Numerical Analysis

gouziwu

June 26, 2019

Contents

1 Chap1 Mathematical Preliminaries

1.1 1.2 Roundoff Errors and Computer Arithmetic

Truncation Error : the error involved in using a truncated, or finite, summation to approximate the sum of an infinite series

Roundoff Error: the error produced when performing real number calculations. It occurs because the arithmetic performed in a machine involves numbers with only a finite number of digits.

Suppose $y = 0.d_1 d_2 \dots d_k d_{k+1} d_{k+2} \dots \times 10^n$, then

$$fl(y) = \begin{cases} 0.d_1 d_2 \dots d_k \times 10^n & \text{chopping} \\ chop(y + 5 \times 10^{n-(k+1)}) = 0.\delta_1 \delta_2 \dots \delta_k \times 10^n & \text{Rounding} \end{cases}$$

Definition 1.1. *If $p*$ is an approximation to p, the absolute error is $|p - p*|$, and the relative error is $\frac{|p-p*|}{|p|}$, provided that $p \neq 0$*

Definition 1.2. *The number $p*$ is said to approximate p to t significant digits if t is the largest nonnegative integer for which $\frac{|p-p*|}{|p|} < 5 \times 10^{-t}$*

chopping $|\frac{y-fl(y)}{y}| = |\frac{0.d_1d_2...d_kd_{k+1}\cdots \times 10^n - 0.d_1d_2...d_k \times 10^n}{0.d_1d_2...d_kd_{k+1} \times 10^n}| = |\frac{0.d_{k+1}\cdots}{0.d_1d_2\ldots}| \times 10^{-k} \leqslant \frac{1}{0.1} \times 10^{-k} = 10^{-k+1}$

rounding $|\frac{y-fl(y)}{y}| \leqslant \frac{0.5}{0.1} \times 10^{-k} = 0.5 \times 10^{-k+1}$

{ Subtraction of nearly equal numbers will cause a cancellation of significant digits.}

{Dividing by a number with small magnitude (or, equivalently, multiplying by a number with large magnitude) will cause an enlargement of the error.}

Finite digit arithmetic

- $x \oplus y = fl(fl(x) + fl(y))$
- $x \otimes y = fl(fl(x) \times fl(y))$
- $x \ominus y = fl(fl(x) - fl(y))$
- $x \oplus y = fl(fl(x) \div fl(y))$

1.2 1.3 ALgorithms and Convergence

An algorithm that satisfies that small changes in the initial data produce correspondingly small changes in the final results is called **stable**; otherwise it is **unstable**. An algorithm is called **conditionally stable** if it is stable only for certain choices of initial data.

Suppose that E > 0 denotes an initial error and En represents the magnitude of an error after n subsequent operations. If $E_n \approx CnE_0$, where C is a constant independent of n, then the growth of error is said to be **linear**. If $E_n \approx C^n E_0$, for some C > 1, then the growth of error is called **exponential**

Suppose $\{\beta_n\}_{n=1}^{\infty}$, $\lim\limits_{n\to\infty} \beta_n = 0, \{\alpha_n\}_{n=1}^{\infty}$, $\lim\limits_{n\to\infty} \alpha_n = \alpha$. If a positive constant K exists with $|\alpha_n - \alpha| \leqslant K|\beta_n|$ for large n, then $\{\alpha_n\}_{n=1}^{\infty}$ converges to with **rate, or order, of convergence** $O(\beta_n)$

Suppose $\lim\limits_{h\to 0} G(h) = 0, \lim\limits_{h\to 0} F(h) = L$ and $|F(h) - L| \leqslant K|G(h)|$ for sufficiently small h, then we write $F(h) = L + O(G(h))$

2 Chap2 Solutions of equations in one variable

2.1 2.1 Bisection method

Theorem 2.1. *Intermediate Value Theorem If* $f \in C[a,b]$, $K \in (f(a), f(b))$, *then there exists a number* $p \in (a,b)$ *for which* $f(p) = K$

Theorem 2.2. *Suppose that* $f \in C[a,b]$ *and* $f(a) \cdot f(b) < 0$. *The bisection method generates a sequence* $\{p_n\}, n = 0, 1, \ldots$ *approximating a zero* p *of* f *with*

$$|p_n - p| \leqslant \frac{b-a}{2^n}, \quad \textit{when } n \geqslant 1$$

2.2 2.2 Fixed-Point Iteration

$f(x) = 0 \xleftrightarrow{\text{equivalent}} x = f(x) + x = g(x)$

Theorem 2.3. *Fixed-Point Theorem Let* $g \in C[a,b]$ *be s.t.* $g(x) \in [a,b]$ *for all* $x \in [a,b]$. *Suppose that* g' *exists on* (a,b) *and that a constant* $0 < k < 1$ *exists with* $|g'(x)| \leqslant k$ *for all* $x \in (a,b)$ *(hence* g' *can't converge to 1). Then for any number* p_0 *in* $[a,b]$, *the sequence defined by* $p_n = g(p_{n-1}), n \geqslant 1$ *converges to the unique point* p *in* $[a,b]$

Corollary 2.1. $|p_n - p| \leqslant \frac{1}{1-k}|p_{n+1} - p_n|$ *and* $|p_n - p| \leqslant \frac{k^n}{1-k}|p_1 - p_0|$

2.3 2.3 Newton's method

Linearize a nonlinear function using **Taylor's expansion**

Let $p_0 \in [a,b]$ be an approximation to p s.t. $f'(p_0) \neq 0$, hence $f(x) = f(p_0) + f'(p_0)(x - p_0) + \frac{f''(\xi_x)}{2!}(x - p_0)^2$, then $0 = f(p) \approx f(p_0) + f'(p_0)(p - p0) \to p \approx p_0 - \frac{f(p_0)}{f'(p_0)}$ $p_n = p_{n-1} - \frac{f(p_{n-1})}{f'(p_{n-1})}, \quad \text{for} n \geqslant 1$

Theorem 2.4. *Let* $f \in C^2[a,b]$. *If* $p \in [a,b]$ *is s.t.* $f(p) = 0, f'(p) \neq 0$, *then there exists a* $\delta > 0$ *s.t. Newton's method generates a sequence* $\{p_n\}, n \in \mathbb{N}\backslash\{0\}$ *converging to* p *for any initial approximation* $p \in [p - \delta, p + \delta]$.

2.4 2.4 Error analysis for iterative methods

Definition 2.1. *Suppose* $\{p_n\}(n = 0, 1, \ldots)$ *is a sequence that converges to* p *with* $p_n \neq p$ *for all* n. *If positive constants* α *and* λ *exist with*

$$\lim_{n \to \infty} \frac{|p_{n+1} - p|}{|p_n - p|^\alpha} = \lambda$$

then $\{p_n\}(n = 0, 1, \dots)$ *converges to* p *of order* α, *with asymptotic error constant* λ

Theorem 2.5. *Let* p *be a fixed point of* $g(x)$. *If there exists some constant* $\alpha \geqslant 2$ *s.t.* $g \in C^{\alpha}[p-\delta, p+\delta]$, $g'(p) = \cdots = g^{\alpha-1}(p) = 0$ *and* $g^{\alpha}(p) \neq 0$. *Then the iterations with* $p_n = g(p_{n-1})$, $n \geqslant 1$ *is of order* α

$$p_{n+1} = g(p_n) = g(p) + g'(p)(p_n - p) + \cdots + \frac{g^{\alpha}(\xi_n)}{\alpha!}(p_n - p)^{\alpha}$$

Theorem 2.6. *Let* $g \in C[a, b]$ *be s.t.* $g(x) \in [a, b]$ *for all* $x \in [a, b]$. *Suppose in addition that* g' *is continuous on* (a, b) *and a positive constant* $k < 1$ *exists with*

$$|g'(x)| \leqslant k, \quad \text{for all } x \in (a, b)$$

If $g'(p) \neq 0$, *then for any number* $p_0 \neq p$ *in* $[a, b]$, *the sequence*

$$p_n = g(p_{n-1}), \quad \text{for } n \geqslant 1$$

converges only linearly to the unique fixed point in $[a, b]$

Proof.

$$\begin{aligned}\lim_{n\to\infty} \frac{|p_{n+1} - p|}{|p_n - p|} &= \lim_{n\to\infty} \frac{|g(p_n) - p|}{|p_n - p|} \\ &= \lim_{n\to\infty} \frac{|g'(\xi)(p_n - p)|}{|p_n - p|} \\ &= |g'(p)|\end{aligned}$$

□

Theorem 2.7. *Let* p *be a solution of the equation* $x = g(x)$. *Suppose that* $g'(p) = 0$ *and g" is continuous with* $|g''(x)| < M$ *on an open interval* I *containing* p. *Then there exists a* $\delta > 0$ *s.t. for* $p_0 \in [p-\delta, p+\delta]$, *the sequence defined by* $p_n = g(p_{n-1})$, *when* $n \geqslant 1$ *converges at least quadratically to* p. *Moreover, for sufficiently large values of* n,

$$|p_{n+1} - p| < \frac{M}{2}|p_n - p|^2$$

Proof. Choose $k \in (0,1), \delta > 0$ s.t. $[p-\delta, p+\delta] \subseteq I$ and $|g'(x)| < k$ and g'' is continuous.

$$g(x) = g(p) + g'(p)(x-p) + \frac{g''(\xi)}{2}(x-p)^2$$

Hence $g(x) = p + \frac{g''(\xi)}{2}(x-p)^2$. $p_{n+1} = g(p_n) = p + \frac{g''(\xi_n)}{2}(p_n - p)^2$. Thus $p_{n+1} - p = \frac{g''(\xi_n)}{2}(p_n - p)^2$. We get

$$\lim_{n\to\infty} \frac{|p_{n+1} - p|}{|p_n - p|^2} = \frac{g''(p)}{2}$$

□

Definition 2.2. *A solution p of $f(x) = 0$ is a zero of multiplicity m of f if for $x \neq p$, $f(x) = (x-p)^m q(x)$ where $\lim_{x\to p} q(x) \neq 0$*

Theorem 2.8. *The function $f \in C^m[a,b]$ has a zero of multiplicity m at p in (a,b) if and only if*

$$0 = f(p) = f'(p) = \cdots = f^{(m-1)}(p), \quad \textit{but } f^{(m)}(p) \neq 0$$

To handle the problem of multiple roots of a function f is to define $\mu(x) = \frac{f(x)}{f'(x)}$.

If p is a zero of f of multiplicity m with $f(x) = (x-p)^m q(x)$, then

$$\begin{aligned}\mu(x) &= \frac{(x-p)^m q(x)}{m(x-p)^{m-1}q(x) + (x-p)^m q'(x)} \\ &= (x-p)\frac{q(x)}{mq(x) + (x-p)q'(x)}\end{aligned}$$

And $q(x) \neq 0$.

Now Newton's method:

$$\begin{aligned}g(x) &= x - \frac{\mu(x)}{\mu'(x)} \\ &= x - \frac{f(x)/f'(x)}{(f'(x)^2 - f(x)f''(x))/f'(x)^2} \\ &= x - \frac{f(x)f'(x)}{f'(x)^2 - f(x)f''(x)}\end{aligned}$$

3 Chap3 Interpolation and polynomial approximation

3.1 3.1 Interpolation and the Lagrange polynomial

$P_n(x) = \sum_{i=0}^{n} L_{n,i}(x)y_i$. Find $L_{n,i}(x)$ for $i = 0, \ldots, n$ s.t. $L_{n,j}(x_j) = \delta_{ij}$. δ_{ij} Kronecker delta. Each $L_{n,i}$ has n roots $x_0, \ldots, \hat{x_i}, \ldots, x_n$. $L_{n,j}(x) = C_i(x - x_0)\ldots(\widehat{x - x_i})\ldots(x - x_n) = C_i \prod_{\substack{j\neq i\\ j=0}}^{n}(x - x_j)$. $L_{n,j}(x_i) = 1 \rightarrow C_i = \prod_{j\neq i} \frac{1}{x_i - x_j}$. Hence $L_{n,i}(x) = \prod_{\substack{j\neq i\\ j=0}}^{n} \frac{x - x_j}{x_i - x_j}$

Theorem 3.1. *If $x_0, x_1, \ldots, x_n$ are n+1 distinct numbers and f is a function whose values are given at these numbers, then the n-th Lagrange interpolating polynomial is unique*

Analyze the remainder. Suppose $a \leqslant x_0 < x_1 < \cdots < x_n \leqslant b$ and $f \in C^{n+1}[a, b]$. Consider $R_n(x) = f(x) - P_n(x)$. $R_n(x)$ has at least n+1 roots => $R_n(x) = K(x)\prod_{i=0}^{n}(x - x_i)$. For any $x \neq x_i$. Define $g(t) = R_n(t) - K(x)\prod_{i=0}^{n}(t - x_i)$. $g(x)$ has n+2 distinct roots $x_0 \ldots x_n x$. Hence $g^{(n+1)}(\xi_x) = 0, \xi_x \in (a, b)$. $f^{(n+1)}(\xi_x) - Pn^{(n+1)}(\xi_x) - K(x)(n + 1)! = R_n^{(n+1)}(\xi_x) - K(x)(n + 1)!$. Thus $R_n(x) = \frac{f^{(n+1)}(\xi_x)}{(n+1)!}\prod_{i=0}^{n}(x - x_i)$.

Definition 3.1. *Let f be a function defined at $x_0, \ldots, x_n$ and suppose $m_1, \ldots, m_k$ are k distinct integers with $0 \leqslant m_i \leqslant n$ for each i. The Lagrange polynomial that agrees with $f(x)$ at the k points $x_{m_1}, \ldots, x_{m_k}$ denoted by $P_{m_1,;m_k}(x)$*

Theorem 3.2. *Let f be defined at $x_0, \ldots, x_k$ and let x_i and x_j be two distinct numbers in this set. Then*

$$P(x) = \frac{(x - x_j)P_{0,1,\ldots,j-1,j+1,\ldots,k(x)} - (x - x_i)P_{0,\ldots,i-1,i+1,\ldots,k(x)}}{x_i - x_j}$$

describes the k-th Lagrange polynomial that interpolates f at the k+1 points $x_0, \ldots, x_k$

Neville's Method
$$\begin{array}{lllll} x_0 & P_0 & & & \\ x_1 & P_1 & P_{0,1} & & \\ x_2 & P_2 & P_{1,2} & P_{0,1,2} & \\ x_3 & P_3 & P_{2,3} & P_{1,2,3} & P_{0,1,2,3} \end{array}$$

3.2 3.2 Divied differences

$f[x_i, x_j] = \frac{f(x_i)-f(x_j)}{x_i-x_j}(i \neq j, x_i \neq x_j)$. $f[x_i, x_j, x_k] = \frac{f[x_i,x_j]-f[x_j,x_k]}{x_i-x_k}$.

3.3 Additional Newton Interpolation

3.3.1 Simple idea

Given $x_0, \dots, x_n$

1. Fitting x_0 first: $f(x) \approx f_0, f_0 = f(x_0)$
2. Add one more point x_1, $f_1 = f(x_1)$

$$f(x) \approx f_0 + \alpha_1(x - x_0), \alpha_1 = \frac{f_1 - f_0}{x_1 - x_0}$$

3. More points $f(x) \approx f_0 + \alpha_1(x - x_0) + \alpha_2(x - x_0)(x - x_1)$

The pattern and coefficients. $f(x) = \sum_{i=0}^{n} \alpha_i \prod_{j=0}^{j<i}(x-x_j) = \sum_{i=0}^{n} \alpha_i N^{(i)}(x)$

$$\begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ f_n \end{pmatrix} = \begin{pmatrix} N^{(0)}(x_0) & N^{(1)}(x_0) & \dots & N^{(n)}(x_0) \\ N^{(0)}(x_1) & N^{(1)}(x_1) & \dots & N^{(n)}(x_1) \\ \vdots & \vdots & \ddots & \vdots \\ N^{(0)}(x_n) & N^{(1)}(x_n) & \dots & N^{(n)}(x_n) \end{pmatrix} \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_n \end{pmatrix}$$

$N^{(i)}(x_k) = \begin{cases} 0 & k < i \\ \prod_{j=0}^{j<i}(x_k - x_j) & k \geqslant i \end{cases}$ with $N^{(0)}(x) = 1$. Newton interpolation matrix is lower triangular. Lagrange matrix is identity.

3.3.2 Basis transformation

$$\begin{pmatrix} 1 \\ (x - x_0) \\ (x - x_0)(x - x_1) \\ \vdots \end{pmatrix} = (?) \begin{pmatrix} 1 \\ x \\ x^2 \\ \vdots \end{pmatrix}$$

Hence $(\Phi_B)^T = (T_A^B)^T(\Phi_A)^T$. $\Phi_B = \Phi_A T_A^B$

$$\begin{aligned}(\Phi_A)(\alpha_A) = (f) &= (\Phi_B)(\alpha_B)\\ &= (\Phi_A)(T_A^B)(\alpha_B)\\ &\Rightarrow\\ (\alpha_A) &= (T_A^B)(\alpha_B)\\ (\alpha_B) &= (T_A^B)^{-1}(\alpha_A)\\ &= (T_B^A)(\alpha_A)\end{aligned}$$

3.4 3.3 Hermite interpolation

Find the **osculating polynomial** $P(x)$ s.t. $P(x_i) = f(x_i), P'(x_i) = f'(x_i), \dots, P^{(m_i)}(x_i) = f^{(m_i)}(x_i)$ for all $i = 0, 1, \dots, n$.

Just the Taylor polynomial $P(x) = f(x_0) + f'(x_0)(x - x_0) + \cdots + \frac{f^{(m_0)}(x_0)}{m_0!}(x - x_0)^{m_0}$ with remainder $R(x) = f(x) - \varphi(x) = \frac{f^{(m_0+1)}(\xi)}{(m_0+1)!}(x - x_0)^{(m_0+1)}$

$m_i = 1$ gives **Hermite polynomial**

Example 3.1. *Suppose $x_0 \neq x_1 \neq x_2$. Given $f(x_0), f(x_1), f(x_2), f'(x_1)$ find the polynomial $P(x)$ s.t. $P(x_i) = f(x_i), P'(x_1) = f'(x_1)$ and analyze the errors.*

Proof. $P_3(x) = \sum_{i=0}^{2} f(x_i)h_i(x) + f'(x_1)\hat{h}_1(x)$ where $h_i(x_j) = \delta_{ij}, h_i'(x_i) = 0, \hat{h}_i(x_i) = 0, \hat{h}_i'(x_1) = 1$.

- $h_0(x)$. Has roots x_1, x_2 and x_1 is a multiple root. $h_0(x) = C_0(x - x_1)^2(x - x_2)$ and $h_0(x_0) = 1 \Longrightarrow C_0$
- $\hat{h}_1(x)$ has root $x_0, x_1, x_2 \Longrightarrow \hat{h}_1(x) = C_1(x - x_0)(x - x_1)(x - x_2)$

□

In general, given $x_0, \dots, x_n; y_0, \dots, y_n$ and $y_0', \dots, y_n'$. The Hermite polynomial $H_{2n+1}(x)$ satisfies $H_{2n+1}(x_i) = y_i$ and $H_{2n+1}'(x_i) = y_i'$

Solution. $H_{2n+1}(x) = \sum_{i=0}^{n} y_i h_i(x) + \sum_{i=0}^{n} y_i' \hat{h}_i(x)$

3.5 3.4 Cubic spline interpolation

Piecewise linear interpolation. Approximate $f(x)$ by linear polynomials on each subinterval $[x_i, x_{i+1}]$.

$f \approx P_1(x) = \frac{x-x_{i+1}}{x_i-x_{i+1}} y_i + \frac{x-x_i}{x_{i+1}-x_i} y_{i+1} \quad \text{for } x \in [x_i, x_{i+1}]$

Let $h = \max|x_{i+1} - x_i|$. Then $P_1^h(x) \xrightarrow{uniform} f(x)$ as $h \to 0$ However, this is no longer smooth.

Hermite piecewise polynomials. Given $x_0, \dots, x_n; y_0, \dots, y_n, y_0', \dots, y_n'$, construct the Hermite polynomial of degree 3 with y and y' on the two endpoints of $[x_i, x_{i+1}]$

Cubic Spline.

Definition 3.2. *Given a function f define on $[a, b]$ and a set of nodes $a = x_0 < x_1 < \cdots < x_n = b$, cubic spline interpolant S for f is a function that satisfies the following conditions*

- *$S(x)$ is a cubic polynomial, denoted by $S_i(x)$ on the subinterval $[x_i, x_{i+1}]$ for each $i = 0, \dots, n-1$*
- *$S(x_i) = f(x_i)$ for each $i = 0, \dots, n$*
- *$S_{i+1}(x_{i+1}) = S_i(x_{i+1})$*
- *$S'_{i+1}(x_{i+1}) = S'_i(x_{i+1})$*
- *$S''_{i+1}(x_{i+1}) = S''_i(x_{i+1})$*

Method of Bending moment. Let $h_j = x_j - x_{j-1}$ and $S(x) = S_j(x)$ for $x \in [x_{j-1}, x_j]$. Then S''_j is a polynomial of degree 1, which can be determined by the values of f on 2 nodes .

Assume $S''_j(x_{j-1}) = M_{j-1}, S''_j(x_j) = M_j$. Then for all $x \in [x_{j-1}, x_j]$, $S''_j(x) = M_{j-1}\frac{x_j - x}{h_j} + M_j \frac{x - x_{j-1}}{h_j}$. Hence we get

$$S'_j(x) = -M_{j-1}\frac{(x_j - x)^2}{2h_j} + M_j\frac{(x - x_{j-1})^2}{2h_j} + A_j$$

$$S_j(x) = M_{j-1}\frac{(x_j - x)^3}{6h_j} + M_j\frac{(x - x_{j-1})^3}{6h_j} + A_j x + B_j$$

Solve this by $S_j(x_{j-1}) = y_{j-1}, S_j(x_j) = y_j$, we get

$$A_j = \frac{y_j - y_{j-1}}{h_j} - \frac{M_j - M_{j-1}}{6} h_j$$
$$A_j x + B_j = (y_{i-1} - \frac{M_{j-1}}{6} h_j^2) \frac{x_j - x}{h_j} + (y_j - \frac{M_j}{6} h_j^2) \frac{x - x_{j-1}}{h_j}$$

Now solve for M_j: Since S' is continuous at x_j

$$[x_{j-1}, x_j] : S'_j(x) = -M_{j-1} \frac{(x_j - x)^2}{2h_j} + M_j \frac{(x - x_{j-1})^2}{2h_j} + f[x_{j-1}, x_j] - \frac{M_j - M_{j-1}}{6} h_j$$
$$[x_j, x_{j+1}] : S'_{j+1}(x) = -M_j \frac{(x_{j+1} - x)^2}{2h_{j+1}} + M_{j+1} \frac{(x - x_j)^2}{2h_{j+1}} + f[x_j, x_{j+1}] - \frac{M_{j+1} - M_j}{6} h_{j+1}$$

From $S'_j(x_j) = S'_{j+1}(x_j)$, let $\lambda_j = \frac{h_{j+1}}{h_j + h_{j+1}}, \mu_j = 1 - \lambda_j, g_j = \frac{6}{h_j + h_{j+1}}(f[x_j, x_{j+1}] - f[x_{j-1}, x_j])$ we get

$$\mu_j M_{j-1} + 2M_j + \lambda_j M_{j+1} = g_j \quad \text{for } \ 1 \leqslant j \leqslant n-1$$

$$\begin{pmatrix} \mu_1 & 2 & \lambda_1 & & \\ & \ddots & \ddots & \ddots & \\ & & \mu_{n-1} & 2 & \lambda_{n-1} \end{pmatrix} \begin{pmatrix} M_0 \\ \vdots \\ \vdots \\ M_n \end{pmatrix} = \begin{pmatrix} g_1 \\ \vdots \\ g_{n-1} \end{pmatrix}$$

And $S'(a) = y'_0, S'(b) = y'_n$

If $S''(a) = y''_0 = M_0, S''(b) = y''_n = M_n$, then $\lambda_0 = 0, g_0 = 2y''_0, \mu_n = 0 g_n = 2y''_n$.

The case when $M_0 = M_n = 0$ is called a **free boundary**, the spline is called **natural spline**

4 chap4 numerical differentiation and integration

4.1 4.1 numerical differentiation

Target: Given x_0, approximate $f'(x_0)$

$$f'(x_0) = \lim_{h \to 0} \frac{f(x_0 + h) - f(x_0)}{h}$$

Approximate $f(x)$ by its lagrange polynomial with interpolating points x_0 and $x_0 + h$

$$
\begin{aligned}
f(x) &= \frac{f(x_0)(x-x_0-h)}{x_0-x_0-h} + \frac{f(x_0+h)(x-x_0)}{x_0+h-x_0} \\
&\quad + \frac{(x-x_0)(x-x_0-h)}{2} f''(\xi_x) \\
f'(x) &= \frac{f(x_0+h)-f(x_0)}{h} + \frac{2(x-x_0)-h}{2} f''(\xi_x) \\
&\quad + \frac{(x-x_0)(x-x_0-h)}{2} \frac{d}{dx}[f''(\xi_x)] \\
f'(x_0) &= \frac{f(x_0+h)-f(x_0)}{h} - \frac{h}{2} f''(\xi)
\end{aligned}
$$

Approximate $f(x)$ by its Lagrange polynomial with interpolating points $\{x_0, x_1, \ldots, x_n\}$

$$
\begin{aligned}
f(x) &= \sum_{k=0}^{n} f(x_k)L_k(x) + \frac{(x-x_0)\ldots(x-x_n)}{(n+1)!} f^{(n+1)}(\xi_x) \\
f'(x_j) &= \sum_{k=0}^{n} f(x_k)L_k'(x_j) + \frac{f^{(n+1)}(\xi_j)}{(n+1)!} \prod_{\substack{k=0 \\ k\neq j}}^{n} (x_j - x_k)
\end{aligned}
$$

4.2 4.3 elements of numerical integration

Target: approximate $I = \int_a^b f(x)dx$

Integrate the **Lagrange interpolating polynomial** of $f(x)$ instead

Select a set of distinct nodes $a \leqslant x_0 < x_1 < \cdots < x_n \leqslant b$ from $[a, b]$.

The Lagrange polynomial is $P_n(x) = \sum_{k=0}^{n} f(x_k)L_k(x)$

$$
\int_a^b f(x)dx \approx \sum_{k=0}^{n} f(x_k) \overbrace{\int_a^b L_k(x)dx}^{A_k}
$$

Error

$$
\begin{aligned}
R[f] &= \int_a^b f(x)dx - \sum_{k=0}^{n} A_k f(x_k) \\
&= \int_a^b [f(x) - P_n(x)]dx = \int_a^b R_n(x)dx \\
&= \int_a^b \frac{f^{(n+1)}(\xi_x)}{(n+1)!} \prod_{i=0}^{n}(x - x_i)dx
\end{aligned}
$$

Definition 4.1. *The degree of accuracy, or precision of a quadrature formula is the largest positive integer n s.t. the formula is exact for x^k for each $k = 0, 1, \ldots, n$*

Example. Consider the linear interpolation on $[a, b]$, we have

$$
P_1(x) = \frac{x-b}{a-b} f(a) + \frac{x-a}{b-a} f(b)
$$

$A_1 = A_2 = \frac{b-a}{2}$, $\int_a^b f(x)dx \approx \frac{b-a}{2}[f(a) + f(b)]$. This is trapezoidal rule.

Consider x^k

$$
\begin{aligned}
1 : &\quad \int_a^b 1dx = b - a = \frac{b-a}{2}[1+1] \\
x : &\quad \int_a^b xdx = b - a = \frac{b-a}{2}[a+b] \\
x^2 : &\quad \int_a^b x^2dx = b - a \neq \frac{b-a}{2}[a^2+b^2]
\end{aligned}
$$

For equally spaced nodes: $x_i = a + ih, h = \frac{b-a}{n}, i = 0, 1, \ldots, n$

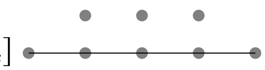
$$
\begin{aligned}
A_i &= \int_{x_0}^{x_n} \prod_{j \neq i} \frac{x - x_j}{x_i - x_j} dx \\
&= \int_0^n \prod_{i \neq j} \frac{(t-j)h}{(i-j)h} \times hdt \quad x = a + th \\
&= \frac{(b-a)(-1)^{n-i}}{n\ i!(n-i)!} \int_0^n \prod_{i \neq j}(t-j)dt
\end{aligned}
$$

$\frac{(-1)^{n-i}}{n\ i!(n-i)!} \int_0^n \prod_{i \neq j}(t-j)dt$ is the **Cotes coefficients**

4.3 4.4 composite numerical integration

Due to the oscillatory nature of high-degree polynomials, **piecewise** interpolation is applied to approximate $f(x)$. A piecewise approach that uses the low-order Newton-Cotes formulae

Composite Trapezoidal rule: $h = \frac{b-a}{n}, x_k = a + kh$.

Apply Trapezoidal Rule on each $[x_{k-1}, x_k]$ 

$$\int_{x_{k-1}}^{x_k} f(x)d(x) \approx \frac{x_k - x_{k-1}}{2}[f(x_{k-1}) - f(x_k)]$$

$$\int_a^b f(x)dx \approx \sum_{k=1}^{n} \frac{h}{2}[f(x_{k-1}) + f(x_k)] = \frac{h}{2}\left[f(a) + 2\sum_{k=1}^{n-1} f(x_k) + f(b)\right] = T_n$$

$$R[f] = \sum_{k=1}^{n}\left[-\frac{h^3}{12}f''(\xi_k)\right] = -\frac{h^2}{12}(b-a)\frac{\sum_{k=1}^{n} f''(\xi_k)}{n} = -\frac{h^2}{12}(b-a)f''(\xi), \xi \in (a,b)$$

Composite simpson's rule

$$\int_{x_k}^{x_{k+1}} f(x)dx \approx \frac{h}{6}\left[f(x_k) + 4f(x_{k+1/2}) + f(x_{k+1})\right]$$

In fact, it's just a mean value $(f(x_k) + 4f(x_{k+1/2}) + f(x_{k+1}))/6$

4.4 4.5 Romberg integration

$$\begin{aligned} R_n[f] &= -\frac{h^2}{12}(b-a)f''(\xi) \\ R_{2n}[f] &= -\frac{h^2/4}{12}(b-a)f''(\xi') \approx \frac{1}{4}R_n[f] \end{aligned}$$

Hence we have

$$\frac{I - T_{2n}}{I - T_n} \approx \frac{1}{4}$$

and $I \approx \frac{4}{3}T_{2n} - \frac{1}{3}T_n = S_n$.$\frac{4^2 S_{2n} - S_n}{4^2 - 1} = C_n$, $\frac{4^3 C_{2n} - S_n}{4^3 - 1} = R_n$, the **Romberg sequence**

4.5 4.2 Richardson's Extrapolation

generate high-accuracy results while using low-order formulae

For some $h \neq 0$, suppose we have $T_0(h)$ that approximates an unknown I, and

$$T_0(h) - I = \alpha_1 h + \alpha_2 h + \dots$$
$$T_0(h/2) - I = \alpha_1 (h/2) + \alpha_2 (h/2)^2 + \dots$$

Hence can improve accuracy by substituting

4.6 4.6 Adaptive quadrature methods

Predict the amount of functional variation and adapt the step size to the varing requirement

using the composite integration

- recursively halve the step size
- waste large number of computations
- only need to halve the interval with large error
- THIS is **adaptive**

A simple strategy to bound the total error by ϵ of

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

In an interval with length h, the error is smaller than $h\frac{\epsilon}{b-a}$

$$\epsilon(f, a, b) = \int_a^b f(x)dx - S(a, b) = \frac{h^5}{90} f^{(4)}(\xi)$$

4.7 4.7 Gaussian Quadrature

Construct formula

$$\int_a^b w(x)f(x)dx \approx \sum_{k=0}^{k} A_k f(x_k)$$

of precision degree 2n+1 with n+1 points

Theorem 4.1. *$x_0, \ldots, x_n$ are Gaussian points iff $W(x) = \prod_{k=0}^{n}(x - x_k)$ is orthogonal to all the polynomials of degree no greater than n*

Proof. 1. If $x_0, \ldots, x_n$ are Gaussian points, then the degree of precision of the formula $\int_a^b w(x)f(x) \approx \sum_{k=0}^{n} A_k f(x_k)$ is at least $2n+1$.

For any polynomial $P_m(x)$ with $m \leqslant n$, the degree of $P_m(x)W(x)$ is no greater than $2n+1$. Hence

$$\int_a^b w(x)P_m(x)W(x)dx = \sum_{k=0}^{n} A_k P_m(x_k)W(x_k) = 0Let$$

$\mathrm{P}_m(x) = W(x)q(x) + r(x)$. Then

$$\int_a^b w(x)P_m(x)dx = \int_a^b w(x)W(x)q(x)dx + \int_a^b w(x)r(x)dx = \sum_{k=0}^{n} A_k r(x_k)$$
$$= \sum_{k=0}^{n} A_k P_m(x_k)$$

Since $r(x)$'s degree is less then $n+1$ and can be approximate by $n+1$ points

□

Suppose $\{\varphi_0, \ldots, \varphi_n, \ldots\}$ are linearly independent and φ_{n+1} is orthogonal to any polynomial $P_m(x)$ with $m \leqslant n$. If we take $varphi_{n+1}$ to be $W(x)$, the the roots of φ are the Gaussian points

5 chap5 Initial-value problems for ordinary differential equations

5.1 5.1 the elementary theory of initial-value problems

$$\begin{cases} \frac{dy}{dt} = f(t, y) & t \in [a, b] \\ y(a) = \alpha \end{cases}$$

Compute the approximation of $y(t)$ at a set of mesh points $a = t_0 < t_1 < \cdots < t_n = b$

Definition 5.1. *A function $f(t,y)$ is said to satisfy a Lipschitz condition in the varaible y on a set $D \subset R^2$ if a constatn $L > 0$ exists with*

$$|f(t,y_1) - f(t,y_2)| \leqslant L|y_1 - y_2|$$

whenever $(t,y_1),(t,y_2) \in D$. The constant L is a Lipschitz condition

Theorem 5.1. *Suppose that $D = \{(t,y)|a \leqslant t \leqslant b, -\infty < y < \infty\}$ and that $f(t,y)$ is continuous on D. If f satisfies a Lipschitz condition on D in the variable y, then the IVP*

$$y'(t) = f(t,y), a \leqslant t \leqslant b, y(a) = \alpha$$

has a /textbfunique solution $y(t)$

Definition 5.2. *The initial-value problem*

$$y'(t) = f(t,y), a \leqslant t \leqslant b, y(a) = \alpha$$

is said to be a well-posed problems if:

1. *A unique solution $y(t)$ to the problem*
2. *For any $\epsilon > 0$, there exists a positive constant $k(\epsilon)$ s.t. whenever $|\epsilon_0|$ $<\epsilon, and \delta(t)$ is continuous with $|\delta(t)| < \epsilon$ on [a,b], a unique solution $z(t)$*

   $$z'(t) = f(t,z) + \delta(t), a \leqslant t \leqslant b, z(a) = \alpha + \epsilon_0$$

   exists with $|z(t) - y(t)| < k(\epsilon)\epsilon$, for all $a \leqslant t \leqslant b$

Theorem 5.2. *Suppose that $D = \{(t,y)|a \leqslant t \leqslant b, -\infty < y < \infty\}$ and that $f(t,y)$ is continuous on D. If f satisfies a Lipschitz condition on D in the variable y, then the IVP is well-posed*

5.2 5.2 Euler's Method

$$y'(t_0) \approx \frac{y(t_0 + h) - y(t_0)}{h}$$
$$y(t_1) \approx y(t_0) + hy'(t_0) = \alpha + hf(t_0, \alpha)$$

5.3 5.3 Higher Order Taylor Methods

Definition 5.3. *The difference method*

$$w_0 = \alpha \quad w_{i+1} = w_i + h\phi(t_i, w_i), \textit{for each } i = 0, 1, \ldots, n-1$$

has local truncation error

$$\tau_{i+1}(h) = \frac{y_{i+1} - (y_i + h\phi(t_i, y_i))}{h} = \frac{y_{i+1} - y_i}{h} - \phi(t_i, y_i)$$

$$y_{i+1} = y_i + hf(t_i, y_i) + \frac{h^2}{2} f'(t_i, y_i) + \cdots + \frac{h^n}{n!} f^{(n-1)}(t_i, y_i) + \frac{h^{n+1}}{(n+1)!} f^{(n)}(\xi, y(\xi_i))$$

$$\begin{aligned} &w_0 = \alpha \\ &w_{i+1} = w_i + hT^{(n)}(t_i, w_i) \\ &\text{where } T^{(n)}(t_i, w_i) = f(t_i, w_i) + \frac{h}{2} \end{aligned}$$

6 Chap6 Direct Methods for Solving Linear Systems

6.1 6.1 Linear Systems of Equations

Gaussian elimination with backward substitution

6.2 6.2 Pivoting Strategies

Problem: small pivot element may cause trouble

Paritial Pivoting: Determine the smallest pk s.t. $|a_{pk}^{(k)}| = \max_{k \leqslant j \leqslant n} |a_{ik}^{(k)}|$ and interchange the pth and the kth rows

Scaled Partial Pivoting:

1. Define a scale factor s_i for each row as $s_i = \max_{1 \leqslant j \leqslant n} |a_{ij}|$
2. Determine the smallest $p \geqslant k$ s.t. $\frac{|a_{pk}^{(k)}|}{s_p} = \max_{k \leqslant i \leqslant n} \frac{|a_{ik}^{(k)}|}{s_i}$ and interchange the pth and the kth rows

Complete Pivoting: Search all the entries a_{ij} to find the entry with the largest magnitude

6.3 6.5 Matrix Factorization

$m_{ik} = a_{ik}/a_{kk}$

$$L_k = \begin{pmatrix} 1 & & & & \\ & \ddots & & \text{\Large 0} & \\ & & 1 & & \\ & & -m_{k+1,k} & & \\ & & \vdots & \ddots & \\ & & -m_{n,k} & & 1 \end{pmatrix}$$

Hence

$$L_1^{-1}L_2^{-1}\dots L_{n-1}^{-1} = \begin{pmatrix} 1 & & & \text{\Large 0} \\ & 1 & & \\ & & \ddots & \\ \text{\Large $m_{i,j}$} & & & 1 \end{pmatrix}$$

$$U = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ & a_{22} & \dots & a_{2n} \\ & & \dots & \vdots \\ & & & a_{nn} \end{pmatrix}$$

$A = LU$

6.4 6.6 Special Types of Matrices

Strictly Diagonally Dominant Matrix. $|a_{ii}| > \sum_{\substack{j=1,\\ j\neq i}}^{n} |a_{ij}| \quad \text{for each } i = 1, \dots, n$

Theorem 6.1. *A strictly diagonally dominant matrix A is nonsingular. Moreover, Gaussian elimination can be performed without row or column interchanges, and the computations will be stable w.r.t. the growth of roundoff errors*

Choleski's Method for Positive Definite Matrix:

Definition 6.1. *A matrix A is positive definite if ti's symmetric and if $\mathbf{x}^T\mathbf{A}\mathbf{x} > 0$ for every n-dimensional vector $\mathbf{x} \neq 0$*

Lemma 6.1. *A is positive definite*

1. A^{-1} *is positive definite as well, and* $a_{ii} > 0$
2. $\sum |a_{ij}| \leqslant \max |a_{kk}|$; $(a_{ij})^2 < a_{ii}a_{jj}$ *for each* $i \;\; j$
3. *Each of /A's leading principal submatrices* A_k*/ has a positive determinant*

$$U = \begin{pmatrix} & & \\ & u_{ij} & \\ & & \end{pmatrix} = \begin{pmatrix} u_{11} & & \\ & \ddots & \\ & & u_{nn} \end{pmatrix} \begin{pmatrix} 1 & & u_{ij}/u_{ii} \\ & 1 & \\ & & 1 \end{pmatrix} = D\tilde{U}$$

A is symmetric, hence

$$L = \tilde{U}^t, A = LDL^t$$

Let

$$D^{1/2} = \begin{pmatrix} \sqrt{u_{11}} & & \\ & \ddots & \\ & & \sqrt{u_{nn}} \end{pmatrix}, \tilde{L} = LD^{1/2/}, A = \tilde{L}\tilde{L}^t$$

Crout Reduction for tridiagonal Linear System

$$\begin{pmatrix} b_1 & c_1 & & & \\ a_2 & b_2 & c_2 & & \\ & \ddots & \ddots & \ddots & \\ & & a_{n-1} & b_{n-1} & c_{n-1} \\ & & & a_n & b_n \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_{n-1} \\ f_n \end{pmatrix}$$

$$A = \begin{pmatrix} \alpha_1 & & & \\ \gamma_2 & \ddots & & \\ & \ddots & \ddots & \\ & & \gamma_n & \alpha_n \end{pmatrix} \begin{pmatrix} 1 & \beta_1 & & \\ & \ddots & \ddots & \\ & & \ddots & \beta_{n-1} \\ & & & 1 \end{pmatrix}$$

7 Chap7 Iterative techiniques in Matrix algebra

7.1 7.1 Norms of vectors and matrices

Definition 7.1. *A vector norm on* R^n *is a function* $||\cdot|| : \mathbb{R}^n \to \mathbb{R}$ *with following properties for all* $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n, \alpha \in C$

1. $||\mathbf{x}|| \leqslant 0$; $||\mathbf{x}|| = 0 \Longleftrightarrow \mathbf{x} = \mathbf{0}$

2. $||\alpha\mathbf{x}|| = |\alpha| \cdot ||\mathbf{x}||$

3. $||\mathbf{x} + \mathbf{y}|| \leqslant ||\mathbf{x}|| + ||\mathbf{y}||$

$||\mathbf{x}||_1 = \sum_{i=1}^{n} |x_i|$. $||\mathbf{x}_p|| = (\sum_{i=1}^{n} |x_i|^p)^{1/p}$

Definition 7.2. *A sequence $\{\mathbf{x}^{(k)}\}_{k=1}^{\infty}$ of vectors in R^n converge to $\mathbf{x}$ w.r.t the norm $||\cdot||$ if given any $\epsilon > 0$ there exists an integer $N(\epsilon)$ s.t. $||\mathbf{x}^{(k)} - \mathbf{x}|| < \epsilon$ for all $k \geqslant N(\epsilon)$*

Theorem 7.1. *The sequence of vectors $\{\mathbf{x}^{(k)}\}$ converges to $\mathbf{x} \in R^n$ w.r.t. $||\cdot||$ if and only if $\lim_{k\to\infty} \mathbf{x}_i^{(k)} = x_i$ for each $i = 1, 2, \ldots, n$*

Definition 7.3. *If there exist positive constants C_1, C_2 s.t. $C_1||\mathbf{x}||_B \leqslant ||\mathbf{x}||_A \leqslant C_2||\mathbf{x}|_B|$. Then $||\cdot||_A, ||\cdot||_B$ are equivalent*

Theorem 7.2. *All the vector norm in R^n are equivalent*

Definition 7.4. *A matrix norm on the set of $n \times n$:*

1. $||\mathbf{A}|| \geqslant 0; ||\mathbf{A}|| = 0 \Longleftrightarrow \mathbf{A} = \mathbf{0}$

2. $||\alpha\mathbf{A}|| = |\alpha| \cdot ||\mathbf{A}||$

3. $||\mathbf{A} + \mathbf{B}|| \leqslant ||\mathbf{A}|| + ||\mathbf{B}||$

4. $||\mathbf{AB}|| \leqslant ||\mathbf{A}|| \cdot ||\mathbf{B}||$

Frobenius Norm: $||\mathbf{A}||_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n} |a_{ij}|^2}$

Natural Norm: $||\mathbf{A}||_p = \max_{\mathbf{x}\neq\mathbf{0}} \frac{||\mathbf{Ax}||_p}{||\mathbf{x}||_p} = \max_{\mathbf{z}\neq\mathbf{0}} ||\mathbf{A}\frac{\mathbf{z}}{||\mathbf{z}||}|| = \max_{||\mathbf{x}||_p=1} ||\mathbf{Ax}||_p$

$||\mathbf{A}||_\infty = \max_{1\leqslant i\leqslant n} \sum_{j=1}^{n} |a_{ij}|$, $||\mathbf{A}||_1 = \max_{1\leqslant j\leqslant n} \sum_{i=1}^{n} |a_{ij}|$, $||\mathbf{A}||_2 = \sqrt{\lambda_{\max}(\mathbf{A}^T\mathbf{A})}$

7.2 7.2 Eigenvalues and Eigenvectors

spectral radius.

Definition 7.5. *The spectral radius $\rho(A)$ of a matrix A is defined as* $\rho(A) = \max|\lambda|$ *where λ is an eigenvalue of A*

Theorem 7.3. *If A is an $n \times n$ matrix, then $\rho(A) \leqslant ||A||$ for any natural norm*

Proof. $|\lambda| \cdot ||\boldsymbol{x}|| = ||\lambda \boldsymbol{x}|| = ||A\boldsymbol{x}|| \leqslant ||A|| \cdot ||\boldsymbol{x}||$ □

Definition 7.6. *We call an $n \times n$ matrix A convergent if for all* $i, j = 1, \dots, n$ $\lim_{k\to\infty}(A^k)_{ij} = 0$

7.3 7.3 Iterative techniques for solving linear systems

Jacobi iterative method.

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \cdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = b_n \end{cases} \implies \begin{cases} x_1 = \frac{1}{a_{11}}(-a_{12}x_2 - \cdots - a_{1n}x_n + b_1) \\ x_2 = \frac{1}{a_{22}}(-a_{21}x_1 - \cdots - a_{2n}x_n + b_2) \\ \cdots \\ x_1 = \frac{1}{a_{nn}}(-a_{n2}x_1 - \cdots - a_{nn-1}x_{n-1} + b_n) \end{cases}$$

In matrix form,

$$A = \begin{pmatrix} D & -U & -U \\ -L & D & -U \\ -L & -L & D \end{pmatrix}$$

$$\begin{aligned} A\boldsymbol{x} = \boldsymbol{b} &\Leftrightarrow (D - L - U)\boldsymbol{x} = \boldsymbol{b} \\ &\Leftrightarrow D\boldsymbol{x} = (L + U)\boldsymbol{x} + \boldsymbol{b} \\ &\Leftrightarrow \boldsymbol{x} = \underbrace{D^{-1}(L + U)}_{T_j}\boldsymbol{x} + \underbrace{D^{-1}}_{\boldsymbol{c}_j}\boldsymbol{b} \end{aligned}$$

. T_j is Jacobi iterative matrix. $\boldsymbol{x}^{(k)} = T_j\boldsymbol{x}^{(k-1)} + \boldsymbol{c}_j$

Gauss-Seidel iterative method

$$\begin{aligned} &\boldsymbol{x}^{(k)} = D^{-1}(L\boldsymbol{x}^{(k)} + U\boldsymbol{x}^{(k-1)}) + D^{-1}\boldsymbol{b} \\ \Leftrightarrow &(D - L)\boldsymbol{x}^{(k)} = U\boldsymbol{x}^{(k-1)} + \boldsymbol{b} \\ \Leftrightarrow &\boldsymbol{x}^{(k)} = \underbrace{(D - L)^{-1}U\boldsymbol{x}^{(\boldsymbol{k-1})}}_{T_g} + \underbrace{(D - L)^{-1}\boldsymbol{b}}_{\boldsymbol{c}_g} \end{aligned}$$

convergence of iterative methods

Theorem 7.4. *the following are equivalent:*

1. *A is a convergent matrix*

2. $\lim_{n\to\infty} ||A^n|| = 0$ *for some natural norm*

3. $\lim_{n\to\infty} ||A^n|| = 0$ *for all natural norms*

4. $\rho(A) < 1$

5. $\lim_{n\to\infty} A^n\boldsymbol{x} = \mathbf{0}$ *for every $\boldsymbol{x}$*

$\boldsymbol{e}^{(k)} = \boldsymbol{x}^{(k)} - \boldsymbol{x}^* = (T\boldsymbol{x}^{(k-1)} + \boldsymbol{c}) - (T\boldsymbol{x}^* + \boldsymbol{c}) = T(\boldsymbol{x}^{(k-1)} - \boldsymbol{x}^*) = T\boldsymbol{e}^{(k-1)} \Rightarrow \boldsymbol{e}^{(k)} = T^k\boldsymbol{e}^{(0)}$. $||\boldsymbol{e}^{(k)} \leqslant ||T|| \cdot ||\boldsymbol{e}^{(k-1)}|| \leqslant \cdots \leqslant ||T||^k \cdot ||ble^{(0)}||$

Theorem 7.5. *For any $\boldsymbol{x}^{(0)} \in R^n$, the sequence $\{\boldsymbol{x}^{(k)}\}_{k=0}^{\infty}$ defined by $\boldsymbol{x}^{(k)} = T\boldsymbol{x}^{(k-1)} + \boldsymbol{c}$ for each k, converges to the unique solution of $\boldsymbol{x} = T\boldsymbol{x} + \boldsymbol{c}$ if and only if $\rho(T) < 1$*

$$\rho(T) < 1 \Longrightarrow (I - T)^{-1} = \sum_{j=0}^{\infty} T^j$$

Theorem 7.6. *If $\|T\| < 1$ for any natural matrix norm and $\boldsymbol{c}$ is a given vector, then the sequence $\{\boldsymbol{x}^{(k)}\}_{k=0}^{\infty}$ defined by $\boldsymbol{x}^{(k)} = T\boldsymbol{x}^{(k-1)} + \boldsymbol{c}$ converges for any $\boldsymbol{x}^{(0)} \in R^n$ to a vector $\boldsymbol{x}$. And the following error bounds hold*

1. $\left\|\boldsymbol{x} - \boldsymbol{x}^{(k)}\right\| \leqslant \|T\|^k \left\|\boldsymbol{x} - \boldsymbol{x}^{(0)}\right\|$

2. $\left\|\boldsymbol{x} - \boldsymbol{x}^{(k)}\right\| \leqslant \frac{\|T\|^k}{1-\|T\|} \left\|\boldsymbol{x}^{(1)} - \boldsymbol{x}^{(0)}\right\|$

Theorem 7.7. *If A is a strictly diagonally dominant, then for any choice of $\boldsymbol{x}^{(0)}$, both the Jacobi and Gauss-Seidel methods give sequences $\{\boldsymbol{x}^{(k)}\}_{k=0}^{\infty}$ that converges to the unique solution*

relaxation methods. $x_i^{(k)} = \frac{1}{a_{ii}}(b_i - \sum_{j=1}^{i-1} a_{ij}x_i^{(k)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k-1)}) = x_i^{(k-1)} + \dfrac{r_i^{(k)}}{a_{ii}}$ and relaxation method is $x_i^{(k)} = x_i^{(k-1)} + \omega\frac{r_i^{(k)}}{a_{ii}}$

Theorem 7.8. *(kahan) If $a_{ii} \neq 0$ for each i. Then $\rho(T_\omega) \geqslant |\omega - 1|$.*

This implies the SOR method can converge only if $0 < \omega < 2$

Theorem 7.9. *(Ostrowski-Reich) If A is positive definite and* $0 < \omega < 2$*, the SOR converges*

Theorem 7.10. *If A is positive definite and tridiagonal, then* $\rho(T_g) = (\rho(T_j))^2 < 1$*, and the optimal choice of* ω *for the SOR method is* $\omega = \frac{2}{1+\sqrt{1-(\rho(T_j))^2}}$*. With this choice of* ω*, we have* $\rho(T_\omega) = \omega - 1$

7.4 7.4 Error bounds and iterative refinement

Assume that A is accurate and $\boldsymbol{b}$ has the error $\delta\boldsymbol{b}$, then $\boldsymbol{A}(\boldsymbol{x} + \delta\boldsymbol{x}) = \boldsymbol{b} + \delta\boldsymbol{b}$

Theorem 7.11. *Suppose* $\tilde{\boldsymbol{x}}$ *is an approximation to the solution of* $\boldsymbol{Ax} = \boldsymbol{b}$ *A is nonsingular matrix. Then for any natural norm,*

$$||\boldsymbol{x} - \tilde{\boldsymbol{x}}|| \leqslant ||\boldsymbol{r}|| \cdot ||A^{-1}||$$

and if $\boldsymbol{x} \neq \boldsymbol{0}, \boldsymbol{b} \neq \boldsymbol{0}$*,*

$$\frac{||\delta\boldsymbol{x}||}{||\boldsymbol{x}||} \leqslant ||\boldsymbol{A}|| \cdot ||\boldsymbol{A}^{-1}|| \cdot \frac{||\delta\boldsymbol{b}||}{||\boldsymbol{b}||}$$

Proof. $\boldsymbol{r} = \boldsymbol{b} - \boldsymbol{A}\tilde{\boldsymbol{x}} = A\boldsymbol{x} - A\tilde{\boldsymbol{x}}$ and A is nonsingular. Hence $\boldsymbol{x} - \tilde{\boldsymbol{x}} = A^{-1}\boldsymbol{r}$. Since $\frac{||A^{-1}\boldsymbol{r}||}{||\boldsymbol{r}||} \leqslant ||A^{-1}||$, $||\boldsymbol{x} - \tilde{\boldsymbol{x}}|| = ||A^{-1}\boldsymbol{x}|| \leqslant ||A^{-}1|| \cdot ||\boldsymbol{r}||$. Also $||\boldsymbol{b}|| \leqslant ||A|| \cdot ||\boldsymbol{x}||$. So $1/||\boldsymbol{x}|| \leqslant ||A||/||\boldsymbol{b}||$ □

Theorem 7.12. *If a matrix B satisfies* $||B|| < 1$ *for some natural norm, then*

1. *$I \pm B$ is nonsingular*
2. $||(I \pm B)^{-1}|| \leqslant \frac{1}{1-||B||}$

Assume $\boldsymbol{b}$ is accurate, A has the error δA, then $(A + \delta A)(\boldsymbol{x} + \delta\boldsymbol{x}) = \boldsymbol{b}$. Hence $\frac{||\delta\boldsymbol{x}||}{||\boldsymbol{x}||} \leqslant \frac{||A^{-1}||\cdot||\delta A||}{1-||A^{-1}||\cdot||\delta A||} = \frac{||A||\cdot||A^{-1}||}{1-||A||\cdot||A^{-1}||\cdot\frac{||\delta A||}{||A||}}$

condition number K(A) is $||A|| \cdot ||A^{-1}||$

Theorem 7.13. *Suppose A is nonsingular and* $||\delta A|| \leqslant \frac{1}{||A^{-1}||}$*. The solution* $\boldsymbol{x} + \delta\boldsymbol{x}$ *to* $(A + \delta A)(\boldsymbol{x} + \delta\boldsymbol{x})$ *approximates the solution* $\boldsymbol{x}$ *of* $A\boldsymbol{x} = \boldsymbol{b}$ *with the error estimate*

$$\frac{||\delta\boldsymbol{x}||}{||\boldsymbol{x}||} \leqslant \frac{K(A)}{1 - K(A)||\delta A||/||A||}(\frac{||\delta A||}{||A||} + \frac{||\delta\boldsymbol{b}||}{||\boldsymbol{b}||})$$

note:

1. If A is symmetric, then $K(A)_2 = \frac{\max|\lambda|}{\min|\lambda|}$
2. $K(A)_p \geqslant 1$ for all natural norm
3. $K(\alpha A)_= K(A)$ for any $\alpha \in R$
4. $K(A)_2 = 1$ if A is orthogonal
5. $K(RA)_2 = K(AR)_2 = K(A)_2$ for all orthogonal matrix R_

iterative refinement:

Theorem 7.14. *Suppose $\boldsymbol{x}^*$ is an approximation to the solution of $A\boldsymbol{x} = \boldsymbol{b}$, A is nonsingular matrix and $\boldsymbol{r} = \boldsymbol{b} - A\boldsymbol{x}$. Then for any natural norm, $||\boldsymbol{x} - \boldsymbol{x}^* \leqslant ||\boldsymbol{r}|| \cdot ||A^{-1}||$, and if $\boldsymbol{x}, \boldsymbol{b} \neq \boldsymbol{0}$*

$$\frac{||\boldsymbol{x} - \boldsymbol{x}^*||}{||\boldsymbol{x}||} \leqslant K(A)\frac{||\boldsymbol{r}||}{||\boldsymbol{b}||}$$

refinement

1. $A\boldsymbol{x} = \boldsymbol{b}$ => approximation $\boldsymbol{x}_1$
2. $\boldsymbol{r}_1 = \boldsymbol{b} - A\boldsymbol{x}_1$
3. $A\boldsymbol{d}_1 = \boldsymbol{r}_1$ => $\boldsymbol{d}_1$
4. $\boldsymbol{x}_2 = \boldsymbol{x}_1 + \boldsymbol{d}_1$

8 Chap8 Approximation theory

Given $x_1 \dots x_m$ and $y_1 \dots y_m$ find a **simpler** function $P(x) \approx f(x)$

8.1 8.1 Discrete least squares approximation

Determine the polynomial $P_n(x) = a_0 + a_1 x + \cdots + a_n x^n$ to approximate the data $\{(x_i, y_i) \mid i = 1, 2, \dots, m\}$ s.t. the least squares error $E_2 = \sum_{i=1}^{m}(P_n(x_i) - y_i)^2$ is minimized. Here $n \ll m$

$E_2(a_0, \dots, a_n) = \sum_{i=1}^{m}(a_0 + a_1 x_i + \cdots + a_n x_i^n - y_i)^2$

For E_2 to be minimized it's necessary that $\frac{\partial E_2}{\partial a_k} = 0$

$$\begin{aligned}
0 = \frac{\partial E_2}{\partial a_k} &= 2\sum_{i=1}^{m}(P_n(x_i) - y_i)\frac{\partial P_N(x_i)}{\partial a_k} \\
&= 2\sum_{i=1}^{m}(\sum_{j=0}^{n} a_j x_i^j - y_i)x_i^k \\
&= 2(\sum_{j=0}^{n} a_j(\sum_{i=1}^{m} x_i^{j+k}) - \sum_{i=1}^{m} y_i x_i^k)
\end{aligned}$$

Let $b_k = \sum_{i=1}^{m} x_i^k, c_k = \sum_{i=1}^{m} y_i x_i^k$, then

$$\begin{pmatrix} b_{0+0} & \dots & b_{0+n} \\ \vdots & \vdots & \vdots \\ b_{n+0} & \dots & b_{n+n} \end{pmatrix} \begin{pmatrix} a_0 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} c_0 \\ \vdots c_n \end{pmatrix}$$

8.2 8.2 orthogonal polynomials and least squares approximation

Theorem 8.1. *If $\varphi_j(x)$ is a polynomial of degree j for each $j = 0, \dots, n$, then $\{\varphi_0(x), \dots, \varphi_n(x)\}$ is linearly independent on any interval $[a, b]$*

Theorem 8.2. *Let $\prod_n$ be the set of all polynomials of degree at most n. If $\{\varphi_0(x), \dots, \varphi_n(x)\}$ is a collection of linearly independent polynomials in $\prod_n$ then any polynomials in $\prod_n$ can be written uniquely as a linear combination of $\{\varphi_0(x), \dots, \varphi_n(x)\}$*

Definition 8.1. *For a general linear independent set of functions $\{\varphi_0(x), \dots, \varphi_n(x)\}$, a linear combination of $\{\varphi_0(x), \dots, \varphi_n(x)\}$. $P(x) = \sum_{j=0}^{n} \alpha_j \varphi_j(x)$ is called a generalized polynomial*

Weight function

$$E = \sum w_i[P(x_i) - y_i]^2$$
$$E = \int_a^b w(x)[P(x) - f(x)]^2 dx$$

$$\sum w_i \|P(x) - f(x)\|_2^2 = \sum w_i \boldsymbol{e}^T \boldsymbol{e} = \boldsymbol{e}^T W \boldsymbol{e}$$

where #+ATTR$_{\text{LATEX}}$:mode math :environment pmatrix :math-preffix W=

w1

. . .

wn__

The **general least squares approximation problem**. E is minimized

Inner product and **norm**

$$(f,g) = \begin{cases} \sum_{i=1}^{m} w_i f(x_i) g(x_i) \\ \int_a^b w(x) f(x) g(x) dx \end{cases}$$

It can be shown that (f,g) is an **inner proudct** and $\|f\| = \sqrt{(f,f)}$ is a **norm**

Hence, The general least squares approximation problem is to find a generalized polynomial $P(x)$ such that $E = (P-y, P-y) = \|P-y\|^2$ is minimized.

Let $P(x) = a_0\phi_0(x) + \cdots + a_n\phi_n(x)$. $\frac{\partial E}{\partial a_k} = 0 \Longrightarrow \sum_{j=0}^{n} (\phi_k, \phi_j) a_j = (\phi_k, f)$.

$$\begin{pmatrix} \\ b_{ij} = (\phi_i, \phi_j) \\ \\ \end{pmatrix} \begin{pmatrix} a_0 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} (\phi_0, f) \\ \cdots \\ (\phi_n, f) \end{pmatrix} = \vec{c}$$

Example. When approximating $f(x) \in C[0,1]$ with $\phi_j(x) = x^j$ and $w(x) = 1$, then

$$(\phi_i, \phi_j) = \int_0^1 x^i x^j dx = \frac{1}{i+j+1}$$

Hilbert matrix.

Improvement: Find a general linear independent set of functions s.t. any pair is **orthogonal**, then the matrix will be diagonal. And

$$a_k = \frac{(\phi_k, f)}{(\phi_k, \phi_k)}$$

Construction

Theorem 8.3. *the set of polynomial functions defined in the following way* *is orthogonal on [a,b] w.r.t. weight function w*

$$\begin{aligned}\phi_0(x) &= 1\\ \phi_1(x) &= x - B_1\\ \phi_k(x) &= (x - B_k)\phi_{k-1}(x) - C_k\phi_{k-2}(x)\\ B_k &= \frac{(x\phi_{k-1}, \phi_{k-1})}{(\phi_{k-1}, \phi_{k-1})}\\ C_k &= \frac{(x\phi_{k-1}, \phi_{k-2})}{(\phi_{k-2}, \phi_{k-2})}\end{aligned}$$

Example. Approximate

$$\begin{pmatrix} x & 1 & 2 & 3 & 4 \\ y & 4 & 10 & 18 & 26 \end{pmatrix}$$

with $y = a_0 + a_1x + a_2x^2, w = 1$
Solution. $y = a_0\phi_0(x) + a_1\phi_1(x) + a_2\phi_2(x)$. $\phi_0(x) = 1$

8.3 8.3 Chebyshev polynomials and economization of power series

Minimize $\|P - y\|_\infty$, minimax problem

1. Find a polynomial $P_n(x)$ of degree n s.t. $\|P_n - f\|_\infty$ is minimized

   Definition 8.2. *If $P(x_0) - f(x_0) = \pm\|P - f\|_\infty$, x_0 is called a ($\pm$) deviation point*

   We can estimate the features of the polynomial

   (a) If $f \in C[a, b]$ and f is not a polynomial of degree n, then there exists a unique polynomial $P_n(x)$ s.t. $\|P_n - f\|_\infty$ is minimized
   (b) $P_n(x)$ exists, and must have both + and - deviation points
   (c)

   Theorem 8.4. *Chebyshev Theorem $P_n(x)$ minimizes$\|P_n - f\| \Longleftrightarrow$ $P_n(x)$ has at least n+2 alternating + and - deviation points w.r.t. f. That is, there exists a set of points $a \leqslant t_1 < \cdots < t_{n+2} \leqslant b$ s.t.*

   $$P_n(t_k) - f(t_k) = \pm(-1)^k\|P_n - f\|_\infty$$

   The set $\{t_k\}$ is called the {Chebyshev altenating sequence}

2. Determine the interpolating points $\{x_0, \dots, x_n\}$ s.t. $P_n(x)$ minimizes the remainder

$$|P_n(x) - f(x)| = |R_n(x)| = \left| \frac{f^{(n+1)}(\xi_x)}{(n+1)!} \prod_{i=0}^{n} (x - x_i) \right|$$

2.1 Find $\{x_1, \dots, x_n\}$ s.t. $\|\omega_n\|_\infty$ is minimized on $[-1, 1]$, where $\omega_n(x) = \prod_{i=1}^{n} (x - x_i)$.

Since $\omega_n(x) = x^n - P_{n-1}(x)$, the problem becomes to

3. Find a polynomial $P_{n-1}(x) s.t. \|x^n - P_{n-1}(x)\|_\infty$ is minimized on $[-1, 1]$

Chebyshev polynomials. Consider the $n+1$ extreme values of $\cos(n\theta)$ on $[0, \pi]$.

Let $x = \cos(\theta)$, then $x \in [-1, 1]$, $T_n(x) = \cos(n\theta) = cos(n \cdot \arccos x)$ is called the Chebyshev polynomial.

Properties:

1. $t_k = \cos(\frac{k}{n}\pi), k = 0, \dots, n, T_n(t_k) = (-1)^k \|T_n(x)\|_\infty$
2. $T_n(x)$ has n roots $x_k = \cos(\frac{2k-1}{2n}\pi), k = 1, \dots, n$
3. T_n has recurrence relation

$$T_0(x) = 1, T_1(x) = x, T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)$$

4. $\{T_0(x), T_1(x), \dots\}$ are orthogonal on $[-1, 1]$ w.r.t. weight function $w(x) = 1/\sqrt{1 - x^2}$

$$(T_n, T_m) = \int_{-1}^{1} \frac{T_n(x)T_m(x)}{\sqrt{1 - x^2}} dx = \begin{cases} 0 & n \neq m \\ \pi & n = m = 0 \\ \pi/2 & n = m \neq 0 \end{cases}$$

$w_n(x) = x^n - P_{n-1}(x) = T_n(x)/2^{n-1}$. Let $\widetilde{\prod}$ ={monic polynomials of degree n}.

$$\min_{w_n \in \widetilde{\prod}} \|w_n\|_\infty = \left\| \frac{1}{2^{n-1}} T_n(x) \right\|_\infty = \frac{1}{2^{n-1}}$$

$$|P_n(x) - f(x)| = |R_n(x)| = \left| \frac{f^{(n+1)}(\xi_x)}{(n+1)!} \prod_{i=0}^{n} (x - x_i) \right|$$

Take the $n+1$ roots of $T_{n+1}(x)$ as the interpolating points, then the interpolating polynomial $P_n(x)$ assumes the minimum upper bound of the absolute error $\frac{M}{2^n (n+1)!}$

Economization of power series. Given $P_n(x) \approx f(x)$, economization of ppppppower series is to reduce the degree of polynomial with a **minimal loss of accuracy**

Consider approximating an arbitrary n-th degree polynomial

$$P_n(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$$

with a polynomial $P_{n-1}(x)$ by removing an n-th degree polynomial $Q_n(x)$ that has the coefficient a_n for x_n. Then

$$\max_{[-1,1]} |f(x) - P_{n-1}(x)| \leqslant \max_{[-1,1]} |f(x) - P_n(x)| + \max_{[-1,1]} |Q_n(x)|$$

To minimize the loss of accuracy, $Q_n(x) = a_n \frac{T_n(x)}{2^{n-1}}$

Example. The 4-th order Taylor polynomial for $f(x) = e^x$ on $[-1, 1]$ is

$$P_4 = 1 + x + \frac{x^2}{2} + \frac{x^3}{5} + \frac{x^4}{24}$$

.The upper bound of truncation error is $|R_4(x)| \leqslant \frac{e}{5!} \left| x^5 \right| \approx 0.023$

solution. $T_4 = 8x^4 - 8x^2 + 1, Q_4$

9 chap9 Approximating Eigenvalues

9.1 9.3 the power method

the original method Assumptions: A is an $n \times n$ matrix with eigenvalues satisfying $|\lambda_1| > |\lambda_2| \geqslant \cdots \geqslant |\lambda_n| \geqslant 0$

$$\boldsymbol{x}^{(0)} = \sum_{j=1}^{n} \beta_j \boldsymbol{v}_j, \quad \beta_1 \neq 0$$

$$\boldsymbol{x}^{(1)} = A\boldsymbol{x}^{(0)} = \sum_{j=1}^{n} \beta_j \lambda_j \boldsymbol{v}_j$$

$$\boldsymbol{x}^{(2)} = A\boldsymbol{x}^{(1)} = \sum_{j=1}^{n} \beta_j \lambda_j^2 \boldsymbol{v}_j$$

$$\ldots$$

$$\boldsymbol{x}^{(k)} \approx \lambda_1^k \beta_1 \boldsymbol{v}_1, \quad \lambda_1 \approx \frac{\boldsymbol{x}_i^{(k)}}{\boldsymbol{x}_i^{(k-1)}}$$

Normalization. Suppose $||\boldsymbol{x}||_\infty = 1$. Let $||\boldsymbol{x}^{(k)}||_\infty = |x_{p_k}^{(k)}|$.Then $\boldsymbol{u}^{(k-1)} = \frac{\boldsymbol{x}^{(k-1)}}{|x_{p_{k-1}}^{(k-1)}|}$ and $\boldsymbol{x}^{(k)} = A\boldsymbol{u}^{(k-1)}$. Then $\boldsymbol{u}^{(k)} = \frac{\boldsymbol{x}^{(k)}}{|x_{p_k}^{(k)}|} \to \boldsymbol{v}_1$. $\lambda_1 \approx \frac{\boldsymbol{x}_i^{(k)}}{\boldsymbol{u}_i^{(k-1)}} = \boldsymbol{x}_{p_{k-1}}^{(k)}$

Note:

1. the method works for **multiple** eigenvalues $\lambda_1 = \lambda_2 = \cdots = \lambda_r$
2. the method fails to converge if $\lambda_1 = -\lambda_2$
3. Aitken's Δ^2 can be used

Rate of convergence. $\boldsymbol{x}^{(k)} = A\boldsymbol{x}^{(k-1)} = \lambda_1^k \sum_{j=1}^{n} \beta_j (\frac{\lambda_j}{\lambda_1})^k \boldsymbol{v}_j$. Make $|\lambda_2/\lambda_1|$ as small as possible. Assume $\lambda_1 > \lambda_2 \geqslant \cdots \geqslant \lambda_n, |\lambda_2| > |\lambda_n|$. Let $B = A - pI$, then $|\lambda I - A| = |\lambda I - (B + pI)| = |(\lambda - p)I - B|$. Hence $\lambda_A - p = \lambda_B$. Since $\frac{|\lambda_2 - p|}{|\lambda_1 - p|} < \frac{|\lambda_2|}{|\lambda_1|}$. The iteration is fast

Inverse power method. If A has $|\lambda_1| \geqslant |\lambda_2| \geqslant \cdots > |\lambda_n|$, then A^{-1} has $|\frac{1}{\lambda_n}| > |\frac{1}{\lambda_{n-1}}| \geqslant \cdots \geqslant |\frac{1}{\lambda_1}|$

10 TODO hw [0/15]

`C-u C-c C-c`

- ☐ NA01-CH1-A
- ☐ NA02-CH2-A

- NA03-CH6-AB
- NA04-CH6-A
- NA04-CH7-A
- NA05-CH7-A
- NA06-CH3-A
- NA06-CH7-A conditional number hilber matrix
- NA06 CH9 -A
- NA07-CH3-AB
- NA08-CH3-A
- NA08-CH8-A least squares polynomial
- NA09-CH8-A least squares polynomial orthogonal
- NA10-CH4-A numerical differentiation
- NA10-CH8-A