Вводная лекция

Методы оптимизации

Александр Безносиков

Московский физико-технический институт

4 сентября 2025

ФПМИ

Команда курса: лектор

- Безносиков Александр Николаевич
- почта: beznosikov.an@phystech.edu, anbeznosikov@gmail.com
- tg: @abeznosikov

Команда курса: семинаристы

- Андреев Артем Викторович
  tg: @artyomandreyev
- Богданов Александр Иванович
  tg: @d0dos
- Былинкин Дмитрий Андреевич
  tg: @lxstsvund
- Кормаков Георгий Владимирович
  tg: @gkormakov
- Корнилов Никита Максимович
  tg: @Tugnir

Команда курса: лектор и семинаристы

- Моложавенко Александр Александрович
  tg: @MetaMelon
- Ребриков Алексей Витальевич
  tg: @NoblFriend
- Ткаченко Светлана
  tg: @Aikiseito
- Чежегов Савелий Андреевич
  tg: @Savochak
- Хафизов Фанис Адикович
  tg: @faniskhafizov

Команда курса: ассистенты

- Давыденко Григорий
- Иванов Максим
- Левин Леонид
- Максимов Роман
- Парфенова Анна
- Терехова Ольга
- Трифонов Степан
- Чирков Георгий
- Шалыгин Игорь

Правила игры: система оценивания

Вид активности	Баллы
тесты на 10 минут в начале каждого семинара по теме прошлой лекции и прошлого семинара	3
домашнее задание (выдается каждую неделю)	$3 + 3$
контрольная работа в середине семестра по темам лекций и семинаров	3
коллоквиум в конце семестра по темам лекций и семинаров	3
Итого:	15

Правила игры: система оценивания

- Для получения оценки удовлетворительно и выше необходимо, чтобы было выполнено хотя бы одно из следующих условий: оценка за коллоквиум ≥ 1, оценка за КР ≥ 1.
- Для получения оценки хорошо и выше необходимо, чтобы было выполнено оба следующих условия: оценка за коллоквиум ≥ 1, оценка за КР ≥ 1.

Правила игры: комментарии

- Ни один из видов активности не является 100% обязательным, но смотри дополнительные правила выше.
- Тесты будут проводиться на каждом семинаре.
- ДЗ будут появляться в четверг. Время на выполнение: 2 недели
- ДЗ состоит из двух частей: основной и дополнительной. Основная часть легче и предполагается, что ее достаточно для хорошего погружения в курс, дополнительная часть – для более глубокого погружения в заинтересовавшие темы.
- При подозрении в списывании ДЗ баллы за конкретное домашнее задание обнуляются у всех авторов, подозреваемых в списывании (в том числе и у тех, кто дал списать).
- КР состоится 8 ноября в 9:00.

Правила игры: комментарии

- Коллоквиум проходит в конце семестра во время последнего семинара и на зачетной неделе (на выбор). Программа коллоквиума соответствует всей программе курса, изученной в рамках лекций и семинаров. Принимают коллоквиум семинарист и несколько приглашенных преподавателей. Процедура коллоквиума соответствует процедуре проведения обычного устного экзамена на Физтехе с билетами, дополнительными вопросами/задачами и беседой в рамках курса.

Немного истории

- 1847: Коши и градиентный спуск для линейных систем
- 1950ые: линейное программирование (быстро перешло в нелинейное программирование), появление стохастических методов
- 1980ые: появление теории для общих задач.
- 2010ые: задачи оптимизации большого размера, теория стохастических методов

Задача оптимизации

$$\min_{\substack{g_i(x)\&0,\\ i=1,\ldots,m,\\ x\in Q}} f(x)$$

- $Q \subseteq \mathbb{R}^d$ — подмножество d-мерного пространства
- $f : Q \to \mathbb{R}$ — некоторая функция, заданная на множестве Q
- В качестве & берётся $\leq$ либо $=$
- $g_i(x) : Q \to \mathbb{R}$, $i = 1, \ldots, m$ — функции, задающие ограничения

Задача оптимизации

$$\min_{\substack{g_i(x)\&0,\\ i=1,\dots,m,\\ x\in Q}} \quad f(x)$$

- $Q \subseteq \mathbb{R}^d$ — подмножество d-мерного пространства
- $f : Q \to \mathbb{R}$ — некоторая функция, заданная на множестве Q
- В качестве & берётся $\leq$ либо $=$
- $g_i(x) : Q \to \mathbb{R}$, $i = 1, \dots, m$ — функции, задающие ограничения

Вопрос: что можно сказать про эту задач? сложная ли эта задача?

Задачи оптимизации. Первые наблюдения.

1. В общем случае задачи оптимизации могут не иметь решения. Например, задача $\min_{x \in \mathbb{R}} x$ не имеет решения.
2. Задачи оптимизации часто нельзя решить аналитически.
3. Их сложность зависит от вида целевой функции f, множества Q и может зависеть от размерности x.

Задачи оптимизации. Первые наблюдения.

1. В общем случае задачи оптимизации могут не иметь решения. Например, задача $\min_{x \in \mathbb{R}} x$ не имеет решения.
2. Задачи оптимизации часто нельзя решить аналитически.
3. Их сложность зависит от вида целевой функции f, множества Q и может зависеть от размерности x.

Если же задача оптимизации имеет решение, то на практике её обычно решают, вообще говоря, приближённо. Для этого применяются специальные алгоритмы, которые и называют методами оптимизации.

Методы оптимизации

- Нет смысла искать лучший метод для решения конкретной задачи. Например, лучший метод для решений задачи $\min_{x \in \mathbb{R}^d} \|x\|^2$ сходится за 1 итерацию: этот метод просто всегда выдаёт ответ $x^* = 0$. Очевидно, что для других задач такой метод не пригоден.
- Эффективность метода определяется для класса задач, т.к. обычно численные методы разрабатываются для *приближённого* решения множества однотипных задач.
- Метод разрабатывается для класса задач $\Longrightarrow$ метод не может иметь с самого начала полной информации о задаче. Вместо этого метод использует модель задачи, например, формулировку задачи, описание функциональных компонент, множества, на котором происходит оптимизация и т.д.

- Предполагается, что численный метод может накапливать специфическую информацию о задаче при помощи некоторого *оракула*. Под оракулом можно понимать некоторое устройство (программу, процедуру), которое отвечает на последовательные вопросы численного метода.

- Предполагается, что численный метод может накапливать специфическую информацию о задаче при помощи некоторого *оракула*. Под оракулом можно понимать некоторое устройство (программу, процедуру), которое отвечает на последовательные вопросы численного метода.

Вопрос: Какого рода вопросы хочется задавать оракулу?

- Предполагается, что численный метод может накапливать специфическую информацию о задаче при помощи некоторого *оракула*. Под оракулом можно понимать некоторое устройство (программу, процедуру), которое отвечает на последовательные вопросы численного метода.

Вопрос: Какого рода вопросы хочется задавать оракулу?

- Предполагается, что численный метод может накапливать специфическую информацию о задаче при помощи некоторого *оракула*. Под оракулом можно понимать некоторое устройство (программу, процедуру), которое отвечает на последовательные вопросы численного метода.

Вопрос: Какого рода вопросы хочется задавать оракулу?

Примеры оракулов

- **Оракул нулевого порядка** в запрашиваемой точке x возвращает значение целевой функции $f(x)$.
- **Оракул первого порядка** в запрашиваемой точке возвращает значение функции $f(x)$ и её градиент в данной точке $\nabla f(x) = \left(\frac{\partial f(x)}{\partial x_1}, \ldots, \frac{\partial f(x)}{\partial x_n}\right)$.
- **Оракул второго порядка** в запрашиваемой точке возвращает значение и градиент функции $f(x), \nabla f(x)$, а также её гессиан в данной точке $\left(\nabla f^2(x)\right)_{ij} = \frac{\partial^2 f(x)}{\partial x_j \partial x_j}$.

Общая итеративная схема метода оптимизации $\mathcal{M}$

Входные данные: начальная точка x^0 (0 – верхний индекс), требуемая точность решения задачи $\varepsilon > 0$.

Общая итеративная схема метода оптимизации $\mathcal{M}$

Входные данные: начальная точка x^0 (0 – верхний индекс), требуемая точность решения задачи $\varepsilon > 0$.
Настройка. Задать $k = 0$ (счётчик итераций) и $I_{-1} = \varnothing$ (накапливаемая информационная модель решаемой задачи).

Общая итеративная схема метода оптимизации $\mathcal{M}$

Входные данные: начальная точка x^0 (0 – верхний индекс), требуемая точность решения задачи $\varepsilon > 0$.

Настройка. Задать $k = 0$ (счётчик итераций) и $I_{-1} = \varnothing$ (накапливаемая информационная модель решаемой задачи).

Основной цикл

1. Задать вопрос к оракулу $\mathcal{O}$ в точке x^k.

Общая итеративная схема метода оптимизации $\mathcal{M}$

Входные данные: начальная точка x^0 (0 – верхний индекс), требуемая точность решения задачи $\varepsilon > 0$.
Настройка. Задать $k = 0$ (счётчик итераций) и $I_{-1} = \varnothing$ (накапливаемая информационная модель решаемой задачи).
Основной цикл

1. Задать вопрос к оракулу $\mathcal{O}$ в точке x^k.
2. Пересчитать информационную модель: $I_k = I_{k-1} \cup (x^k, \mathcal{O}(x^k))$.

Общая итеративная схема метода оптимизации $\mathcal{M}$

Входные данные: начальная точка x^0 (0 – верхний индекс), требуемая точность решения задачи $\varepsilon > 0$.
Настройка. Задать $k = 0$ (счётчик итераций) и $I_{-1} = \varnothing$ (накапливаемая информационная модель решаемой задачи).
Основной цикл

1. Задать вопрос к оракулу $\mathcal{O}$ в точке x^k.
2. Пересчитать информационную модель: $I_k = I_{k-1} \cup (x^k, \mathcal{O}(x^k))$.
3. Применить правило метода $\mathcal{M}$ для получения новой точки x^{k+1} по модели I_k.

Общая итеративная схема метода оптимизации $\mathcal{M}$

Входные данные: начальная точка x^0 (0 – верхний индекс), требуемая точность решения задачи $\varepsilon > 0$.
Настройка. Задать $k = 0$ (счётчик итераций) и $I_{-1} = \varnothing$ (накапливаемая информационная модель решаемой задачи).
Основной цикл

1. Задать вопрос к оракулу $\mathcal{O}$ в точке x^k.
2. Пересчитать информационную модель: $I_k = I_{k-1} \cup (x^k, \mathcal{O}(x^k))$.
3. Применить правило метода $\mathcal{M}$ для получения новой точки x^{k+1} по модели I_k.
4. Проверить критерий остановки $\mathcal{T}_\varepsilon$. Если критерий выполнен, то выдать ответ $\bar{x}$, иначе положить $k := k + 1$ и вернуться на шаг 1.

Примеры итерационных методов. Градиентный спуск

Рассмотрим задачу оптимизации

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

где функция $f(x)$ дифференцируема. Предположим, что в любой точке мы можем посчитать её градиент.

Примеры итерационных методов. Градиентный спуск

Рассмотрим задачу оптимизации

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

где функция $f(x)$ дифференцируема. Предположим, что в любой точке мы можем посчитать её градиент.

Алгоритм 1 Градиентный спуск с постоянным размером шага

Вход: размер шага $\gamma > 0$, стартовая точка $x^0 \in \mathbb{R}^d$, количество итераций K

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

Выход: x^K

Примеры итерационных методов. Градиентный спуск

Рассмотрим задачу оптимизации

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

где функция $f(x)$ дифференцируема. Предположим, что в любой точке мы можем посчитать её градиент.

Алгоритм 1 Градиентный спуск с постоянным размером шага

Вход: размер шага $\gamma > 0$, стартовая точка $x^0 \in \mathbb{R}^d$, количество итераций K

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

Выход: x^K

Вопрос: в чем Алгоритм 1 отличается от определения общей итеративной схемы?

Примеры итерационных методов. Градиентный спуск

Рассмотрим задачу оптимизации

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

где функция $f(x)$ дифференцируема. Предположим, что в любой точке мы можем посчитать её градиент.

Алгоритм 1 Градиентный спуск с постоянным размером шага

Вход: размер шага $\gamma > 0$, стартовая точка $x^0 \in \mathbb{R}^d$, количество итераций K

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

Выход: x^K

Вопрос: в чем Алгоритм 1 отличается от определения общей итеративной схемы? В итеративной схеме использовался $\mathcal{T}_\varepsilon$.

Критерии останова

- По аргументу: $$\|x^k - x^*\| \leq \varepsilon.$$

  Вопрос: какие проблемы тут видим?

Критерии останова

- По аргументу: $$\|x^k - x^*\| \leq \varepsilon.$$

Вопрос: какие проблемы тут видим?

- x^* – неизвестно, но можно так:
$$\|x^{k+1} - x^k\| \leq \|x^{k+1} - x^*\| + \|x^k - x^*\|.$$

Тогда если $\|x^{k+1} - x^*\| \leq \|x^k - x^*\| \leq \varepsilon$, следует $\|x^{k+1} - x^k\| \leq 2\varepsilon$ (в обратную сторону, очевидно, неверно). $\|x^{k+1} - x^k\| \leq \varepsilon$ – это скорее практический вариант критерия, который работает, если есть понимание (интуиция), что $\|x^k - x^*\| \to 0$.

Критерии останова

- По аргументу: $$\|x^k - x^*\| \le \varepsilon.$$

Вопрос: какие проблемы тут видим?

- x^* – неизвестно, но можно так:
$$\|x^{k+1} - x^k\| \le \|x^{k+1} - x^*\| + \|x^k - x^*\|.$$
Тогда если $\|x^{k+1} - x^*\| \le \|x^k - x^*\| \le \varepsilon$, следует $\|x^{k+1} - x^k\| \le 2\varepsilon$ (в обратную сторону, очевидно, неверно). $\|x^{k+1} - x^k\| \le \varepsilon$ – это скорее практический вариант критерия, который работает, если есть понимание (интуиция), что $\|x^k - x^*\| \to 0$.
- x^* – не уникально. Тогда можно поменять следующий критерий

Критерии останова

- По аргументу: $$\|x^k - x^*\| \leq \varepsilon.$$

  Вопрос: какие проблемы тут видим?

  - x^* – неизвестно, но можно так:
    $$\|x^{k+1} - x^k\| \leq \|x^{k+1} - x^*\| + \|x^k - x^*\|.$$
    Тогда если $\|x^{k+1} - x^*\| \leq \|x^k - x^*\| \leq \varepsilon$, следует $\|x^{k+1} - x^k\| \leq 2\varepsilon$ (в обратную сторону, очевидно, неверно). $\|x^{k+1} - x^k\| \leq \varepsilon$ – это скорее практический вариант критерия, который работает, если есть понимание (интуиция), что $\|x^k - x^*\| \to 0$.
  - x^* – не уникально. Тогда можно поменять следующий критерий

- По функции: $$f(x^k) - f^* \leq \varepsilon.$$

  Часто f^* известно, например, для $f(x) = \|Ax - b\|^2$. На практике можно использовать $|f(x^k) - f(x^{k+1})|$.

Критерии останова

- По аргументу: $$\|x^k - x^*\| \leq \varepsilon.$$

  Вопрос: какие проблемы тут видим?
  - x^* – неизвестно, но можно так:
  $$\|x^{k+1} - x^k\| \leq \|x^{k+1} - x^*\| + \|x^k - x^*\|.$$
  Тогда если $\|x^{k+1} - x^*\| \leq \|x^k - x^*\| \leq \varepsilon$, следует $\|x^{k+1} - x^k\| \leq 2\varepsilon$ (в обратную сторону, очевидно, неверно). $\|x^{k+1} - x^k\| \leq \varepsilon$ – это скорее практический вариант критерия, который работает, если есть понимание (интуиция), что $\|x^k - x^*\| \to 0$.
  - x^* – не уникально. Тогда можно поменять следующий критерий
- По функции: $$f(x^k) - f^* \leq \varepsilon.$$

  Часто f^* известно, например, для $f(x) = \|Ax - b\|^2$. На практике можно использовать $|f(x^k) - f(x^{k+1})|$.
- По норме градиента: $$\|\nabla f(x^k)\| \leq \varepsilon.$$

  Вопрос: когда такой критерий можно использовать?

Критерии останова

- По аргументу: $$\|x^k - x^*\| \leq \varepsilon.$$

  Вопрос: какие проблемы тут видим?

  - x^* – неизвестно, но можно так:
    $$\|x^{k+1} - x^k\| \leq \|x^{k+1} - x^*\| + \|x^k - x^*\|.$$
    Тогда если $\|x^{k+1} - x^*\| \leq \|x^k - x^*\| \leq \varepsilon$, следует $\|x^{k+1} - x^k\| \leq 2\varepsilon$ (в обратную сторону, очевидно, неверно). $\|x^{k+1} - x^k\| \leq \varepsilon$ – это скорее практический вариант критерия, который работает, если есть понимание (интуиция), что $\|x^k - x^*\| \to 0$.
  - x^* – не уникально. Тогда можно поменять следующий критерий
- По функции: $$f(x^k) - f^* \leq \varepsilon.$$

  Часто f^* известно, например, для $f(x) = \|Ax - b\|^2$. На практике можно использовать $|f(x^k) - f(x^{k+1})|$.
- По норме градиента: $$\|\nabla f(x^k)\| \leq \varepsilon.$$

  Вопрос: когда такой критерий можно использовать? В безусловной оптимизации

Сложность методов оптимизации

- **Аналитическая/Оракульная сложность** — число обращений к оракулу, необходимое для решения задачи с точностью ε.
- **Арифметическая/Временна'я сложность** — общее число вычислений (включая работу оракула), необходимых для решения задачи с точностью ε.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

$$\min_{x \in B_d} f(x)$$

- $B_d = \{x \in \mathbb{R}^d \mid 0 \leq x_i \leq 1, \quad i = 1, \ldots, d\}$
- Функция $f(x)$ является M-липшицевой на B_d относительно ℓ_∞-нормы:

$$\forall x, y \; |f(x) - f(y)| \leq M\|x - y\|_\infty = M \max_{i=1,\ldots,d} |x_i - y_i|.$$

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Наблюдение

Множество B_d является ограниченным и замкнутым, т.е. компактом, а из липшицевости функции f следует и её непрерывность, поэтому задача (22) имеет решение, ибо непрерывная на компакте функция достигает своих минимального и максимального значений. Пусть $f^* = \min_{x \in B_d} f(x)$.

- **Класс методов.** Для данной задачи рассмотрим методы нулевого порядка.
- **Цель:** найти $\bar{x} \in B_d$: $f(\bar{x}) - f^* \leq \varepsilon$.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Рассмотрим один из самых простых способов решения этой задачи — метод равномерного перебора.

Алгоритм 2 Метод равномерного перебора

Вход: целочисленный параметр перебора $p \geq 1$

1: Сформировать $(p+1)^d$ точек вида $x_{(i_1,\ldots,i_d)} = \left(\frac{i_1}{p}, \frac{i_2}{p}, \ldots, \frac{i_d}{p}\right)^\top$, где $(i_1, \ldots, i_d) \in \{0, 1, \ldots, p\}^d$

2: Среди точек $x_{(i_1,\ldots,i_d)}$ найти точку $\bar{x}$ с наименьшим значением целевой функции f.

Выход: $\bar{x}, f(\bar{x})$

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Теорема 1

Алгоритм 2 с параметром p возвращает такую точку $\bar{x}$, что

$$f(\bar{x}) - f^* \leq \frac{M}{2p},$$

откуда следует, что методу равномерного перебора нужно в худшем случае

$$\left(\left\lfloor \frac{M}{2\varepsilon} \right\rfloor + 2 \right)^d$$

обращений к оракулу, чтобы гарантировать $f(\bar{x}) - f^* \leq \varepsilon$.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Доказательство Теоремы 1

Пусть x^* — решение задачи (точка минимума функции f). Тогда в построенной «сетке» из точек найдётся такая точка $x_{(i_1,\ldots,i_d)}$, что $x := x_{(i_1,\ldots,i_d)} \leq x^* \leq x_{(i_1+1,\ldots,i_d+1)} =: y$, где знак «$\leq$» применяется покомпонентно.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Доказательство Теоремы 1

Пусть x^* — решение задачи (точка минимума функции f). Тогда в построенной «сетке» из точек найдётся такая точка $x_{(i_1,\ldots,i_d)}$, что $x := x_{(i_1,\ldots,i_d)} \leq x^* \leq x_{(i_1+1,\ldots,i_d+1)} =: y$, где знак «$\leq$» применяется покомпонентно. Во-первых, $y_i - x_i = \frac{1}{p}$ и $x_i^* \in [x_i, y_i]$ для всех $i = 1, \ldots, d$.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Доказательство Теоремы 1

Пусть x^* — решение задачи (точка минимума функции f). Тогда в построенной «сетке» из точек найдётся такая точка $x_{(i_1,\ldots,i_d)}$, что $x := x_{(i_1,\ldots,i_d)} \leq x^* \leq x_{(i_1+1,\ldots,i_d+1)} =: y$, где знак «$\leq$» применяется покомпонентно. Во-первых, $y_i - x_i = \frac{1}{p}$ и $x_i^* \in [x_i, y_i]$ для всех $i = 1, \ldots, d$. Кроме того, рассмотрим точки $\hat{x}$ и $\tilde{x}$ такие, что $\hat{x} = \frac{x+y}{2}$ и

$$\tilde{x}_i = \begin{cases} y_i, & \text{если } x_i^* \geq \hat{x}_i, \\ x_i, & \text{иначе.} \end{cases}$$

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Доказательство Теоремы 1 (продолжение)

Заметим, что $\tilde{x}$ принадлежит «сетке» и $|\tilde{x}_i - x_i^*| \leq \frac{1}{2p}$, а значит, $\|\tilde{x} - x^*\|_\infty \leq \frac{1}{2p}$.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Доказательство Теоремы 1 (продолжение)

Заметим, что $\tilde{x}$ принадлежит «сетке» и $|\tilde{x}_i - x_i^*| \leq \frac{1}{2p}$, а значит, $\|\tilde{x} - x^*\|_\infty \leq \frac{1}{2p}$. Поскольку $f(\bar{x}) \leq f(\tilde{x})$ (по определению), получаем

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Доказательство Теоремы 1 (продолжение)

Заметим, что $\tilde{x}$ принадлежит «сетке» и $|\tilde{x}_i - x_i^*| \leq \frac{1}{2p}$, а значит, $\|\tilde{x} - x^*\|_\infty \leq \frac{1}{2p}$. Поскольку $f(\bar{x}) \leq f(\tilde{x})$ (по определению), получаем

$$f(\bar{x}) - f^* \leq f(\tilde{x}) - f^* \leq M\|\tilde{x} - x^*\|_\infty \leq \frac{M}{2p}.$$

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Доказательство Теоремы 1 (продолжение)

Заметим, что $\tilde{x}$ принадлежит «сетке» и $|\tilde{x}_i - x_i^*| \leq \frac{1}{2p}$, а значит, $\|\tilde{x} - x^*\|_\infty \leq \frac{1}{2p}$. Поскольку $f(\bar{x}) \leq f(\tilde{x})$ (по определению), получаем

$$f(\bar{x}) - f^* \leq f(\tilde{x}) - f^* \leq M\|\tilde{x} - x^*\|_\infty \leq \frac{M}{2p}.$$

Выписанная выше оценка достигается методом равномерного перебора за $(p+1)^d$ обращений к оракулу. Следовательно, чтобы гарантировать $f(\bar{x}) - f^* \leq \varepsilon$, необходимо взять $p = \left\lfloor \frac{M}{2\varepsilon} \right\rfloor + 1$, т.е. метод сделает $\left(\left\lfloor \frac{M}{2\varepsilon} \right\rfloor + 2\right)^d$ обращений к оракулу.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Вопрос: хороший результат получили или нет?

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Вопрос: хороший результат получили или нет?

- Предположим $M = 2$, $d = 13$ И $\varepsilon = 0.01$, то есть размерность задачи сравнительно небольшая и точность решения задачи не слишком высокая.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Вопрос: хороший результат получили или нет?

- Предположим $M = 2$, $d = 13$ И $\varepsilon = 0.01$, то есть размерность задачи сравнительно небольшая и точность решения задачи не слишком высокая.
- Необходимое число обращений к оракулу: $\left(\left\lfloor \frac{M}{2\varepsilon} \right\rfloor + 2\right)^d = 102^{13} > 10^{26}$.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Вопрос: хороший результат получили или нет?

- Предположим $M = 2$, $d = 13$ И $\varepsilon = 0.01$, то есть размерность задачи сравнительно небольшая и точность решения задачи не слишком высокая.
- Необходимое число обращений к оракулу: $\left(\left\lfloor \frac{M}{2\varepsilon} \right\rfloor + 2\right)^d = 102^{13} > 10^{26}$.
- Сложность одного вызова оракула не менее 1, но если потребовать, чтобы он обязательно считал, переданную ему точки, то сложность не менее d операции.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Вопрос: хороший результат получили или нет?

- Предположим $M = 2$, $d = 13$ И $\varepsilon = 0.01$, то есть размерность задачи сравнительно небольшая и точность решения задачи не слишком высокая.
- Необходимое число обращений к оракулу: $\left(\left\lfloor \frac{M}{2\varepsilon} \right\rfloor + 2\right)^d = 102^{13} > 10^{26}$.
- Сложность одного вызова оракула не менее 1, но если потребовать, чтобы он обязательно считал, переданную ему точки, то сложность не менее d операции.
- Производительность компьютера: 10^{11} арифметических операций в секунду.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Вопрос: хороший результат получили или нет?

- Предположим $M = 2$, $d = 13$ И $\varepsilon = 0.01$, то есть размерность задачи сравнительно небольшая и точность решения задачи не слишком высокая.
- Необходимое число обращений к оракулу: $\left(\left\lfloor \frac{M}{2\varepsilon} \right\rfloor + 2\right)^d = 102^{13} > 10^{26}$.
- Сложность одного вызова оракула не менее 1, но если потребовать, чтобы он обязательно считал, переданную ему точки, то сложность не менее d операции.
- Производительность компьютера: 10^{11} арифметических операций в секунду.
- Общее время: хотя бы 10^{15} секунд, что больше 30 миллионов лет.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

- **Вопрос:** что мы сейчас получили? верхнюю или нижнюю оценку? что такое верхняя оценка?

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

- **Вопрос:** что мы сейчас получили? верхнюю или нижнюю оценку? что такое верхняя оценка?
- **Верхняя оценка** – гарантии нахождения решения определенным методом из рассматриваемого класса методов (например, методы с оракулом нулевого порядка) для любой задачи из класса (липшецева целевая функция).
- **Нижняя оценка** – гарантия, что существует «плохая» задача из класса, что любой метод из класса методов будет сходиться не лучше утверждает нижняя оценка.
- Возникает вопрос: может мы плохо вывели верхнюю оценку (неидеальный анализ), может ли предложить другой метод из рассматриваемого класса, который будет находить приближённое решение существенно быстрее? На этот вопрос и даст ответ нижняя оценка.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Теорема 2

Пусть $\varepsilon < \frac{M}{2}$. Тогда аналитическая сложность описанного класса задач, т.е. аналитическая сложность метода на «худшей» для него задаче из данного класса, составляет по крайней мере

$$\left(\left\lfloor \frac{M}{2\varepsilon} \right\rfloor \right)^d \quad \text{вызовов оракула.}$$

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Схема доказательства Теоремы 2

Пусть $p = \left\lfloor \frac{M}{2\varepsilon} \right\rfloor$. Доказываем от противного: предположим, что существует такой метод, который решает задачу за $N < (p^d - 1)$ обращений к оракулу, чтобы решить задачу с точностью ε (по функции).

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Схема доказательства Теоремы 2

Пусть $p = \left\lfloor \frac{M}{2\varepsilon} \right\rfloor$. Доказываем от противного: предположим, что существует такой метод, который решает задачу за $N < (p^d - 1)$ обращений к оракулу, чтобы решить задачу с точностью ε (по функции). Построим такую функцию, на которой метод не сможет найти ε-решение, при помощи сопротивляющегося оракула: пусть изначально наша целевая функция $f(x)$ всюду равна 0.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Схема доказательства Теоремы 2

Пусть $p = \left\lfloor \frac{M}{2\varepsilon} \right\rfloor$. Доказываем от противного: предположим, что существует такой метод, который решает задачу за $N < (p^d - 1)$ обращений к оракулу, чтобы решить задачу с точностью ε (по функции). Построим такую функцию, на которой метод не сможет найти ε-решение, при помощи сопротивляющегося оракула: пусть изначально наша целевая функция $f(x)$ всюду равна 0. Запустим метод, он запросит значение f в N точках, везде получит 0 и выдаст какую-то точку (возможно, отличную от всех предыдущих N, как ответ). В итоге мы в ходе работы алгоритма заглянули в $N + 1 < p^d$ точку.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Схема доказательства Теоремы 2

Пусть $p = \left\lfloor \frac{M}{2\varepsilon} \right\rfloor$. Доказываем от противного: предположим, что существует такой метод, который решает задачу за $N < (p^d - 1)$ обращений к оракулу, чтобы решить задачу с точностью ε (по функции). Построим такую функцию, на которой метод не сможет найти ε-решение, при помощи сопротивляющегося оракула: пусть изначально наша целевая функция $f(x)$ всюду равна 0. Запустим метод, он запросит значение f в N точках, везде получит 0 и выдаст какую-то точку (возможно, отличную от всех предыдущих N, как ответ). В итоге мы в ходе работы алгоритма заглянули в $N + 1 < p^d$ точку. Тогда по принципу Дирихле найдётся такой «кубик» $B = \{x \mid \hat{x} \preceq x \preceq \hat{x} + \frac{1}{p} e\}$ (где $\hat{x}$ и $\hat{x} + \frac{1}{p} e$ — точки из «сетки» с шагом p, e — вектор из единиц), который не содержит ни одной из $N + 1$ точки (в том числе и выхода метода).

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Схема доказательства Теоремы 2 (продолжение)

Пусть x^* — это центр «кубика» B, т.е. $x^* = \hat{x} + \frac{1}{2p}e$.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Схема доказательства Теоремы 2 (продолжение)

Пусть x^* — это центр «кубика» B, т.е. $x^* = \hat{x} + \frac{1}{2p}e$. Немного модифицируем функцию $\bar{f}(x) = \min\{0, M\|x - x^*\|_\infty - \varepsilon\}$.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Схема доказательства Теоремы 2 (продолжение)

Пусть x^* — это центр «кубика» B, т.е. $x^* = \hat{x} + \frac{1}{2p}e$. Немного модифицируем функцию $\bar{f}(x) = \min\{0, M\|x - x^*\|_\infty - \varepsilon\}$. Функция $\bar{f}(x)$ липшицева с константой M относительно ℓ_∞-нормы и принимает своё минимальное значение $-\varepsilon$ в точке x^*.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Схема доказательства Теоремы 2 (продолжение)

Пусть x^* — это центр «кубика» B, т.е. $x^* = \hat{x} + \frac{1}{2p}e$. Немного модифицируем функцию $\bar{f}(x) = \min\{0, M\|x - x^*\|_\infty - \varepsilon\}$. Функция $\bar{f}(x)$ липшицева с константой M относительно ℓ_∞-нормы и принимает своё минимальное значение $-\varepsilon$ в точке x^*. Более того, функция $\bar{f}(x)$ отлична от нуля только внутри куба $B' = \{x \mid \|x - x^*\| \leq \frac{\varepsilon}{M}\}$, который лежит внутри куба B, т.к. $2p \leq \frac{M}{\varepsilon}$.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Схема доказательства Теоремы 2 (продолжение)

Пусть x^* — это центр «кубика» B, т.е. $x^* = \hat{x} + \frac{1}{2p}e$. Немного модифицируем функцию $\bar{f}(x) = \min\{0, M\|x - x^*\|_\infty - \varepsilon\}$. Функция $\bar{f}(x)$ липшицева с константой M относительно ℓ_∞-нормы и принимает своё минимальное значение $-\varepsilon$ в точке x^*. Более того, функция $\bar{f}(x)$ отлична от нуля только внутри куба $B' = \{x \mid \|x - x^*\| \leq \frac{\varepsilon}{M}\}$, который лежит внутри куба B, т.к. $2p \leq \frac{M}{\varepsilon}$. Следовательно, рассмотренный метод на данной функции не может найти ε-решение. Противоречие.

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Схема доказательства Теоремы 2 (продолжение)

Пусть x^* — это центр «кубика» B, т.е. $x^* = \hat{x} + \frac{1}{2p}e$. Немного модифицируем функцию $\bar{f}(x) = \min\{0, M\|x - x^*\|_\infty - \varepsilon\}$. Функция $\bar{f}(x)$ липшицева с константой M относительно ℓ_∞-нормы и принимает своё минимальное значение $-\varepsilon$ в точке x^*. Более того, функция $\bar{f}(x)$ отлична от нуля только внутри куба $B' = \{x \mid \|x - x^*\| \leq \frac{\varepsilon}{M}\}$, который лежит внутри куба B, т.к. $2p \leq \frac{M}{\varepsilon}$. Следовательно, рассмотренный метод на данной функции не может найти ε-решение. Противоречие.

Итак, в указанном классе у любого метода оценки на скорость сходимости весьма пессимистичные. Возникает вопрос: какие свойства нужно потребовать от класса оптимизируемых функций, чтобы оценки стали более оптимистичными?

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Схема доказательства Теоремы 2 (продолжение)

Пусть x^* — это центр «кубика» B, т.е. $x^* = \hat{x} + \frac{1}{2p}e$. Немного модифицируем функцию $\bar{f}(x) = \min\{0, M\|x - x^*\|_\infty - \varepsilon\}$. Функция $\bar{f}(x)$ липшицева с константой M относительно ℓ_∞-нормы и принимает своё минимальное значение $-\varepsilon$ в точке x^*. Более того, функция $\bar{f}(x)$ отлична от нуля только внутри куба $B' = \{x \mid \|x - x^*\| \leq \frac{\varepsilon}{M}\}$, который лежит внутри куба B, т.к. $2p \leq \frac{M}{\varepsilon}$. Следовательно, рассмотренный метод на данной функции не может найти ε-решение. Противоречие.

Итак, в указанном классе у любого метода оценки на скорость сходимости весьма пессимистичные. Возникает вопрос: какие свойства нужно потребовать от класса оптимизируемых функций, чтобы оценки стали более оптимистичными? Вернемся к этим вопросам на следующей лекции.