Содержание

1 Лекцион 1

Ваш лектор сегодня не смог прийти. Книга Бахвалова, Жидкова, Кобелькова будет рассказана за два семестра.

Почему нечётко всё будет формулироваться с точки зрения математика. В вычислительной математике мало предъявить алгоритм или доказать, что решение есть. Нужно ещё оценить качество этого алгоритма.

1.1 Вычислительная погрешность

Три требования к задаче и алгоритму.

1. Устойчивость задачи.
2. Устойчивость алгоритма.
3. Вычислительные ресурсы.

Если вам дают задачу, сначала смотрите как обычный математик, а затем загибаете пальцы, проверяя эти три пункта.

Что такое устойчивость задачи.

1.1.1 Матрица Уилкинсона

Вот такая матрица 20×20.

$$\begin{pmatrix} 20 & 20 & 0 & 0 \\ 0 & 19 & 20 & 0 \\ & \ddots & \ddots & \ddots \\ 0 & & & \dots 1 \end{pmatrix}$$

Считаем определелитель (на главной диагонали $i!$ на следующей справа по 20), получаем 20!. А теперь пусть нас просят его посчитать, про том, что числа могут чуть-чуть отличаться на ε. Например, левый нижний элемент ε.

$$\det(A_\varepsilon - \lambda I) = (20 - \lambda) \dots (1 - \lambda) - \varepsilon 20^{19} = 0.$$

Тогда $\varepsilon = 20^{-19} \cdot 20! \sim 5 \cdot 10^7$. Тогда $|\lambda|_{\min}(A_\varepsilon) = 0$, $\det A_\varepsilon = 0$.

Чтобы понимать масштаб то, что происходит. Пусть $5 \cdot 10^{-7}$ км. Это пол миллиметра. А что такое $20! \sim 2{,}4 \cdot 10^{18}$ км — млечный путь.

Это был пример неустойчивость задачи. Введение маленького возмущения колоссально меняет результат.

1.1.2 Услойчивость алгоритма

Рассмотрим $f(x) = (x - 10^3)(x - 10^{-3}) = 0$. $x_1 = 10^{-3}$, $x_2 = 10^3$. Ну или

$$x^2 - 2ax + 1 = 0.$$

$2a$ это на самом деле $10^3 + 10^{-3}$. Итого $x_1^{(1)} = a - \sqrt{a^2 - 1}$.

Если говорить в терминах C, мы кладём результат в double. $x_1^{(1)}$ имеет 10 верных цифр. 64 бита для хранения.

$$\left|f(x^{(1)}\right| \sim 10^{-11}.$$

Как меняется переменная $x \sim 10^{\pm 308}$. Машинная точность double exp=1.

```
double eps=1;
while (1+eps<1) eps = eps/2;
```

Если x порядка единицы, то машинная точность $\varepsilon \sim 10^{-16}$.

То есть плохой наш алгоритм, как быть? Домножим на сопряжённое

$$x_1^{(2)} = \frac{1}{a + \sqrt{a^2 - 1}}.$$

В этом случае $x_1^{(2)}$ имеет 17 значащих цифр. При этом $\left|f(x_1^{(2)})\right| \sim 10^{-18}$.

Это был пример неустойчивого алгоритма. С большим количеством операций, скопится неточность.

Рассмотрим $\sum\limits_{i=1}^{10^3} \frac{1}{i^2}$. В каком порядке считать? Слева направо или справа налево? Совпадение получится в 14 знаках, а не 18. Какой более устойчивый: с маленьких чисел к большим.

Поставим абсурдную задачу $\sum\limits_{i=1}^{\infty} \frac{1}{i}$. Всем известно, что ряд расходится, сможем ли мы это зафиксировать? Мы с какого-то момента будем прибавлять к большой накопленной сумме слишком маленькую добавку. Сумма просто не будет меняться.

1.1.3 Вычислительные ресурсы

СЛАУ $Ax = b$. Метод Крамера. Считаем $(n + 1)$ определитель методом миноров, каждый определитель считается за $n! \cdot n$ арифметических действий. В итоге получается $n \cdot (n + 1)!$.

Возьмём $n = 20$ — количество строк; и параллельно $n = 100$. Тогда сложность алгоритма $N(20) \sim 20 \cdot 21 \cdot 20! \sim 10^{21}$ арифметических операций. То есть в процессоре число атомов соизмеримо с числом арифметических операций. Процессором i7 $50 \cdot 10^9 flops$, то есть столько операций в секунду. В году $3 \cdot 10^7$ секунд. То есть нам потребует 650 лет.

Плохая программа в секунду делает больше ошибок, чем человек за всю жизнь.

Дома посчитайте $n = 100$. Посчитайте сколько времени займёт на Ломоносове. Сравните с возрастом вселенной.

В вычислительной математике важно не число, а понимание того, что вы сделаете.

1.2 Теория приближения функции одной переменной многочленами

Приближаем $f(x)$, $x \in \mathbb{R}$. Очень важно правильно сформулировать задачку.

Дано. На отрезочке $[a, b]$ рассыпан набор точек $a \leqslant< x_1 < x_2 < \cdots < x_n \leqslant b$. Имеем f — гладкая на $[a, b]$. Требуется построить полином.

Найти $p(x)\colon p(x_i) = f(x_i)$ для всех $i = 1, \ldots, n$.

Можно поставить другую задачу: не точное совпадение, а близкое. Получится по-другому.

Имеем $p_{n-1}(x) = a_0 + a_1 x + \cdots + a_{n-1} x^{n-1}$. Описываем систему

$$\begin{pmatrix} 1 & x_1 & \ldots & x_1^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \ldots & x_n^{n-1} \end{pmatrix} \begin{pmatrix} a_0 \\ \vdots \\ a_{n-1} \end{pmatrix} = \begin{pmatrix} f(x_1) \\ \vdots \\ f(x_n) \end{pmatrix}$$

При этом условие $\det A \neq 0$ может нарушаться из-за машинной точности. Формально определитель Вандермонда не ноль, но строки могут быть очень близки при большом количестве точек. Получится в итогу нулевая функция, быстро уходящая в 1 в правом конце отрезка.

Лемма 1.1. *Пусть $x_1 < x_2 < \cdots < x_{n_1} < x_n$, заданы $\left\{f(x_i)\right\}_{i=1}^n$. Тогда*

$$\exists!\ P_{n_1}(x),\ \deg P_{n-1} = n - 1\colon P_{n-1}(x_i) = f(x_i).$$

Мы это показали. Но не надо формально систему уравнений на коэффициенты считать.

Рассмотрим пример: функция Рунге на отрезке $[-1,1]$. Берётся равномерная сеточка $x_i = -1 + (i-1)\cdot h$, $x_n = 1$. Функция сама $f(x) = \frac{1}{1+25x^2}$. Многочлен $P_{n-1}(x)$ существует и единственный. Любопытно с помощью какого-нибудь пакета посмотреть, как это получается. В отрезке есть мёртвая зона, где при $n \to \infty$ нет будет сходимости по норме Чебышёва. Но где-то будет хорошо.

Второй пример. $f(x) = |x|$. В этом случае $\|f - p_{n-1}\|_C \not\to 0$ на равномерной сетке.

Перед тем как приближать, нужно понять, а хорошо ли эту функцию можно приблизить полиномами.

Теорема 1.1 (Фабера)**.** *Для произвольной таблицы узлов интерполяции*

$$\begin{array}{ccccc} x_1^{(1)} & & & & \\ x_1^{(2)} & x_2^{(2)} & & & \\ x_1^{(3)} & x_2^{(3)} & x_3^{(3)} & & \\ \vdots & \vdots & \vdots & \ddots & \end{array}$$

Все $x_i^{(j)} \in [a,b]$. *Для такой таблицы существует* $f \in C[a,b]\colon \|f - p_{n-1}\|_C \not\to 0$.

Доказывать не будем, нам не надо. Хотя функция строится конструктивно.

Замечание. Пусть $f \in C^\infty[a,b]$, $\{x_i\}_{i=1}^n$ — чебышёвские узлы. Тогда $\|f - p_{n-1}\|_C \leqslant Cq^n$, где $q < 1$, $c = c(f)$.

1.3 Интерполяционный полином в форме Лагранжа

Он будет записывать не через коэффициенты. Ведь $1, x, x^2, \ldots$ почти линейно зависимы. Мы будем брать базис ортогональных многочленов.

Пусть $\Phi_i(x) = \prod\limits_{j\neq i} \frac{(x-x_j)}{(x_i-x_j)}$. Как выглядит первая $\Phi_1(x) = \frac{(x-x_2)\ldots(x-x_1)}{(x_1-x_2)\ldots(x_1-x_n)}$.

Тогда $p_{n-1}(x) = \sum\limits_{i=1}^{n} f(x_i)\cdot\Phi_i(x)$. Обозначение $L_n(x)$. По существованию и единственности это тот же полином.

Замечание $\deg\big(L_n(x)\big) = n-1$. Этот момент уточните с лектором. Может быть, он обозначает L_{n-1}.

Замечание. Оптимальное вычисление. Если есть полином $p_{n-1}(x) = a_0 + a_1x + \cdots + a_{n-1}x^{n-1}$. Нужно вычислить значение в точке.

```
p = 0;
for (i=0; i<n; i++){
  p = p + a[i] * pow(x,i);
}
```

Это долго. Нужно домножать предыдущий $x_n = x_n \cdot \overline{x}$ на точку, а не считать заново степень. Это уже лучше.

Есть на самом деле ещё более быстрый способ. Оптимальная схема: схема Горнера.

$$p_{n-1}(x) = a_0 + x\Big(a_1 + x\big(a_2 + \cdots + x(a_{n-2} + xa_{n-1})\big)\Big).$$

Это способ имеет сложность $O(n)$. Такое алгоритм достаточно устойчив к погрешности.

А как вычислять значение многочлена Лагранжа.

$$L_n(x) = f(x_1)\cdot\frac{(x-x_2)\ldots(x-x_n)}{(x_1-x_2)\ldots(x_1-x_{n-1}} + \ldots$$

Лучше делить сначала $\frac{x-x_2}{x_1-x_2}$, потом следующую пару. Так меньше ошибок накопится.

Теорема 1.2. *Пусть* $f \in C^n[a,b]$. *Соответственно* $a = x_1 < \cdots < x_n = b$. *Тогда*

$$\forall\ x \in [a,b] \exists\ \xi = \xi(x) =\in [a,b]\colon f(x) - L_n(x) = \frac{f^{(n)}\big(\xi(x)\big)}{n!}\omega_n(x),$$

где $\omega_n(x) = (x-x_1)\ldots(x-x_n)$.

Доказательство. Пошаговое

1. Пусть $x = x_i$, $i = 1..n$. Тогда $0 = 0$ верно.
2. Пусть $x \in [a,b]$, $x \neq x_i$. Рассмотрим функцию

$$\varphi(t) = f(t) - L_n(t) - k\omega_n(t), \tag{1}$$

где коэффициент $k\colon \varphi(t)\big|_{t=x} = 0$. Отсюда следует, что $k = \frac{f(x)-L_n(x)}{\omega_n(x)}$.

Тогда $\varphi(t)$ обращается в нуль в точках $t = x, x_1, x_2, \ldots, x_n$. Отсюда следует, что $\varphi'(t)$ обращается в нуль в n точках, $\varphi^{(n)}(t)$ образается в ноль в некоторой точке $\xi = \xi(x) \in [a,b]$.

Возьмём теперь равенство (1) и продифференцируем

$$\varphi^{(n)}(t)\big|_{t=\xi} = f^{(n)}(t)\big|_{t=\xi} - 0 - kn!.$$

Отсюда $k = \frac{f^{(n)}(\xi)}{n!}$. И ведь $k = \frac{f(x)-L_n(x)}{\omega_n(x)}$ и теорема доказана.

■

Следствие 1.1. *В условиях теоремы верна оценка*

$$\|f - L_n\|_C \leqslant \frac{\|f^{(n)}\|_C}{n!}\|\omega_n\|_C.$$

1.4 Константа Лебега

Пусть $L_n(x) = \sum\limits_{i=1}^{n} f(x_i)\Phi_i(x)$. Пусть изменится $f(x_i)$ на $f(x_i) + \varepsilon_i$. Константой Лебега называют величину

$$\lambda_n = \max_{x\in[a,b]} \sum_{i=1}^{n} \big|\Phi_i(x)\big|.$$

Если $x_i = x_{i-1} + h$ — равномерные узлы, то $c_1 \cdot \frac{2^n}{n^{3/2}} \leqslant \lambda_n \leqslant c_2 \cdot 2^n$.
Если $\{x_i\}$ — чебышёвские узлы, то $\lambda_n c_3 \ln n$. А логирифм это почти константа.

2 Лекция 4

В прошлый раз мы с вами остановились на чём. Имеется гильбертоо пространство R со скалярным произведением (f, g) и, соответственно, норма $\|f\| = \sqrt{(f, f)}$. Имеелась линейно независимая система $g_1, \ldots, g_n$, какая-то функция f. Нужно было построить элемент наилучшего приближения.

$$f \sim \sum_{j=1}^{n} c_j g_j.$$

Определение 2.1. *Элементом наилучшего приближения называется кобминация* $\sum\limits_{j=1}^{n} c_j^0 g_j$, *для которой*

$$\left\| f - \sum_{j=1}^{n} c_j^0 g_j \right\| = \min_{c_1,\ldots,c_n} \left\| f - \sum_{j=1}^{n} c_j g_j \right\|.$$

Находится из системы $G\overline{c} = \overline{F}$, где $G_{ij} = (g_i, g_j)$, $F_i = (f, g_i)$.

Вернёмся с высоких материй. Пусть работаем в $\mathcal{L}_2[-1, 1]$. Здесь $(f, g) = \int\limits_{-1}^{1} fg\,dx$.

Вводили определение числа обусловленности матрицы $\mathrm{cond}(A) = \|A\| \cdot \|A^{-1}\|$. Если выполняется положительная определённость и симметричность $A = A^T > 0$, то есть второе определение

$$\mathrm{cond}_2(A) = \frac{\lambda_{\max}(A)}{\lambda_{min}(A)}.$$

При этом $\|A\|_2 = \sup\limits_{x\neq 0} \frac{\|Ax\|_2}{\|x\|_2}$.

Если за базис брать многочлены, число обусловленности будет неограниченно очень быстро расти. С вычислительной точки зрения это не годится.

Будем использовать функции с конечным носителем. Начало будет не очень жизнерадостным. Мы разобьём наш отрезок на n точек $x_1, x_2, \ldots, x_n$. Будем считать, что сеточка равномерная $x_{i+1} - x_i = h$. В качестве i-й функции $g_i(x)$ возьмём такую, что $g_i(x_i) = 1$, g_i линейная на $[x_{i-1}, x_i]$ и линейная на $[x_i, x_{i+1}]$. Вне отрезка $[x_{i-1}, x_{i+1}]$ $g_i \equiv 0$ и g_i непрерывна.

Матрица $G_{ij} = (g_i, g_j)$ будет три-диагональна, то есть считать надо только $G_{i,i\pm1}$, G_{ii}, остальные нули.

Считаем для

$$g_i = \begin{cases} 1 - \frac{x}{h}, & x \in [0, h]; \\ 1 + \frac{x}{h}, & x \in [-h, 0]. \end{cases}$$

скалярное произведением

$$(g_i, g_i) = 2\int\limits_0^h (1 - \frac{x}{h})^2\,dx = \frac{2}{h}3.$$

Ну и по краям отрезок интегрирования напополам обрезан. Значит, $(g_1, g_1) = (g_n, g_n) = \frac{h}{3}$.

Теперь считаем

$$(g_i, g_{i+1}) = \int\limits_0^h \left(1 - \frac{x}{h}\right)\frac{x}{h}\,dx = \frac{h}{6} = (g_i, g_{i-1}).$$

Таким образом, получается матрица G

$$\begin{pmatrix} h/3 & h/h & 0 & \dots & \\ h/6 & 2h/3 & 0 & \dots & \\ 0 & \ddots & \ddots & \ddots & \dots \end{pmatrix}$$

Правая часть $F_i = \int\limits_{-1}^{1} f g_i\,dx = \int\limits_{x_i-h}^{x_i+h} f(x)g_i(x)\,dx$. Мы может и не сможем интеграл взять. Давайте приблизим

$$F_i \approx f(x_i)\int\limits_{x_i-h}^{x_i+h} g_i(x)\,dx.$$

Насколько это будет хорошее приближение — упражнение.

Можно искать второе число обусловленности. Искать собственные значения. Нам нужны оценка сверху и оценка снизу. Обычно теорема, которую сейчас назову, почему-то перестала входить в число обязательных на мехмате.

2.1 Круги Гершгорина

Я её в упрощённой постановке сформулирую. Пусть есть Матрица A вещественная и $n \times n$. Так вот все её собственные значения $\lambda_i(A) \in \bigcup\limits_{i=1}^{n} K_i(R_i)$, где круг $K_i(R_i) = \left\{z\colon |z - a_{ii}| \leqslant \sum\limits_{i\neq j} |a_{ij}|\right\}$. Центр в a_ii, радиус есть сумма модулей всех недиагональных элементов.

В нашем случае все собственные значения вещественные, так как матрица симметричная. Первой и последней строке соответствует круг $[h/6, h/3]$, остальным точкам соответствует круг $[h/3, h]$. То есть фактически мы можем сказать, что собственные значения нашей матрицы не такие уж ужасные.

$$h/6 \leqslant \lambda(G) \leqslant h.$$

Стало быть её число обусловленности

$$\mathrm{cond}_2(G) = \frac{\lambda_{\max}}{\lambda_{\min}} \leqslant 6.$$

Это число, оно не зависит от количества элементов в базисе. Это ситуация почти идеальная. При современном подходе к задачам это идеал, к которому нужно стремиться. Ошибка максимум в 6 раз увеличится.

Теперь такой вопрос. Как эту систему решать.

2.2 Решение систем линейных уравнений с три-диагональной матрицей

Решаем систему $Ax = b$. Будем считать, что матрица A у нас вот такая

$$A = \begin{pmatrix} \alpha_1 & \beta_1 & 0 & 0 & \dots \\ \gamma_2 & \alpha_2 & \beta 2 & 0 & \dots \\ 0 & \gamma_3 & \alpha_3 & \beta_3 & \dots \\ 0 & \ddots & \ddots & \ddots & \ddots \end{pmatrix}.$$

Давайте наложим дополнительные требования, которые могут показаться неестественными. Но на самом деле они вполне адекватные. Пусть

- $|\alpha_i| \geqslant |\gamma_i| + |\beta_i|$ для $i = 2, \dots, n-1$
- $\alpha_1 = \alpha_n = 1$;

- $|\beta_1| \leqslant 1$, $|\gamma_n| \leqslant 1$, $|\beta_1| + |\gamma_n| < 2$.
- $\gamma_i, \beta_i \neq 0$.

Типовой пример такой задачи: $\alpha_i = 2$, $\beta_i = \gamma_i = -1$. Она у нас возникнет, когда мы будем считать численно вот такую краевую задачу

$$\begin{cases} -u''(x) = f(x); \\ u(0) = u(1) = 0. \end{cases}$$

Итак, задачу поставили. Давайте её решать. Перепишем систему вот в таком виде

$$\begin{cases} \alpha_1 x_1 + \beta_1 x_2 = b, \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ \gamma_i x_{i-1} + \alpha_i x_i + \beta_i x_{i+1} = b_i; \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ \gamma_n x_{n-1} + \alpha_n x_n = b_n. \end{cases}$$

Это будет некая модификация метода Гаусса. Так её обычно описывают. Но я буду несколько проще.

На первом этапе из каких-то соображений нахожу $\{A_i, B_i\}$, для которых $x_i = A_{i+1}x_{i+1} + B_{i+1}$. Это прямой метод прогонки. A_i, B_i называются прогоночными коэффициентами. На втором этапе обратная прогонка. Берём x_n и находим x_i.

Но не всё так просто. Нужно, чтобы $|A_i| \leqslant 1$. Если $A_i \geqslant A > 1$, будет вот что. x_n у нас посчитано будет с какой-то ошибкой. Каждый раз эта ошибка будет умножаться на что-то больше единицы.

Теперь давайте подумаем, как такой набор соотношений получить. Вариантов на самом деле множество.

$$x_{i-1} = A_i x_i + B_i$$

подставим в i-е уравнение

$$\gamma_i A - i x_i + \gamma_i B_i + \alpha_i x_i + \beta_i x_{i+1} = b_i.$$

Мы хотим видеть $x_i = A_{i+1}x_{i+1} + B_{i+1}$. Значит, надо выразить x_i.

$$x_i = -\frac{\beta_i}{\gamma_i A_i + \alpha_i} x_{i+1} + \frac{b_i - \gamma_i B_i}{\gamma_i A_i + \alpha_i}.$$

Мы пока не задумываемся, почему на эти знаменатели можно делить.

$$A_{i+1} = \frac{-\beta_i}{\gamma_i A_i + \alpha_i} \quad B_{i+1} = \frac{b_i - \gamma_i B_i}{\gamma_i A_i + \alpha_i}.$$

При этом $A_2 = -\frac{\beta_1}{\alpha_1}$, $B_2 = \frac{b_1}{\alpha_1}$. Осталось определить x_n.

$$x_n = -\frac{B_n - \frac{b_n}{\gamma_n}}{A_n + \frac{\alpha_n}{\gamma_n}}.$$

Нам осталось показать, что знаменатели у нас не нули и что $|A_i| \leqslant 1$. Начнём мы парадоксальным образом. Мы знаем, что $\alpha_1 = \alpha_2 = 1$; значит, $A_2 = -\beta_1$, но $|\beta_1| \leqslant 1$ Значит, $|A_2| \leqslant 1$. База индукции есть.

Пусть $|A_i| \leqslant 1$. Покажем, что и $|A_{i+1}| \leqslant 1$. Так как $\beta_i \neq 0$, нужно доказать, что знаменатель не меньше числителя, что и докажет его отличие от нуля

$$|\gamma_i A_i + \alpha_i| - |\beta_i| \geqslant |\alpha_i| - |\gamma_i||A_i| - |\beta_i| \geqslant |\beta_i| + |\gamma_i| - |\gamma_i||A_i| - |\beta_i| = |\gamma_i|\big(1 - |A_i|\big) \geqslant 0.$$

Таким образом, мы получили, что $|\gamma_i A_i + \alpha_i| \geqslant |\beta_i| > 0$. То есть мы и поделили хорошо и получили устойчивость алгоритма в целом. Осталось про знаменатель x_n поговорить.

$$x_n = \frac{B_n - \frac{b_n}{\gamma_n}}{A_n + \frac{1}{\gamma_n}} = \frac{\ldots}{\gamma_n A_n + 1}.$$

У нас две возможности.

- $|\beta_i| < 1$. В этом случае $|A_1| < 1$, отсюда все $|A_i| < 1$ и последняя $|A_n| < 1$. Отсюда $A_n\gamma_n + 1 \neq 0$.
- $|\gamma_n| \leqslant 1$. Тут ещё приятнее. $|A_n| \leqslant 1$ и $|\gamma_n| < 1$. Опять всё хорошо.

А теперь наш второй этап завишем:

$$x_i = A_{i+1}x_{i+1} + B_{i+1}.$$

Посчитаем сложность алгоритма на двух этапах в целом. $8n + O(1)$ — число действий пропорциональна числу неизвестных. Это фактически идеальная ситуация. Это фактически как из массива в массив переложить.

2.3 Многочлен наилучшего равномерного приближения

Задача ставится следующим образом. Есть отрезок $[a, b]$, $x \in [a, b)$. Есть функция $f(x)$. Норма в нашим пространстве $\|g\|_C = \sup\limits_{[a,b]} \big|g(x)\big|$. Будем пытаться приблизить функцию многочленами

$$f(x) \sim Q_n(x).$$

Наша цель, чтобы при фиксированным n мы построили такой многочлен степени не выше n, для которого

$$\Delta_n(f) = \|f - Q_n^0\|_C \leqslant \min_{Q_n} \|f - Q_n\|_C.$$

Так как пространство у нас линейное нормированное, такой многочлен будет существовать. Вот будет ли он единственный.

Сразу расстрою: общего адекватного алгоритма нет. Даётся только ради традиций, чтоы у преподавателей старой закалки не было культурного шока, что вы это не знаете.

Я сформулирую сразу утверждение, которое в следующий раз будем доказывать. В следующей лекции вы впечатлитесь масштабностью задачи, которая перед нами стоит.

Теорема 2.1 (Чебышёва)**.** *Пусть $f \in C[a, b]$. Тогда два утверждения равносильны:*

1. *$Q_n(x)$ — МНРП степени n для f на $[a, b]$;*
2. *Существуют $(n + 2)$ точки $\{x_i\} \subset [a, b]$, для которых $x_0 < x_1 < \cdots < x_{n+1}$ и*

$$f(x_i) = Q_n(x_i) = \alpha(-1)^i \|f - Q_n\|_C, \quad j = 0, \ldots, n + 1,$$

$\alpha = 1$ или $\alpha = -1$ для всех i.

3 Лекция 5

У нас есть отрезок $[a, b]$. Есть функция $f(x)$, где $x \in [a, b]$. Есть натуральное n. $Q_n^0(x)$ — многочлен наилучшего равномерного приближения (МНРП) степени не выше n, если

$$\|f - Q_n^0\|_C \leqslant \|f - Q_n\|_C, \quad \forall Q_n.$$

У нас есть обозначение такое. Так как Q_n^0 (МНРП) существует, то $\|f - Q_n^0\| = \Delta_n(f)$.

Мы в прошлый раз сформулировали теорему о необходимом и достаточном условии.

Теорема 3.1 (Чебышёв)**.** *Q_n — МНРП для $f \in C[a, b]$, если и только если*

$$(n + 2)\colon x_0 < x_1 < \cdots < x_{n+1},\ x_i \in [a, b], \quad f(x_i) - Q_n(x_0) = (-1)^i \alpha \cdot \|f - Q_n\|_C.$$

Причём $\alpha = 1$ или -1 сразу для всех i.

Точки x_i называются точками чебышёвского альтернанса.

Доказательство. Докажем только справа налево. Слева направо муторно и неприятно. Непрерывность в основном нужна слева направо.

Теорема 3.2 (Валле-Пуссейна)**.** *Пусть у нас имеется некоторая функция $f(x)$ и многочлен степени n $Q_n(x)$. Функция определена на $[a, b]$. Пусть существуют $n + 2$ точки на $[a, b]$ $x_0 < x_1 < x_2 < \cdots < x_{n+1}$, такие, что*

$$\operatorname{sgn}\big(f(x_i) - Q_n(x_i)\big) \cdot (-1)^i = \text{const}\,.$$

Тогда отсюда следует, что $\Delta_n(f) := \|f - Q_n^0\| \geqslant \min\limits_i \big|f(x_i) - Q_n(x_i)\big| =: \mu$.

Такое вот странное утверждение. Потом на самом деле из него всё будет следовать.

Доказательство. Если $\mu = 0$, всё доказано. Пусть $\mu > 0$. Рассмотрим знак вот такой вот штуки

$$\operatorname{sgn}\Big(\big(f(x) - Q_n(x)\big) - \big(f(x) - Q_n^0(x_i)\big)\Big).$$

Предположим, что $\Delta_n(f) < \mu$. Появился набор точек, в которых разность больше чем норма. Поставлю в нашем выражениии x_i

$$\operatorname{sgn}\Big(\big(f(x_i) - Q_n(x_i)\big) - \big(f(x_i) - Q_n^0(x_i)\big)\Big) =$$

Знак этого выражения определяется только следующей величиной

$$\operatorname{sgn}\big(f(x_i) - Q_n(x_i)\big) = (-1)^i.$$

Это по условию теоремы. Но при этом эта штука разность двух многочленов. Многочлен в $n+2$ точек имеет разные знаки, значит, $n+1$ корень. Следовательно это нулевой многочлен, то есть $Q_n(x) = Q_n^0(x)$. Значит, понятно, что есть неравенство

$$\Delta_n(f) \leqslant \big|f(x_i) - Q_n(x_i)\big|.$$

Вопрос: нужна ли в этой теореме непрерывность f. □

Доказываем справа налево теорему Чебышёва. Считаем, что у нас есть тот безумный набор $n+2$ точек. В каждой точке достигается норма и знак чередуется. Обозначим $\|f - Q_n\|_C = L$. В точках этого самого альтернанса мы получим, что

$$\big|f(x_i) - Q_n(x_i)\big| = L \overset{3.2}{=} \mu \leqslant \Delta_n(f) \leqslant \|f - Q_n\| = L.$$

Раз неравенство замкнулось, Q_n это тоже МНРП. ■

Теперь я хочу с помощью этой теорему Чебышёва доказать что-то полезное, а именно единственность.

Утверждение 3.1. *Есть функция $f \in C[a,b]$. Зафиксировали какое-то n. Значит, у нас есть $\Delta_n(f)$. Пусть существуют два многочлена Q_n^k $(k = 1, 2)$, на которых эта штука достигается*: $\|f - Q_n^k\| = \Delta_n(f)$.

Тогда $Q_n^1 = Q_n^k$.

Доказательство. Рассмотрим $\left\|f - \frac{Q_n^1 + Q_n^2}{2}\right\| \leqslant \frac{1}{2}\|f - Q_n^1\| + \frac{2}{1}\|f - Q_n^2\| = \Delta_n(f)$. Меньше в этом неравенстве мы получить не можем. Значит, $\frac{Q_n^1 + Q_n^2}{2}$ — МНРП, он не хуже остальных, работает теорема Чебышёва, существуют точки альтернанса.

$$\left|\frac{1}{2}\big(Q_n^1(x_i) + Q_n^2(x_i)\big) - f(x_i)\right| = \Delta_n(f).$$

Чуть-чуть пересоберём скобки

$$\Big|\big(Q_n^1(x_i) - f(x_i)\big) + \big(Q_n^2(x_i) - f(x_i)\big)\Big| = 2\Delta_n.$$

Значит, обе разности одного знака и равны по модулю Δ_n. Таким образом, в $n+2$ точках многочлены Q_n^1 и Q_n^2 совпадают. Этого с избытком хватает, чтобы многочлены совпадали полностью. ■

Рисовать примеры не буду. Это вы сделаете на семинарах. Лучше кое-что ещё докажу.

Пусть $x \in [-1,1]$ и функция $f(x) = -f(-x)$, Q_n^0 — МНРП. Покажем, что $Q_n^0(-x) = -Q_n^0(x)$.

У нас $\forall\ x \in [-1,1]\ \ \big|f(x) - Q_n^0(x)\big| \leqslant \Delta_n(f) = \|f - Q_n^0\| = \min\limits_{Q_n} \|f - Q_n\|$. Подставим $f(-x)$, тоже должно работать

$$\big|f(-x) - Q_n^0(-x)\big| \leqslant \Delta_n(f).$$

Минус можно из f вынести

$$\Big|f(x) - \big(-Q_n^0(-x)\big)\Big| \leqslant \Delta_n(f).$$

Значит, это тоже многочлен наилучшего приближения. А мы установили единственность.

Что ещё можно получить бесплатно. Пусть f гладкая. По теореме Чебышёва разность $f(x) - Q_n^0$ меняет знак в $n+2$ точке. Значит, $\exists\ y_1, \ldots, y_{n+1}\colon f(y_i) = Q_n^0(y_i)$. То есть наш многочлен наилучшего приближения является интерполяционным многочленом по $n+1$ точке. Мы можем воспользоваться старыми результатами

$$\|f - Q_n^0\|_C \leqslant \|f - L_{n+1}\|_C \leqslant \frac{\|f^{(n+1)}\|_C (b-a)^{n+1}}{(n+1)!\,2^{2n+1}}.$$

Вот получили оценку сверху. Давайте закончим приближать многочленами.

4 Быстрое дискретное преобразование Фурье

Мне надо сейчас обозначить некий алгоритм. А воспользуемся этими знаниями мы только в мае.

Пусть у нас имеется некоторая функция периодическая с периодом единица, то есть $f(x+1) = f(x)$. Её в принципе можно разложить в ряд Фурье, то есть написать вот так

$$f(x) = \sum_{k=-\infty}^{\infty} a_k \exp(2\pi i k x), \qquad \sum_k |a_k| < \infty.$$

Зафиксируем $N > 0$. И будем рассматривать точки $x_l = \frac{l}{N}$, где $k \in \mathbb{Z}$. Для дальшейшего, чтобы не писать $f(x_l)$, обозначим $f(x_l) = f_l$. Тогда из представления ряда Фурье совершенно грандиозным образом можно привести подобные слагаемые.

Пусть $k_2 - k_1 = kN$. Тогда $k_2 x_l - k_1 x_l = k\frac{Nl}{N} = kl$. Соответственно

$$\exp(2\pi i k_1 x_l) = \exp(2\pi i k_2 x_l).$$

Таким образом,

$$f_l = \sum_{k=0}^{N-1} A_k \exp(2\pi i k x_l), \qquad A_k = \sum_{p=-\infty}^{\infty} a_{k+pN}.$$

Получили конечное разложение, однако сами коэффициенты A_k — это, конечно, целая история.

Вопрос, если есть значения функции в узлах, можем ли мы посчитать A_k и как не считать бесконечную сумму. И обратный вопрос: как, зная A_k, восстановить исходную функцию f. Я расскажу несколько приёмов.

Пусть есть отрезок $[0, 1]$. И у нас есть такая сеточка $x_k = \frac{k}{N}$, $k = 0, \ldots, N-1$. Будем рассматривать всевозможные сеточные функции, то есть определённые на такой сеточке. Введём скалярное произведение

$$(f, g) = \frac{1}{N} \sum_{k=0}^{N-1} f_k \overline{g}_k.$$

Для этого скалярного произведения существует ортонормированная система функций. Давайте эту систему предъявим.

$$g^k(x_l) = \exp(2\pi i k x_l), \quad 0 \leqslant k < N.$$

давайте покажем, что эта система функций — то, что нам надо. Возьмём две функции и скалярно перемножим.

$$(g^k, g^j) = \frac{1}{N} \sum_{l=0}^{N-1} \exp\left(2\pi i \frac{k-j}{N} l\right).$$

Если $k = j$, то каждая exp даёт единичку, всего и N, в сумме дают единичку. Пусть $k \neq j$. Тогда перед нами геометрическая прогрессия

$$(g^k, g^j) = \frac{1}{N} \frac{\exp\big(2\pi i(k-j)\big) - 1}{\exp\left(2\pi i \frac{(k-j)}{N}\right) - 1} = 0.$$

Из этого всего мы сейчас получим прямое и обратное преобразования Фурье.

$$f_l = \sum_{k=0}^{N-1} A_k \exp(2\pi i k x_l) = \sum_{k=0}^{N-1} A_k g^k(x_l).$$

Это по сути и есть обратное преобразование Фурье: знаем A_k, восстанавливаем Функцию. Теперь хотим прямое. Умножим нашу функцию скалярно на g^j.

$$A_j = (f, g^j) = \frac{1}{N} \sum_{l=0}^{N-1} f_l \exp\left(2\pi i j x_l\right).$$

На следующей лекции мы разберём алгоритм, который позволяет это преобразование делать быстро.

5 Лекция 6

В прошлый раз мы разговаривали о дискретном преобразовании Фурье. Сегодня мы с вами разберём, что такое быстрое дискретное преобразование Фурье. Ускорение вычисления. Будем исследовать прямое преобразование Фурье. Нам надо вычислить

$$A_k = \frac{1}{N} \sum_{j=1}^{N-1} f_j \exp\left(-2\pi i \frac{kj}{N}\right), \quad k = 0, \ldots, N-1.$$

У нас тут N сумм по N слагаемых. Сложность N^2. Можно ли ускорить этот процесс. Идея тут какая: мы попытаемся среди всех этих экспонент будем приводить подобные.

Мы рассмотрим случай, когда $N = p_1 \cdot P_p$, причём $p_1, p_2 \neq 1$. Тогда $k = k_1 + p_1 k_2$, $j = j_2 + p_2 j_1$ (нумерация такая себя оправдает позже). Здесь $0 \leqslant k, j \leqslant p_1 - 1$, $0 \leqslant k_2, j_2 \leqslant p_2 - 1$ (уже какой-то смысл нумерации просматривается).

Теперь берём эти разложения k и j подставим в эту сумму

$$A_k = \frac{1}{N}\sum_{j_1=0}^{p_1-1}\sum_{j_2=0}^{p_2-1} f_{j_2+p_2 j_1}\exp\left(-2\pi i\frac{(k_1+p_1k_2)(j_2+p_2j_1)}{N}\right)$$

Я попробую в аргументе экспоненты (поделённом на $-2\pi i$) выделить целую часть. Будем в сторонке вычислять вспомагательный результат, чтобы было понятно.

$$\frac{(k_1+p_1k_2)(j_2+p_2j_1)}{p_1p_2} = k_2j_1 + \frac{k_1j_1}{p_1} + \frac{k_1j_2+k_2j_2p_1}{p_1p_2} = k_2j_1 + \frac{k_1j_1}{p_1} + \frac{j_2k}{N}.$$

Первое слагаемое просто будем игнорировать. Остаются ещё два. Тем самым у нас на самом деле всё должно радикально упроститься.

$$A_k = \frac{1}{p_2}\sum_{j_2=0}^{p_2-1}\hat{A}(k_1,j_2)\exp\left(-2\pi i\frac{kj_2}{N}\right), \qquad \hat{A}_k(k_1,j_2) = \frac{1}{p_1}\sum_{j_1=0}^{p_1-1} f_{j_2+p_2j_1}\exp\left(-2\pi i\frac{k_1j_1}{p_1}\right).$$

Давайте оценивать алгоритм. $\hat{A}$ это $O(p_1^2p_2)$, A есть $O(p_1p_2^2)$. Таким образом $O(p_1^2p_2+p_1p_2^2)$. Если получилось так, что $p_1, p_2 \sim \sqrt{N}$, то $O(N^{\frac{3}{2}})$. А если вы смогли разложить не на два, а на три множителя. Ситуацию надо доводить до абсурда, то есть когда $N = 2^m$ и алгоритм оценивается, как $O(N\log_2 N)$.

К этому мы вернёмся в мае, когда будем решать уравнения в частных производных.

6 Численное дифференцирование

Тема большая, мы её ужмём на полторы лекции. Например, ограничимся вычисление производных от функций одной переменной.

Пусть есть некоторая функция f, гладкая настолько, насколько нам потребуется. И есть точка x_0. Нам надо посчитать k-ю производную $f^{(k)}(x_0)$. Задача в таком виде не очень хороша, неясно, чем можно пользоваться. Будем пытаться делать как раньше. Считаем, что есть набор точек $x_1,\dots,x_n$, в которых функция известна $f(x_1),\dots,f(x_n)$.

Вариант считать производную у интерполяционного многочлена. Но суммы чудовищные у Лагранжа, у Ньютона ещё хуже. Поэтому будет чуть по-другому оформлено. Интерполяционный многочлен — это первый подход.

Второй подход: приблизим $f^{(k)}(x_0 \approx \sum_{j=1}^{n} c_j f(x_j)$. От формулы потребуем, чтобы на многочленах формула была точна. На коэффициенты при этом получаются линейные соотношения. Результат получается, как бы это ни было смешно, такой же, как и при дифференцировании интерполяционного многочлена.

Давайт подставим в нашу формулу многочлен $f(x) = \sum_{j=1}^{m} a_j x^j$.

$$\sum_{j=0}^{m} a_j (x^j)^{(k)}\Big|_{x_0} = \sum_{j=1}^{n} c_i\Big(\sum_{j=1}^{m} a_j x_i^j\Big)..$$

Надо собрать подобные. Имеем $(x^J)^{(k)} = j(j-1)\dots(j-k+1)x^{j-k}$.

На c_i система с матрицей определителя Вандермонда. Для этого, $m = n-1$. И такие коэффициенты единственны.

Примем без доказательства приятный факт, а именно, что можно каждый раз не мучиться, а говорить, что если точки будем брать определённым регулярным образом, можно пользоваться для коэффициентов пользоваться едиными формулами.

Пусть $h > 0$, $h \ll 1$. Набор $x_0 + ih$, $i = 0, \pm1, \pm2, \dots$ называется шаблоном. Точки расположены равномерно. В этом случае у нас будет получаться, что формула получается такой

$$f^{(k)}(x_0) \approx \frac{\sum\limits_i c_i f(x_i)}{h^k}.$$

Коэффициенты для всех производных те же.

Давайте чего-нибудь посчитаем. Посчитаем $f'(x)$. Одной точки недостаточно, потому что в этом случае интерполяционный многочлен будет константой и производная будет ноль. Берём две точки, x, x_h или две

точки $x-h$, x. Тогда будем брать вместо наклона касательной, наклон секущей

$$f'(x) \approx \frac{f(x+h)-f(x)}{h} = D_+^{(1)}(h)f(x).$$

Для другой пары точек получим другую формуру

$$f'(x) \approx \frac{f(x)-f(x-h)}{h} = D_-^{(1)}(h)f(x).$$

Дальше нам надо увеличивать количество точек. Возьмём три точки $x-h, x, x+h$. Формально, это, конечно, парабола, потому что точек три. Можно выписать метод неопределённых коэффициентов, я выпишу результат: по факту отработают только две точки

$$f'(x) \approx \frac{f(x+h)-f(x-h)}{2h} = D_0^{(1)}(h)f(x).$$

Как мы будем оценивать их эффективность? Во всех этих формулах присутствует малый параметр, если h будем брать меньше, точность будет больше. Нам нужен способ оценки погрешности этих формул.

Начнём с первых формулок.

$$\frac{f(x+h)-f(x)}{h} = \frac{1}{h}\left(f(x) = hf'(x) + \frac{h^2}{2}f''(\xi) - f(x)\right) = f'(x) + \frac{h}{2}f''(\xi), \qquad \xi \in (x, x+h).$$

Такая же примерно формула получится для второй формулы. Погрешность имеет линейный порядок по h.

$$\frac{f(x+h)-f(x-h)}{2h} = \frac{1}{2h}\left(f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{6}f'''(\xi) - f(x) + hf'(x) - \frac{h^2}{2}f''(x) + \frac{h^3}{6}f'''(\xi)\right) =$$
$$= f'(x) + \frac{h^2}{12}\big(f'''(\xi_1) + f'''(\xi_2)\big) = f'(x) + \frac{h^2}{6}f'''(\xi_3).$$

Здесь ξ_3 получается из теоремы о среднем.

Давайте теперь поймём: есть метод построить с точностью h, есть метод, дающий h^2. Давайте попробуем общим метод выстроить. Вот считаем мы

$$f^{(k)}(x) = \frac{\sum\limits_{j=1}^{n} c_j(x_j)}{h^k} + ch^p.$$

Мы функцию на самом деле считаем с погрешностью $f(x) \sim f^*(x)$, $|f - f^*| \sim \varepsilon$. Как будет выглядеть реальная погрешность? Формально она будет зависеть от h.

$$R(h) \leqslant ch^p + \frac{\sum\limits_{j=1}^{n} |c_j|\varepsilon}{h^k} = ch^p + \frac{M\cdot\varepsilon}{h^k} = g(h).$$

Картинка довольно грустная. Слишком маленьким h мы брать не должны. Как определить оптимальную величину h? Определять будем, конечно, только по порядку, но результат здесь довольно поучительный. Надо просто у $g(h)$ производную приравнять к нулю

$$cph^{p-1} - \frac{Mk\varepsilon}{h^{k+1}} = 0 \qquad \Rightarrow \qquad h_0 \sim \varepsilon^{\frac{1}{p+k}}, \quad g(h_0) \sim \varepsilon\varepsilon^{\frac{p}{p+k}}.$$

Нам k дано свыше, это заказ. Если мы p выберем 1 для $k=1$, то $h_0 = \varepsilon^{\frac{1}{2}}$ и $g(h_0) \sim \varepsilon^{\frac{1}{2}}$. Если же для $k=1$, $p=2$, то $h_0 \sim \varepsilon^{1/3}$ и $g(h_0) \sim \varepsilon^{\frac{2}{3}}$. И чем выше порядок точности формулы, тем точнее мы сможем посчитать производную.

Теперь давайте построим формулу для порядков производных повыше. Возьмём $k=2$. Тут минимум три точки, и мы получаем некоторый выигрыш, когда точки расположены симметрично. Давайте возьмём $x-h$, x, x_h. Давайте возьём

$$D_-^{(1)}(h)D_+^{(1)}(h)f(x) = D_-^{(1)}(h)\left(\frac{f(x+h)-f(x)}{h}\right) = \frac{f(x-h)-2f(x)+f(x+h)}{h^2} \equiv D^{(2)}(h)f(x).$$

Эту формулу надо знать в лицо.

$$D^{(2)}(h)f(x) = \frac{1}{h^2}\left(f - hf' + \frac{h^2}{2}f'' - \frac{h^3}{6}f''' + \frac{h^4}{24}f^{(4)}(\xi_1) - 2f + f + hf' + \frac{h^2}{2} + \frac{h^3}{6}f''' + \frac{h^4}{24}f^{(4)}(\xi_2)\right) = \\ = f''(x) + \frac{h^2}{12}f^{(4)}(\xi).$$

Формула получилась лучше, чем мы ожидали (за счёт симметрии): она верна для многочленов до третей степени. Других я писать и не буду, поэтому обозначение D без всяких плюсов-минусов.

Дальше можно двигаться в нескольких направлениях: добавлять количество узлов и уточнять формулы или учиться считать третью производную.

$$f'''(x) \approx D_0^{(1)}(h)D^{(2)}(h)f(x)$$

Здесь получатся узлы $x - 2h, x - h, x + h, x + 2h$. Обоснуйте дома разложением в ряд Тейлора, насколько формула точна.

Теперь к другому вопросу. Вот пусть мы k зафиксировали. Как безболезненно увеличивать число узлов. Мы сейчас впервые встретимся таким подходом, который называется «правило Рунге оценки погрешности». Предположим мы с вами затеяли вычислять k-ю производную

$$f^{(k)}(x) = D^{(p)}(h)f(x) + ch^p + O(h^{p+1}).$$

Здесь мы вдруг добавили не остаточный член в промежуточной точке, а O. В качестве упражнения убеиться, что эта формула уточняется до $O(h^{p+2})$ в наших формулах.

Нам надо решить задачу с точностью ε. Надо подбирать $h^p \sim \varepsilon$. Чем это плохо: c неизвестно, вдруг оно здоровенное и этого мало. Или вдруг c мало и вы выбрали сильно маленький шаг, а это плохо, мы уже выяснили, сильно малый шаг брать нельзя из-за машинной точности.

Вот что предлагает Рунге. Пусть $0 < q < 1$. Тогда

$$f^{(k)}(x) = D^{(p)}(qh)f(x) + c(qh)^p + O(h^{p+1}).$$

Идея такова, что c будет либо такая же, либо отличаться будет так, что разницу можно загнать в $O(h^{p+1})$. Вычтем одно из другого

$$ch^p = \frac{D^{(p)}(qh)f(x) - F^{(p)}(h)f(x)}{1 - q^p} + O(h^{p+1}) = \rho(h) + O(h^{p+1}).$$

Давайте на примере $q = \frac{1}{2}$. Тогда считаем для h и для $h/2$. Получаем $\rho(h)$. Если плохая погрешность, давайте считать $\rho(h/2)$. И так делим, пока погрешность нас не устроит.

7 лекция

В прошлый раз мы с вами приступили к численному дифференцированию. Остановились на правиле Рунге оценки погрешности. Сейчас на примере вычисления первой производной опишу алогитм «метод Ромберга». Вещь несколько устаревшая. Но тем не менее добавляет представления, о том, что происходит с ошибками при вычислении и в чём специфика нашей системы счисления.

Итак на надо для $\varepsilon > 0$ получить $f'(x)$. ε берём на пределе наших вычислительных возможностей. Давайте посмотрим, что можно выжать из тех выкладок, которые мы сделали на прошлой лекции.

Вот что у нас есть в активе

$$f'(x) = \frac{f(x+h) - f(x-h}{2h} + ch^2 + O(h^4).$$

Поскольку точки выбрали симметрично, получили не h^3, а h^4. На самом деле тут

$$f'(x) = \underbrace{\frac{f(x+h) - f(x-h}{2h}}_{D_{(2)}^{(1)}(h)f(x)} + ch^2 + c_1h^4 + c^2h^6 + \dots.$$

Будем потихоньку избавляться от этих членов.

$$\begin{aligned} f'(x) &= D^{(1)}_{(2)}(h)f(x) + ch^2 + O(h^4); \\ f'(x) &= D^{(1)}_{(2)}(h/2)f(x) + c\left(\frac{h}{2}\right)^2 + O(h^4); \\ ch^2 &= \frac{D^{(1)}_{(2)}\left(\frac{h}{2}\right)f(x) - D^{(1)}_{(2)}(h)f(x)}{1-\frac{1}{4}} + O(h^4). \end{aligned}$$

Это ещё правило Рунге. Что мы можем получить из этого

$$f'(x) = D^{(1)}_{(2)}(h)f(x) + \frac{D^{(1)}_{(2)}\left(\frac{h}{2}\right)f(x) - D^{(1)}_{(2)}(h)f(x)}{1-\frac{h}{4}} + O(h^4) = D^{(1)}_{(4)}\left(\frac{h}{2}\right)f(x) + c_1h^4 + O(h^6).$$

Дальше можно снова применять правило Рунге и обирать следующие члены.

Из шага h мы получаем формулу $D^{(1)}_{(2)}(h)$. Для двух шагов h и $h/2$ мы можем построить формулу $D^{(1)}_{(4)}$ четвёртого порядка, если нас не устроила погрешность. Если и она нас не устроила, мы можем построить формулу $D^{(1)}_{(6)}$ шестого порядка из трёх шагов h, $h/2$, $h/4$.

Если бы мы пользовались только одним правилом Рунге, то есть выбирали только $D^{(1)}_{(2)}\left(\frac{h}{2^n}\right)$, мы бы всё время получали формулу второго порядка. А мы предъявили алгоритм получения последовательности формул с возрастающим порядком точности.

Всё это будет давать формулы вида $\frac{1}{h}\Sigma c_i f_i$.

8 Численное интегрирование

Задача более широкая. Поставим задачу, что мы хотим вычислить. Хотим получить в результате число. Таким образом, интегралы у нас будут определённые. Конечно, они будут собственные или несобственные. В одномерном случае у нас задачи будут сводиться к постановке

$$\int_a^b f(x)\,dx.$$

Если несобственный, мы будем отрезкать хвост от промежутка интегрирования с какой-то точностью.

$$\int_a^b f(x)\,dx \approx \sum_{i=1}^{n} c_i f(x_i).$$

Числа c_i принято называть коэффициентами, x_i — узлами.

Принято на ряду с этим рассматривать вот такое обобщение

$$\int_a^b f(x)p(x)\,dx \approx \sum_{i=1}^{n} c_i f_i.$$

При таком подходе $p(x)$ называется весовой функцией, c_i будут зависеть от $f(x)$.

Как мы увидим, погрешность, как бы мы её ни определяли, будет зависеть от гладкости функции. Если гладкости нет, то оценки погрешности у нас будет неверные. В функцию $p(x)$ мы все нехорошести занесём, и при соблюдении особых правил, погрешность будет зависеть только от гладкости $f(x)$.

Пока о никаких весовых функциях говорить не будем.

Будем обозначать $I(f) = \int_a^b f(x)\,dx$, когда будет понятно, какие пределы интегрирования. Также будем обозначать $S(f) = \sum_{i=1}^{n} c_i f(x_i)$.

Погрешность будем обозначать, $R(f)$, то есть

$$I(f) = S(f) + R(f).$$

Если нам потребуется вычислять двумерный интеграл или более. Смысл будет тот же, только формулы не будут уже называться квадратурными.

Как мы будем подбирать узлы и коэффициенты наших формул?

8.1 Квадратурные формулы Ньютона—Котеса

Узлы выбираются равномерно по отрезку. Если $n = 1$, то $x_1 = \frac{a+b}{2}$. Если $n \geqslant 2$, то

$$x_1 = a, \quad x_n = b, \quad x_{i+1} - x_i = h = \text{const}\,.$$

Подход для вычисления коэффициентов очень приятный. После того, как мы зафиксировали $x_1, \ldots, x_n$. Мы можем вспомнить, что у нас было при интерполировании функции многочленом Лагранжа.

$$f(x) = L_n(x) + \frac{\omega_n(x) f^{(n)}(\xi)}{n!}, \quad \xi \in [a, b], \quad \omega_n(x) = \prod_{i=1}^{n} (x - x_i).$$

Что такое L_n мы тоже знаем

$$L_n(x) = \sum_{i=1}^{n} f(x_i) \prod_{j \neq i} \frac{x - x_j}{x_i - x_j} = \sum_{i=1}^{n} f(x_i) \Phi_i(x).$$

Подставим в основную формулу

$$I(f) = (b - a) \sum_{i=1}^{n} c_i f(x_i) + R(f),$$

где

$$c_i = \frac{1}{b-a} \int_a^b \Phi_i(x)\, dx, \qquad R(f) = \frac{1}{n!} \int_a^b \omega_n(x) f(n)\big(\xi(x)\big)\, dx.$$

Оказывается, что при таком подходе c_i не будут зависеть от a и b, только от n. Это хорошая часть. Что есть плохое: оценка для погрешности. Она, скажем прямо, непрактичная. Стоит n-я проивздодная не пойми в какой точке, ещё интеграл брать.

Давайте сейчас, ну во-первых, я сформулирую некоторое утверждение в виде задачи.

Утверждение 8.1. *Понятно, что точки расположены равномерно по отрезку не просто так. Оказывается, что коэффициенты c_i в нашей формуле обладают определённой симметрией. Если их количество нечётно, один узел попадают в середину. Все расположены симметрично относительно середеины. Утверждение в том, что $c_i = c_{n+1-i}$.*

Доказательство. Намечу план доказатепльства. Надо здесь всё обезразмерить. Можно сделать замену $x = a + th$. И при вычислении интегралов сделать эту замену и в лоб выписать, чему равняются c_i. Не очень приятная задача, но очень полезно это сделать. ■

Мы приняли тот факт, что $c_i = c_{n+1-i}$. И ещё одно свойство, самое приятное: если функцию взять константу, погрешность интерполяции будет ноль. А значит, и погрешность нашей формулы ноль. Следовательно, $c_1 + \ldots c_n = 1$.

Нам надо договориться, о каком-то критерии эффективности. Будем подставлять многочлены и требовать, чтобы формула была точна. Наибольшая степень многочлена, для которого формула точна, будет порядком точности. Есть ещё малый параметр: расстояние между узлами, через который можно оценивать погрешность. Но как ни смешно это, удонее оценивать через длину отрезка $[a, b]$.

Начнём с многочленов. Какая максимальная степень многочленов, для которых формула точна. Оказывается, если n нечётно, то формула точна для многочленов степени n.

Имеется у нас равномерная сетка и интеграл мы замеряем, как договорились

$$\int_a^b f(x)\, dx \approx (b - a) \sum_{i=1}^{n} c_i f(x_i), \qquad n = 2k + 1.$$

Рассмотрим произвольный многочлен степени n, обозначим

$$Q_n(x) = a_n x^n + \ldots.$$

Давайте через $\overline{x}$ обозначим $\overline{x} = \frac{a+b}{2}$. Тогда

$$Q_n(x) = a_n (x - \overline{x})^n + P_{n-1}(x).$$

Если мы затеем вычислять наш интеграл от первого слагаемого, так как оно нечётно,

$$I\big(a_n(x-\overline{x})^2\big)=0.$$

При этом $S\big(a_n(x\overline{x})^n\big)=0$. Значит,

$$I(Q_n)=I\big(P_{n-1}(x)\big)=S\big(P_{n-1}(x)\big)=S(Q_n).$$

Теперь по поводу малого параметра. Вот подумаем. Что мы можем сказать о погрешности? Есть у нас n-я производная и многочлен. Во всех формулах у нас получалось: чем больше отрезок, на котором происходит интерполяция, тем оценка погрешности хуже. Мы не можем сказать, что интегрировать будем только по маленьким отрезкам. Так как интеграл есть аддитивный функционал. Мы его разобьём на сумму интегралов по элементарным отрезкам.

$$I(f)=\int\limits_a^b f(x)\,dx=\sum_{j=1}^{N-1}I_j(f)=\sum_{j=1}^{N-1}\int\limits_{x_j}^{x_{j+1}}f(x)\,dx.$$

При этом обозначим $x_{j+1}-x_j=h$. Тогда если на каждом элементарном отрезке погрешность порядка $O(h^p)$, то на всём $[a,b]$ будет ошибка $\sum\limits_{j=1}^{N-1}O(h^p)=O(h^{p-1})$.

Начнём с простых ситуаций. Пусть у нас $n=1$. Мы обозначаем $h=b-a$.

$$I(f)=\int\limits_a^b\approx=hc_1f(\overline{x}).$$

Коэффициент c_1 подбирается так, чтобы на многочленах до первой степени была точная формула.

Пусть $f\equiv1$. Тогда $c_1=1$. Значит,

$$I(f)=hf(\overline{x})=\Pi(f).$$

Это называется формула прямоугольника. Ясно, что в линейном случае справа и слева в разные стороны будет торчать треугольник.

Пусть теперь $n=2$. Тогда

$$I(f)=h\big(c_1f(a)-c_2f(b)\big).$$

Спокойно сюда подставляем $f=1$ и $f=x$. После определённых усилий получим, что $c_1=c_2=\frac12$. Формулу

$$I(f)=h\left(\frac{f(a)+f(b)}{2}\right)=T(f)$$

традиционно принято называть формулой трапеции.

Теперь нам хотелось бы получить разложение ошибки по h.

Вопрос маленький. Чему равен

$$\int\limits_a^b(x-\overline{x})^j=\begin{cases}h, & j=0;\\ 0, & j=1;\\ \frac{h^3}{12}, & j=2;\\ 0, & j=3;\\ \frac{h^5}{80}, & j=4.\end{cases}$$

Этого мне будет достаточно.

Разложим нашу функцию в ряд относительно середины отрезка

$$f(x)=f(\overline{x})+(x-\overline{x})f'(\overline{x})+\frac{(x-\overline{x})^2}{2}f''(\overline{x})+\dots$$

И подставим это разложение в наш интеграл

$$\int\limits_a^b f(x)\,dx=\underbrace{hf(\overline{x})}_{\Pi(f)}+\underbrace{\frac{h^3}{24}f''(\overline{x})+O(h^5)}_{\mathbb{R}_\Pi}.$$

Таким образом, $\big|I(f)-\Pi(f)\big|=O(h^3)=\frac{h^3}{24}f''(\overline{x})+O(h^5)$. На самом деле мы фактически показали, как выписать

весь ряд погрешностей.

С формулой трапеции немножко больше потрудиться надо. Подставим в то же разложение f относительно $\overline{x}$ значения в концах отрезка.

$$\begin{aligned} f(a) &= f(\overline{x}) - \frac{h}{2}f'(\overline{x}) + \frac{h^2}{8}f''(\overline{x}) - \frac{h^3}{48}f'''(\overline{x}) + \frac{h^4}{384}f^{(4)}(\overline{x}) + \dots \\ f(b) &= f(\overline{x}) + \frac{h}{2}f'(\overline{x}) + \frac{h^2}{8}f''(\overline{x}) + \frac{h^3}{48}f'''(\overline{x}) + \frac{h^4}{384}f^{(4)}(\overline{x}) + \dots \end{aligned}$$

Давайте их сложим и умножим на h.

$$hf(\overline{x}) = h\frac{f(a)+f(b)}{2} - \frac{h^3}{8}f''(\overline{x}) - \frac{h^5}{384}f^{(4)}(\overline{x}) + \dots \tag{2}$$

Подставим это в формулу для треугольника, которая выглядела так

$$I(f) = hf(\overline{x}) + \frac{h^3}{24}f''(\overline{x}) + \dots = \frac{h}{2}\big(f(a) = f(b)\big) - \frac{h^3}{12}f''(x) - \frac{h^5}{480}f^{(4)}(\overline{x}) + \dots.$$

По модулю получилось в два раза хуже. Но главное, конечно, порядок.

Давайте попробуем на основе этих двух формул построить формулу третьего порядка.

$$\begin{aligned} I(f) - \Pi(f) &= \frac{h^3}{24}f''(\overline{x}) + \frac{h^5}{1\,920}f^{(4)}(\overline{x}) + \dots; \\ I(f) - T(f) &= -\frac{h^3}{12}f''(\overline{x}) - \frac{h^5}{480}f^{(4)}(\overline{x}) + \dots. \end{aligned}$$

Первое умножим на два, сложим со вторым, и всё поделим на три

$$I(f) = \frac{2}{3}\Pi(f) + \frac{1}{3}T(f) - \frac{h^5}{2\,880}f^{(4)}(\overline{x}) + \dots = C(f) + O(h^5).$$

Это называется формулой Симпсона. Узлы здесь задействованы такие: $a, b, \overline{x}$. Это формула из нашего семейства. Точна для многочленов до третьей степени включительно. И мы получили даже её общий вид

$$C(f) = \frac{h}{6}\big(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b).$$

В качестве хорошей задачи можно взять и посчитать интерполяционный многочлен по трём точкам и убедиться, что получается такая же формула.

9 Полулекция перед Коробейниковым

В прошлый раз вывели три простейших формулы численного интегрирования. Когда равномерная сетка по отрезку и берётся интеграл от интерполяционного многочлена.

$$\int_a^b f(x)\,dx \approx \sum_{j=1}^{n} c_j f(x_i).$$

Если $n = 1$, то $x_1 = \frac{a+b}{2} = \overline{x}$.

Если $n \geqslant 2$, $x_1 = a$, $x_n = b$, $x_j - x_{j-1} = \text{const}$.

Обозначали $b - a = h$.

$$R(f) = I(f) - S(f) =\equiv \int_a^b f(x)\,dx - \sum_{j=1}^{n} c_j f(x_j).$$

Для $n = 1$ имеем $R(f) = \frac{h^3}{24}f''(\overline{x}) + O(h^5)$.

Для $n = 2$ формулу назвали формулой трапеций и

$$R(f) = -\frac{h^3}{12}f''(\overline{x}) + O(h^5).$$

Для $h = 3$ получили формулу Симпсона и

$$R(f) = \frac{-h^5}{2\,880} f^{(4)}(\overline{x}) + O(h^7).$$

9.1 Простейшие составные квадратурные формулы

Давайте поступим следующим образом. Наш отрезок $[a, b]$ разобьём на некоторое количество отрезков уже малой длины. Обозначим $x_1 = a$, $x_n = b$. Разбиваем так, чтобы $x_{i+1} - x_i = h$. Теперь весь интеграл разбиваем

$$I_(f) = \sum_{j=1}^{n-1} I_j(f) = \sum_{j=1}^{n-1} \int\limits_{x_j}^{x_{j+1}} f(x)\,dx.$$

Обозначим $\overline{x}_i = \frac{x_i + x_{i+1}}{2}$. И перейдём к составлению квадратурных формул.

Для формулы средних прямоугольников

$$I(f) = \sum_{j=1}^{n-1} hf(\overline{x}_j) + R_j^{\Pi}(f) = \sum_{j=1}^{n-1} hf(\overline{x}_j) + \frac{h^3}{24} \sum_{j=1}^{n-1} \big(f''(\overline{x}_j) + O(h^2)\big).$$

Пусть $\big|f''(x)\big| \leqslant M_2$. Можем просуммировать

$$\big|R^{\Pi}(f)\big| \approx \frac{h^2}{24} M.$$

Можно по-другому оценить. Сказать, что

$$R^{\Pi}(f) \approx \frac{h^2}{24} \sum_{j=1}^{n-1} hf''(\overline{x}_j) \approx$$

Это похоже на квадратурную формулу средних прямоугольников

$$\approx \frac{h^2}{24} \int\limits_a^b f''(x)\,dx.$$

Дальше можем посмотреть на формулу трапеции.

$$I(f) = \sum_{j=1}^{n-1} \frac{h}{2}\big(f(x_i)+f(x_{i+1})\big) - \frac{h^3}{12} \sum_{j=1}^{n-1} f''(\overline{x}) + O(h^5) = h\left(\frac{1}{2}f(x_1) + f(x_2) + \cdots + f(x_{n-1}) + \frac{1}{2}f(x_n)\right) - \frac{h^2}{12} \int\limits_a^b f''(x)\,dx + O(h$$

В качестве упражнения остаётся написать составную квадратурную формулу для формулы Симпсона.

Для вычисления погрешности можно придумать аналог правила Рунге. Можно сделать шаг вдвое меньше, формулы вычесть и так далее.

Надо мысль как-то развивать. Если функция не дифференцируемая, такая техника у нас не пройдёт. А если больше узлов, как всё устраивать? Надо как-то обобщать. Пока приёмы индивидуальны. Попробуем получить более грубую оценку погрешности, но чтобы оценка работала для любого типа формул. Ну и давайте избавимся от необходимости ограниченной второй производной. Что делать, если этого нет. Как быть, если функция не достаточно гладкая. Если проблема носит искусственный характер (конечное число плохих точек), ну разбили на кусочки. Если $\sqrt{x}$ уже хуже.

Положим, что подынтегральную функцию можно разбить на два множителя

$$\int\limits_a^b f(x)p(x)\,dx$$

Здесь $f(x)$ гладкая, а все особенности лежат в $p(x)$. Квадратурную формулу подбираем следующим образом

$$\int\limits_a^b f(x)p(x)\,dx \approx \sum_{j=1}^{n} c_j f(x_j).$$

Можно снова сделать то же самое, что мы только что делали. Во всех наших формулах добавятся значения $p(x)$ в узлах. Коэффициенты при симметричных узлах снова будут равны, если весовая функция симметрична. Иначе не будет. Как здесь оуеним погрешность. Наши способы не пройдут. Получим более грубую оценку. Хотим чтобы погрешность выражалась в терминах $b-a$, то есть длины отрезка. Идея будет следующая. Попытаемся получить единый результат на все формулы. Такая нужна какая-то характеристика формулы. Максимальная степень многочлена, для которого формула точна. Будем считать, что формула $S(f)$ верна для многочленов степени до m включительно.

10 Лекция какая-то

В прошлый раз мы закончили правилом Рунге. Давайте теперь попытаемся изыскать внутренние резервы наших методов численного интегрирования.

Представили функцию $\int\limits_a^b f(x)p(x)\,dx \approx \sum\limits_{j=1}^n c_j f(x_j)$.

Неизвестных у нас здесь $2n$ штук. Подставлять мы будем многочлены $1, x, x^2, \ldots, x^m$, их $m+1$ штука. Неизвестные c_j и x_j, их $2n$ штук. Количество решений бывает всякое. Мы выдвигаем гипотезу, что у нас получится $m = 2n-1$, то есть решение будет единственно. Ну только вот кто сказал, что когда мы решим, получим, что x_j попадут в отрезок $[a,b]$.

Наложим ограничения на весовую функцию $p(x)$. Пусть она больше нуля почти всюду на нашем отрезке. Ну и желательно, чтобы сама она была интегрируема.

Давайтпе обсудим, как квадратурная формула будет строиться. Я буду кое-что давать без доказательства. Будете шуметь, с доказательством дам.

Предположим мы формулу многочлена степени $2n+1$ построили. Если бы она существовала, каким свойством должны обладать её узлы?

$$\forall\ Q_{2n-1}(x) \quad \int\limits_a^b Q_{2n-1}(x)p(x)\,dx = \sum_{j=1}^n c_j Q_{2n-1}(x_j).$$

Пусть эта формула верна для многочленов до степени $2n-1$ включительно и мы знаем узлы этой формулы x_j. Тогда

$$\int\limits_a^b \omega_n(x)P_{n-1}(x)p(x)\,dx = 0,$$

где $\omega_n(x) = \prod\limits_{j=1}^n (x - x_j)$.

Давайте смотреть $\omega_n(x) = P_{n-1}(x) = Q_{2n-1}(x)$, $Q_{2n-1}(x_j) = 0$.

Что это означает. Что ω_n ортогональна всем многочленам степени $n-1$.

Если есть вес $p(x) > 0$, то существует (по Грамму—Шмидту) ортогональная система многочленов, перпендикулярную $p(x)$. Главное, что любой из этих многочленов имеет ровно n корней и все они пребывают на $[a,b]$. Поэтому с точностью до множителя, это $\omega_n(x)$.

Поэтому в качестве узлов, мы обязаны брать нули соответствующего ортогонального многочлена.

10.1 Пример

Пусть $p = 1$ — вес единичка. Тогда соответствующий ортогональный многочлен на $[-1,1]$, многочлен Лежандра

$$\psi_n(x) = \frac{1}{2^n n!}\frac{d^n}{dx^n}\big((x^2-1)^n\big).$$

Пусть теперь

$$p = \frac{2}{\pi\sqrt{1-x^2}}.$$

Тогда берём $\psi_n(x) = T_n(x)$ — многочлен Чебышёва.

10.2 Формулы для коэффициентов

Ну хорошо, а как быть с коэффициентами.

Итак сначала мы находим x_j из уравнения

$$\int_a^b \omega_n(x) P_{n-1}(x) p(x)\,dx.$$

Далее подбираем c_j таким образом, чтобы формулы была точной для многочленов степени до $n-1$ включительно. То есть

$$\int_a^b f(x)p(x)\,dx \approx \int_a^b L_n(x)p(x)\,dx = \sum c_j f(x_j)$$

должно быть нашей итоговой формулой. Сейчас мы попробуем это доказать. Здесь L_n — интерполяционный многочлен Лагранжа.

Возьмём произвольный $Q_{2n-1}(x)$. Поделим его с остатком на $\omega_n(x)$.

$$Q_{2n-1}(x) = \omega_n(x) g_{n-1}(x) + r_{n-1}(x).$$

Что есть погрешность. На остатке она ноль, а не первом слагаемом уже обсуждали

$$R\big(Q_{2n-1}(x)\big) = \underbrace{R\big(\omega_n(x)g_{n-1}(x)\big)}_{=0} + \underbrace{R\big(r_{n-1}(x)\big)}_{=0}.$$

Мы постоили квадратурные формулы наибольшей алгебраической точности. Они называются квадратурными формулами Гаусса.

Какие свойства.

Утверждение 10.1. *Если $p > 0$ и p — симметрична относительно $\frac{a+b}{2}$, то*

1. *x_j симметричны;*
2. *$c_{n+1-j} = c_j$.*

Эти два свойства доказываются противно. Мы их доказывать не будем. Докажем третье.

Утверждение 10.2. *Пусть $p > 0$. Тогда $c_j > 0$.*

Доказательство. Это свойство доказывается просто примером. Зафиксируем какое-то $k \in \{1..n\}$. Построим

$$f_k(x) = \left(\frac{\omega_n(x)}{x - x_k}\right)^2$$

Это многочлен степени $2n-1$, значит, наша формулы для него точна

$$0 < \int_a^b f_k(x)p(x)\,dx = \sum_{j=1}^n c_j f_k(x_j) = c_k \underbrace{f_k(x_k)}_{>0}.$$

■

В качестве упражнения можете выписать оценки погрешности. Грубо говоря, в том, что мы делали, заменить n на $2n-1$. Есть такая тактика провоцировать экзаменатора на сложные вопросы. Писать в билете всё, ктоме убойного вопроса, который вы знаете. Слишком скрытничать нельзя. Но какой-то изюм лючше выкладывать не сразу. Вас спрашивают этот сложный вопрос, думая, что вы не знаете, а вы сходу пишете. Обычно на этом все довольны, всё заканчивается. К сожалению, понимание такой тактики приходит не сразу.

10.3 Двойные интегралы

Мы ограничимся случаем прямоугольника за нехваткой времени. Давайте-ка возьмём плоскость (x, y). На нём у на будет прямоугольничек $[a, b] \times [\alpha, \beta]$. Будем считать такой интегральчик

$$I(f) = \int_\alpha^\beta \int_a^b f(x, y)\,dx\,dy.$$

Будем разбивать прямоугольник на много маленьких. Будем считать, что у нас уже стороны прямоугольника есть малые. Давайте через S я обозначу $S = (b-a)(\beta-\alpha)$, $\overline{x} = \frac{a+b}{2}$, $\overline{y} = \frac{\alpha+\beta}{2}$. Простейший способ считать интеграл, первая простейшая формулу.

$$I(f) \approx S f(\overline{x}, \overline{y}).$$

Понятно, что если мы будем иметь какую-то большую область-прямоугольник, мы можем разбить на много маленьких равных прямоугольников. По вертикали M штук, по горизонтали N штук. Тогда

$$I = \sum_{i,j} Sf(\overline{x}_i, \overline{y}_j).$$

Локальная формула должны быть точной для линейных функций.

$$f(x,y) = f(\overline{x},\overline{y}) + (x-\overline{x})f'_x + (y-\overline{y})f'_y + \frac{1}{2}(x-\overline{x})^2 f''_{xx} + \frac{1}{2}(y-y\overline{y})^2 f''_{yy} + (x-\overline{x})(y-\overline{y})f''_{xy} + \dots$$

Ну вот давайте подставлять

$$I(f) = Sf(\overline{x},\overline{y}) + \frac{1}{24}S\left((b-a)^2 f''_{xx} + (\beta-\alpha)^2 f''_{yy}\right) + \dots.$$

Теперь берём общий случай, когда $b-a$ и $\beta-\alpha$ не являются малыми параметрами. Будем делить на маленькие прямоугольники. N по горизонтали, M по вертикали. Для каждого элементарного прямоугольника погрешность с точностью до малых более высокого порядка будет записываться следущим образом

$$R_i(f) \approx \frac{1}{24}S_i\left(\left(\frac{b-a}{N}\right)^2 f''_{xx} + \left(\frac{\beta-\alpha}{M}\right)^2 f''_{yy}\right).$$

Отсюда $R(f) = \sum\limits_i R_i = O\left(N^{-2} + M^{-2}\right)$. Хорошо это или плохо? Мы можем работать как и в одномерном случае. Можем придумывать правило Рунге, дробить больше сетки. Также можем уточнять локальные формулы.

Давайте сделаем себе жизнь потяжелее. Займёмся последовательным интегрированием.

$$I(f) = \int\limits_\alpha^\beta \int\limits_a^b f(x,y)\,dx\,dy = \int\limits_\alpha^\beta F(y)\,dy, \quad F(y) = \int\limits_a^b f(x,y)\,dx.$$

Пусть $I \approx \sum\limits_j \overline{c}_j F(y_j)$. Теперь для вычисления каждого и $F(y_j)$ строим формулы

$$F(y_j) \approx \sum_i c_i f(x_i, y_j).$$

Теперь собираем это вместе

$$I \approx \sum_{i,j} c_i \overline{c}_j f(x_i, y_j).$$

Ну допустим мы с вами взяли большой прямоугольник. Разбили его с шагами h_x и h_y. И по каждой стороне взяли составную формулу трапеций. Что в результате получится. Тогда

$$\frac{c_i\overline{c}_j}{h_x h_y} = \begin{cases} 1 & \text{внутренняя точка} \\ \frac{1}{2} & \text{внутренная только по одной переменной} \\ \frac{1}{n} & \text{угловая} \end{cases}.$$

Отсюда получим, что $R(f) = O\left(h_x^2 + h_y^2\right)$. Ну сюда можно любые наши формулы одномерные перетащить.

Поверьте мне, что вычисление интегралов в десятимерном пространстве — это актуально. Когда стали их считать, столкнулись с совершенно неожиданными проблемами. Предположим мы считаем по составной формуле трапеций или даже по составной формуле средних прямоугольников.

$$I(f) = \underbrace{\int\limits_0^1 \dots \int\limits_0^1}_{10} f dx_1 \dots dx_{10} \approx \sum_{j=1}^n c_j f(x_j).$$

Пусть у нас есть $\varepsilon > 0$. Тогда $R(f) = O(N^{-2})$. Осюда $N^{-2} \approx \varepsilon$ и $N \sim \frac{1}{\sqrt{\varepsilon}}$.

Пусть $\varepsilon = 10^{-4}$. Тогда $N = 10^2$. Но это количество делений по одной грани. Ну тогда $n = N^{10}$. Это мы подставляем дрожащей рукой.

Вот для двумерного всё было хорошо, для десятимерного всё плохо. А где-то посередие есть граница.

Решение этой проблемы называется методом Монте-Карло. На экзамене нужно будет описать проблему и выписать те соотношения, которые я вам сейчас дам. Я дам вам в общих чертах.

Будем считать, что живём в нашем кубе или области, не важно. Пусть $\Omega = [0,1]^k$. Сгенерируем N штук попарно независимых точек $P_j \in \Omega$. (То есть попарно независимыми генераторами.) В качестве квадратурной формулы возьмём вот такую штуку

$$S_N(f) = \frac{1}{N}\sum_{j=1}^{N} f(P_j).$$

А дальше имеет место вот такое соотношение

Теорема 10.1. *С вероятностью* $(1-\theta)$ *выполнено такое неравенство*

$$\left|I(f) - S_n(f)\right| \leqslant \sqrt{\frac{D(f)}{\theta N}}, \quad D(f) = I(f^2) - I^2(f), \quad \theta \in (0,1).$$

На следующей лекции займёмся численными методами алгебры.

11 Численные методы алгебры

Приступаем к последней большой теме этого семестра. Надо сказать, что не всё, что обычно входит в курс, будет рассказано. Что нас будет интересовать.

1. $Ax = b$, матрица A размерности $n \times n$, $\exists\ A^{-1}$. нас интересует вектор x. Все попутные вещи, обратная матрица или, не дай боже, определитель, здесь не вычисляются.

2. $Ax = b$, $A(m \times n)$, $m > n$. Это хорошая ситуация. Здесь запросто решения может не существовать. Здесь надо на самом деле договориться, что мы будем называть как-бы решением, а затем придумать, как мы будем это решение находить.

Есть ещё класс вопросов связанных с нахождением собственных значений. Не дай бог все. Обычно наибольшее и наименьшее по модулю. Методы будут непрямые.

Есть ещё часть, называемая итерационными методами. Какое-то время назад все проблемы вычислительной математики упирались в то, что времени нехватало. Сейчас со временем всё хорошо относительно, а с памятью уже появились проблемы. Предположим матрица обладает свойством, что элементы расставлены по какому-то закону и закон известен (например, на каждой диагонали одинаковые элементы), можно хранить меньше данных, но не все действия с этой матрицей можно будет выполнять. Всё что будет можно делать, это умножать её на вектор.

Мы будем обращать внимание на то, что мы находимся не в реальной арифметике, а в машинной.

11.1 Метод Гаусса

Он не просто так. Относится к методам на основе LU-разложения, то есть $A = LU$. Здесь L — нижнетреугольная матрица, U — верхнетреугольная матрица[1].

Вообще, что можно сказать про элементы главной диагонали L и U. Там нет нулей, иначе A вырождена, а это нехорошо.

В матрицах L и U неизвестных нам элементов $n^2 + n$. Число уравнений на них n^2. То есть n явно лишних. И мы можем в это разложение можем впихнуть $A = LDD^{-1}U$, диагональную матрицу D. Таким образом можем управлять диагональными элементами L или U. Если мы добиваемся $u_{ii} = 1$, это метод Гаусса. Если $l_{ii} = 1$, это метод Краута.

Это первое, что надо было сказать.

Дальше. Почему это разложение LU существует.

Введём какой-нибудь формализм. Что из себя представляет метод Гаусса. Мы берём матрицу A и её во что-то торжественно превращаем. Обозначим через $A^{(0)} = (A \mid b)$ расширенную матрицу системы.

$$A^{(0)} \to A^{(1)} \to A^{(2)} \to \dots$$

[1] Вообще все методы будут основаны на разложениях $A = LU$ и $A = QR$ — ортогональная $Q^TQ = E$ и право-треугольная.

Переход от матрицы $A^{(i-1)}$ к матрице $A^{(i)}$ состоит вот в чём. Сначала домножаем на c_i ($c_i A^{(i-1)}$)

$$c_i = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & & (a_{ii}^{(i-1)})^{-1} & & & \\ & & & & 1 & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{pmatrix}$$

Дальше умножаем ещё на c_i', где

$$c_i = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & & a_{ii}^{(i-1)} & & & \\ & & & \vdots & 1 & & \\ & & & \vdots & & \ddots & \\ & & & a_{i,n}^{i-1} & & & 1 \end{pmatrix}$$

Итак у нас $LUx = Lb$. Или же $Lx = Lb$. Таким образом сначала решаем $Lx = Lb$, затем $Ux = b$, это всё $O(n^2)$. Посчитать $LUx = Lb$ есть $O(n^3)$.

Что делать, если в первой строке на первом месте ноль. В первом столбце выбираем самое большое число и ставим на первое место. Далее на k столбце ищем самое большое число среди оставшихся строк ($j \geqslant k$) и переставляем с наибольшим элементом.

Пусть P_{ij} — элементарная матрица. При умножении слева её, меняются строки местами. А если её умножить справа, то столбцы меняются местами. Если нам хочется поменять местами строки, появляется

$$P_{ij}Ax = P_{ij}b$$

Матрица меняется, правая часть меняется, но иксы остаются на месте.

Если менять местами столбцы, то дело хуже обстоит

$$AP_{ij}P_{ij}^{-1}x = b.$$

Поменяли столбцы в A и поменялись иксы. Про это часто забывают, нужно представлять, от чего это происходит.

Исторически этот алгоритм на первом месте. Но считается не таким устойчивым, как основанные на разложении QR.

11.2 Метод Холевского

Пусть у нас матрица симметричная и положительно определённая. Тогда её можно трансформировать методом Холевского.

Матрица называется симметричной, если $A^T = A$. Положительно определённой, если $\forall x \neq 0 \quad (Ax, x) > 0$. Записывается это вот так $A = A^T > 0$. Она автоматически является невырожденной. Действительно пусть $Ax = 0$ и $x \neq 0$, то ведь $(Ax, x) = 0$ при ненулевом иксе, что невозможно.

Утверждение 11.1. *Если у нас $A = A^T > 0$, то $a_{ii} > 0$.*

Это легко понять, потому что $a_{ii} = (Ae_i, e_i) > 0$.

Это хорошо, потому что здесь можно не переставлять строки, деления на ноль не произойдёт на автомате.

Отсюда возникает идея метода Холевского. Здесь LU-разложение из себя что представляет: $A = LL^T$. Здесь число уравнений совпадает с числом неизвестных. Кладём $l_{11} = \sqrt{a_{11}}$, первый столбец сооружаем $a_{i1} = \frac{a_{i1}}{l_{11}}$, $i > 2$. Далее

$$l_{ii} = \left(a_{ii} - \sum_{j=1}^{i-1} l_{ij}^2 \right)^{\frac{1}{2}}, \quad i = 2, \ldots, n.$$

Далее

$$l_{ij} = \frac{1}{l_{jj}} \left(a_{ij} - \sum_{k=1}^{i-1} l_{jk} l_{ik} \right), \quad j = 2, \ldots, n, i = j+1, \ldots, n.$$

Ну и наконец $l_{ij} = 0$ для $i < j$.

Теперь перейдём к альтернативному подходу. Не буду оговаривать, что определение касается только вещественного случая.

Определение 11.1. *Матрица Q называется ортогональной, если $QQ^T = E$.*

Можно ряд свойств определить

1. $Q^{-1} = Q^T$;
2. $\det Q = \pm 1$;
3. Q^T ортогональна;
4. Произведение ортогональных ортогонально.

Пусть решаем $Ax = b$. Пусть мы можем представить $A = QR$. То есть наша система равносильна такой $QRx = b$. Дальше $Rx = Q^T b$, её быстро решаем. На самом деле всё будет не совсем так происходить, сейчас разберёмся.

11.2.1 Метод вращений

Первый подход QR-разложения. Сначала определим матрицу элементарного вращения $T_{ij}(\varphi)$, определяется номерами строк, на которые она воздействует, и угол φ.

$$c_i = \begin{pmatrix} 1 & & & & & & & & \\ & \ddots & & & & & & & \\ & & 1 & & & & & & \\ & & & \cos\varphi & -\sin\varphi & & & & \\ & & & & 1 & & & & \\ & & & & & \ddots & & & \\ & & & & & & \sin\varphi & \cos\varphi \end{pmatrix}$$

Допишу потом, понятно, что тут. Тут $t_{ii} = t_{jj} = \cos\varphi$, $t_{ij} = -tji = \sin\varphi$, $t_{kk} = 1$.

Начинаем вращать $A^{(1)} = T_{ij}(\varphi)A$. Получаем

$$a_{ik}^{(1)} = a_{ik}\cos\varphi - a_{jk}\sin\varphi; \qquad a_{jk}^{(1)} = a_{ik}\sin\varphi + a_{jk}\cos\varphi.$$

Хотим, чтобы какой-то $a_{jk}^{(1)} = 0$, для этого

(1) $a_{ik}^2 + a_{jk}^2 \neq 0$. Тогда $\cos\varphi = \frac{a_{ik}}{\sqrt{a_{ik}^2+a_{jk}^2}}$, $\sin\varphi = \frac{-a_{jk}}{\sqrt{a_{ik}^2+a_{jk}^2}}$;

(2) Если же $a_{ik}^2 + a_{jk}^2 = 0$, $\cos\varphi = 1, \sin\varphi = 0$, то есть ничего не делаем.

Вот у нас есть матрица A. Мы хотим, чтобы в первом столбце первый элемент был каким-то числом, остальные нули. Делаю $T_{12}(\varphi_2)A$ так, чтобы $a_{21}^{(1)} = 0$. И дальше также. На первом месте явно не ноль, потому что тогда в этом равенстве будет только одна вырожденная матрица, чего не может быть.

$$\underbrace{T_{1n}(\varphi_n)\dots T_{13}(\varphi_3)T_{12}(\varphi_2)}_{T_1} A = \begin{pmatrix} a_{11}^{(1)} & \dots \\ 0 & \dots \\ \vdots & \end{pmatrix}$$

Обозначаю $A^{(1)} = T_1 A$. Дальше зануляю как обычно следующий столбец

$$T_{2n}\dots T_{23}A^{(1)}.$$

Обозначаю всё же T_2. И так далее $T^T = T_n \dots T_1$, Выходит $Ax = b$ переходит в $T^T Rx = b$ или же $Rx = Tb$.

Когда будем вычислять углы, может быть деление на маленькие знаменатели. Но есть литература, в которой показывается, что здесь устойчивость больше, чем в методе Гаусса. Здесь тот же самый $O(n^3)$ как и в методе Гаусса.

11.2.2 Метод Отражений

Можно было и одним ограничиться. Но не каждый раздел высшей математики может похвастаться, то может быть объяснён на пальцах, но вещь серьёзная.

Метод отражений стоит особняком. Тут сразу за один проход будем весь столбец получать.

Мы работаем в $\mathbb{R}^n$. Рассмотрим единичный вектор $\overline{\omega}$, то есть $\|\overline{\omega}\|_2 = \sqrt{(\overline{\omega}, \overline{\omega})} = 1$. Составим такую матрицу

$$U = E - 2\overline{\omega}\overline{\omega}^T.$$

Это всё есть матрица с очень быстро вычисляемыми элементами. Какими полезными свойствами она обладает.

1. $U = U^T$.

   Как мы траспонируем сумму $(A+B)^T = A^T + B^T$. А если произведение $(AB)^T = B^T A^T$. Вот тут и всё.

2. $UU^T = E$.

   Оставлю в качестве домашнего задания. Очень полезное упражнение.

А что будет, если мы применим нашу матрицу U к какому-то вектору z. Вектор z мы можем разбиать на $z = z_1 + z_2$, причём $z_1 \perp \overline{\omega}$, ну а $z_2 \parallel \overline{\omega}$. Оказывается, что в результате получится

$$Uz = z_1 - z_2.$$

Отражение. Ортогональная составляющая без изменения, а параллельная меняется. Как это удобно показать

$$Uz_2 = z_2 - 2(z_2, \omega)\omega = z_2 - 2z_2 = -z_2.$$

А теперь поставим такую задачу. Пусть имеются два вектора: $\overline{S}$ и $\overline{e}$, причём $(\overline{e}, \overline{e}) = 1$. Вопрос: подобрать такой вектор ω, чтобы матрица отражения, построенная на этом векторе ω давала бы

$$U\overline{S} = \alpha\overline{e}.$$

Ну на самом деле решение здесь достаточно простое. Нужно, чтобы $U\overline{S} = \alpha\overline{e}$. Какая может быть α, ну например (можно и с минусом) $\alpha = \sqrt{(S, S)}$. Положим

$$\overline{\omega} = \frac{1}{\rho}(\overline{S} - \alpha\overline{e}),$$

где ρ: $\|\omega\|_2 = 1$. Проверите, что этот вектор подходит дома. Если один раз проделать руками, то дальше уже не забудете.

Как теперь это применить к решению системы. Пусть есть система $Ax = b$, где A невырождена. В качестве вектора $\overline{S}$ возьмём первый столбец

$$\overline{S} = \begin{pmatrix} a_{11} \\ \vdots \\ a_{n1} \end{pmatrix}$$

В качестве вектора $\overline{e}^T = (1, 0, \ldots, 0)$. Отсюда строим U_1. Домножаем слева на уравнение (обратите внимание, что матрица умножается дёшево)

$$U_1 Ax = U_1 b.$$

На месте первого столбца $\|a_1\|$ сосредоточена на первом же элементе получившейся матрицы, остальные в первом столбце элементы нули.

Теперь начинаем работать в $\mathbb{R}^{n-1}$. В качестве $\overline{S}^T = (a_{22}^{(1)}, \ldots, a_{n2}^{(1)})$, $\overline{e}^T = (1, 0, \ldots, 0) \in \mathbb{R}^{n-1}$. Далее

$$\begin{pmatrix} 1 & 0 \\ 1 & U_2 \end{pmatrix} U_1 Ax = \begin{pmatrix} 1 & 0 \\ 1 & U_2 \end{pmatrix} U1b.$$

Снова затраты $O(n^3)$, а при обратном ходе затраты $O(n^2)$. Последние два подхода предпочтительней, чем метод Гаусса. Есть такие матрицы, которые методом Гаусса не обрабатываются, а этими получается нормально. Есть конечно такие, где всё плохо. Есть такая матрица Гильберта

$$G_{ij} = \frac{1}{i + j - 1}.$$

До $n = 6$ ещё будет работать, дальше система сломается. Матрица отвратительно обусловлена. Причём алгоритм не кричит, что делит на ноль, он всё старательно считает. Но если потом результат подставить в систему, ничего похожего на правую часть вы не получите.

12 Лекция 12

Сегодня мы будем рассматривать ситуацию, когда матрица системы не является квадратной. Уже обсудили, что когда число уравнений меньше числа неизвестных, нам неинтересно. Вас уже на всех предметах этому обучали.

Интереснее, когда число уравнений больше числа неизвестных. Такая система может быть и совместной, но вообще говоря нет.

Определение 12.1. *Матрицы A и AB ортогонально подобны, если есть ортогональная Q: $QQ^T = E$, для которой $A = Q^T BQ$.*

Утверждение 12.1. *У ортогонально подобный собственные значения одинаковые*

Доказательство. Доказательство очень простое

$$|A - \lambda E| = |Q^T BQ - \lambda Q^T T| = |Q^T (B - \lambda E)Q| = |Q^T||B - \lambda E||Q| = |B - \lambda E|.$$

■

Отсюда QR-метод нахождения собственных значений. Доказательство очень сложно, я его рассказывать не буду.

Пусть матрица у нас простой структуры, то есть все собственные значения различны, $|\lambda_1| > \cdots > |\lambda_n|$. Тогда есть процедура превращения её в верхне треугольную, причём на диагонали будут собственные числа.

Положим $A^{(0)} = A$, $A^{(0)} = Q_0 R_0$. Обозначим $R_0 Q_0 = A^{(1)}$. Они, $A^{(0)}$ и $A^{(1)}$, действительно ортогонально подобны. Дальше процедура очень простая

$$A^{(1)} = R_1 Q_1; A^{(2)} = Q_1 R_1.$$

В матрице $A^{(k)}$ при $i > j$, то $|a_{ij}| \leqslant c \left|\frac{\lambda_i}{\lambda_j}\right|^k$, значит, на диагонали с ростом k будет что-то похожее на собственные числа.

12.1 Переопределённые системы

Чтобы их изучать, нужно знать сингулярное разложение матрицы. Пусть у нас имеется A размера $m \times n$, причём $m \geqslant n$.

Теорема 12.1. $\forall\ A(m \times n), m \geqslant n$, $\operatorname{rank} A = r\ \ \exists\ U(m \times m), V(n \times n)$ *ортогональные, для которых*

$$U^T AV = \Sigma$$

Где Σ матрица $m \times n$. Где сверху слева блок $\operatorname{diag}(\sigma_1, \ldots, \sigma_r)$*, остальные нули, а $\sigma_1 \geqslant \ldots \sigma_r$.*

Доказательства не будет. Рассмотрим

$$\Sigma\Sigma^T = U^T AVV^T A^T U = U^T AA^T U.$$

Это матрица $m \times m$. На диагонали квадраты сингулярных значений. Таким образом, сингулярные значния в квадрате, это собственные значения AA^T. С другой стороны

$$\Sigma^T\Sigma = V^T A^T UU^T AV = V^T A^T AV.$$

Это уже матрица $n \times n$. Ненулевые собственные числа образуют одинаковый набор и в AA^T и в $A^T A$. Вместо с нулевыми из разное количество.

Почему нельзя запускать QR алгоритм для AA^T, чтоы найти сингулярные значения. Пример

$$A = \begin{pmatrix} 1 & 1 \\ \beta & 0 \\ 0 & \beta \end{pmatrix}$$

Пусть $\varepsilon > 0$ и $\beta^2 < \varepsilon < \beta$. То есть β^2 мы в наших вычислениях не видим, а β вполне ощутима.

$$A^T A = \begin{pmatrix} 1 + \beta^2 & 1 \\ 1 & 1 + \beta^2 \end{pmatrix}$$

Если считать честно, то $\sigma_1(A) = \sqrt{2 + \beta^2}$, $\sigma_2(A) = |\beta|$. Если теперь на этом этапе счесть β^2 малым, то $\sigma_1(A) = \sqrt{2}$, $\sigma_2(A) = |\beta|$. Если бы мы сразу не видеть β^2, то матрица имеет вил

$$\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

Тут одно из собственных значений ноль. А это уже нехорошо.

Пусть у нас есть некоторая матрица $A(m \times n)$, $m > n$, $\operatorname{rank}(A) = n$. Формально напишем задачу

$$Ax = b.$$

В общем случае решать систему нельзя. Она имеет решение лишь в случае хорошего b, если b лежит в пространстве, натянутом на столбцы матрицы A. Записывать эту систему мы не можем.

Введём вектор невязки $\overline{r} = b - Ax$. И подбором x будем минимизировать его длину.

$$\|\overline{r}\| = \sqrt{\Sigma r_i^2}.$$

Чтобы не мучиться, минимизируют не корень, а сумму квадратов.

У нас $\overline{b} = \overline{b}_1 + \overline{b}_2$, $b_1 \in \operatorname{Ln}(A)$, $a_2 \perp \overline{a}_i$. Подберём $|hatx\colon (b - A\hat{x}) \perp \overline{a}_i$. Отсюда $A^T(\overline{b} - A\hat{x}) = 0$.

$$A^T A\hat{x} = A^T b.$$

Это называется нормальной системой.

Теперь с обратной стороны к этому подойдём. Тут интересна сама техника, а не результат. Те же условия на матрицу A. Рассмотрим просто так нормальную систему

$$A^T A\hat{x} = A^T b.$$

Обозначим $\hat{r} = b - A\hat{x}$, и без крышки $r = b - Ax$. Хотим доказать, что

$$\|\hat{r}\|_2^2 \leqslant \|r\|_2^2 \quad \forall\ x.$$

Берём произвольный вектор x. По нему строим $r = b - Ax = b - A\hat{x} + A\hat{x} - Ax$. Могу это немножечко причесать

$$r = \hat{r} + A(\hat{x} - x).$$

Берём вторую норму в квадрате

$$\|r\|_2^2 = (r, r) = r^T r = \big(\hat{r} + A(\hat{x} - x)\big)^T \big(\hat{r} + A(\hat{x} - x)\big) = \big(\hat{r}^T + (\hat{x} - x)^T A^T\big)\big(\big(\hat{r} + A(\hat{x} - x)\big).$$

Но так как $\hat{x}$ есть решение нормальной системы, то $A^T\hat{r} = 0$. Раскрываем и учитываем

$$\|r\|_2^2 = \hat{r}^T\hat{r} + (\hat{x} - x)^T A^T A(\hat{x} - x) = \|\hat{r}\|_2^2 + \big\|A(\hat{x} - x)\big\|_2^2 \geqslant \|\hat{r}\|_2^2.$$

Получили то же самое без геометрической интерпретации. Ещё тут единственность вылезла.

12.2 Погрешность

А как погрешность вычислений считать? Для квадратного случая у нас было понятие обусловленности матрицы. Тут попытаемся ввести что-то подобное.

Нам нужно какое-то подобие обратной матрицы.

Определение 12.2. *Пусть $A(m \times n)$, $m \geqslant n$, $\operatorname{rank}(A) = n$. Тогда $A^+(n \times m)$ назовём псевдообратной к матрице A, если $AA^+A = A$.*

Пусть $m = n$. Как можно определять обратную матрицу? Например через систему

$$Ax = b \Rightarrow x = A^{-1}b.$$

Если $m > n$, мы можем писать нормальную систему $A^T A\hat{x} = A^T b$. Её решение

$$\hat{x} = (A^T A)^{-1} A^T b.$$

Гипотеза такая: $A^+ = (A^T A)^{-1} A^T$.

Ну действительно $A(A^T A)^{-1} A^T A = A$. В случае матрицы полного ранга есть единственность.

Будем рассматривать две задачи наименьших квадратов. Исходную и невозмущённую.

$$\|b - Ax\|_2 \to \min, \quad \|\widetilde{b} - Ax\|_2 \to \min.$$

Матрица одна и та же, подпортили только правую часть. К ним нормальные системы

$$A^T Ax = A^T b; \quad A^T A\widetilde{x} = A^T\widetilde{b}.$$

Оказывается, всё не так просто. Разложим $b = b_1 + b_2$, причём $x_2 \perp \overline{a}_i$. То же самое с волнами $\widetilde{b} = \widetilde{b}_1 + \widetilde{b}_2$, $\widetilde{b}_2 \perp \overline{a}_i$.

Имеет место следующая теорема

Теорема 12.2. $\frac{\|x-\widetilde{x}\|_2}{\|x\|_2} \leqslant \operatorname{cond}_2(A) \frac{\|b_1-\widetilde{b}_1\|_2}{\|b_1\|_2}$, *где*

$$\operatorname{cond}_2(A) = \|A\|_2 \|A^+\|_2.$$

Доказательство. Икс есть решение нормальной системы

$$x = (A^T A)^{-1} A^T (b_1 + b_2),$$

причём $A^T b_2 = 0$. Значит, $x = A^+ b_1$. Аналогично $\widetilde{x} = A^+ \widetilde{b}_1$. Мы получили, что

$$x - \widetilde{x} = A^+ (b_1 - \widetilde{b}_1).$$

Отсюда

$$\|x - \widetilde{x}\|_2 \leqslant \|A^+\| \cdot \|b_1 - \widetilde{b}_1\|_2.$$

А как связаны x и b_1. Это просто $Ax = b_1$. Это именно правильная решаемая система и решение наши иксы. Таким образом, $\|b_1\| \leqslant \|A_2\| \|x\|$. Или,

$$\frac{1}{\|x\|} \leqslant \frac{\|A\|_2}{\|b_1\|}.$$

Вот мы всё и получили. ■

Забываем про псевдообратные конструкции. Решаем задачу

$$A^T A x = A^T b.$$

Рассматриваем правую часть как цельную. Левая часть имеет матрицу $A^T A$. У неё есть обусловленность.

Утверждение 12.2. $\operatorname{cond}_2(A^T A) = \operatorname{cond}_2^2(A)$.

Доказательство. Нам надо доказать, что $\|A^T A\| \|(A^T A)^{-1}\| = \|A\| \|A^+\|$.

Что такое квадрат нормы матрицы A

$$\|A\|_2^2 = \max_i \lambda_i(A^T A); \quad \|A^T A\|^2 = \max_i \lambda_i(A^T A).$$

Таким образом, $\|A\|_2^2 = \|A^T A\|_2$. Первую половину равенства мы доказали.

Попробуйте доказать, что $\|A\|_2 = \|A^T\|_2$.

И ещё в качестве задачи $A^+ (A^+)^T = (A^T A)^{-1}$.

А отсюда следует, что $\|A^+\|_2^2 = \|(A^T A)^{-1}\|_2$. ■

12.3 Как же её решать эту задачу МНК

Единственный разумный алгоритм для больших размерностей и плохих обусловленностей, это построение сингулярного разложение.

13 Как решать задачу МНК

Сегодня закончим вопросы переопределённых систем. Будем считать, что у нас имеется матрица A размера $m \times n$, причём чётко $m > n$. При этом пусть $\operatorname{rank}(A) \leqslant n$. Писать $Ax = b$ мы не можем. Ставим задачу наименьших квадратов

$$\min_x \|b - Ax\|_2^2.$$

Если $\operatorname{rank}(A) = n$, то можно написать нормальную систему, получить её решение $\hat{x}$. Подход, мы обсудили, плохой, он обусловленность матрицы возмодит в квадрат. Но хоть как-то.

Пусть теперь $\operatorname{rank}(A) < n$. Что тогда делать. Есть разные подходы. Мы рассмотрим один.

Итак работаем с системами, в которых либо $\operatorname{rank}(A) < n$, либо $\operatorname{rank}(A) = n$, то матрица плохо обусловлена.

Я не помню, давал ли такое свойство. Если нет, то в качестве упражнения.

Утверждение 13.1. *Пусть Q ортогональная, тогда $\forall\ x \quad \|Qx\| = \|x\|$.*

Возьмём норму невязки $\|Ax - b\|_2$. Нам её надо минимизировать. Пусть A размера $m \times n$. У нас есть сингулярное разложение

$$A = U \Sigma V^T,$$

где $U(m\times m)$, $V(n\times n)$ ортогональные, $\operatorname{rank}(A)=k$, Σ имеет диагональный блок слева сверху $k\times k$ и остальные нули.

$$\|Ax-b\|_2=\|AVV^Tx-b\|=\left\|U^T(AVV^T-b)\right\|_2=\|\Sigma\underbrace{V^Tx}_{z}-\underbrace{U^Tb}_{b}\|_2=\|\Sigma z-d\|.$$

Матрица Σ, слава богу, у нас диагональная, хоть и не квадратная, какая смогла быть.

Положим $z_j=\frac{d_j}{\sigma_j}$, где σ_j — сингулярное значение, $j=1,\ldots,k$. В этих обозначениях

$$\|Ax-b\|^2=\sum_{j=1}^{k}(\sigma_jz_j-d_j)^2+\sum_{j=k+1}^{n}(0\cdot z_j-d_j)+\sum_{j=n+1}^{m}d_j^2.$$

Таким образом z_j при $j=n+1,\ldots,n$ можно брать любые. Мы уже можем ещё сказать, что при $k=n$

$$\min\|Ax-b\|=\sqrt{\sum_{j=n+1}^{m}d_j^2}.$$

То есть если сингулярное разложение выполнено, то дальше всё достаточно легко. Но само разложение довольно нетривиально. Есть пакеты, вопрос только, сколько они будут работать.

13.1 Выравнивание данных методом МНК

Это пример обработки модели, основанной на МНК.

Предположим заказчик исследует физический или биологический закон. Причём формально $y=y(t)$. Всё, что заказчик может, поставить эксперимент. Получаем (t_i,y_i). Таких пар можно получить от заказчика сколько угодно.

Мы такие подумали и взяли систему линейно независимых функций $\{\varphi_i(t)\}$, $i=1,\ldots,n$. Модель будем строить такой

$$y(t)\approx\sum_{j=1}^{n}c_j\varphi_j(t).$$

Всё вроде хорошо. А дальше оказывается, что точек измерений на нас вывалили больше, чем n, чем то число, которое мы можем по каким-то причинам можно себе позволить в качестве количества линейно независимых функций. А эксперимент настолько дорогой, что заказчик непременно хочет использовать все его данные.

Попытаемся составить вектор невязок.

$$r_i=\sum_{j=1}^{n}c_j\varphi_i(t_i)-y_i.$$

Ну и попытаемся минимизировать его норму. То есть задачу поставим такую

$$\|\overline{r}\|_2^2\to\min.$$

Я ничего нового не рассказываю. Но на всякий случай формально напишем постановку задачи.

$$\|\overline{r}\|_2^2=\sum_{i=1}^{m}\left(\sum_{j=1}^{n}c_j\varphi_j(t_i)-y_i\right)^2$$

Переменными являются c_j. Нам надо считать

$$\frac{\partial\|\overline{r}\|_2^2}{\partial c_k}=0,\quad k=1,\ldots,n.$$

Получается

$$\sum_{j=1}^{n}\left(\sum_{i=1}^{m}\varphi_j(t_i)\varphi_k(t_i)\right)c_j=\sum_{i=1}^{m}y_i\varphi_k(t_i).$$

Здесь $k=1,\ldots,n$. Явно это система алгебраических уравнений. Надо её записать в более человеческом виде, чтобы можно было наши знания применять.

$$P\overline{C}=\overline{f},$$

где $P_{kj} = \sum\limits_{i=1}^{m} \varphi_k(t_i)\varphi_j(t_i)$, $f_k = \sum\limits_{i=1}^{m} y_i\varphi_k(t_i)$. Можно представить матрицу P в виде

$$P = \Phi^T\Phi, \qquad \Phi = \begin{pmatrix} \varphi_1(t) & \dots & \varphi_n(t) \\ \vdots & \ddots & \vdots \\ \varphi_1(t_n) & \dots \varphi_m(t_n) & \end{pmatrix}$$

При этом $\overline{f} = \Phi^T\overline{y}$. Можем решать задачу в такой постановке

$$\min_{\overline{C}} \|\Phi\overline{C} - \overline{y}\|.$$

Пусть теперь простейшая линейная задача метода наименьшах квадратов. Есть (t_i, y_i), $t = 1, \dots, m$. Пусть $n = 2$, $\varphi_1(t) = 1$, $\varphi_2(t) = t$. Таким образом

$$y(t) = c_1 + c_2 t.$$

Фактически мы строим прямую, до которой точки находятся максимально близко. Система имеет вид

$$P\overline{C} = \overline{f}, \quad P = \begin{pmatrix} m & \sum\limits_{i=1}^{m} t_i \\ \sum\limits_{i=1}^{m} t_i & \sum\limits_{i=1}^{m} t_i^2 \end{pmatrix}, \quad \overline{f} = \begin{pmatrix} \sum\limits_{i=1}^{m} y_i \\ \sum\limits_{i=1}^{m} t_i y_i \end{pmatrix}.$$

14 Итерационные методы

Решается система $Ax = b$, матрица невырожденная и $N \times N$. Проблема у нас здесь будет с памятью. Массив N^2 матрицы помещаться не будет. Либо можно хранить формулы, по которым коэффициенты матрицы считаются, либо матрица у нас будет разреженная: количество ненулевых элементов конечно и не зависит от N. Типичный представитель тридиагональная матрица, которую обрабатывали методом прогонки.

Если здесь затеять прямой метод, как метод Гаусса и вообще всё, что мы изучали, элементы матрицы начнут куда-то перемещаться, например, строку будем к строке прибавлять. А нам желательно проделывать процедуры, которые ничего не делают с самой матрицу A. Обращаться матрицу мы не будем, всё это под запретом. Матрица может только дёшево умножаться на вектор. Эта операция имеет порядок сложности N.

Заменим нашу систему на эквивалентную. Как правило вот такую используют

$$x = Bx + c.$$

Как это сделать, способов на самом деле вагон. $x = x - D(Ax - b)$. Вся экзотика заключается в матрице D, выбираем любую.

Если всё получилось, то берём x^0. А далее

$$x^{n+1} = Bx^n + C.$$

Ну и как выбрать матрицы, чтобы метод сходился, мы и будем обсуждать.

Начнём немножко не с начала. Рассмотрим систему $x = Bx + c$. Пусть $\overline{x}$ — решение. Возьмём какой-то x^0. И далее будем шагать $x^{n+1} = Bx^n + c$.

Пусть $\|B\| < 1$. Тогда мы знаем, что $\exists\ (E - B)^{-1}$. Тогда система $(E - B)x = c$ разрешима. Ну хорошо, а итерационный метод при этому сходится? Имеем

$$\overline{x} = B\overline{x} + c; \quad x^{n+1} = Bx^n + c, \quad x^0.$$

Спрашивается, верно ли, что $\|x^n - \overline{x}\| \to 0$.

Рассмотрим $z^n = x^n - \overline{x}$, $z^{n+1} = Bz^n$. Понятно, что

$$\|z^{n+1}\| \leqslant \|B\|\|z^n\| \Rightarrow \|z^n\| \leqslant \|B\|^n\|z^0\| \to 0.$$

Это всё хорошо, но норма это не матричный инвариант. Вдруг от выбора нормы от чего-то зависит. Вдруг мы получим в одной норме, что не сходится. Но может и в другой сходится.

Теорема 14.1. *Пусть $\exists!\ x\colon x = Bx + c$ (здесь $\|B\|$ может быть не обязательно меньше единицы). Тогда следующие утверждения эквивалентны*

1. *$x^{n+1} = Bx^n + \overline{c}$ сходится для всякого x^0;*
2. *Все собственные значения B по модулю меньше единички.*

Доказательство. Утверждение на самом деле сильное. Нам потребуется дополнительная конструкция.
Лемма 14.1. *Пусть* $\exists\, q > 0\colon \big|\lambda(\beta)\big| < q$ *(тут написано, что все собственные значения матрицы* B *по модулю меньше* q*). Тогда существует* $\Lambda = D^{-1}BD$*, где* $\|\Lambda\|_\infty < q$*,* D *— некоторая невырожденная матрица.*

Доказательство. Обозначим $\theta = q - \max\limits_i |\lambda_i|$, $\lambda(B) = \lambda_1, \lambda_2, \ldots$

Рассмотрим матрицу $\theta^{-1}B$. Какие у неё собственные числа

$$\lambda(\theta^{-1}B) \sim \theta^{-1}\lambda_i.$$

Приведём $\theta^{-1}B$ к жордановой нормальной форме. Что это такое. Есть некоторая невырожденная матрица $D\colon D^{-1}(\theta^{-1}B)D$ имеет вид

$$\begin{pmatrix} \theta^{-1}\lambda_1 & \alpha_{12} & & & \\ & \theta^{-1}\lambda_2 & \alpha_{23} & & \\ & & \ddots & \alpha_{34} & \\ & 0 & & \ddots & \ddots \end{pmatrix}$$

Здесь $\alpha_{ij} = 0, 1$. При этом $D^{-1}BD$ имеет вид

$$\begin{pmatrix} \lambda_1 & \alpha_{12} & & & \\ & \lambda_2 & \alpha_{23} & & \\ & & \ddots & \alpha_{34} & \\ & 0 & & \ddots & \ddots \end{pmatrix} = \Lambda.$$

Отсюда $|\lambda_j| + |\theta\alpha_{j,j+1} \leqslant |\lambda_j| + \theta < q$. □

Сначала достаточность. Пусть $\big|\lambda(B)\big| < 1$. Тогда найдётся $q\colon 0 < q < 1\colon \big|\lambda(B)\big| < q$. При этом $z^n = B^n z^0$. По лемме $B = D\Lambda D^{-1}$. При этом

$$B^n = D\Lambda^n D^{-1}.$$

Тогда $\|B^n\|_\infty \leqslant \|D\|\|\Lambda\|_\infty^n\|D^{-1}\|_\infty \leqslant \|D\|\|D^{-1}\|q^n \to 0$. Отсюда получаем что $\|z^n\|_\infty \leqslant \|B^n\|\|z^0\| \to 0$.

Теперь необходимость. Пусть у нас существует $\lambda_i\colon |\lambda_i| \geqslant 1$. Предъявим начальное приближение, при котором сходимость не получится. Обозначим решение $\overline{x} = B\overline{x} + c$, пусть $|\lambda_k| \geqslant 1$, Рассмотрим соответствующий собственный вектор $B\overline{e}^k = \lambda_k\overline{e}^k$. Пусть $x^0 = \overline{x} + \overline{e}^k$. Тогда $z^0 = x^0 - \overline{x} = \overline{e}^k$. Соответственно

$$z^1 = Bz^0 = \lambda_k\overline{e}^k, \quad z^n(\lambda_k)^n\overline{e}_k.$$

И у нулю эта норма стремиться совершенно не собирается.

■ Если стартовать с другого начального x^0, то ошибки при вычислениях, машинные, будут накапливаться и постепенно $\overline{e}^k$ появится. Однако можно каждый раз проецировать x^n на ортогональное дополнение $\overline{e}^k$.

Метод $x^{n+1} = Bx^n + c$ называется методом простой итерации.

Пусть есть $A = A^T > 0$ и мы собрались решать систему вида $Ax = b$. Хотим решать итерациями. Тогда простейший способ

$$x = x - \alpha(Ax - b).$$

Тогда итерационный процесс имеет вид

$$x^{n+1} = x^n - \lambda(Ax - b).$$

Вычитаем одно из другого и получаем формулу на погрешность

$$z^{n+1} = (E - \alpha A)z^n.$$

Так как $B = E - \alpha A$, то $B^T = B$ имеет место симметричность матрицы B. Мы хотим, чтобы

$$\big|\lambda(E - \alpha A)\big| < 1.$$

С другой стороны, а что такое вторая норма матрицы B

$$\|B\|_2 = \sqrt{\max\lambda(B^TB)}.$$

Так как матрица симметрична

$$\|B\|_2 = \sqrt{\max\lambda(B^2)} = \max\big|\lambda(B)\big|.$$

Очень удобно. А что такое собственные значения матрицы B. Имеем

$$\|B\|_2 = \max_{\lambda(A)} |1 - \alpha\lambda|.$$

Если мы сможем получить, что $\max\limits_{\lambda(A)} |1 - \alpha\lambda| \leqslant q$, то будет сходимость итерационного метода.

Возникает следующая минимаксная задача.

$$q = \min_{\alpha} \left(\max \left|1 - \alpha\lambda(A)\right| \right).$$

В таком виде эта задача нерешаема. Эта задача сложнее, чем решить систему. Огрубим задачу. У нас ведь $A = A^T > 0$. Тогда вдруг у нас получится найти оценку $0 < m \leqslant \lambda(A) \leqslant M < \infty$. Тогда

$$q \leqslant \min_{\alpha} \left(\max_{m \leqslant \lambda \leqslant M} \left|1 - \alpha\lambda\right| \right).$$

Ну давайте эту задачу решим как она есть. Рисуем графики нашего модуля с горизонтальной осью λ при разных α. Ответ такой: $\alpha_0 = \frac{2}{m+M}$. Тогда $q_0 = \frac{M-m}{m+M}$. И у нас получается оценка сходимости

$$\|z^n\|_2 \leqslant \left(\frac{M-m}{M+m}\right)^n \|z^0\|_2.$$

В практической ситуации такая оценка сходимости не работает. Она как правило получается q близко к единице.

15 3 декабря

Никак вас не отучу вставать. У нас итерационные методы решений линейных уравнений. Наш метод приспособлен для системы $Ax = b$, где $A = A^T > 0$. Было известно, что собственные значения жили на отрезке

$$0 < m \leqslant \lambda(A) \leqslant M < \infty.$$

Метод простой итерации

$$x^{n+1} = x^n - \alpha(Ax^n - b).$$

Было доказано, что оптимальное значение вот такое

$$\alpha = \frac{2}{M+m}.$$

Если обозначить $z^n = x^n - \overline{x}$, где $\overline{x}$ — решение, то $\|z^n\|_2 \leqslant q_0^n \|z^0\|_2$, где $q_0 = \frac{M-m}{M+m}$.

Почему это надо улучшать. Наши предположения о точных оценках m и M несут проблемы. Что такое q_0

$$q_0 = \frac{\frac{M}{m} - 1}{\frac{M}{m} + 1}.$$

А дробь $\frac{M}{m}$ есть оценка обусловленности матрицы. Если матрица плохо обусловлена, то q_0 близка к единицы.

Если

$$A = \begin{pmatrix} 2 & -1 & 0 & \dots \\ -1 & 2 & -1 & \\ 0 & \ddots & \ddots & \ddots \end{pmatrix}$$

Здесь $\mathrm{cond}_2(A) = O(N^2)$. Скорость сходимости будет такая, что ошибки вычислений на шаге не будут давать вообще методу сходиться.

Итак нам надо получить метод с более высокой скоростью сходимости. А также нужно смягчить входные данные. Ослабить условия на m и M.

Итак, что можно улучшить. Что здесь хромает. Что везде α одно и то же. За n шагов у нас будет n параметров теперь.

$$x^{n+1} = x^n - \alpha_{n+1}(Ax^n - b); \qquad \overline{x} = \overline{x} - \alpha_{n+1}(A\overline{x} - b).$$

Сразу ситуация ухудшится. Мы теперь не можем вычесть из одного уравнения другое. Раньше при вычитании вылезало z^n, а теперь

$$z^{n+1} = (E - \alpha_{n+1}A)z^n.$$

Отсюда

$$z^n = \prod_{j=1}^{n}(E - \alpha_j A)z^0 = P_n(A)z^0.$$

Получили последовательность матричных многочленов $P_n(A)$. От которых нам нужно лишь то, чтобы их норма стремилась к нулю. Какую норму брать? Опять удобнее будет взять вторую норму.

Как сформулировать задачу. По-прежнему минимаксная.

$$\|P_n(A)\|_2 = \max_{\lambda(A)} \big|P_n\big(\lambda(A)\big)\big| \leqslant \max_{m\leqslant\lambda\leqslant M} \big|P_n(\lambda)\big|.$$

Имеем

$$P_n(\lambda) = \prod_{j=1}^{n}(1 - \alpha_j\lambda)$$

Вспоминаем, что такое многочлены Чебышёва

$$T_n(x) = \begin{cases} \cos(n\arccos x), & |x| \leqslant 1; \\ \frac{(x+\sqrt{x^2-1})^n + (x-\sqrt{x^2-1})^n}{2}, & |x| > 1. \end{cases}$$

При этом отрезок $[m, M]$ переведём в $[-1, 1]$. Тогда $\lambda = \frac{M+m}{2} + \frac{M-m}{2}$. Отсюда выражаем x. Нужно только правильную нормировку сделать

$$P_n(\lambda) = \frac{T_n\left(\frac{2\lambda-(M+m)}{M-m}\right)}{T_n\left(-\frac{M+m}{M-m}\right)}$$

Можно вернутся к нашему доказательсву того, что приведённые многочлены Чебышёва наиболее близкие к нулю в соответствующем классе. Оно сюда перетаскивается.

У нас задача ставится так. Надо подобрать набор $\{\alpha_j\}_{j=1}^n$. Совпадение корней не даёт совпадение многочленов. Но мы уже отнормировали как надо. Итак, какие корни

$$\cos(n\arccos x) = 0 \Rightarrow n\arccos x = -\frac{\pi}{2} + \pi m \Rightarrow x_m = \cos\frac{\pi(2m-1)}{2n}, \quad m = 1, \ldots, n.$$

Надо в терминах λ написать. Нам надо, чтобы $P_n(\lambda_j) = 0$. То есть $\lambda_j = \frac{M+m}{2} + \left(\frac{M-m}{2}\right)\cos\frac{\pi(2j-1)}{2n}$. Мы хотим, чтобы это были корни нужного нам многочлена. Таким образом

$$\alpha_j = \frac{1}{\lambda_j}.$$

Итого, если воспользоваться положительной определённостью матрица A и границами спектра, то предлагаемый метод является лучшим. Эту задачу нельзя решить лучше.

Итак, $\|P_n(A)\|_2 \leqslant \left|\frac{T_n\left(x(\lambda)\right)}{T_n\left(x(0)\right)}\right| \leqslant \frac{1}{\left|T_n\left(x(0)\right)\right|}$.

Я напомнб, что $x(\lambda) = \frac{2\lambda-(M+m)}{M-m}$.

Имеем $\big|x(0)\big| > 1$. Осюда $T_n\big(x(0)\big) = \frac{\left(x(0)+\sqrt{x(0)^2-1}\right)^n + \left(x(0)-\sqrt{x(0)^2-1}\right)^n}{2}$. При этом $x(0) = -\frac{M+m}{M-m}$.

Мы должна написать такую вещь

$$-\left(\frac{M+m}{M-m}\right) \pm \sqrt{\left(\frac{M+m}{M-m}\right)^2 - 1} = \frac{-(M+m) \pm 2\sqrt{Mm}}{M-m} = -\frac{\left(\sqrt{M} \mp \sqrt{m}\right)^2}{M-m} = \left[\begin{array}{l} -\left(\dfrac{\sqrt{M}-\sqrt{m}}{\sqrt{M}+\sqrt{m}}\right) = -q_1; \\ -\left(\dfrac{\sqrt{M}+\sqrt{m}}{\sqrt{M}-\sqrt{m}}\right) = -\dfrac{1}{q_1}. \end{array}\right.$$

Таким образом есть два случая, но в любом выходит либо $-q_1$, либо $-\frac{1}{q_1}$.

$$q_1 = \frac{\sqrt{M}-\sqrt{m}}{\sqrt{M}-\sqrt{m}}.$$

Отсюда

$$\big\|P_n(A)\big\|_2 \leqslant \frac{1}{\left|\frac{(-1)^n}{2}(q_1^n + q_1^{-n})\right|} = \frac{2q_1^n}{1+q_1^{2n}} < 2q_1^n.$$

Получили такой метод. $\alpha_j = \frac{1}{\lambda_j}$, $j = 1..n$. А погрешность

$$\|z^n\|_2 < 2q_1^n \|z^0\|_2.$$

Гораздо лучше. Но всё равно плохо для той трёхдиагональной матрицы.

Итерационный метод с Чебышевским набором параметров Метод Ричардсона.

Получился такой итерационный метод

$$x^{n+1}x^n - \lambda_{n+1}(Ax^n - b).$$

К сожалению он работает по совокупности. Выбираем x^0, выбираем n. Находим $x^1, x^2, \dots, x^n$. Тогда

$$z^n = \prod_{j=1}^{n}(E - \alpha_j A)z^0.$$

Именно множитель будет мал по норме.

Но если мы хотим сделать следующий шаг, то у нас α_{n+1} уже нет.

Вообще z^n мы вичислить не можем. $z^n = x^n - \overline{x}$, а $\overline{x}$ мы не знаем.

$$\|z^n\|_2 < 2q_1^n \|z^0\|_2.$$

Что мы можем сделать. Найти номер n, для которого $2q_1^n \leqslant \varepsilon$. Но на практике может вылести всё что угодно.

Но часто смотрят в терминах невязки $\|r^n\| = \|Ax^n - b\|$.

Можно даже на каждом шаге смотреть на невязку. Если на n шаге она нас не устраивает, то делаем ещё раз, но уже в качестве x^0 берём то, что получилось на n-й операции.

Ещё есть проблема вычисления слишком больших чисел, которые не влезают в машинную арифметику. Когда мы вычисляем всё подряд, то есть α_j подряд, по тем формулам, которые я выписывал $\alpha_j = \frac{1}{\lambda_j}$, $j = 1..n$. Мы знаем, что

$$\prod_{j=1}^{n}(E - \alpha_j A)$$

по норме меньше единицы. Но это не значит, что всякая меньше единицы. Там могут быть несколько множителей, которые больше единицы по норме. И асимптотически всё хорошо. Но может быть шаг, где накопленная погрешность перевалит за максимально допустимую границу.

Будем перемешивать. Приведу алгоритм без доказательства.

15.1 Алгоритм перемешивания

Нам надо перемешать числа $1, 2, \dots, n$, для простоты пусть $n = 2^p$. Всё делаем по шагам. Берём последовательно $k = 1, \dots, p$. И каждый раз будем перемешивать числа $1, 2, \dots, 2^k$.

Пусть $k = 1$. Тогда $\{1, 2\}$ будет результатом перемешивания (тут не важно, но что-то надо выбрать).

Пусть для $(k-1)$ мы уже все $1, \dots, 2^{k-1}$ перемешали в $\{b_1^{(k-1)}, \dots, b_{2^{k-1}}^{(k-1)}\}$. Как тогда перемешать для k. Число элементов увеличивается в два раза. Делаем следующее

$$\{b_1^{(k-1)}, 2^k + 1 - b_1^{(k-1)}, b_2^{(k-1)}, 2^k + 1 - b_2^{(k-1)}, \dots\}$$

Для 16 имеем $\{1, 16, 8, 9, 4, 13, 5, 12, 2, 15, 7, 10, 3, 14, 7, 11\}$.

Итак у нас есть два алгоритма для решения системы. Они сходились как геометрическая погрессия с показателями

$$q_0 = \frac{M - m}{M + m} \sim \frac{\mathrm{cond}_2(A) - 1}{\mathrm{cond}_2(A) + 1}; \qquad q = \frac{\sqrt{\mathrm{cond}_2(A)} - 1}{\sqrt{\mathrm{cond}_2(A)} + 1}.$$

15.2 Другая норма

Пусть имеется матрица $A = A^T > 0$. Введём матрицу $A^{\frac{1}{2}}$. Попытаемся определить её таким образом $A^{\frac{1}{2}}A^{\frac{1}{2}} = A$. Существует ли такая?

Если матрица $A = A^T > 0$, то в принципе существует ортогональная Q, для которой $A = Q\Lambda Q^T$, Причём $\Lambda = \mathrm{diag}(\lambda_1, \dots, \lambda_N)$. Другое дело, что в общем случае конструктивно построить матрицу Q нельзя, но она существует. Определим

$$\Lambda^{\frac{1}{2}} = \mathrm{diag}(\sqrt{\lambda_1}, \dots, \sqrt{\lambda_N}).$$

Далее определим $A^{\frac{1}{2}} = Q\Lambda^{\frac{1}{2}}Q^T$. Нам такую матрицу строить не придётся, но мы будем постоянно пользоваться её наличием. Такая конструкция позволяет ввести своеобразную норму, которая оказывается очень эффективной.

Рассмотрим векторную норму $\|x\|_A = \sqrt{(Ax, x)}$.

Утверждение 15.1. *$\|x\|_A$ — норма.*

Доказательство. Можно выписывать аксиомы и проверять. Но можно и по-другому. Для всякого x

$$\|x\|_A = \sqrt{(A^{\frac{1}{2}}A^{\frac{1}{2}}x, x)}$$

Я утверждаю, что корень конструкция симметричная. $(A^{\frac{1}{2}})^T = A^{\frac{1}{2}} > 0$. Почему, ну распишите, через Λ и Q. Тогда

$$\|x\|_A = \sqrt{(A^{\frac{1}{2}}x, A^{\frac{1}{2}}x)} = \|A^{\frac{1}{2}}x\|_2.$$

■

Норму называют энергетической. Когда нет информации о границе спектра, её и используют.

15.3 Без границ спектра

Есть система $Ax = b$, $A = A^T > 0$. Будем строить итерационный метод такого вида

$$x^{n+1} = x^n - \alpha_{n+1}(Ax^n - b).$$

Из каких соображений будем его строить? Пусть точное решение $\overline{x} = \overline{x} - \alpha_{n+1}(A\overline{x} - b)$. Вычтем, получим погрешность

$$z^{n+1} = (E - \alpha_{n+1}A)z^n.$$

Предположим мы добрались x^n, как по нему построить α_{n+1}, чтобы $\|z^{n+1}\| \to \min$, то есть следующий шаг мы хотим сделать наилучшим образом.

15.4 Неудачная попытка

Сейчас будет попытка, которая закончится неудачной. Попытаюсь без энергетической нормы всё сделать. Это объяснит, почему в итоге всё будет так замысловато.

У нас есть $z^{n+1} = (E - \alpha_{n+1}A)z^n$. Возведём это в скалярный квадрат.

$$(z^{n+1}, z^{n+1}) = \big(E - \alpha_{n+1}A)z^n, E - \alpha_{n+1}A)z^n\big).$$

И раскрываем

$$\|z^{n+1}\|_2^2 = \|z^n\|_2^2 - 2\alpha_{n+1}(Az^n, z^n) + \alpha_{n+1}^2\|Az^n\|_2^2.$$

Отсносительно α_{n+1} это парабола с ветвями вверх. Минимум находится в вершине.

$$\alpha_{n+1} = \left(\frac{-b}{2a}\right) = \frac{(Az^n, z^n)}{(Az^n, Az^n)}.$$

Вопрос, а можем ли мы эту конструкцию посчитать?

$$Az^n = A(x^n - \overline{x}) = Ax^n - A\overline{x} = Ax^n - b = r^n.$$

Ну ничего. Отсюда

$$\alpha_{n+1} = \frac{(r^n, z^n)}{(r^n, r^n)}.$$

Остался z^n и мы упёрлись в тупик. Подход не прошёл, хотя идея была красивая.

15.5 Теперь как надо

Домножим $z^{n+1} = E - \alpha_{n+1}A)z^n$ на $A^{\frac{1}{2}}$. У нас если одна матрица в умножении единичная, то порядок умноженмия не важен. А также $A^{\frac{1}{2}}A = AA^{\frac{1}{2}}$. Получается

$$A^{\frac{1}{2}}z^{n+1} = (E - \alpha_{n+1}A)A^{\frac{1}{2}}z^n.$$

Возводим в скалярный квадрат

$$\|z^{n+1}\|_A^2 = \|z^n\|_A^2 - 2\alpha_{n+1}(AA^{\frac{1}{2}}z^n, A^{\frac{1}{2}z^n}) + \alpha_{n+1}^2\|AA^{\frac{1}{2}}z^n\|_2^2.$$

Вершина параболы теперь

$$\alpha_{n+1} = \frac{-b}{a} = \frac{(AA^{\frac{1}{2}}z^n, A^n z^n)}{(AA^{\frac{1}{2}}z^n, AA^{\frac{1}{2}}z^n)} = \frac{(Az^n, Az^n)}{(Az^n, A^2 z^n)} = \frac{(r^n, r^n)}{(r^n, Ar^n)}.$$

О сходимости пока не говорю.

16 Новый сем

Спасибо, что дождались. Так ну что, поздравляю, что преодолели сессию. Надеюсь, что все преодолели. У нас в конце семестра экзамен. У нас четыре лекции пропадают в праздничные дни. Буду ужимать материал. Обычно я в начале выдаю программу курса, теперь повременю. Сам не знаю, что ещё успею прочитать. Есть вещи, которые пропускать нельзя.

Сначала нелинейные системы, потом дифференциальные, потом некоторые виды уравнений в частных производных.

17 Решение нелинейных уравнений и систем

Бывают линейные, бывают квадратные. Бывают кубические, которые так-то тоже решаются. Остальные в общем виде не решаюся. Задачники по этому поводу: сборники исключений. А в общем виде, если у нас есть

$$f(x) = 0,$$

явной формулы у нас с вами не будет.

Давайте договоримся, что мы должны сделать. Найти x, для которого $f(x) = 0$, этого недостаточно.

Раскусим случай, когда решений нет и когда решений бесконечно много.

Сузим задачу. Пусть имеется интервал (a, b), есть $f(x)$. Будем искать алгоритм, который с заранее заданной точностью находит решение. А кратность корня? Многие алгоритмы отрабатывают хуже на кратных корнях, чем на простых.

Мы будем получать естественно некоторую последовательность $\{x_n\}$. Можно требовать сильную сходимость $x_n \to x$, а можно слабую $f(x_n) \to 0$. При этом необязательно стремление $x_n \to x$ при слабой сходимости.

На интервале (a, b) пусть имеется единственный корень уравнения $f(x) = 0$. Будет рассуждать в терминах сильной сходимости. В идеале надо задать $\varepsilon > 0$ и остановиться тогда, когда $|x_k - \overline{x}| < \varepsilon$.

Один из алгоритмов у вас сразу есть. Метод деления отрезка. Пусть $f(a)f(b) < 0$, тогда есть хотя бы один корень, делим отрезок пополам. Выбираем половину, где меняется знак. В конце концов мы оказываемся на отрезке достаточно малой длины. Середину берём за ответ. Всё замечательно, кроме того, что алгоритм медленно сходится. Убывает как геометрическая прогрессия с множителем $\frac{1}{2}$. Основной плюс простота. Если каждое вычисление функции дорогостоящая процедура, такое количество шагов может оказаться недопустимым.

Введём некий формализм. Чтобы получить приближение x_{n+1}, что надо сделать. От чего оно зависит? Самый простой вариант $x_{n+1} = \varphi(x_n)$ — одношаговый метод или метод простой итерайии. Или же $x_{n+1} = \varphi(x_n, x_{n-1}, \ldots, x_{n-k})$.

Начнём с одношагового. Если перейдём к пределу, вроде должны получить $\overline{x} = \varphi(\overline{x})$. То есть функцию φ нужно подобрать, а потобрать можно кучей способов. Как её строить? Само простое $x = x - g(x)f(x)$, где $f(x) \neq 0$ на (a, b). Это не единственный способ, но самый простой. Уже здесь неимоверное число способов. Но достаточно ли мы потребовали? Надо, чтобы была сходимость.

Предположим, что φ достаточно гладкая. Мы хотим оценить $x_n - \overline{x} = \varphi(x_{n-1}) - \varphi(\overline{x}) = \varphi(\xi_n)(x_{n-1} - \overline{x}) = \varphi'(\xi_n) \ldots \varphi'(\xi_1)(x_0 - \overline{x})$. Если мы потребуем, чтобы $\left|\varphi'(x)\right| \leqslant q < 1$, то $|x_n - \overline{x}| \leqslant q^n |x_0 - \overline{x}|$.

Можете меня укорить за то, что я поругал метод половинного деления. Но метод половинного деления вот так $x_{n+1} = \varphi(x_n)$ не выписывается. Там две точки важны и вообще целая песня.

Как то, что я анонсировал, может выглядеть на практике. Пусть сначала производная φ бегает от нуля до единицы. А как остановиться? $|x_{n+1} - x_n| < \varepsilon$ достаточно? Да кто его знает, может и нет.

Пусть теперь производная φ от минус единицы до нуля. Тогда Тут корень всегда между двумя соседними итерациями, то есть $\overline{x} \approx \frac{x_{k+1} + x_k}{2}$.

Как получить более высокую сходимость метода?

Под скоростью сходимости метода я буду понимать такую вещь

$$|x_{n+1} - \overline{x}| \leqslant C|x_n - \overline{x}|^m.$$

Если мы можем подобрать начальное приближение, что для всех остальных (последующих) существуют постоянные C и m, что это соотношение выполнено, то m называется скоростью сходимости метода. Если $m = 1$,

то метод сходится как геометрическая прогрессия и нам надо $C < 1$. Если $m > 1$, то на C ограничений меньше, зато больше условий для начального приближения.

А что надо требовать от φ, чтобы m оказалось больше единички? Начнём

$$x_{n+1} - \overline{x} = \varphi(x_n) - \varphi(\overline{x}) = \varphi\big(\overline{x} + (x_n - \overline{x})\big) - \varphi(\overline{x}) = \varphi(\overline{x}) + (x_n - \overline{x})\varphi'(\overline{x}) + \frac{(x_n - \overline{x})^2}{2}\varphi'(\overline{x}) + \ldots - \varphi(\overline{x}).$$

Если $\varphi'(\overline{x}) \neq 0$, то $m \approx 1$, $C \approx \varphi'(\overline{x})$. Согласуется с тем, что мы получали.

Если же у нас $\varphi'(\overline{x}) = 0$, $\varphi''(\overline{x}) \neq 0$. Тогда $m \approx 2$. Пишу нестрого, потому что потом будет целая теорема про метод второго порядка. Но набросок к действию у нас уже есть.

Попытаемся чуть по-другому действовать. Вот уравнение $f(x) = 0$, вот корень $\overline{x}$, то есть $f(\overline{x}) = 0$. Пусть $x_n \approx \overline{x}$. Представим его в виде корня уравнения, которое мы умеем решать. Самое простое, что мы умеем решить это линейное. Предположим, что у нас даже несколько точек x_n, x_{n-1}, … Через две точки простейшая идея провести секущую. Дальше следующая пара даёт секущую. На картинке получается очень хорошо. Можно выписать общую формулу, это в общем несложно.

$$x_{n+1} = x_n - \frac{x - x_{n-1}}{f(x_n) - f(x_{n-1})} f(x_n).$$

Для этого метода (при достаточно жёстких ограничениях) $m = \frac{\sqrt{5}+1}{2} \approx 1{,}62$.

Если проводить не секущие, а касательные, это называется методом Ньютона. Здесь формула

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

Если корень простой кратности, то $m = 2$.

Дальше возникает вопрос, а давайте теперь параболой функцию приближать. Пусть есть x_{n-2}, x_{n-1}, x_n. Можно взять параболу через эти точки. Вещественных корней может и не быть. Ну возьмём какой-нибудь (по единому на весь алгоритм) комплексный корень. Следующий многочлен будет уже с комплексными коэффициентами. И работаем до тех пор, пока разность последних двух по модулю меньше ε. Такие программые есть. Не найдено многочлена, на котором программа бы сломалась (за лет пятьдесят), но и нет доказательства, что на всех многочленах работает.

Можно повернуть ситуацию чуть по-другому. Предположим то же самое. Есть три приближения, взять параболу. Она обязательно в одной точке пересечёт ось y. Можем повернуть ситуацию $x = x(y)$. И такого сорта строим параболу $x = x(y)$. Ось иксов обязательно парабола пересечёт. Её и берём в качестве следующего приближений. Следующий шаг здесь всегда есть. А вот будет ли сходимость, это уже отдельный вопрос.

Можно продолжать приближать многочленами более высого порядка. Но это шаткий путь.

Вернёмся к методу Ньютона, который мы с вами обсудили

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

Здесь $\varphi(x) = x - \frac{f(x)}{f'(x)}$. Попробуем его прогнать через наши допущения. У нас пусть есть корень $f(\overline{x}) = 0$. Посчитаем

$$\varphi'(\overline{x}) = 1 - \frac{\big(f'(\overline{x})\big)^2 - f(\overline{x})f''(\overline{x})}{\big(f'(\overline{x})\big)^2} = 0.$$

У нас было рассуждение, что если производная ноль, то возможен метод порядка 2.

А теперь будет решать уравнение $f(x) = x^p$, $p \in \mathbb{N}, p \geqslant 2$.

Итак $x^p = 0$,

$$x_{n+1} = x_n - \frac{x_n^p}{p x_n^{p-1}} = \left(\frac{p-1}{p}\right) x_n.$$

Для нашего случая $m = 1$, $C = \frac{p-1}{p}$. Итак чем выше кратность корня, тем метод работает хуже. Это нас наводит на мысль, что делать с кратными корнями. Есть несколько подходов. Первый излишне энергичный. Вместо уравнения $f(x) = 0$ решаем $\frac{f(x)}{f'(x)} = 0$. Если такой процесс нас не устраивает (большие производные или не умеем считать вторые производные).

Другой способ. Взяли мы x_0. Сгенерировали последовательность $\{x_n^{(1)}\} \to \overline{x}^{(1)}$. Дальше берём другое начальное приближение и прогоняем алгоритм для $\frac{f(x)}{x - \overline{x}^{(1)}} = 0$. Получаем $x_0^{(2)}$ и последовательность $\{x_n^{(2)}\} \to \overline{x}^{(2)}$. Дальше снова делим на разность $x - \overline{x}^{(2)}$. Если два подряд одинаковый, одному добавляем кратость.

Дальше поговорим о системах. Единственный метод, который не зависит от размерности это метод Ньютона.

Давайте теперь поговорим. Решали нелинейные скалярные уравнения. Теперь нас интересует ситуация, когда вообще говоря

$$\begin{cases} f_1(x_1, \ldots, x_n) = 0; \\ \ldots\ldots\ldots\ldots\ldots \\ f_m(x_1, \ldots, x_n) = 0. \end{cases}$$

Мы должны с грустью убедиться, что почти все наши методы не переносятся. Даже метод половинного деления.

Было у нас $f(x) = 0$, было решение $\overline{x}$: $f(\overline{x}) = 0$. Пусть нам известно x_n. Как получить x_{n+1}? Разложим в ряд

$$f(x) \approx f(x_n) + (x - x_n)f'(x_n) = 0.$$

Положим это принудительно равным нулю. А тот x, что оттуда получится и назовём через x_{n+1}. Это метод Ньютона или метод касательных

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

Как нам это обобщить? Пусть есть X и Y — линейные нормированные пространства. Пусть есть нелинейный оператор $F(x)\colon X \to Y$. Пусть в пространствах есть нормы $\|\cdot\|_X$, $\|\cdot\|_Y$. Введём одно из возможных определений производной для этого самого оператора.

Определение 17.1. *Линейный оператор $P\colon X \to Y$ мы назовём производной, то есть $P = F'(x)$, если*

$$\big\|F(x+h) - F(x) - Ph\big\|_Y = o\big(\|h\|_X\big).$$

при $\|h\|_X \to 0$. Это одно из возможных определений. Оно нам будет удобно.

В нашем случае X и Y это $\mathbb{R}^m$. Здесь $x = (x_1, \ldots, x_m)^T \in \mathbb{R}^m$. Что такое оператор F, это

$$F(x) = \big(f_1(x), \ldots, f_m(x)\big)^T.$$

Что такое производная P? Это есть матрица

$$P = F'(x) = \left[\frac{\partial f_i}{\partial x_j}\right]_{i,j=1}^{m}.$$

Мы решаем уравнение $F(x) = 0$, где $0 \in Y$. Пусть $\overline{x}$ — решение. Пусть x^n — какое-то приближение. Пусть существует $F'(x_n)$. Что тогда у нас с вами получится. Будем считать, что x_n достаточно близок к корню, можем перефразировать (заменить о определении производнойй x на x^n, а $x + h$ на $\overline{x}$)

$$\big\| \underbrace{F(\overline{x})}_{0} - F(x^n) - F'(x^n)(\overline{x} - x_n)\big\|_Y = o\big(\|\overline{x} - x^n\|_X\big).$$

Итого у нас

$$-F(x^n) - F'(x^n)(\overline{x} - x^n) \approx 0.$$

Если мы захотим поставить там прямо равенство, придётся отказаться от $\overline{x}$. Поставим вместо $\overline{x}$ следующее приближение

$$F(x^n) = F'(x^n)(x^{n+1} - x^n) = 0.$$

Как выудить отсюда x^{n+1}? Надо домножить на обратную матрицу (предполагаем, что она есть). Выйдет у нас окончательно базовое соотношение

$$x^{n+1} = x^n - \big[F'(x^n)\big]^{-1} F(x^n).$$

Это тоже принято называть методом Ньютона. Размерность здесь не важна.

А теперь перед нами вопросы. Это вообще осуществимо? Сходится ли и с какой скоростью?

Редкая ситуация, когда формулировка теоремы зайём больше места, чем доказательство.

Обозначим $\Omega_a = \big\{x\colon x \in X,\ \|x - \overline{x}\|_X < a\big\}$.

Теорема 17.1. *Пусть существуют такие постоянные $a, a_1, a_2\colon a > 0, 0 \leqslant a_1, a_2 < \infty$ и существует $F'(x)$ для всех $x \in \Omega_a$, для которых*

(1) $\forall\ x \in \Omega_a\ \ \Big\|\big(F'(x)\big)^{-1}\Big\|_X \leqslant a1$ *(исключает много всего, в том числе кратные корни; $x^2 = 0$ нехорошо);*

(2) $\forall\ u_1, u_2 \in \Omega_a\ \ \big\|F(u_1) - F(u_2) - F'(u_2)(u_1 - u_2)\big\|_Y \leqslant a_2\|u_2 - u_1\|_X^2$ *(не просто о-малое, а убывает как квадрат, то $x^{\frac{3}{2}} = 0$ тоже не хорошо).*

Если всё это хозяйство выполнено, то обозначим $c = a_1 a_2, b = \min\{a, c^{-1}\}$. Тогда отсюда следует, что если $x^0 \in \Omega_b$, то метод Ньютона сходится и выполнено

$$\|x^n - \overline{x}\| \leqslant c^{-1}\big(c\|x^0 - \overline{x}\|_X\big)^{2^n},$$

то есть $\|x^{n+1} - \overline{x}\| \leqslant \mathrm{const}\,\|x^n - \overline{x}\|^2$.

На пальцах условия: пусть нет кратных корней и всё достаточно гладкое.

Доказательство. Считаем, что у нас $x^0 \in \Omega_b$. Это похоже на то, что мы делали с методом прогонки. Попытаемся показать, что из того, что $x^n \in \Omega_b$, следует, что и $x^{n+1} \in \Omega_b$. А в итоге докажем попутно сразу всё.

Положим $u_1 = \overline{x}$, $u_2 = x^n$. (Формулой Ньютона пока не пользуюсь.) Тогда второе условие даёт нам

$$\big\| \underbrace{F(\overline{x})}_{0} - F(x^n) - F'(x^n)(\overline{x} - x^n)\big\|_Y \leqslant a_2\|x^n - \overline{x}\|^2.$$

А дальше в скобках что стоит? Вспоминаем метод Ньютона

$$F(x^n) + F'(x^n)(x^{n+1} - x^n) = 0.$$

Из этой расчётной формулы выудим $F(x^n) = -F'(x^n)(x^{n+1} - x^n)$. И подставим

$$\big\|F'(x^n)(x^{n+1} - x^n) - F'(x^n)(\overline{x} - x^n)\big\| \leqslant a_2\|x^n - \overline{x}\|_X^2.$$

И мы уже получили то, с чем можно работать

$$\big\|F'(x^n)(x^{n+1} - \overline{x})\big\|_Y \leqslant a_2\|x^n - \overline{x}\|_X.$$

Если бы не было $F'(x^n)$, это была бы написана квадратичная сходимость. Покажем, что $x^{n+1} \in \Omega_b$.

$$\|x^{n+1} - \overline{x}\|_X = \Big\|\big(F'(x^n)\big)^{-1}F'(x^n)(x^{n+1} - \overline{x}\Big\|_X \leqslant$$
$$\leqslant \Big\|\big(F'(x^n)\big)^{-1}\Big\| \cdot \big\|F'(x^n)(x^{n+1} - \overline{x})\big\| \leqslant a_1 a_2\|x^n - \overline{x}\|^2 = c\|x^n - \overline{x}\|^2 < cb^2 = (cb)b < b.$$

Надо было остановиться на моменте $\|x^{n+1} - \overline{x}\|_X \leqslant c\|x^n - \overline{x}\|_X^2$.

Обозначим $q_n = c\|x^n - \overline{x}\|$. У нас получилось, что $\|x^{n+1} - \overline{x}\|_X \leqslant c\|x^n - \overline{x}\|_X^2$. Давайте домножим наше соотношение на c. Тогда

$$q_{n+1} \leqslant q_n^2.$$

Теперь попытаемся показать, что раз такое условие выполнено, то $q_n \to 0$. Но ведь $q_0 = c\|x^0 - \overline{x}\| < cb < 1$.

Я утверждаю, что $q_n \leqslant (q_0)^{2^n}$. Это совпадает с утверждением теоремы. Доказывать будем по индукции. База $n = 0$. Пусть $q_k \leqslant (q_0)^{2^k}$. Тогда

$$q_{k+1} \leqslant q_k^2 \leqslant \big((q_0)^{2^k}\big)^2 = q_0^{2^k}.$$

■

К сожалению, даже для скалярного уравнения, трудно поймать начальное приближение, для которого сходимость будет именно квадратичная. Часто начальное приближение находят опытным путём, как раз проверяя условия теоремы.

18 Дифференциальные уравнения

Рассмотрим для начала скалярную задачу Коши

$$\begin{cases} y'(x) = f\big(x, y(x)\big); \\ y(x_0) = y_0. \end{cases} \qquad x \in [x_0, X].$$

Часто нас будет интересовать поведение решения не везде, в только лишь в конечной точке.

Здесь есть несколько подходов. Мы будем искать не функцию, которая хорошо приближает решение. А будем искать, как решение ведём себя в конкретно заданных точках.

Возьмём на отрезке сетку шагом h, $x_n = x_0 + hn$, $x_0 + hN = X$. Если значения функции на сетке мы знаем, мы можем заменить производные разностными соотношениями. Скажем, что

$$y'(x_n) = \frac{y(x_{n+1}) - y(x_n)}{h} + O(h).$$

После этого можем записать $y(x_{n+1}) = y(x_n) + hf\big(x_n, y(x_n)\big) + O(h^2)$. К этому присовокупить $y(x_0) = y_0$.

Если бы мы знали $O(h^2$, то мы бы просто могли посчитать $y(x_{n+1})$. У нас возникает соблазн $O(h^2)$ просто убрать. Тогда мы не будем иметь право результать обозначать $y(x_n)$. Будем писать y_n. Получаем приближённый метод решения. Это так называемый метод Эйлера

$$y_{n+1} = y_n + hf(x_n, y_n), \quad y_0.$$

Вопрос, насколько метод эффективен. Что у нас будет происходить на самом деле? Первая ошибка будет порядка h^2. если бы второй шаг начинали с правильной точки, то тоже был бы h^2. Но мы начали второй шаг с неправильной точки. Насколько же быстро ошибка будет разрастаться?

Но метод реализуем хотя бы. Здесь нет деления на ноль, число шагов конечно, в бесконечность не уйдём. Вопрос только в том, что мы здесь получим в результате.

Нам нужно ввести некоторые определения, а именно классификацию погрешностей.

Введём глобальную погрешность $E_n = y_n - y(x_n)$. И введём понятие локальной погрешности $e_n = y_n - \widetilde{y}(x_n)$, где $\widetilde{y}$ определяется, как решение задачи Коши

$$\widetilde{y}\colon \begin{cases} \widetilde{y}'(x) = f\big(x, \widetilde{y}(x)\big); \\ \widetilde{y}(x_{n-1}) = y_{n-1}. \end{cases}$$

Теперь теоремка. Я упрощаю, а можно было и более жёстко дать.
Теорема 18.1. *Пусть $f(x, y) \in C^2$. Тогда отсюда следует, что $e_n = O(h^2$.*

Доказательство. Рассмотрим вот такую задачу

$$\begin{cases} y' = f(x, y); \\ y(x_n) = y_n. \end{cases}$$

При этом $y_{n+1} = y_n + hf(x_n, y_n)$. Что есть $y(x_{n+1}) = y(x_n + h) = y(x_n) + h \underbrace{y'(x_n)}_{f\left(x_n, y(x_n)\right)} + \frac{h^2}{2} y''(\xi)$.

Отсюда у нас

$$e_{n+1} = y_{n+1} - y(x_{n+1}) = -\frac{h^2}{2} y''(\xi).$$

Собственно, это даже не доказательство, а так просто. ■

Теперь исследуем глобальную погрешность. Тут начальные требования будут Жёстче.
Теорема 18.2. *Пусть $\forall\ x \in [x_0, X]\ \ |y''| \leqslant M$, $\big|f(x, \overline{y}) - f(x, \overline{\overline{y}})\big| \leqslant L|\overline{y} - \overline{\overline{y}}|$ Тогда*

$$E_n = O(h), \quad n = 1, \ldots, N.$$

Доказательство. Давайте попробуем это доказать. Мы можем расписать то, что у нас уже было

$$y(x_{n+1}) = y(x_n) + hy'(x_n) + \frac{h^2}{2} y''(\xi_n) = y(x_n) + hf\big(x_n, y(x_n)\big) + \frac{h^2}{2} y''(\xi_n).$$

Приближённое у нас есть $y_{n+1} = y_n + hf(x_n, y_n)$. И давайте вычтем

$$E_{n+1} = -\big(y(x_{n+1}) - y_{n+1}\big) = -\bigg(- E_n + h\Big(f\big(x_n, y(x_n)\big) - f(x_n, y_n)\Big) + \frac{h^2}{2} y''(\xi_n)\bigg).$$

Таким образом,

$$|E_{n+1}| \leqslant |E_n| + hL|E_n| + \frac{h^2}{2} M.$$

Обозначим, $A = 1 + Lh$, $B = \frac{h^2}{2}$. Тогда $|E_{n+1}| = AE_n + B$. И отсюда

$$|E_n| \leqslant A|E_{n-1}| + B \leqslant A^n E_0 + \left(\sum_{k=1}^{n-1} A^k\right) B.$$

При этом $E_0 = y_0 - y(x_0) = 0$. Рассмотрим $A = 1$ в качестве задачи. А если $A \neq 1$, то

$$|E_n| \leqslant \frac{A^n - 1}{A - 1} B.$$

Мы знаем, что $1 + x \leqslant e^x$. При этом $A^n = (1 + hL)^n \leqslant e^{Lhn} = e^{L(x_n - x_0)}$. Тогда

$$|E_n| \leqslant \left(\frac{e^{L(x_n - x_0)} - 1}{1 + Lh - 1} \right) \frac{Mh^2}{2} \leqslant \frac{hM}{2L} \big(e^{L(X - x_0)} - 1 \big) = O(h).$$

■<++> Рассматриваем численные методы решения задачи Коши.

$$\begin{cases} y'(x) = f(x, y); \\ y(x_0) = y_0. \end{cases} \quad , \quad x \in [x_0, X].$$

Мы рассмотрели метод Эйлера, где производная заменяется на простейшее разностное соотношение. Мы считали, что отрезок разбит сеткой. Не обязательно равномерной.

$$y_{n+1} = y_n + h_n f_(x_n, y_n), \quad y_0.$$

Пока будем считать, что $h_n = h$ не зависит от номера шага. У нас есть точность решения $E_n = y(x_n) - y_n$. Показали для метода Эйлера $E_n = O(h)$. Также рассмотрели глобальную погрешность $e_{n+1} = \widetilde{y}(x_{n+1}) - y_{n+1}$, где

$$\begin{cases} \widetilde{y}'(x) = f(x, \widetilde{y}), \\ \widetilde{y}(x_n) = y_n. \end{cases}$$

Выяснили, что $e_{n+1} = O(h^2)$.

18.1 Методы Рунге—Кутта

Та же самая задача Коши у нас остаётся. Я могу сразу выписать формулу, но попробую обосновать.

Будем считать, что мы находимся в такой ситуации: у нас есть сетка. Имеем точку x_n, дотопали. Хотим сделать один шаг в $x_{n+1} = x_n + h$. Предполагается знание решение только в одной точке. Будем минимизировать, стало быть, локальную погрешность, то есть считать, что в x_n значение функции мы знаем точно.

$$y_{n+1} = y(x_n) + \int\limits_{x_n}^{x_{n+1}} f\big(x, y(x)\big)\, dx.$$

Ага, говорим, интеграл мы умеем численно интегрировать. Давайте на отрезке $[x_n, x_{n+1}]$ возьмём как-нибудь узлы: $x_n^{(1)} = x_n + \alpha_1 h$, $x_n^{(2)} = x_n + \alpha_2 h$, $\dots x_n^{(m)} = x_n + \alpha_m h$, где $0 = \alpha_1 < \alpha_2 < \dots < \alpha_m \leqslant 1$. Построим квадратурную формулу. Способов много. Получим

$$y(x_{n+1}) \approx y(x_n) + h \sum_{i=1}^{m} c_i f\big(x_n^{(i)}, y(x_n^{(i)})\big).$$

Мы по этой формуле считать, не можем. мы же не знаем y в точках правее x_n. Давайте попробуем считать последовательно. Воспользуемся такой вот вещью

$$y(x_n^{(i)}) = y(x_n) + \int\limits_{x_n}^{x_n^{(i)}} dx.$$

Скажем, что

$$y(x_n^{(2)}) \approx y(x_n) + h\beta_{21} f\big(x_n^{(1)}, y(x_n^{(1)})\big).$$

Как-то потом подберём β_{21}. Ну то есть мы здесь по одному узлу посчитали и успокоились. Что делать дальше-то?

$$y(x_n^{(3)}) \approx y(x_n) + h\beta_{31} f\big(x_n^{(1)}, y(x_n^{(1)})\big) + h\beta_{32} f\big(x_n^{(2)}, y(x_n^{(2)})\big).$$

Ну и так далее, формула будет разрастаться.

Мы за погрешностью на этом этапе не пытались следить. Просто абы как выбрали формулы. А теперь давайте всё вместе объединим и в совокупности уже будем оценивать погрешности и подбирать лучшие коэффициенты из всей совокупности.

Фиксируем целое положительное число m. Затем по этому числу m мы фиксируем три набора параметров

$p_1, \dots, p_m$, $\alpha_2, \dots, \alpha_m$, β_{ij}, $0 < j < i \leqslant m$. Далее пишем

$$\begin{aligned} y_{n+1} &= y_n + \sum_{i=1}^{m} P_i k_i(h), \\ k_1(h) &= hf(x_n, y_n); \\ k_2(h) &= hf\big(x_n + \alpha_2 h, y_n + \beta_{21} k_1(h)\big); \\ &\vdots \\ k_m(h) &= hf\big(x_n + \alpha_m h, y_n + \beta_{m1} k_1(h) + \dots + \beta_{m,m-1} k_{m-1}(h)\big). \end{aligned}$$

Количество параметров, конечно, ужасает. Но мы же поняли уже, что это квадратурные формулы.

Теперь оценим ошибки. Мы считаем, что точное значение в x_n мы знаем, то есть $y(x_n) = y_n$. Введём функцию ошибки (локальную погрешность) $\varphi(h) = y(x_n + h) - y_{n+1}$. Безусловно $\varphi(0) = 0$.

Разложим функцию φ в тейлоровский ряд, чтобы как можно больше коэффициентов разложения ушли в ноль. То есть потребуем, чтобы $\varphi'(0) = \varphi''(0) = \dots = \varphi^{(s)}(0) = 0 \neq \varphi^{(s+1)}(0)$. Тогда $\varphi(h) = h^{s+1}\varphi^{(s+1)}(0) + O(h^{s+2})$.

Примем следующее без доказательства.

Теорема 18.3. *Глобальная погрешность метода $E_h = O(h^s)$.*

А как собственно эти методы строить? Занятие очень трудоёмкое. Настолько, что большинство методов носит именной характер. Мы построим для $m = 2$.

18.2 Построение метода Рунге—Кутта

Определение 18.1. *Величина s — порядок точности метода.*

Пусть сначала $m = 1$. Тогда

$$y_{n+1} = y_n + p_1 k_1(h), \quad k_1(h) = hf(x_n, y_n).$$

Давайте я, пока мы будем выводить методы, буду писать вместо $x_n, y_n, y(x_{n+1})$ соответственно $x, y, y(x + h)$.

$$\varphi(h) = y(x + h) - y(x) - p_1 hf(x, y).$$

Радостно видим, что $\varphi(0) = 0$. Теперь надо потребовать, чтобы $\varphi'(0) = 0$. Это мы хотим потребовать. Давайте выпишем вообще φ'. Что это вообще такое будет.

$$\varphi'(h) = y'(x + h) - p_1 f(x, y), \quad y' = f.$$

то есть $\varphi'(0) = f(x, y)(1 - p_1)$. То есть при $m = 1$, если мы хотим, чтобы производная была равна нулю, то надо требовать $p_1 = 1$. Вполне естественно. Дальше можно, конечно, поэкспериментировать и убедиться, что вторая производная не ноль. Соответственно, метод первого порядка.

18.2.1 При $m = 2$

Теперь посмотрим, что будет, если $m = 2$. Всё будет уже впечатляюще.

$$\begin{aligned} y_{n+1} &= y_n + p_1 k_1(h) + p_2 k_2(h); \\ k_1(h) &= hf(x_n, y_n); \\ k_2(h) &= hf\big(x_n + \alpha_2 h, y_n + \beta_{21} k_1(h)\big). \end{aligned}$$

Как выглядит функция ошибки

$$\varphi(h) = y(x + h) - y - p_1 hf(x, y) - p_2 hf\big(x + \alpha_2 h, y + \beta_{21} hf(x, y)\big).$$

Давайте обозначим $\overline{x} = x + \alpha_2 h$, $\overline{y} = y + \beta_{21} hf(x, y)$.

$$\begin{aligned} \varphi'(h) &= y'(x + h) - p_1 f(x, y) - p_2 f(\overline{x}, \overline{y}) - p_2 h\big[\alpha_2 f'_x(\overline{x}, \overline{y}) + \beta_{21} f'_y(\overline{x}, \overline{y}) f(x, y)\big]; \\ \varphi''(h) &= y''(x + h) - 2p_2\big(\alpha_2 f_x(\overline{x}, \overline{y}) + \beta_{21} f'_y(\overline{x}, \overline{y}) f(x, y)\big) - \\ &- p_2 h\big(\alpha_2^2 f''_{xx}(\overline{x}, \overline{y}) + 2\alpha_2\beta_{21} f''_{xy}(\overline{x}, \overline{y}) f(\overline{x}, \overline{y}) + \beta_{21}^2 f''_{yy}(\overline{x}, \overline{y}) f^2(\overline{x}, \overline{y}). \end{aligned}$$

Давайте вы мне поверите на слово, что третью производную не надо выписывать. Всё равно её сделать нулём не удастся.

$$\begin{aligned}\varphi'(0) &= (1-p_1-p_2)f(x,y);\\ \varphi''(0) &= (1-2p_2\alpha_2)f'_x(x,y)+(1-2p_2\beta_{21})f'_y(x,y)f(x,y).\end{aligned}$$

Порадуемся некоторое время. Мы хотим, чтобы в ноль обратились производные при достаточно широком классе правых частей. Итак у нас получается три уравнения на коэффициенты, вообще говоря, не линейных.

$$\begin{cases}1-p_1-p_2=0\\ 1-2p_2\alpha_2=0\\ 1-2p_2\alpha_{b1}=0.\end{cases}$$

Если мы сможем эту систему решить, то получим, что $\varphi(h)=\frac{h^3}{6}\varphi'''(0)+O(h^4)$. Из системы мы неизбежно получаем $\beta_{21}=\alpha_2$. Решений много. Завирсируем, допусим α_2. Тогда $\beta_{21}=\alpha_2$, $p_2=\frac{1}{2\alpha_2}$, $p_1=1-p_2$.

Итак, мы получаем метод с глобальной погрешностью h^2.

Если положить, $\alpha_2=1$, мы получим вот такой метод

$$\begin{cases}y_{n+1}=y_n+\frac{1}{2}(k_1+k_2);\\ k_1=hf(x_n,y_n);\\ k_2=hf(x_n+h,y_n+k_1).\end{cases}$$

Ещё симпатичная формула получается для $\alpha=\frac{1}{2}$.

18.2.2 Большие порядки

Что будет в этом случае. Мы рассмотрели $m=1$, получили $s=1$, рассмотрели $m=2$, получили $s=2$. Примем без доказательства примем, что для $m=3$ получим $s=3$.

$$\begin{cases}y_{n+1}=y_n+\frac{1}{6}(k_1+4k_2+k_3);\\ k_1=hf(x_n,y_n);\\ k_2=hf\left(x_n+\frac{h}{2},y_n+\frac{k1}{2}\right);\\ k_3=hf(x_n+h,y_n-k_1+2k_2)\end{cases}$$

Обратите внимание, что коэффициенты могут быть и отрицательны.

Для $m=4$ получится $s=4$. Это чаще всего называется методом Рунге.

$$\begin{cases}y_{n+1}=y_n+\frac{1}{6}(k+1+2k_2+2k_3+k_4);\\ k_1=hf(x_n,y_n);\\ k_2=hf\left(x_n+\frac{h}{2},y_n+\frac{k_1}{2}\right);\\ k_3=hf\left(x_n+\frac{h}{2},y_n+\frac{k_2}{2}\right);\\ k_4=hf\left(x_n+h,y_n+k_3\right).\end{cases}$$

А дальше получается, что при $m=5$ снова удаётся построить только метод $s=4$. И дальше такой явной прямой зависимости между m и s не наблюдается.

18.3 Правило Рунге

Мы уже два раза при вычисления, производных и интегрировании, использовали правила Рунге для оценки погрешности. Попробуем и сюда тоже применить правило Рунге.

Предположим, что мы зафиксировали m и построили метод Рунге—Кутты с порядком s.

$$y_{n+1}+y_n+\sum_{i=1}^{m}p_ik_i(h),$$

где k_i как-то вычисляются. Локальная погрешность

$$\varphi(h)=\frac{\varphi^{(s+1)}(0)h^{s+1}}{(s+1)!}+O(h^{s+2}).$$

Предлагается метод контроля над локальной погрешностью на шаге. Пусть у нас есть $\varepsilon > 0$. Сначала мы делаем шаг h, получили $y^{(1)} = y_n + \sum\limits_{i=1}^{m} p_i k_i(h)$. Теперь сделаем два шага по $h/2$, снова приходим в ту же точку, но со значением $y^{(2)}$.

Мы полагаем

$$y^{(1)} - y(x_n + h) \approx ch^{s+1}.$$

А вторым методом

$$y^{(2)} - y(x_n + h) \approx 2c\left(\frac{h}{2}\right)^{s+1}.$$

Эта c для нас вещь неизвестная. Это какая-то производная правой части уравнения. Но вот эти $y^{(1)}$, $y^{(2)}$ у нас есть. Вычитаем выражения, одно из другого.

$$ch^{s+1} \approx \frac{y^{(1)} - y^{(2)}}{1 - 2^{-s}} = \rho(h).$$

Далее мы сравнимаем $\rho(h)$ с ε. Если ρ меньше, считаем следующуу точку; если больше, уменьшаем шаг.

18.4 Ошибки в начальных данных

До сих пор, пока мы строили численные методы решения дифференциальных уравнений, мы не учитывали ошибок округления при арифметике. Мы считали, что все ошибки из того, что дифференциальное уравнение заменяем разностным.

В реальности надо учитывать округление. Рассмотрим дифференциальную задачу

$$\begin{cases} y'(x) = f\big(x, y(x)\big); \\ y(x_0) = y_0. \end{cases}$$

Вопрос учёта всех ошибок округления достаточно сложный. Разберём простейшую ситуацию: мы ошибаемся только в начальных условиях. И даже тут мы столкнёмся с серьёзными проблемами.

Я возьму специфическую задачу.

$$\begin{cases} y' = \lambda y; \\ y(0) = y_0. \end{cases}$$

Чему может равняться λ? У уравнения есть решение $y(x) = y_0 e^{\lambda x}$. Добавим дрожащей рукой возмущение

$$\begin{cases} \widetilde{y}' = \lambda \widetilde{y}; \\ Tily(0) = y_0 + \varepsilon. \end{cases}$$

Нам нужны $\lambda < 0$, иначе даже точные решения будут сильно отличаться при малых ε. Нам нужна устойчивая дифференциальная задача.

Если же $\lambda < 0$, исходная задача хорошая. Мы вправе требовать, чтобы наши численные методы были устойчивыми к возмущениям начальных данных.

Строгое определение устойчивости у нас появится только через пару лекций. Пока буду давать определения не совсем строгие.

Будем применять для нашей задачи метод Эйлера.

$$\begin{cases} \frac{y_{n+1} - y_n}{h} = f(x_n, y_n); \\ y_0. \end{cases}$$

То есть расчётная формула $y_{n+1} = y_n + h\lambda y_n$, y_0.

На ряду с этой задачей рассмотрим возмущённую $\widetilde{y}_{n+1} = \widetilde{y}_n + h\lambda\widetilde{y}_n$, $\widetilde{y}_0 = y_0 + \varepsilon$. При этом у нас строго $\lambda < 0$.

Что из себя представляет разность

$$\widetilde{y}_n - y_n = (1 + h\lambda)^n(y_0 + \varepsilon) - (1 + h\lambda)^n y_0 = (1 + h\lambda)^n \varepsilon.$$

То есть для того, чтобы ошибка не росла, нам надо, чтобы $|1 + h\lambda| \leqslant 1$, отсюда $h \leqslant \frac{2}{|\lambda|}$. Если ни о чём особо не задумываться, то ну ок, пусть такой шаг и будет. Но предположим, что у нас $|\lambda| \gg 1$, $\lambda < 0$. Тогда шаг будет очень маленький и до единицы мы будем проходить за очень даже ощутимое время.

18.4.1 Двумерный случай

Теперь попытаемся перейти к двумерной задаче. Проблему я обозначил, теперь увидим, как она приведёт нас в затруднительной положение.

$$\begin{cases} y_1'(x) = f_1(x, y_1, y_2); \\ y_2'(x) = f_2(x, y_1, y_2); \\ y_1(0) = y_0^1; \\ y_2(0) = y_0^2. \end{cases}$$

Пока не пробовали, но на самом деле решать умеем.

$$\begin{aligned} \frac{y_{n+1}^1 - y_n^1}{h} &= f_1(x_n, y_n^1, y_n^2); \\ \frac{y_{n+1}^2 - y_n^2}{h} &= f_2(x_n, y_n^1, y_n^2); \end{aligned}$$

Можно повторить усилия, получить, что локальная ошибка порядка h^2, глобальная — порядка h.

С ошибками в начальных данных у нас в одномерном случае возникло ограничение на шаг. Нужен достаточно малый шаг, чтобы задача была устойчивой. Такие задачи называют жёсткими. При этом на практике можно выделять отрезки, на которых знаем, что точное решение почти ноль и нечего его там считать.

Давайте какой-нибудь пример приведу.

Пусть $\boldsymbol{y} = (y_1, y_2)^T$, $\boldsymbol{y}'(x) = Ay(x)$, $\boldsymbol{y}(0) = \begin{pmatrix} 1 & 1 \end{pmatrix}^T$,

$$A = \begin{pmatrix} 998 & 1998 \\ -999 & -1999 \end{pmatrix}$$

Ой, этот пример не очень удачный.

Пусть всё-таки $\boldsymbol{y}(0) = \begin{pmatrix} 2 & 1 \end{pmatrix}^T$,

$$A = \begin{pmatrix} -2 & -998 \\ 0 & -1000 \end{pmatrix}$$

Решение имеет вид

$$\begin{aligned} y_1(x) &= e^{-2x} + e^{-1000x}; \\ y_2(x) &= e^{-1000x}. \end{aligned}$$

Чтобы считать y_2, нам нужен шаг $h \leqslant \frac{2}{1\,000}$. Но выкинуть отрезок мы не можем, потому что есть ещё y_1.

18.4.2 Методом Рунге—Кутта

У нас появилось понятия жёсткой задачи, когда есть условие на шаг для того, чтобы задача была устойчивой. Попытаемся взять что-то более сильное, чем метод Эйлера. Сможем ли хотя бы на порядок глобальной ошибки метода снизить ограничение на шаг, то будет хорошо.

Применияем

$$\begin{aligned} y_{n+1} &= y_n + \frac{1}{2}(k_1 + k_2); \\ k_1 &= hf(x_n, y_n); \\ k_2 = hf(x_n + h, y_n + h). & \end{aligned}$$

Решаем простейшую задачу $y' = \lambda y$, $\lambda < 0$, $y(0) = y_0$, $|\lambda| \gg 1$. И возмутим $\widetilde{y}(0) = y_0 + \varepsilon$.

$$y_n = y_{n-1} + \frac{1}{2}\big(\lambda h y_{n-1} + \lambda h(y_{n-1} + \lambda h y_{n-1})\big) = \left(1 + \lambda h + \frac{\lambda^2 h^2}{2}\right) y_{n=1} = \left(1 + \lambda h + \frac{\lambda^2 h^2}{2}\right)^n y_0.$$

Что в общем логично, получили кусок разложения $e^{\lambda h}$ в ряд. Разность имеет вид

$$\widetilde{y}_n - y_n = \left(1 + \lambda h + \frac{\lambda^2 h^2}{2}\right)^n \varepsilon.$$

Чтобы была устойчивость, нужно $\left|1 + \lambda h + \frac{\lambda^2 h^2}{2}\right| \leqslant 1$. Если решить, получим удручающий результат $h \leqslant \frac{2}{|h|}$. Ничего не изменилось.

Если взять классический метод Рунге с погрешностью порядка h^4.

$$\begin{aligned} y_{n+1} &= y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4); \\ k_1 &= hf(x_n, y_n); \\ k_2 &= hf\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}\right); \\ k_3 &= hf\left(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}\right); \\ k_4 &= hf(x_n + h, y_n + k_3). \end{aligned}$$

Тут если поковыряться, получим

$$\left|1 + \lambda h + \frac{h^2\lambda^2}{2} + \frac{\lambda^3 h^3}{6} + \frac{h^4 h^4}{24}\right| \leqslant 1.$$

Условие в итоге $h < \frac{2{,}785}{|\lambda|}$.

Надо понять, что такое всё-таки задача жёсткая. Оказывается, ограничение завивит не от метода, а от самой задачи.

18.5 Устойчивость

Идея такая. Есть у нас $\boldsymbol{y}'(x) = A(x)\boldsymbol{y}(x)$ система с начальным условием $\boldsymbol{y}(x_0) = \boldsymbol{y}_0$. Решается система на отрезке $x \in [x_0, x_0 + X]$. Назовём эту систему жёсткой, если выполнено три условия

1. $X \max\left|\lambda_i(A)\right| \gg 1$. (Собственные значения зависят от x, поэтому максимум берётся и по собственным значениям и по всем x.)
2. $X \max \mathrm{Re}\big(\lambda_i(A)\big) \sim 1$.
3. $X \max\Big|\operatorname{Im}\big(\lambda_i(A)\big)\Big| \sim 1$.

А что такое $\gg 1$? Смотря в какой ситуации мы находимся. С современным развитием техники $h \sim 10^{-6}$ — нормально. Никто наши страдания не поймёт. $h \sim 10^{-8}$, ну уже определённое напряжение возникает. Если $h \sim 10^{-12}$, это уже, конечно, недопустимо.

Хороший был бы семестровый курс на методы борьбы с жёсткими системами. Мы просто посмотрим, в какие стороны можно идти.

18.5.1 Метод решения проблемы на примере самой простой задачи

Давайте попробуем сдвинуться с другого пути. По-прежнему $y' = f(x, y)$, $y(0) = y_0$, $x \in [0, X]$.

Как был построен метод Эйлера? Взяли и заменили производную разностью вперёд, а могли взять разность назад. Никто нам не мешал написать

$$\frac{y(x+h) - y(x)}{h} = f\big(x + h, y(x+h)\big) + O(h).$$

К какой разностной схеме это приведёт?

$$\frac{y_{n+1} - y_n}{h} f(x_{n+1}, y_{n+1}).$$

Как вообще отсюда y_{n+1} получить? Чтобы об этом думать, надо сначала понять, а зачем это вообще нужно.

Это называется неявный метод Эйлера. Погрешность у него будет такая же, как у прямого.

Рассмотрим ту же задачу $y' = \lambda y$, $\lambda < 0$, $y(0) = y_0$. Тогда

$$y_n = y_{n-1} + h\lambda y_n.$$

Выражаем отсюда y_n (волноваться при $\lambda < 0$ не приходится, на ноль не делим)

$$y_n = \frac{y_{n_1}}{1 - h\lambda} = \cdots = \frac{y_0}{(1 - h\lambda)^n}.$$

А разность имеет вид

$$|\widetilde{y}_n - y_n| = \left|\frac{\varepsilon}{(1 - h\lambda)^n}\right| < |\varepsilon|.$$

Получили метод, который для нашей задачи просто устойчивый.

Можно ли сформулировать класс задач с такой же проблемой. Среди задач, которые не являются линейными, например. Задача

$$f'_y < 0; \quad |f'_y| \gg 1.$$

Для такой задаче неявный методй Эйлера уже не очень хорошо. Как выуживать y_{n+1}? Эту проблему мы унесём на следующую лекцию.

18.5.2 Альтернативный подход решения проблемы неустойчивости

Даже явный метод Эйлера можно переделать так, чтобы метод стал устойчивым. Метод именной, называется методом Лебедева. Мы доказывать ничего не будем, доказательство там не тривиальное.

Рассмотрим задачу

$$y' + My = 0, \quad M \gg 1, \quad y(x_0) = y_0.$$

Хотим использовать явный метод Эйлера

$$\frac{y_{n+1} - y_n}{h} + My_n = 0.$$

Нам известно, что тогда для устойчивости нужно $h \leqslant \frac{2}{M}$.

Пусть мы можем сделать только N шагов, тогда мы можем только ушагать до $x_0 + \frac{2N}{M}$. Но давайте попробуем начать менять шаг. Хуже не будет.

$$\frac{y_{n+1} - y_n}{h_{n+1}} + My_n = 0.$$

Тогда ограничение на шаги имеет вид

$$\big|(1 - h_1M)(1 - h_2M)\dots(1 - h_NM)\big| \leqslant 1.$$

При этом ограничении решим задачу $\sum\limits_{i=1}^{N} h_i \to \max$.

Оказывается, что прошагаем мы до $x_0 + \frac{2(2N)^2}{M}$. Это существенное улучшение.

Всё, конечно, гораздо тяжелее. Если система, нужны гарантии, чтобы не было дополнительных причин для роста погрешности. Просто обратите внимание, что бывает взгляд и с совсем другой стороны.

19 Линейные уравнения в частных производных

Мы сформулировали аппроксимацию, устойчивость и следующую из них сходимость (это следование мы доказали).

Общий вид задачи $Lu = f, lu = \varphi$. Всё происходит в области Ω.

Разностная задача на сетке

$$\begin{cases} L_h\omega_h = f_h, & \Omega_h; \\ l_h\omega_h = \varphi_h. \end{cases}$$

Нам надо разобраться со сходимостью. Мы под ней понимали, что $\big\|[u]_h - u_h\big\|_{\Omega_h} \to 0$. Теперь у нас теорема

Теорема 19.1. *Из аппроксимации и устойчивости следует сходимость.*

Работает всегда, но докажем только для линейных.

Доказательство. Пусть у нас аппроксимация с порядком $p = (p_1, p_2)$ (это мультииндекс). Проделаем такой фокус. Сеточный оператор применим вот к такой штуке

$$L_h\big(\underbrace{u_h - [u]_h}_{w_h}\big) = L_hu_h - L_h[u]_h \pm [Lu]_h = \big([Lu]_h - L_h[u]_h\big) + \big(f_h - [f]_h\big) = z_h.$$

Аналогично для оператора на границе

$$l_h\big(u_h - [u]_h\big) = \big([lu]_h - l_h[u]_h\big) + \big(\varphi_h - [\varphi]_h\big) = \zeta_h.$$

У нас получилось

$$\begin{cases} L_hw_h = z_h, & \Omega_h; \\ l_hw_h = \zeta_n. \end{cases}$$

Для этой задачи у нас есть оценка нормы решения через нормы правых частей.

Так как у нас есть устойчивость, то $\|w_h\| \leqslant C\big(\|z_h\| + \|\zeta_h\|\big)$. А так как есть аппроксимация, то это всё $\leqslant c_1 h^p$. ■

Обычно с аппроксимацией у нас будет всё в порядке. А вот наличие устойчивости будет представлять интерес.

19.1 Задачи

Рассмотрим первую простейшую задачу

$$\begin{cases} \frac{\partial u}{\partial t} - \frac{\partial u}{\partial x} = f(x,t); \\ u(x,0) = \varphi(x). \end{cases}$$

Решать будем в полосе $[0,T]$. Вы должны быть готовы на экзамене ответить на вопрос, что будет, если перед $\frac{\partial u}{\partial x}$ добавить множитель a.

Выбираем шаг h и шаг τ так, что по времени отрезок разбивается на целое число частей. Будем приближать $u(mh, n\tau) \sim u_m^n$.

Здесь есть два простейших способа замена дифференциального оператора на разностный. Разности вперёд, разность назад.

$$\begin{cases} \frac{u_m^{n+1}}{\tau} - \frac{u_{m+1}^n}{u_m^n} = f_m^n; \\ u_m^0 = \varphi_m. \end{cases} \tag{3}$$

$$\begin{cases} \frac{u_m^{n+1}}{\tau} - \frac{u_m^n}{u_{m-1}^n} = f_m^n; \\ u_m^0 = \varphi_m. \end{cases} \tag{4}$$

В дифференциальной задаче на линиях $x + t = \text{const}$ значения постоянны. Значение на точке $x = 0, t = T$ определяется значением в точке $t = 0, x = T$. Если мы в последней внесём возмущение, оно перейдём в точку $x = 0, t = T$.

В приближении второго типа значение в рассматриваемой точке определеяется значением в трегольнике слева. Возмущение в точке $t = 0, x = T$ не передаётся в точку $x = 0, t = T$. Отбрасываем вторую схему.

Рассмотрим первую схему. Если $\frac{\tau}{h} \leqslant 1$, то задача по крайней мере не разваливается (характеристика попадает в треугольник, от которого зависит точка последнего слоя).

Итак у нас есть подозрение, что первая схема годится. Но мы ещё ничего не доказали. Введём

$$\|u_h\| = \max_n \sup_m |u_m^n|, \quad 0 \leqslant h \leqslant N, N\tau = T.$$

Такая же для $\|f_h\| = \max\limits_n \sup\limits_m |f_m^n|$. И ещё

$$\|\varphi_h\| = \sup_m |\varphi_m|, \quad \|u^n\| = \sup_m = |u_m^n|.$$

Если нам нужно считать не на всей числовой прямой по x верхний слой, то нижний слой задаём на большем промежутке, чем верхний.

Теорема 19.2. *Если $\tau/h \leqslant 1$, то схема один устойчива.*

Доказательство. Пусть $\tau/h = q \leqslant 1$. Тогда $u_m^{n+1} = (1-q)u_m^n + qy_{m+1}^n + \tau f_m^n$.

$$|u_m^{n+1}| \leqslant (1-q)|u_m^n| + q|u_{m+1}^n| + \tau|f_m^n| \leqslant \|u^n\|(1 - q + q) + \tau\|f_h\|.$$

Надо перейти к супремуму

$$\|u^{n+1}\| \leqslant \|u^n\| + \tau\|f_h\|, \quad \|u^n\| \leqslant \|u^{n-1}\| + \tau\|f_h\|, \quad \dots \quad \|u^1\| \leqslant \|u^0\| + \tau\|f_h\|.$$

Мы это дело всё сложим

$$\|u^{n+1}\| \leqslant \|\varphi_h\| + (n+1)\tau\|f_h\| \leqslant \|\varphi_h\| + T\|f_h\|.$$

Итак, получили

$$\|u_h\| \leqslant C\big(\|\varphi_h\| + \|f_h\|\big), \quad C = \begin{cases} 1, & T < 1; \\ T, T \geqslant 1. \end{cases}$$

Интересно посмотреть, что будет, если добавить в уравнение параметр a. ■

Смогли отбросить одну из схем, потому что знали, как решать уравнение. Но не всегда же мы умеем уравнение решать.

19.2 Спектральная устойчивость

Вернёмся. Распишем нашу задачу

$$\begin{cases} \frac{\partial u}{\partial t} - a\frac{\partial u}{\partial x} = f; \\ u(x,0) = \varphi. \end{cases}$$

Будем считать, что у нас $a > 0$. Напишем аналог первой схемы.

$$\begin{cases} \frac{u_m^{n+1} - u_m^n}{\tau} - a\frac{u_{m+1}^n - u_m^n}{h} = f_m^n; \\ u_m^0 = \varphi_m. \end{cases}$$

На этом примере я сейчас расскажу спектральный признак. Но сначала рассмотрим такую задачу

$$\begin{cases} \frac{u_m^{n+1} - u_m^n}{\tau} - a\frac{u_{m+1}^n - u_m^n}{h} = 0; \\ u_m^0 = e^{im\alpha}. \end{cases}$$

Здесь $i^2 = -1$, $\alpha \in [0, 2\pi]$. Будем говорить, что имеется выполнение спектрального признака, если решение будет устойчивость (ограничено по норме сверху через правую часть и правую часть на границе) любой такой задачи. То есть устойчивость в такой задаче это просто ограниченность решения.

$$u_m^1 = u_m^0 + \frac{a\tau}{h}(u_{m+1}^0 - u_m^0) = e^{im\alpha} + \frac{a\tau}{h}\left(e^{i(m+1)\alpha} - e^{im\alpha}\right) = e^{im\alpha}\left(1 - \frac{a\tau}{h} + \frac{a\tau}{h}e^{i\alpha}\right) = e^{im\alpha}\lambda.$$

Аналогично можно показать, что $u_m^2 = u_m^1\alpha = e^{im\alpha}\lambda^2$. И в конце концов $u_m^n = \lambda^n e^{im\alpha}$. Получили, что у нашей задачи λ выступает в роли собственного значения. Отсюда название признака «спектральный».

Итак, если $|\lambda| \leqslant 1$, то спектральный признак выполнен. Это сильная форма.

Но если мы работаем в полосе, то можно задать лишь слабую форму $|\lambda| \leqslant 1 + C\tau$. В этом случае

$$|u_m^n|(1 + c\tau)^n \leqslant \left(1 + c\frac{\tau}{N}\right)^N \leqslant e^{cT}.$$

На слабую форму у нас времени особо нет. Будем говорить о сильной.

19.3 Будет ли спектральная устойчивость в нашей задаче

Итак

$$\lambda 1 - \frac{a\tau}{h} + \frac{a\tau}{h}e^{i\alpha}.$$

Это окружность с центром $1 - \frac{a\tau}{h}$ и радиусом $\frac{a\tau}{h}$. Нам нужно, чтобы вся эта окружность лежала внутри единичного круга.

Если $\frac{a\tau}{h} \leqslant 1$, то всё хорошо.

20 Уравнение теплопроводности

Уравнение теплопроводности имеет вид

$$\frac{\partial u}{\partial t} = a^2\frac{\partial^u}{\partial x^2} + f(x,t).$$

Если мы рассматриваем задачу в полосе $[0,T]$, то достаточно взять начальное условие $u(x,0) = \varphi(x)$.

Будем брать следующую схему

$$\begin{cases} \frac{u_m^{n+1} - u_m^n}{\tau} = a^2\frac{u_{m+1}^n - 2u_m^n + u_{m-1}^n}{h} + h_m^n; \\ u_m^0 = \varphi_m. \end{cases}$$

На ряду с этой, рассмотрим схему

$$\begin{cases} \frac{u_m^{n+1} - u_m^n}{\tau} = a^2\frac{u_{m+1}^{n+1} - 2u_m^{n+1} + u_{m-1}^{n+1}}{h} + h_m^{n+1}; \\ u_m^0 = \varphi_m. \end{cases}$$

В первой схеме всё логично, чтобы получить информацию о верхнеё точке, пользуемся информацией уже известных нижних. Вторая схема какая-то странная.

Будем брать $u_m^0 = e^{im\alpha}$ будем получать $u_m^n = \lambda^n e^{im\alpha}$. Для первой схемы выполнение спектрального признака только при $\tau \sim h^2$ (задача для читателя получить точную оценку). Во второй схеме получим безусловную спектральную устойчивость.

А если мы будем рассматривать задачу на прямоугольнике $[0, X] \times [0, T]$. Нужны ещё условия на левой и правой границе $\mu_1(t)$ и $\mu_2(t)$. В этом случае какие условия у нас добавятся.

Условие здесь будет общее для обеих схем

$$u_m^0 = \varphi_m, u_0^n = \mu_1^n,\ u_M^n = \mu_2^n.$$

В первой схеме у нас $O(M)$ арифметических операций для расчёта каждого слоя. На второй схеме надо решать систему уравнений. $u^{n+1} = (u_1^{n+1}, \dots, u_{M-1}^{n+1})$. Хочу написать $Au^{n+1} = F^n$.

$$-\frac{a^2\tau}{h^2}u_{m-1}^{n+1} + \left(1 + \frac{2a^2\tau}{h^2}\right)u_m^{n+1} - \frac{a^2\tau}{h^2}u_{m+1}^{n+1} = \tau f_m^{n+1} + u_m^n.$$

Как будет выглядеть матрица. Она будет трёхдиагональная. Решаем методом прогонки, получаем $8M + O(1)$. То есть она по порядку аримфетических операций не уступает явному методу. Теперь вместо a поставлю единичку. Задача в конечной области $[0, T] \times [0, X]$.

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + f.$$

Есть граничные условия $u(x, 0) = \varphi(x)$, $u(X, t) = \mu_2(t)$, $u(0, t) = \mu_1(t)$. Шаг $Mh = X$, $N\tau = T$.

Выписали две схемы

$$\frac{u_m^{n+1} - u_m^n}{\tau} = \frac{u_{m-1}^n - 2u_m^n u_{m+1}^n}{h^2} + f_m^n; \qquad u_m^0 = \varphi_m, \quad u_0^n = \mu_1^n, \quad u_M^n = \mu_2^n.$$

Вывод, который мы сделали. Если условие $\tau \leqslant \frac{h^2}{2}$ не выполнено, то схема неустойчива.

Теперь сформулируем теорему.

Теорема 20.1. *Если $\tau \leqslant \frac{h^2}{2}$, то схема устойчива.*

Соглашение о нормах прежние, то есть $\|u_h\| = \max\limits_{m,n} |u_m^n|$, $\|u^n\| = \max\limits_m |u_m^n|$, $\|\varphi_h\| = \max\limits_m |\varphi_n|$, $\|\mu_k\| = \max\limits_n |\mu_k^n|$.

Доказательство. Нужно показать, что норма разности решений разностной задачи оценивается через норму разности начальных условий. Обозначим $\frac{\tau}{h^2} = \rho$. Тогда

$$u_m^{n+1} = (1 - 2\rho)u_m^n + \rho u_{m-1}^n + \rho u_{m+1}^n + \tau f_m^n.$$

Предполагаем, что $\|u^{n+1}\| = |u_0^{n+1}| \bigvee |u_M^{n+1}|$. В этом случае $\|u^{n+1}| \leqslant \max(\|\mu_1\|, \|\mu_2\|)$.

Теперь пусть максимум достигается внутри $\|u^{n+1}\| = |u_{m_0}^n|$, где $1 \leqslant m_0 \leqslant M - 1$.

$$\|u^{n+1}\| \leqslant \max_m \left(\left|(1 - 2\rho)u_m^n + \rho u_{m-1}^n + \rho u_{m+1}^n + \tau f_m^n\right|\right) \leqslant (1 - 2\rho + 2\rho)\|u^n\| + \tau\|f_m^n\|.$$

Таким образом, $\|u^{n+1}\| \leqslant \max\left(\|\mu_1\|, \|\mu_2\|, \|u^n\| + \tau\|f_h\|\right)$.

Дальше надо аккуратно. Наше существующее и единственное решение разобъём на два слагаемых $u_h = y_h + v_h$. При этом $L_h y_h = 0$, $y_h|_{t=0} = \varphi_h$, $y_h\big|_{x=0} = \mu_1$, $y_h|_{x=X} = \mu_2$ и $L_h v_h = f_h$, $y_h|_{t=0} = y_h|_{x=0,X} = 0$.

Берём y. С ним будет посложнее

$$\|y^{n+1}\| \leqslant \max\left(\|\mu_1\|, \|\mu_2\|, \|y^n\|\right) \leqslant \max\left(\|\mu_1\|, \|\mu_2\|, \|y^{n-1}\|\right) \leqslant \max\left(\|\mu_1\|, \|\mu_2\|, \|\varphi_h\|\right).$$

Теперь для v. Тут совсем оценка приятная

$$\|v^{n+1}\| \leqslant \|v^n\| + \tau\|f_h\| \leqslant \|v^{n-1}\| + 2\tau\|f_h\| \leqslant \ldots \leqslant \|v^0\| + \sum_{k=0}^{n} \tau\|f_h\| \leqslant T\|f_h\|.$$

Значит, схема устойчива. ■

20.1 Устойчивость неявной схемы

Вот такая схема

$$\frac{u_m^{n+1} - u_m^n}{\tau} = \frac{u_{m-1}^{n+1} - 2u_m^{n+1} + u_{m+1}^{n+1}}{h^2} + f_m^{n+1}.$$

Хотим показать, что схема безусловно устойчива, то есть неважно, как именно мы выбираем шаги.

Запишем схему таким же образом.

$$u_m^{n+1} + \frac{\tau}{h^2}(-u_{m-1}^{n+1} + 2u_m^{n+1} - u_{m+1}^{n+1}) = u_m^n + \tau f_m^{n+1}.$$

Ясно, что $\|u^{n+1}\| = |u_{m_0}^{n+1}|$, m_0 — первый встретившийся номер, на котором достиглась норма.

Если $m_0 = 0$ или M. Тогда выполнено

$$\|u^{n+1}\| \leqslant \max\left(\|\mu_1\|, \|\mu_2\|, \|u^n\| + \tau\|f_h\|\right)$$

И в этом случае мы всё доказали.

Если норма достигается внутри. Тогда внутри скобки по модулю $u_{m_0}^{n+1}$ самая большая. И знак скобки определяетс знаком этого эдемента. Пусть у нас $m = m_0$.

$$\|u^{n+1}\| = |u_m^{n+1}| \leqslant \left|u_m^{n+1} + \frac{\tau}{h^2}(-u_{m-1}^{n+1} + 2u_m^{n+1} - u_{m+1}^{n+1})\right| \leqslant \|u^n\| + \tau\|f_h\|.$$

Дальше повторяем тот кусок доказательства. И устойчивость доказана.

20.2 Скорость сходимости

Скорость сходимости будет такая же, как порядок аппроксимации. В явной схеме мы будем медленно шагать, вынуждены. Но в неявной схеме, чтобы была сходимость с тем же порядком, что и аппроксимация, нам необходимо и тут тоже $\tau \sim h^2$. Соответственно будем медленно считать.

Надо сооружать новую схему. Прежде чем это делать, соорудим оператор второго дифференцирования.

20.3 Оператор второго дифференцирования

Рассмотрим пространство сеточных (на отрезке $Mh = X$) функций $v = (0, v_1, v_2, \ldots, v_{M_1}, 0)$ таки, что на границе нули. На этом пространстве определим оператор численного дифференцирования

$$\Lambda v_m := \Lambda v\big|_m = \frac{v_{m-1} - 2v_m + 2_{m+1}}{h^2}, \quad m = 1, \ldots, M-1$$

Будем писать $\Lambda v = (0, \Lambda v_1, \ldots, \Lambda v_{M-1}, 0)$.

Рассматриваем собственные значения оператора (будем рассматривать оператор с обратным знаком) и собственные функции v^k: $(-\Lambda)v^k = \lambda_k v^k$. Вообще говоря, индекс k не может пробегать бесконечно много значений. Мы даже матрицу этого оператора выписывали. Мне удобно работать не на языке матриц, а на языке разностных уравнений.

$$\frac{-v_{m-1} + 2v_m - v_{m+1}}{h^2} = \lambda v_m, v_0 = v_M = 0, \quad h = \frac{X}{M}.$$

Надо найти λ такие, что задача имеет ненулевые решения.

$$v_{m+1} - (2 - \lambda h^2)v_m + v_{m-1} = 0, \quad v_0 = v_M = 0.$$

Будем искать решение виде $v_m = q^n$. Подставляем $q^2 - (2 - \lambda h^2)q + 1 = 0$.

Первый случай $D > 0$. Тогда у нас два неравных вещественных корня $q_1 \neq q_2 \in \mathbb{R}$. Тогда $v_m = c_1 q_1^m + C_2 q_2^m$. Начинаем подставлять краевые условия $C_1 + C_2 = 0$, $C_1 q_1^M + C_2 q_2^M = 0$. Отсюда получаем только нулевое решение.

Мы не оставляем надежду. Пусть $D = 0$. Тогда $q_1 = q_2 = q \in \mathbb{R}$. Решение должно выглядеть следующим образом $v_m = c_1 q^m + c_2 m q^m$. Опять подставляем всё, что можно: $C_1 = 0$. $C_2 M q^M = 0$. Но q у нас точно не ноль, потому что по теореме Виета произведение корней единичка. И опять нулевое.

Придётся рассматривать $D < 0$. Отсюда можно уже получить некую оценку на λ. Наше $q_{1,2} = \cos\varphi \pm i\sin\varphi$ (ведь произведение единичка). Должны были вы разбирать, что в этом случае решение $v_m = c_1 \cos m\varphi + C_2 \sin m\varphi$. Подставляем $m = 0$, тут же гибнет синус и $C_1 = 0$. Теперь подставляем дрожащей рукой $m = M$, получаем $C_2 \sin M\varphi = 0$, причём C_2 равным нулю нас никак не устроит, значит, нулём должен быть синус $\sin M\varphi = 0$. Отсюда $\varphi = \frac{\pi k}{M}$.

Можно выписывать дискриминант, а можно попроще. А именно сумма корней в наших обозначениях $2\cos\varphi$, а с другой по теореме Виета $(2 - \lambda h^2)$. Значит

$$\frac{2 - \lambda h^2}{2} = \cos\frac{\pi k}{M}.$$

Можно λ заиндексировать

$$\lambda_k = \frac{2}{h^2}\left(1 - \cos\frac{\pi k}{M}\right) = \frac{4}{h^2}\sin^2\frac{\pi k}{2M}.$$

На k причём пока ограничений нет. Но задача была об операторе на конечномерном пространстве. Значит, либо какие-то k не годятся, либо при некоторых k значения повторяются.

Пусть $k = 0$. Тогда дискриминант равен нулю, и мы этот случае забраковали. Значит, $k \neq 0$.

Дальше $k = 1, \dots, k = M - 1$ нам подходят. А уже λ_M опять даёт нулевой дискриминант. Дальше идёт некое отражение $\lambda_{M+1} = \lambda_{M-1}$, $\lambda_{M+2} = \lambda_{M-2}$ и так далее.

Ладно, а что за собственные функции? Это решения v^k: $v^k_m = \sin\frac{\pi k m}{M}$. Здесь $m = 1, \dots, M-1$. Кстати автоматом получаем, что и при $m = 0, M$ эти функции в наше пространство годятся.

20.4 Возвращаемся к задаче теплопроводности

Предлагается построить вот такую разностную схему

$$\frac{u^{n+1}_m - u^n_m}{\tau} = \sigma\Lambda u^{n+1}_m + (1-\sigma)\Lambda u^n_m + f^n_m.$$

Здесь $\sigma \in [0, 1]$. При $\sigma = 0$ получается явная схема, которую мы рассматривали, при $\sigma = 1$ неявная.

Сейчас мы будем играться с аппроксимацией. Мы хотим получить аппроксимацию по τ. Только аппроксимацию относительно точки $(m, n + 0.5)$. Есть надежда, что при $\sigma = 0.5$ как-то что-то сбалансируется.

Сейчас надо будет вспоминать, что мы выписывали в первом семестре про главные члены погрешности формул численного дифференцирования. У нас будет $(x, t) = (mh, n\tau)$. Будем выписывать аппрроксимацию на решении (на решении менее громоздко)

$$\frac{u(x, t+\tau) - u(x,t)}{\tau} = \left.\frac{\partial u}{\partial x}\right|_{(x,t)} + \frac{\tau^2}{24}\left.\frac{\partial^3 u}{\partial t^3}\right|_{(x,t)} + \dots$$

Теперь посмотрим на второе дифференцирование

$$\Lambda u\Big|_{\left(x, t+\frac{\tau}{2}\pm\frac{\tau}{2}\right)} = \Lambda u\Big|_{(x,t+\frac{\tau}{2})} \pm \frac{\tau}{2}\Lambda\left.\frac{\partial u}{\partial t}\right|_{(x,t+\tau/2)} + O(\tau^2) =$$
$$= \left.\frac{\partial^2 u}{\partial x^2}\right|_{(x,t+\tau/2)} + \frac{h^2}{12}\left.\frac{\partial^4 u}{\partial x^4}\right|_{(x,t+\tau/2)} \pm \frac{\tau}{2}\Lambda\left.\frac{\partial u}{\partial t}\right|_{(x,t+\tau/2)} + O(\tau^2, h^4).$$

Мы хотим оценить такую разность

$$\big(L_h[u] - f_h\big)\Big|_{(m,n+0.5)} = \Big(f\big(mh, (n+0.5)\tau\big) - f^n_m\Big) + \tau(\sigma - 0.5)\Lambda\left.\frac{\partial u}{\partial t}\right|_{(x,t+\tau/2)} + U(\tau^2, h^2).$$

Мы на чём остановились. Рассматривали уравнение теплопроводности

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + f.$$

В прямоугольной области с соответствующими краевыми условиями.

$$\frac{u^{n+1}_m - u^n_m}{\tau} = \sigma\Lambda u^{n+1}_m + (1-\sigma)\Lambda u^n_m + f^n_m, \quad \sigma \in [0, 1].$$

Если рассматриваем аппроксимацию в точке $(m, n + 0.5)$, И при этом берём $f^n_m = f\big(mh, (n+0.5)\tau\big)$ и $\sigma = 0.5$. В этом случае $O(\tau^2 + h^2)$.

Возникает проблема. Устойчивость и аппроксимацию будут записаны в разных нормах, то есть по разным точкам. Со сходимостью видимо тоже будет какая-то беда.

У нас будет упрощённая ситуация. Возьмём упрощённую задачу. Мы возьмём правую часть нулём. И нашу область вместе со своей сеточкой. Будем говорить, что у нас $u^n_0 = 0 = u^n_M = 0$, $Mh = X$. Единственное, что у нас будет, это начальное условие $u^0_m = \varphi_m$.

У нас было понятие нормы на слое. Но у нас был максимум, а теперь возьмём

$$\|u^n\| = \sqrt{\sum_{m=1}^{M-1}(u^n_m)^2 h}$$

Это аналог нормы в L_2. Эта норма не имеет отношение к той норме, которая была в аппроксимации.
Определение 20.1. *Устойчивость в этой норме*

$$\max_{0\leqslant n\leqslant N}\|u^n\|\leqslant c\|u^0\|$$

Давайте исследовать. Норма не совсем обычная. Те приёмы, которые у нас были, здесь не пройдут. Но вот этот оператор Λ, который разностная вторая производная, мы его в прошлый раз изучили. Мы выписали его собственные значения и собственные функции.

У нас $u^n=(0,u_1^n,\dots,u_{M-1}^n)$. Пусть E — единичный оператор

$$(E-\sigma\tau\Lambda)u^{n+1}=\big(E+(1-\sigma)\tau\Lambda\big)u^n. \tag{5}$$

Собственные значения оператора Λ назовём $-\alpha_k$, где $\alpha_k=\frac{4}{h^2}\sin^2\frac{\pi kh}{2}$. А собственные вектора $v^{(k)}$: $c_n^{(k)}=\sin\frac{\pi knh}{X}$.

Давайте решение будем раскладывать по этим собственным векторам. В нашем равенстве (5) собственные векторы правой и левой часть такие же, как и у оператора Λ, а собственные значения чуть другие. Можем написать

$$u^{n+1}=\underbrace{(E-\sigma\tau\Lambda)^{-1}\big(1+(1-\sigma)\tau\Lambda\big)}_{S}u^n.$$

Буквой S обозначили оператор перехода к новому слою. Собственные значения его мы сейчас найдём. Раскладываем начальное условие по собственным векторам

$$u_m^0=\sum_{k=1}^{M-1}c_k\sin\frac{\pi kmh}{X},\quad m=1,\dots,M-1.$$

На первом слое

$$u_m^1=Su_m^0=\sum_{i=1}^{M-1}\lambda_kc_k\sin\frac{\pi kmh}{X}.$$

Отсюда собственные значения

$$\lambda_k=\frac{1-(1-\sigma)\tau\alpha_k}{1+\sigma\tau\alpha_k}.$$

Что у нас будет на втором слое, всё будет по аналогии

$$u_m^n=\sum_{k=1}^{M-1}\lambda_k^nc_k\sin().$$

Ага, значит, нам достаточно, чтобы $|\lambda_k|<1$. Значит, $\alpha_k\in\big(0,\frac{4}{h^2}\big)$. Отсюда

$$-1\leqslant\frac{1-(1-\sigma)\tau\alpha_k}{1+\sigma\tau\alpha_k}.$$

Перепишем

$$-1-\sigma\tau\alpha_k\leqslant1-(1-\sigma)\tau\alpha_k\leqslant1+\sigma\tau\alpha_k.$$

Правое равенство очевидно. Осталось только левое

$$\tau(1-2\sigma)\alpha_k\leqslant2.$$

Если $\sigma=0.5$, то всё заведомо выполнено. Программу минимум сделали. Но мы сделаем подробнее

- Случай $\sigma\in[0.5,1]$. Тогда устойчивость для всех h,τ.

- Если $\sigma\in[0,0.5)$. Тогда нужно

$$\tau\leqslant\frac{2}{(1-2\sigma)\max\alpha_k}.$$

  Потребуем, чтобы $\tau<\frac{h^2}{2(1-2\sigma)}$.

20.5 Сходимость

Сходимость будет по совсем хитрой норме. Я этот факт доказывать не буду. Доказательство требует совсем другой техники, это было бы уже слишком. Норма тут такая

$$\|v\|_1 = \sqrt{\sum_{m=0}^{M-1} h \left(\frac{v_{m+1} - v_m}{h}\right)^2}.$$

Здесь $v = (0, v_1, \ldots, v_{M-1}, 0)$.

$$\|u_h\|_1 = \max_n \|u^n\|_1.$$

В такой норме сходимость будет с порядком $O(\tau^2 + h^2)$.

21 Стационарные задачи

Времени никакого нет, все переменные равноправны. Основная задача, которую мы будем рассматривать. А может на ней-то мы и остановимся.

Пусть у нас будет прямоугольник в плоскости (x_1, x_2) с углом в нуле, ширины l_1, высоты l_2. Уравнение

$$-\Delta u = f, \quad x = (x_1, x_2) \in \Omega \quad u\big|_{\partial\Omega} = \alpha.$$

Пусть $h_1 N1 = l_1$, $h_2 N_2 = l_2$.

Рассмотрим операторы

$$\Lambda_1 u_{m,n} = \frac{u_{m-1,n} - 2u_{m,n} + u_{m+1,n}}{h_1^2}, \quad \Lambda_2 u_{m,n} = \frac{u_{m,n-1} - 2u_{m,n} + u_{m,n+1}}{h_2^2}.$$

И рассмотрим схему

$$-(\Lambda_1 + \Lambda_2) u_m^n = f_m^n.$$

При этом $u_{m,0} = \alpha_{m,0}$, $u_{m,N2} = \alpha_{m,N_2}$. И далее

Схема такова, что углы нигде не используются. Для эллиптических задач извесная проблема таких особенностей.

Что требовала у нас устойчивость: она требовала существования и единственности решения. Ну ладно. Как нам хранить в памяти точки? Нужно их как-то перенумеровать, чтобы был линейный порядок.

Давайте считать, что у нас $l_1 = l_2$, $h_1 = h_2 = h$, $N_1 = N_2 = N$. Как выглядит наша схема в этом случае. Очень симпатично выглядит

$$\frac{-u_{m-1,n} - u_{m+1,n} + 4u_{m,n} - u_{m,n-1} - u_{m,n+1}}{h^2} = f_{m,n}.$$

Давайте нумеровать слева-направо и снизу вверх. Можно и в других направлениях, но лучше не получится. Наша нумерация окажется оптимальной. Первый слой нумеруем $1, 2, \ldots, N-1$, далее $N, N+1, \ldots, 2N-2$. Я хочу записать нашу схему в виде

$$A\boldsymbol{u} = \boldsymbol{F}.$$

Как выглядит наша матрица

$$A = \frac{1}{h^2}\begin{pmatrix} 4 & -1 & 0 & \ldots & 0 & -1 & \\ -1 & 4 & -1 & 0 & \ldots & 0 & -1 \end{pmatrix} < ++ >$$

Получится ленточная матрица размерности $A = \big((N-1)^2 \times (N-1)^2\big)$. Мы покажем, что эта матрица положительно определена. Мы запустим метод Гаусса сложностью $O(N^4)$. А затем если $N = 2^p$ сделаем метод сложности $O(N^2 \log N)$.