

Вводная лекция

Методы оптимизации

Александр Безносиков

Московский физико-технический институт

7 сентября 2023



Команда курса: лектор

- Безносиков Александр Николаевич
- кандидат физико-математических наук
- 10+ статей по оптимизации на A^*/A конференции или в Q1 журналах в области ML; более 30 статей всего
- заведующий лабораторией МФТИ-Яндекс, исследователь в Иннополисе, Сколтехе, ВШЭ, Яндексе, ИППИ РАН, MBZUAI (ОАЭ)
- сайт: anbeznosikov.github.io
- почта: beznosikov.an@phystech.edu, anbeznosikov@gmail.com
- tg: @abeznosikov

Команда курса: семинаристы

- Бородич Екатерина Дмитриевна – 6 статей (2 в A^*/A и Q1)
почта: borodich.ed@phystech.edu tg: @e_borodich
- Кормаков Георгий Владимирович – 2 статьи, большой опыт в DS
почта: Georgiy.Kormakov@skoltech.ru tg: @gkormakov
- Корнилов Никита Максимович – 2 статьи
почта: kornilov.nm@phystech.edu tg: @Tugnir
- Курузов Илья Алексеевич – 2 статьи, несколько промышленных проектов
почта: kuruzov.ia@phystech.edu tg: @EliasKur
- Лобанов Александр Владимирович – 10 статей
почта: lobbsasha@mail.ru

Команда курса: лектор и семинаристы

- Метелев Дмитрий Сергеевич – 3 статьи (одна из них на A*)
почта: metelev.ds@phystech.edu tg: @FlyBot239
- Моложавенко Александр Александрович
почта: molozhavenko.aa@phystech.edu tg: @MetaMelon
- Чежегов Савелий Андреевич – 4 статьи
почта: chezhegov.sa@phystech.edu tg: @Savochak
- Юдин Никита Евгеньевич – 2 статьи
почта: iudin.ne@phystech.edu tg: @nikitayudin
- Ярмошик Демьян Валерьевич – 8 статей
почта: yarmoshik.dv@phystech.edu tg: @niquepolice

Команда курса: ассистенты

- Андреев Артем – 1 статья
почта: andreev.artem@phystech.edu tg: @artyomandreyev
- Былинкин Дмитрий – 1 статья
почта: bylinkin.da@phystech.edu tg: @lxstsvund
- Богданов Александр – 1 статья
почта: bogdanov.ai@phystech.edu tg: @d0dos
- Рябинин Евгений
почта: riabinin.ev@phystech.edu tg: @REV757
- Ткаченко Светлана – 1 статья
почта: tkachenko.s@phystech.edu tg: @Aikiseito
- Шестаков Александр – 1 статья
почта: shestakov.aa@phystech.edu tg: @brigada_al

Telegram беседа



Диск с материалами



Форма обратной связи



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 каждые 2 недели)	3
контрольная работа в середине семестра по темам лекций и семинаров	3
коллоквиум в конце семестра по темам лекций и семинаров	3
разбор статьи	2
Итого:	14

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 в начале каждого семинара.
- Обновленные конспекты и ДЗ будут появляться на диске в выходные (ДЗ раз в 2 недели).
- 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 проводится в середине семестра на лекции. Программа контрольной соответствует программе курса, изученной в рамках лекций и семинаров за 8 недель.

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.
- Разбор статьи предполагает полный разбор популярной статьи в области оптимизации. Необходимо разобраться в идее, в доказательствах, а также воспроизвести численные эксперименты, добавив к ним свои. Отчетность состоит из двух пунктов: 15 минутное выступление с презентацией на семинарском занятии, а также часовой доклад на семинаре лабораторий МФТИ-Яндекс и ММО. Заявки на разбор статьи принимаются до 31 октября. ☰ 🔍 ↺

Задача оптимизации

- **Вопрос:** какие приложения задач оптимизации знаете/уже встречали?

Задача оптимизации

- **Вопрос:** какие приложения задач оптимизации знаете/уже встречали? Приложений у оптимизации масса: машинное обучение, анализ данных, статистика, финансы, логистика, планирование, управление и многие другие.

Задача оптимизации

- **Вопрос:** какие приложения задач оптимизации знаете/уже встречали? Приложений у оптимизации масса: машинное обучение, анализ данных, статистика, финансы, логистика, планирование, управление и многие другие.
- Оптимизация часто выступает инструментом во многих прикладных задачах, люди пользуются готовыми решениями/пакетами/"черными ящиками", который способны решать задачи оптимизации. Цель курса – как познакомиться с такими "черными ящиками", так и заглянуть внутрь и понять, что лежит внутри (в том числе с точки зрения теории).

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

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

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

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

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 $I_{-1} = \emptyset$
(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 $I_{-1} = \emptyset$
(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 $I_{-1} = \emptyset$
(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 $I_{-1} = \emptyset$ (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}_ε . 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

$$\min_{x \in \mathbb{R}^d} f(x), \quad (2)$$

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

Примеры итерационных методов. Градиентный спуск

Рассмотрим задачу оптимизации

$$\min_{x \in \mathbb{R}^d} f(x), \quad (2)$$

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** $k = 0, 1, \dots, K - 1$ **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

$$\min_{x \in \mathbb{R}^d} f(x), \quad (2)$$

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** $k = 0, 1, \dots, K - 1$ **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

$$\min_{x \in \mathbb{R}^d} f(x), \quad (2)$$

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** $k = 0, 1, \dots, K - 1$ **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}_ϵ .

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

$$\|x^{k+1} - x^k\| \leq \|x^{k+1} - x^*\| + \|x^k - x^*\| \leq 2\varepsilon.$$

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

- x^* – не уникально. Тогда можно поменять 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

$$\|x^{k+1} - x^k\| \leq \|x^{k+1} - x^*\| + \|x^k - x^*\| \leq 2\varepsilon.$$

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

- x^* – не уникально. Тогда можно поменять 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/Временная сложность** — общее число вычислений (включая работу оракула), необходимых для решения задачи с точностью ε .

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

$$\min_{x \in B_d} f(x) \quad (3)$$

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

$$\forall x, y \quad |f(x) - f(y)| \leq M \|x - y\|_\infty = M \max_{i=1, \dots, d} |x_i - y_i|. \quad (4)$$

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 (3) 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 $x_{(i_1, \dots, i_d)} = \left(\frac{i_1}{p}, \frac{i_2}{p}, \dots, \frac{i_d}{p} \right)^T$, где $(i_1, \dots, i_d) \in \{0, 1, \dots, p\}^d$
- 2: Среди точек $x_{(i_1, \dots, i_d)}$ 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

$$f(\bar{x}) - f^* \leq \frac{M}{2p}, \quad (5)$$

откуда следует, что методу равномерного перебора нужно в худшем случае

$$\left(\left|\frac{M}{2\varepsilon}\right| + 2\right)^d \quad (6)$$

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 $x_{(i_1, \dots, i_d)}$, что $x := x_{(i_1, \dots, i_d)} \leq x^* \leq x_{(i_1+1, \dots, i_d+1)} =: y$, 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 $x_{(i_1, \dots, i_d)}$, что $x := x_{(i_1, \dots, i_d)} \leq x^* \leq x_{(i_1+1, \dots, i_d+1)} =: y$, где знак « \leq » применяется покомпонентно. Во-первых, $y_i - x_i = \frac{1}{p}$ и $x_i^* \in [x_i, y_i]$ для всех $i = 1, \dots, d$.

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 $x_{(i_1, \dots, i_d)}$, что $x := x_{(i_1, \dots, i_d)} \leq x^* \leq x_{(i_1+1, \dots, i_d+1)} =: y$, где знак « \leq » применяется покомпонентно. Во-первых, $y_i - x_i = \frac{1}{p}$ и $x_i^* \in [x_i, y_i]$ для всех $i = 1, \dots, d$. 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 $p = \lfloor \frac{M}{2\varepsilon} \rfloor + 1$, т.е. метод сделает $(\lfloor \frac{M}{2\varepsilon} \rfloor + 2)^d$ 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 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 определённым методом из рассматриваемого класса методов (например, методы с оракулом нулевого порядка) для любой задачи из класса (липшицева 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

$$\left(\left\lfloor \frac{M}{2\varepsilon} \right\rfloor \right)^d \text{ вызовов оракула.} \quad (7)$$

Сложность задач оптимизации. Класс задач минимизации липшицевых функций

Схема доказательства Теоремы 2

Пусть $p = \lfloor \frac{M}{2\varepsilon} \rfloor$. 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 $p = \lfloor \frac{M}{2\varepsilon} \rfloor$. 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 $p = \lfloor \frac{M}{2\varepsilon} \rfloor$. 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 $p = \lfloor \frac{M}{2\varepsilon} \rfloor$. 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. $x^* = \hat{x} + \frac{1}{2^p}e$. 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.

Скорости сходимости/приближения к решению

- Сублинейная:

$$\|x^k - x^*\| \leq \frac{C}{k^\alpha},$$

где $\alpha > 0$, $C > 0$ – константа.

- Линейная:

$$\|x^k - x^*\| \leq Cq^k,$$

где $q \in (0; 1)$, $C > 0$ – константа.

- Сверхлинейная:

$$\|x^k - x^*\| \leq Cq^{k^p},$$

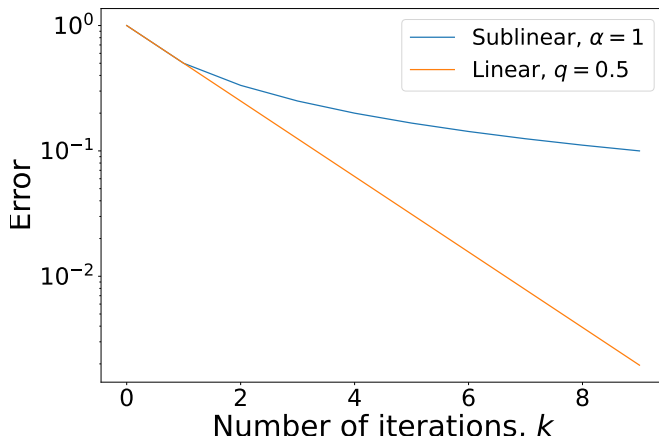
где $q \in (0; 1)$, $p > 1$, $C > 0$ – константа.

- Квадратичная:

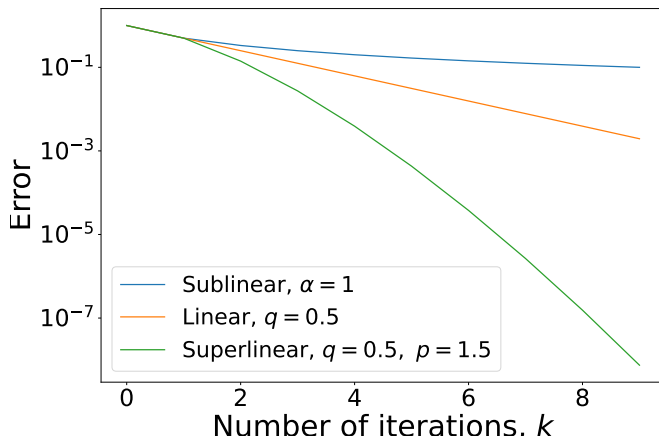
$$\|x^k - x^*\| \leq Cq^{2^k} \quad \text{или} \quad \|x^{k+1} - x^*\| \leq C\|x^k - x^*\|^2,$$

где $q \in (0; 1)$, $C > 0$ – константа.

Скорости сходимости/приближения к решению



Скорости сходимости/приближения к решению



Скорости сходимости/приближения к решению

