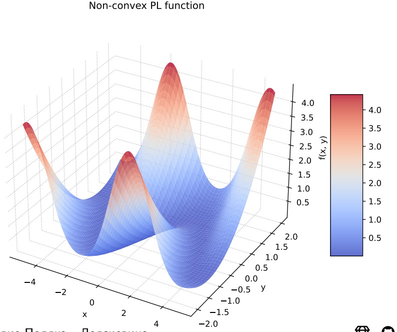
При выполнении условия PL алгоритм градиентного спуска имеет линейную сходимость.

Следующие функции удовлетворяют условию PL, но не являются выпуклыми. Ссылка на код

$$f(x) = x^2 + 3\sin^2(x)$$

$$f(x,y) = \frac{(y - \sin x)^2}{2}$$





Метод наименьших квадратов aka линейная регрессия

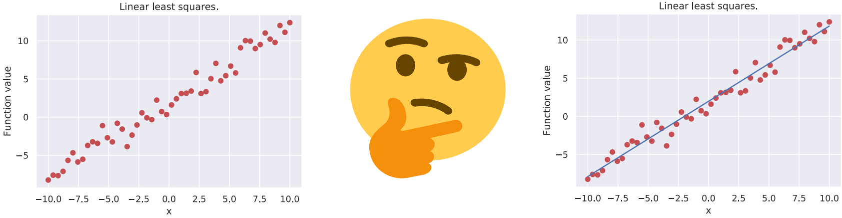


Рис. 28: Иллюстрация

В задаче линейной регрессии у нас есть измерения $X \in \mathbb{R}^{m \times n}$ и $y \in \mathbb{R}^m$ и мы ищем вектор $\theta \in \mathbb{R}^n$ такой, что $X\theta$ близок к y. Близость определяется как сумма квадратов разностей:

$$
\sum_{i=1}^{m} (x_i^\top \theta - y_i)^2 = \|X\theta - y\|_2^2 \to \min_{\theta \in \mathbb{R}^n}
$$

Например, рассмотрим набор данных, содержащий m пользователей, каждый из которых представлен n признаками. Каждая строка $x_i^\top$ матрицы признаков X соответствует признакам пользователя i, а соответствующий элемент y_i вектора откликов y представляет собой измеряемую величину, которую мы хотим предсказать на основе $x_i^\top$, например, расходы на рекламу. Предсказание значения осуществляется по формуле $x_i^\top \theta$.

Метод наименьших квадратов aka линейная регрессия [2]

1. Является ли эта задача выпуклой? Сильно выпуклой?

[2] Посмотрите на пример реальных данных линейной задачи метода наименьших квадратов

l_2-регуляризованный метод наименьших квадратов

Сделать задачу сильно-выпуклой, а не (строго-)выпуклой, можно, добавив l_2-штраф, также известный как регуляризация Тихонова, l_2-регуляризация или демпфирование весов.

$$
\|X\theta - y\|_2^2 + \frac{\mu}{2}\|\theta\|_2^2 \to \min_{\theta \in \mathbb{R}^n}
$$

Примечание: С этой модификацией целевая функция становится μ-сильно выпуклой.

Посмотрите на код

Наиболее важная разница между выпуклостью и сильной выпуклостью

$$
f(x) = \frac{1}{2m}\|Ax - b\|_2^2 + \frac{\mu}{2}\|x\|_2^2 \to \min_{x \in \mathbb{R}^n}, \quad A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m
$$

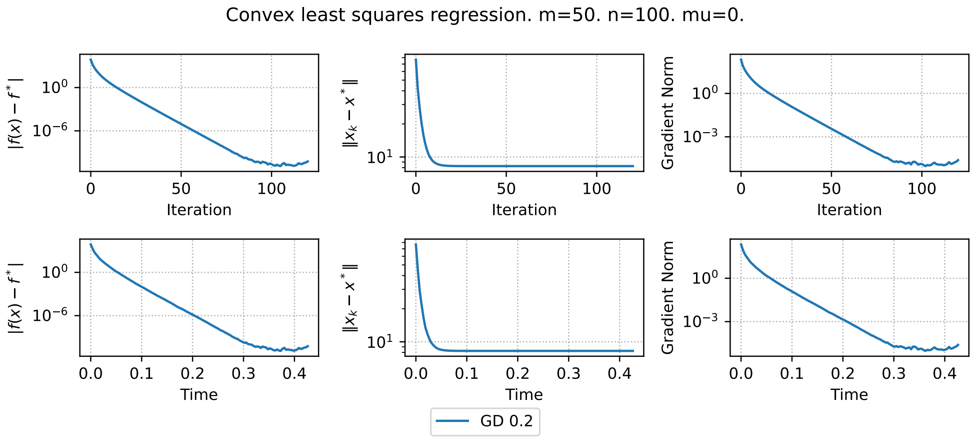


Рис. 29: Выпуклая задача не имеет сходимости по аргументу

Наиболее важная разница между выпуклостью и сильной выпуклостью

$$f(x) = \frac{1}{2m}\|Ax - b\|_2^2 + \frac{\mu}{2}\|x\|_2^2 \to \min_{x \in \mathbb{R}^n}, \quad A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m$$

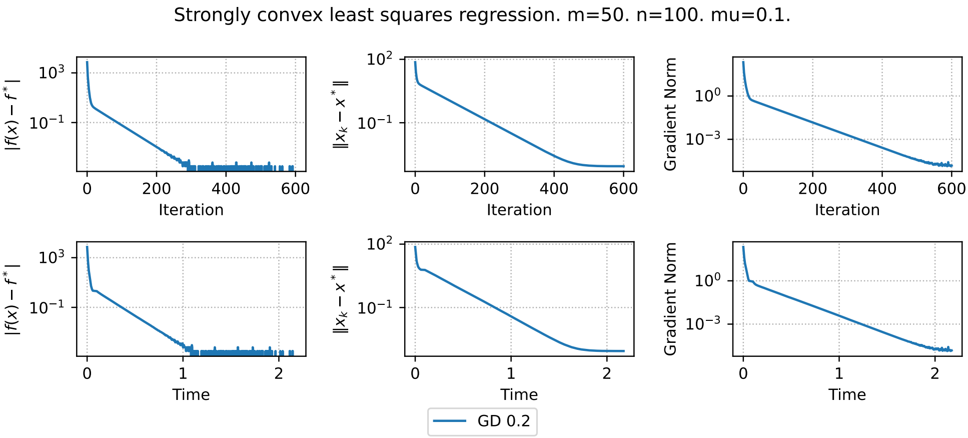


Рис. 30: Но если добавить даже небольшую регуляризацию, вы гарантируете сходимость по аргументу

Наиболее важная разница между выпуклостью и сильной выпуклостью

$$
f(x) = \frac{1}{2m}\|Ax - b\|_2^2 + \frac{\mu}{2}\|x\|_2^2 \to \min_{x \in \mathbb{R}^n}, \quad A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m
$$

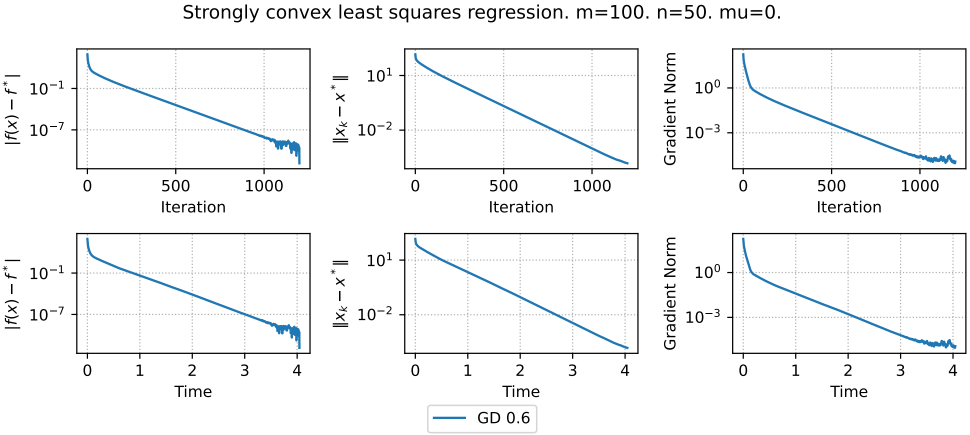


Рис. 31: Другой способ обеспечить сходимость в предыдущей задаче - поменять местами значения размерности задачи

Для сходимости к решению с высокой точностью необходима сильная выпуклость (или выполнение условия Поляка-Лоясиевича).

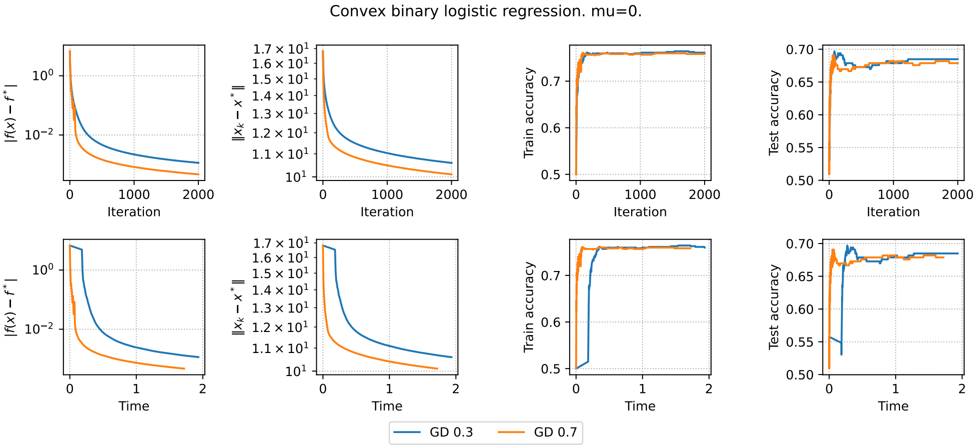


Рис. 32: Лишь небольшая точность может быть достигнута с сублинейной сходимостью

Для сходимости к решению с высокой точностью необходима сильная выпуклость (или выполнение условия Поляка-Лоясиевича).

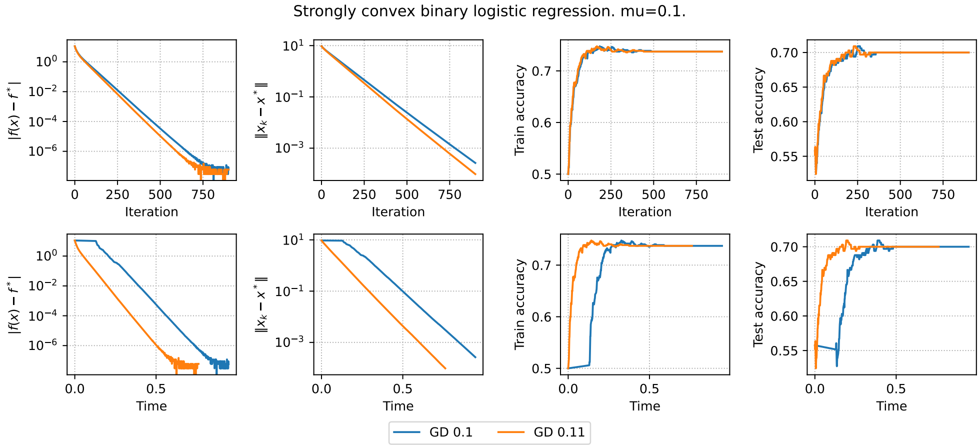


Рис. 33: Сильная выпуклость обеспечивает линейную сходимость

Лекция 5. Условия оптимальности. Функция Лагранжа. Задачи с ограничениями. ККТ.

Постановка задачи. Бюджетное множество, критические точки.

$$f(x) \to \min_{x \in S}$$

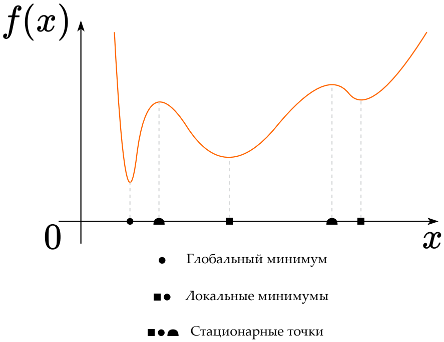


Рис. 34: Иллюстрация различных стационарных (критических) точек

Постановка задачи. Бюджетное множество, критические точки.

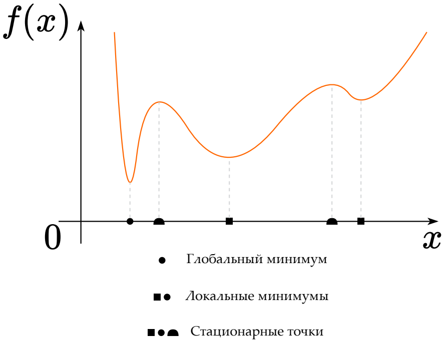


Рис. 34: Иллюстрация различных стационарных (критических) точек

$$f(x) \to \min_{x \in S}$$

Множество S обычно называется **допустимым множеством** (или **бюджетным множеством**).

Постановка задачи. Бюджетное множество, критические точки.

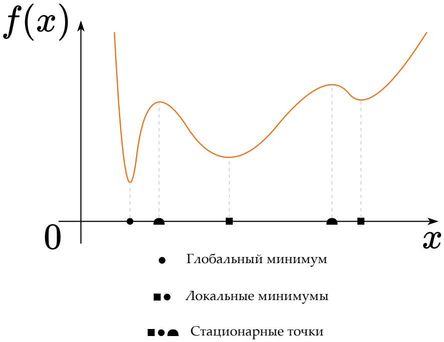


Рис. 34: Иллюстрация различных стационарных (критических) точек

$$f(x) \to \min_{x \in S}$$

Множество S обычно называется **допустимым множеством** (или **бюджетным множеством**).

Мы говорим, что задача имеет решение, если бюджетное множество, в котором достигается минимум или инфимум данной функции, **не пусто**: $x^* \in S$.

Постановка задачи. Бюджетное множество, критические точки.

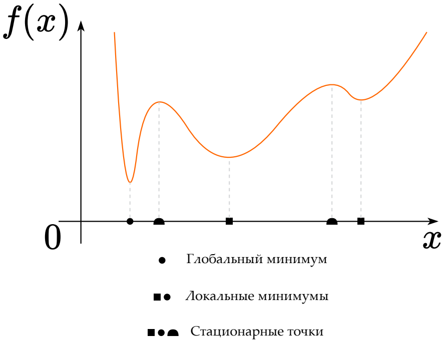


Рис. 34: Иллюстрация различных стационарных (критических) точек

$$f(x) \to \min_{x \in S}$$

Множество S обычно называется **допустимым множеством** (или **бюджетным множеством**).

Мы говорим, что задача имеет решение, если бюджетное множество, в котором достигается минимум или инфимум данной функции, **не пусто**: $x^* \in S$.

- Точка x^* является **глобальным минимумом**, если $f(x^*) \leq f(x)$ для всех $x \in S$.

Постановка задачи. Бюджетное множество, критические точки.

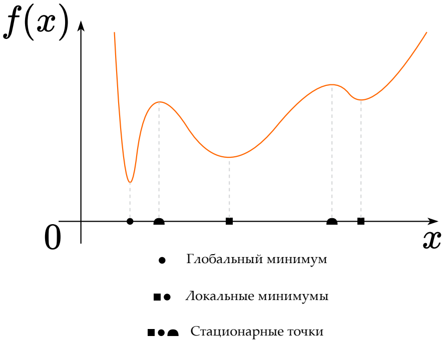


Рис. 34: Иллюстрация различных стационарных (критических) точек

$$f(x) \to \min_{x \in S}$$

Множество S обычно называется **допустимым множеством** (или **бюджетным множеством**).

Мы говорим, что задача имеет решение, если бюджетное множество, в котором достигается минимум или инфимум данной функции, **не пусто**: $x^* \in S$.

- Точка x^* является **глобальным минимумом**, если $f(x^*) \leq f(x)$ для всех $x \in S$.
- Точка x^* является **локальным минимумом**, если существует окрестность N точки x^* такая, что $f(x^*) \leq f(x)$ для всех $x \in N \cap S$.

Постановка задачи. Бюджетное множество, критические точки.

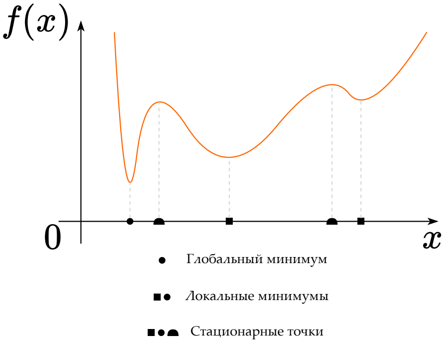


Рис. 34: Иллюстрация различных стационарных (критических) точек

$$f(x) \to \min_{x \in S}$$

Множество S обычно называется **допустимым множеством** (или **бюджетным множеством**).

Мы говорим, что задача имеет решение, если бюджетное множество, в котором достигается минимум или инфимум данной функции, **не пусто**: $x^* \in S$.

- Точка x^* является **глобальным минимумом**, если $f(x^*) \leq f(x)$ для всех $x \in S$.
- Точка x^* является **локальным минимумом**, если существует окрестность N точки x^* такая, что $f(x^*) \leq f(x)$ для всех $x \in N \cap S$.
- Точка x^* является **строгим локальным минимумом**, если существует окрестность N точки x^* такая, что $f(x^*) < f(x)$ для всех $x \in N \cap S$ с $x \neq x^*$.

Постановка задачи. Бюджетное множество, критические точки.

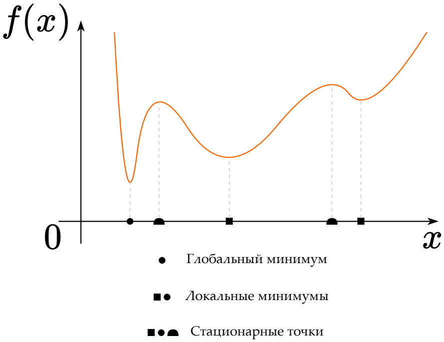


Рис. 34: Иллюстрация различных стационарных (критических) точек

$$f(x) \to \min_{x \in S}$$

Множество S обычно называется **допустимым множеством** (или **бюджетным множеством**).

Мы говорим, что задача имеет решение, если бюджетное множество, в котором достигается минимум или инфимум данной функции, **не пусто**: $x^* \in S$.

- Точка x^* является **глобальным минимумом**, если $f(x^*) \leq f(x)$ для всех $x \in S$.
- Точка x^* является **локальным минимумом**, если существует окрестность N точки x^* такая, что $f(x^*) \leq f(x)$ для всех $x \in N \cap S$.
- Точка x^* является **строгим локальным минимумом**, если существует окрестность N точки x^* такая, что $f(x^*) < f(x)$ для всех $x \in N \cap S$ с $x \neq x^*$.
- Мы называем точку x^* **стационарной точкой** (или критической точкой), если $\nabla f(x^*) = 0$. Любой локальный минимум дифференцируемой функции должен быть стационарной точкой.

Безусловная оптимизация. Необходимые и достаточные условия экстремума.

- **Необходимое условие оптимальности (I)**. x^* - **локальный** минимум $f(x)$ и f дифференцируема в некоторой окрестности $x^* \Rightarrow \nabla f(x^*) = 0$

Заметим, что если $\nabla f(x^*) = 0$, $\nabla^2 f(x^*) \succeq 0$ (гессиан положительно полуопределён), то мы не можем быть уверены, что x^* является локальным минимумом.

Безусловная оптимизация. Необходимые и достаточные условия экстремума.

- **Необходимое условие оптимальности (I)**. x^* - **локальный** минимум $f(x)$ и f дифференцируема в некоторой окрестности $x^* \Rightarrow \nabla f(x^*) = 0$
- **Достаточные условия оптимальности (II)**. Пусть $\nabla^2 f$ непрерывна в открытой окрестности x^*, и выполнено $\nabla f(x^*) = 0, \quad \nabla^2 f(x^*) \succ 0 \Rightarrow x^*$ - **строгий локальный** минимум $f(x)$.

Заметим, что если $\nabla f(x^*) = 0$, $\nabla^2 f(x^*) \succeq 0$ (гессиан положительно полуопределён), то мы не можем быть уверены, что x^* является локальным минимумом.

Безусловная оптимизация. Необходимые и достаточные условия экстремума.

- **Необходимое условие оптимальности (I)**. x^* - **локальный** минимум $f(x)$ и f дифференцируема в некоторой окрестности $x^* \Rightarrow \nabla f(x^*) = 0$
- **Достаточные условия оптимальности (II)**. Пусть $\nabla^2 f$ непрерывна в открытой окрестности x^*, и выполнено $\nabla f(x^*) = 0, \quad \nabla^2 f(x^*) \succ 0 \Rightarrow x^*$ - **строгий локальный** минимум $f(x)$.
- Для выпуклых функций необходимое условие становится достаточным.

Заметим, что если $\nabla f(x^*) = 0$, $\nabla^2 f(x^*) \succeq 0$ (гессиан положительно полуопределён), то мы не можем быть уверены, что x^* является локальным минимумом.

Безусловная оптимизация. Необходимые и достаточные условия экстремума.

- **Необходимое условие оптимальности (I)**. x^* - **локальный** минимум $f(x)$ и f дифференцируема в некоторой окрестности $x^* \Rightarrow \nabla f(x^*) = 0$
- **Достаточные условия оптимальности (II)**. Пусть $\nabla^2 f$ непрерывна в открытой окрестности x^*, и выполнено $\nabla f(x^*) = 0, \quad \nabla^2 f(x^*) \succ 0 \Rightarrow x^*$ - **строгий локальный** минимум $f(x)$.
- Для выпуклых функций необходимое условие становится достаточным.

Заметим, что если $\nabla f(x^*) = 0$, $\nabla^2 f(x^*) \succeq 0$ (гессиан положительно полуопределён), то мы не можем быть уверены, что x^* является локальным минимумом.

Безусловная оптимизация. Необходимые и достаточные условия экстремума.

- **Необходимое условие оптимальности (I)**. x^* - **локальный** минимум $f(x)$ и f дифференцируема в некоторой окрестности $x^* \Rightarrow \nabla f(x^*) = 0$
- **Достаточные условия оптимальности (II)**. Пусть $\nabla^2 f$ непрерывна в открытой окрестности x^*, и выполнено $\nabla f(x^*) = 0, \quad \nabla^2 f(x^*) \succ 0 \Rightarrow x^*$ - **строгий локальный** минимум $f(x)$.
- Для выпуклых функций необходимое условие становится достаточным.

Заметим, что если $\nabla f(x^*) = 0$, $\nabla^2 f(x^*) \succeq 0$ (гессиан положительно полуопределён), то мы не можем быть уверены, что x^* является локальным минимумом.

$$
f(x, y) = (2x^2 - y)(x^2 - y)
$$

Безусловная оптимизация. Необходимые и достаточные условия экстремума.

- **Необходимое условие оптимальности (I)**. x^* - **локальный** минимум $f(x)$ и f дифференцируема в некоторой окрестности $x^* \Rightarrow \nabla f(x^*) = 0$
- **Достаточные условия оптимальности (II)**. Пусть $\nabla^2 f$ непрерывна в открытой окрестности x^*, и выполнено $\nabla f(x^*) = 0, \quad \nabla^2 f(x^*) \succ 0 \Rightarrow x^*$ - **строгий локальный** минимум $f(x)$.
- Для выпуклых функций необходимое условие становится достаточным.

Заметим, что если $\nabla f(x^*) = 0$, $\nabla^2 f(x^*) \succeq 0$ (гессиан положительно полуопределён), то мы не можем быть уверены, что x^* является локальным минимумом.

$$
f(x, y) = (2x^2 - y)(x^2 - y)
$$

Если точка начинает движение в начале координат $(0, 0)$ вдоль любой прямой линии, то значение $(2x^2 - y)(x^2 - y)$ будет увеличиваться в начале движения. Тем не менее, $(0, 0)$ не является локальным минимумом функции, потому что движение вдоль параболы, такой как $y = \sqrt{2}x^2$, приведет к уменьшению значения функции.

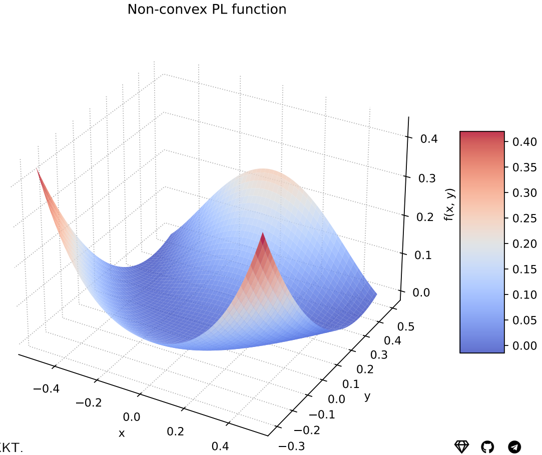
Безусловная оптимизация. Необходимые и достаточные условия экстремума.

- **Необходимое условие оптимальности (I)**. x^* - **локальный** минимум $f(x)$ и f дифференцируема в некоторой окрестности $x^* \Rightarrow \nabla f(x^*) = 0$
- **Достаточные условия оптимальности (II)**. Пусть $\nabla^2 f$ непрерывна в открытой окрестности x^*, и выполнено $\nabla f(x^*) = 0, \quad \nabla^2 f(x^*) \succ 0 \Rightarrow x^*$ - **строгий локальный** минимум $f(x)$.
- Для выпуклых функций необходимое условие становится достаточным.

Заметим, что если $\nabla f(x^*) = 0$, $\nabla^2 f(x^*) \succeq 0$ (гессиан положительно полуопределён), то мы не можем быть уверены, что x^* является локальным минимумом.

$$f(x,y) = (2x^2 - y)(x^2 - y)$$

Если точка начинает движение в начале координат $(0,0)$ вдоль любой прямой линии, то значение $(2x^2 - y)(x^2 - y)$ будет увеличиваться в начале движения. Тем не менее, $(0,0)$ не является локальным минимумом функции, потому что движение вдоль параболы, такой как $y = \sqrt{2}x^2$, приведет к уменьшению значения функции.



Условная оптимизация. Идея необходимого условия экстремума.

Вектор $d \in \mathbb{R}^n$ является допустимым направлением в точке $x^* \in S \subseteq \mathbb{R}^n$, если малые шаги вдоль d не выводят нас за пределы S.

Условная оптимизация. Идея необходимого условия экстремума.

Вектор $d \in \mathbb{R}^n$ является допустимым направлением в точке $x^* \in S \subseteq \mathbb{R}^n$, если малые шаги вдоль d не выводят нас за пределы S.

Пусть $S \subseteq \mathbb{R}^n$ и функция $f : \mathbb{R}^n \to \mathbb{R}$. Предположим, что $x^* \in S$ является точкой локального минимума для f над S, и предположим далее, что f непрерывно дифференцируема в окрестности x^*.

Условная оптимизация. Идея необходимого условия экстремума.

Вектор $d \in \mathbb{R}^n$ является допустимым направлением в точке $x^* \in S \subseteq \mathbb{R}^n$, если малые шаги вдоль d не выводят нас за пределы S.

Пусть $S \subseteq \mathbb{R}^n$ и функция $f : \mathbb{R}^n \to \mathbb{R}$. Предположим, что $x^* \in S$ является точкой локального минимума для f над S, и предположим далее, что f непрерывно дифференцируема в окрестности x^*.

1. Тогда для любого допустимого направления $d \in \mathbb{R}^n$ в x^* выполняется $\nabla f(x^*)^\top d \geq 0$.

Условная оптимизация. Идея необходимого условия экстремума.

Вектор $d \in \mathbb{R}^n$ является допустимым направлением в точке $x^* \in S \subseteq \mathbb{R}^n$, если малые шаги вдоль d не выводят нас за пределы S.

Пусть $S \subseteq \mathbb{R}^n$ и функция $f : \mathbb{R}^n \to \mathbb{R}$. Предположим, что $x^* \in S$ является точкой локального минимума для f над S, и предположим далее, что f непрерывно дифференцируема в окрестности x^*.

1. Тогда для любого допустимого направления $d \in \mathbb{R}^n$ в x^* выполняется $\nabla f(x^*)^\top d \geq 0$.
2. Если, кроме того, S выпукло, то

$$\nabla f(x^*)^\top (x - x^*) \geq 0, \forall x \in S.$$

Условная оптимизация. Идея необходимого условия экстремума.

Вектор $d \in \mathbb{R}^n$ является допустимым направлением в точке $x^* \in S \subseteq \mathbb{R}^n$, если малые шаги вдоль d не выводят нас за пределы S.

Пусть $S \subseteq \mathbb{R}^n$ и функция $f : \mathbb{R}^n \to \mathbb{R}$. Предположим, что $x^* \in S$ является точкой локального минимума для f над S, и предположим далее, что f непрерывно дифференцируема в окрестности x^*.

1. Тогда для любого допустимого направления $d \in \mathbb{R}^n$ в x^* выполняется $\nabla f(x^*)^\top d \geq 0$.
2. Если, кроме того, S выпукло, то

$$\nabla f(x^*)^\top (x - x^*) \geq 0, \forall x \in S.$$

Условная оптимизация. Идея необходимого условия экстремума.

Вектор $d \in \mathbb{R}^n$ является допустимым направлением в точке $x^* \in S \subseteq \mathbb{R}^n$, если малые шаги вдоль d не выводят нас за пределы S.

Пусть $S \subseteq \mathbb{R}^n$ и функция $f : \mathbb{R}^n \to \mathbb{R}$. Предположим, что $x^* \in S$ является точкой локального минимума для f над S, и предположим далее, что f непрерывно дифференцируема в окрестности x^*.

1. Тогда для любого допустимого направления $d \in \mathbb{R}^n$ в x^* выполняется $\nabla f(x^*)^\top d \geq 0$.
2. Если, кроме того, S выпукло, то

$$\nabla f(x^*)^\top (x - x^*) \geq 0, \forall x \in S.$$

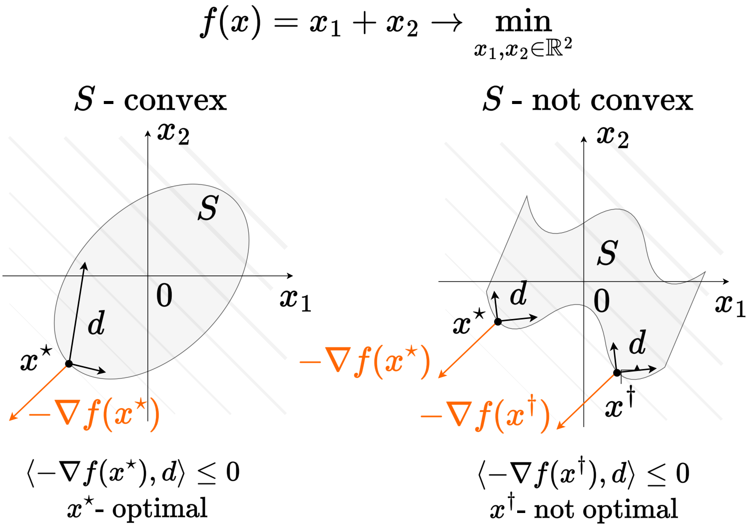


Рис. 35: Общее условие локальной оптимальности первого порядка

Важные свойства выпуклого случая

Следует отметить, что в **выпуклом** случае (то есть при **выпуклых** f **и** S) необходимое условие становится достаточным.

Важные свойства выпуклого случая

Следует отметить, что в **выпуклом** случае (то есть при **выпуклых** f **и** S) необходимое условие становится достаточным.

Еще один важный результат для выпуклого случая звучит следующим образом: если $f(x) : S \to \mathbb{R}$ — выпуклая функция, определённая на выпуклом множестве S, то:

Важные свойства выпуклого случая

Следует отметить, что в **выпуклом** случае (то есть при **выпуклых** f **и** S) необходимое условие становится достаточным.

Еще один важный результат для выпуклого случая звучит следующим образом: если $f(x) : S \to \mathbb{R}$ — выпуклая функция, определённая на выпуклом множестве S, то:

- Любой **локальный минимум** является **глобальным**.

Важные свойства выпуклого случая

Следует отметить, что в **выпуклом** случае (то есть при **выпуклых** f **и** S) необходимое условие становится достаточным.

Еще один важный результат для выпуклого случая звучит следующим образом: если $f(x) : S \to \mathbb{R}$ — выпуклая функция, определённая на выпуклом множестве S, то:

- Любой **локальный минимум** является **глобальным**.
- Множество локальных (= глобальных) минимумов S^* **выпукло**.

Важные свойства выпуклого случая

Следует отметить, что в **выпуклом** случае (то есть при **выпуклых** f **и** S) необходимое условие становится достаточным.

Еще один важный результат для выпуклого случая звучит следующим образом: если $f(x) : S \to \mathbb{R}$ — выпуклая функция, определённая на выпуклом множестве S, то:

- Любой **локальный минимум** является **глобальным**.
- Множество локальных (= глобальных) минимумов S^* **выпукло**.
- Если $f(x)$ — **строго или сильно выпуклая функция**, то S^* содержит только **одну точку**: $S^* = \{x^*\}$.

Задачи с ограничениями-равенствами

$$
\begin{split}
& f(x) \to \min_{x \in \mathbb{R}^n} \\
\text{s.t. } & h(x) = 0
\end{split}
$$

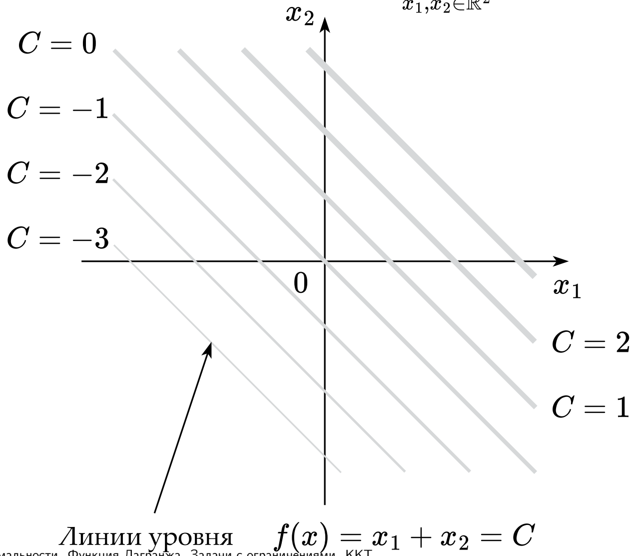
Задачи с ограничениями-равенствами

$$
\begin{split}
& f(x) \to \min_{x \in \mathbb{R}^n} \\
\text{s.t. } & h(x) = 0
\end{split}
$$

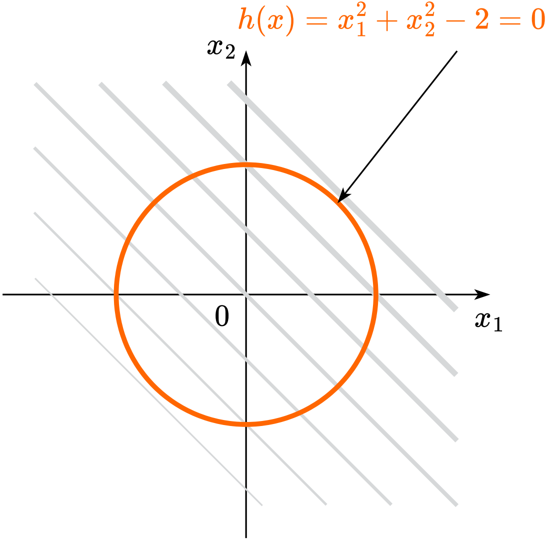
Мы попробуем проиллюстрировать подход к решению этой задачи через простой пример с $f(x) = x_1 + x_2$ и $h(x) = x_1^2 + x_2^2 - 2$.

Задачи с ограничениями-равенствами

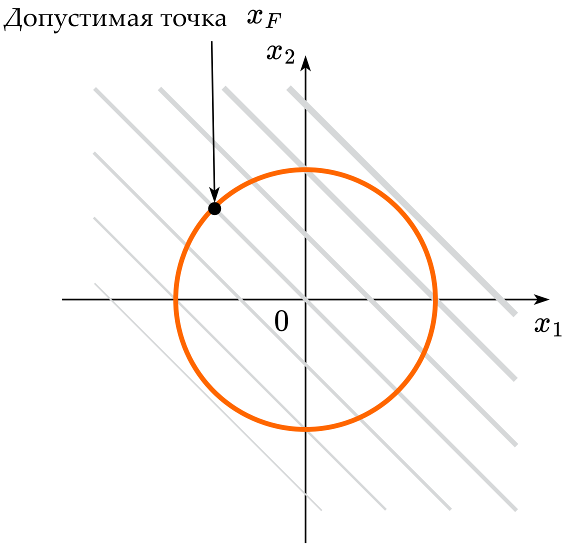
$$f(x) = x_1 + x_2 \to \min_{x_1, x_2 \in \mathbb{R}^2}$$



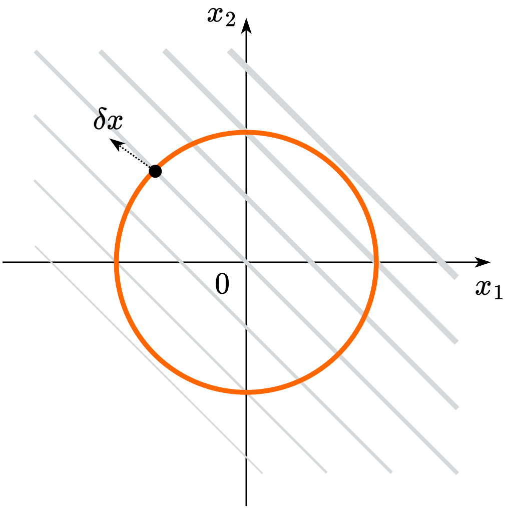
Задачи с ограничениями-равенствами



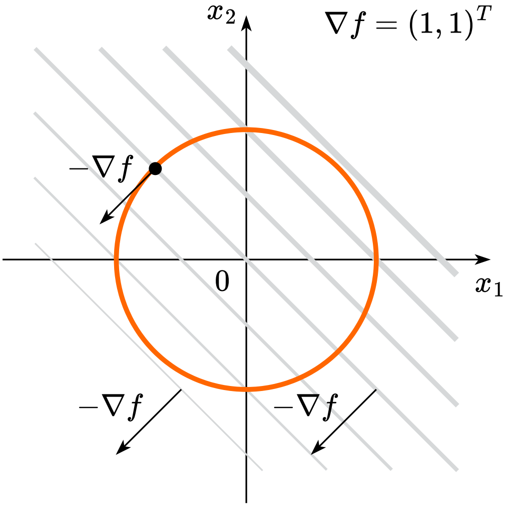
Задачи с ограничениями-равенствами



Задачи с ограничениями-равенствами

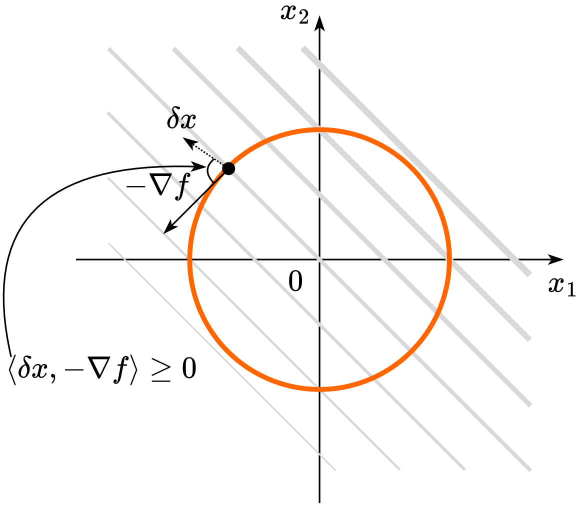


Задачи с ограничениями-равенствами



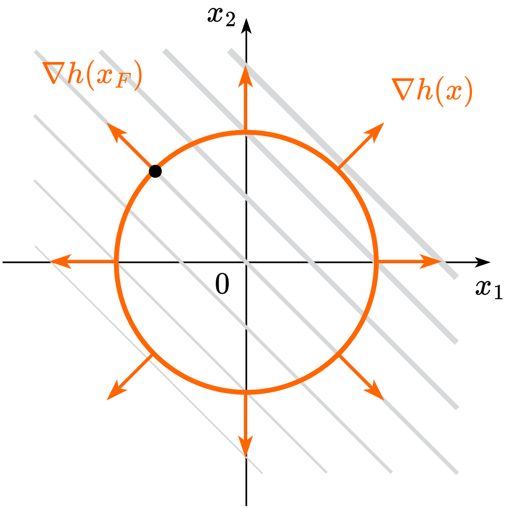
Задачи с ограничениями-равенствами

Мы хотим: $f(x_F + \delta x) \leq f(x_F)$

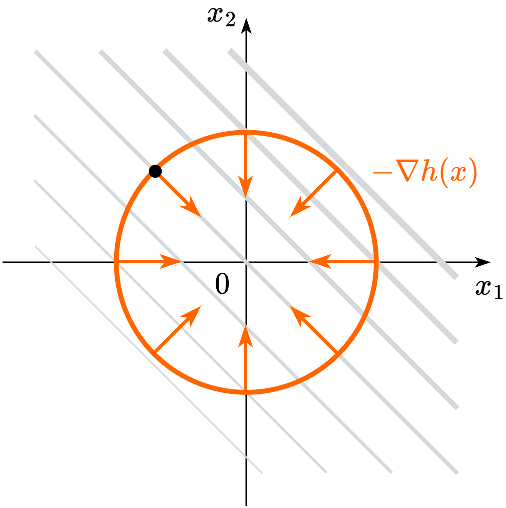


Задачи с ограничениями-равенствами

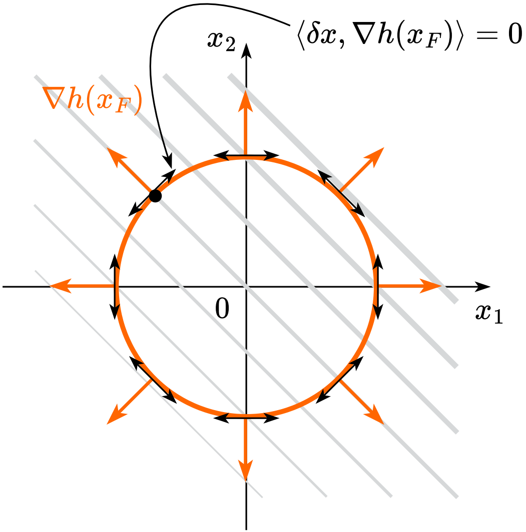
$$\nabla h = (2x_1, 2x_2)^T$$



Задачи с ограничениями-равенствами



Задачи с ограничениями-равенствами



Задачи с ограничениями-равенствами

В общем случае, чтобы двигаться от x_F вдоль допустимого множества и уменьшать значение функции, необходимо обеспечить два условия:

Задачи с ограничениями-равенствами

В общем случае, чтобы двигаться от x_F вдоль допустимого множества и уменьшать значение функции, необходимо обеспечить два условия:

$$\langle \delta x, \nabla h(x_F) \rangle = 0$$

Задачи с ограничениями-равенствами

В общем случае, чтобы двигаться от x_F вдоль допустимого множества и уменьшать значение функции, необходимо обеспечить два условия:

$$\langle \delta x, \nabla h(x_F) \rangle = 0$$

$$\langle \delta x, -\nabla f(x_F) \rangle > 0$$

Задачи с ограничениями-равенствами

В общем случае, чтобы двигаться от x_F вдоль допустимого множества и уменьшать значение функции, необходимо обеспечить два условия:

$$\langle \delta x, \nabla h(x_F) \rangle = 0$$

$$\langle \delta x, -\nabla f(x_F) \rangle > 0$$

Предположим, что в процессе такого движения мы пришли в точку, где

Задачи с ограничениями-равенствами

В общем случае, чтобы двигаться от x_F вдоль допустимого множества и уменьшать значение функции, необходимо обеспечить два условия:

$$\langle \delta x, \nabla h(x_F) \rangle = 0$$

$$\langle \delta x, -\nabla f(x_F) \rangle > 0$$

Предположим, что в процессе такого движения мы пришли в точку, где

$$-\nabla f(x) = \nu \nabla h(x)$$

Задачи с ограничениями-равенствами

В общем случае, чтобы двигаться от x_F вдоль допустимого множества и уменьшать значение функции, необходимо обеспечить два условия:

$$\langle \delta x, \nabla h(x_F) \rangle = 0$$

$$\langle \delta x, -\nabla f(x_F) \rangle > 0$$

Предположим, что в процессе такого движения мы пришли в точку, где

$$-\nabla f(x) = \nu \nabla h(x)$$

$$\langle \delta x, -\nabla f(x) \rangle = \langle \delta x, \nu \nabla h(x) \rangle = 0$$

Задачи с ограничениями-равенствами

В общем случае, чтобы двигаться от x_F вдоль допустимого множества и уменьшать значение функции, необходимо обеспечить два условия:

$$\langle \delta x, \nabla h(x_F) \rangle = 0$$

$$\langle \delta x, -\nabla f(x_F) \rangle > 0$$

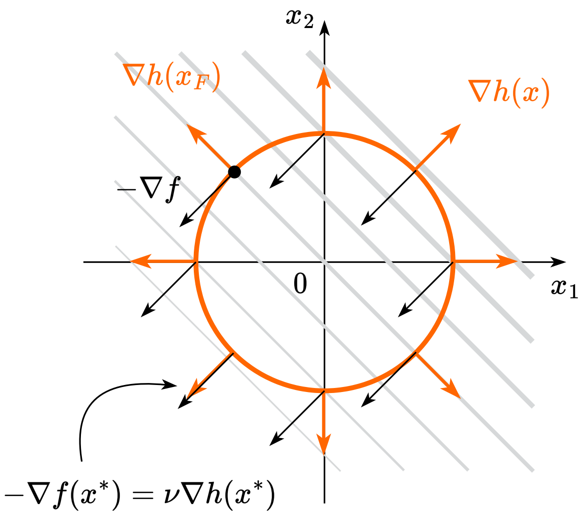
Предположим, что в процессе такого движения мы пришли в точку, где

$$-\nabla f(x) = \nu \nabla h(x)$$

$$\langle \delta x, -\nabla f(x) \rangle = \langle \delta x, \nu \nabla h(x) \rangle = 0$$

Тогда мы достигли такой точки допустимого множества, из которой нельзя уменьшить значение функции при допустимых малых сдвигах. Это и есть условие локального минимума в задаче с ограничением.

Задачи с ограничениями-равенствами



Лагранжиан и его связь с необходимыми условиями

$$
\begin{split}
& f(x) \to \min_{x \in \mathbb{R}^n} \\
\text{s.t. } & h_i(x) = 0, \; i = 1, \ldots, p
\end{split}
\tag{ECP}
$$

$$
L(x, \nu) = f(x) + \sum_{i=1}^{p} \nu_i h_i(x) = f(x) + \nu^\top h(x)
$$

Пусть $f(x)$ и $h_i(x)$ дважды дифференцируемы в точке x^* и непрерывно дифференцируемы в некоторой окрестности x^*. Условия локального минимума для $x \in \mathbb{R}^n, \nu \in \mathbb{R}^p$ записываются как

Необходимые условия

$$
\nabla_x L(x^*, \nu^*) = 0
$$

$$
\nabla_\nu L(x^*, \nu^*) = 0
$$

Пример. Задача наименьших квадратов

Example

Поставим задачу оптимизации и решим ее для линейной системы $Ax = b, A \in \mathbb{R}^{m \times n}$ для трех случаев (предполагая, что матрица имеет полный ранг):

- $m < n$

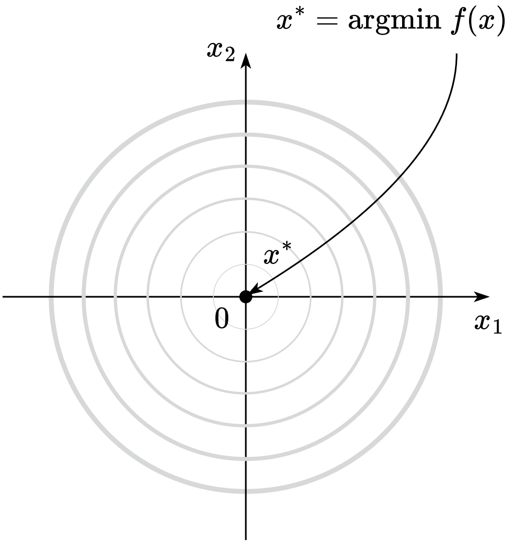
Задачи с ограничениями-неравенствами.

$$
\begin{split}
& f(x) \to \min_{x \in \mathbb{R}^n} \\
\text{s.t. } & g(x) \leq 0
\end{split}
$$

Рассмотрим на примере

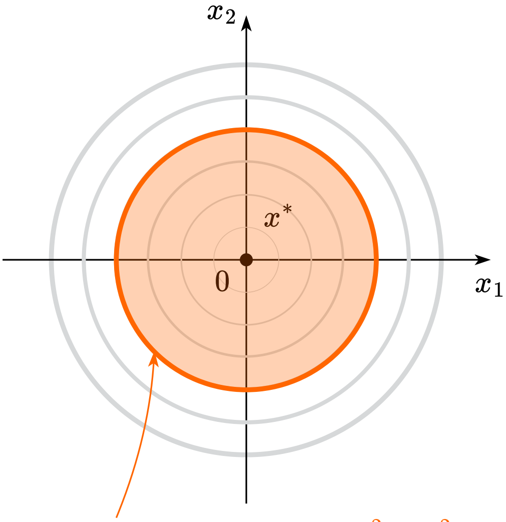
$$
f(x) = x_1^2 + x_2^2 \quad g(x) = x_1^2 + x_2^2 - 1
$$

Задачи с ограничениями-неравенствами



Линии уровня $f(x) = x_1^2 + x_2^2 = C$

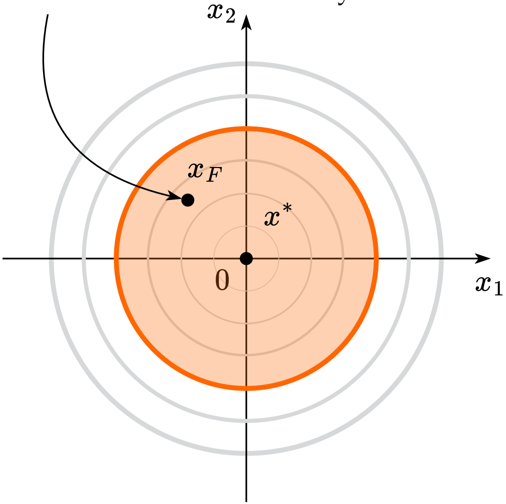
Задачи с ограничениями-неравенствами



Бюджетное множество $g(x) = x_1^2 + x_2^2 - 1 \leq 0$

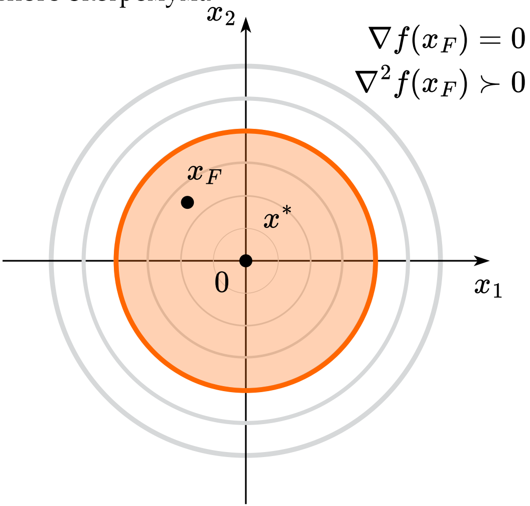
Задачи с ограничениями-неравенствами

Как понять, что некоторая допустимая точка является локальным минимумом?



Задачи с ограничениями-неравенствами

Просто! Проверим достаточные условия локального экстремума



Задачи с ограничениями-неравенствами

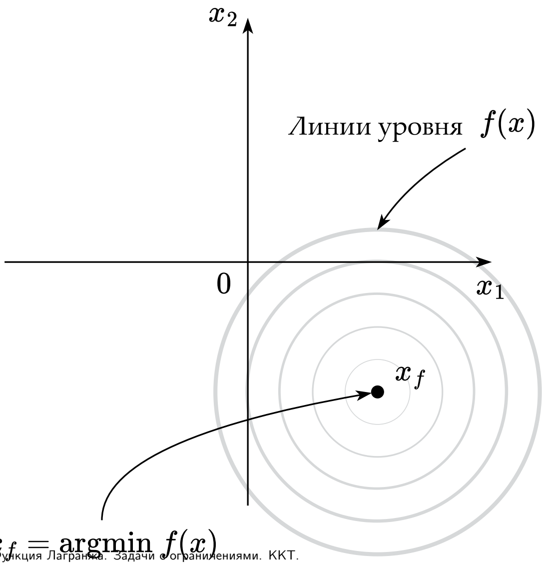
Таким образом, если ограничения типа неравенства неактивны в условной задаче, то мы можем решать задачу без ограничений. Однако так бывает не всегда. Рассмотрим второй простой пример.

$$
f(x) = (x_1 - 1)^2 + (x_2 + 1)^2 \quad g(x) = x_1^2 + x_2^2 - 1
$$

$$
\begin{split}
& f(x) \to \min_{x \in \mathbb{R}^n} \\
\text{s.t. } & g(x) \leq 0
\end{split}
$$

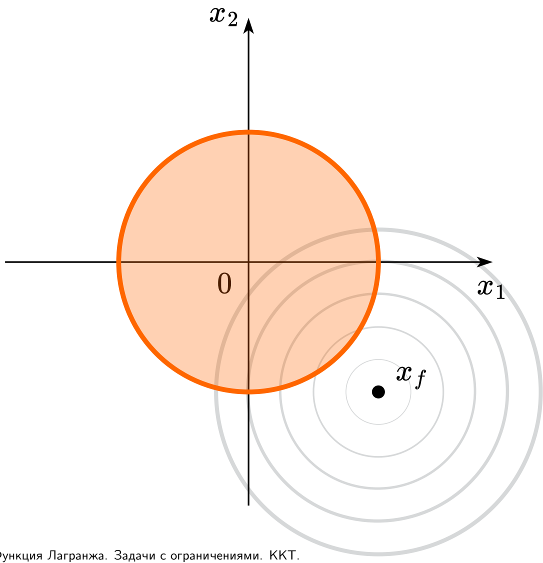
Задачи с ограничениями-неравенствами

$$f(x) = (x_1 - 1)^2 + (x_2 + 1)^2 = C$$



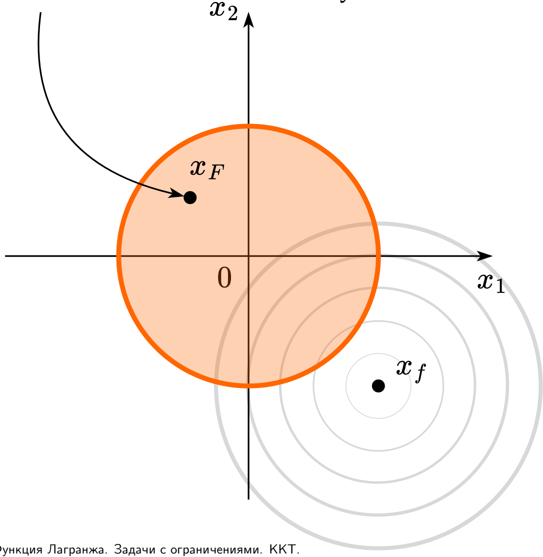
Задачи с ограничениями-неравенствами

Бюджетное множество $g(x) = x_1^2 + x_2^2 - 1 \leq 0$



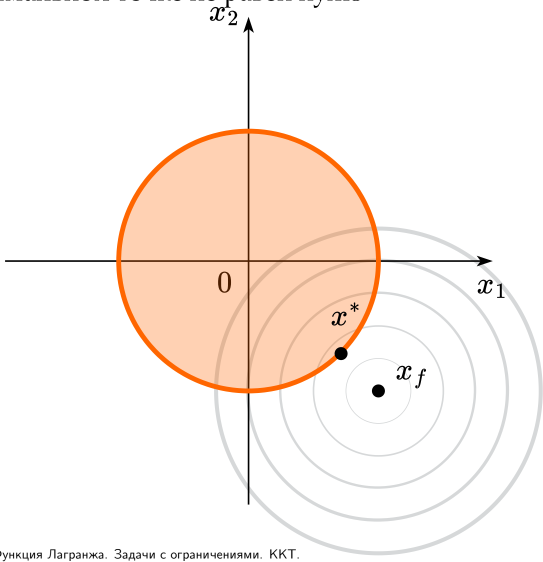
Задачи с ограничениями-неравенствами

Как понять, что некоторая допустимая точка является локальным минимумом?



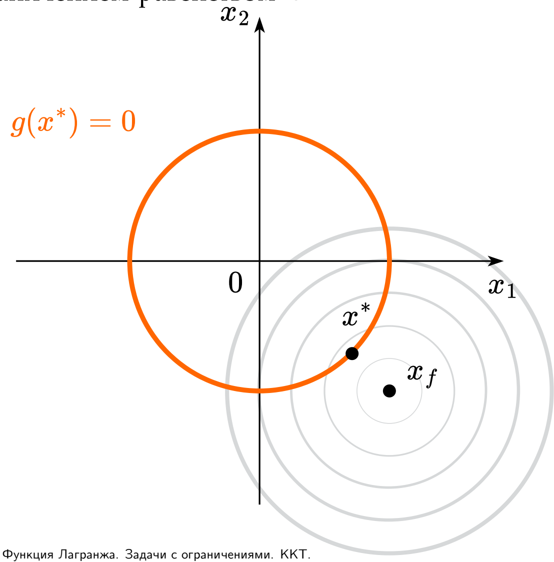
Задачи с ограничениями-неравенствами

Не так просто! Даже градиент
в оптимальной точке не равен нулю 😭

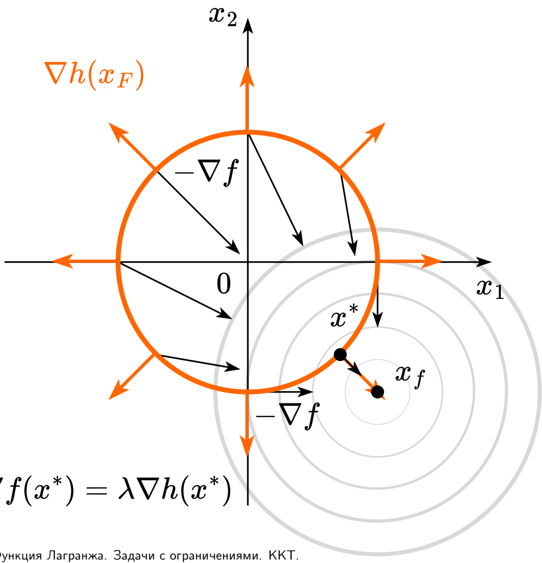


Задачи с ограничениями-неравенствами

Фактически имеем задачу
с ограничением-равенством 💡



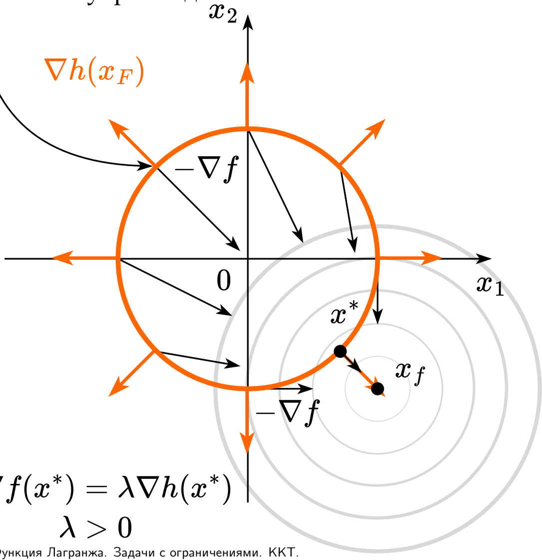
Задачи с ограничениями-неравенствами



$$-\nabla f(x^*) = \lambda \nabla h(x^*)$$

Задачи с ограничениями-неравенствами

Не является локальным минимумом, т.к. $-\nabla f(x)$ направлен внутрь бюджетного множества



$$-\nabla f(x^*) = \lambda \nabla h(x^*)$$
$$\lambda > 0$$

Задачи с ограничениями-неравенствами

Итак, у нас есть задача:

$$
f(x) \to \min_{x \in \mathbb{R}^n}
$$
$$
\text{s.t. } g(x) \leq 0
$$

Хотим проверить, является ли некоторый x^* точкой локального минимума. Два возможных случая:

$g(x) \leq 0$ неактивно: $g(x^*) < 0$

- $g(x^*) < 0$

Задачи с ограничениями-неравенствами

Итак, у нас есть задача:

$$f(x) \to \min_{x \in \mathbb{R}^n}$$

$$\text{s.t. } g(x) \leq 0$$

Хотим проверить, является ли некоторый x^* точкой локального минимума. Два возможных случая:

$g(x) \leq 0$ неактивно: $g(x^*) < 0$

- $g(x^*) < 0$
- $\nabla f(x^*) = 0$

Задачи с ограничениями-неравенствами

Итак, у нас есть задача:

$$
f(x) \to \min_{x \in \mathbb{R}^n}
$$

$$
\text{s.t. } g(x) \leq 0
$$

Хотим проверить, является ли некоторый x^* точкой локального минимума. Два возможных случая:

$g(x) \leq 0$ неактивно: $g(x^*) < 0$

- $g(x^*) < 0$
- $\nabla f(x^*) = 0$
- $\nabla^2 f(x^*) \succ 0$

Задачи с ограничениями-неравенствами

Итак, у нас есть задача:

$$
f(x) \to \min_{x \in \mathbb{R}^n}
$$
$$
\text{s.t. } g(x) \leq 0
$$

Хотим проверить, является ли некоторый x^* точкой локального минимума. Два возможных случая:

$g(x) \leq 0$ неактивно: $g(x^*) < 0$

- $g(x^*) < 0$
- $\nabla f(x^*) = 0$
- $\nabla^2 f(x^*) \succ 0$

Задачи с ограничениями-неравенствами

Итак, у нас есть задача:

$$
f(x) \to \min_{x \in \mathbb{R}^n}
$$

$$
\text{s.t. } g(x) \leq 0
$$

Хотим проверить, является ли некоторый x^* точкой локального минимума. Два возможных случая:

$g(x) \leq 0$ неактивно: $g(x^*) < 0$

- $g(x^*) < 0$
- $\nabla f(x^*) = 0$
- $\nabla^2 f(x^*) \succ 0$

$g(x) \leq 0$ активно: $g(x^*) = 0$

- $g(x^*) = 0$

Задачи с ограничениями-неравенствами

Итак, у нас есть задача:

$$
f(x) \to \min_{x \in \mathbb{R}^n}
$$

$$
\text{s.t. } g(x) \leq 0
$$

Хотим проверить, является ли некоторый x^* точкой локального минимума. Два возможных случая:

$g(x) \leq 0$ неактивно: $g(x^*) < 0$

- $g(x^*) < 0$
- $\nabla f(x^*) = 0$
- $\nabla^2 f(x^*) \succ 0$

$g(x) \leq 0$ активно: $g(x^*) = 0$

- $g(x^*) = 0$
- Необходимые условия: $-\nabla f(x^*) = \lambda \nabla g(x^*)$, $\lambda > 0$

Лагранжиан для задач с ограничениями-неравенствами

$$
\begin{split}
& f(x) \to \min_{x \in \mathbb{R}^n} \\
\text{s.t. } & g(x) \leq 0
\end{split}
$$

Определим функцию Лагранжа:

$$
L(x, \lambda) = f(x) + \lambda g(x)
$$

Классические условия Каруша-Куна-Таккера для локального минимума x^*, сформулированные при некоторых условиях *регулярности*:

Лагранжиан для задач с ограничениями-неравенствами

$$
f(x) \to \min_{x \in \mathbb{R}^n}
$$
$$
\text{s.t. } g(x) \leq 0
$$

Определим функцию Лагранжа:

$$
L(x, \lambda) = f(x) + \lambda g(x)
$$

Классические условия Каруша-Куна-Таккера для локального минимума x^*, сформулированные при некоторых условиях *регулярности*:

Пусть x^* является локальным минимумом для описанной выше задачи и задача *регулярна*, то существует единственный множитель Лагранжа λ^* такой, что:

$$
\begin{aligned}
&(1)\ \nabla_x L(x^*, \lambda^*) = 0 \\
&(2)\ \lambda^* \geq 0 \\
&(3)\ \lambda^* g(x^*) = 0 \\
&(4)\ g(x^*) \leq 0
\end{aligned}
$$

Общая формулировка

$$
\begin{aligned}
f_0(x) &\to \min_{x \in \mathbb{R}^n} \\
\text{s.t. } f_i(x) &\leq 0, \; i = 1, \ldots, m \\
h_j(x) &= 0, \; j = 1, \ldots, p
\end{aligned}
$$

Данная формулировка является **общей задачей математического программирования**. Если $f_i(x), i = 0, \ldots, n$ выпуклы, а $h_j(x), j = 1, \ldots, m$ аффинны, такая задача называется **задачей выпуклого программирования**.

$$
L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x)
$$

Пусть x^*, (λ^*, ν^*) является решением **регулярной** задачи математического программирования. Пусть также функции f_0, f_i, h_i дифференцируемы.

- $\nabla_x L(x^*, \lambda^*, \nu^*) = 0$

Общая формулировка

$$
\begin{aligned}
& f_0(x) \to \min_{x \in \mathbb{R}^n} \\
\text{s.t. } & f_i(x) \leq 0, \; i = 1, \ldots, m \\
& h_j(x) = 0, \; j = 1, \ldots, p
\end{aligned}
$$

Данная формулировка является **общей задачей математического программирования**. Если $f_i(x), i = 0, \ldots, n$ выпуклы, а $h_j(x), j = 1, \ldots, m$ аффинны, такая задача называется **задачей выпуклого программирования**.

$$
L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{i=1}^p \nu_i h_i(x)
$$

Пусть x^*, (λ^*, ν^*) является решением **регулярной** задачи математического программирования. Пусть также функции f_0, f_i, h_i дифференцируемы.

- $\nabla_x L(x^*, \lambda^*, \nu^*) = 0$
- $\nabla_\nu L(x^*, \lambda^*, \nu^*) = 0$

Общая формулировка

$$
\begin{split}
& f_0(x) \to \min_{x \in \mathbb{R}^n} \\
\text{s.t. } & f_i(x) \leq 0, \; i = 1, \ldots, m \\
& h_j(x) = 0, \; j = 1, \ldots, p
\end{split}
$$

Данная формулировка является **общей задачей математического программирования**. Если $f_i(x), i = 0, \ldots, n$ выпуклы, а $h_j(x), j = 1, \ldots, m$ аффинны, такая задача называется **задачей выпуклого программирования**.

$$
L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{i=1}^p \nu_i h_i(x)
$$

Пусть x^*, (λ^*, ν^*) является решением **регулярной** задачи математического программирования. Пусть также функции f_0, f_i, h_i дифференцируемы.

- $\nabla_x L(x^*, \lambda^*, \nu^*) = 0$
- $\nabla_\nu L(x^*, \lambda^*, \nu^*) = 0$
- $\lambda_i^* \geq 0, i = 1, \ldots, m$

Общая формулировка

$$
\begin{split}
& f_0(x) \to \min_{x \in \mathbb{R}^n} \\
\text{s.t. } & f_i(x) \leq 0, \; i = 1, \ldots, m \\
& h_j(x) = 0, \; j = 1, \ldots, p
\end{split}
$$

Данная формулировка является **общей задачей математического программирования**. Если $f_i(x), i = 0, ..., n$ выпуклы, а $h_j(x), j = 1, ..., m$ аффинны, такая задача называется **задачей выпуклого программирования**.

$$
L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{i=1}^p \nu_i h_i(x)
$$

Пусть x^*, (λ^*, ν^*) является решением **регулярной** задачи математического программирования. Пусть также функции f_0, f_i, h_i дифференцируемы.

- $\nabla_x L(x^*, \lambda^*, \nu^*) = 0$
- $\nabla_\nu L(x^*, \lambda^*, \nu^*) = 0$
- $\lambda^*_i \geq 0, i = 1, \ldots, m$
- $\lambda^*_i f_i(x^*) = 0, i = 1, \ldots, m$

Общая формулировка

$$
\begin{split}
& f_0(x) \to \min_{x \in \mathbb{R}^n} \\
\text{s.t. } & f_i(x) \leq 0, \; i = 1, \ldots, m \\
& h_j(x) = 0, \; j = 1, \ldots, p
\end{split}
$$

Данная формулировка является **общей задачей математического программирования**. Если $f_i(x), i = 0, ..., n$ выпуклы, а $h_j(x), j = 1, ..., m$ аффинны, такая задача называется **задачей выпуклого программирования**.

$$
L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{i=1}^p \nu_i h_i(x)
$$

Пусть x^*, (λ^*, ν^*) является решением **регулярной** задачи математического программирования. Пусть также функции f_0, f_i, h_i дифференцируемы.

- $\nabla_x L(x^*, \lambda^*, \nu^*) = 0$
- $\nabla_\nu L(x^*, \lambda^*, \nu^*) = 0$
- $\lambda_i^* \geq 0, i = 1, \ldots, m$
- $\lambda_i^* f_i(x^*) = 0, i = 1, \ldots, m$
- $f_i(x^*) \leq 0, i = 1, \ldots, m$

Общая формулировка

$$
\begin{aligned}
f_0(x) &\to \min_{x \in \mathbb{R}^n} \\
\text{s.t. } f_i(x) &\le 0, \; i = 1, \ldots, m \\
h_j(x) &= 0, \; j = 1, \ldots, p
\end{aligned}
$$

Данная формулировка является **общей задачей математического программирования**. Если $f_i(x), i = 0, \ldots, n$ выпуклы, а $h_j(x), j = 1, \ldots, m$ аффинны, такая задача называется **задачей выпуклого программирования**.

$$
L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{i=1}^p \nu_i h_i(x)
$$

Пусть x^*, (λ^*, ν^*) является решением **регулярной** задачи математического программирования. Пусть также функции f_0, f_i, h_i дифференцируемы.

- $\nabla_x L(x^*, \lambda^*, \nu^*) = 0$
- $\nabla_\nu L(x^*, \lambda^*, \nu^*) = 0$
- $\lambda^*_i \ge 0, i = 1, \ldots, m$
- $\lambda^*_i f_i(x^*) = 0, i = 1, \ldots, m$
- $f_i(x^*) \le 0, i = 1, \ldots, m$

Общая формулировка

$$
\begin{split}
& f_0(x) \to \min_{x \in \mathbb{R}^n} \\
\text{s.t. } & f_i(x) \leq 0, \; i = 1, \ldots, m \\
& h_j(x) = 0, \; j = 1, \ldots, p
\end{split}
$$

Данная формулировка является **общей задачей математического программирования**. Если $f_i(x), i = 0, \ldots, n$ выпуклы, а $h_j(x), j = 1, \ldots, m$ аффинны, такая задача называется **задачей выпуклого программирования**.

$$
L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{i=1}^p \nu_i h_i(x)
$$

Пусть x^*, (λ^*, ν^*) является решением **регулярной** задачи математического программирования. Пусть также функции f_0, f_i, h_i дифференцируемы.

- $\nabla_x L(x^*, \lambda^*, \nu^*) = 0$
- $\nabla_\nu L(x^*, \lambda^*, \nu^*) = 0$
- $\lambda_i^* \geq 0, i = 1, \ldots, m$
- $\lambda_i^* f_i(x^*) = 0, i = 1, \ldots, m$
- $f_i(x^*) \leq 0, i = 1, \ldots, m$

- **Условие Слейтера.** Если задачи выпуклого программирования существует точка x такая, что $h(x) = 0$ и $f_i(x) < 0$ (существует строго допустимая точка), и условия Каруша—Куна—Таккера становятся **необходимыми и достаточными**.

Общая формулировка

$$
\begin{aligned}
& f_0(x) \to \min_{x \in \mathbb{R}^n} \\
\text{s.t. } & f_i(x) \leq 0, \; i = 1, \ldots, m \\
& h_j(x) = 0, \; j = 1, \ldots, p
\end{aligned}
$$

Данная формулировка является **общей задачей математического программирования**. Если $f_i(x), i = 0, ..., n$ выпуклы, а $h_j(x), j = 1, ..., m$ аффинны, такая задача называется **задачей выпуклого программирования**.

$$
L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{i=1}^p \nu_i h_i(x)
$$

Пусть x^*, (λ^*, ν^*) является решением **регулярной** задачи математического программирования. Пусть также функции f_0, f_i, h_i дифференцируемы.

- $\nabla_x L(x^*, \lambda^*, \nu^*) = 0$
- $\nabla_\nu L(x^*, \lambda^*, \nu^*) = 0$
- $\lambda^*_i \geq 0, i = 1, \ldots, m$
- $\lambda^*_i f_i(x^*) = 0, i = 1, \ldots, m$
- $f_i(x^*) \leq 0, i = 1, \ldots, m$

- **Условие Слейтера.** Если задачи выпуклого программирования существует точка x такая, что $h(x) = 0$ и $f_i(x) < 0$ (существует строго допустимая точка), и условия Каруша—Куна—Таккера становятся **необходимыми и достаточными**.
- Существуют и другие условия регулярности, но они дают только **необходимость ККТ**.

Пример.

Question

Функция $f : E \to \mathbb{R}$ определена как

$$f(x) = \ln\left(-Q(x)\right)$$

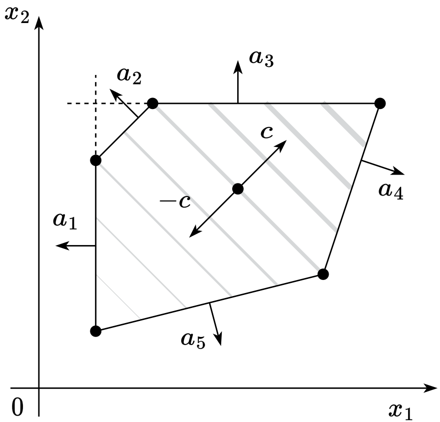
где $E = \{x \in \mathbb{R}^n : Q(x) < 0\}$ и

$$Q(x) = \frac{1}{2}x^\top Ax + b^\top x + c$$

с $A \in \mathbb{S}^n_{++}$, $b \in \mathbb{R}^n$, $c \in \mathbb{R}$.
Найдите точку максимума x^* функции f.

Лекция 6. Линейное программирование. Симплекс-метод.

Постановка задачи линейного программирования

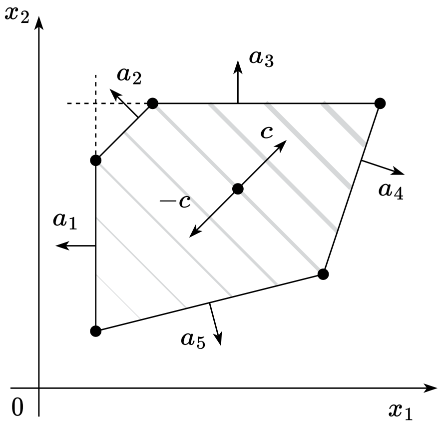


Задача ЛП в базовой (канонической) форме

$$\min_{x \in \mathbb{R}^n} c^\top x \quad \text{s.t. } Ax \leq b \tag{LP.Basic}$$

$c \in \mathbb{R}^n$, $b \in \mathbb{R}^m$, $A \in \mathbb{R}^{m \times n}$, где неравенства — покомпонентные.

Постановка задачи линейного программирования



Задача ЛП в базовой (канонической) форме

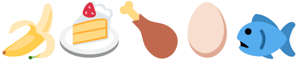
$$
\begin{aligned}
&\min_{x \in \mathbb{R}^n} c^\top x \\
\text{s.t. } &Ax \leq b
\end{aligned}
\tag{LP.Basic}
$$

$c \in \mathbb{R}^n$, $b \in \mathbb{R}^m$, $A \in \mathbb{R}^{m \times n}$, где неравенства — покомпонентные.
Задача ЛП в стандартная форме.

$$
\begin{aligned}
&\min_{x \in \mathbb{R}^n} c^\top x \\
\text{s.t. } &Ax = b \\
&x_i \geq 0, \; i = 1, \dots, n
\end{aligned}
\tag{LP.Standard}
$$

$c \in \mathbb{R}^n$, $b \in \mathbb{R}^m$ и матрица $A \in \mathbb{R}^{m \times n}$.

Пример: задача о диете



$$
\begin{array}{r}
\text{Белки} \\[2em]
\text{Жиры} \\[2em]
\text{Углеводы} \\[2em]
\text{Калории} \\[2em]
\text{Витамин D}
\end{array}
\begin{bmatrix}
\\[2em]
\\[2em]
\text{Количество на 100г} \\
W \in \mathbb{R}^{n \times p} \\[2em]
\\[2em]
\end{bmatrix}
$$

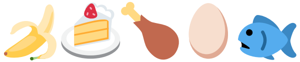
$c \in \mathbb{R}^p$, цена за 100г

$r \in \mathbb{R}^n$, ограничения

$x \in \mathbb{R}^p$, количество продуктов

$$
\begin{aligned}
\min_{x \in \mathbb{R}^p} \; & c^T x \\
Wx & \succeq r \\
x & \succeq 0
\end{aligned}
$$

Пример: задача о диете



$$
\begin{array}{l c}
 & \\
\text{Белки} & \\
\text{Жиры} & \\
\text{Углеводы} & \\
\text{Калории} & \\
\text{Витамин D} &
\end{array}
\left[\begin{array}{c}
\\
\\
\text{Количество на 100г} \\
W \in \mathbb{R}^{n \times p} \\
\\
\\
\end{array}\right]
$$

$c \in \mathbb{R}^p$, цена за 100г

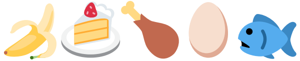
$r \in \mathbb{R}^n$, ограничения

$x \in \mathbb{R}^p$, количество продуктов

$$
\begin{aligned}
\min_{x \in \mathbb{R}^p} c^T x \\
Wx \succeq r \\
x \succeq 0
\end{aligned}
$$

Представьте, что вам нужно составить план диеты из некоторых продуктов: бананы, пироги, курица, яйца, рыба. Каждый из продуктов имеет свой вектор питательных веществ. Таким образом, все питательные вещества можно представить в виде матрицы W.

Пример: задача о диете



$$
\begin{array}{c}
\text{Белки} \\
\text{Жиры} \\
\text{Углеводы} \\
\text{Калории} \\
\text{Витамин D}
\end{array}
\left[
\begin{array}{c}
\\
\\
\text{Количество на 100г} \\
W \in \mathbb{R}^{n \times p} \\
\\
\end{array}
\right]
$$

$c \in \mathbb{R}^p$, цена за 100г

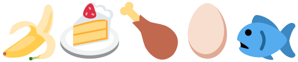
$r \in \mathbb{R}^n$, ограничения

$x \in \mathbb{R}^p$, количество продуктов

$$
\begin{split}
& \min_{x \in \mathbb{R}^p} c^T x \\
& Wx \succeq r \\
& x \succeq 0
\end{split}
$$

Представьте, что вам нужно составить план диеты из некоторых продуктов: бананы, пироги, курица, яйца, рыба. Каждый из продуктов имеет свой вектор питательных веществ. Таким образом, все питательные вещества можно представить в виде матрицы W. Предположим, что у нас есть вектор требований для каждого питательного вещества $r \in \mathbb{R}^n$. Нам нужно найти самую дешёвую диету, которая удовлетворяет всем требованиям:

Пример: задача о диете



Белки

Жиры

Углеводы

Калории

Витамин D

Количество на 100г

$W \in \mathbb{R}^{n \times p}$

$c \in \mathbb{R}^p$, цена за 100г

$r \in \mathbb{R}^n$, ограничения

$x \in \mathbb{R}^p$, количество продуктов

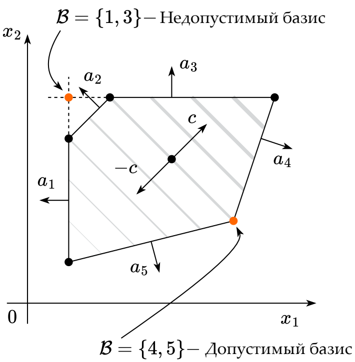
$$
\begin{aligned}
&\min_{x \in \mathbb{R}^p} c^T x \\
&Wx \succeq r \\
&x \succeq 0
\end{aligned}
$$

Представьте, что вам нужно составить план диеты из некоторых продуктов: бананы, пироги, курица, яйца, рыба. Каждый из продуктов имеет свой вектор питательных веществ. Таким образом, все питательные вещества можно представить в виде матрицы W. Предположим, что у нас есть вектор требований для каждого питательного вещества $r \in \mathbb{R}^n$. Нам нужно найти самую дешёвую диету, которая удовлетворяет всем требованиям:

$$
\begin{aligned}
&\min_{x \in \mathbb{R}^p} c^\top x \\
\text{s.t. } & Wx \succeq r \\
& x_i \geq 0, \; i = 1, \dots, p
\end{aligned}
$$

Open In Colab

Симплекс-метод. Основная идея.

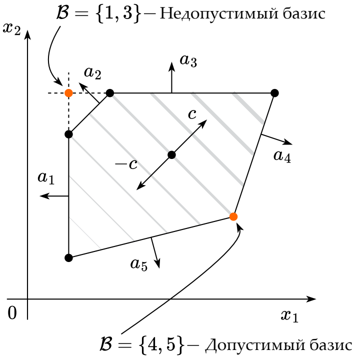


$$
\begin{aligned}
&\min_{x \in \mathbb{R}^n} c^\top x \\
\text{s.t. } & Ax \leq b
\end{aligned}
\tag{LP.Basic}
$$

- Определение: **базис** $\mathcal{B}$ — это подмножество n (целых) чисел между 1 и m, такое что $\text{rank} A_{\mathcal{B}} = n$.

Верхнеуровневая идея симплекс-метода:

Симплекс-метод. Основная идея.

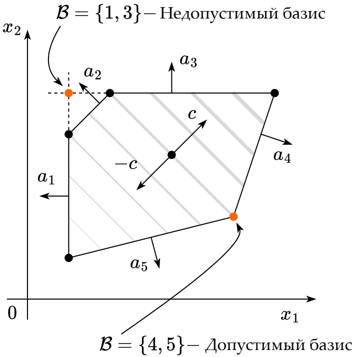


$$
\begin{aligned}
&\min_{x \in \mathbb{R}^n} c^\top x \\
&\text{s.t. } Ax \leq b
\end{aligned}
\tag{LP.Basic}
$$

- Определение: **базис** $\mathcal{B}$ — это подмножество n (целых) чисел между 1 и m, такое что $\text{rank} A_{\mathcal{B}} = n$.
- Обратите внимание, что мы можем связать подматрицу $A_{\mathcal{B}}$ и соответствующую правую часть $b_{\mathcal{B}}$ с базисом $\mathcal{B}$.

Верхнеуровневая идея симплекс-метода:

Симплекс-метод. Основная идея.

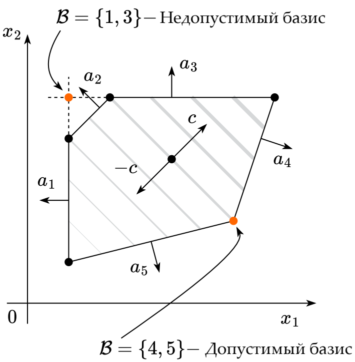


$$
\begin{aligned}
&\min_{x \in \mathbb{R}^n} c^\top x \\
&\text{s.t. } Ax \leq b
\end{aligned}
\tag{LP.Basic}
$$

- Определение: **базис** $\mathcal{B}$ — это подмножество n (целых) чисел между 1 и m, такое что $\text{rank} A_{\mathcal{B}} = n$.
- Обратите внимание, что мы можем связать подматрицу $A_{\mathcal{B}}$ и соответствующую правую часть $b_{\mathcal{B}}$ с базисом $\mathcal{B}$.
- Также мы можем получить точку пересечения всех этих гиперплоскостей из базиса: $x_{\mathcal{B}} = A_{\mathcal{B}}^{-1} b_{\mathcal{B}}$.

Верхнеуровневая идея симплекс-метода:

Симплекс-метод. Основная идея.

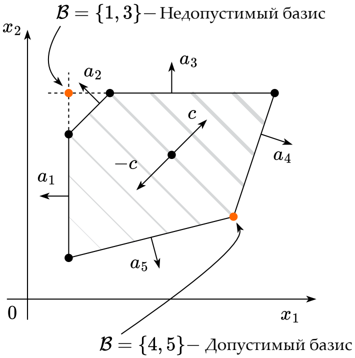


$$
\begin{aligned}
&\min_{x \in \mathbb{R}^n} c^\top x \\
&\text{s.t. } Ax \leq b
\end{aligned}
\tag{LP.Basic}
$$

- Определение: **базис** $\mathcal{B}$ — это подмножество n (целых) чисел между 1 и m, такое что $\text{rank} A_{\mathcal{B}} = n$.
- Обратите внимание, что мы можем связать подматрицу $A_{\mathcal{B}}$ и соответствующую правую часть $b_{\mathcal{B}}$ с базисом $\mathcal{B}$.
- Также мы можем получить точку пересечения всех этих гиперплоскостей из базиса: $x_{\mathcal{B}} = A_{\mathcal{B}}^{-1} b_{\mathcal{B}}$.
- Если $Ax_{\mathcal{B}} \leq b$, то базис $\mathcal{B}$ является **допустимым**.

Верхнеуровневая идея симплекс-метода:

Симплекс-метод. Основная идея.

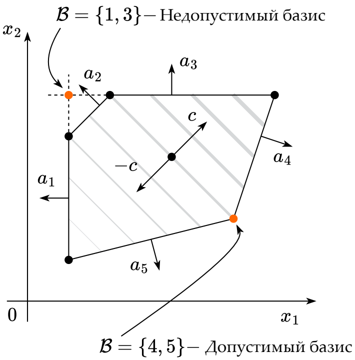


$$
\begin{split}
&\min_{x \in \mathbb{R}^n} c^\top x \\
\text{s.t. } &Ax \leq b
\end{split}
\qquad \text{(LP.Basic)}
$$

- Определение: **базис** $\mathcal{B}$ — это подмножество n (целых) чисел между 1 и m, такое что $\text{rank} A_{\mathcal{B}} = n$.
- Обратите внимание, что мы можем связать подматрицу $A_{\mathcal{B}}$ и соответствующую правую часть $b_{\mathcal{B}}$ с базисом $\mathcal{B}$.
- Также мы можем получить точку пересечения всех этих гиперплоскостей из базиса: $x_{\mathcal{B}} = A_{\mathcal{B}}^{-1} b_{\mathcal{B}}$.
- Если $Ax_{\mathcal{B}} \leq b$, то базис $\mathcal{B}$ является **допустимым**.
- Базис $\mathcal{B}$ оптимален, если $x_{\mathcal{B}}$ является решением задачи LP.Inequality.

Верхнеуровневая идея симплекс-метода:

Симплекс-метод. Основная идея.

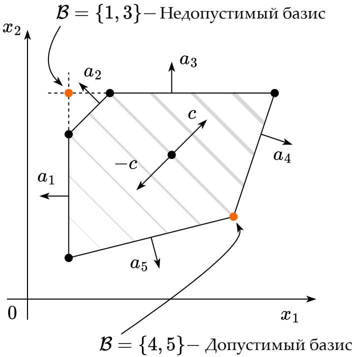


$$
\begin{aligned}
&\min_{x \in \mathbb{R}^n} c^\top x \\
&\text{s.t. } Ax \leq b
\end{aligned}
\tag{LP.Basic}
$$

- Определение: **базис** $\mathcal{B}$ — это подмножество n (целых) чисел между 1 и m, такое что $\text{rank} A_{\mathcal{B}} = n$.
- Обратите внимание, что мы можем связать подматрицу $A_{\mathcal{B}}$ и соответствующую правую часть $b_{\mathcal{B}}$ с базисом $\mathcal{B}$.
- Также мы можем получить точку пересечения всех этих гиперплоскостей из базиса: $x_{\mathcal{B}} = A_{\mathcal{B}}^{-1} b_{\mathcal{B}}$.
- Если $Ax_{\mathcal{B}} \leq b$, то базис $\mathcal{B}$ является **допустимым**.
- Базис $\mathcal{B}$ оптимален, если $x_{\mathcal{B}}$ является решением задачи LP.Inequality.
- $x_{\mathcal{B}}$ называют **базисной точкой** или базисным решением (иногда её тоже называют **базисом**).

Верхнеуровневая идея симплекс-метода:

Симплекс-метод. Основная идея.



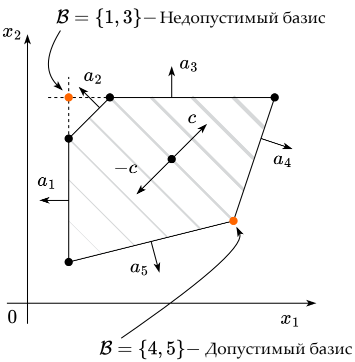
$$
\begin{aligned}
&\min_{x \in \mathbb{R}^n} c^\top x \\
\text{s.t. } & Ax \leq b
\end{aligned}
\tag{LP.Basic}
$$

- Определение: **базис** $\mathcal{B}$ — это подмножество n (целых) чисел между 1 и m, такое что $\text{rank} A_{\mathcal{B}} = n$.
- Обратите внимание, что мы можем связать подматрицу $A_{\mathcal{B}}$ и соответствующую правую часть $b_{\mathcal{B}}$ с базисом $\mathcal{B}$.
- Также мы можем получить точку пересечения всех этих гиперплоскостей из базиса: $x_{\mathcal{B}} = A_{\mathcal{B}}^{-1} b_{\mathcal{B}}$.
- Если $Ax_{\mathcal{B}} \leq b$, то базис $\mathcal{B}$ является **допустимым**.
- Базис $\mathcal{B}$ оптимален, если $x_{\mathcal{B}}$ является решением задачи LP.Inequality.
- $x_{\mathcal{B}}$ называют **базисной точкой** или базисным решением (иногда её тоже называют **базисом**).

Верхнеуровневая идея симплекс-метода:

- Убедитесь, что вы находитесь в вершине.

Симплекс-метод. Основная идея.



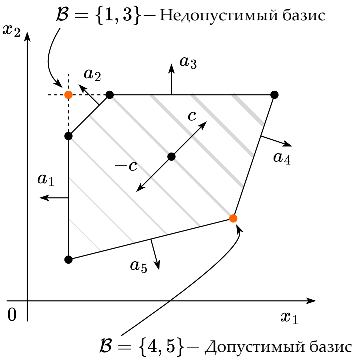
$$
\begin{aligned}
&\min_{x \in \mathbb{R}^n} c^\top x \\
&\text{s.t. } Ax \leq b
\end{aligned}
\tag{LP.Basic}
$$

- Определение: **базис** $\mathcal{B}$ — это подмножество n (целых) чисел между 1 и m, такое что $\text{rank} A_{\mathcal{B}} = n$.
- Обратите внимание, что мы можем связать подматрицу $A_{\mathcal{B}}$ и соответствующую правую часть $b_{\mathcal{B}}$ с базисом $\mathcal{B}$.
- Также мы можем получить точку пересечения всех этих гиперплоскостей из базиса: $x_{\mathcal{B}} = A_{\mathcal{B}}^{-1} b_{\mathcal{B}}$.
- Если $Ax_{\mathcal{B}} \leq b$, то базис $\mathcal{B}$ является **допустимым**.
- Базис $\mathcal{B}$ оптимален, если $x_{\mathcal{B}}$ является решением задачи LP.Inequality.
- $x_{\mathcal{B}}$ называют **базисной точкой** или базисным решением (иногда её тоже называют **базисом**).

Верхнеуровневая идея симплекс-метода:

- Убедитесь, что вы находитесь в вершине.
- Проверьте оптимальность.

Симплекс-метод. Основная идея.



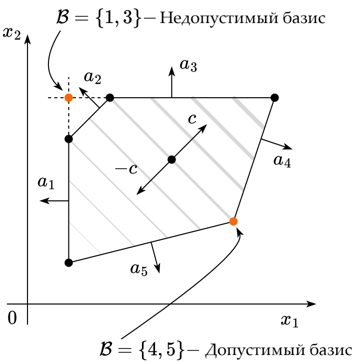
$$
\begin{aligned}
&\min_{x \in \mathbb{R}^n} c^\top x \\
&\text{s.t. } Ax \leq b
\end{aligned}
\tag{LP.Basic}
$$

- Определение: **базис** $\mathcal{B}$ — это подмножество n (целых) чисел между 1 и m, такое что $\text{rank} A_{\mathcal{B}} = n$.
- Обратите внимание, что мы можем связать подматрицу $A_{\mathcal{B}}$ и соответствующую правую часть $b_{\mathcal{B}}$ с базисом $\mathcal{B}$.
- Также мы можем получить точку пересечения всех этих гиперплоскостей из базиса: $x_{\mathcal{B}} = A_{\mathcal{B}}^{-1} b_{\mathcal{B}}$.
- Если $Ax_{\mathcal{B}} \leq b$, то базис $\mathcal{B}$ является **допустимым**.
- Базис $\mathcal{B}$ оптимален, если $x_{\mathcal{B}}$ является решением задачи LP.Inequality.
- $x_{\mathcal{B}}$ называют **базисной точкой** или базисным решением (иногда её тоже называют **базисом**).

Верхнеуровневая идея симплекс-метода:

- Убедитесь, что вы находитесь в вершине.
- Проверьте оптимальность.
- Если необходимо, перейдите к другой вершине (измените базис).

Симплекс-метод. Основная идея.



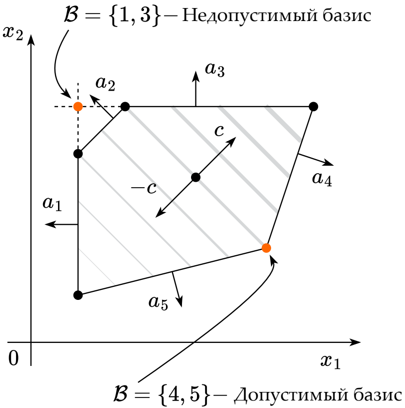
$$
\begin{aligned}
&\min_{x \in \mathbb{R}^n} c^\top x \\
&\text{s.t. } Ax \leq b
\end{aligned}
\tag{LP.Basic}
$$

- Определение: **базис** $\mathcal{B}$ — это подмножество n (целых) чисел между 1 и m, такое что $\text{rank} A_{\mathcal{B}} = n$.
- Обратите внимание, что мы можем связать подматрицу $A_{\mathcal{B}}$ и соответствующую правую часть $b_{\mathcal{B}}$ с базисом $\mathcal{B}$.
- Также мы можем получить точку пересечения всех этих гиперплоскостей из базиса: $x_{\mathcal{B}} = A_{\mathcal{B}}^{-1} b_{\mathcal{B}}$.
- Если $Ax_{\mathcal{B}} \leq b$, то базис $\mathcal{B}$ является **допустимым**.
- Базис $\mathcal{B}$ оптимален, если $x_{\mathcal{B}}$ является решением задачи LP.Inequality.
- $x_{\mathcal{B}}$ называют **базисной точкой** или базисным решением (иногда её тоже называют **базисом**).

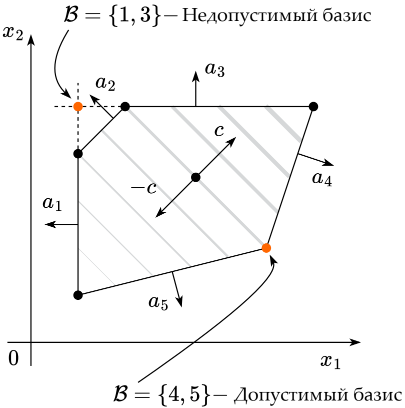
Верхнеуровневая идея симплекс-метода:

- Убедитесь, что вы находитесь в вершине.
- Проверьте оптимальность.
- Если необходимо, перейдите к другой вершине (измените базис).
- Повторяйте, пока не сойдётесь.

Оптимальный базис



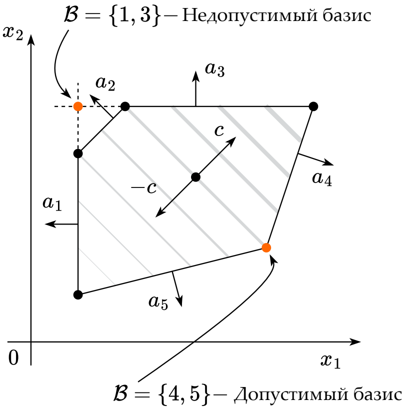
Оптимальный базис



Поскольку у нас есть базис, мы можем разложить наш целевой вектор c в этом базисе и найти скалярные коэффициенты $\lambda_{\mathcal{B}}$:

$$\lambda_{\mathcal{B}}^T A_{\mathcal{B}} = c^T \leftrightarrow \lambda_{\mathcal{B}}^T = c^T A_{\mathcal{B}}^{-1}$$

Оптимальный базис



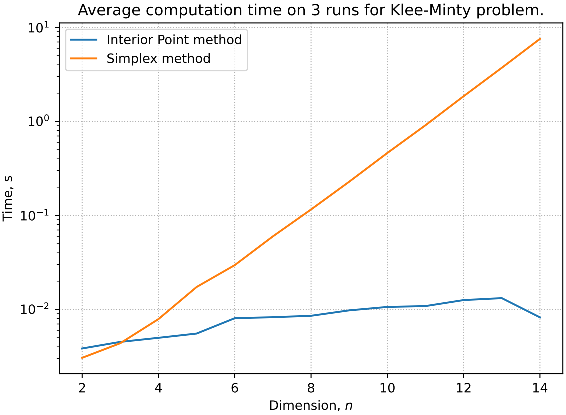
Поскольку у нас есть базис, мы можем разложить наш целевой вектор c в этом базисе и найти скалярные коэффициенты $\lambda_{\mathcal{B}}$:

$$\lambda_{\mathcal{B}}^T A_{\mathcal{B}} = c^T \leftrightarrow \lambda_{\mathcal{B}}^T = c^T A_{\mathcal{B}}^{-1}$$

Theorem

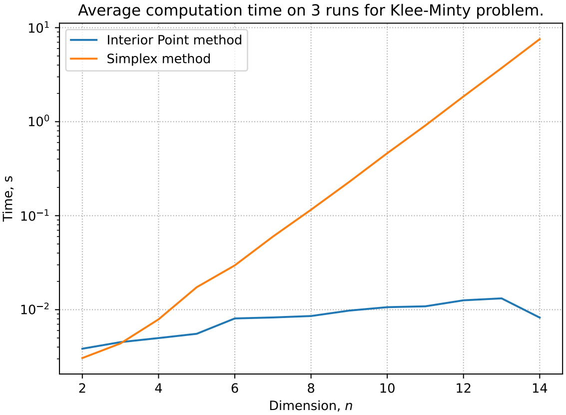
Если все компоненты $\lambda_{\mathcal{B}}$ неположительны и $\mathcal{B}$ допустим, то $\mathcal{B}$ оптимален.

Экспоненциальная сходимость



- Много прикладных задач может быть сформулировано в виде задач линейного программирования.

Экспоненциальная сходимость



- Много прикладных задач может быть сформулировано в виде задач линейного программирования.
- Симплекс-метод прост в своей основе, но в худшем случае может работать экспоненциально долго.

Примеры задач линейного программирования. Различные приложения

Посмотрите на различные практические приложения задач линейного программирования и симплекс-метода в Related Collab Notebook.

Лекция 7. Градиентный спуск. Скорости сходимости.

Виды выпуклости

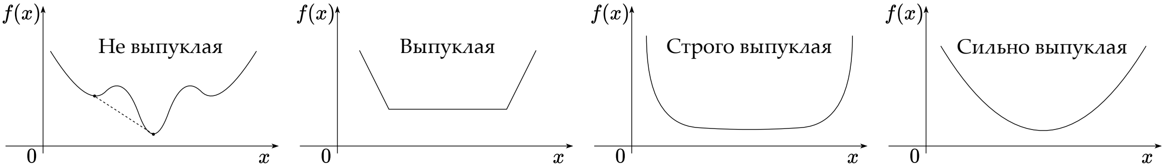


Рис. 57: Примеры выпуклых функций

Гладкость

Definition

Будем говорить, что функция $f : \mathbb{R}^n \to \mathbb{R}$ является L **-гладкой**, если $\forall x, y \in \mathbb{R}^n$ выполнено:

$$\|\nabla f(y) - \nabla f(x)\| \leqslant L\|y - x\|$$

Гладкость

Definition

Будем говорить, что функция $f : \mathbb{R}^n \to \mathbb{R}$ является L **-гладкой**, если $\forall x, y \in \mathbb{R}^n$ выполнено:

$$\|\nabla f(y) - \nabla f(x)\| \leqslant L\|y - x\|$$

Обратим внимание, что значение константы гладкости (Липшицевости градиента) зависит от выбора нормы.

Дифференциальные критерии L-гладкости (1 и 2 порядка)

Definition

- **Критерий 1-го порядка**: Пусть $f : \mathbb{R}^n \to \mathbb{R}$ дифференцируема. Тогда

$$f(x) \ L\text{-гладкая} \Leftrightarrow |f(y) - f(x) - \langle \nabla f(x), y - x \rangle| \leqslant \frac{L}{2} \|y - x\|^2, \ \forall x, y \in \mathbb{R}^n.$$

Дифференциальные критерии L-гладкости (1 и 2 порядка)

Definition

- **Критерий 1-го порядка**: Пусть $f : \mathbb{R}^n \to \mathbb{R}$ дифференцируема. Тогда

$$f(x)\ L\text{-гладкая} \Leftrightarrow |f(y) - f(x) - \langle \nabla f(x), y - x \rangle| \leqslant \frac{L}{2}\|y - x\|^2,\ \ \forall x, y \in \mathbb{R}^n.$$

- **Критерий 2-го порядка**: Пусть $f : \mathbb{R}^n \to \mathbb{R}$ дифференцируема. Тогда

$$f(x)\ L\text{-гладкая} \Leftrightarrow \nabla^2 f(x) \preceq LI, \forall x \in \mathbb{R}^n.$$

Липшицева парабола

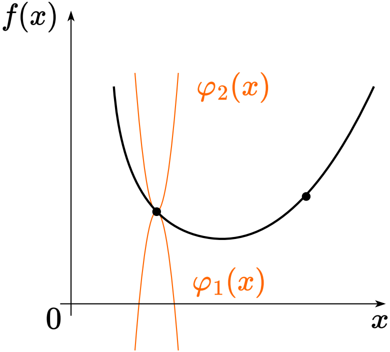


Рис. 58: Иллюстрация Липшицевых парабол, между которыми зажата гладкая функция. Чаще нас интересует мажорирующая из них.

Если зафиксируем $x_0 \in \mathbb{R}^n$, то:

$$\varphi_1(x) = f(x_0) + \langle f(x_0), x - x_0 \rangle - \frac{L}{2}\|x - x_0\|^2$$

$$\varphi_2(x) = f(x_0) + \langle f(x_0), x - x_0 \rangle + \frac{L}{2}\|x - x_0\|^2$$

Липшицева парабола

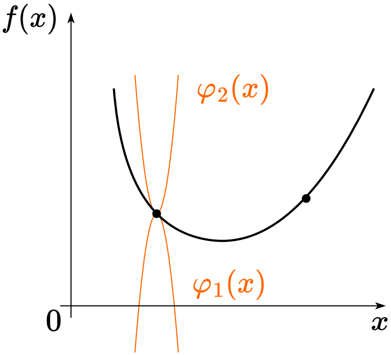


Рис. 58: Иллюстрация Липшицевых парабол, между которыми зажата гладкая функция. Чаще нас интересует мажорирующая из них.

Если зафиксируем $x_0 \in \mathbb{R}^n$, то:

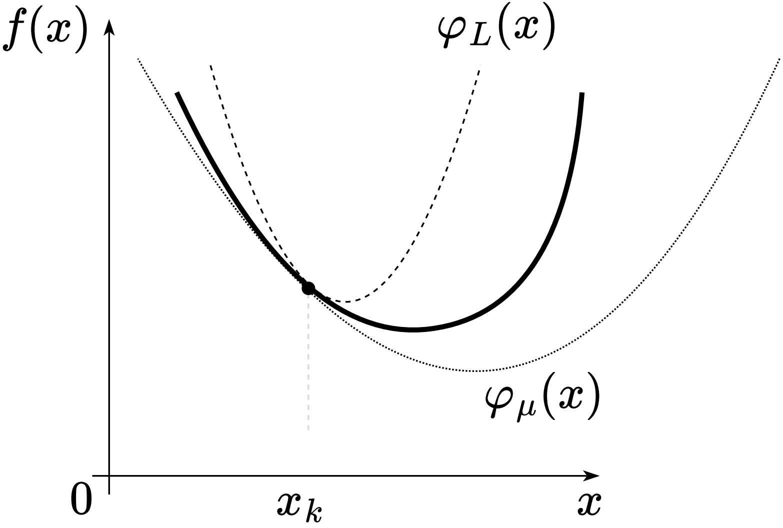
$$\varphi_1(x) = f(x_0) + \langle f(x_0), x - x_0 \rangle - \frac{L}{2}\|x - x_0\|^2$$

$$\varphi_2(x) = f(x_0) + \langle f(x_0), x - x_0 \rangle + \frac{L}{2}\|x - x_0\|^2$$

Это две параболы, и для них верно, что

$$\varphi_1(x) \leqslant f(x) \leqslant \varphi_2(x) \ \forall x$$

Гладкость и сильная выпуклость



Направление локального наискорейшего спуска

Рассмотрим линейное приближение дифференцируемой функции f вдоль направления h, где $\|h\|_2 = 1$:

Направление локального наискорейшего спуска

Рассмотрим линейное приближение дифференцируемой функции f вдоль направления h, где $\|h\|_2 = 1$:

$$
f(x + \alpha h) = f(x) + \alpha \langle \nabla f(x), h \rangle + o(\alpha)
$$

Направление локального наискорейшего спуска

Рассмотрим линейное приближение дифференцируемой функции f вдоль направления h, где $\|h\|_2 = 1$:

$$f(x + \alpha h) = f(x) + \alpha \langle \nabla f(x), h \rangle + o(\alpha)$$

Хотим, чтобы h было направлением убывания:

$$f(x + \alpha h) - f(x) < 0$$

$$\alpha \langle \nabla f(x), h \rangle + o(\alpha) < 0$$

Направление локального наискорейшего спуска

Рассмотрим линейное приближение дифференцируемой функции f вдоль направления h, где $\|h\|_2 = 1$:

$$f(x + \alpha h) = f(x) + \alpha \langle \nabla f(x), h \rangle + o(\alpha)$$

Хотим, чтобы h было направлением убывания:

$$f(x + \alpha h) - f(x) < 0$$

$$\alpha \langle \nabla f(x), h \rangle + o(\alpha) < 0$$

Переходя к пределу при $\alpha \to 0$:

$$\langle \nabla f(x), h \rangle < 0$$

Направление локального наискорейшего спуска

Рассмотрим линейное приближение дифференцируемой функции f вдоль направления h, где $\|h\|_2 = 1$:

$$
f(x + \alpha h) = f(x) + \alpha\langle \nabla f(x), h\rangle + o(\alpha)
$$

Хотим, чтобы h было направлением убывания:

$$
f(x + \alpha h) - f(x) < 0
$$

$$
\alpha\langle \nabla f(x), h\rangle + o(\alpha) < 0
$$

Переходя к пределу при $\alpha \to 0$:

$$
\langle \nabla f(x), h\rangle < 0
$$

Более того, мы хотим, чтобы разница $f(x) - f(x + \alpha h)$ была максимальна:

$$
h = \arg\max_h \left(-\langle \nabla f(x), h\rangle\right) = \arg\min_h \langle \nabla f(x), h\rangle.
$$

Также из неравенства Коши–Буняковского получаем:

$$
\begin{aligned}
|\langle \nabla f(x), h\rangle| &\leq \|\nabla f(x)\|_2 \|h\|_2 \\
\langle \nabla f(x), h\rangle &\geq -\|\nabla f(x)\|_2 \|h\|_2 = -\|\nabla f(x)\|_2
\end{aligned}
$$

Направление локального наискорейшего спуска

Рассмотрим линейное приближение дифференцируемой функции f вдоль направления h, где $\|h\|_2 = 1$:

$$f(x + \alpha h) = f(x) + \alpha \langle \nabla f(x), h \rangle + o(\alpha)$$

Хотим, чтобы h было направлением убывания:

$$f(x + \alpha h) - f(x) < 0$$

$$\alpha \langle \nabla f(x), h \rangle + o(\alpha) < 0$$

Переходя к пределу при $\alpha \to 0$:

$$\langle \nabla f(x), h \rangle < 0$$

Более того, мы хотим, чтобы разница $f(x) - f(x + \alpha h)$ была максимальна:

$$h = \arg\max_h \left(-\langle \nabla f(x), h \rangle \right) = \arg\min_h \langle \nabla f(x), h \rangle.$$

Также из неравенства Коши–Буняковского получаем:

$$|\langle \nabla f(x), h \rangle| \leq \|\nabla f(x)\|_2 \|h\|_2$$
$$\langle \nabla f(x), h \rangle \geq -\|\nabla f(x)\|_2 \|h\|_2 = -\|\nabla f(x)\|_2$$

Таким образом, направление антиградиента

$$h = \arg\min_h \langle \nabla f(x), h \rangle = -\frac{\nabla f(x)}{\|\nabla f(x)\|_2}$$

представляет собой направление **наискорейшего локального** убывания функции f.

Направление локального наискорейшего спуска

Рассмотрим линейное приближение дифференцируемой функции f вдоль направления h, где $\|h\|_2 = 1$:

$$f(x + \alpha h) = f(x) + \alpha \langle \nabla f(x), h \rangle + o(\alpha)$$

Хотим, чтобы h было направлением убывания:

$$f(x + \alpha h) - f(x) < 0$$

$$\alpha \langle \nabla f(x), h \rangle + o(\alpha) < 0$$

Переходя к пределу при $\alpha \to 0$:

$$\langle \nabla f(x), h \rangle < 0$$

Более того, мы хотим, чтобы разница $f(x) - f(x + \alpha h)$ была максимальна:

$$h = \arg\max_h \left(-\langle \nabla f(x), h \rangle\right) = \arg\min_h \langle \nabla f(x), h \rangle.$$

Также из неравенства Коши–Буняковского получаем:

$$|\langle \nabla f(x), h \rangle| \leq \|\nabla f(x)\|_2 \|h\|_2$$
$$\langle \nabla f(x), h \rangle \geq -\|\nabla f(x)\|_2 \|h\|_2 = -\|\nabla f(x)\|_2$$

Таким образом, направление антиградиента

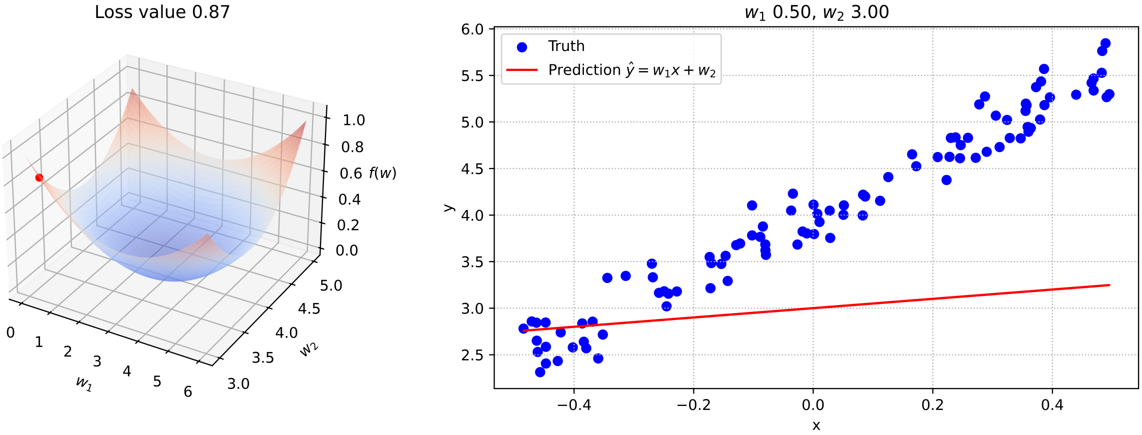
$$h = \arg\min_h \langle \nabla f(x), h \rangle = -\frac{\nabla f(x)}{\|\nabla f(x)\|_2}$$

представляет собой направление **наискорейшего локального** убывания функции f.
Итерация метода имеет вид:

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

Сходимость алгоритма градиентного спуска

Код для построения анимации ниже. Сходимость существенно зависит от выбора шага α:



Точный линейный поиск (метод наискорейшего спуска)

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

Подход скорее теоретический, чем практический: он удобен для анализа сходимости, но точный линейный поиск часто затруднён, если вычисление функции занимает слишком много времени или стоит слишком дорого.
Интересное теоретическое свойство этого метода заключается в том, что градиенты на соседних итерациях ортогональны. Условие оптимальности по α_k даёт

$$
\frac{d}{d\alpha} f(x^k - \alpha \nabla f(x^k))\Big|_{\alpha = \alpha_k} = 0.
$$

Точный линейный поиск (метод наискорейшего спуска)

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

Подход скорее теоретический, чем практический: он удобен для анализа сходимости, но точный линейный поиск часто затруднён, если вычисление функции занимает слишком много времени или стоит слишком дорого.

Интересное теоретическое свойство этого метода заключается в том, что градиенты на соседних итерациях ортогональны. Условие оптимальности по α_k даёт

$$
\frac{d}{d\alpha} f(x^k - \alpha \nabla f(x^k))\Big|_{\alpha = \alpha_k} = 0.
$$

Условия оптимальности:

Точный линейный поиск (метод наискорейшего спуска)

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

Подход скорее теоретический, чем практический: он удобен для анализа сходимости, но точный линейный поиск часто затруднён, если вычисление функции занимает слишком много времени или стоит слишком дорого.

Интересное теоретическое свойство этого метода заключается в том, что градиенты на соседних итерациях ортогональны. Условие оптимальности по α_k даёт

$$
\frac{d}{d\alpha} f(x^k - \alpha \nabla f(x^k))\Big|_{\alpha = \alpha_k} = 0.
$$

Условия оптимальности:

$$
\nabla f(x^{k+1})^\top \nabla f(x^k) = 0
$$

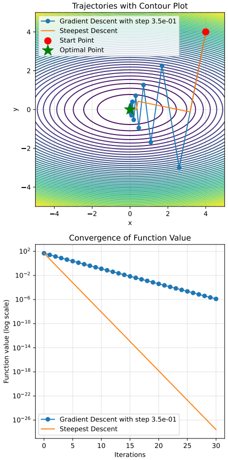


Рис. 59: Наискорейший спуск

Открыть в Colab ♣

Сходимости в гладком выпуклом случае

Theorem

Предположим, что $f : \mathbb{R}^d \to \mathbb{R}$ является **выпуклой и L-гладкой** функцией, для некоторого $L > 0$.
Пусть $(x^k)_{k \in \mathbb{N}}$ — последовательность итераций, сгенерированная методом градиентного спуска из точки x^0 с постоянным шагом α, удовлетворяющим $0 < \alpha \leq \frac{1}{L}$. Тогда для всех $k \in \mathbb{N}$ справедливо:

$$
f(x^k) - f(x) \leq \frac{\|x^0 - x\|^2}{2\alpha k}.
$$

Сходимость в гладком случае с выполнением условия PL

Theorem

Рассмотрим задачу

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

и предположим, что f является **PL-функцией с константой μ и L-гладкой**, для некоторых $L \geq \mu > 0$. Рассмотрим последовательность $(x^k)_{k\in\mathbb{N}}$, сгенерированную методом градиентного спуска из точки x^0 с постоянным шагом α, удовлетворяющим $0 < \alpha \leq \frac{1}{L}$. Пусть $f^* = \min_{x\in\mathbb{R}^d} f(x)$. Тогда:

$$f(x^k) - f^* \leq (1-\alpha\mu)^k (f(x^0) - f^*).$$

Уведомление

Так как для сильно-выпуклой дифференцируемой функции выполняется условие PL, на такой задача сходимость будет такой же.

Код

Примеры: code snippet.