链滴

数值分析期末公式总结

作者：zyk
原文链接：https://ld246.com/article/1609850303729
来源网站：链滴

<h2 id="第一章-误差">第一章 误差</h2>
<h3 id="1--误差的分类">1. 误差的分类</h3>

模型误差：数学模型本身包含的误差；
观测误差：观测结果带来的误差；
数据误差：数据可能由先前的计算等途径得到，从而导致的误差；
截断误差（方法误差）：模型的准确解和用数值方法求得的解之差；
舍入误差：计算过程中只能取有限位数字进行运算而引来的误差.

<h3 id="2--近似计算中需要注意的一些现象">2. 近似计算中需要注意的一些现象</h3>

要避免两个相近的数相减；
两个相差很大的数进行运算时，要防止小的那个数被“吃掉”；
要注意计算步骤的简化，减少运算次数；
要避免做被除数的绝对值远远大于除数绝对值的除法；
要选用数值稳定的计算公式.

<h2 id="第二章-插值法与数值微分">第二章 插值法与数值微分</h2>
<h3 id="1--线性插值">1. 线性插值</h3>
<p>线性插值就是用直线近似地代替函数 f(x
 。用两个点(x_0 , <span class="language-
ath"> y_0)和(x_1 , <span class="language
math"> y_1)通过两点式来构建直线方程 <span class="language-
ath"> \phi_1(x) ，得到的 \phi_1(x) 就是<str
ng>插值函数：</p>
<div class="language-math">\phi_1(x)=y_0\frac{x-x_1}{x_0-x_1}+y_1\frac{x-x_0}{x_1-x_0}</di
>
<p>记 l_0(x) = \frac{
-x_1}{x_0-x_1} ， l_1(x) = <span class="lang
age-math">\frac{x-x_0}{x_1-x_0} ，并把 l_0(x)
叫做 x_0 的一次插值基函数， <sp
n class="language-math">l_1(x) 叫做 x_1 的
strong>一次插值基函数.</p>
<h4 id="Lagrange-插值">Lagrange 插值</h4>
<p>插值函数 \phi_1(x) 是两个插值基函数 <span class=
language-math">l_0(x) 和 l_1(x) 的<strong
线性组合，其组合系数就是对应点的函数值 y_0</span
和 y_1 ，即：</p>
<div class="language-math">\phi_1(x)=y_0l_0(x)+y_1l_1(x)</div>
<p>这种形式的插值函数称为 Lagrange 插值。</p>
<h4 id="Newton-插值">Newton 插值</h4>
<p>把直线用点斜式表示，有：</p>
<div class="language-math">\phi_1(x)=y_0+\frac{f(x_1)-f(x_0)}{x_1-x_0}(x-x_0)</div>
<p>函数 f(x) 在 x_i
/span> ， x_j 的一阶均方差的定
是：</p>
<div class="language-math">f[x_i,x_j]=\frac{f(x_i)-f(x_j)}{x_i-x_j}</div>
<p>利用均方差的对称性，可以将点斜式表示为：</p>
<div class="language-math">\phi_1(x)=y_0+(x-x_0)f[x_0,x_1]</div>
<p>这种形式的插值叫做 Newton 插值。</p>
<h3 id="2--二次插值">2. 二次插值</h3>
<p>过三个点 (x_0 , y
0), (x_1 , y
1) 和 (x_2 , <span class="language-math">

```
_2</span> ) 来构造 <span class="language-math">y=f(x)</span> 的插值函数，就是二次插值
该插值函数的次数不高于二次）。</p>
<h4 id="二次-Lagrange-插值">二次 Lagrange 插值</h4>
<p>二次拉格朗日插值多项式为 <span class="language-math">\phi_2(x)=y_0l_0(x)+y_1l_1(x)+y
2l_2(x)</span> ，其中三个插值基函数分别为：</p>
<div class="language-math">l_0(x)=\frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}</div>
<div class="language-math">l_1(x)=\frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}</div>
<div class="language-math">l_2(x)=\frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}</div>
<h4 id="二次-Newton-插值">二次 Newton 插值</h4>
<p>二次牛顿插值多项式为：</p>
<div class="language-math">\phi_2(x)=f(x_0)+(x-x_0)f[x_0,x1]+(x-x_0)(x-x_1)f[x_0,x_1,x_2]\\=f
x_0)+(x-x_0)\frac{f(x_0)-f(x_1)}{x_0-x_1}+(x-x_0)(x-x_1)\frac{f[x_0-x_1]-f[x_1-x_2]}{x_0-x_2}</div

<h3 id="3--n-次插值">3. n 次插值</h3>
<h4 id="n-次-Lagrange-插值">n 次 Lagrange 插值</h4>
<p>n 次拉格朗日插值多项式为：</p>
<div class="language-math">\phi_n(x)=\sum_{i=0}^ny_il_i(x)</div>
<p>其中插值基函数 <span class="language-math">l_i(x)</span> 为：</p>
<div class="language-math">l_i(x)=\frac{(x-x_0)...(x-x_{i-1})(x-x_{i+1})...(x-x_n)}{(x_i-x_0)...(x_i-x
{i-1})(x_i-x_{i+1})...(x_i-x_n)},\;i=0,1,2,...,n</div>
<p><span class="language-math">l_i(x)</span> 还有另一种形式：</p>
<div class="language-math">l_i(x)=\sum_{i=0}^n\frac{\omega_n(x)}{(x-x_i)\omega_n^{'}(x_i)}
/div>
<p>其中，</p>
<div class="language-math">\omega_n(x)=(x-x_0)(x-x_1)...(x-x_n)\\
\omega_n^{'}(x_i)=(x_i-x0)...(x_i-x_{i-1})(x_i-x_{i+1})...(x_i-x_n)</div>
<h4 id="n-次-Newton-插值">n 次 Newton 插值</h4>
<p>n 次牛顿插值多项式需要先构造<strong>均差表（差商表）</strong>：</p>
<table>
<thead>
<tr>
<th><span class="language-math">x</span></th>
<th><span class="language-math">y</span></th>
<th><span class="language-math">f[x_i,x_j]</span></th>
<th><span class="language-math">f[x_i,x_j,x_k]</span></th>
<th>...</th>
<th><span class="language-math">f[x_0,x_1,...,x_n]</span></th>
</tr>
</thead>
<tbody>
<tr>
<td><span class="language-math">x_0</span></td>
<td><span class="language-math">\underline{y_0}</span></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><span class="language-math">x_1</span></td>
<td><span class="language-math">y_1</span></td>
<td><span class="language-math">\underline{f[x_0,x_1]}</span></td>
<td></td>
<td></td>
```

<td></td>
</tr>
<tr>
<td>x_2</td>
<td>y_2</td>
<td>f[x_1,x_2]</td>
<td>\underline{f[x_0,x_1,x_2]}</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x_3</td>
<td>y_3</td>
<td>f[x_2,x_3]</td>
<td>f[x_1,x_2,x_3]</td>
<td>...</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td></td>
</tr>
<tr>
<td>x_n</td>
<td>y_n</td>
<td>f[x_{n-1},x_n]</td>
<td>f[x_{n-2},x_{n-1},x_n]</td>
<td>...</td>
<td>\underline{f[x_0,x_1,...,x_n]}</td>
</tr>
</tbody>
</table>
<p>其中 f[x_0,x_1,...,x_n] 是 n 阶均差</stron
>，它的表达式为：</p>
<div class="language-math">f[x_0,x_1,...,x_n]=\frac{f[x_0,x_1,...,x_{n-1}]-f[x_1,x_2,...,x_n]}{x_0-x_
}</div>
<p>利用均差表，可以写出 n 次牛顿插值多项式：</p>
<blockquote>
<p>n 次牛顿插值多项式中的所含均差都是均差表中对角线上的项。</p>
</blockquote>
<div class="language-math">\phi_n(x)=f(x_0)+(x-x_0)f[x_0,x_1]+(x-x_0)(x-x_1)f[x_0,x_1,x_2]+..
+\\(x-x_0)(x-x_1)...(x-x_{n-1})f[x_0,x_1,...,x_n]</div>
<h3 id="4--Hermite-插值">4. Hermite 插值</h3>
<p>若给定 n+1 个结点和相应的函数值与微商值，可以
造 2n+1 次的 Hermite 插值多项式 H(x)</s
an> ： </p>

H(x) 是不超过 2n+1 次的插
多项式；
H(x_i)=y_i ， H'(x

=y'_i\;(i=0,1,2,...,n) .

<p>H(x)=\sum_{i=0}^{n}[y_ih_i(x)+y'_iH_i(x)]</p>
<p>由上述条件可知：</p>
<div class="language-math">h_i(x_j)=
\begin{cases}
1 ,& i=j \\
0 ,&i\neq{j}\;(j=0,1,2,...,n)
\end{cases}</div>
<div class="language-math">h'_i(x_j)=0,\;j=0,1,2,...,n</div>
<div class="language-math">H'_i(x_j)=
\begin{cases}
1, & i=j \\
0, & i\neq{j}\;(j=0,1,2,...,n)
\end{cases}</div>
<div class="language-math">H_i(x_j)=0,\;j=0,1,2,...,n</div>
<h2 id="第三章-数据拟合法">第三章 数据拟合法</h2>
<h3 id="1--正交多项式">1. 正交多项式</h3>
<h4 id="Legendre-多项式">Legendre 多项式</h4>
<div class="language-math">P_n(x)=\frac{1}{2^{n}n!}\frac{d^n}{dx^n}(x^2-1)^n,\;n=0,1,2,\c
ots</div>
<h4 id="Laguerre-多项式">Laguerre 多项式</h4>
<div class="language-math">L_n(x)=e^x\frac{d^n}{dx^n}(x^ne^{-x}),\;n=0,1,2,\cdots</div>
<h4 id="Chebyshev-多项式">Chebyshev 多项式</h4>
<div class="language-math">T_n(x)=cos(n\;arccosx)\\
=\frac{1}{2}[(x+\sqrt{x^2-1})^n+(x-\sqrt{x^2-1})^n],\;\lvert{x}\rvert\leq1</div>
<h2 id="第五章-数值积分">第五章 数值积分</h2>
<h3 id="1--梯形求积公式">1. 梯形求积公式</h3>
<p>梯形求积公式就是过 (a, f(a)) 和 <span class="lan
uage-math">(b, f(b)) 两点做直线 <span class="language-math"
P_1(x) ，用 P_1(x) 来代替 <span class="lang
age-math">f(x) ： </p>
<div class="language-math">\int_{a}^{b}f(x)dx\approx{\int_{a}^{b}P_1(x)dx}={\frac{b-a}{2}[f(a
+f(b)]}</div>
<h3 id="2--抛物线求积公式-Simpson-公式-">2. 抛物线求积公式（Simpson 公式）</h3>
<p>将 [a,b] 区间二等分，过点 <span class="language
math">(a, f(a)) ， (\frac{a+b}{2},f(\frac{a+b}{2}))</sp
n> 和 (b,f(b)) 三点做抛物线 P_2(x) ， 用 P_2(x)
代替 f(x) :</p>
<div class="language-math">\int_{a}^{b}f(x)dx\approx{\int_{a}^{b}P_2(x)dx}={\frac{b-a}{6}[f(a
+4f(\frac{a+b}{2})+f(b)]}</div>
<h3 id="3--Newton-Cotes-公式">3. Newton-Cotes 公式</h3>
<p>把区间 [a,b] 作 n 等分，分点为：</p>
<div class="language-math">x_i=a+ih,\;i=0,1,2,...,n,\;h=\frac{a+b}{n}</div>
<p>过这 n+1 个节点，构造一个 n 次多项式：</p>
<div class="language-math">P_n(x)=\sum_{i=0}^n\frac{\omega(x)}{(x-x_i)\omega'(x_i)}f(x_i)<
div>
<p>其中 \omega(x)=(x-x_0)(x-x_1)...(x-x_n) ，用 <spa
class="language-math">P_n(x) 代替被积函数 f(x)</s
an> ，有：</p>
<div class="language-math">\int_a^bf(x)dx\approx\int_a^bP_n(x)dx=\sum_{i=0}^nA_if(x_i)=
b-a)\sum_{i=0}^nc_i^{(n)}f(x_i)</div>
<p>其中 c_i^{(n)} 是不依赖于函数 <span class="langu

ge-math">f(x) 和区间 [a,b] 的常数，叫做 Newton-Cotes 系数，它的值如下表所示：</p>
<blockquote>
<p>期末考试只要求掌握到 n=4 的情况即可.</p>
</blockquote>
<table>
<thead>
<tr>
<th>n</th>
<th>c_0^{(n)}</th>
<th>c_1^{(n)}</th>
<th>c_2^{(n)}</th>
<th>c_3^{(n)}</th>
<th>c_4^{(n)}</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>\frac{1}{2}</td>
<td>\frac{1}{2}</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>\frac{1}{6}</td>
<td>\frac{4}{6}</td>
<td>\frac{1}{6}</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>\frac{1}{8}</td>
<td>\frac{3}{8}</td>
<td>\frac{3}{8}</td>
<td>\frac{1}{8}</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>\frac{7}{90}</td>
<td>\frac{16}{45}</td>
<td>\frac{2}{15}</td>
<td>\frac{16}{45}</td>
<td>\frac{7}{90}</td>
</tr>
</tbody>
</table>
<h4 id="Newton-Cotes-公式的代数精度">Newton-Cotes 公式的代数精度</h4>
<p>对一个一般求积公式</p>
<div class="language-math">\int_a^bf(x)dx\approx\sum_{k=0}^nA_kf(x_k)</div>

<p>若 f(x) 为一个次数不高于 m
的代数多项式时，上式等号成立，但当 f(x) 为
+1 次多项式时，上式不能精确成立，说明该求积公式具有 m 次代
精度.</p>

若 f(x) 是 n 次多项式，则 <span class="language
math">f^{(n+1)}\equiv0 ， 所以 f(x)=P_n(x)
Newton-Cotes 公式代数精度至少为 n.
当 n 为偶数时，Newton-Cotes 公式的代数精度可达到 n+1.

<h3 id="4--复化梯形公式">4. 复化梯形公式</h3>
<p>将区间 [a,b] 作 n 等分，节点 <span class="langua
e-math">x_k=a+kh\;(k=0,1,2,...,n,\;h=\frac{b-a}{n}) .对每个小区间 <span class="langu
ge-math">[x_k,x_{k+1}] 使用梯形求积公式，有：</p>
<div class="language-math">I=\int_a^bf(x)dx=\sum_{k=0}^n\int_{x_k}^{x_{k+1}}f(x)dx\\\app
ox\sum_{k=0}^n\frac{x_{k+1}-x_k}{2}[f(x_k)+f(x_{k+1})]\\=\frac{h}{2}[f(a)+f(b)+2\sum_{k=1}^{n
1}f(a+kh)]</div>
<p>记复化梯形公式 T_n 为：</p>
<div class="language-math">T_n=\frac{h}{2}[f(a)+f(b)+2\sum_{k=1}^{n-1}f(a+kh)]</div>
<h3 id="5--复化抛物线公式">5. 复化抛物线公式</h3>
<p>由于抛物线求积公式用到了区间的中点，所以在构造复化抛物线公式时必
将区间 [a,b] 进行 2n 等分，在每
小区间 [x_{2k-2},x_{2k}] 上使用抛物线求积公式：</p>
<div class="language-math">\int_{x_{2k-2}}^{x_{2k}}f(x)dx\approx\sum_{k=1}^n\frac{2h}{6}[f(
_{2k-2})+4f(x_{2k-1})+f(x_{2k})],\;h=\frac{b-a}{2n}</div>
<p>于是有：</p>
<div class="language-math">I=\int_a^bf(x)dx=\sum_{k=1}^n\int_{x_{2k-2}}^{x_{2k}}f(x)dx\\
pprox\sum_{k=1}^n\frac{h}{3}[f(x_{2k-2})+4f(x_{2k-1})+f(x_{2k})]\\=\frac{h}{3}[f(a)+f(b)+4\sum
{k=1}^nf(x_{2k-1})+2\sum_{k=1}^{n-1}f(x_{2k})]</div>
<p>记复化抛物线公式 S_n 为：</
>
<div class="language-math">S_n=\frac{h}{3}[f(a)+f(b)+4\sum_{k=1}^nf(x_{2k-1})+2\sum_{k=
}^{n-1}f(x_{2k})]</div>
<h2 id="第六章-解线性代数方程组的直接法">第六章 解线性代数方程组的直接法</h2>
<h3 id="1--LU-分解法">1. LU 分解法</h3>
<p>当矩阵 A 的前 n-1 个顺序主子式都不为零时，矩阵 A 可以唯一地分解为两个三角矩阵的乘积：<
p>
<div class="language-math">A=LU</div>
<p>其中 L 是单位下三角矩阵， U 是上三角矩阵。</p>
<blockquote>
<p>如何求 LU 分解矩阵：</p>

利用初等行变换将矩阵 A 化为上三角矩阵 U；
有下三角可逆矩阵 P，使得 PA=U，从而有 LU 分解： <span class="lang
age-math">A=P^{-1}U\;(L=P^{-1}) .

</blockquote>
<p>原方程 Ax=b 等价于 <span class="language-mat
">LUx=b ， 令 Ux=y ， 有</p>
<div class="language-math">Ly=b</div>
<p>从 Ly=b 中解出
 ， 再将 y 代入 <span class="language-math
>Ux=y 解出 x 即可.</p>
<h3 id="2--向量范数">2. 向量范数</h3>

<h4 id="向量范数">向量范数</h4>
<p>任一向量 x\in{R^n} ，对应一个非负实数 <span cla
s="language-math">\parallel{x}\parallel ，具有下面三个性质：</p>

正定性：对所有的 x\in{R^n}
 {\parallel{x}\parallel}\;\geq0 ，而且 <span class="lan
uage-math">{\parallel{x}\parallel}\;=0 当且仅当 x=0<
span> .
齐次性：对所有的 x\in{R^n}
 \alpha\in{R} ，有 {\p
rallel{\alpha{x}}\parallel}\;=\lvert\alpha\rvert\;\cdot{\parallel{x}\parallel} .
三角不等式：对所有的 x，y\in{R^n}</
pan> ，有： {\parallel{x+y}\parallel}\;\leq\;{\parallel{x}\parallel
+{\parallel{y}\parallel} .

<p>则称 \parallel{x}\parallel 为向量 <span class="lan
uage-math">x 的范数.</p>
<h4 id="常见向量范数">常见向量范数</h4>
<p>常见的向量范数有：</p>

1 范数： {\parallel{x}\parallel}_1=\sum_{i=1}^{n}\lvert{x_i}\
vert
2 范数： {\parallel{x}\parallel}_2=(\sum_{i=1}^{n}x_i^2)^{\
rac{1}{2}}
\infty 范数： {\pa
allel{x}\parallel}_{\infty}=max(\lvert{x_1}\rvert,\lvert{x_2}\rvert,...,\lvert{x_n}\rvert)
p 范数： {\parallel{x}\parallel}_p=(\sum_{i=1}^n\lvert{x_i}\
vert^p)^{\frac{1}{p}},\;p\geq1

<h3 id="3--矩阵范数">3. 矩阵范数</h3>
<h4 id="矩阵范数">矩阵范数</h4>
<p>设 A 是一个 n \times n 的实矩阵，按一定规则对应
个非负实数 \parallel{A}\parallel ，称为矩阵 A 的<stron
>范数，必须满足以下四个性质：</p>

正定性：对所有的 A\in{R^{n\times{n}}
 有 \parallel{A}\parallel\;\geq{0} ，而且 <spa
 class="language-math">\parallel{A}\parallel=0 当且仅当 <span class="language-ma
h">A=0 .
齐次性：对所有的 A\in{R^{n\times{n}}
 和 \alpha\in{R} 有 <span class="language-
ath">\parallel{\alpha{A}}\parallel=\lvert{\alpha}\rvert\cdot\lvert{A}\rvert .
三角不等式：对所有的 A,B\in{R^{n\tim
s{n}}} 有 \parallel{A+B}\parallel\;\leq\;\parallel{A}\para
lel+\parallel{B}\parallel .
相容性：对所有的 A,B\in{R^{n\times{n}
} 有 \parallel{AB}\parallel\;\leq\;\parallel{A}\parallel\c
ot\parallel{B}\parallel .

<h4 id="常见矩阵范数">常见矩阵范数</h4>
<p>常见的矩阵范数有：</p>

1 范数（列和范数）： \parallel{A}\parallel_1=max\sum_{i=
}^n\lvert{a_{ij}}\rvert\;(1\leq{j}\leq{n})

```
<li>2 范数： <span class="language-math">\parallel{A}\parallel_2=\sqrt{\lambda_1}</span>
， <span class="language-math">\lambda_1</span> 为矩阵 <span class="language-math">
^TA</span> 的最大特征值</li>
<li><span class="language-math">\infty</span> 范数（行和范数）： <span class="language
math">\parallel{A}\parallel_{\infty}=max\sum_{j=1}^n\lvert{a_{ij}}\rvert\;(1\leq{i}\leq{n})</spa
></li>
<li>F 范数： <span class="language-math">\parallel{A}\parallel_F=(\sum_{i,j=1}^n\lvert{a_{ij
^2}\rvert)^{\frac{1}{2}}</span></li>
</ul>
<h3 id="4--谱半径">4. 谱半径</h3>
<p>设 n <span class="language-math">\times</span> n 阶矩阵 A 的特征值为 <span class="l
nguage-math">\lambda_i\;(i=1,2,...,n)</span> ， 则称</p>
<div class="language-math">\rho(A)=max\lvert{\lambda_i}\rvert\;(1\leq{i}\leq{n})</div>
<p>为矩阵 A 的<strong>谱半径</strong>.</p>
<h3 id="5--条件数">5. 条件数</h3>
<h4 id="条件数">条件数</h4>
<p>数 <span class="language-math">\parallel{A}\parallel\cdot\parallel{A^{-1}}\parallel</sp
n> 称为矩阵 <span class="language-math">A</span> 的<strong>条件数</strong>，记为 <s
an class="language-math">Cond(A)</span> 。</p>
<p>当线性方程组的系数矩阵 <span class="language-math">A</span> 的条件数 <span class="
anguage-math"> Cond(A)</span> 很大时，说明在系数矩阵或右端项产生微小变化时会引起解的
大变化，则称此方程组是 “<strong>病态</strong>” 方程组，否则称方程组为 “<strong>良态</s
rong>” 方程组。</p>
<p>条件数和范数有关，可以在条件数上加下标，对应的范数也应加相同的下标：</p>
<div class="language-math">Cond(A)_{\infty}=\parallel{A}\parallel_{\infty}\cdot\parallel{A^{-
}}\parallel_{\infty} \\
Cond(A)_{2}=\parallel{A}\parallel_{2}\cdot\parallel{A^{-1}}\parallel_{2}=\sqrt{\lambda_1/\lamb
a_n} \\</div>
<p>其中， <span class="language-math">\lambda_1</span> ， <span class="language-ma
h">\lambda_n</span> 是矩阵 <span class="language-math">A^TA</span> 的<strong>最
</strong>和<strong>最小</strong>特征值，故 <span class="language-math">Cond(A)_2</
pan> 又称<strong>谱条件数</strong>。</p>
<h4 id="条件数的性质">条件数的性质</h4>
<ul>
<li><span class="language-math">Cond(A)\geq{1}</span> .</li>
<li>单位矩阵、置换矩阵和正交矩阵的<strong>谱条件数</strong>都为 1.</li>
<li>对任意<strong>非零</strong>实数 <span class="language-math">\alpha</span> ，有 <
pan class="language-math">Cond(\alpha{A})=Cond(A)</span> .</li>
</ul>
<h2 id="第八章-解线性方程组的迭代法">第八章 解线性方程组的迭代法</h2>
<h3 id="1--Jacobi-迭代">1. Jacobi 迭代</h3>
<p>设有方程组</p>
<div class="language-math">\begin{cases}
a_{11}x_1+a_{12}x_2+...+a_{1n}x_n=b_1 \\
a_{21}x_1+a_{22}x_2+...+a_{2n}x_n=b_2 \\
............ \\
a_{n1}x_1+a_{n2}x_2+...+a_{nn}x_n=b_n
\end{cases}</div>
<p>用矩阵表示为</p>
<div class="language-math">Ax=b</div>
<p>假设 <span class="language-math">a_{ii\neq{0}}</span> ， 令</p>
<div class="language-math">\begin{cases}
b_{ij}=-a_{ij}/a_{ii},\;i\neq{j} \\
g_i=bi/a_{ii},\;i=1,2,...,n
```

```
\end{cases}</div>
<p>方程组可改写成</p>
<div class="language-math">\begin{cases}
x_1=\qquad\qquad{b_{12}x_2+b_{13}x_3+...+b_{1n}x_n}+g_1 \\
x_2=b_{21}x_1\qquad\qquad+{b_{23}x_3+...+b_{2n}x_n}+g_2 \\
............ \\
x_n=b_{n1}x_1+{b_{n2}x_2+b_{n3}x_3+...+b_{n,n-1}x_{n-1}}+g_n \\
\end{cases}</div>
<p>若令</p>
<div class="language-math">B=\begin{bmatrix}
0 &amp; b_{12} &amp; b_{13} &amp; \cdots &amp; b_{1n} \\
b_{21} &amp; 0 &amp; b_{23} &amp; \cdots &amp; b_{2n} \\
\vdots &amp; \vdots &amp; \vdots &amp; &amp; \vdots \\
b_{n1} &amp; b_{n2} &amp; b_{n3} &amp; \cdots &amp; 0
\end{bmatrix}</div>
<div class="language-math">D=\begin{bmatrix}
a_{11} &amp; &amp; &amp; \\
&amp; a_{22} &amp; &amp; \\
&amp; &amp; \ddots &amp; \\
&amp; &amp; &amp; a_{nn} \\
\end{bmatrix},
\;
g=\begin{pmatrix}
g_1 \\
g_2 \\
\vdots \\
g_3
\end{pmatrix}</div>
<p>则</p>
<div class="language-math">B=D^{-1}(D-A)=I-D^{-1}A,\;g=D^{-1}b</div>
<p>方程可用矩阵表示为</p>
<div class="language-math">x=Bx+g</div>
<p>选取初始向量 <span class="language-math">x^{(0)}=(x_1^{(0)},x_2^{(0)},\cdots,x_n^{(0)})
{T}</span> ， 代入 <span class="language-math">x^{(1)}=Bx^{(0)}+g</span> 得到一个新向量
<span class="language-math">x^{(1)}=(x_1^{(1)},x_2^{(1)},\cdots,x_n^{(1)})^{T}</span> ，
将 <span class="language-math">x^{(1)}</span> 代入 <span class="language-math">x^{(2)
=Bx^{(1)}+g</span> ， 可以得到 <span class="language-math">x^{(2)}</span> ， 依此类推
可得到迭代格式</p>
<div class="language-math">\begin{cases}
x_1^{(k+1)}=\qquad\qquad{b_{12}x_2^{(k)}+b_{13}x_3^{(k)}+...+b_{1n}x_n^{(k)}}+g_1 \\
x_2^{(k+1)}=b_{21}x_1^{(k)}\qquad\qquad+{b_{23}x_3^{(k)}+...+b_{2n}x_n^{(k)}}+g_2 \\
............ \\
x_n^{(k+1)}=b_{n1}x_1^{(k)}+{b_{n2}x_2^{(k)}+b_{n3}x_3^{(k)}+...+b_{n,n-1}x_{n-1}^{(k)}}+g_n
\
\end{cases}</div>
<p>即</p>
<div class="language-math">x^{(k+1)}=Bx^{(k)}+g,\;k=0,1,2,\cdots</div>
<p>通过上式的计算，可得到向量序列 <span class="language-math">\lbrace{x^{(k)}}\rbrace</
pan> ， 当 <span class="language-math">k\rightarrow+\infty</span> 时，若序列 <span clas
="language-math">\lbrace{x^{(k)}}\rbrace</span> 收敛到 <span class="language-math">x^
</span> ，  <span class="language-math">x^*</span> 就是方程组的解. 以上计算过程过程称为
<strong>Jacobi 迭代法</strong>，矩阵 <strong>B</strong> 为 jacobi 迭代法的<strong>选
矩阵</strong>。</p>
<h3 id="2--Gauss-Seidel-迭代">2. Gauss-Seidel 迭代</h3>
```

<p>Gauss-Seidel 迭代法和 Jacobi 迭代法相似，但不同点是：在计算 <span class="language-ma
h">x_{i}^{(k+1)} 时，Jacobi 迭代法用 (x_1^{(k)},x_2^{(
)},...,x_n^{(k)})^T 代入迭代公式，而 Gauss-Seidel 迭代法会利用之前已经计算出的结果，用
(x_1^{(k+1)},x_2^{(k+1)},...,x_{i-1}^{(k+1)},x_{i+1}^{(k)},...,x_n^{
k)})^{T} 代入迭代公式进行计算.</p>
<p>其迭代过程为：</p>
<div class="language-math">\begin{cases}
x_1^{(k+1)}=\qquad\qquad{b_{12}x_2^{(k)}+b_{13}x_3^{(k)}+...+b_{1n}x_n^{(k)}}+g_1 \\
x_2^{(k+1)}=b_{21}x_1^{(k+1)}\qquad\qquad+{b_{23}x_3^{(k)}+...+b_{2n}x_n^{(k)}}+g_2 \\
.... \\
x_n^{(k+1)}=b_{n1}x_1^{(k+1)}+{b_{n2}x_2^{(k+1)}+b_{n3}x_3^{(k+1)}+...+b_{n,n-1}x_{n-1}^{(
+1)}}+g_n \\
\end{cases}</div>
<p>令</p>
<div class="language-math">L=\begin{bmatrix}
0 & & & & \\
b_{21} & 0 & & & \\
b_{31} & b_{32} & 0 & & \\
\vdots & \vdots & & \ddots & \\
b_{n1} & b_{n2} & \cdots & & 0 \\
\end{bmatrix},\;
U=\begin{bmatrix}
0 & b_{12} & b_{13} & \cdots & b_{1n} \\
 & 0 & b_{23} & \cdots & b_{2n} \\
 & & 0 & & \vdots \\
 & & & \ddots & b_{n-1,n} \\
 & & & & 0 \\
\end{bmatrix}\;</div>
<p>迭代过程用矩阵表示为</p>
<div class="language-math">x^{(k+1)}=Lx^{(k+1)}+Ux^{(k)}+g,\;k=0,1,2,\cdots</div>
<p>因 (I-L) 存在，所以上述迭代格式可表示为</p>
<div class="language-math">x^{(k+1)}=(I-L)^{-1}Ux^{(k)}+(I-L)^{-1}g</div>
<p>称 B_1=(I-L)^{-1}U 为 Gauss-Seidel 迭代法的<str
ng>迭代矩阵.</p>
<blockquote>
<p>注意：</p>
<p>交换方程和未知数的次序都会影响 Gauss-Seidel 迭代法的计算结果，但这种交换对 Jacobi 迭
法没有影响.</p>
</blockquote>
<h3 id="3--迭代法的收敛性判别">3. 迭代法的收敛性判别</h3>
<h4 id="定理-1">定理 1</h4>
<p>Jacobi 迭代法和 Gauss-Seidel 迭代法收敛的充分必要条件是迭代矩阵的
半径小于 1.</p>
<h4 id="定理-2">定理 2</h4>
<p>若迭代矩阵 M 的范数 <span class="language-ma
h">\parallel{M}\parallel<1 ，则 Jacobi 迭代和 Gauss-Seidel 迭代一定收敛，但这只
迭代收敛的充分条件 （因为 \rho(M)\leq{\pa
allel{M}\parallel} ）.</p>
<h4 id="定理-3">定理 3</h4>
<p>若方程组 Ax=b 的系数矩阵 <span class="languag
-math">A 按行严格对角占优或按列严格对角占优</stron
>，即满足条件</p>
<div class="language-math">\begin{vmatrix}a_{ii}\end{vmatrix}>\sum_{j=1,i\neq{j}}^n\beg
n{vmatrix}a_{ij}\end{vmatrix},\;i=1,2,\cdots,n</div>

<p>或</p>
<div class="language-math">\begin{vmatrix}a_{ii}\end{vmatrix}>\sum_{i=1,i\neq{j}}^n\beg
n{vmatrix}a_{ij}\end{vmatrix},\;j=1,2,\cdots,n</div>
<p>则 Jacobi 迭代法和 Gauss-Seidel 迭代法都收敛.</p>
<h4 id="定理-4">定理 4</h4>
<p>若方程组 Ax=b 的系数矩阵 <span class="languag
-math">A 为对称正定矩阵，则 Gauss-Seidel 迭代法收敛.</p>
<blockquote>
<p>注意：</p>
<p>通常判断 Jacobi 或 Gauss-Seidel 迭代法是否收敛，先看方程组的系数矩阵 A 是否按
严格对角占优或按列严格对角占优，若严格对角占优，则收敛；否则，继续判断对应<stro
g>迭代矩阵的谱半径是否小于 1，小于 1 则收敛，否则不收敛.</p>
</blockquote>
<h2 id="第十章-非线性方程组及非线性方程组解法">第十章 非线性方程组及非线性方程组解法</h2

<h3 id="1--二分法">1. 二分法</h3>

找出 f(x)=0 的根所在区间 <span class="language
math">(a,b)，并计算出端点的函数值 f(a),f(b)

计算 f(x) 在区间中点的值 <span class="language
math">f(\frac{a+b}{2}) .
若 f(\frac{a+b}{2})\approx0，则停止迭代. 否则，若
f(\frac{a+b}{2}) 与 f(a
 异号，则根位于 (a,\frac{a+b}{2}) 中，用 <spa
class="language-math"> \frac{a+b}{2} 代替 b</spa
>；若 f(\frac{a+b}{2}) 与 <span class="language-
ath"> f(b) 异号，则根位于 (\frac{a+b}{2},b)
中，用 \frac{a+b}{2} 代替 <span class="language-m
th"> a .
重复上述过程中的第二步和第三步，直到区间缩小到容许误差范围之内.

<h3 id="2--等步长扫描法">2. 等步长扫描法</h3>
<p>设 h 是给定的步长，方程 <span class="language
math"> f(x)=0 的有根区间为 [a,b]，一般初
时 h 取 0.1，取 x_0
a,x_1=a+h，若 f(x_0)f(x_1)<0 表明扫描成
；否则令 x_0=x_1,x_1=x_0+h，继续上述方法，直到成
. 若 x_1>b 表明扫描失败，将 <span class="languag
-math">h 缩小（一般 h 按照 10 的倍数进行
小，从 0.1 到 0.01 再到 0.001 等等），重新开始扫描.</p>
<h3 id="3--简单迭代法">3. 简单迭代法</h3>
<p>若给定方程 f(x)=0 后，把它改写成等价形式：</p>
<div class="language-math">x=\phi(x)</div>
<p>再作迭代</p>
<div class="language-math">x_{n+1}=\phi(x_n)</div>
<h4 id="简单迭代法敛散性判别">简单迭代法敛散性判别</h4>

若 \lvert{\phi'(x)}\rvert<1，则简单迭代法收敛.</
i>
若 \phi'(\xi)=\cdots=\phi^{(p-1)}(\xi)=0,\phi^{(p)}(\xi)\no
=0,，则迭代在 \xi 处 p 阶收敛.

<h3 id="4--牛顿法">4. 牛顿法</h3>

```
<div class="language-math">x_{n+1}=x_n-\frac{f(x_n)}{f'(x_n)}</div>
<h3 id="5--弦截法">5. 弦截法</h3>
<div class="language-math">x_{n+1}=x_n-\frac{f(x_n)}{f(x_n)-f(x_{n-1})}(x_n-x_{n-1})</div>
<h2 id="第十一章-常微分方程初值问题的数值解法">第十一章 常微分方程初值问题的数值解法</h2

<h3 id="1--Euler-方法-显示欧拉法-">1. Euler 方法（显示欧拉法）</h3>
<div class="language-math">\begin{cases}
y_{n+1}=y_n+hf(x_n,y_n)\qquad n=0,1,2,\cdots,N-1 \\
y(x_0)=y_0 \\
y'=f(x,y)
\end{cases}</div>
<h3 id="2--向后-Euler-方法-隐式欧拉法-">2. 向后 Euler 方法（隐式欧拉法）</h3>
<div class="language-math">\begin{cases}
y_{n+1}=y_n+hf(x_{n+1},y_{n+1})\qquad n=0,1,2,\cdots \\
y(x_0)=y_0 \\
y'=f(x,y)
\end{cases}</div>
<h3 id="3--改进-Euler-方法">3. 改进 Euler 方法</h3>
<div class="language-math">\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))]\quad n=0,1,2,\cdots \\
y(x_0)=y_0 \\
y'=f(x,y)
\end{cases}</div>
<h3 id="4--梯形公式">4. 梯形公式</h3>
<div class="language-math">\begin{cases}
y_{n+1}^{(0)}=y_n+hf(x_n,y_n) \\
y_{n+1}^{(k+1)}=y_n+\frac{h}{2}[f(x_{n},y_{n})+f(x_{n+1},y_{n+1}^{(k)})]\quad k=0,1,2,\cdots \\
y'=f(x,y)
\end{cases}</div>
<blockquote>
<p><em><strong>注意</strong></em>：</p>
<ul>
<li>使用改进 Euler 方法进行迭代时，需要解隐式方程.</li>
<li>若题中不给出迭代步长 <span class="language-math">h</span>，则 <span class="langua
e-math">h=0.1</span>.</li>
</ul>
</blockquote>
```