

《工程硕士数学》期末复习

第二章 解线性方程组的直接解法

1. Gauss消去法

- Gauss顺序消去法
 - 消去 ($O(n^3)$)：将 $Ax = b$ 通过初等行变换化简为 $Ux = b$ 。
 - 从第二行开始，给每行乘以一个不同的系数 l ，使得每行第一位都等于第一行第一位的相反数；
 - 从第二行开始，给每行加上第一行，使得每行第一位都等于0；
 - 从第三行开始，给每行乘以一个不同的系数 l ，使得每行第二位都等于第二行第二位的相反数；
 - 从第三行开始，给每行加上第二行，使得每行第二位都等于0；
 -
 - 最后得到上三角阵 U 。
 - 回代 ($O(n^2)$)：从最后一行开始逐行解 $Ux = b$ 。
- Gauss列主元消去法：若消去到第 $k-1$ 步 ($k \geq 1$)，此时第 k 行及以下部分的第 k 位待消去，则挑选所有这些数中绝对值最大的，把它所在行换到第 k 行，再消去。

$$\left[\begin{array}{cccccc|c} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1k}^{(1)} & \cdots & a_{1n}^{(1)} & b_1^{(1)} \\ & a_{22}^{(2)} & \cdots & a_{2k}^{(2)} & \cdots & a_{2n}^{(2)} & b_2^{(1)} \\ & & \ddots & \vdots & & \vdots & \vdots \\ & & & \boxed{a_{kk}^{(k)}} & \cdots & a_{kn}^{(k)} & b_k^{(k)} \\ & & & \vdots & & \vdots & \vdots \\ & & & a_{nk}^{(k)} & \cdots & a_{nn}^{(k)} & b_n^{(k)} \end{array} \right].$$

◦ 示例：

例 2.2.4 用列主元法解方程组 $Ax=b$, 计算过程取五位数字, 其中

$$[A \mid b] = \left[\begin{array}{ccc|c} -0.002 & 2 & 2 & 0.4 \\ 1 & 0.78125 & 0 & 1.3816 \\ 3.996 & 5.5625 & 4 & 7.4178 \end{array} \right].$$

这一题的第一步，就需要先把3.996所在行换到第一行，再进行第一次消去。

2. 顺序主子式

- **顺序主子式**：对 $n \times n$ 的方阵，求其左上角 1×1 、 2×2 、.....、 $n \times n$ 这 n 个部分的行列式，这个过程称为求顺序主子式 Δ_1 、 Δ_2 、...、 Δ_n 。
- **Gauss消去法的可行性**：当顺序主子式任何一项都不为零时，方阵 A 可以使用上述的Gauss消去。

3. LU分解

- **Doolittle分解（LU分解）**：通过Gauss消去法得到 U 以后，一定能找到下三角阵 L ，使 $A = LU$ 。
 - L 的对角元素显然全为1。
- **Crout分解**：对换Doolittle分解中 L 的对角元素和 U 的对角元素，使得 U 的对角元素全为1。
- **LDU分解**：提取Doolittle分解中 U 的对角元素，形成对角矩阵 D ，使得 U 变成对角元素全为1的上三角阵 \tilde{U} ，而此时 A 可以分解为 $A = LD\tilde{U}$ 。
- **单位上（下）三角阵**：对角元素全为1的上（下）三角阵。
- **Doolittle分解的存在唯一性**：存在唯一Doolittle分解的条件为方阵 $A_{n \times n}$ 的顺序主子式 Δ_1 到 Δ_{n-1} 都不为零。
 - 显然，如果Doolittle分解存在且唯一，则Crout、LDU分解也都存在且唯一。

4. 三对角矩阵

- **三对角矩阵**：三对角矩阵是形如下图的方阵。

$$A = \begin{bmatrix} b_1 & c_1 & & & \\ a_2 & b_2 & c_2 & & \\ & \ddots & \ddots & \ddots & \\ & & a_{n-1} & b_{n-1} & c_{n-1} \\ & & & a_n & b_n \end{bmatrix}$$

- **三对角矩阵LU分解的形状**：三对角矩阵的LU分解一定形如下图。

$$L = \begin{bmatrix} 1 & & & & \\ l_2 & 1 & & & \\ & l_3 & 1 & & \\ & & \ddots & \ddots & \\ & & & l_n & 1 \end{bmatrix}, \quad U = \begin{bmatrix} u_1 & c_1 & & & \\ & u_2 & c_2 & & \\ & & \ddots & \ddots & \\ & & & c_{n-1} & \\ & & & & u_n \end{bmatrix}$$

- **追赶法**：
 - 先进行LU分解，使得 $LUx = b$ 。
 - 计算 $Ly = b$ ，此时有 $y = Ux$ 。
 - 计算 $Ux = y$ ，求出 x 。

5. 正定矩阵的Cholesky分解

- **正定矩阵**：顺序主子式全部大于0的实对称方阵。
- **Cholesky分解**：对正定对称阵作LU分解，必有 $U = L^T$ 、 $A = LL^T$ 。这种分解称为Cholesky分解。
- **平方根法**（Cholesky法）：
 - 先进行Cholesky分解，使得 $LL^T x = b$ 。
 - 计算 $Ly = b$ ，此时有 $y = L^T x$ 。
 - 计算 $L^T x = y$ ，求出 x 。

6. 范数

- **向量范数**：
 - **1-范数**：向量所有元素的和。
 - **2-范数**：向量模长。
 - **无穷范数**（ ∞ -范数）：向量所有元素的绝对值的最大值。
- **谱半径**：方阵 A 的谱半径为其所有特征值的绝对值的最大值，记为 $\rho(A)$ 。
 - 回忆：特征多项式为 $|\lambda I - A|$ 计算出的多项式，特征值为特征多项式的零点。
- **矩阵范数**：
 - **1-范数**（列范数）：对所有元素取绝对值，再对每列进行求和。取最大的和为1-范数。
 - **2-范数**： $\sqrt{\rho(A^T A)}$ 。
 - 当 A 为对称矩阵时，有 $\sqrt{\rho(A^T A)} = \sqrt{\rho(A^2)} = \rho(A)$ 。
 - **无穷范数**（ ∞ -范数、行范数）：对所有元素取绝对值，再对每行进行求和。取最大的和为无穷范数。

7. 条件数与病态

- **条件数**： $Cond(A)_n = \|A\|_n \|A^{-1}\|_n$ ，其中 $\|A\|$ 表示矩阵范数， $\|A\|_1$ 为1-范数， $\|A\|_2$ 为2-范数，.....。
 - 回忆：
 - 矩阵元素的代数余子式为 $B_{ij} = (-1)^{i+j} A_{ij}$ 。
 - 矩阵所有元素取代数余子式会得到伴随矩阵 A^* ，而 $A^{-1} = \frac{1}{|A|} A^*$ 。
 - **病态**：对病态矩阵 A 的任意元素进行扰动（加减一个微小量），都会导致 $Ax = b$ 求解结果的巨大变化。
 - **病态矩阵的判别**：当 A 的任意一种条件数的数量级远大于 A 的数量级时， A 就是病态的。
 - 比较可能出现病态的矩阵：
 - 各元素数量级差别很大的矩阵；
 - 进行列主元消去或LU分解时，发现主元也很小的矩阵；
 - 行列式很小的矩阵。
-

第三章 解线性方程组的迭代法

1. Jacobi、Gauss-Seidel法与SOR法

当矩阵 A 满足下图所示形式时：

$$A = (a_{ij})_{n \times n} \in R^{n \times n}$$
$$A = D - L - U; D = \text{diag}[a_{11}, a_{22}, \dots, a_{nn}]$$

$$L = \begin{bmatrix} 0 & & & & \\ -a_{21} & 0 & & & \\ -a_{31} & -a_{32} & 0 & & \\ \vdots & & & \ddots & \\ -a_{n1} & -a_{n2} & & -a_{n,n-1} & 0 \end{bmatrix} \quad U = \begin{bmatrix} 0 & -a_{12} & -a_{13} & \dots & -a_{1n} \\ & 0 & -a_{23} & & -a_{2n} \\ & & 0 & & \\ & & & \ddots & \\ & & & & -a_{n-1,n} \\ & & & & & 0 \end{bmatrix}$$

- **Jacobi法** : $\begin{cases} B = D^{-1}(L + U) \\ x^{(k+1)} = Bx^{(k)} + D^{-1}b \end{cases}$;
- **Gauss-Seidel法** : $\begin{cases} B = (D - L)^{-1}U \\ x^{(k+1)} = Bx^{(k)} + (D - L)^{-1}b \end{cases}$;
- **SOR法** : $\begin{cases} B = L_{\omega} = (D - \omega L)^{-1}[(1 - \omega)D + \omega U] \\ x^{(k+1)} = L_{\omega}x^{(k)} + \omega(D - \omega L)^{-1}b \end{cases}$, 式中 ω 为自主选择的松弛因子。
- **迭代矩阵** : 以上三式中的 B 称为迭代矩阵。

2. 迭代法的收敛

- **迭代法的收敛性判定** :
 - 对任何迭代法, 若迭代矩阵 B 满足 $\rho(B) < 1$, 则迭代法收敛。
 - 若 A 严格对角占优, 则无论用Jacobi法还是Gauss-Seidel法求解 $Ax = b$, 均收敛。
 - 回忆: 若每行的对角元素绝对值均大于其他元素绝对值之和, 则这样的矩阵为严格对角占优矩阵。
 - 因此, 若 $Ax = b$ 无法直接用迭代法求解, 可以先进行行交换, 使 A 为严格对角占优矩阵, 再求解。
 - 对任何对称正定矩阵 A , Gauss-Seidel法收敛。
 - 对任何对称正定矩阵 A , 若将非对角元素全部取相反数, 新矩阵仍然对称正定, 则Jacobi法收敛。
 - 对任何 $\omega \in (0, 2)$, 有SOR法收敛。
 - 特别的, $\omega < 1$ 称为低松弛, $\omega > 1$ 为高松弛, $\omega = 1$ 时SOR法退化回Gauss-Seidel法。
- **迭代法的渐进收敛速度** : $R(B) = -\ln \rho(B) > 0$ 。 $\rho(B)$ 越小, 迭代法收敛越快。
 - **最佳松弛因子** : 对三对角对称正定矩阵, SOR法有最佳松弛因子 $\begin{cases} B_J = D^{-1}(L + U) \\ \omega_b = \frac{2}{1 + \sqrt{1 - \rho^2(B_J)}} \end{cases}$ 。
 - 此时有 $\rho(L_{\omega_b}) = \omega_b - 1$ 。
 - 当 $\omega \approx \omega_b$ 时, SOR法远快于Jacobi法。

3. 共轭梯度法

- **最速下降法**：
$$\begin{cases} r^{(k)} = b - Ax^{(k)} \\ \alpha_k = \frac{(r^{(k)}, r^{(k)})}{(Ar^{(k)}, r^{(k)})} \\ x^{(k+1)} = x^{(k)} + \alpha_k r^{(k)} \end{cases}。$$
- **共轭梯度法 (CG法)**：
$$\begin{cases} \alpha_k = \frac{(r^{(k)}, r^{(k)})}{(Ap^{(k)}, p^{(k)})} \\ x^{(k+1)} = x^{(k)} + \alpha_k p^{(k)} \\ r^{(k+1)} = r^{(k)} - \alpha_k Ap^{(k)}, \text{ 初值可取 } \begin{cases} r^{(0)} = b - Ax^{(0)} \\ p^{(0)} = r^{(0)} \end{cases} \\ \beta_k = \frac{(r^{(k+1)}, r^{(k+1)})}{(r^{(k)}, r^{(k)})} \\ p^{(k+1)} = r^{(k+1)} + \beta_k p^{(k)} \end{cases}。$$
 - CG法显著快于最速下降法和J法，一般也显著快于SOR法。
 - CG法能保证对 $A_{n \times n}$ 最多只需 n 步就求出精确解。

第四章 非线性方程的数值解法

1. 不动点迭代法

- **收敛**：把方程化为 $x = f(x)$ 的形式，若当 $\lim_{k \rightarrow +\infty} x_k = x^*$ ，则 $x^* = f(x^*)$ ，则此时不动点迭代法收敛。
- **不动点迭代法的整体收敛性**：
 - **映内性**：若 $\forall x \in [a, b]$ 有 $f(x) \in [a, b]$ ，则 $[a, b]$ 中一定存在 $f(x)$ 的不动点。
 - **压缩性**：在映内性满足的情况下，若 $\exists L \in (0, 1)$ 使得 $|f(x) - f(y)| \leq L|x - y|$ 对任意 x, y 成立，则映内性所说的不动点是唯一的，因此 $f(x)$ 确认能收敛到这个不动点。
 - **推论**：在映内性满足的情况下，若 $\exists L \in (0, 1)$ 使得 $|f'(x)| \leq L$ 对任意 $x \in [a, b]$ 成立，则 $f(x)$ 也必然存在唯一的不动点，从而收敛到这一不动点。
 - 显然，一般使用推论证明整体收敛性，不用压缩性证明。（参考第九章第1节“Lipschitz条件”）
- **不动点迭代法的局部收敛性**：
 - **局部收敛**：若已有 x^* ，且确认 $f'(x)$ 在 x^* 附近的邻域内连续，满足 $|f'(x^*)| < 1$ ，则不动点迭代法是局部收敛的。
 - **阶**：给定式子 $\lim_{k \rightarrow +\infty} \frac{f^{(p)}(x^*)}{p!} \neq 0$ ，满足该式的最小正整数 p 称为局部收敛阶数。
 - 显然，计算阶数时可以忽略 $p!$ ，直接求 f 在 x^* 处的导数、二阶导数……直到不等于0为止。

2. 不动点迭代法的加速方法

- **Aitken加速法**：从迭代的第三项 x_{k+2} 起进行修正：
$$x_i = \frac{x_i x_{i-2} - x_{i-1}^2}{x_i + x_{i-2} - 2x_{i-1}}。$$
- **Steffensen加速法**：对于不动点迭代法 $x = \varphi(x)$ ，Steffensen加速法为：
$$\begin{cases} m = \varphi(x_k) \\ n = \varphi(m) \\ x_{k+1} = x_k - \frac{(n-m)^2}{n-2m+x_k} \end{cases}。$$

- Steffensen加速法至少能加速到二阶。
- Steffensen加速法有时能将不收敛的不动点迭代法加速为收敛。

3. Newton迭代法

此处待求解的方程变为 $f(x) = 0$ 而非 $f(x) = x$ 。

- (带导数的) **Newton迭代法**： $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$
 - 若要求的根为重根，则Newton迭代法是线性的。否则，Newton迭代法是至少二阶的。
- **不带导数的Newton迭代法**（割线法）： $x_{n+1} = x_n - f(x_n) \frac{(x_n - x_{n-1})}{f(x_n) - f(x_{n-1})}$ 。这个方法需要两个初值。
- **改进的Newton迭代法**： $x_{n+1} = x_n - m \frac{f(x_n)}{f'(x_n)}$ ，式中 m 需要按照原方程情况进行指定。

第五章 矩阵特征值问题

本章讨论的矩阵特征值均用 λ 表示，其中 λ_1 是主特征值（最大的特征值），其余特征值按 λ_2 、 λ_3 、...、 λ_n 从大到小排列。

1. 幂法

- **幂法**：对任给 v_0 ，有
$$\begin{cases} z^{(k)} = Av^{(k-1)} \\ m^{(k)} = \max(z^{(k)}) \\ v^{(k)} = \frac{z^{(k)}}{m^{(k)}} \end{cases}$$
，其中 $m^{(k)}$ 收敛到主特征值 λ_1 ， $v^{(k)}$ 收敛到 λ_1 对应的特征向量。
- **幂法的收敛速度**：用 $\frac{\lambda_2}{\lambda_1}$ 表示。 λ_2 越接近 λ_1 ，收敛越慢。
- **幂法的Aitken加速**： $\lambda_1^{(k)} = \frac{m_k m_{k+2} - m_{k+1}^2}{m_{k+2} - 2m_{k+1} + m_k}$ ，这个算法比 $m^{(k)}$ 收敛得更快。
- **逆幂法**：
$$\begin{cases} A'z^{(k)} = v^{(k-1)} \\ m^{(k)} = \max(z^{(k)}) \\ v^{(k)} = \frac{z^{(k)}}{m^{(k)}} \end{cases}$$
，其中第一步需要解方程， $m^{(k)}$ 收敛到最小特征值 λ_n 的倒数， $v^{(k)}$ 收敛到 λ_n 对应的特征向量。
 - **逆幂法的收敛速度**：用 $\frac{\lambda_n}{\lambda_{n-1}}$ 表示。 λ_n 越接近 λ_{n-1} ，收敛越慢。
- **原点位移的逆幂法**：
$$\begin{cases} (A' - qI)z^{(k)} = v^{(k-1)} \\ m^{(k)} = \max(z^{(k)}) \\ v^{(k)} = \frac{z^{(k)}}{m^{(k)}} \end{cases}$$
，其中第一步需要解方程。当 q 最接近某一特征值 λ_i 时， $m^{(k)}$ 收敛到 $\frac{1}{\lambda_i - q}$ ， $v^{(k)}$ 收敛到 λ_i 对应的特征向量。
 - 若已知 λ_i 对应的特征向量近似为 x ，可直接取 $q = \frac{(Ax, x)}{(x, x)}$ 。

2. Householder矩阵的求法

- **一般情况下的Householder矩阵**：对任意 n 维向量 x ，总存在对称正交矩阵 P ，使得 Px 的计算结果为 n 维向量 $v = (v_1, \dots, v_j = -\text{sgn}(x_j)\alpha, v_{j+1} = 0, \dots, v_k = 0, v_{k+1}, \dots, v_n)$ （相当于将任意变量变换为有一段连续为0的向量，一般记为第 $j+1$ 到 k 项为0， $1 \leq j < k, j < k \leq n$ ）。这样的矩阵 P 称为Householder矩阵。

- **一般情况下的Householder矩阵求法**：

$$\begin{cases} \alpha = \sqrt{x_j^2 + x_{j+1}^2 + \dots + x_n^2} \\ u = [0, \dots, 0, x_j + \text{sgn}(x_j)\alpha, x_{j+1}, \dots, x_k, 0, \dots, 0]^T \\ P = I - \frac{2uu^T}{\|u\|_2^2} \end{cases}$$

- $\|u\|_2^2$ 的简便算法： $\|u\|_2^2 = 2\alpha(\alpha + x_j)$ 。
- 如此计算出的Householder矩阵 P ，从 j 行 j 列到 k 行 k 列仍为Householder矩阵（记为 \tilde{P} ），其余部分则为单位矩阵，如下图所示。

$$P = I - \frac{2uu^T}{\|u\|_2^2} = \begin{pmatrix} I_{j-1} & & \\ & \tilde{P} & \\ & & I_{n-k} \end{pmatrix}$$

- **等式右边为矩阵的Householder矩阵求法**：当等式变为 $Px = V$ 时，若矩阵第一列 v 仍满足上述的条件，则仍可按上述方法求取Householder矩阵 P 。
- **特殊情况下的Householder矩阵求法**：当计算结果为 n 维向量 $v = [v_1, 0, \dots, 0]^T$ （即从第二项开始全为0）时，Householder矩阵的求法变为如下形式：

$$\begin{cases} v_1 = -\text{sgn}(x_1)\|x\|_2 \\ u = [x_1 + \text{sgn}(x_1)\|x\|_2, x_2, \dots, x_n]^T \\ \beta = \frac{1}{\|x\|_2 + |x_1| \cdot \|x\|_2} \\ P = I - \beta uu^T \end{cases}$$

3. 上Hessenberg矩阵的求法

- **上Hessenberg矩阵**：次对角线以下元素均为0的矩阵，如下图所示。

$$B = \begin{bmatrix} * & * & \dots & * & * \\ * & * & \dots & * & * \\ & * & \dots & * & * \\ & & \ddots & \vdots & \vdots \\ & & & * & * \end{bmatrix}$$

- **上Hessenberg矩阵的求法**：

- 将待变换成上Hessenberg矩阵的矩阵 A ，取第一列 v_1 ，寻找Householder矩阵 $P_1 v_1 = v'_1$ 使得 v'_1 形如 $(*, *, 0, \dots, 0)^T$ 。
- 计算 $H_1 = P_1 A P_1$ 。如 H_1 还不是上Hessenberg矩阵，则除去 A 最左上角的一行一列，对右下角的分块 A_2 ，取新的第一列 v_2 ，寻找 $n-1$ 维Householder矩阵 $P'_2 v_2 = v'_2$ 使得 v'_2 形如 $(*, *, 0, \dots, 0)^T$ ，构造 n 维Householder矩阵 $P_2 = \begin{bmatrix} I & \\ & P'_2 \end{bmatrix}$ 。

- 计算 $H_2 = P_2 H_1 P_2$ 。如 H_2 还不是上 Hessenberg 矩阵，则再除去 A_2 最左上角的一行一列，对右下角的分块 A_3 ，取新的第一列 v_3 ，寻找 $n-2$ 维 Householder 矩阵 $P'_3 v_3 = v'_3$ 使得 v'_3 形如 $(*, *, 0, \dots, 0)^T$ ，构造 n 维矩阵 $P_3 = \begin{bmatrix} I & \\ & P'_3 \end{bmatrix}$ 。
-
- 直到 H_k 为上 Hessenberg 矩阵时，结束。

4. QR分解

- QR分解**：将非奇异矩阵 A 分解为 $A = QR$ ，其中 Q 为正交矩阵， R 为对角线均为正数的上三角阵。
- QR分解方法**（类似于上 Hessenberg 矩阵的求法）：
 - 将待分解的矩阵 A ，取第一列 v_1 ，寻找 $P_1 v_1 = v'_1$ 使得 v'_1 形如 $(*, 0, \dots, 0)^T$ 。
 - 计算 $R_1 = P_1 A$ 。如 R_1 还不是上三角矩阵，则除去 A 最左上角的一行一列，对右下角的分块 A_2 ，取新的第一列 v_2 ，寻找 $n-1$ 维矩阵 $P'_2 v_2 = v'_2$ 使得 v'_2 形如 $(*, 0, \dots, 0)^T$ ，构造 n 维矩阵 $P_2 = \begin{bmatrix} I & \\ & P'_2 \end{bmatrix}$ 。
 - 计算 $R_2 = P_2 P_1 A$ 。如 R_2 还不是上三角矩阵，则除去 A 最左上角的一行一列，对右下角的分块 A_3 ，取新的第一列 v_3 ，寻找 $n-2$ 维矩阵 $P'_3 v_3 = v'_3$ 使得 v'_3 形如 $(*, 0, \dots, 0)^T$ ，构造 n 维矩阵 $P_3 = \begin{bmatrix} I & \\ & P'_3 \end{bmatrix}$ 。
 -
 - 直到 $R_k = P_k P_{k-1} \dots P_1 A$ 为上三角矩阵时，结束。此时有：
 - $Q = (P_1 P_2 \dots P_n)^{-1} = (P_1 P_2 \dots P_n)^T = P_k^T P_{k-1}^T \dots P_1^T$ 。
 - $R = P_k P_{k-1} \dots P_1 A$ 。
 - 有时希望对角线为正，因此需要取合适的对角矩阵 D ，使 $\bar{Q} = QD$ 、 $\bar{R} = D^{-1}R$ 对角线为正。

- 示例：求 $A = \begin{bmatrix} 1 & 1 & 1 \\ 2 & -1 & -1 \\ 2 & -4 & 5 \end{bmatrix}$ 的 QR 分解，使 Q 和 R 对角线均为正。

解答：

- 先找到 P 使 $R = PA$ 为上三角阵：

- $k = -3, u = [4, 2, 2]^T, \beta = \frac{1}{12}$

- $P = I - \beta u u^T = \begin{bmatrix} -\frac{1}{3} & -\frac{2}{3} & -\frac{2}{3} \\ -\frac{2}{3} & \frac{2}{3} & -\frac{1}{3} \\ -\frac{2}{3} & -\frac{1}{3} & \frac{2}{3} \end{bmatrix}$ ，此时 $PA = \begin{bmatrix} -3 & 3 & -3 \\ 0 & 0 & -3 \\ 0 & -3 & 3 \end{bmatrix}$ 。

- 注意到第二列对角线以下仍然不是 0，因此再对 $A_0 = \begin{bmatrix} 0 & -3 \\ -3 & 3 \end{bmatrix}$ 找 P_0 ，使 $R_0 = P_0 A_0$ 为上三角阵。

- $k_0 = -3, u_0 = [3, -3]^T, \beta_0 = \frac{1}{9}$ 。

- $P_0 = I - \beta_0 u_0 u_0^T = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ ，此时 $P_0 A_0 = \begin{bmatrix} -3 & 3 \\ 0 & -3 \end{bmatrix}$ 。

■ 于是取 $P_1 = \begin{bmatrix} 1 & & \\ & P_0 & \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$ ，此时：

■ $R = P_1 * P * A = \begin{bmatrix} -3 & 3 & -3 \\ 0 & -3 & 3 \\ 0 & 0 & -3 \end{bmatrix}$ 。

■ $Q = \begin{bmatrix} -\frac{1}{3} & -\frac{2}{3} & -\frac{2}{3} \\ -\frac{2}{3} & -\frac{1}{3} & \frac{2}{3} \\ -\frac{2}{3} & \frac{2}{3} & -\frac{1}{3} \end{bmatrix}$ 。

但这是对角线非正的。因此取 $\bar{D} = \text{diag}(-1, -1, -1)$ ：

■ $\bar{Q} = Q\bar{D} = \begin{bmatrix} \frac{1}{3} & \frac{2}{3} & \frac{2}{3} \\ \frac{2}{3} & \frac{1}{3} & -\frac{2}{3} \\ \frac{2}{3} & -\frac{2}{3} & \frac{1}{3} \end{bmatrix}$ 。

■ $\bar{R} = \bar{D}^{-1}R = \begin{bmatrix} 3 & -3 & 3 \\ 0 & 3 & -3 \\ 0 & 0 & 3 \end{bmatrix}$ 。

检验得 $\bar{Q}\bar{R} = A$ ，符合题意。

- 基本QR迭代方法： $\begin{cases} A_k = Q_k R_k \\ A_{k+1} = R_k Q_k \end{cases}$ ，该法中 A_k 会收敛与上三角阵，对角元素为 A 特征值。

第六章 插值法

1. Lagrange插值

- n 次Lagrange插值多项式： $L_n(x) = \sum_{k=0}^n f(x_k) [\prod_{j=0, j \neq k}^n (\frac{x-x_j}{x_k-x_j})]$ 。
 - 显然， n 次Lagrange插值多项式是 n 次多项式，对应 $n-1$ 个插值点，在这些插值点上 $L_n(x) = f(x)$ 。

2. 均差与Hermite插值

- 均差： $f[x_0, x_1, \dots, x_k] = \frac{f[x_0, x_1, \dots, x_{k-1}] - f[x_1, x_2, \dots, x_k]}{x_k - x_0}$ 。又称差商。
- 均差表：一个函数的自变量和因变量可以画成形如下图的均差表。

3. 设 $f(x) \in C^n[a, b]$, $x_j \in [a, b], j = 0, 1, \dots, n$ 为相异节点，那么

$$f[x_0, x_1, \dots, x_n] = \frac{1}{n!} f^{(n)}(\xi), \xi \in (a, b)$$

均差表（差商表）

x_k	$f(x_k)$	一阶均差	二阶均差	三阶均差
x_0	$f(x_0)$			
x_1	$f(x_1)$	$f[x_0, x_1]$		
x_2	$f(x_2)$	$f[x_1, x_2]$	$f[x_0, x_1, x_2]$	
x_3	$f(x_3)$	$f[x_2, x_3]$	$f[x_1, x_2, x_3]$	$f[x_0, x_1, x_2, x_3]$

- **Hermite插值**：

$$H_n(x) = f(x_0) + f[x_0, x_1](x - x_0) + f[\cdots + f[x_0, x_1, \cdots, x_n] \prod_{k=0}^{n-1} (x - x_k)。$$
 - 简记：即只用 x 分别减对应每个 f 的括号里的前面几项，再相乘。最后一项不减不乘。
 - **次数**：即 n 。
 - 显然， n 次Hermite插值需要 n 对 $(x, f(x))$ 进行均差求解。
- **重节点**：如果提供的 $(x, f(x))$ 不够要求的次数，那么可以使用重节点。在带有重节点的均差表中，如果出现 $\frac{f(x_i) - f(x_i)}{x_i - x_i}$ 的情况，则视为求一阶导 $f'(x_i)$ 。其余情况类推。
 - 使用重节点均差的Hermite插值，如 $f[x_0, x_0, x_1, x_1]$ ，后面就需要乘 $(x - x_0)^2(x - x_1)$ 。这仍然符合上面的简记。

3. 三次样条插值

- **三次样条函数**：对函数 $f(x)$ 进行三次样条插值得到的函数 $S(x)$ ，一般为满足如下性质的分段函数。
 - 每个分段上均为最高次数相同的多项式函数 $\sum_{k=0}^n a_k x^k$ 。
 - 在每个分段点上，函数值、一阶导数值、二阶导数值相等。
 - 如此时约束条件仍然不够，使得插值得到的系数只能求出关系而不能求值，则可以补足条件。
- 常见的补足条件方法：
 - **I型边界条件**：对分段的开头 x_0 与结尾 x_n 这两处，要求 $S'(x) = f'(x)$ 。
 - **II型边界条件**：对分段的开头 x_0 与结尾 x_n 这两处，要求 $S''(x) = f''(x)$ 。
 - **自然边界条件**：对分段的开头 x_0 与结尾 x_n 这两处，要求 $S''(x) = 0$ 。
 - 此时的三次样条函数又称自然样条函数。
 - **非扭矩边界条件**：对每个分段上的函数 $\sum_{k=0}^n a_k x^k$ ，有 $a_n = 0$ 。

第七章 函数逼近

1. 正交多项式

- **正交多项式**：设有函数 $f(x)$ ， $g(x)$ ，其定义域均为 $[a, b]$ ，则若找到 $\rho(x)$ 满足 $\int_a^b \rho(x) f(x) g(x) dx = 0$ ，则 f 与 g 在 $[a, b]$ 上互为基于权函数 ρ 的正交多项式。

2. 最小二乘法

- **线性拟合的最小二乘法**：设有 n 维向量 x 和 n 维向量 y 需要拟合成直线 $y = a_0 + a_1 x$ ，则方法如下：
 - 定义每个元素全为1的 n 维向量 c 。
 - $\begin{bmatrix} (c, c) & (x, c) \\ (c, x) & (x, x) \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} (c, y) \\ (x, y) \end{bmatrix}$ ，其中 $(,)$ 表示向量内积（点乘）。
 - 由于 c 、 x 、 y 均已知，因此相当于求解 $Ax = b$ 。解出 a_0 、 a_1 即为所求。

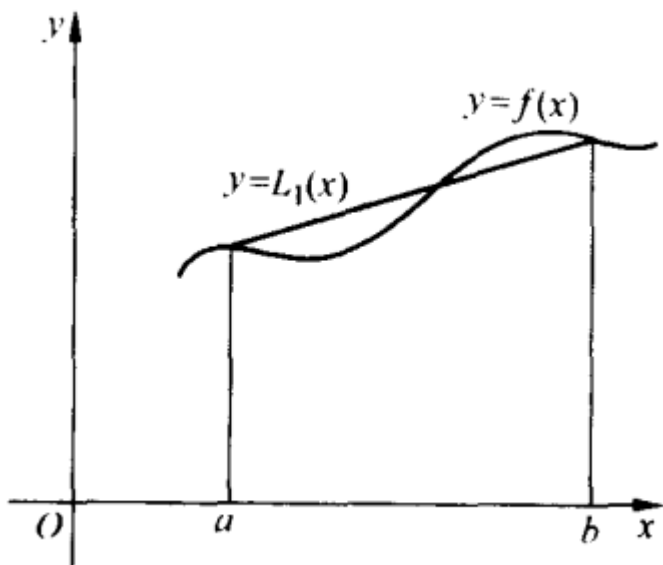
- **指数拟合的最小二乘法**：设有 n 维向量 x 和 n 维向量 y 需要拟合成指数曲线 $y = be^{ax}$ ，，则方法如下：
 - 两边取 \ln ： $\ln y = \ln b + ax$
 - 于是记 $z = \ln y$ ， $a_0 = \ln b$ ， $a_1 = a$ ，变成 $z = a_0 + a_1 x$ ，按线性拟合的最小二乘法求解即可。

第八章 数值积分

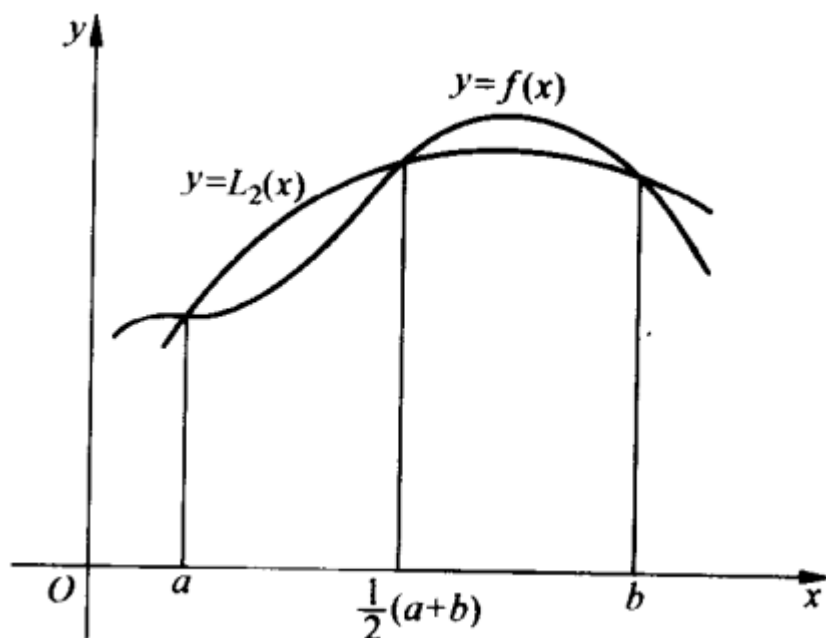
本章讨论的数值积分均形如 $I(f) = \int_a^b f(x)dx \approx \int_a^b L_1(x)dx$ 。

1. Newton-Cotes型数值积分

- **插值型公式**：设对 $f(x)$ 进行 n 次Lagrange插值，得到插值多项式 $L_n(x) = \sum_{k=0}^n f(x_k) [\prod_{j=0, j \neq k}^n (\frac{x-x_j}{x_k-x_j})]$ ，则插值型公式定义为 $I(f) \approx \sum_{k=0}^n f(x_k) [\prod_{j=0, j \neq k}^n (\frac{x-x_j}{x_k-x_j})]$ 。
 - 下述的梯形、Simpson、Newton-Cotes求积公式，事实上全部为插值型公式。
- **代数精度**：若某个插值型求积公式 $I(f) \approx \int_a^b L_1(x)dx$ 能在 $f(x) = 1, x, x^2, \dots, x^p$ 严格取等，而在 $f(x) = x^{p+1}$ 只能取约等，则该求积公式的代数精度为 p 次。
- **梯形公式**： $I(f) \approx \frac{b-a}{2} [f(a) + f(b)]$ ，如下图所示。



- **梯形公式的代数精度**：1。
- **Simpson公式**： $I(f) = \frac{b-a}{6} [f(a) + f(b) + 4f(\frac{b+a}{2})]$ ，如下图所示。



- Simpson公式的代数精度：3。
- Newton-Cotes求积公式： $I(f) \approx (b-a) \sum_{k=0}^n C_k^{(n)} f(x_k)$ ，其中 $C_k^{(n)}$ 的值见下表。

n	$C_k^{(n)}$				
1	$\frac{1}{2}$				
2	$\frac{1}{6}$	$\frac{4}{6}$			
3	$\frac{1}{8}$	$\frac{3}{8}$			
4	$\frac{7}{90}$	$\frac{16}{45}$	$\frac{2}{15}$		
5	$\frac{19}{288}$	$\frac{25}{96}$	$\frac{25}{144}$		
6	$\frac{41}{840}$	$\frac{9}{35}$	$\frac{9}{280}$	$\frac{34}{105}$	
7	$\frac{751}{17\ 280}$	$\frac{3577}{17\ 280}$	$\frac{1323}{17\ 280}$	$\frac{2989}{17\ 280}$	
8	$\frac{989}{28\ 350}$	$\frac{5888}{28\ 350}$	$\frac{-928}{28\ 350}$	$\frac{10\ 496}{28\ 350}$	$\frac{-4540}{28\ 350}$

- $C_k^{(n)}$ 满足 $C_k^{(n)} = C_n^{(n-k)}$ 。
- $\{x_0 = a, x_1, \dots, x_n = b\}$ 等分了区间 $[a, b]$ 。
- $n = 1$ 时，Newton-Cotes求积公式退化为梯形公式； $n = 2$ 时退化为Simpson公式， $n = 3$ 时称为Simpson 3/8公式； $n = 4$ 时称为Boole公式（或Cotes公式）。
- Newton-Cotes求积公式的稳定性： $n \leq 7$ 的Newton-Cotes都是稳定的，因为此时所有的 C 均为正数。
- 开型Newton-Cotes求积公式：即令 $\{a, x_0, x_1, \dots, x_n, b\}$ 等分 $[a, b]$ 的Newton-Cotes公式。
 - 中点公式：取 $n = 0$ ，有 $I(f) \approx (b-a)f(\frac{a+b}{2})$ 。
 - 两点公式：取 $n = 1$ ，有 $I(f) \approx \frac{b-a}{2}[f(x_0) + f(x_1)]$ 。
 - 三点公式：取 $n = 2$ ，有 $I(f) \approx \frac{b-a}{3}[2f(x_0) - f(x_1) + 2f(x_2)]$ 。

2. 复合求积公式

- **复合梯形公式**：将 $[a, b]$ 等分为 $\{x_0 = a, x_1, \dots, x_n = b\}$ ，并对每个部分使用梯形公式求积分，得到 $I_n(f) \approx \frac{b-a}{2n} [f(a) + 2 \sum_{k=1}^{n-1} f(x_k) + f(b)]$ ，即为复合梯形公式。
- **复合Simpson公式**：将 $[a, b]$ 等分为 $\{x_0 = a, x_1, \dots, x_n = b\}$ ，并对每个部分使用Simpson公式求积分，得到 $I_n(f) \approx \frac{b-a}{6n} [f(a) + 2 \sum_{k=1}^{n-1} f(x_k) + f(b) + 4 \sum_{k=1}^n f(\frac{x_k+x_{k-1}}{2})]$ ，即为复合Simpson公式。
- **Romberg算法**：
 1. $h = b - a$, $T(0, 0) = \frac{h}{2}(f(a) + f(b))$ ，转2。
 2. 将区间 $[a, b]$ 分半， $T(1, 0) = I_2(f)$, $T(1, 1) = \frac{4T(1, 0) - T(0, 0)}{4^1 - 1}$ ， $1 \rightarrow j$ ，转4。
 3. 对区间作 2^j 等分， $T(j, 0) = I_{2^j}(f)$, $T(j, k) = \frac{4^k T(j, k-1) - T(j-1, k-1)}{4^k - 1}$, $k = 1, 2, \dots, j$ ，求出 $T(j, j)$ ，转4。
 4. $|T(j, j) - T(j-1, j-1)| < \varepsilon$ ，则 $T(j, j)$ 即为所求；否则 $j+1 \rightarrow j$ ，转3。
 - 式中 $I_n(f)$ 使用复合梯形公式求解。
 - Romberg算法的计算结果可以打印为 T -表，如下表所示。

$T(0,0)$				
$T(1,0)$	$T(1,1)$			
$T(2,0)$	$T(2,1)$	$T(2,2)$		
$T(3,0)$	$T(3,1)$	$T(3,2)$	$T(3,3)$	
...

3. Gauss型求积公式

- **最高精度**：对于有 $n+1$ 个节点的求积公式，其最高代数精度为 $2n+1$ 。
- **Gauss型求积公式**：达到了最高代数精度的求积公式。
- **Gauss型求积公式的求法**：若有Gauss型求积公式 $I(f) = \int_a^b \rho(x)f(x)dx$ ，希望使用 $n+1$ 个节点得到 $2n+1$ 精度，则求法如下：

◦ 设 $\phi(x) = x^{n+1} + ax^n + bx^{n-1} + \dots + z$ ，则联立

$$\begin{cases} \int_a^b \rho(x)\phi(x)dx - 0 \\ \int_a^b x\rho(x)\phi(x)dx - 0 \\ \dots \\ \int_a^b x^n \rho(x)\phi(x)dx - 0 \end{cases}, \text{由此解出}$$

a, b, \dots, z 。

- 即是要求 $\phi(x)$ 同时与多项式序列 $\{a_{n+1}x^{n+1}, a_n x^n, \dots, a_0 x^0\}$ 在给定的区间内基于权函数 $\rho(x)$ 正交。

- 解 $\phi(x) = 0$ ，得到 x_1, x_2, \dots, x_{n+1} 共 $n+1$ 个解，于是知

$$I(f) = \int_a^b \rho(x)f(x)dx \approx \sum_{k=1}^{n+1} A_k f(x_k)。$$

- 由于精度为 $2n+1$ ，则显然可以根据代数精度的求法，令 $I(f)$ 在 $f(x) = 1, x, \dots, x_n$ 严格取等，解出 A_1, A_2, \dots, A_{n+1} 。此时即有Gauss型求积公式 $I(f) \approx \sum_{k=1}^{n+1} A_k f(x_k)$ 。

- 从以上过程中可以看出, 只要给定求积区间 $[a, b]$ 和权函数 $\rho(x)$, 则不论对于什么样的 $f(x)$, 都能解出唯一一组 $\{A_k\}$, 使得 $I(f) \approx \sum_{k=1}^{n+1} A_k f(x_k)$ 。

○ 示例:

例 8.5.2 确定求积公式

$$\int_0^1 \sqrt{x} f(x) dx \approx A_0 f(x_0) + A_1 f(x_1)$$

中的系数 A_0, A_1 及节点 x_0, x_1 , 使该求积公式具有最高代数精度。

解 具有最高代数精度的求积公式为 Gauss 求积公式. 节点 x_0, x_1 为 $[a, b] = [0, 1]$ 上以权函数 $\rho(x) = \sqrt{x}$ 的两次正交多项式的零点. 不妨假定 $\phi_2(x)$ 的首项系数为 1 (不改变零点).

设 $\phi_2(x) = x^2 + ax + b$, 如果有

$$\int_0^1 \sqrt{x} \phi_2(x) dx = 0, \quad \int_0^1 \sqrt{x} \cdot x \phi_2(x) dx = 0, \quad (8.5.8)$$

那么 $\phi_2(x)$ 在 $[0, 1]$ 上以权函数 $\rho(x) = \sqrt{x}$ 与零次和一次多项式正交. 所以 $\phi_2(x)$ 是 $[a, b]$ 上, 以权函数 $\rho(x) = \sqrt{x}$ 的二次正交多项式 (首项系数为 1).

由 (8.5.8) 式得出 $a = -\frac{10}{9}, b = \frac{5}{21}$, 从而得出

$$\phi_2(x) = x^2 - \frac{10}{9}x + \frac{5}{21}.$$

解 $\phi_2(x) = 0$, 有

$$x_0 = \frac{35 - 2\sqrt{70}}{63}, \quad x_1 = \frac{35 + 2\sqrt{70}}{63},$$

即 $x_0 = 0.289\ 949, x_1 = 0.821\ 162$.

由于两个节点的 Gauss 求积公式具有 3 次代数精度, 因此对于 $f(x) = 1, x$, 求积公式准确成立. 即

$$f(x) = 1, \quad \text{有} \quad \int_0^1 \sqrt{x} dx = A_0 + A_1,$$

$$f(x) = x, \quad \text{有} \quad \int_0^1 \sqrt{x} \cdot x dx = A_0 x_0 + A_1 x_1.$$

定积分计算后得

$$A_0 + A_1 = \frac{2}{3}, \quad x_0 A_0 + x_1 A_1 = \frac{2}{5}.$$

由此解得 $A_0 = 0.277\ 556, A_1 = 0.389\ 111$. 求积公式为

$$\int_0^1 \sqrt{x} f(x) dx \approx 0.277\ 556 f(0.289\ 949) + 0.389\ 111 f(0.821\ 162).$$

- Gauss-Legendre 求积公式: $\int_{-1}^1 f(x) dx = \sum_{k=0}^n A_k f(x_k)$, 式中对不同的 n 有不同的 A_k , 如下图所示。

表 8.4

n	x_k	A_k	n	x_k	A_k
0	0	2	5	$\pm 0.932\ 469\ 514\ 2$	0.171 324 492 4
1	$\pm 0.577\ 350\ 269\ 2$	1		$\pm 0.661\ 209\ 386\ 5$	0.360 761 573 0
2	$\pm 0.774\ 596\ 669\ 2$	0.555 555 555 6		$\pm 0.238\ 619\ 186\ 1$	0.467 913 934 6
	0	0.888 888 888 9	6	$\pm 0.949\ 107\ 912\ 3$	0.129 484 966 2
3	$\pm 0.861\ 136\ 311\ 6$	0.347 854 845 1		$\pm 0.741\ 531\ 185\ 6$	0.279 705 391 5
	$\pm 0.339\ 981\ 043\ 6$	0.652 145 154 9		$\pm 0.405\ 845\ 151\ 4$	0.381 830 050 5
				0	0.417 959 183 7
4	$\pm 0.906\ 179\ 845\ 9$	0.236 926 885 1	7	$\pm 0.960\ 289\ 856\ 5$	0.101 228 536 3
	$\pm 0.538\ 469\ 310\ 1$	0.478 628 670 5		$\pm 0.796\ 666\ 477\ 4$	0.222 381 034 5
	0	0.568 888 888 9		$\pm 0.525\ 532\ 409\ 9$	0.313 706 645 9
				$\pm 0.183\ 434\ 642\ 5$	0.362 683 783 4

- 当求积区间不为 $[-1, 1]$ 而为 $[a, b]$ 时, 应将 x 变换为 $t = \frac{2x-b-a}{b-a}$, 将 $\int_a^b f(x)dx$ 变为 $\frac{2}{b-a} \int_{-1}^1 f(t)dt$ 。
- Gauss-Chebyshev求积公式: $\int_{-1}^1 \frac{f(x)}{\sqrt{1-x^2}} dx = \sum_{k=0}^n \frac{\pi}{n+1} f(\cos(\frac{2k+1}{2n+2}\pi))$ 。
- 类似地, 当求积区间不为 $[-1, 1]$ 而为 $[a, b]$ 时, 应将 x 变换为 $t = \frac{2x-b-a}{b-a}$ 。

第九章 常微分方程数值解法

本章讨论的常微分方程均形如 $\begin{cases} y' = f(x, y) \\ y_0 = \alpha \end{cases}$, 迭代步长一律为 h , 即 $x_k = x_0 + kh$, $y_k = y(x_0 + kh)$ 。

1. Lipschitz条件

- Lipschitz条件: 若存在常数 L 使任意 y_1, y_2 满足 $|f(y_1) - f(y_2)| \leq L(y_1 - y_2)$, 则 f 满足Lipschitz条件。
- 加强的判别条件: $\forall (x, y)$ 满足 $|\frac{\partial f}{\partial y}(x, y)| \leq L$ 的函数 $f(x, y)$ 一定对 y 满足Lipschitz条件。
 - 这个条件是充分不必要条件, 反推不一定成立。

2. 一阶单步方法

- Euler法
 - 显式Euler法: $\begin{cases} y_{n+1} = y_n + hf(x_n, y_n) \\ y_0 = \alpha \end{cases}$;
 - 隐式Euler法: $\begin{cases} y_{n+1} = y_n + hf(x_{n+1}, y_{n+1}) \\ y_0 = \alpha \end{cases}$; 这个方法在给定 y_n 时并不能直接求解 y_{n+1} , 而是还要解方程, 因此一般在近似解 y_n 已知时使用。
 - Euler法的收敛性: 若能找到使 f 对 y 满足Lipschitz条件的 L , 则 $h < \frac{1}{L}$ 可使Euler法收敛。
- 梯形方法
 - $\begin{cases} y_{n+1} = y_n + \frac{h}{2}[f(x_n, y_n) + f(x_{n+1}, y_{n+1})] \\ y_0 = \alpha \end{cases}$, 即对显式、隐式Euler方法做了折中。
 - 梯形方法的收敛性: 若能找到使 f 对 y 满足Lipschitz条件的 L , 则 $h < \frac{2}{L}$ 可使梯形方法收敛。
 - 梯形方法比显式Euler法一般更快。
- 预估-校正方法
 - 改进Euler法: $\begin{cases} y_{n+1} = y_n + \frac{h}{2}[f(x_n, y_n) + f(x_{n+1}, y_n + hf(x_n, y_n))] \\ y_0 = \alpha \end{cases}$ 。
 - 改进Euler法比显式Euler法一般更精确。
- 中点公式: $\begin{cases} y_{n+1} = y_n + h(f(x_n + \frac{h}{2}, y_n + \frac{h}{2}f(x_n, y_n))) \\ y_0 = \alpha \end{cases}$

3. 截断误差

- **局部截断误差**：对方法 $\sum_{k=0}^p y_{n+k} = \sum_{k=0}^q y'_{n+k}$ 中的每项进行Taylor展开，得到的式子 $T_{n+1} = \sum_{i=0}^{+\infty} \beta h^i y_n^{(i)}$ 即为局部截断误差。
 - 此处所用的Taylor展开公式：
$$\begin{cases} y_{n+k} = y(x+kh) = \sum_{i=0}^{+\infty} \frac{(kh)^i}{i!} y_n^{(i)} \\ y'_{n+k} = f(x+kh) = \sum_{i=0}^{+\infty} \frac{(kh)^i}{i!} y_n^{(i+1)} \end{cases}$$
- **主局部截断误差**： T_{n+1} 的前面几项往往是0。对第一个非零项，称之为主局部截断误差。
- **阶**：取主局部截断误差中 h 的次数 p ，有 $p-1$ 为该方法的阶。
 - 示例：梯形公式的 $T_{n+1} = -\frac{h^3}{12} y^{(3)}(x_n) + O(h^4)$ ，因此是二阶的。
 - 显式Euler、隐式Euler法都是一阶的，梯形公式是二阶的。
 - 改进Euler公式需要使用二维Taylor展开（即需要考虑 f 对 x 和 y 求取偏导数、二阶偏导数等）。最终计算得一维部分是 $\frac{h^3}{6} y^{(3)}(x_n) + O(h^4)$ ，二维部分的 h 系数也为3，因此是二阶的。

4. 相容性、收敛性与绝对稳定性

- **相容性**：若把显式单步法写成 $y_{n+1} = \phi(x_n, y_n; h)$ 形式，并对所有的 h 都取0，发现 $y_{n+1} = f(x_n, y_n)$ ，则这个显式单步法是相容的。
- **收敛性**：若显式单步法 $y_{n+1} = \phi(x_n, y_n; h)$ 的阶数 ≥ 1 ，且 ϕ 满足对 y 的Lipschitz条件 $|\frac{\phi(x, y; h) - \phi(x, \bar{y}; h)}{y - \bar{y}}| \leq L$ ，则这个显式单步法是收敛的。
 - **相容性-收敛性推论**：若显式单步法的阶数 ≥ 1 ，则它相容且收敛。
- **绝对稳定性**：取 $y'(x) = f(x, y) = \lambda y$ ，则单步法可以写成 $y_{n+1} = E(\lambda h) y_n$ 。
 - **绝对稳定性的判别方法**：令 λh 满足 $|E(\lambda h)| < 1$ 即可。
 - 当显式单步法中的 f 不是 $f(x, y)$ 而是 $f(g(x), z(y))$ 时，则 λy 变成 $\lambda z(y)$ 。
 - 示例：求改进Euler方法 $y_{n+1} = y_n + \frac{h}{2} [f(x_n, y_n) + f(x_{n+1}, y_n + hf(x_n, y_n))]$ 的绝对稳定区间。

对于改进Euler方法有 $E(\lambda h) = 1 + \lambda h + \frac{(\lambda h)^2}{2}$

绝对稳定性条件为 $\left| 1 + \lambda h + \frac{(\lambda h)^2}{2} \right| < 1$

当 $\lambda h \in R$ 时，此条件等价于 $-1 < 1 + \lambda h + \frac{(\lambda h)^2}{2} < 1$

即 $-2 < \lambda h(1 + \frac{\lambda h}{2}) < 0$

考虑区间 $(-\infty, -2], (-2, 0), [0, +\infty)$

当 $\bar{h} = \lambda h \in (-2, 0)$ 时，有 $|E(\lambda h)| < 1$ 因此，改进Euler方法的绝对稳定性区间为 $(-2, 0)$ 。

未包括的内容：

- 第一章：

- 误差、有效数字、数值方法稳定性（送分题）
 - 第二章：
 - 误差分析（病态？）
 - 第六章：
 - Newton-Hermite插值（没写Newton插值）
 - 插值方法的余项
 - 分段低次插值
 - 三次样条插值（只写了三次样条函数的系数求法和常见边界条件）
 - 第七章：
 - 线性以及特殊非线性化成线性问题（最小二乘法？）
 - 第八章：
 - 梯形公式、Simpson公式以及相应的复合求积公式的余项（代数精度？）
 - 第九章：
 - Runge-Kutta方法
-