

Методы оптимизации

Михайлов Максим

6 мая 2021 г.

Оглавление

| | | |
|-----------------|-----------------------------------------------------------------------------|-----------|
| Лекция 1 | 10 февраля | 4 |
| 1 | Теория погрешности | 4 |
| 2 | Задачи оптимизации. Вводное. | 8 |
| 3 | Одномерная минимизация функций. Прямые методы. | 9 |
| 3.1 | Метод дихотомии | 9 |
| Лекция 2 | 17 февраля | 11 |
| 3.2 | Метод золотого сечения | 11 |
| 3.3 | Метод Фибоначчи | 12 |
| 3.4 | Метод парабол | 13 |
| 3.5 | Комбинированный метод Брента | 14 |
| Лекция 3 | 24 февраля | 15 |
| 3.6 | Метод равномерного перебора | 15 |
| 4 | Методы оптимизации, использующие производную | 15 |
| 4.1 | Методы средней точки | 15 |
| 4.2 | Метод хорд (<i>метод секущей</i>) | 16 |
| 4.3 | Метод Ньютона (<i>метод касательной</i>) | 17 |
| Лекция 4 | 3 марта | 18 |
| 4.3.1 | Достаточное условие монотонной сходимости метода Ньютона | 19 |
| 4.4 | Модификации метода Ньютона | 20 |
| 4.4.1 | Метод Ньютона-Рафсона | 20 |
| 4.4.2 | Метод Марквардта | 20 |
| 5 | Метод минимизации многомодальных функций (<i>метод ломаных</i>) | 20 |
| Лекция 5 | 10 марта | 22 |
| 6 | Минимизация функций многих переменных | 22 |
| 6.1 | Постановка задачи | 22 |
| 6.2 | Свойства выпуклых множеств и выпуклых функций | 24 |
| 6.3 | Необходимое и достаточное условие безусловного экстремума | 25 |
| 6.3.1 | Необходимое условие экстремума первого порядка | 25 |
| 6.3.2 | Необходимое условие экстремума второго порядка | 25 |
| 6.3.3 | Достаточное условие экстремума | 25 |
| 6.3.4 | Проверка выполнений условий экстремума | 25 |
| Лекция 6 | 17 марта | 26 |
| 6.3.5 | Критерии Сильвестра проверки достаточных условий экстремума | 26 |
| 6.3.6 | Критерии Сильвестра проверки необходимых условий экстремума | 26 |

| | | |
|-----------|-------------------------------------------------------------------------------------|----|
| 6.4 | Квадратичные функции | 26 |
| 6.5 | Общие принципы многомерной оптимизации | 28 |
| 6.6 | Скорость сходимости минимизирующей последовательности | 28 |
| Лекция 7 | 22 марта (<i>дополнительная лекция</i>) | 30 |
| 6.7 | Метод градиентного спуска | 31 |
| 6.8 | Метод наискорейшего спуска | 33 |
| Лекция 8 | 24 марта | 35 |
| 6.9 | Метод сопряженных градиентов | 35 |
| 6.10 | Метод стохастического градиентного спуска | 36 |
| 6.10.1 | Adagrad | 36 |
| 6.11 | Метод покоординатного спуска | 37 |
| Лекция 9 | 31 марта | 38 |
| 7 | Форматы хранения матриц | 38 |
| 7.1 | Диагональный формат | 38 |
| 7.2 | Ленточный формат | 39 |
| 7.3 | Профильный формат | 40 |
| 7.4 | Разреженный формат | 41 |
| Лекция 10 | 7 апреля | 43 |
| 8 | Решение СЛАУ. Метод Гаусса. | 44 |
| 8.1 | Модификация метода Гаусса (<i>постолбцовый выбор главного элемента</i>) | 46 |
| Лекция 11 | 14 апреля | 47 |
| 9 | LU -метод | 47 |
| 9.1 | Алгоритм разложения | 48 |
| 10 | Дополнительные рассуждения о точности получаемого численного решения | 49 |
| 10.1 | Близкие к нулю главные элементы | 49 |
| 10.2 | Вектор ошибки и невязка | 49 |
| 10.2.1 | Векторные нормы | 50 |
| Лекция 12 | 21 апреля | 52 |
| 10.2.2 | Нормы и анализ ошибок | 52 |
| 10.2.3 | Оценивание числа обусловленности | 53 |
| 11 | Дополнительно о градиентных методах | 53 |
| 11.1 | Метод градиентного спуска | 54 |

Лекция 1

10 февраля

Этот курс — о минимизации (*максимизации*) функционалов. Кроме конкретных методов оптимизации, планируется рассмотреть форматы хранения матриц, о методах работы с ними и рассмотреть 1-2 (*может быть 3*) СЛАУ с использованием различных форматов.

Т.к. значения, получаемые компьютерами — не точные, нам требуется теория погрешности.

1 Теория погрешности

Все погрешности разделяются на два класса:

1. Неустраняемая — обусловлена неточностью исходных данных. Например, неточное знание физических констант или других параметров задачи. Тем не менее, необходимо знать эту погрешность, чтобы ставить рамки погрешности для решения.
2. Устранимая — погрешность процесса решения задачи. Эту погрешность можно уменьшить выбором метода решения задачи.

(a) Погрешность модели

(b) Остаточная погрешность (*погрешность аппроксимации*)

Например, аппроксимация ряда первыми n его членами или аппроксимация по теореме Вейерштрасса квадратичной функцией.

(c) Погрешность округления

(d) Накапливаемая погрешность

2c и **2d** часто объединяют в вычислительную погрешность.

Определение. Пусть X^* — точное решение, а X — найденное (*приближенное*) решение. Тогда $X^* - X$ называется **погрешностью**, а её модуль $\Delta X = |X^* - X|$ — **абсолютная погрешность**.

Разумеется, ΔX представляет сугубо теоретический интерес, т.к. X^* неизвестна и ΔX нельзя вычислить.

Определение. В качестве требования к решению часто предоставляется **предельная абсолютная погрешность** $\Delta_X \geq |X^* - X|$.

Определение. Также существует **относительная погрешность** $\delta X = \left| \frac{X^* - X}{|X|} \right|$

Относительная погрешность позволяет выражать погрешность относительно значений самой величины. Например, при измерении длины парты погрешность 1 см не очень хорошо, а при измерении расстояния между городами — приемлемо.

Определение. **Предельная относительная погрешность** $\delta_X \geq \left| \frac{X^* - X}{|X|} \right|$

Определение. **Значащие цифры** некоторого числа — все цифры в его изображении, отличные от нуля, а также нули, если они содержатся между значащими цифрами или расположены в конце числа и указывают на сохранение разряда точности.

Определение. Если значащая цифра приближенного значения a , находящаяся в разряде, в котором выполняется условие $\Delta \leq 0.5 \cdot 10^k$, т.е. абсолютное значение погрешности не превосходит половину единицы этого разряда (k — номер этого разряда), то такая цифра называется **верной в узком смысле**.

Цифра называется **верной в широком смысле**, если в определении выше используется 1 вместо 0.5.

Пример. $a = 3.635$, $\Delta a = 0.003$

- $k = 0 \quad \frac{1}{2} \cdot 10^0 = \frac{1}{2} \geq \Delta a$
- $k = -1 \quad \frac{1}{2} \cdot 10^{-1} = 0.05 \geq \Delta a$
- $k = -2 \quad \frac{1}{2} \cdot 10^{-2} = 0.005 \geq \Delta a$
- $k = -3 \quad \frac{1}{2} \cdot 10^{-3} = 0.0005 < \Delta a$

Таким образом, цифра 5 является сомнительной, остальные — верные.

Пример. Рассмотрим следующие способы записи одного и того же выражения:

$$\left(\frac{\sqrt{2}-1}{\sqrt{2}+1} \right)^3 = (\sqrt{2}-1)^6 = (3-2\sqrt{2})^3 = 99-70\sqrt{2}$$

Посчитаем все выражения с различными приближениями $\sqrt{2}$:

- $\frac{7}{5} = 1.4$
- $\frac{17}{12} = 1.41666$
- $\frac{707}{500} = 1.414$
- $\sqrt{2} = 1.4142135624$

| | | | | |
|-------------------|------------------------------------------------|--------------------------------------------------------------|--------------------------------------------|--------------------------|
| $\sqrt{2}$ | $\left(\frac{\sqrt{2}-1}{\sqrt{2}+1}\right)^3$ | $(\sqrt{2}-1)^6$ | $(3-2\sqrt{2})^3$ | $99-70\sqrt{2}$ |
| $\frac{7}{5}$ | $\frac{1}{216} \approx 0.0046$ | $\frac{64}{15625} \approx 0.0051$ | $\frac{1}{125} = 0.008$ | 1 |
| $\frac{17}{12}$ | $\frac{125}{24389} \approx 0.00513$ | $\frac{15625}{2985354} \approx 0.0052$ | $\frac{1}{216} \approx 0.0046$ | $-\frac{1}{6} = -0.6(6)$ |
| $\frac{707}{500}$ | $\frac{8869743}{1758416743} \approx 0.005044$ | $\frac{78672340886049}{15625 \cdot 10^{12}} \approx 0.00504$ | $\frac{636056}{125000000} \approx 0.00509$ | 0.02 |

$$\Delta_{(X \pm Y)} = \Delta_X + \Delta_Y$$

$$\Delta_{(X \cdot Y)} \approx |Y| \Delta_X + |X| \Delta_Y$$

$$\Delta_{(X/Y)} \approx \left| \frac{1}{Y} \right| \Delta_X + \left| \frac{X}{Y^2} \right| \Delta_Y$$

$$|\Delta u| = |f(x_1 + \Delta x_1, \dots, x_n + \Delta x_n) - f(x_1 \dots x_n)|$$

$$|\Delta u| \approx |df(x_1 \dots x_n)| = \left| \sum_{i=1}^n \frac{\partial u}{\partial x_i} \Delta x_i \right| \leq \sum_{i=1}^n \left| \frac{\partial u}{\partial x_i} \right| |\Delta x_i|$$

$$\Delta_u = \sum_{i=1}^n \left| \frac{\partial u}{\partial x_i} \right| \Delta x_i$$

$$|\delta u| = \sum_{i=1}^n \left| \frac{\partial \ln u}{\partial x_i} \right| |\Delta x_i|$$

$$\delta_u = \sum_{i=1}^n \left| \frac{\partial \ln u}{\partial x_i} \right| |\Delta x_i|$$

$$\delta_{(X \pm Y)} = \left| \frac{X}{X \pm Y} \right| \delta_X + \left| \frac{Y}{X \pm Y} \right| \delta_Y$$

$$\delta_{(X \cdot Y)} = \delta_X + \delta_Y$$

$$\delta_{(X/Y)} = \delta_X + \delta_Y$$

Вернемся к прошлому примеру и посчитаем относительную погрешность.

$$\triangleleft x = \frac{7}{5}$$

$$\delta_{f_1} = 3 \left| \frac{1}{x-1} - \frac{1}{x+1} \right| \cdot |\delta x| = 6.25|\delta x|$$

$$\delta_{f_2} = 6 \left| \frac{1}{x-1} \right| \cdot |\delta x| = 15|\delta x|$$

$$\delta_{f_3} = 6 \left| \frac{1}{3-2x} \right| \cdot |\delta x| = 30|\delta x|$$

$$\delta_{f_4} = \left| \frac{90}{99-70x} \right| \cdot |\delta x| = 70|\delta x|$$

Таким образом, наибольшую погрешность даёт f_4 , наименьшую — f_1 .

Пример.

$$y^2 - 140y + 1 = 0$$

$$y = 70 - \sqrt{4899}$$

$$\sqrt{4899} \approx 69.99$$

$$y \approx 70 - 69.99 = 0.01$$

Посчитаем другим методом — избавимся от вычитания похожих чисел.

$$y = \frac{1}{70 + \sqrt{4899}}$$

$$y = \frac{1}{139.99} \approx \frac{1}{140} = 0.00714285 \approx 0.007143$$

Можно заметить, что результат весьма точнее.

Пример. Рассмотрим задачу вычисления суммы $S = \sum_{j=1}^{10^6} \frac{1}{j^2}$.

Если суммировать по формуле $S_n = S_{n-1} + \frac{1}{n^2}$, то из-за того, что сначала суммируются большие числа, а потом малые, погрешность велика: $\Delta = 10^6 \cdot 2^{-1} \approx 2 \cdot 10^{-4}$

Если же суммировать с конца, то $\Delta = \mathcal{O}\left(\frac{1}{n}\right) \approx 6 \cdot 10^{-8}$

Рекомендации для увеличения точности вычислений:

1. Если складывать или вычитать последовательность чисел, то лучше начинать с малых членов.
2. Желательно избегать вычитания двух почти равных чисел, по возможности преобразую формулу.
3. Необходимо сводить к минимуму число математических операций. Это также способствует ускорению работы алгоритма.

4. Если ЯП и компьютер позволяют использовать числа разных типов, то числа с большим числом разрядов всегда повышают точность вычислений (*в ущерб памяти*).

Дробные числа нужно сравнивать с помощью ε , т.е. $|a - b| \leq \varepsilon$

2 Задачи оптимизации. Вводное.

Здесь и далее целевая функция — функция, которую мы минимизируем.

Обозначение. Пусть целевая функция — $f(x)$. Это обозначается как $f(x) \xrightarrow{x \in U} \min$.

$f(x) \rightarrow \max \Rightarrow -f(x) \rightarrow \min$. Таким образом, мы без потери общности рассматриваем задачу минимизации.

Определение. Если $\exists x^* \in U \quad f(x^*) \leq f(x) \quad \forall x \in U$, то такой x^* называется **точкой (глобального) минимума**

Обозначение. Множество всех точек минимума обозначается $U^* = \{x_i^* \mid i = 1 \dots k\}$

Мы рассматриваем класс функций таких, что $U^* \neq \emptyset$

Определение. Функция $f(x)$ называется **унимодальной** на $[a, b]$, если она:

1. Непрерывна на $[a, b]$
2. $\exists \alpha, \beta : a \leq \alpha \leq \beta \leq b$, такие что:
 - (а) Если $a < \alpha$, то на $[a, \alpha]$ $f(x)$ строго монотонно убывает.
 - (б) Если $\beta < b$, то на $[\beta, b]$ $f(x)$ строго монотонно возрастает.
 - (с) $\forall x \in [\alpha, \beta] \quad f(x) = f_* = \min_{[a, b]} f(x)$

Свойства.

1. Если функция унимодальна на $[a, b]$, то она унимодальна и на $[c, d] \subset [a, b]$
2. Если f унимодальна на $[a, b]$, $a \leq x_1 < x_2 \leq b$, тогда:
 - (а) Если $f(x_1) \leq f(x_2)$, то $x^* \in [a, x_2]$
 - (б) Если $f(x_1) > f(x_2)$, то $x^* \in [x_1, b]$

Определение. $f(x)$, заданная на $[a, b]$, называется **выпуклой** на этом отрезке, если

$$\forall x', x'' \in [a, b], \alpha \in [0, 1] \quad f(\alpha x' + (1 - \alpha)x'') \leq \alpha f(x') + (1 - \alpha)f(x'')$$

Свойства.



Рис. 1.1:
Вырожденные α и β ,
унимодальная
функция



Рис. 1.2:
Унимодальная
функция

1. Если $f(x)$ выпукло на $[a, b]$, то $\forall [x', x''] \subset [a, b]$, то её график расположен ниже хорды между x' и x''
2. Всякая выпуклая функция на отрезке является унимодальной на нём.

Определение. Стационарные точки — точки x , для которых $f'(x) = 0$.

Мы будем рассматривать одномерные задачи оптимизации, т.к. многомерные задачи часто сводятся к одномерным.

3 Одномерная минимизация функций. Прямые методы.

Прямые методы — методы, не использующие производные целевой функции.

3.1 Метод дихотомии

Этот метод — тернарный поиск.

$$x_1 = \frac{b + a - \delta}{2} \quad x_2 = \frac{b + a + \delta}{2}$$

$$\tau = \frac{b - x_1}{b - a} = \frac{x_2 - a}{b - a} \rightarrow \frac{1}{2}$$

$$x^* \in [a_i, b_i] \quad \forall i$$

Шаг 1: Находим x_1 и x_2 , вычисляем $f(x_1)$ и $f(x_2)$

Шаг 2: Сравниваем $f(x_1)$ и $f(x_2)$.

- Если $f(x_1) \leq f(x_2)$, переходим к отрезку $[a, x_2]$, т.е. $b = x_2$
- Иначе переходим к $[x_1, b]$, т.е. $a = x_1$

Шаг 3: $\varepsilon_n = \frac{b-a}{2}$, где n — номер итерации.

- Если $\varepsilon_n > \varepsilon$, переходим к новой итерации.
- Если $\varepsilon_n \leq \varepsilon$, завершаем поиск и переходим к шагу 4.

Шаг 4: $X^* \approx \overline{X} = \frac{a+b}{2}$

Примечание. δ выбирается на интервале $(0, 2\varepsilon)$. Чем меньше δ , тем больше относительное уменьшение длины отрезка на каждой итерации. При чрезмерно малом δ сравнение $f(x_1)$ и $f(x_2)$ будет затруднительно, т.к. они близки.

Мы можем оценить число необходимых итераций:

$$n \geq \log_2 \frac{b-a-\delta}{2\varepsilon-\delta}$$

Лекция 2

17 февраля

3.2 Метод золотого сечения

Рассмотрим отрезок $[0, 1]$. Пусть $x_2 = \tau$, тогда симметрично расположенная $x_1 = 1 - \tau$. Пусть дальше был выбран отрезок $[0, \tau]$, тогда пусть $x'_2 = 1 - \tau$. Чтобы новые точки делили отрезок в таком же соотношении, необходимо, чтобы $\frac{1}{\tau} = \frac{\tau}{1-\tau} \Rightarrow \tau^2 = 1 - \tau \Rightarrow \tau = \frac{\sqrt{5}-1}{2} \approx 0.61803$. Таким образом, $x_1 = 1 - \tau = \frac{3-\sqrt{5}}{2}$, $x_2 = \tau = \frac{\sqrt{5}-1}{2}$

В общем случае для отрезка $[a, b]$:

$$x_1 = a + \frac{3 - \sqrt{5}}{2}(b - a), x_2 = a + \frac{\sqrt{5} - 1}{2}(b - a) \quad (1)$$

Вычислим погрешность:

$$\Delta_n = \tau^n(b - a) \quad \varepsilon_n = \frac{\Delta_n}{2} = \frac{1}{2} \left(\frac{\sqrt{5} - 1}{2} \right)^n (b - a)$$

Для заданного ε условия окончания $\varepsilon_n \leq \varepsilon$.

Результат метода:

$$x^* = \frac{a_{(n)} + b_{(n)}}{2}$$

Оценка числа шагов для достижения искомой точности:

$$n \geq \ln \left(\frac{\frac{2\varepsilon}{b-a}}{\ln \tau} \right) \approx 2 \cdot 1 \cdot \ln \left(\frac{b-a}{2\varepsilon} \right)$$

Шаг 1: Находим x_1 и x_2 по формуле (1), вычисляем $f(x_1)$ и $f(x_2)$. $\varepsilon_n = \frac{b-a}{2}$, $\tau = \frac{\sqrt{5}-1}{2}$.

Шаг 2: – Если $\varepsilon_n > \varepsilon$, переходим к шагу 3.

– Если $\varepsilon_n \leq \varepsilon$, переходим к шагу 4.

Шаг 3: Сравниваем $f(x_1)$ и $f(x_2)$.

– Если $f(x_1) \leq f(x_2)$, то $b = x_2, x_2 = x_1, x_1 = b - \tau(b - a)$. Мы запоминаем $f(x_2)$ для следующего шага, т.к. оно равно $f(x_1)$ на этом шаге.

– Иначе $a = x_1, x_1 = x_2, f(x_1) = f(x_2)$. Мы запоминаем $f(x_1)$ для следующего шага, т.к. оно равно $f(x_2)$ на этом шаге.

Шаг 4: $X^* \approx \bar{X} = \frac{a(n)+b(n)}{2}$

3.3 Метод Фибоначчи

Мы знаем, что $F_n = \frac{\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n}{\sqrt{5}}$, а также при $n \rightarrow +\infty$ $F_n \approx \frac{\left(\frac{1+\sqrt{5}}{2}\right)^n}{\sqrt{5}}$

Рассмотрим нулевую итерацию:

$$x_1 = a + \frac{F_n}{F_{n+2}}(b - a) \quad x_2 = a + \frac{F_{n+1}}{F_{n+2}}(b - a)$$

Рассмотрим k -тую итерацию:

$$x_1 = a_{(k)} + \frac{F_{n-k+1}}{F_{n-k+3}}(b_k - a_k) = a_k + \frac{F_{n-k+1}}{F_{n+2}}(b_0 - a_0)$$

$$x_2 = a_{(k)} + \frac{F_{n-k+2}}{F_{n-k+3}}(b_k - a_k) = a_k + \frac{F_{n-k+2}}{F_{n+2}}(b_0 - a_0)$$

Пусть $k = n$, тогда:

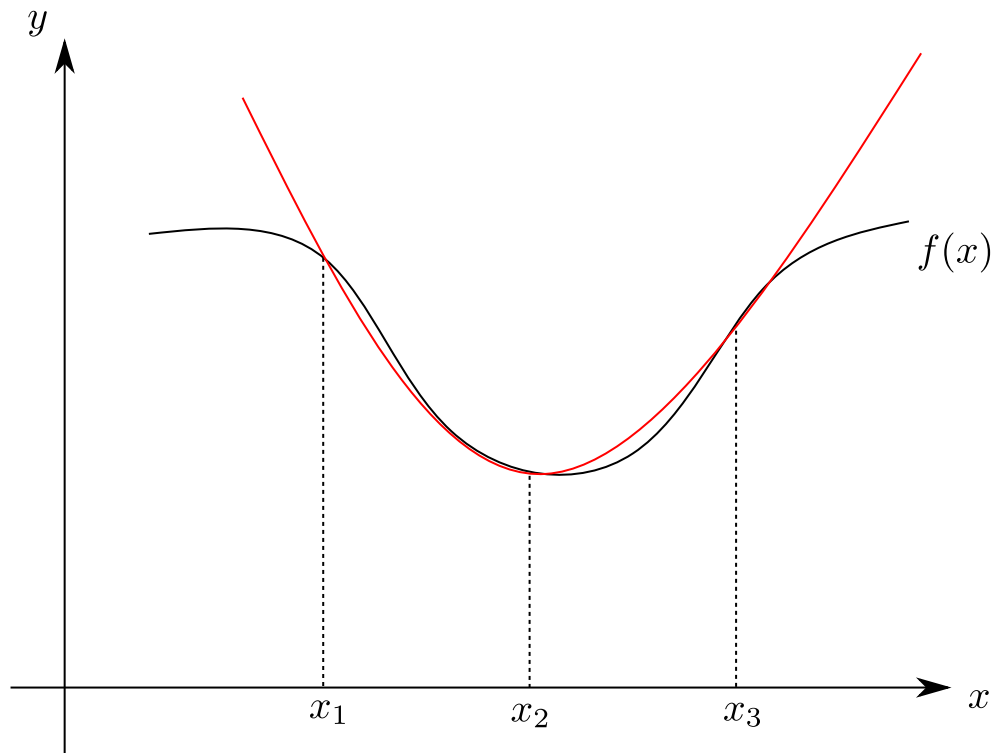
$$x_1 = a_n + \frac{F_1}{F_{n+2}}(b_0 - a_0) \quad x_2 = a_n + \frac{F_2}{F_{n+2}}(b_0 - a_0)$$

Условие на погрешность:

$$\frac{b_n - a_n}{2} = \frac{b_0 - a_0}{F_{n+2}} < \varepsilon$$

Какое брать n ? Такое, что $\frac{b_0 - a_0}{\varepsilon} < F_{n+2}$

Есть проблема, при большом n $\frac{F_n}{F_{n+2}}$ есть бесконечная десятичная дробь, вследствие чего образуется погрешность.

Рис. 2.1: Функция $f(x)$ и её приближение параболой.

3.4 Метод парабол

Пусть $\exists x_1, x_2, x_3 \in [a, b]$, такие что $\begin{cases} x_1 < x_2 < x_3 \\ f(x_1) \geq f(x_2) \leq f(x_3) \end{cases}$

Тогда приближающая парабола имеет вид $q(x) = a_0 + a_1(x - x_1) + a_2(x - x_1)(x - x_2)$.

Мы имеем условия на коэффициенты этой параболы: $\begin{cases} q(x_1) = f(x_1) = f_1 \\ q(x_2) = f(x_2) = f_2 \\ q(x_3) = f(x_3) = f_3 \end{cases}$

Коэффициенты можно найти следующим образом:

$$a_0 = f_1 \quad a_1 = \frac{f_2 - f_1}{x_2 - x_1} \quad a_2 = \frac{1}{x_3 - x_2} \left(\frac{f_3 - f_1}{x_3 - x_1} - \frac{f_2 - f_1}{x_2 - x_1} \right)$$

Тогда результат итерации есть $\bar{x} = \frac{1}{2} \left(x_1 + x_2 - \frac{a_1}{a_2} \right)$, на следующей лекции будет рассказан переход к следующей итерации.

Точки x_1, x_2, x_3 для новой итерации выбираются следующим образом:

1. (а) Если $x_1 < \bar{x} < x_2 < x_3$ и $f(\bar{x}) \geq f(x_2)$, то $x^* \in [\bar{x}, x_3]$, $x_1 = \bar{x}$, точки x_2 и x_3 не меняются.

- (b) Если $x_1 < \bar{x} < x_2 < x_3$ и $f(\bar{x}) < f(x_2)$, то $x^* \in [x_1, x_2]$, $x_3 = x_2$, $x_2 = \bar{x}$, точка x_1 не меняется.
2. (a) Если $x_1 < x_2 < \bar{x} < x_3$ и $f(\bar{x}) \leq f(x_2)$, то $x^* \in [x_2, x_3]$, $x_1 = x_2$, $x_2 = \bar{x}$, точка x_3 не меняется.
- (b) Если $x_1 < x_2 < \bar{x} < x_3$ и $f(\bar{x}) > f(x_2)$, то $x^* \in [x_1, \bar{x}]$, $x_3 = \bar{x}$, точки x_1 и x_2 не меняются.

Примечание. Метод парабол имеет квадратичную сходимость.

Примечание. Метод парабол требует гладкость функции, что неверно для предыдущих методов.

3.5 Комбинированный метод Брента

Для собственного изучения.

Лекция 3

24 февраля

3.6 Метод равномерного перебора

Шаг 1: Если $f(x_0) > f(x_0 + \delta)$, то $k = 1, x_1 = x_0 + \delta, h = \delta$

иначе $x_1 = x_0, h = -\delta$

Шаг 2: $h = 2h, x_{k+1} = x_k + h$

Шаг 3: Если $f(x_k) > f(x_{k+1})$, то $k = k + 1$ и переходим к шагу 2. Иначе прекращаем поиск и искомое лежит в $[x_{k-1}, x_{k+1}]$

4 Методы оптимизации, использующие производную

В рамках этой главы $f(x)$ — дифференцируемая или дважды дифференцируемая выпуклая функция.

Есть три классических метода, использующих производную:

- Средней точки
- Метод хорд
- Метод Ньютона

$f'(x) = 0$ — необходимое и достаточное условие глобального минимума. Таким образом, условие остановки вычислений — $f'(x) \approx 0$, т.е. $|f'(x)| \leq \varepsilon$

4.1 Методы средней точки

Средняя точка $\bar{x} = \frac{a+b}{2}$.

Общая идея алгоритма:

- Если $f'(x) > 0$, то $\bar{x} \in$ участку монотонного возрастания $f(x)$ и $x^* < \bar{x}$, т.е. минимум лежит на $[a, \bar{x}]$
- Если $f'(x) < 0$, то аналогично можем вывести, что минимум лежит на $[\bar{x}, b]$
- Если $f'(x) = 0$, то мы нашли решение.

Перепишем это в виде алгоритма:

Шаг 1: $\bar{x} = \frac{a+b}{2}$, вычислим $f'(\bar{x})$

Шаг 2: Если $|f'(x)| \leq \varepsilon$, то $x^* = \bar{x}$ и завершаем вычисление.

Шаг 3: Сравниваем $f'(x)$ с нулём:

- Если $f'(x) > 0$, то $x^* \in [a, \bar{x}]$ и $b = \bar{x}$
- Иначе $x^* \in [\bar{x}, b]$ и $a = \bar{x}$

Длина отрезка после n итераций есть $\Delta_n = \frac{b-a}{2^n}$

4.2 Метод хорд (метод секущей)

Если $\exists f'(x)$ на $[a, b]$, $f'(a) \cdot f'(b) < 0$ и $f'(x)$ непрерывна на $[a, b]$, то $\exists x \in (a, b) : f'(x) = 0$.

$F(x) = f'(x)$. Пусть \tilde{x} — точка пересечения хорды $F(x)$ с осью Ox на $[a, b]$



Можем тривиально вывести \tilde{x} из уравнения прямой по двум точкам:

$$\tilde{x} = a - \frac{f'(a)}{f'(a) - f'(b)}(a - b) \quad (2)$$

Шаг 1: Считаем \tilde{x} по (2)

Шаг 2: Если $|f'(\tilde{x})| \leq \varepsilon$, то $x^* = \tilde{x}$ и мы заканчиваем вычисление.

Иначе шаг 3.

Шаг 3: Переходим к новому отрезку:

- Если $f'(\tilde{x}) > 0$, то $x^* \in [a, \tilde{x}]$, $b = \tilde{x}$, $f'(b) = f'(\tilde{x})$, переходим к шагу 1
- иначе $x^* \in [\tilde{x}, b]$, $a = \tilde{x}$, $f'(a) = f'(\tilde{x})$, переходим к шагу 1

Примечание. Если $f'(a) \cdot f'(b) \geq 0$, то $x^* = a$ или $x^* = b$.

4.3 Метод Ньютона (*метод касательной*)

Если f выпуклая на $[a, b]$ и дважды непрерывно дифференцируемая, то уравнение $f'(x) = 0$ решается методом Ньютона.

Пусть $x_0 \in [a, b]$ — начальное приближение x^* . $F(x) = f'(x)$ линеаризуема в окрестности x_0 , т.е.

$$F(x) \approx F(x_0) + F'(x_0)(x - x_0)$$

Пусть x_1 — следующее приближение к x^* . Это будет пересечение касательной с Ox . Найдём эту точку.

$$\begin{aligned} F(x_0) + F'(x_0)(x_1 - x_0) &= 0 \\ x_1 &= x_0 - \frac{F(x_0)}{F'(x_0)} \end{aligned}$$

Таким образом, мы можем получить $\{x_k\}_{k=1}^n$ — итерационную последовательность.

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)}$$

Условие остановки такое же, как в предыдущих методах: $|f'(x_k)| \leq \varepsilon$

Лекция 4

3 марта

Пусть x_k — текущая оценка решения x^*

Рассмотрим ряд Тейлора:

$$f(x_k + p) = f(x_k) + pf'(x_k) + \frac{1}{2}p^2 f''(x_k) + \dots$$

$$\begin{aligned} f(x^*) &= \min_x f(x) \\ &= \min_p f(x_k + p) \\ &= \min_p \left(f(x_k) + pf'(x_k) + \frac{1}{2}p^2 f''(x_k) + \dots \right) \\ &\approx \min_p \left(f(x_k) + pf'(x_k) + \frac{1}{2}p^2 f''(x_k) \right) \end{aligned}$$

Приравняем производную выражения под \min к нулю:

$$f'(x_k) + pf''(x_k) = 0$$

$$p = -\frac{f'(x_k)}{f''(x_k)}$$

Тогда $x^* \approx x_k + p$ и $x_{k+1} = x_k + p = x_k - \frac{f'(x_k)}{f''(x_k)}$

Главное преимущество метода Ньютона — квадратичная скорость сходимости, т.е. если x_k достаточно близка к x^* и $f''(x^*) > 0$, то $|x_{k+1} - x^*| \leq \beta |x_k - x^*|^2$

Метод Ньютона может потерпеть неудачу в следующих случаях:

1. $f(x)$ плохо аппроксимируется первыми тремя членами в ряде Тейлора. Тогда x_{k+1} может быть хуже (как аппроксимация) x_k .
2. $f''(x_k) = 0$, тогда p не определен.
3. Кроме f нужно вычислять f' и f'' , что затруднительно в реальных задачах.

Мы можем аппроксимировать производную по определению:

$$f'(x_k) \approx \frac{f(x_k + h) - f(x_k)}{h}$$

Эта формула называется правой разностной схемой, у нее есть улучшение, называемое центральной разностной схемой:

$$f'(x_k) \approx \frac{f(x_k + h) - f(x_k - h)}{2h}$$

Если $f(x)$ — квадратичная функция, то метод Ньютона сходится за один шаг при любом выборе x_0 .

4.3.1 Достаточное условие монотонной сходимости метода Ньютона

Пусть $x^* \in [a, b]$ и $f(x)$ трижды непрерывно дифференцируемая и выпуклая на $[a, b]$ функция. Тогда $\{x_k\}$ будет сходиться к пределу x^* монотонно, если $0 < \frac{x^* - x_{k+1}}{x^* - x_k} < 1$

$$f'(x^*) = 0 = f'(x_k) + f''(x_k)(x^* - x_k) + \frac{f'''(x)}{2}(x^* - x_k)^2$$

$$\frac{x^* - x_{k+1}}{x^* - x_k} = \frac{x^* - x_k + \frac{f'(x_k)}{f''(x_k)}}{x^* - x_k} = 1 - \frac{2}{2 + \frac{f'''(x)(x^* - x_k)^2}{f'(x_k)}}$$

Последовательность итераций $\{x_k\}$ монотонна, если $\frac{f'''(x)}{f'(x_k)} > 0$, таким образом условие монотонной сходимости метода Ньютона — постоянство на $x \in [x^*, x_0]$ знака $f'''(x)$ и его совпадение с $f'(x_0)$.

Пример. $f(x) = x \cdot \arctg(x) - \frac{1}{2}$???

$$f'(x) = \arctg x \quad f''(x) = \frac{1}{1+x^2} > 0 \quad f'''(x) = -\frac{2x}{(1+x^2)^2}$$

$f'(x) \cdot f'''(x) < 0$, таким образом не будет монотонной сходимости.

Пусть $x_0 = 1$.

| k | x_k | $f'(x_k)$ | $f''(x_k)$ |
|-----|-------------------|-----------|---------------|
| 0 | 1 | 0.785 | $\frac{1}{2}$ |
| 1 | -0.57 | -0.518 | a |
| 2 | 0.117 | 0. | |
| 4 | $9 \cdot 10^{-8}$ | | |

4.4 Модификации метода Ньютона

4.4.1 Метод Ньютона-Рафсона

$$x_{k+1} = x_k - \tau_k \frac{f'(x_k)}{f''(x_k)}, 0 < \tau_k \leq 1$$

τ_k — константы. Если $\tau = 1$, то метод Ньютона-Рафсона вырождается в метод Ньютона.

Для нахождения τ_k зададим $\varphi(\tau)$:

$$\varphi(\tau) = f\left(x_k - \tau \frac{f'(x_k)}{f''(x_k)}\right) \rightarrow \min$$

Тогда

$$\tau_k = \frac{(f'(x_k))^2}{(f'(x_k))^2 + (f'(\tilde{x}))^2}, \text{ где } \tilde{x} = x_k - \frac{f'(x_k)}{f''(x_k)}$$

4.4.2 Метод Марквардта

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k) + \mu_k}$$

, где $\mu_k > 0$

μ_0 выбирают на порядок выше значения $f''(x_0)$, $\mu_{k+1} = \begin{cases} \frac{\mu_k}{2} & , \text{ если } f(x_{k+1}) < f(x_k) \\ \mu_{k+1} = 2\mu_k & , \text{ если } f(x_{k+1}) \geq f(x_k) \end{cases}$

5 Метод минимизации многомодальных функций (метод ломаных)

Определение. $f(x)$, $x \in [a, b]$ удовлетворяет условию Липшица, если $\forall x_1, x_2 \in [a, b] |f(x_1) - f(x_2)| \leq L|x_1 - x_2|$

Шаг 1 Возьмём $x_1^* = \frac{1}{2L}(f(a) - f(b) + L(a + b))$ и $p_1^* = \frac{1}{2}(f(a) + f(b) + L(a - b))$. Добавим в рассматриваемое множество $x'_1 = x_1^* - \Delta_1$ и $x''_1 = x_1^* + \Delta_1$, где $\Delta_1 = \frac{1}{2L}(f(x_1^*) - p_1)$

Шаг 2 Из пар (x'_1, p_1) и (x''_1, p_1) выберем пару с минимальной $p : (x_2^*, p_2^*)$ и исключим из рассматриваемого множества.

Шаг n В результате мы получим множество из n пар (x, p) . Исключаем пару с минимальной p и вместо неё

Пример. $f(x) = \frac{\sin x}{x}$, $[a, b] = [10, 15]$, $\varepsilon = 0.01$

Проверим условие Липшица:

$$|f'(x)| = \left| \frac{x \cos x - \sin x}{x^2} \right| < \frac{x |\cos x| + \sin |x|}{x^2} < \frac{x + 1}{x^2} \leq 0.11$$

| n | x_n^* | p_n^* | $2L\Delta_n$ | x_n' | x_n'' | p_n |
|-----|---------|---------|-----------------------|--------|---------|--------|
| 1 | 12.056 | -0.281 | 0.240 | 10.963 | 13.149 | -0.161 |
| 2 | 10.963 | -0.161 | 0.070 | 10.646 | 11.280 | -0.126 |
| 3 | 13.149 | -0.161 | 0.203 | 12.227 | 14.701 | -0.096 |
| 4 | 10.646 | -0.126 | 0.038 | 10.474 | 10.818 | -0.107 |
| 5 | 11.280 | -0.126 | 0.041 | 11.094 | 11.466 | -0.106 |
| 6 | 10.474 | -0.107 | 0.024 | 10.364 | 10.584 | -0.095 |
| 7 | 10.818 | -0.107 | 0.160 | 10.745 | 10.891 | -0.099 |
| 8 | 11.094 | -0.106 | 0.016 | 11.020 | 11.168 | -0.098 |
| 9 | 11.466 | -0.106 | 0.028 | 11.338 | 11.594 | -0.092 |
| 10 | 10.891 | -0.099 | $0.008 < \varepsilon$ | | | |

Лекция 5

10 марта

6 Минимизация функций многих переменных

6.1 Постановка задачи

Необходимо найти $x^* = (x_1 \ x_2 \ \dots \ x_n)^T \in U \subset E_n$, где U — множество допустимых значений, а E_n — евклидово пространство размера n , при этом $f(x^*) = \min_{x \in U} f(x)$.

Примечание.

1. Как и в одномерном случае, задача минимизации эквивалентна задачи максимизации и в общем случае называется задачей поиска экстремума.
2. Если U задается ограничениями на вектор x , то такая задача оптимизации называется задачей поиска условного экстремума.
3. Если $U = E_n$, т.е. не имеет ограничений, то такая задача оптимизации называется задачей поиска безусловного экстремума.
4. Решением задачи поиска экстремума называется пара $(x^*, f(x^*))$.

Определение. Если $f(x^*) \leq f(x) \ \forall x \in U$, то x^* называется **глобальным минимумом**.

Определение. Если $\exists \varepsilon > 0 : \|x - x^*\| < \varepsilon \Rightarrow f(x^*) \geq f(x)$, то x^* называется **локальным минимумом**.

Примечание.

$$\|x\| = \sqrt{\sum_i x_i^2}$$

Определение. Поверхностью уровня функции $f(x)$ называется множество точек, в которых функция принимает постоянное значение.

Определение. Градиентом $\nabla f(x)$ непрерывно дифференцируемой функции $f(x)$ в x называется:

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

Примечание. Градиент направлен по нормали к поверхности уровня, т.е. перпендикулярно к касательной плоскости, проведенной в точке x в сторону наибольшего возрастания функции.

Определение. Матрица Гессе $\mathbf{H}(x)$ дважды непрерывно дифференцируемой в точке x функции $f(x)$ называется матрица частных производных производных второго порядка, вычисленных в данной точке.

$$\mathbf{H}(x) = \begin{pmatrix} \frac{\partial^2 f(x)}{\partial x_1^2} & \frac{\partial^2 f(x)}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f(x)}{\partial x_1 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f(x)}{\partial x_n \partial x_1} & \frac{\partial^2 f(x)}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f(x)}{\partial x_n^2} \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} & \cdots & h_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ h_{n1} & h_{n2} & \cdots & h_{nn} \end{pmatrix}$$

1. $\mathbf{H}(x)$ симметрична, имеет размер $n \times n$.
2. Можно определить антиградиент — вектор, равный по модулю градиенту и направленный противоположно. Антиградиент указывает в сторону наибольшего убывания $f(x)$.
3. $\Delta f(x) = f(x + \Delta x) - f(x) = \nabla f(x)^T \Delta x + \frac{1}{2} \Delta x^T \mathbf{H}(x) \Delta x + o(\|\Delta x\|^2)$, где $o(\|\Delta x\|^2)$ есть сумма всех членов разложения, имеющих порядок выше второго. Можем заметить, что $\Delta x^T \mathbf{H}(x) \Delta x$ — квадратичная форма.

Определение. Квадратичная форма $\Delta x^T \mathbf{H}(x) \Delta x$ ¹ называется:

- Положительно определенной, если $\forall \Delta x \neq 0 \quad \Delta x^T \mathbf{H}(x) \Delta x > 0$
- Отрицательно определенной, если $\forall \Delta x \neq 0 \quad \Delta x^T \mathbf{H}(x) \Delta x < 0$
- Положительно полуопределенной, если $\forall \Delta x \quad \Delta x^T \mathbf{H}(x) \Delta x \geq 0$ и имеется $\Delta x \neq 0 : \Delta x^T \mathbf{H}(x) \Delta x = 0$
- Отрицательно полуопределенной, если $\forall \Delta x \quad \Delta x^T \mathbf{H}(x) \Delta x \leq 0$ и имеется $\Delta x \neq 0 : \Delta x^T \mathbf{H}(x) \Delta x = 0$
- Неопределенной, если $\exists \Delta x, \tilde{\Delta x} : \Delta x^T \mathbf{H}(x) \Delta x > 0, \tilde{\Delta x}^T \mathbf{H}(x) \tilde{\Delta x} < 0$
- Тождественно равной нулю, если $\forall \Delta x \quad \Delta x^T \mathbf{H}(x) \Delta x = 0$

¹ и соответствующая ей матрица $\mathbf{H}(x)$

6.2 Свойства выпуклых множеств и выпуклых функций

Определение. Пусть $x, y \in E_n$, множество точек вида $\{z\} \subset E_n : z = \alpha x + (1 - \alpha)y$, т.е. z это отрезок $[x, y]$.

Определение. $U \subset E_n$ выпуклое, если вместе с точками $x, y \in U$ оно содержит весь отрезок z .

Определение. Функция $f(x)$, заданная на выпуклом множестве $U \subset E_n$, называется:

- **выпуклой**, если:

$$\forall x, y \in U, \alpha \in [0, 1] \quad f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y)$$

- **строго выпуклой**, если:

$$\forall x, y \in U, \alpha \in (0, 1) \quad f(\alpha x + (1 - \alpha)y) < \alpha f(x) + (1 - \alpha)f(y)$$

- **сильно выпуклой** с константой $l > 0$, если:

$$\forall x, y \in U, \alpha \in [0, 1] \quad f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y) - \frac{l}{2}\alpha(1 - \alpha)\|x - y\|^2$$

Свойства.

1. Функция $f(x)$ выпуклая, если её график целиком лежит не выше отрезка, соединяющего две её произвольные точки.
2. Функция $f(x)$ строго выпуклая, если её график целиком лежит ниже отрезка, соединяющего две её произвольные, но не совпадающие точки.²
3. Если функция сильно выпуклая, то она одновременно строго выпуклая и выпуклая.
4. Если функция строго выпуклая, то она выпуклая.
5. Выпуклость функции можно определить по $H(x)$:
 - Если $H(x) \geq 0 \quad \forall x \in E_n$, то $f(x)$ выпуклая.
 - Если $H(x) > 0 \quad \forall x \in E_n$, то $f(x)$ строго выпуклая.
 - Если $H(x) \geq lE^3 \quad \forall x \in E_n$, то $f(x)$ сильно выпуклая.

Свойства (выпуклых функций).

1. Если $f(x)$ — выпуклая функция на множестве U , то всякая точка локального минимума — глобальный минимум на U .

² Пример будет на следующей лекции

³ единичная матрица

2. Если выпуклая функция достигает своего минимума в двух различных точках, то она достигает минимума во всех точках отрезка, соединяющего эти точки.
3. Если $f(x)$ строго выпуклая функция на множестве U , то она может достигать своего глобального минимума на U не более чем в одной точке.

6.3 Необходимое и достаточное условие безусловного экстремума

6.3.1 Необходимое условие экстремума первого порядка

Пусть $x^* \in E_n$ — точка локального минимума⁴ $f(x)$ на E_n и $f(x)$ дифференцируема в точке x^* . Тогда $\nabla f(x)$ в точке x^* равен нулю: $\nabla f(x^*) = 0$ или $\frac{\partial f(x^*)}{\partial x_i} = 0 \quad \forall i \in 1 \dots n$. Точка x^* называется **стационарной**.

6.3.2 Необходимое условие экстремума второго порядка

Пусть $x^* \in E_n$ — точка локального минимума⁵ $f(x)$ на E_n и $f(x)$ дважды дифференцируема в точке x^* . Тогда $\mathbf{H}(x^*)$ положительно полуопределена или отрицательно полуопределена.

6.3.3 Достаточное условие экстремума

Пусть $f(x)$ в $x^* \in E_n$ дважды дифференцируема, $\nabla f(x^*) = 0$ и $\mathbf{H}(x) > 0$ (или $\mathbf{H}(x) < 0$). Тогда x^* — точка локального минимума⁶ $f(x)$ на E_n .

6.3.4 Проверка выполнений условий экстремума

- Вычисление угловых миноров $\mathbf{H}(x)$
- Вычисление главных миноров $\mathbf{H}(x)$

Есть два способа это сделать:

1. Исследование положительной или отрицательной определенности угловых и главных миноров $\mathbf{H}(x)$.
2. Анализ собственных значений $\mathbf{H}(x)$.

⁴ или максимума

⁵ или максимума

⁶ или максимума

Лекция 6

17 марта

6.3.5 Критерии Сильвестра проверки достаточных условий экстремума

1. Для того, чтобы $H(x^*) > 0$ и x^* являлась точкой локального минимума, необходимо и достаточно, чтобы **угловые** миноры были строго положительными, т.е. $\Delta_1 > 0, \Delta_2 > 0 \dots \Delta_n > 0$.
2. Для того, чтобы $H(x^*) < 0$ и x^* являлась точкой локального максимума, необходимо и достаточно, чтобы знаки **угловых** миноров чередовались, т.е. $\Delta_1 < 0, \Delta_2 > 0 \dots (-1)^n \Delta_n > 0$

6.3.6 Критерии Сильвестра проверки необходимых условий экстремума

1. Для того, чтобы $H(x^*) \geq 0$ и x^* мог быть точкой локального минимума, необходимо и достаточно, чтобы **главные** миноры были положительными, т.е. $\Delta_1 \geq 0, \Delta_2 \geq 0, \dots \Delta^n \geq 0$
2. Для того, чтобы $H(x^*) \leq 0$ и x^* мог быть точкой локального максимума, необходимо и достаточно, чтобы знаки **главных** миноров чередовались, т.е. $\Delta_1 \leq 0, \Delta_2 \geq 0, \dots (-1)^n \Delta^n \geq 0$

Определение. Собственные значения λ_i матрицы $H(x^*)$ находятся как корни характеристического уравнения $|H(x^*) - \lambda E| = 0$

Если $H(x)$ — вещественная, симметричная матрица, то λ_i тоже вещественные.

6.4 Квадратичные функции

Определение. Функция вида

$$f(x) = \sum_{i=1}^n \sum_{j=1}^n a_{ij} x_i x_j + \sum_{j=1}^n b_j x_j + c$$

называется **квадратичной функцией** n переменных.

Положим $a_{ij} = a_{ji}$ ¹, тогда a_{ij} задаёт симметричную матрицу A .

$$f(x) = \frac{1}{2} \langle Ax, x \rangle + \langle b, x \rangle + c \quad (3)$$

, где $b = (b_1 \ \dots \ b_n)^T \in E_n$ — вектор коэффициентов, $x = (x_1 \ \dots \ x_n)^T$

Свойства (квадратичных функций).

$$1. \ \nabla f(x) = Ax + b$$

$$\begin{aligned} \frac{\partial f}{\partial x_k} &= \frac{\partial}{\partial x_k} \left(\frac{1}{2} \sum_{j=1}^n a_{ij} x_i x_j + \sum_{j=1}^n b_j x_j + c \right) \\ &= \frac{1}{2} \sum_{i=1}^n (a_{ik} + a_{ki}) x_i + b_k \\ &= \sum_{i=1}^n a_{ki} x_i + b_k \end{aligned}$$

$$2. \ H(x) = A$$

$$\frac{\partial^2 f}{\partial x_l \partial x_k} = \frac{\partial}{\partial x_l} \left(\frac{\partial f}{\partial x_k} \right) = \frac{\partial}{\partial x_l} \left(\sum_{i=1}^n a_{ki} x_i + b_k \right) = a_{kl}$$

3. Квадратичная функция $f(x)$, для которой выполнено (3), с положительно определенной матрицей A сильно выпуклая, т.к. $H(x) = A$ — симметричная и положительно определенная, а следовательно $\lambda_i > 0$ и \exists ортонормированный базис из собственных векторов этой матрицы. В этом базисе:

$$A = \begin{pmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_n \end{pmatrix} \quad A - lE = \begin{pmatrix} \lambda_1 - l & 0 & \dots & 0 \\ 0 & \lambda_2 - l & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_n - l \end{pmatrix}$$

В этом базисе все угловые миноры матрицы A и матрицы $A - lE$ положительны при достаточно малом $l : 0 < l < \lambda_{\min} \Rightarrow f$ сильно выпуклая.

¹ На лекции было дано $a_{ij} = a_{ij} + a_{ji}$, но это не похоже на правду, т.к. тогда $a_{ji} = 0 \ \forall i, j$. Нулевая матрица действительно симметрична, но вряд ли это подразумевалось.

6.5 Общие принципы многомерной оптимизации

Алгоритмы многомерной оптимизации обычно используют итерационную процедуру, описываемую следующим образом: $x^{k+1} = \Phi(x^k, x^{k-1} \dots x^0)$, $x^0 \in E_n$. Эти алгоритмы строят последовательность промежуточных результатов $\{x_k\}$, которая обладает следующими свойствами:

$$\begin{cases} \lim_{k \rightarrow +\infty} f(x^k) = f^* = \min_{E_n} f(x), & \text{если } U^* \neq \emptyset \\ \lim_{k \rightarrow +\infty} f(x^k) = f^* = \inf_{E_n} f(x), & \text{если } U^* = \emptyset \end{cases} \quad (4)$$

, где U^* — множество точек глобального минимума функции $f(x)$.

Определение. Если для $\{x^k\}$ выполняется условие (4), то эта последовательность называется минимизирующей.

Определение. Если для $U^* \neq \emptyset$ выполняется условие $\lim_{k \rightarrow +\infty} \rho(x^k, U^*) = 0$, то $\{x^k\}$ сходится к множеству U^* .

Определение (расстояние от точки до множества). $\rho(x, U) = \inf_{y \in U} \rho(x, y)$

Если U^* состоит из одной точки x^* , то для $\{x^k\}$, сходящейся к U^* , $\lim_{k \rightarrow +\infty} x^k = x^*$. Минимизирующая последовательность может и не сходиться к точке минимума.

Теорема 1 (Вейерштрасса). Если $f(x)$ непрерывна в E_n и множество $U^\alpha = \{x : f(x) \leq \alpha\}$ для некоторого α непусто и ограничено, то $f(x)$ достигает глобального минимума в E_n .

6.6 Скорость сходимости минимизирующей последовательности

Определение. $\{x^k\}$ сходится к точке x^* линейно (со скоростью геометрической прогрессии), если

$$\exists q \in (0, 1) : \rho(x^k, x^*) = q \rho(x^{k-1}, x^*)$$

, т.е. $\rho(x^k, x^*) \leq q^k \rho(x^0, x^*)$

Определение. Сходимость называется сверхлинейной, если

$$\rho(x^k, x^*) \leq q_k \rho(x^{k-1}, x^*)$$

и $q_k \rightarrow +0$ при $k \rightarrow +\infty$

Определение. Сходимость называется квадратичной, если

$$\rho(x^k, x^*) \leq (c \rho(x^{k-1}, x^*))^2, c > 0$$

Критерий окончания итерационного процесса:

1. $\rho(x^{k+1}, x^k) < \varepsilon_1$
2. $|f(x^{k+1}) - f(x^k)| < \varepsilon_2$

$$3. \|\nabla f(x^k)\| < \varepsilon_3$$

$$x^{k+1} = x^k + \alpha_k p^k \quad (5)$$

, где p^k — направление поиска из x^k в x^{k+1} , а α_k — величина шага. Алгоритмы, которые мы будем рассматривать, различаются этими двумя величинами.

Определение. В итерационном процессе (5) производится исчерпывающий спуск, если величина шага α_k находится из решения одномерной задачи минимизации

$$\Phi_k(\alpha) \rightarrow \min_{\alpha}, \Phi_k(\alpha) = f(x^k + \alpha p^k)$$

Теорема 2. Если функция $f(x)$ дифференцируема в E_n , то в итерационном процессе (5) с выбором шага с исчерпывающим спуском для любого $k \geq 1$ выполняется следующее условие:

$$\langle \nabla f(x^{k+1}), p^k \rangle = 0$$

Доказательство. Для $\Phi_k(\alpha)$ необходимое условие минимума функции:

$$\frac{d\Phi_k(\alpha)}{d\alpha} = \sum_{j=1}^n \frac{\partial f(x^{k+1})}{\partial x_j} \frac{dx_j^{k+1}}{d\alpha} = 0$$

Учитывая, что $x_j^{k+1} = x_j^k + \alpha p_j^k$, получаем, что $\frac{dx_j^{k+1}}{d\alpha} = p_j^k$ □

Теорема 3. Для квадратичной функции $f(x) = \frac{1}{2} \langle Ax, x \rangle + \langle b, x \rangle + c$ величина α_k исчерпывающего спуска в итерационном процессе (5) равна:

$$\alpha_k = -\frac{\langle \nabla f(x^k), p^k \rangle}{\langle Ap^k, p^k \rangle} = -\frac{\langle Ax^k + b, p^k \rangle}{\langle Ap^k, p^k \rangle}$$

Доказательство.

$$\begin{aligned} x^{k+1} &= x^k + \alpha_k p^k \\ Ax^{k+1} + b &= Ax^k + b + \alpha_k Ap^k \\ \nabla f(x^{k+1}) &= \nabla f(x^k) + \alpha_k Ap^k \\ \langle \nabla f(x^{k+1}), p^k \rangle &= 0 \\ \langle \nabla f(x^k) + \alpha_k Ap^k, p^k \rangle &= 0 \\ \langle \nabla f(x^k), p^k \rangle + \langle \alpha_k Ap^k, p^k \rangle &= 0 \\ \langle \nabla f(x^k), p^k \rangle + \alpha_k \langle Ap^k, p^k \rangle &= 0 \\ \alpha_k &= -\frac{\langle \nabla f(x^k), p^k \rangle}{\langle Ap^k, p^k \rangle} \end{aligned}$$

□

Лекция 7

22 марта (дополнительная лекция)

Определение. Направление вектора p^k называется направлением убывания функции $f(x)$ в точке x^k , если при всех достаточно малых положительных α выполняется неравенство $f(x^k + \alpha p^k) < f(x^k)$

Теорема 4 (достаточное условие направления убывания). Пусть функция $f(x)$ дифференцируема в точке x^k . Если вектор p^k удовлетворяет условию $\langle \nabla f(x^k), p^k \rangle < 0$, то направление вектора p^k является направлением убывания.

Доказательство. Из свойства дифференцируемости функции и условия теоремы следует, что

$$f(x^{k+1}) - f(x^k) = f(x^k + \alpha p^k) - f(x^k) = \langle \nabla f(x^k), \alpha p^k \rangle + o(\alpha) = \alpha \langle \nabla f(x^k), p^k \rangle + \frac{o(\alpha)}{\alpha} < 0$$

при всех достаточно малых $\alpha > 0$, т.е. p^k задает направление убывания $f(x)$ в точке x^k . \square

Геометрическая интерпретация: $\langle \nabla f(x^k), p^k \rangle < 0 \Rightarrow p^k$ составляет тупой угол с $\nabla f(x^k)$.

Рассмотрим $f(x)$, дифференцируемую в E_n и запишем итерационную процедуру минимизации:

$$x^{k+1} = x^k + \alpha_k p^k \quad (6)$$

, где p^k определяется с учетом информации о частных производных, а величина α_k такова, что:

$$f(x^{k+1}) < f(x^k) \quad (7)$$

Условие остановки итерационного процесса: $\|\nabla f(x^k)\| < \varepsilon$.

6.7 Метод градиентного спуска

Предположим, что в (6) $p^k = -\nabla f(x^k)$. Если $\nabla f(x^k) \neq 0$, то $\langle \nabla f(x^k), p^k \rangle < 0$, следовательно p^k — направление убывания $f(x)$, причём в малой окрестности точки x^k направление p^k обеспечивает наискорейшее убывание функции. Таким образом, $\exists \alpha_k > 0$, такое что (7) выполнено.

Алгоритм метода:

1. Выбрать $\varepsilon > 0, \alpha > 0, x \in E_n$, вычислить $f(x)$.
2. Вычислить $\nabla f(x)$. Проверить условие $\|\nabla f(x)\| < \varepsilon$. Если оно выполнено, то завершить процесс, иначе перейти к шагу 3.
3. Найти $y = x - \alpha \nabla f(x)$ и $f(y)$. Если $f(y) < f(x)$, то положить $x = y, f(x) = f(y)$ и перейти к шагу 2, иначе — к 4.
4. Положить $\alpha = \frac{\alpha}{2}$ и перейти к шагу 3.

Примечание. В окрестности стационарной точки величина градиента мала, вследствие чего сходимость процесса замедляется. Поэтому в (6) иногда полагают

$$p^k = -\frac{\nabla f(x^k)}{\|\nabla f(x^k)\|}$$

Теорема 5. Пусть симметричная матрица A квадратичной функции $f(x)$ положительно определена, l и L — наименьшее и наибольшее собственные значения A ($0 < l \leq L$). Тогда при любых $\alpha \in (0, \frac{2}{L})$ и $x^0 \in E_n$ (6) сходится к единственной точке глобального минимума x^* функции $f(x)$ линейно:

$$\rho(x^k, x^*) \leq q^k \rho(x^0, x^*)$$

Доказательство. Т.к. A положительно определена, то $f(x)$ сильно выпукла. Следовательно точка x^* существует и единственна. $\nabla f(x^*) = 0$, тогда:

$$\nabla f(x^k) = Ax^k + b = Ax^k + b - Ax^* - b = A(x^k - x^*)$$

$$\begin{aligned} \|x^k - x^*\| &= \|x^{k-1} - \alpha \nabla f(x^{k-1}) - x^*\| \\ &= \|x^{k-1} - x^* - \alpha A(x^{k-1} - x^*)\| \\ &= \|(E - \alpha A)(x^{k-1} - x^*)\| \end{aligned}$$

$$\begin{aligned} \|x^k - x^*\| &\leq \|E - \alpha A\| \cdot \|x^{k-1} - x^*\| \\ &\leq q \|x^{k-1} - x^*\| \\ &\leq q^k \|x^0 - x^*\| \end{aligned}$$

□

q — оценка нормы матрицы через величину её собственных значений: $\|E - \alpha A\| \leq q = \max\{|1 - \alpha l|, |1 - \alpha L|\}$. Величина q принимает наименьшее значение при $q^* = \frac{L-l}{L+l}$ при $\alpha = \alpha^* = \frac{2}{L+l}$

Доказательство. Т.к. $l < L$, то $1 - \alpha l = -(1 - \alpha L)$. Тогда $q = 1 - \alpha l = 1 - \frac{2l}{L+l} = \frac{L-l}{L+l}$ \square

От соотношения l и L существенно зависит число итераций градиентного метода при минимизации выпуклой квадратичной функции.

Пример ($L = l > 0$). $f(x) = x_1^2 + x_2^2 \rightarrow \min, x^0 = (1 \ 1)^T, \alpha = \alpha^*$

Решение:

$$A = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} \Rightarrow l = L = 2 \Rightarrow \alpha^* = \frac{2}{2+2} = \frac{1}{2}$$

$$x^1 = x^0 - \frac{1}{2} \nabla f(x^0) = (0 \ 0)^T$$

Несложно заметить, что $x^1 = x^*$.

Таким образом, точка минимума нашлась за один шаг.

При $l = L$ линии уровня $f(x)$ — концентрические окружности. При $L \gg l > 0$ линии уровня $f(x)$ — эллипсы:

Пример ($L \gg l > 0$). $f(x) = x_1^2 + 100x_2^2 \rightarrow \min, x_0 = (1 \ 1)^T, \alpha = \alpha^*$

$$A = \begin{pmatrix} 2 & 0 \\ 0 & 200 \end{pmatrix} \Rightarrow l = 2, L = 200$$

$$-\nabla f(x^0) = (-2 \ -200)^T$$

$x^* - x^0 = (-1 \ -1)^T$ — направление к точке глобального минимума, сильно отличается от направления спуска, минимизирующая последовательность сходится зигзагообразно.

Определение. Число обусловленности для симметричной положительно определенной матрицы: $\mu = \frac{L}{l}$. Оно характеризует степень вытянутости линий уровня $f(x) = C$.

- Если μ велико, то линии уровня сильно вытянуты, функция имеет овальный характер, т.е. резко меняется по одним направлениям и слабо по другим. В таком случае задачу минимизации называют плохо обусловленной.
- Если $\mu \sim 1$, линии уровня близки к окружности и задача называется хорошо обусловленной.

6.8 Метод наискорейшего спуска

Идея: после вычисления в начальной точке градиента функции делает в направлении антиградиента не малый шаг, а передвигается до тех пор, пока функция убывает. Достигнув точки минимума на выбранном направлении, повторяет описанную процедуру.

α_k находится из решения задачи одномерной оптимизации:

$$\Phi_k(\alpha) \rightarrow \min, \Phi_k(\alpha) = f(x^k - \alpha \nabla f(x^k)), \alpha > 0 \quad (8)$$

Алгоритм метода:

1. Выбрать $\varepsilon > 0$, $x^0 \in E_n$, вычислить $f(x^0)$
2. Вычислить $\nabla f(x)$. Проверить условие $\|\nabla f(x)\| < \varepsilon$. Если оно выполнено, то завершить процесс, иначе перейти к шагу 3.
3. Решить задачу (8) для $x^k = x$, т.е. найти α^* . Положить $x = x - \alpha^* \nabla f(x)$, перейти к шагу 2.

Определение. Ненулевые вектора $p^1 \dots p^k$ называются **сопряженными** относительно матрицы A размера $n \times n$ или **A -ортогональными**, если $\langle Ap^i, p^j \rangle = 0$, если $i \neq j$.

Система из n векторов $p^1 \dots p^n$, сопряженных относительно положительно определенной матрицы A , линейно независима и образует базис в E_n .

Рассмотрим минимизацию квадратичной функции $f(x) = \frac{1}{2} \langle Ax, x \rangle + \langle b, x \rangle + c$ в E_n , где A положительно определенная и итерационный процесс (6), где p^k — A -ортогональные.

Если в таком итерационном процессе на каждом шаге исчерпывающий спуск, то:

$$\alpha_k = - \frac{\langle \nabla f(x^0), p^k \rangle}{\langle Ap^k, p^k \rangle}$$

Доказательство.

$$x^k = x^{k-1} + \alpha_k p^k = x^0 + \sum_{i=1}^n \alpha_i p^i$$

$$\nabla f(x) = Ax + b$$

$$\nabla f(x^k) = \nabla f(x^0) + \sum_{i=1}^k \alpha_i Ap^i$$

Домножим на p^k :

$$\langle \nabla f(x^k), p^k \rangle = \langle \nabla f(x^0), p^k \rangle + \langle \alpha_k Ap^k, p^k \rangle$$

$$\langle \nabla f(x^0), p^k \rangle + \langle \alpha_k Ap^k, p^k \rangle = 0$$

Т.к. A положительно определено, $\langle Ap^k, p^k \rangle > 0$ и для α_k :

$$\alpha_k = -\frac{\langle \nabla f(x^0), p^k \rangle}{\langle Ap^k, p^k \rangle}$$

□

Теорема 6. Последовательный исчерпывающий спуск по A -ортогональным направлениям приводит квадратичной формы не более чем за n шагов.

Лекция 8

24 марта

6.9 Метод сопряженных градиентов

$$p^{k+1} = -\nabla f(x^{k+1}) + \beta_k p^k \quad (9)$$

β_k выбираются так, чтобы получалась последовательность A -ортогональных векторов p^0, p^1, \dots . Из условия $\langle Ap^{k+1}, p^k \rangle = 0$ имеем:

$$\beta_k = \frac{\langle A \nabla f(x^{k+1}), p^k \rangle}{\langle Ap^k, p^k \rangle}$$

Для квадратичной функции:

$$\alpha_k = -\frac{\langle \nabla f(x^k), p^k \rangle}{\langle Ap^k, p^k \rangle}$$

Вышеуказанный итерационный процесс дает точки $x^0 \dots x^k$ и векторы $p^0 \dots p^k$, такие, что если $\nabla f(x^i) \neq 0$ при $0 \leq i < k \leq n-1$, то векторы $p^0 \dots p^k$ A -ортогональны, а $\nabla f(x^0) \dots \nabla f(x^i)$ взаимно ортогональны.

Т.к. в (9) p^k A -ортогональны, то метод гарантирует нахождение точки минимума сильно выпуклой функции не более, чем за n шагов.

Следующие формулы описывают итерационный процесс метода сопряженных градиентов:

$$x^{k+1} = x^k + \alpha_k p^k \quad x^0 \in E_n, p^0 = -\nabla f(x^0)$$

$$f(x^k + \alpha_k p^k) = \min_{\alpha > 0} f(x^k + \alpha p^k)$$

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

$$\beta_k = \frac{\|\nabla f(x^{k+1})\|^2}{\|\nabla f(x^k)\|^2}$$

Можем заметить, что мы не используем матрицу A , поэтому этот метод может применяться для минимизации не только квадратичных функций. Но этот метод может не находить точку минимума не квадратичной функции за конечное число шагов.

Вектора p^k вообще говоря могут не образовывать A -ортогональную систему, вследствие чего реализация этого метода будет сопровождаться неизбежными накапливающимися погрешностями, из-за чего сходимость метода может нарушиться. Чтобы с этим бороться, через каждые N шагов производят обновление метода, т.е. $\beta_{m \cdot N} = 0$, $m \in \mathbb{N}$, где $m \cdot N$ называются моментами обновления метода (*рестарта*), а N обычно принимают за n — размерность пространства E_n .

6.10 Метод стохастического градиентного спуска

Этот метод используется, когда дано множество пар (x, y) , называемых тренировочными наборами. Это множество разделяется на K подмножеств размера M , называемых minibatch.¹

$$\begin{aligned} X^{(k)} &= \{x_i \mid i = M_k, \dots, (M_k + M - 1)\} \\ Y^{(k)} &= \{y_i \mid i = M_k, \dots, (M_k + M - 1)\} \\ L^{(k)}(w) &= \sum_{i=0}^M L(w, x_{M_k+i}, y_{M_k+i}) \end{aligned}$$

Есть большие итерации по p , называемые эпохами и малые итерации $w_p^{(k+1)} = w_p^{(k)} - \eta \cdot \nabla L^{(k)}(w_p^{(k)})$, $w_{p+1}^{(0)} = w_p^{(K)}$, где $\eta = \text{const}$ ². При переходе от одной эпохи к другой minibatch-и случайно перемешиваются.

6.10.1 Adagrad

Примечание. Куда более адекватная статья по adagrad и прочим модификациям стохастического спуска: <https://habr.com/ru/post/318970/>

Примечание. Лучшая модификация SGD — Adam, если не хочется думать, надо всегда использовать его.

Идея алгоритма — в покоординатном изменении η . Пусть $\eta_p = \begin{pmatrix} \eta_p^{(1)} & \dots & \eta_p^{(d)} \end{pmatrix}$, η_0 — константный вектор $\eta_0^{(i)} = \eta \ \forall i$.

Вспомогательные данные:

$$\nabla L(w_p) = \begin{pmatrix} g_p^{(1)} & \dots & g_p^{(d)} \end{pmatrix} \quad G_p^{(i)} = \sum_{j=1}^p (g_j^{(i)})^2$$

¹ Проще говоря, этот метод используется в машинном обучении

² И называется learning rate

Тогда пусть

$$\eta_p^{(i)} = \frac{\eta}{\sqrt{G_p^{(i)} + \epsilon}}$$

, где $\epsilon \approx 10^{-8}$ — сглаживающий параметр, который позволяет избежать деления на 0.

Тогда правило перехода будет:

$$w_{p+1} = w_p - \eta_p \odot \nabla L(w_p)$$

, где \odot — поэлементное умножение векторов.

6.11 Метод покоординатного спуска

Алгоритм метода:

1. Фиксируем значения всех переменных вектора $x = (x_1 \dots x_n)$, кроме x_i .
2. $f(x_i) \rightarrow \min$ методом одномерной оптимизации (*наиболее популярный метод — золотого сечения*).
3. Проверка критерия остановки:
 - $\|x^{k+1} - x^k\| \leq \varepsilon_1$
 - $\|f(x^{k+1}) - f(x^k)\| \leq \varepsilon_2$

Лекция 9

31 марта

7 Форматы хранения матриц

Определение. Матрица с большим количеством нулевых элементов называется **разреженной**. Такие матрицы хранятся особым образом.

Определение. Не разреженная матрица называется **плотной**.

Формат записи матриц зависит от алгоритма, который будет использовать данную матрицу. Наиболее распространены следующие 4 разреженных формата:

1. Диагональный
2. Ленточный
3. Профильный
4. Разреженный

Мы будем рассматривать эти форматы только в применении к квадратным матрицам. Первые два формата используются реже и мы не будем их рассматривать в лабораторных работах.

Характеристики форматов:

- Учитывается ли симметрия матрицы.
- Используются ли отдельно верхний и нижний треугольники матрицы.
- Требуется ли ускоренный доступ к строкам или столбцам матрицы.

7.1 Диагональный формат

Этот формат используется, когда все ненулевые элементы матрицы находятся на относительно небольшом числе диагоналей.

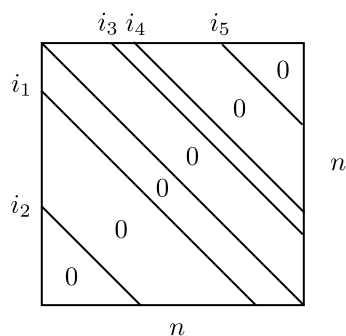


Рис. 9.1: Матрица с ненулевыми элементами на диагоналях,
 $m = 5$

Матрица хранится в виде плотной матрицы $n \times m$, где n — размерность исходной матрицы, а m — количество ненулевых диагоналей. Также необходимо хранить одномерный массив размерностью $m - 1$, где для каждой диагонали указан сдвиг относительно главной диагонали. В примере значения этого массива $(-i_2, -i_1, i_3, i_4, i_5)$.

7.2 Ленточный формат

Этот формат используется, когда все ненулевые элементы матрицы расположены на диагоналях, прилегающих к главной диагонали, т.е. $a_{ij} = 0$, если $|i - j| > k$. При этом k называется **полушириной**, а ширина ленты $m = 2k + 1$.

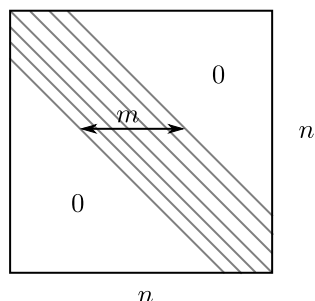


Рис. 9.2: Матрица ленточного типа

Хранить такие матрицы в виде m массивов различных длин не представляется возможным, т.к. требуется быстрый доступ к элементам матрицы.

Иногда главную диагональ хранят отдельно, в зависимости от алгоритма.

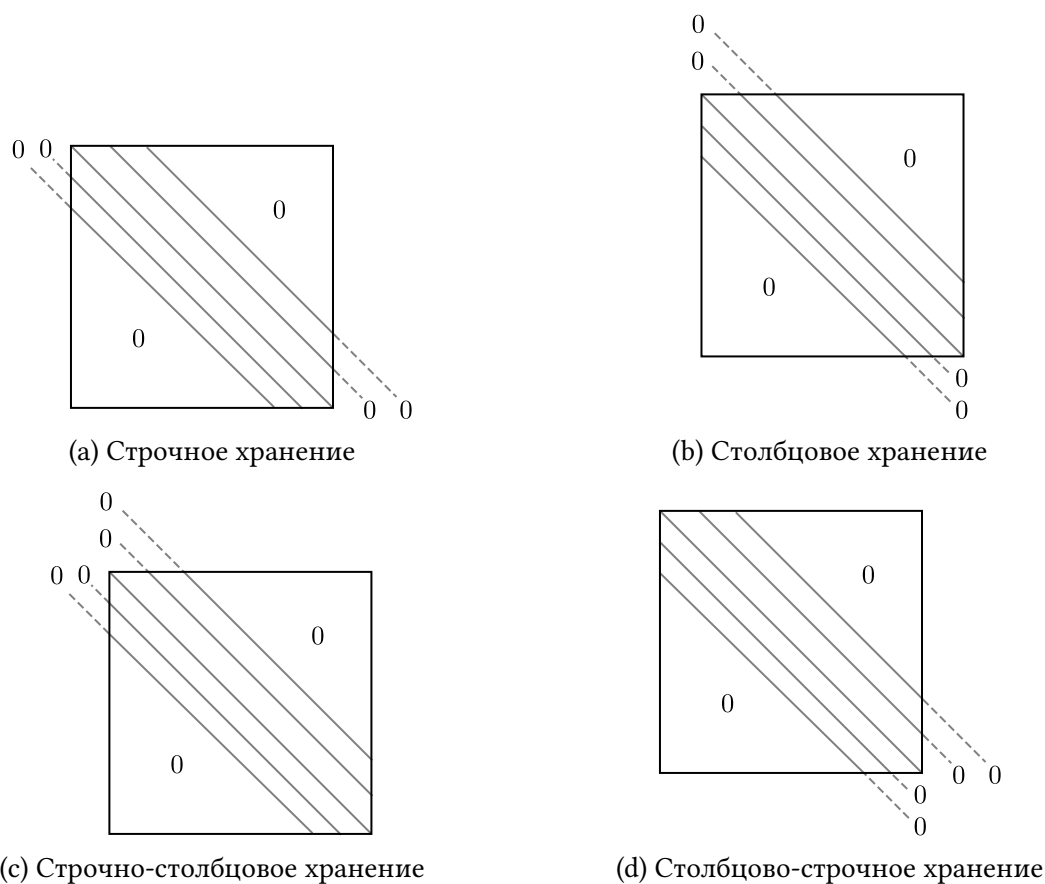
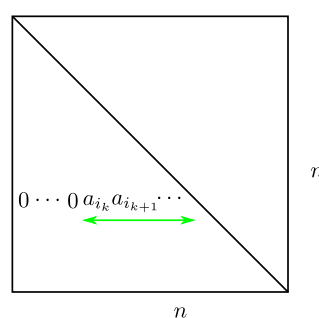


Рис. 9.3: Способы хранения матрицы в ленточном формате

7.3 Профильный формат

Профильные форматы хранения матриц используется, когда матрица не обладает определенной структурой и ненулевые элементы расположены в произвольном порядке, но при этом они сосредоточены у главной диагонали, так что в строке можно выделить **профиль** — часть строки от первого ненулевого элемента в строке до диагонального элемента.

Матрицы ленточного формата — матрицы профильного формата с фиксированным профилем.

Рис. 9.4: Матрица профильного типа.
Зеленым — профиль строки i

Используемые структуры:

- Вещественный массив $di[n]$
- Вещественные массивы:
 - al — элементы нижнего треугольника по строкам
 - au — элементы верхнего треугольника по столбцам.
- Целочисленный массив ia — информация о профиле: $ia[k]$ = индекс *(в нумерации с 1)*, с которого начинаются элементы k -той строки или столбца в массивах al или au .

$ia[n+1]$ = индекс первого незанятого элемента в массивах al и au .

$ia[i+1] - ia[i]$ — значение профиля i -той строки (столбца) нижнего (верхнего) треугольника.

$ia[1] = ia[2] = 1$.

Примечание. Если матрица симметрична по значениям, то $al = au$.

Пример.

$$\begin{bmatrix} a_{11} & & & & & & & & \\ & a_{22} & a_{23} & a_{24} & & & & & \\ & a_{32} & a_{33} & 0 & a_{35} & a_{36} & & & \\ & a_{42} & 0 & a_{44} & a_{45} & 0 & a_{47} & & \\ & & a_{53} & a_{54} & a_{55} & a_{56} & 0 & a_{58} & a_{59} \\ & & a_{63} & 0 & a_{65} & a_{66} & 0 & a_{68} & 0 \\ & & & a_{74} & 0 & 0 & a_{77} & 0 & a_{79} \\ & & & & a_{85} & a_{86} & 0 & a_{88} & 0 \\ & & & & a_{95} & 0 & a_{97} & 0 & a_{99} \end{bmatrix}$$

$$di = \{a_{11}, a_{22}, a_{33}, a_{44}, a_{55}, a_{66}, a_{77}, a_{88}, a_{99}\}$$

$$ia = \{1, 1, 1, 2, 4, 6, 9, 12, 15, 19\}$$

$$al = \{a_{32}, a_{42}, 0, a_{53}, a_{54}, a_{63}, 0, a_{65}, a_{74}, 0, 0, a_{85}, a_{86}, 0, a_{95}, 0, a_{97}, 0\}$$

$$au = \{a_{23}, a_{24}, 0, a_{35}, a_{45}, a_{36}, 0, a_{56}, a_{47}, 0, 0, a_{58}, a_{68}, 0, a_{59}, 0, a_{79}, 0\}$$

Первый элемент для 6-ой строки: $al[ia[6]] = al[6] = a_{63}$

Профиль 6-ой строки: $ia[7] - ia[6] = 9 - 6 = 3$.

7.4 Разреженный формат

Этот формат бывает:

- строчным

- столбцовым
- смешанным: строчно-столбцовым или столбцово-строчным.

Лекция 10

7 апреля

Рассмотрим строчно-столбцовый формат.

Используемые структуры:

1. Вещественный массив di — диагональные элементы.
2. Вещественные массивы al (по строкам), au (по столбцам), хранящие внедиагональные элементы нижнего или верхнего треугольника соответственно.
3. Целочисленный массив ja — номера столбцов (строк) хранимых внедиагональных элементов нижнего или верхнего треугольника матрицы. $ja[j]$ — номер столбца для $al[j]$, номер строки для $au[j]$.
4. Целочисленный массив ia , где $ia[k]$ равен индексу (в нумерации с 1), с которого начинаются элементы k -той строки или столбца в массивах al , au , ja .

Размерность ja , al , au есть $ia[n+1]-1$.

$ia[i] - ia[i-1]$ — количество хранимых внедиагональных элементов i -той строки (столбца) нижнего (верхнего) треугольника.

$ia[1] = ia[2] = 1$.

Пример.

$$\begin{bmatrix} a_{11} & & & & & & & & \\ & a_{22} & a_{23} & a_{24} & & & & & \\ & a_{32} & a_{33} & 0 & a_{35} & a_{36} & & & \\ & a_{42} & 0 & a_{44} & a_{45} & 0 & a_{47} & & \\ & & a_{53} & a_{54} & a_{55} & a_{56} & 0 & a_{58} & a_{59} \\ & & a_{63} & 0 & a_{65} & a_{66} & 0 & a_{68} & 0 \\ & & & a_{74} & 0 & 0 & a_{77} & 0 & a_{79} \\ & & & & a_{85} & a_{86} & 0 & a_{88} & 0 \\ & & & & a_{95} & 0 & a_{97} & 0 & a_{99} \end{bmatrix}$$

$$di = \{a_{11}, a_{22}, a_{33}, a_{44}, a_{55}, a_{66}, a_{77}, a_{88}, a_{99}\}$$

$$ia = \{1, 1, 1, 2, 3, 5, 7, 8, 10, 12\}$$

$$ja = \{2, 2, 3, 4, 3, 5, 4, 5, 6, 5, 7\}$$

$$al = \{a_{32}, a_{42}, a_{53}, a_{54}, a_{63}, a_{65}, a_{74}, a_{85}, a_{86}, a_{95}, a_{97}\}$$

$$au = \{a_{23}, a_{24}, a_{35}, a_{45}, a_{36}, a_{56}, a_{47}, a_{58}, a_{68}, a_{59}, a_{79}\}$$

Для 6-ой строки: $ia[6] = 5$ — начало 6-ой строки в массивах ja и al . $ia[6 + 1] - ia[6] = 7 - 5 = 2$ — количество элементов в 6-ой строке.

Первый элемент: $ja[ia[6]] = ja[5] = 3$, второй элемент: $ja[ia[6] + 1] = ja[5 + 1] = 5$.

8 Решение СЛАУ. Метод Гаусса.

СЛАУ:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2 \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n \end{cases}$$

То же самое, но в матричной форме:

$$Ax = b$$

, где:

- $A = (a_{ij})_{i,j=1}^n$ — вещественные числа
- $b = (b_1 \dots b_n)^T$
- $x = (x_1 \dots x_n)^T$

Эффективность способов решения СЛАУ зависит от структуры и свойств матрицы A , т.е. от размера, обусловленности, симметричности, заполненности и от её профиля.

Рассмотрим прямой ход метода Гаусса. Первый шаг — домножение уравнений на коэффициенты $-\frac{a_{21}}{a_{11}}, -\frac{a_{31}}{a_{11}} \dots -\frac{a_{n1}}{a_{11}}$:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ a_{22}^{(1)}x_2 + \dots + a_{2n}^{(1)}x_n = b_2^{(1)} \\ \vdots \\ a_{n2}^{(1)}x_2 + \dots + a_{nn}^{(1)}x_n = b_n^{(1)} \end{cases}$$

$$a_{ij}^{(1)} = a_{ij} - \frac{a_{i1}}{a_{11}} a_{1j} \quad b_i^{(1)} = b_i - \frac{a_{i1}}{a_{11}} b_1$$

На $n - 1$ шаге метода система будет приведена к следующему виду:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \dots + a_{1n}x_n = b_1 \\ a_{22}^{(1)}x_2 + a_{23}^{(1)}x_3 + \dots + a_{2n}^{(1)}x_n = b_2^{(1)} \\ \vdots \\ a_{nn}^{(n-1)}x_n = b_n^{(n-1)} \end{cases}$$

Дальше производятся обратный ход метода Гаусса:

$$\begin{aligned} x_n &= \frac{b_n^{(n-1)}}{a_{nn}^{(n-1)}} \\ &\vdots \\ x_2 &= \frac{b_2^{(1)} - a_{23}^{(1)}x_3 - \dots - a_{2n}^{(1)}x_n}{a_{22}^{(1)}} \\ x_1 &= \frac{b_1 - a_{12}x_2 - \dots - a_{1n}x_n}{a_{11}} \end{aligned}$$

В общем виде:

$$x_k = \frac{1}{a_{kk}^{(k-1)}} \left(b_k^{(k-1)} - \sum_{j=k+1}^n a_{kj}^{(k-1)} \cdot x_j \right)$$

Алгоритм:

```

1  for  $k = 1 \dots n - 1$ :
2      for  $i = k + 1 \dots n$ :
3           $t_{ik} = a_{ik} / a_{kk}$ 
4           $b_i = b_i - t_{ik} b_k$ 
5      for  $j = k + 1 \dots n$ 
6           $a_{ij} = a_{ij} - t_{ik} \cdot a_{kj}$ 
7   $x_n = b_n / a_{nn}$ 
8  for  $k = n - 1 \dots 1$ 
9       $x_k = \left( b_k - \sum_{j=k+1}^n a_{kj} \cdot x_j \right) / a_{kk}$ 

```

У этого алгоритма есть проблема — арифметика компьютеров не точна. В частности, если a_{kk} мало, то при делении можно получить немалую ошибку. Чтобы бороться с этим, есть модификация:

8.1 Модификация метода Гаусса (*постолбцовый выбор главного элемента*)

Необходимо найти $m \geq k$, где k — номер рассматриваемого шага, а $|a_{mk}| = \max_{i \geq k} \{|a_{ik}|\}$.

- Если $a_{mk} = 0 (\approx \varepsilon)$, однозначного решения нет, остановка алгоритма.
- Если $a_{mk} \neq 0$, меняем местами b_k и b_m ; a_{kj} и a_{mj} при $j = k \dots n$.

При такой замене порядок x_i в общем векторе решения не меняются. Есть и другие модификации, которые его меняют. В таких алгоритмах необходимо поддерживать матрицу перестановок неизвестных и умножить ответ на эту матрицу.

Лекция 11

14 апреля

Прямые методы основаны на разложениях матрицы L , например:

- LU , где L — нижнетреугольная матрица, а U — верхнетреугольная матрица.
- LL^T — метод квадратного корня
- LDL^T , где $L_{ii} = 1$, D — диагональная матрица.

Мы рассмотрим первый.

9 LU -метод

$$\begin{aligned}AX &= b \\LUx &= b \\y := Ux & \qquad (10) \\Ly &= b \qquad (11)\end{aligned}$$

Таким образом, решение задачи сводится к трём этапам:

1. По A получить L, U .
2. Решить (11) прямым ходом метода Гаусса, тем самым найти y .
3. Решить (10) обратным ходом метода Гаусса, тем самым найти x .

Основные временные затраты происходят на первом этапе метода.

$$L = \begin{bmatrix} L_{11} & 0 & 0 & \cdots \\ L_{21} & L_{22} & 0 & \cdots \\ L_{31} & L_{32} & L_{33} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \quad U = \begin{bmatrix} 1 & U_{12} & U_{13} & \cdots \\ 0 & 1 & U_{23} & \cdots \\ 0 & 0 & 1 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

Пример. Красным отмечены вычисляемые на данной итерации элементы:

$$\begin{aligned} A_{11} &= L_{11} \\ A_{21} &= L_{21} \\ A_{12} &= L_{11} \cdot U_{12} \\ A_{22} &= L_{21} \cdot U_{12} + L_{22} \\ A_{31} &= L_{31} \\ A_{13} &= L_{11} \cdot U_{13} \\ A_{32} &= L_{31} \cdot U_{12} + L_{32} \\ A_{23} &= L_{21} \cdot U_{13} + L_{22} \cdot U_{23} \\ A_{33} &= L_{31} \cdot U_{13} + L_{32} \cdot U_{23} + L_{33} \end{aligned}$$

9.1 Алгоритм разложения

$L_{11} = A_{11}$, для i от 2 до n :

$$L_{ij} = A_{ij} - \sum_{k=1}^{j-1} L_{ik} \cdot U_{kj} \quad j \in \overline{1, i-1}$$

$$U_{ji} = \frac{1}{L_{jj}} \left(A_{ji} - \sum_{k=1}^{j-1} L_{jk} \cdot U_{ki} \right) \quad j \in \overline{1, i-1}$$

$$L_{ii} = A_{ii} - \sum_{k=1}^{i-1} L_{ik} \cdot U_{ki}$$

$$U_{ii} = 1$$

10 Дополнительные рассуждения о точности получаемого численного решения

10.1 Близкие к нулю главные элементы

Пример.

$$\begin{pmatrix} 10 & -7 & 0 \\ -3 & 2.099 & 6 \\ 5 & -1 & 5 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 7 \\ 3.901 \\ 6 \end{pmatrix}$$

Точное решение: $x = (0, -1, 1)^T$.

Предположим, что мы решаем эту задачу на ЭВМ с десятичной пятиразрядной арифметикой с плавающей точкой.

Решим обычным методом Гаусса без модификаций.

$$\begin{pmatrix} 10 & -7 & 0 \\ 0 & -1.0 \cdot 10^3 & 6 \\ 0 & 2.5 & 5 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 7 \\ 6.001 \\ 2.5 \end{pmatrix}$$

$$6.001 \cdot 2.5 \cdot 10^3 = 1.5002 \cdot 10^4 \approx 1.5003 \cdot 10^4$$

$$1.5005 \cdot 10^4 \cdot x_3 = 1.5004 \cdot 10^4$$

$$x_3 = \frac{1.5004 \cdot 10^4}{1.5005 \cdot 10^4} = 0.99993$$

$$x_2 = \frac{1.5 \cdot 10^{-3}}{-1.0 \cdot 10^{-3}} = -1.5$$

$$x_1 = -0.35$$

Итого ошибка очень крупная, 0.5 для одного из элементов. Ошибка возникла на шаге исключения, т.к. не использовалась модификация метода.

10.2 Вектор ошибки и невязка

$$\begin{pmatrix} 0.78 & 0.563 \\ 0.457 & 0.330 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0.217 \\ 0.127 \end{pmatrix}$$

x^*

Арифметика трёхразрядная.

Вычисления опущены.

$$x = (1.71, -1.98)^T$$

Определение. Невязка $r = b - Ax$.

$r = (-0.00206, -0.00107)^T$, что, казалось бы, хорошо. Но при этом верное решение $x^* = (1, -1)^T$ на несколько порядков больше отличается от полученного ответа. Таким образом, невязка не всегда показывает точность полученного решения.

Величина ошибки в решении \approx величина решения $\times \text{cond}(A) \times \varepsilon_{\text{машины}}$, где $\text{cond}(A)$ — число обусловленности¹ A .

Пример. Если $\text{cond}(A) = 10^5$, $\varepsilon = 10^{-8}$, то в решении три верных разряда.

10.2.1 Векторные нормы

Примеры:

- Евклидова или 2-норма: $\|x\|_2 = \left(\sum_{i=1}^n |x_i|^2 \right)^{1/2}$
- 1-норма или Манхеттенское расстояние: $\|x\|_1 = \sum_{i=1}^n |x_i|$
- max-норма или ∞ -норма: $\|x\|_\infty = \max_i |x_i|$

Условия на норму:

- $\|x\| > 0$, если $x \neq 0$
- $\|0\| = 0$
- $\|cx\| = |c| \cdot \|x\| \quad \forall c$
- $\|x + y\| \leq \|x\| + \|y\|$

Если матрица A **вырождена**, то решение уравнения $Ax = b$ может не существовать для одних b и не быть единственным для других b . Если же матрица **почти вырождена**, то малые изменения A и b вызовут большие изменения в x .

Определение.

$$M = \max_x \frac{\|Ax\|}{\|x\|} \Rightarrow \|Ax\| \leq M \cdot \|x\|$$

$$m = \min_x \frac{\|Ax\|}{\|x\|} \Rightarrow \|Ax\| \geq m \cdot \|x\|$$

$\frac{M}{m}$ — число обусловленности матрицы.

$$Ax = b$$

¹ Отношение максимального и минимального собственного числа матрицы.

$$A(x + \Delta x) = b + \Delta b$$

Посмотрим, как ошибка в b , обозначенная Δb , влияет на Δx — ошибку в x .

$$\begin{aligned} \|Ax\| &= \|b\| \leq M \cdot \|x\| \\ \|A\Delta x\| &= \|\Delta b\| \geq m \cdot \|\Delta x\| \end{aligned}$$

Тогда при $M \neq 0$:

$$\frac{\|\Delta x\|}{\|x\|} \leq \text{cond}(A) \cdot \frac{\|\Delta b\|}{\|b\|}$$

Свойства числа обусловленности:

1. $\text{cond}(A) \geq 1, \text{cond}(I) = 1$

Если P — матрица перестановок, то $\text{cond}(P) = 1$.

2. $\text{cond}(c \cdot A) = \text{cond}(A)$

3. D — диагональная матрица, тогда $\text{cond}(D) = \frac{\max |d_{ii}|}{\min |d_{ii}|}$

Пример. $D = \text{diag}(0.1), n = 100$

$\det D = 10^{-100}$ — малое число.

При этом $\text{cond}(A) = \frac{0.1}{0.1} = 1$.

Таким образом, если рассмотреть определитель как меру вырожденности, то матрица очень близка к вырожденной, а если рассмотреть число обусловленности, то это не так.

Лекция 12

21 апреля

Пример. Опущено.

10.2.2 Нормы и анализ ошибок

$$\|A\| = \sum_{i=1}^n \sum_{j=1}^n |a_{ij}|$$

$$\|Ax\| \leq \|A\| \cdot \|x\|$$

Можем выразить другим образом:

$$\|A\| = M = \max_{x \neq 0} \frac{\|Ax\|}{\|x\|}$$

$$\|A\| = \max_j \|a_j\|$$

Лемма 1 (результат Уилкинсона). Вычисленное решение x^* удовлетворяет системе $(A + E)x^* = b$, где элементы E имеют уровень ошибок округления.

$$(A + E)x^* = b$$

$$b - Ax^* = Ex^*$$

$$\|b - Ax^*\| = \|Ex^*\| \leq \|E\| \cdot \|x^*\|$$

$$\frac{\|b - Ax^*\|}{\|A\| \cdot \|x^*\|} \leq C \cdot \varepsilon_{\text{машины}}$$

Если A не вырождена, то:

$$x - x^* = A^{-1}(b - Ax)$$

$$\begin{aligned} \|x - x^*\| &\leq \|A^{-1}\| \cdot \|E\| \cdot \|x^*\| \\ \frac{\|x - x^*\|}{\|x^*\|} &\leq C \cdot \|A\| \cdot \|A^{-1}\| \cdot \varepsilon_{\text{машины}} \end{aligned}$$

Т.к. $\|A^{-1}\| = \frac{1}{m}$, то:

$$\text{cond}(A) = \|A\| \cdot \|A^{-1}\|$$

И таким образом:

$$\frac{\|x - x^*\|}{\|x^*\|} \leq C \cdot \text{cond}(A) \cdot \varepsilon_{\text{машины}}$$

Можно заметить, что вычисление $\text{cond}(A)$ требует вычисления обратной матрицы. Это можно несколько упростить, заметив, что если a_j — столбцы A , \tilde{a}_j — столбцы матрицы A^{-1} , то:

$$\text{cond}(A) = \max_j \|a_j\| \cdot \max_j \|\tilde{a}_j\|$$

Тем не менее, такое вычисление примерно утраивает время вычисления. Но на практике точное значение $\text{cond}(A)$ не требуется и используются приближительные оценки.

10.2.3 Оценивание числа обусловленности

$$\|A^{-1}\| = \frac{1}{\min_x \frac{\|Ax\|}{\|x\|}} = \max_x \frac{\|x\|}{\|Ax\|} = \max_y \frac{\|A^{-1}y\|}{\|y\|}$$

Решим систему $Az = y$, тогда

$$\frac{\|z\|}{\|y\|} = \underbrace{\frac{\|A^{-1}y\|}{\|y\|}}_{\text{оценка } \|A^{-1}\|}$$

Но если брать произвольный y , то оценка может быть неточной. Будем использовать такой y , что $A^T y = C$, где C — вектор с компонентами ± 1 .

Пример. Опущено.

11 Дополнительно о градиентных методах

Релаксационная последовательность задается рекуррентно как $x^k = x^{k-1} + \alpha_k u^k$, $k \in N$, $u^k \in E_n$. Условие спуска при этом $\langle \nabla f(x), u \rangle < 0$.

Какое брать α_k ? Такое, чтобы выполнялось следующее:

$$f(x^{k-1}) + \alpha_k u^k \leq (1 - \lambda_k) f(x^{k-1}) + \lambda_k \min_{\alpha \in E} f(x^{k-1} + \alpha u^k) \quad (12)$$

Очевидно, что $\lambda_k \in [0, 1]$. Чтобы оценить это соотношение, используются эвристические приёмы. В частности, если

$$f(x^{k-1} + \alpha_k u^k) \leq f(x^{k-1})$$

, то $\{x_k\}$ будет релаксационной. Несложно заметить, что мы рассмотрели (12) для случая $\lambda_k = 0$.

Если $\lambda_k = 1$, то для нахождения наилучшего значения α_k^* необходимо решить задачу одномерной минимизации, что мы делали в лабораторной работе.

Если $\lambda_k \in (0, 1)$, то:

$$f(x^{k-1}) - f(x^k) \geq \lambda_k (f(x^{k-1}) - f(x^{k-1} + \alpha_k^* u^k))$$

Это равносильно (12) и из этого можно предположить, что λ_k характеризует наименьшую долю из максимально возможного уменьшения $f(x)$ вдоль направления u^k , которое должна обеспечивать релаксационная последовательность $\{x_k\}$.

Будем обозначать антиградиент как $\omega(x) = -\nabla f(x)$

11.1 Метод градиентного спуска

$$x^k = x^{k-1} + \beta_k \underbrace{\frac{\omega^k}{|\omega^k|}}_{u_k}$$

$$\beta_k = \underbrace{\alpha}_{\text{const}} |\omega^k|$$

Тогда можно переписать:

$$x^k = x^{k-1} + \alpha_k \omega^k$$

Один из главных недостатков градиентного спуска состоит в том, что в окрестности стационарной точки \tilde{x} шаг может оказаться слишком большим, и тогда метод “проскочит” \tilde{x} . Шаг также может быть настолько большим, что произойдёт $f(x^k) > f(x^{k-1})$ и последовательность перестанет быть релаксационной. Можно уменьшить шаг, но тогда замедлится сходимость релаксационной последовательности. Все эти проблемы сводятся к задаче оценки возможной величины α , которая обеспечивала бы высокую скорость сходимости без проскакивания стационарной точки.

Теорема 7. Пусть $f(x)$ ограничена снизу и дифференцируема в пространстве E_n , а её градиент удовлетворяет условию Липшица, т.е.:

$$\forall x, y \in E_n \quad |\nabla f(x) - \nabla f(y)| \leq L|x - y|$$

, где $L > 0 - \text{const}$.

Тогда $\{x_k\} : x^k = x^{k-1} + \alpha \omega^k$ с $\alpha \in (0, \frac{2}{L})$ является релаксационной. При этом справедлива оценка:

$$f(x^k) \leq f(x^{k-1}) - \alpha \left(1 - \frac{\alpha L}{2}\right) |\nabla f(x^{k-1})|^2$$

и $|\nabla f(x^k)| \rightarrow 0$ при $k \rightarrow +\infty$.

Мы уже рассматривали схожую теорему для квадратичных функций. В том случае $L = \max \lambda_i$, где $\{\lambda_i\}$ — собственные значения. Эти теоремы согласованы.

Если $f(x)$ удовлетворяет теореме 7, то при $\lambda = \frac{1}{L}$ $\{x_k\}$ — релаксационная последовательность и не происходит “проскакивание” стационарной точки.

$$\|\omega^k\| = \|\nabla f(x^{k-1})\| \leq L \cdot |x^{k-1} - \tilde{x}|$$

Следовательно, при $\alpha \leq \frac{1}{L}$:

$$|x^k - x^{k-1}| = \alpha |\omega^k| \leq |x^{k-1} - \tilde{x}|$$

Пусть $f(x)$ ограничена снизу, а $\{x^k\}$ такое, что $\exists \gamma_0 > 0$:

$$f(x^{k-1}) - f(x^k) \geq \gamma_0 |\omega^k|^2 \quad (13)$$

$$f(x^0) - f(x^m) \geq \gamma_0 \sum_{k=1}^m |\nabla f(x^{k-1})|^2$$

$$\sum_{k=1}^{+\infty} |\nabla f(x^{k-1})|^2 - \text{знакоположительный сходящийся ряд}$$

$\nabla f(x^k) \rightarrow 0$ при $k \rightarrow +\infty$. Таким образом, при построении $\{x_k\}$ если удастся выполнить условие (13), то последовательность градиентов $\rightarrow 0$, а следовательно $\{x_k\}$ будет сходиться к стационарной точке.

Найти значение константы L в реальных функция бывает трудно, если не невозможно. Тогда градиентный метод не даст гарантий, что он сойдётся. В таком случае модифицируем рекуррентное соотношение:

$$x^k = x^{k-1} + \alpha_k \omega^k$$

, т.е. α меняется на каждом шаге. Есть разные методы выбора α , например:

$$\varphi_k(\alpha) := f(x^{k-1} + \alpha \omega^k)$$

$$\varphi'_k(0) = \langle \nabla f(x), \omega^k \rangle = -|\omega^k|^2$$

В окрестности $\alpha = 0$ φ'_k убывает до α_k^* .

Если рассматривать $\alpha_k = \alpha_k^*$, то такой метод называется **исчерпывающий спуск**. При этом если $f(x)$ удовлетворяет теореме 7, то $\{x_k\}$ по исчерпывающему спуску удовлетворяет условиям (13).

$$\begin{aligned}\varphi'_k(\alpha) &= \langle \nabla f(x^{k-1} + \alpha \omega^k), \omega^k \rangle \\ \langle \omega^{k+1}, \omega^k \rangle &= \langle -\nabla f(x^k), \omega^k \rangle = -\varphi'_k(\alpha_k^*) = 0\end{aligned}$$

$$\begin{aligned}||\omega^k||^2 &= \langle \omega^k, \omega^k - \omega^{k+1} \rangle \\ &\leq |\omega^k| \cdot |\omega^k - \omega^{k+1}| \\ &= |\omega^k| \cdot |\nabla f(x^{k-1}) - \nabla f(x^k)| \\ &\leq L \cdot |\omega^k| \cdot |x^{k-1} - x^k| \\ &= L \cdot \alpha_k^* \cdot |\omega^k|^2\end{aligned}$$

Таким образом:

$$\begin{aligned}\alpha_k^* &\geq \frac{1}{L} \\ f(x^{k-1}) - f(x^k) &\geq f(x^{k-1}) - f(\tilde{x}^k) \geq \frac{1}{2L} |\omega^k|^2 \\ \tilde{x}^k &= x^{k-1} + \underbrace{\frac{1}{L}}_{\alpha} \omega^k\end{aligned}$$