

Numerical Computing Methods

数值计算方法

李晓鹏
Li Xiao-Peng

电子与信息工程学院
School of Electronic and Information Engineering

Semester-I 25/26

逐次逼近法

① 基本概念

② 线性方程组的迭代法

③ 非线性方程的迭代法

引论

① 基本概念

② 线性方程组的迭代法

③ 非线性方程的迭代法

基本概念

逐次逼近法也称为迭代法，它是对所求问题建立一种规则，将所求问题转化为利用初值或已求出的元素计算后继元素，从而形成一个序列。该法在数值计算上有着广泛地应用。

由于逐次逼近法设计两个元素得逼近问题，所以，首先应明确非实数之间得距离以及极限过程得收敛性概念。

我们已知：

- 一维空间，距离原点的距离： $|x|$;
- 二维空间，距离原点的距离： $\sqrt{x^2 + y^2}$;
- 三维空间，距离原点的距离： $\sqrt{x^2 + y^2 + z^2}$;

有必要给出一种实用更广泛、应用更灵活且又具有普通长度所具有的特征的“度量”概念。

向量与矩阵的范数

定义: 设 \mathbb{R} 是一个数域, V 是一个线性空间。若 V 中的任一元素 x 都对应一个实数 $N(x)$, 即 $N(x)$ 是 V 上的实值函数, 且满足如下条件

- (1) $N(x) \geq 0$, 当且仅当 $x = \theta$ (零元素) 时 $N(x) = 0$; (正定性)
- (2) $N(\alpha x) = |\alpha|N(x), x \in \mathbb{R}$ (齐次性)
- (3) $N(x + y) \leq N(x) + N(y), x, y \in V$ (三角不等式)

则称 $N(x)$ 是 V 上的一个范数, 记为 $\|x\|$, 即 $N(x) = \|x\|$ 表示 x 的范数。

若在 V 中具有“乘法”运算, 则可增加一个条件:
对任意 $x, y \in V$, 则 $x \cdot y \in V$ 时, 满足如下条件

$$\|x \cdot y\| \leq \|x\| \|y\|$$

向量范数

设 $\mathbf{x} = [x_1, x_2, \dots, x_n]^T \in \mathbb{R}^n$, 常用的四种范数定义如下:

1. 向量的 ℓ_2 -范数, 也称欧氏范数, 记为

$$\|\mathbf{x}\|_2 = \langle \mathbf{x}, \mathbf{x} \rangle^{1/2} = \left(\sum_{i=1}^n x_i^2 \right)^{1/2}$$

或

$$\|\mathbf{x}\|_2 = \left(\sum_{i=1}^n \bar{x}_i x_i \right)^{1/2} \quad \mathbf{x} \in \mathbb{C}^n$$

其中, \bar{x}_i 是 x_i 的共轭复数。

若 $\mathbf{x} = [x_1, x_2, \dots, x_n]^T \in \mathbb{R}^n$, $\mathbf{y} = [y_1, y_2, \dots, y_n]^T \in \mathbb{R}^n$, 则

$$\mathbf{x}^T \cdot \mathbf{y} = \mathbf{y}^T \cdot \mathbf{x} = \sum_{i=1}^n y_i x_i$$

向量范数

2. 向量的 ∞ -范数, 也称最大范数, 记为

$$\|\mathbf{x}\|_{\infty} = \max_{1 \leq i \leq n} |x_i| = \max_{1 \leq i \leq n} \{|x_1|, |x_2|, \dots, |x_n|\}$$

3. 向量的 ℓ_1 -范数, 记为

$$\|\mathbf{x}\|_1 = \sum_{i=1}^n |x_i|$$

4. 向量的 ℓ_p -范数, 记为

$$\|\mathbf{x}\|_p = \left(\sum_{i=1}^n |x_i|^p \right)^{1/p}, \quad p \in [1, \infty)$$

上述四种常用的实值函数都满足正定性、齐次性和三角不等式。

矩阵的范数

如无特殊说明, 线性空间为 $V = \mathbb{R}^{n \times n}$ 。 V 中存在矩阵乘积运算, 因此定义矩阵范数应满足

$$\|A \cdot B\| \leq \|A\| \cdot \|B\|$$

定义: 设 $x \in \mathbb{R}^n$, $A \in \mathbb{R}^{n \times n}$, 且 $\|x\|_v$ 是一种向量范数, 则

$$\|A\|_v = \max_{x \neq 0} \frac{\|Ax\|_v}{\|x\|_v} = \max_{\|x\|_v=1} \|Ax\|_v$$

称为矩阵 A 的**算子范数**。

显然算子范数满足: $\|Ax\|_v \leq \|A\|_v \|x\|_v$

注意: 矩阵的算子范数是矩阵范数, 但是反之不一定成立。

矩阵的范数

矩阵算子范数的常用三种形式：

设 $\mathbf{A} = (a_{ij})_{n \times n}$

1. 矩阵的行范数：绝对值最大的行和

$$\|\mathbf{A}\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^n |a_{ij}|$$

2. 矩阵的列范数：绝对值最大的列和

$$\|\mathbf{A}\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}|$$

3. 矩阵的 ℓ_2 -范数：

$$\|\mathbf{A}\|_2 = \sqrt{\lambda_{\max}(\mathbf{A}^T \mathbf{A})}$$

其中， $\lambda_{\max}(\mathbf{A}^T \mathbf{A})$ 表示矩阵 $\mathbf{A}^T \mathbf{A}$ 的绝对值最大的特征值。

矩阵的范数

定义: 对于给定的向量范数 $\|\cdot\|_v$ 与矩阵范数 $\|\cdot\|_u$, 若对任何 $A \in \mathbb{R}^{n \times n}$, $x \in \mathbb{R}^n$, 都有

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

则称所给向量范数 $\|\cdot\|_v$ 与矩阵范数 $\|\cdot\|_u$ 相容。

4. 矩阵的 Frobenius-范数:

$$\|A\|_F = \left(\sum_{i=1}^n \sum_{j=1}^n |a_{ij}|^2 \right)^{1/2}$$

简称为 A 的 F -范数。

可以验证: F -范数满足范数定义中的 4 个条件, 但是 F -范数不是算子范数。

例题

例：已知矩阵

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 0 \\ -1 & 2 & -1 \\ 0 & 1 & 1 \end{bmatrix}$$

求 $\|\mathbf{A}\|_\infty$ 、 $\|\mathbf{A}\|_1$ 、 $\|\mathbf{A}\|_2$ 和 $\|\mathbf{A}\|_F$.

例题

例：已知矩阵

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 0 \\ -1 & 2 & -1 \\ 0 & 1 & 1 \end{bmatrix}$$

求 $\|\mathbf{A}\|_\infty$ 、 $\|\mathbf{A}\|_1$ 、 $\|\mathbf{A}\|_2$ 和 $\|\mathbf{A}\|_F$.

解

$$\|\mathbf{A}\|_1 = \max\{(1+1+0), (2+2+1), (0+1+1)\} = 5$$

$$\|\mathbf{A}\|_\infty = \max\{(1+2+0), (1+2+1), (0+1+1)\} = 4$$

$$\|\mathbf{A}\|_F = (1+4+1+4+1+1+1)^{1/2} = \sqrt{13} \approx 3.6056$$

例题

例：设

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 0 \\ -1 & 2 & -1 \\ 0 & 1 & 1 \end{bmatrix}$$

求 $\|\mathbf{A}\|_\infty$ 、 $\|\mathbf{A}\|_1$ 、 $\|\mathbf{A}\|_2$ 和 $\|\mathbf{A}\|_F$.

解

$$\mathbf{A}^T \mathbf{A} = \begin{bmatrix} 2 & 0 & 1 \\ 0 & 9 & -1 \\ 1 & -1 & 2 \end{bmatrix}$$

$$\det(\lambda I - \mathbf{A}^T \mathbf{A}) = \begin{vmatrix} \lambda - 2 & 0 & -1 \\ 0 & \lambda - 9 & 1 \\ -1 & 1 & \lambda - 2 \end{vmatrix} = \lambda^3 - 13\lambda^2 + 38\lambda - 25 = 0$$

解得 $\lambda_1 = 9.1428$, $\lambda_2 = 2.9211$, $\lambda_3 = 0.9361$ 。

因此

$$\|\mathbf{A}\|_2 = \sqrt{9.1428} \approx 3.0237$$

矩阵的范数

定义：设 $A \in \mathbb{R}^{n \times n}$ 的特征值为 $\lambda_i (i = 1, 2, \dots, n)$, 称

$$\rho(A) = \max_{1 \leq i \leq n} |\lambda_i|$$

为 A 的谱半径。注：对一切范数 $\|\cdot\|$, 都有

$$\rho(A) \leq \|A\|$$

定理：如果 $\|A\| < 1$, 则 $I + A$ 为非奇异矩阵, 且

$$\|(I + A)^{-1}\| \leq \frac{1}{1 - \|A\|}$$

其中, I 为单位矩阵, $\|\cdot\|$ 为矩阵的算子范数。

误差分析介绍

第一章中曾提到数值问题的性态会对问题的计算精度产生很大影响，但是如何判断数值问题的性态只是作了定性的描述，没有给出定量的判别方法。在此，我们以求解线性方程组为例，介绍如何度量方程组性态。

定义：如果线性方程组 $Ax = b$ 中， A 或 b 的元素的微小变化就会引起方程组解的巨大变化，则称该方程组为“病态”方程组， A 为“病态”矩阵；否则成该方程组为“良态”方程组， A 为“良态”矩阵。

在 $Ax = b$ 中，设 A 为非奇异矩阵， x 为精确解。若 b 有误差 δb ，则引起解变化，设为 $x + \delta x$ ，即

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

因此

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

取范数，得

$$\|\delta x\| = \|A^{-1}\delta b\| \leq \|A^{-1}\| \|\delta b\|$$

误差分析介绍

因

$$\|b\| = \|Ax\| \leq \|A\|\|x\| \Rightarrow \frac{1}{\|x\|} \leq \frac{\|A\|}{\|b\|}$$

所以

$$\frac{\|\delta x\|}{\|x\|} \leq \|A\| \|A^{-1}\| \frac{\|\delta b\|}{\|b\|}$$

即常数项产生的相对误差，可能将解的相对误差放大 $\|A\| \|A^{-1}\|$ 倍。

类似分析，若 A 有误差 δA ，所得解 $x + \delta x$ ，则

$$\frac{\|\delta x\|}{\|x\|} \leq \frac{\|A\| \cdot \|A^{-1}\| \frac{\|\delta A\|}{\|A\|}}{1 - \|A\| \cdot \|A^{-1}\| \frac{\|\delta A\|}{\|A\|}}$$

大体上，系数矩阵 A 产生的相对误差，将解的相对误差有可能放大 $\|A\| \|A^{-1}\|$ 倍。

$\|A\| \|A^{-1}\|$ 刻画了线性方程组中原始数据变化对解的影响，即刻画了方程组的性态。

误差分析介绍

定义: 设 A 为非奇异矩阵, $\|\cdot\|$ 为矩阵的算子范数, 则称

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

为矩阵 A 的条件数。

当 $\text{cond}(A)$ 很大时, $Ax = b$ 是“病态”的;

当 $\text{cond}(A)$ 很小时, $Ax = b$ 是“良态”的。

$\text{cond}(A)$ 的下限是多少?

误差分析介绍

定义: 设 A 为非奇异矩阵, $\|\cdot\|$ 为矩阵的算子范数, 则称

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

为矩阵 A 的条件数。

当 $\text{cond}(A)$ 很大时, $Ax = b$ 是“病态”的;

当 $\text{cond}(A)$ 很小时, $Ax = b$ 是“良态”的。

$\text{cond}(A)$ 的下限是多少?

$$\text{cond}(A) = \|A\| \|A^{-1}\| \geq \|AA^{-1}\| = \|I\| = 1$$

误差分析介绍

直观判断 $Ax = b$ 性态的方法:

- 用主元消去法求解时出现小主元;
- 某些行 (列) 几乎线性相关;
- 矩阵 A 的元素间数量级相差很大, 且无规律;
- 当 $r = b - Ax$ 的 $\|r\|$ 很小时, x 作为解精度仍不够;

出现上述情况之一, 方程 $Ax = b$ 可能“病态”。

对于“病态”方程组的求解, 常用的方法和措施:

- 提高原始数据和运算的精度。
- 用适当方法改善原始模型的性态, 例如降低矩阵 A 的条件数。

引论

① 基本概念

② 线性方程组的迭代法

③ 非线性方程的迭代法

线性方程组的迭代法

由第二章可知, 利用直接法可解线性方程组

$$Ax = b \quad (6.1)$$

其中, $A \in \mathbb{R}^{n \times n}$, $b \in \mathbb{R}^n$, $x \in \mathbb{R}^n$

问题: 直接法中, 系数矩阵 A 在不断变化, A 的阶数高, 则占用内存就大; 且程序较复杂, 程序设计的技巧也较高.

迭代法的思路利用迭代法求解 (6.1), 先将它变形为如下等价方程组:

$$x = Bx + f \quad (6.2)$$

其中, $B \in \mathbb{R}^{n \times n}$, $f \in \mathbb{R}^n$, $x \in \mathbb{R}^n$, 矩阵 B 被称为迭代矩阵。

注: 利用不同的方法构造 (6.2) 可得到不同的迭代法。

线性方程组的迭代法

迭代过程:

取初始向量 $\mathbf{x}^{(0)}$ 代入 (6.2), 得

$$\mathbf{x}^{(1)} = \mathbf{B}\mathbf{x}^{(0)} + \mathbf{f}$$

$$\mathbf{x}^{(2)} = \mathbf{B}\mathbf{x}^{(1)} + \mathbf{f}$$

$$\mathbf{x}^{(3)} = \mathbf{B}\mathbf{x}^{(2)} + \mathbf{f}$$

⋮

一般形式为

$$\mathbf{x}^{(k+1)} = \mathbf{B}\mathbf{x}^{(k)} + \mathbf{f}, \quad (k = 0, 1, 2, \dots) \quad (6.3)$$

称上述求解的方法为求解线性方程的迭代法, 或迭代过程或迭代格式。

线性方程组的迭代法

迭代收敛:

当 $k \rightarrow \infty$ 时, 若由 (6.3) 所得到的序列 $\mathbf{x}^{(k)} \rightarrow \mathbf{x}^*$, 即

$$x_i^{(k)} \rightarrow x_i^*, \quad (i = 1, 2, \dots, n)$$

其中, $\mathbf{x}^{(k)} = (x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)})^T$, $\mathbf{x}^* = (x_1^*, x_2^*, \dots, x_n^*)^T$, 或写成

$$\lim_{k \rightarrow \infty} \mathbf{x}^{(k)} = \mathbf{x}^* \Leftrightarrow \lim_{k \rightarrow \infty} x_i^{(k)} = x_i^* \quad (i = 1, 2, \dots, n)$$

则称迭代法收敛, 否则迭代法发散。

若迭代法收敛, 则由 (6.3) 得

$$\mathbf{x}^* = \mathbf{B}\mathbf{x}^* + \mathbf{f}$$

即 \mathbf{x}^* 为方程组 (6.2) 的解, 从而也是 (6.1) 的解。

注: 用迭代法求解就是求向量序列 $\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots$ 的极限向量 \mathbf{x}^* 。

简单迭代法

简单迭代法又称为基本迭代法。可以有多种形式的推广。

设线性方程组的一般形式为

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

简单迭代法

设矩阵 $A = (a_{ij})_{n \times n}$ 为非奇异, 且 $a_{ii} \neq 0$, 则

$$\left\{ \begin{array}{l} x_1 = \frac{1}{a_{11}} \left[b_1 - \sum_{j=2}^n a_{1j} x_j \right] \\ x_2 = \frac{1}{a_{22}} \left[b_2 - \sum_{j=1, j \neq 2}^n a_{2j} x_j \right] \\ \vdots \\ x_i = \frac{1}{a_{ii}} \left[b_i - \sum_{j=1, j \neq i}^n a_{ij} x_j \right] \\ \vdots \\ x_n = \frac{1}{a_{nn}} \left[b_n - \sum_{j=1}^{n-1} a_{nj} x_j \right] \end{array} \right.$$

简单迭代法

即

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left[b_i - \sum_{j=1, j \neq i}^n a_{ij} x_j^{(k)} \right],$$
$$(k = 0, 1, 2, \dots, i = 1, 2, \dots, n)$$

上式等价为

$$x_i^{(k+1)} = x_i^{(k)} + \frac{1}{a_{ii}} \left[b_i - \sum_{j=1}^n a_{ij} x_j^{(k)} \right],$$
$$(k = 0, 1, 2, \dots, i = 1, 2, \dots, n)$$

即

$$\begin{cases} x_i^{(k+1)} = x_i^{(k)} + \Delta x_i, \\ \Delta x_i = \frac{1}{a_{ii}} \left[b_i - \sum_{j=1}^n a_{ij} x_j^{(k)} \right], \end{cases}$$
$$(k = 0, 1, 2, \dots, i = 1, 2, \dots, n)$$

上述迭代法被称为 Jacobi 迭代法

简单迭代法

设

$$\mathbf{D} = \text{diag}(a_{11}, a_{22}, \dots, a_{nn})$$

$$\mathbf{L} = \begin{bmatrix} 0 & & & & & 0 \\ -a_{21} & 0 & & & & \\ \vdots & \ddots & \ddots & & & \\ -a_{j1} & & -a_{jj-1} & 0 & & \\ \vdots & & \vdots & \ddots & & \ddots \\ -a_{n1} & \cdots & -a_{nj-1} & \cdots & -a_{nn-1} & 0 \end{bmatrix}$$

$$\mathbf{U} = \begin{bmatrix} 0 & -a_{12} & \cdots & -a_{1j} & \cdots & -a_{1n} \\ \ddots & \ddots & & \vdots & & \vdots \\ 0 & -a_{j-1,j} & \cdots & -a_{j-1,n} & & \vdots \\ \ddots & \ddots & & \vdots & & \vdots \\ 0 & -a_{n-1,n} & & & & 0 \end{bmatrix}$$

简单迭代法

则

$$\mathbf{A} = \mathbf{D} - \mathbf{L} - \mathbf{U}$$

$$\mathbf{D}\mathbf{x} = (\mathbf{L} + \mathbf{U})\mathbf{x} + \mathbf{b}$$

$$\mathbf{x} = \mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})\mathbf{x} + \mathbf{D}^{-1}\mathbf{b}$$

令

$$\mathbf{B}_J = \mathbf{D}^{-1}(\mathbf{L} + \mathbf{U}),$$

$$\mathbf{f} = \mathbf{D}^{-1}\mathbf{b}$$

得 (6.1) 的等价方程组为

$$\mathbf{x} = \mathbf{B}_J\mathbf{x} + \mathbf{f}$$

迭代格式为

$$\mathbf{x}^{(k+1)} = \mathbf{B}_J\mathbf{x}^{(k)} + \mathbf{f}, \quad (k = 0, 1, 2, \dots)$$

例题

例：用 Jacobi 迭代格式解线性方程组

$$\begin{cases} 8x_1 - 3x_2 + 2x_3 = 20 \\ 4x_1 + 11x_2 - x_3 = 33 \\ 2x_1 + x_2 + 4x_3 = 12 \end{cases}$$

例题

例：用 Jacobi 迭代格式解线性方程组

$$\begin{cases} 8x_1 - 3x_2 + 2x_3 = 20 \\ 4x_1 + 11x_2 - x_3 = 33 \\ 2x_1 + x_2 + 4x_3 = 12 \end{cases}$$

解：其矩阵形式为

$$\begin{aligned} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}^{(k+1)} &= \begin{bmatrix} 1/8 & 0 & 0 \\ 0 & 1/11 & 0 \\ 0 & 0 & 1/4 \end{bmatrix} \begin{bmatrix} 0 & 3 & -2 \\ -4 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}^{(k)} + \begin{bmatrix} 2.5 \\ 3 \\ 3 \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0.375 & -0.25 \\ -0.363636 & 0 & 0.090909 \\ -0.5 & -0.25 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}^{(k)} + \begin{bmatrix} 2.5 \\ 3 \\ 3 \end{bmatrix} \end{aligned}$$

例题

取初始向量 $\mathbf{x}^{(0)} = [0, 0, 0]^T$, 得到

$$\mathbf{x}^{(1)} = \mathbf{x}^{(0)} + (2.5, 3, 3)^T = (2.5, 3, 3)^T$$

$$\mathbf{x}^{(2)} = \begin{bmatrix} 0 & 0.375 & -0.25 \\ -0.363636 & 0 & 0.090909 \\ -0.5 & -0.25 & 0 \end{bmatrix} \begin{bmatrix} 2.5 \\ 3 \\ 3 \end{bmatrix} + \begin{bmatrix} 2.5 \\ 3 \\ 3 \end{bmatrix} = \begin{bmatrix} 2.875 \\ 2.363 \\ 1 \end{bmatrix}$$

$$\mathbf{x}^{(3)} = \begin{bmatrix} 0 & 0.375 & -0.25 \\ -0.363636 & 0 & 0.090909 \\ -0.5 & -0.25 & 0 \end{bmatrix} \begin{bmatrix} 2.875 \\ 2.363 \\ 1 \end{bmatrix} + \begin{bmatrix} 2.5 \\ 3 \\ 3 \end{bmatrix} = \begin{bmatrix} 3.136 \\ 2.045 \\ 0.972 \end{bmatrix}$$

⋮

$$\mathbf{x}^{(10)} = (3.000032, 1.999838, 0.999881)^T$$

Gauss-Seidel 迭代法

我们已知

$$\mathbf{D}\mathbf{x}^{k+1} = (\mathbf{L} + \mathbf{U})\mathbf{x}^k + \mathbf{b}$$

对上述方程进行一下改进，得

$$\begin{aligned}\mathbf{D}\mathbf{x}^{k+1} &= \mathbf{L}\mathbf{x}^{k+1} + \mathbf{U}\mathbf{x}^k + \mathbf{b} \\ \Rightarrow \mathbf{x}^{k+1} &= (\mathbf{D} - \mathbf{L})^{-1}\mathbf{U}\mathbf{x}^k + (\mathbf{D} - \mathbf{L})^{-1}\mathbf{b}\end{aligned}$$

令

$$\mathbf{B}_G = (\mathbf{D} - \mathbf{L})^{-1}\mathbf{U}; \mathbf{f}_G = (\mathbf{D} - \mathbf{L})^{-1}\mathbf{b}$$

得到 Gauss-Seidel 迭代法，简称 G-S 法

$$\mathbf{x}^{k+1} = \mathbf{B}_G\mathbf{x}^k + \mathbf{f}_G$$

元素形式：

$$\begin{aligned}x_i^{(k+1)} &= x_i^{(k)} + \frac{1}{a_{ii}} \left[b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i}^n a_{ij}x_j^{(k)} \right], \\ (k &= 0, 1, 2, \dots, i = 1, 2, \dots, n)\end{aligned}$$

例题

例：将下面线性方程组写成G-S迭代格式，并求解

$$\begin{cases} 8x_1 - 3x_2 + 2x_3 = 20 \\ 4x_1 + 11x_2 - x_3 = 33 \\ 2x_1 + x_2 + 4x_3 = 12 \end{cases}$$

例题

例：将下面线性方程组写成G-S迭代格式，并求解

$$\begin{cases} 8x_1 - 3x_2 + 2x_3 = 20 \\ 4x_1 + 11x_2 - x_3 = 33 \\ 2x_1 + x_2 + 4x_3 = 12 \end{cases}$$

解

$$\begin{cases} x_1^{(k+1)} = 2.5 + 0.375x_2^{(k)} - 0.25x_3^{(k)} \\ x_2^{(k+1)} = 3 - 0.363636x_1^{(k+1)} + 0.090909x_3^{(k)} \\ x_3^{(k+1)} = 3 - 0.5x_1^{(k+1)} - 0.25x_2^{(k+1)} \end{cases}$$

取初始向量 $\mathbf{x}^{(0)} = [0, 0, 0]^T$ 代入上式，得

$$\mathbf{x}^{(1)} = [2.5, 2.090909, 1.768939]^T$$

⋮

$$\mathbf{x}^{(5)} = [2.999843, 2.000072, 1.000061]^T$$

迭代法的收敛性

设某种迭代格式

$$\mathbf{x}^{(k+1)} = \mathbf{B}\mathbf{x}^{(k)} + \mathbf{f}$$

且该方程组的精确解为 \mathbf{x}^* , 则

$$\mathbf{x}^* = \mathbf{B}\mathbf{x}^* + \mathbf{f}$$

因此

$$\begin{aligned}\boldsymbol{\varepsilon}^{(k+1)} &= \mathbf{x}^{(k+1)} - \mathbf{x}^* = \mathbf{B} \left(\mathbf{x}^{(k)} - \mathbf{x}^* \right) \\ &= \mathbf{B}\boldsymbol{\varepsilon}^{(k)} = \mathbf{B}^2\boldsymbol{\varepsilon}^{(k-1)} = \dots = \mathbf{B}^{k+1}\boldsymbol{\varepsilon}^{(0)}\end{aligned}$$

其中, $\boldsymbol{\varepsilon}^{(0)} = \mathbf{x}^{(0)} - \mathbf{x}^*$ 是一个非零的不变向量。

于是当 $k \rightarrow +\infty$, $\boldsymbol{\varepsilon}^{(k+1)} \rightarrow \mathbf{0}$ 时, 有

$$\lim_{k \rightarrow \infty} \mathbf{B}^{k+1}\boldsymbol{\varepsilon}^{(0)} = \mathbf{0}$$

$$\Rightarrow \lim_{k \rightarrow \infty} \mathbf{B}^k = \mathbf{0}$$

迭代法的收敛性

定理 2: 迭代法 $\mathbf{x}^{(k+1)} = \mathbf{B}\mathbf{x}^{(k)} + \mathbf{f}$ 对任意 $\mathbf{x}^{(0)}$ 和 \mathbf{f} 均收敛的充分必要条件为

$$\rho(\mathbf{B}) < 1$$

由于 $\rho(\mathbf{B}) \leq \|\mathbf{B}\|$, 得

定理 3: 若 $\|\mathbf{B}\| < 1$, 则迭代法 $\mathbf{x}^{(k+1)} = \mathbf{B}\mathbf{x}^{(k)} + \mathbf{f}$ 收敛。

定理 4: 若 $Ax = b$ 中的 $A = (a_{ij})_{n \times n}$ 为按行严格对角占优, 则 Jacobi 法和 G-S 法均收敛。

$$|a_{ii}| > \sum_{j=1, i \neq j}^n |a_{ij}|, \quad i = 1, 2, \dots, n$$

例题

如何处理下面方程组，确保在使用 G-S 法时，解收敛。

$$\begin{cases} 4x_1 + 11x_2 - x_3 = 33 \\ 8x_1 - 3x_2 + 2x_3 = 20 \\ 2x_1 + x_2 + 4x_3 = 12 \end{cases}$$

引论

① 基本概念

② 线性方程组的迭代法

③ 非线性方程的迭代法

非线性方程的迭代法

问题: 设非线性方程

$$f(\boldsymbol{x}) = 0$$

求一数 $\bar{\boldsymbol{x}}$, 使 $f(\bar{\boldsymbol{x}}) = 0$, 称 $\bar{\boldsymbol{x}}$ 为上述方程的根。

例如:

$$\begin{cases} xy - z = 1 \\ xyz + y^2 = 2 \\ e^x + z = 3 \end{cases}$$

当前流行得深度神经网络 (Deep Neural Networks) 几乎都是非线性方程。

非线性方程的迭代法

假设: 函数 $f(x)$ 是连续的, 它在坐标系 Oxy 中的图象为连续曲线。

- 若在区间 $[a, b]$ 上只有一个根, 称 $[a, b]$ 为单根区间;
- 若在区间 $[a, b]$ 上有多个根, 称为多根区间;

统称为有根区间。

实际问题中, 大多是多跟问题。

简单迭代法

先将方程 $f(\mathbf{x}) = 0$ 化为一个与它同解的方程

$$\mathbf{x} = \varphi(\mathbf{x})$$

即 $f(\bar{\mathbf{x}}) = 0$ 充分必要条件是 $\bar{\mathbf{x}} = \varphi(\bar{\mathbf{x}})$

然后, 任取一个初始值 \mathbf{x}_0 , 进行如下迭代

$$\mathbf{x}_1 = \varphi(\mathbf{x}_0), \mathbf{x}_2 = \varphi(\mathbf{x}_1), \dots, \mathbf{x}_{k+1} = \varphi(\mathbf{x}_k), \dots$$

即迭代公式为

$$\mathbf{x}_{k+1} = \varphi(\mathbf{x}_k), \quad (k = 0, 1, 2, \dots)$$

称上述过程为求解非线性方程的**简单迭代法**, 或迭代法或迭代过程或迭代格式,
 $\varphi(\mathbf{x})$ 称为迭代函数, \mathbf{x}_k 称为第 k 步的迭代值或简单迭代值。

迭代收敛

如果由迭代法产生的数列收敛, 即当 $k \rightarrow \infty$ 时, $\mathbf{x}_k \rightarrow \bar{\mathbf{x}}$, 则称迭代法收敛; 否则称发散。

显然, 收敛时有

$$f(\bar{\mathbf{x}}) \equiv 0$$

证明?

迭代收敛

如果由迭代法产生的数列收敛, 即当 $k \rightarrow \infty$ 时, $\mathbf{x}_k \rightarrow \bar{\mathbf{x}}$, 则称迭代法收敛; 否则称发散。

显然, 收敛时有

$$f(\bar{\mathbf{x}}) \equiv 0$$

证明?

$$\bar{\mathbf{x}} = \varphi(\bar{\mathbf{x}}) \Leftrightarrow f(\bar{\mathbf{x}}) \equiv 0$$

迭代函数的构造方法很多, 例如,

$$\mathbf{x} = \mathbf{x} - f(\mathbf{x}) = \varphi(\mathbf{x})$$

$$\mathbf{x} = \mathbf{x} - k(\mathbf{x})f(\mathbf{x}) = \varphi(\mathbf{x}), k(\mathbf{x}) \neq 0$$

例题

例：用迭代法求 $f(x) = 2x^3 - x - 1 = 0$ 的根

例题

例：用迭代法求 $f(x) = 2x^3 - x - 1 = 0$ 的根

解：用以下两种方法：

1) 化方程为等价方程

$$x = \sqrt[3]{\frac{x+1}{2}} = \varphi(x)$$

取初始值 $x_0 = 0$, 则迭代值为

$$x_1 = \sqrt[3]{\frac{1}{2}} = \sqrt[3]{0.5} \approx 0.79$$

$$x_2 = \sqrt[3]{\frac{x_1+1}{2}} = \sqrt[3]{\frac{1+0.79}{2}} = \sqrt[3]{0.895} \approx 0.964$$

$$x_3 = \sqrt[3]{\frac{x_2+1}{2}} = \sqrt[3]{\frac{1+0.964}{2}} = \sqrt[3]{0.982} \approx 0.994$$

⋮

显然，当 $k \rightarrow \infty$ 时， $x_k \rightarrow 1$ ，且 $f(1) = 0$

例题

2) 化方程为等价方程

$$x = 2x^3 - 1 = \varphi(x)$$

取初始值 $x_0 = 0$, 则迭代值为

$$x_1 = 2 \times 0 - 1 = -1$$

$$x_2 = 2 \times (-1)^3 - 1 = -3$$

$$x_3 = 2 \times (-3)^3 - 1 = -55, \dots$$

显然, 当 $k \rightarrow \infty$ 时, $x_k \rightarrow -\infty$ 。故迭代法发散。

由上例可以看出, 迭代法的收敛与发散, 与迭代函数的构造有关。

迭代法收敛的条件

定理 设迭代函数 $\varphi(x)$ 满足

- 当 $x \in [a, b]$ 时, $a \leq \varphi(x) \leq b$
- 存在实数 $0 < L < 1$, 对任意 $x \in [a, b]$, 都有

$$|\varphi'(x)| \leq L$$

则 $x = \varphi(x)$ 在 $[a, b]$ 内存在唯一根 \bar{x} , 并且对任意初始值 $x_0 \in [a, b]$, 迭代法

$$x_{k+1} = \varphi(x_k), \quad (k = 0, 1, 2, \dots)$$

收敛于 \bar{x} , 且满足

$$|x_k - \bar{x}| \leq \frac{L}{1-L} |x_k - x_{k-1}|$$

$$|x_k - \bar{x}| \leq \frac{L^k}{1-L} |x_1 - x_0|$$

$L \rightarrow 0$ or $L \rightarrow 1$ 收敛快?

迭代法收敛的速度

收敛的阶: 设迭代序列 $x_k \rightarrow \bar{x}, k \rightarrow \infty$

定义: 若存在实数 $p \geq 1$ 和 $c > 0$ 满足

$$\lim_{k \rightarrow \infty} \frac{|e_{k+1}|}{|e_k|^p} = \lim_{k \rightarrow \infty} \frac{|x_{k+1} - \bar{x}|}{|x_k - \bar{x}|^p} = c$$

则称迭代法为 p 阶收敛。

- 当 $p = 1$ 时称为线性收敛;
- 当 $p > 1$ 时称为超线性收敛;
- 当 $p = 2$ 时称为平方收敛。

Newton迭代法及其变形

构造迭代函数是迭代法中很关键的一步, Newton迭代法是按照如下方式构造:
对一切非线性方程 $f(\mathbf{x}) = 0$, 总可构造如下函数:

$$\mathbf{x} = \varphi(\mathbf{x}) = \mathbf{x} - k(\mathbf{x})f(\mathbf{x}), \quad k(\mathbf{x}) \neq 0$$

作为方程 $f(\mathbf{x}) = 0$ 求解的迭代函数。因

$$\varphi'(\mathbf{x}) = 1 - k'(\mathbf{x})f(\mathbf{x}) - k(\mathbf{x})f'(\mathbf{x})$$

且 $|\varphi'(\mathbf{x})|$ 在根 $\bar{\mathbf{x}}$ 附近越小, 其局部收敛速度就越快, 故令

$$\varphi'(\bar{\mathbf{x}}) = 1 - k'(\bar{\mathbf{x}})f(\bar{\mathbf{x}}) - k(\bar{\mathbf{x}})f'(\bar{\mathbf{x}}) = 1 - k(\bar{\mathbf{x}})f'(\bar{\mathbf{x}}) = 0$$

若 $f'(\bar{\mathbf{x}}) \neq 0$ (即 $\bar{\mathbf{x}}$ 不是 $f(\mathbf{x}) = 0$ 的重根), 则

$$k(\bar{\mathbf{x}}) = 1/f'(\bar{\mathbf{x}})$$

因此, 可取 $k(\mathbf{x}) = 1/f'(\mathbf{x})$ 代入 $\varphi(\mathbf{x})$, 得

$$\mathbf{x} = \mathbf{x} - \frac{f(\mathbf{x})}{f'(\mathbf{x})}$$

Newton迭代法

定理: 设方程 $f(\mathbf{x}) = 0$ 的根为 $\bar{\mathbf{x}}$, 且 $f'(\bar{\mathbf{x}}) \neq 0$, 则迭代法

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \frac{f(\mathbf{x}_k)}{f'(\mathbf{x}_k)}, \quad (k = 0, 1, 2, \dots)$$

至少是平方收敛, 称为 Newton 迭代法。

由于在 Newton 迭代法中, 需要利用导数 $f'(\mathbf{x})$, 有时不方便。若利用近似等式

$$f'(\mathbf{x}_k) \approx \frac{f(\mathbf{x}_k) - f(\mathbf{x}_{k-1})}{\mathbf{x}_k - \mathbf{x}_{k-1}}$$

得

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \frac{f(\mathbf{x}_k)}{f(\mathbf{x}_k) - f(\mathbf{x}_{k-1})} (\mathbf{x}_k - \mathbf{x}_{k-1}), \quad (k = 0, 1, 2, \dots)$$

称为弦截法。

例题

例：用 Newton 法和弦截法分别计算方程

$$x^3 - x - 1 = 0$$

在 $x = 1.5$ 附近的根。

例题

例：用 Newton 法和弦截法分别计算方程

$$x^3 - x - 1 = 0$$

在 $x = 1.5$ 附近的根。

解：(1) 使用Newton法，取 $x_0 = 1.5$ ，迭代公式：

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} = x_k - \frac{x_k^3 - x_k - 1}{3x_k^2 - 1}$$

因此

$$x_1 = x_0 - \frac{x_0^3 - x_0 - 1}{3x_0^2 - 1} = 1.5 - \frac{1.5^3 - 1.5 - 1}{3 \times 1.5^2 - 1} \approx 1.34783$$

$$x_2 = x_1 - \frac{x_1^3 - x_1 - 1}{3x_1^2 - 1} \approx 1.32520$$

$$x_3 = x_2 - \frac{x_2^3 - x_2 - 1}{3x_2^2 - 1} \approx 1.32472$$

$$x_4 = x_3 - \frac{x_3^3 - x_3 - 1}{3x_3^2 - 1} \approx 1.32472$$

例题

(2) 使用弦截法, 取 $x_0 = 1.5, x_1 = 1.4$, 迭代公式

$$\begin{aligned}x_{k+1} &= x_k - \frac{f(x_k)}{f(x_k) - f(x_{k-1})} (x_k - x_{k-1}) \\&= x_k - \frac{x_k^3 - x_k - 1}{x_k^2 + x_{k-1}x_k + x_{k-1}^2 - 1}\end{aligned}$$

因此

$$x_2 = x_1 - \frac{x_1^3 - x_1 - 1}{x_1^2 + x_0x_1 + x_0^2 - 1} \approx 1.33522$$

$$x_3 = x_2 - \frac{x_2^3 - x_2 - 1}{x_2^2 + x_1x_2 + x_1^2 - 1} \approx 1.32541$$

例题

(3) 取 $x_0 = 0$, 使用 Newton 法计算方程的根。利用

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} = x_k - \frac{x_k^3 - x_k - 1}{3x_k^2 - 1}$$

进行迭代计算, 得

$$x_1 = -1, x_2 = -0.5, x_3 \approx 0.33, x_4 \approx -1.44$$

可见结果偏离所求的根, 且可能不收敛。

因此, Newton 法收敛与否与初始值有关。

改进Newton法

在 Newton 法中，为了防止迭代发散，增加一个条件

$$|f(x_{k+1})| < |f(x_k)|$$

为了使 $|f(x_k)|$ 满足这种单调性，引入常数 $\lambda \in (0, 1]$ ，迭代公式

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

称为 Newton 下山法， λ 称为下山因子。

在 Newton 下山法中，下山因子可采用试算法，如取

$$\lambda = 1, \frac{1}{2}, \frac{1}{2^2}, \dots$$

改进Newton法

```
for k = 1, 2 ...
    λ = 1
    for p = 1, 2, ...
         $x_{k+1} = x_k - \lambda \frac{f(x_k)}{f'(x_k)}$ 
        if  $|f(x_{k+1})| < |f(x_k)|$ 
            break
        else
             $\lambda = \frac{\lambda}{2}$ 
        end
    end
    if  $|f(x_k)| < \epsilon$ 
        break
    end
end
```

多根区间上的逐次逼近法

多根区间上的逐次逼近法方程 $f(x) = 0$ 在多根区间 $[a, b]$ 上分两种情况：

1. 均为单根
2. 有重根

二分法

$[a, b]$ 是 $f(x) = 0$ 仅有单根的多根区间

1. 求单根区间

设 $f(x) = 0$ 在 $[a, b]$ 上有 m 个根, 将 $[a, b]$ 分成 n 个小区间

$$[b_0, b_1], [b_1, b_2], \dots, [b_{n-1}, b_n], b_0 = a, b_n = b$$

计算 $f(b_i)$ ($i = 0, 1, \dots, n$) 的值。

若 $f(b_i)f(b_{i+1}) < 0$, 则 $f(x) = 0$ 在 $[b_i, b_{i+1}]$ 上至少有一个根。

若这样的有根区间少于根的个数, 则将这些区间继续对分, 对分点为 $b_{i+1/2}$, 计算 $f(b_{i+1/2})$, 再搜索有根区间, 直到有根区间的个数是 m 为止。

二分法

2. 在单根区间 $[c, d]$ 上求根

$f(c)f(d) < 0$, 将 $[c, d]$ 对分, 设对分点 $x_0 = \frac{c+d}{2}$ 计算 $f(x_0)$, 若 $f(x_0)$ 与 $f(c)$ 同号, 则令 $c_1 = x_0, d_1 = d$; 否则令 $c_1 = c, d_1 = x_0$ 。在新的有根区间 $[c_1, d_1]$ 中, 利用上述对分方法重复进行得到新的有根区间 $[c_2, d_2]$, 继续下去得有根区间 $[c_n, d_n]$, 其长度

$$d_n - c_n = \frac{d - c}{2^n} \rightarrow 0$$

因此, 当 n 足够大时, $d_n - c_n$ 可达到根的精度要求, 则

$$x_n = \frac{d_n - c_n}{2}$$

可作为根的近似值。以上求根的方法称为**二分法**。

几何解释?

例题

例：利用二分法求

$$f(x) = x^2 + 2x - 1 = 0$$

在 $[0, 1]$ 中的根。

例题

例：利用二分法求

$$f(x) = x^2 + 2x - 1 = 0$$

在 $[0, 1]$ 中的根。

解：因 $f(0) = -1 < 0, f(1) = 2 > 0$, 所以 $f(x)$ 在 $[0, 1]$ 中有根。

下面利用二分法求根。将有根区间 $[0, 1]$ 二等分，得 $[0, 0.5], [0.5, 1]$ 。因

$$f(0.5) = 0.25 > 0$$

故 $f(x)$ 在 $[0, 0.5]$ 中有根。

因

$$f(0.25) = -0.435 < 0$$

故 $f(x)$ 在 $[0.25, 0.5]$ 中有根。

上述过程重复下去，可得根的近似值。