目录

零点

二分法

$$x_{n+1}-x^*=\frac{x_0-x^*}{2^{n+1}}$$

牛顿法

$$x_{n+1}=\phi(x_n)=x_n-\frac{f(x_n)}{f^{(1)}(x_n)}$$

收敛速度判断

由于

$$\begin{aligned}\phi(x)&=x-\frac{f(x)}{f^{(1)}(x)}\\ \phi^{(1)}(x)&=1-\frac{f^{(1)}(x)^2-f(x)f^{(2)}(x)}{f^{(1)}(x)^2}\\ &=\frac{f(x)f^{(2)}(x)}{f^{(1)}(x)^2}\end{aligned}$$

且在零点 x^* 处

$$f(x^*)=0$$

得

$$\phi^{(1)}(x^*)=0$$

根据泰勒展开，将$\phi(x_n)$在零点x^*展开

$$\phi(x_n)=\phi(x^*)+\phi^{(1)}(x^*)(x_n-x^*)+\frac{1}{2!}\phi^{(2)}(x^*)(x_n-x^*)^2+\ldots$$

且

$$\begin{aligned}x_{n+1}&=\phi(x_n)\\ x^*&=\phi(x^*)\\ \phi^{(1)}(x^*)&=0\end{aligned}$$

得

$$\begin{aligned}x_{n+1}-x^*&=\phi(x_n)-\phi(x^*)\\ &\approx\phi^{(2)}(x^*)(x_n-x^*)^2\\ &\approx\phi^{(2)}(x^*)(\phi^{(2)}(x^*)(x_{n-1}-x^*)^2-x^*)^2\\ &=\ldots\ldots\\ &\approx(x_0-x^*)^{2^{n+1}}\end{aligned}$$

即

$$x_{n+1}-x^*\approx(x_0-x^*)^{2^{n+1}}$$

所以，牛顿法满足2阶收敛，收敛速度较快

重根问题

$f(x)$有m个重根，则$f(x)$可写作

$$f(x) = (x - x^*)^m$$

且

$$f^{(1)}(x^*) = f^{(2)}(x^*) = \ldots = f^{(m-1)}(x^*) = 0$$

此时，对于$\phi(x)$，有

$$\begin{aligned} \phi(x) &= x - \frac{f(x)}{f^{(1)}(x)} \\ &= x - \frac{x - x^*}{m} \\ \phi^{(1)}(x) &= 1 - \frac{1}{m} \end{aligned}$$

由于$\phi^{(1)}(x)$不为0，牛顿法退化为线性收敛(二分法)

改进

可改写递推式：

$$\phi(x) = x - m\frac{f(x)}{f^{(1)}(x)} = x^*$$
$$\phi^{(1)}(x) = 0$$

方程组求解

高斯消去法

上三角矩阵U与下三角矩阵L

计算

$$Ux = b$$
$$x_k = b_k - \sum_{j=k+1}^{n} U_{k,j} x_j$$

```
for (int k = n; 1 <= k; k--)
    for (int j = k+1; j <= n; j++)
        x[k] = b[k] - U[k][j] * x[j]
```

计算

$$Lx = b$$
$$x_k = b_k - \sum_{j=1}^{k-1} L_{k,j} x_j$$

```
for (int k = 1; k <= n; k++)
    for (int j = 1; j <= k-1; j++)
        x[k] = b[k] - L[k][j] * x[j]
```

原理

对于

$$Ax = b$$

使用高斯变换，乘以下三角矩阵L

$$LAx = Lb$$
$$Ux = Lb$$
$$x = U^{-1}Lb$$

因此关键在于，将系数矩阵A转化为上三角矩阵U

$$LA = U$$

即，将矩阵A分解为下三角矩阵L和上三角矩阵之积U

$$A = LU$$

称为矩阵LU分解

$$a_{k,p} = a_{k,p} - \frac{a_{k,j}}{a_{j,j}} * a_{j,p}$$

```
gauss_lu(double* a, int n)--> void {
    for(j = 1; j <= n; ++j){
        for(k = j+1; k <= n; ++k){
            temp = a[k][j] = a[k][j]/a[j][j];
            for(p =j+1; p <= n; ++p)
                a[k][p] = a[k][p] − a[j][p] ∗ temp;
        }
    }
}
```

列选主元高斯消去法

交换行：选取一列中最大值，然后交换。避免很大的数字对结果产生影响

```
guass_lu(double* a, int* p, int n)--> void
{
    p[1,n] = {1, 2, · · · , n};
    for(j = 1; j <= n; ++j) {
        // 找到a[j][j]到a[n][j]的最大值a[t][j]
        swap(a[j], a[t]);
        swap(p[j], p[t]);
        for(k = j+1; k <= n; ++k) {
            temp = a[k][j] = a[k][j]/a[j][j];
            for(s = j+1; s <= n; ++s)
                a[k][s] = a[k][s] − a[j][s] ∗ temp;
        }
```

```
}

```

完全选主元高斯消去法

交换行和列

误差分析

具体内容见讲义

前向误差

由高斯法求解得到x的近似值

$$x^*$$

判断相对误差

$$\frac{\delta(x)}{x}=\frac{x^*-x}{x}$$

的上界

向后舍入误差分析

存在ΔA满足

$$(A+\Delta A)x^*=b$$

上三角矩阵方程组数组解的舍入误差分析

$$|\Delta b|=|Ux^*-b|$$

列选主元高斯消去法*LU*分解的舍入误差分析

$$|\Delta A|=|A-LU|$$

完全选主元高斯消去法*LU*分解的舍入误差分析

$$|\Delta A|=|A-LU|$$

列选主元高斯消去法解方程组舍入误差分析

$$|\Delta b|=|Ax^*-b|$$

范数

将向量和矩阵映射到实数范围，比较大小

向量范数

- 壹范数：绝对值相加

$$\|x\ \|_1=\sum_{j=1}^{n}|x_j|$$

- 贰范数：平方值相加开根

$$\|x\|_2 = \sqrt{\sum_{j=1}^{n}|x_j|^2}$$

- p范数：p次方相加开p次根

$$\|x\|_p = \sqrt[p]{\sum_{j=1}^{n}|x_j|^p} = (\sum_{j=1}^{n}|x_j|^p)^{\frac{1}{p}}$$

- 无穷范数：最大值

$$\|x\|_\infty = max_{1\le j\le n}\{|x_j|\}$$

矩阵范数

- 行范数(无穷范数)

$$\|A\|_\infty = max_{1\le j\le n}\sum_{k=1}^{n}|a_{jk}|$$

- 列范数(壹范数)

$$\|A\|_1 = max_{1\le k\le n}\sum_{j=1}^{n}|a_{jk}|$$

- 贰范数(欧几里得范数、谱范数)

$$\|A\|_2 = \sqrt{\lambda_{max}(A^TA)} = \sqrt{\rho(A^TA)}$$

- p范数

$$\|A\|_p = max_{\|x\|_p=1}\|Ax\|_p$$

- F范数

$$\|A\|_F = \sqrt{\sum_{j,k=1}^{n}a_{jk}^2} = \sqrt{tr(A^TA)} = \sqrt{\sum_{j=1}^{n}\lambda_j(A^TA)}$$

迭代法

计算

$$Ax = b$$

使用迭代法

收敛判断

对于

$$x = Gx + b$$
$$\begin{cases} x^0 = \text{随机选取} \\ x^{k+1} = Gx^k + b \end{cases}$$

上面迭代格式的充要条件是$\rho(G) < 1$

Jacobi迭代

$$
\begin{gathered}
A=L+U+D \\
x=-D^{-1}(L+U)x+D^{-1}b
\end{gathered}
$$

计算格式

$$
\begin{gathered}
x_0=\text{初始向量} \\
x_{k+1}=D^{-1}(b-(L+U)x_k),k=0,1,2...
\end{gathered}
$$

如果A主对角线元行占优(或列占优)，则上面迭代格式收敛

Gauss-Seidel迭代

$$
\begin{gathered}
A=L+U+D \\
(L+D)x=-Ux+b
\end{gathered}
$$

计算格式

$$
\begin{gathered}
x_0=\text{初始向量} \\
x_{k+1}=D^{-1}(b-Ux_k-Lx_{k+1}),k=0,1,2...
\end{gathered}
$$

如果A

1. 主对角线元占优；或
2. 对称正定

则上面迭代格式收敛

代码

```
见讲义
```

逐次松弛迭代

使用*Gauss-Seidel*迭代的求解方向，并使用过松弛以加快收敛速度

令*w*是一个实数，将x_{k+1}定义为*w*乘上*Gauss-Seidel*公式和$1-w$乘上当前估计x_k的平均。

*w*称为松弛参数，$w>1$时称为过松弛

$$
Dx_{k+1}=(1-w)Dx_k+w(b-Lx_{k+1}-Ux_k)
$$

计算格式

$$
\begin{gathered}
x_0=\text{初始向量} \\
x_{k+1}=(wL+D)^{-1}[(1-w)Dx_k-wUx_k]+w(D+wL)^{-1},b,k=0,1,2...
\end{gathered}
$$

如果*A*

1. 主对角线占优并且$w\epsilon(0,1]$；或者

2. 对称正定并且$w\epsilon(0,2)$

则上面迭代格式收敛

代码

```
见讲义
```

用于对称正定矩阵的方法

对称正定矩阵

对称矩阵只有一半数量的独立元素

$$A^T = A$$

是否能以一半的计算代价，并且仅仅使用一半的内存来求解

楚列斯基分解

如果*A*是对称正定矩阵，则存在上三角矩阵*R*满足

$$A = R^T R$$

共轭梯度方法

共轭

定义*A*内积

$$(v,w)_A = v^T A w$$

当$(v,w)_A = 0$时，向量*v*和*w*为*A*共轭

计算格式

$$
\begin{gathered}
x_0 = \text{初始估计} \\
p_0 = r_0 = b - Ax_0 \\
for\ k = 0,1,2,\ldots \\
\alpha_k = \frac{r_k^T r_k}{p_k^T A p_k} \\
x_{k+1} = x_k + \alpha_k p_k \\
r_{k+1} = r_k - \alpha_k A p_k \\
\beta_k = \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k} \\
p_{k+1} = r_{k+1} + \beta_k p_k
\end{gathered}
$$

向量x_k是第*k*步时的近似解. 向量r_k表示近似解x_k的余项

变量p_k表示用于更新x_k得到改进的x_{k+1}时所使用的新的搜索方向

对于α_k和β_k的选择

1. 选择α_k使得新的余项r_{k+1}和方向p_k正交，保证下一余项向量和前面所有的余项向量都正交，即$(r^{k+1}, r^k) = 0$
   $r^0, r^1, \ldots, r^{n-1}$是正交的
2. 选择β_k，保证p_{k+1}和$p_k$$A$共轭，即$(p_{k+1}, p_k)_A = 0$
   $p^0, p^1, \ldots, p^{n-1}$是$A$共轭的

代码

```
见讲义
```

插值

考虑函数

$$y = f(x)$$

在上面取点

$$(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$$

我们希望通过这些有限的点构造多项式，来模拟函数$f(x)$，即取值的逆过程，我们称为**插值**

拉格朗日插值

令

$$L_k(x) = \frac{(x-x_1)\ldots(x-x_{k-1})(x-x_{k+1})\ldots(x-x_n)}{(x_k-x_1)\ldots(x_k-x_{k-1})(x_k-x_{k+1})\ldots(x_k-x_n)}$$
$$L_k(x) = \begin{cases} 0 & x \neq x_k \\ 1 & x = x_k \end{cases}$$

构造多项式

$$P_{n-1}(x) = y_1 L_1(x) + \cdots + y_n L_n(x)$$

满足

$$P_{n-1}(x_k) = y_k$$

牛顿差商

令

$$f[x_k] = f(x_k)$$
$$f[x_1, \dots, x_k] = \frac{f[x_2, \dots, x_k] - f[x_1, \dots, x_{k-1}]}{x_k - x_1}$$

构造多项式

$$P(x) = f[x_1] + f[x_1x_2](x - x_1) + f[x_1x_2x_3](x - x_1)(x - x_2) + \dots + f[x_1 \dots x_n](x - x_1) \dots (x - x_{n-1})$$

插值误差

假设$P(x)$是$n-1$或者更低阶的插值多项式，拟合n个点$(x_1, y_1), \dots, (x_n, y_n)$，则误差满足

$$f(x) - P(x) = (x - x_1)(x - x_2) \dots (x - x_n) \frac{f^{(n)}(c)}{n!}$$
$$min(x_1, \dots, x_n) \le c \le max(x_1, \dots, x_n)$$

龙格现象

极端的“多项式扭动”，插值次数越高，插值结果越偏离原函数的现象

切比雪夫插值

考虑误差公式中的分子

$$(x - x_1)(x - x_2) \dots (x - x_n)$$

找到特定的$x_1, x_2, \dots, x_n$，使得该式足够小

切比雪夫多项式

定义n阶切比雪夫多项式

$$T_n(x) = cos(n \arccos x)$$

得到递推式

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

T_n主导系数为2^{n-1}，满足

$$deg(T_n) = n$$

即x的最高次数为n次，T_n有n个根，因此可写为

$$T_n(x) = 2^{n-1}(x - x_1)(x - x_2) \dots (x - x_n)$$

区间应用

在特定区间，选取切比雪夫多项式的根$x_1, x_2, \ldots, x_n$，

则$(x-x_1)(x-x_2)\ldots(x-x_n)$满足：

[-1,1]区间

$$x_i = cos\frac{(2i-1)\pi}{2n}, i=1,\ldots,n$$
$$(x-x_1)(x-x_2)\ldots(x-x_n) = \frac{T_n(x)}{2^{n-1}}$$

[a,b]区间

$$x_i = \frac{b+a}{2} + \frac{b-a}{2}cos\frac{(2i-1)\pi}{2n}, i=1,\ldots,n$$
$$|(x-x_1)(x-x_2)\ldots(x-x_n)| \leq \frac{(\frac{b-a}{2})^n}{2^{n-1}}$$

样条插值

在多项式插值中，多项式给出的单一公式满足所有数据点。而样条使用多个公式，其中每个都是低阶多项式，来通过所有数据点

我们定义*n*次样条，*n*为*x*的最高阶数

$$S_k(x) = y_k + c_1(x-x_k)^1 + c_2(x-x_k)^2 + \cdots + c_n(x-x_k)^n$$

三次样条

给定*n*个点

$$(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$$

通过这些点的三次样条$S(x)$是一组三次多项式：

$$\begin{gathered}
S_1(x) = y_1 + b_1(x-x_1) + c_1(x-x_1)^2 + d_1(x-x_1)^3 \\
S_2(x) = y_2 + b_2(x-x_2) + c_2(x-x_2)^2 + d_2(x-x_2)^3 \\
\vdots \\
S_{n-1}(x) = y_{n-1} + b_{n-1}(x-x_{n-1}) + c_{n-1}(x-x_{n-1})^2 + d_{n-1}(x-x_{n-1})^3
\end{gathered}$$

满足：

- 性质一：通过数值点

$$S_i(x_i) = y_i, S_i(x_{i+1}) = y_{i+1}, 其中 i = 1, \ldots, n-1$$

- 性质二：相邻的样条段斜率相同

$$S_i^{(1)}(x_i) = S_{i-1}^{(1)}(x_i), 其中 i = 2, \ldots, n-1$$

- 性质三：相邻的样条段曲率相同

$$S_i^{(2)}(x_i) = S_{i-1}^{(2)}(x_i), 其中 i = 2, \ldots, n-1$$

求解

对于通过n个点的三次样条方程组，一共有$3(n-1)=3n-3$个未知数

性质一包含$n-1$个方程，性质二和性质三各包含$n-2$个方程，一共有$n-1+2(n-2)=3n-5$个方程

所以一共有无穷个解。可以添加额外的方程来使解唯一化

自然样条

满足

$$S_1^{(2)}(x)=S_n^{(2)}(x)=0$$

这两个附加条件的三次样条被称为自然三次样条，这种条件称为自然样条的端点条件

通过更改端点条件可以得到不同的样条

贝塞尔曲线

贝塞尔样条允许用户控制节点处斜率，但是不再保证在节点导数的平滑性

最小二乘

考虑方程组

$$Ax=b$$

当解不存在时，可以试着找到近似解，即最小二乘近似

最小二乘解

法线方程

$$Ax=b$$

方程组可写为

$$x_1v_1+x_2v_2+...+x_nv_n=b$$

可以把b看作是A的列向量v_i的线性组合，对应的组合系数是$x_1,...,x_n$

即向量$v_1,v_2,...,v_n$生成的空间中找到组合系数$x_1,...,x_n$，使b在这个空间上

当解不存在时，在向量空间A={$v_1,v_2,...,v_n$}中存在与b最接近的点：
即存在$\overline{x}$，满足

$$b-A\overline{x}\perp_{平面}\{Ax|x\epsilon R^n\}$$

把垂直性表示为矩阵的乘法，对于R^n上所有的x，

$$(Ax)^T(b-A\overline{x})=0$$

可写为

$$x^TA^T(b-A\overline{x})=0$$

这意味着n维向量$A^T(b-A\overline{x})$和R^n中其他的n维向量垂直，这表明

$$A^T(b-A\overline{x})=0$$

即

$$A^TA\overline{x}=A^Tb$$

称为法线方程，它的解$\overline{x}$是方程组$Ax=b$的最小二乘解

余项

最小二乘解$\overline{x}$的余项

$$r=b-A\overline{x}$$

大小度量：

- 向量的欧氏长度(2范数)

$$\|r\|_2=\sqrt{r_1^2+\cdots+r_m^2}$$

- 平方误差

$$SE=r_1^2+\cdots+r_m^2$$

- 平均平方根误差

$$RMSE=\sqrt{\frac{SE}{m}}=\sqrt{\frac{r_1^2+\cdots+r_m^2}{m}}$$

QR分解

Gram-Schmidt正交

对一组向量正交化：给定一组输入的m维向量，找出正交坐标系统，获得由这些向量张成的空间

对于A向量和B向量，找到正交单位向量：

$$\begin{aligned} q_1 &= A \\ q_1 &= \frac{q_1}{\|q_1\|_1} \end{aligned}$$

接着找到B向量在A向量上的投影，然后用B向量减去该投影即可得到与A向量垂直的向量：

$$q_2=B-q_1(q_1^TB)$$

然后转化为单位向量

$$q_2=\frac{q_2}{\|q_2\|_2}$$

推广到多维向量，令$A_1,...,A_n$是R^m中的线性无关向量：

$$y_j=A_j-q_1(q_1^TA_j)-q_2(q_2^TA_j)-\cdots-q_{j-1}(q_{j-1}^TA_j)$$
$$q_j=\frac{y_j}{\|y_j\|_2}$$

对于上面的结果，引入新的符号：

$$r_{jj}=\parallel y_j\parallel_2$$
$$r_{ij}=q_i^TA_j$$

则

$$q_j=\frac{y_j}{r_{jj}}=\frac{A_j-q_1r_{1j}-q_2r_{2j}-\cdots-q_{j-1}r_{j-1,j}}{r_{jj}}$$

即

$$A_1=q_1r_{11}$$
$$A_2=q_1r_{12}+q_2r_{22}$$
$$\cdots$$
$$A_j=q_1r_{1j}+\cdots+q_{j-1}r_{j-1,j}+q_jr_{jj}$$

可写为

$$A_{m\times n}=Q_{m\times n}R_{n\times n}$$

称为消减QR分解

$$A_{m\times n}=Q_{m\times m}R_{m\times n}$$

称为完全QR分解

改进：用y代替A_j

$$y=A_j-q_1(q_1^Ty)-q_2(q_2^Ty)-\cdots-q_{j-1}(q_{j-1}^Ty)$$

Household变换

*Household*反射子是正交矩阵，通过$m-1$维平面反射m维向量。

即给定一个向量x，重新找出一个相同长度的向量w，计算*Household*反射得出矩阵H满足

$$Hx=w$$

定义向量$v=w-x$，考虑投影矩阵

$$P=\frac{vv^T}{v^Tv}$$

对于任何向量u，Pu是u在v上的投影

令$H=I-2P$，则

$$Hx=x-2Px=w-v-\frac{2vv^T}{v^Tv}x=w-\frac{vv^T(w+x)}{v^Tv}=w$$

矩阵H被称为**Household反射子**

对A_{mn}实现QR分解:

对于A的列向量，通过**Household反射子**将其移动到:

$$x = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} \Rightarrow w = \begin{bmatrix} \|x\|_2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

求出**Household反射子**:

$$v = w - x$$
$$P = \frac{vv^T}{v^T v}$$
$$H = I - 2P$$

依次对A做变换:

$$H_n \dots H_2 H_1 A = R$$
$$A = H_1 H_2 \dots H_n R = QR$$

最小二乘实现

实现最小二乘:
求解A的完全QR分解

$$A_{mn} = Q_{mm} R_{mn}$$

代入$Ax = b$，得

$$R_{mn} x_{n1} = Q_{mm}^T b_{m1}$$

即

$$\begin{bmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ & r_{22} & \dots & r_{2n} \\ & & \ddots & \vdots \\ & & & r_{nn} \\ 0 & \dots & \dots & 0 \\ \vdots & & & \vdots \\ 0 & \dots & \dots & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = (q_1 | \dots | q_m) \begin{bmatrix} b_1 \\ \vdots \\ b_n \\ b_{n+1} \\ \vdots \\ b_m \end{bmatrix}$$

化简，得:

$$\begin{bmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ & r_{22} & \dots & r_{2n} \\ & & \ddots & \vdots \\ & & & r_{nn} \\ 0 & \dots & \dots & 0 \\ \vdots & & & \vdots \\ 0 & \dots & \dots & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} d_1 \\ \vdots \\ d_n \\ d_{n+1} \\ \vdots \\ d_m \end{bmatrix}$$

分为两部分

$$\begin{bmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ & r_{22} & \dots & r_{2n} \\ & & \ddots & \vdots \\ & & & r_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} d_1 \\ \vdots \\ d_n \end{bmatrix} \tag{1}$$

$$\begin{bmatrix} 0 & \dots & \dots & 0 \\ \vdots & & & \vdots \\ 0 & \dots & \dots & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} d_{n+1} \\ \vdots \\ d_m \end{bmatrix} \tag{2}$$

简写为

$$R_{nn}\overline{x}_{n1} = d_{n1} \tag{1}$$

解$\overline{x}$即为最小二乘解

误差向量e满足

$$e = Ax - b = QRx - b = Rx - Q^T b = -\begin{bmatrix} 0 \\ \vdots \\ 0 \\ d_{n+1} \\ \vdots \\ d_m \end{bmatrix} \tag{2}$$

最小二乘误差大小为

$$\| e \|_2^2 = d_{n+1}^2 + \cdots + d_m^2$$

数值微分与积分

数值微分

有限差分公式

如果极限存在，函数$f(x)$在x点的导数是

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

计算时，存在误差。将$f(x+h)$在点x处泰勒展开：

$$f(x+h) = f(x) + hf'(x) + h^2 \frac{f''(c)}{2!}$$

得到**二点前向差分公式**

$$f'(x) = \frac{f(x+h) - f(x)}{h} - \frac{h}{2} f''(c) \\ \text{其中} x < c < x+h$$

n阶近似：误差是$O(h^n)$

将$f(x+h)$在点x处泰勒展开：

$$f(x+h)=f(x)+hf'(x)+h^2\frac{f''(c)}{2!}+h^3\frac{f'''(x)}{3!}+h^4\frac{f''''(c)}{4!}$$

将$f(x-h)$在点x处泰勒展开：

$$f(x-h)=f(x)-hf'(x)+h^2\frac{f''(c)}{2!}-h^3\frac{f'''(x)}{3!}+h^4\frac{f''''(c)}{4!}$$

两式相减，得到**三点中心差分公式**

$$f'(x)=\frac{f(x+h)-f(x-h)}{2h}-\frac{h^2}{6}f'''(c)$$
$$\text{其中} x-h<c<x+h$$

两式相加，得到**二阶导数的三点中心差分公式**

$$f''(x)=\frac{f(x-h)-2f(x)+f(x+h)}{h^2}-\frac{h^2}{12}f^{(4)}(c)$$

外推

假设有n阶公式$F(h)$近似一个给定量Q

$$Q\approx F(h)+Kh^n$$

代入$\frac{h}{2}$

$$Q-F(\frac{h}{2})\approx K\frac{h^n}{2^n}\approx\frac{1}{2^n}(Q-F(h))$$

得到**n阶公式的外推**

$$Q\approx\frac{2^nF(\frac{h}{2})-F(h)}{2^n-1}$$

数值积分

梯形法则

给定一组数据点，我们可以使用插值和最小二乘建模，模拟$f(x)$；现在，我们想要模拟函数$f(x)$在数据区间的积分

$$\int f(x)dx$$

我们可以通过模拟$f(x)$进而模拟$f(x)$在数据区间$[a,b]$的积分

对于二阶导数$f(x)$，使用1阶拉格朗日插值，通过$(x_0,y_0),(x_1,y_1)$

$$f(x)=y_0\frac{x-x_1}{x_0-x_1}+y_1\frac{x-x_0}{x_1-x_0}+\frac{(x-x_0)(x-x_1)}{2!}f''(c_x)$$
$$=P(x)+E(x)$$
$$c_x\text{连续依赖于}x$$

在数据区间$[x_0,x_1]$的两侧积分得到

$$\int_{x_0}^{x_1}f(x)dx=\int_{x_0}^{x_1}P(x)dx+\int_{x_0}^{x_1}E(x)dx$$

而

$$\int_{x_0}^{x_1} P(x)dx = y_0\int_{x_0}^{x_1} \frac{x-x_1}{x_0-x_1}dx + y_1\int_{x_0}^{x_1} \frac{x-x_0}{x_1-x_0}dx$$
$$= y_0\frac{h}{2} + y_1\frac{h}{2} = h\frac{y_0+y_1}{2}$$

即**梯形法则**

$$\int_{x_0}^{x_1} f(x)dx = h\frac{y_0+y_1}{2} - \frac{h^3}{12}f''(c)$$
$$\text{其中} h = x_1 - x_0, c\text{在} x_0\text{和} x_1\text{之间}$$

辛普森法则

对于二阶导数$f(x)$，使用2阶拉格朗日插值，通过$(x_0, y_0), (x_1, y_1), (x_2, y_2)$

积分得到

$$\int_{x_0}^{x_1} f(x)dx = \frac{h}{3}(y_0 + 4y_1 + y_2) - \frac{h^5}{90}f^{(4)}(c)$$
$$\text{其中} h = x_2 - x_1 = x_1 - x_0, c\text{在} x_0\text{和} x_2\text{之间}$$

复合牛顿-科特斯公式

梯形和辛普森法则都在单一区间上操作，我们可以通过除法把整个区间变为很多小区间再计算积分

复合数值积分：在每个小区间上使用法则，然后再求和

复合梯形法则

$$\int_a^b f(x)dx = \frac{h}{2}(y_0 + y_m + 2\sum_{i=1}^{m-1} y_i) - \frac{(b-a)h^2}{12}f''(x)$$
$$\text{其中} h = \frac{b-a}{m}, c\text{在} a\text{和} b\text{之间}$$

复合辛普森公式

$$\int_a^b f(x)dx = \frac{h}{3}[y_0 + y_{2m} + 4\sum_{i=1}^{m} y_{2i-1} + 2\sum_{i=1}^{m-1} y_{2i}] - \frac{(b-a)h^4}{180}f^{(4)}(c)$$

开牛顿-科特斯公式

对于闭牛顿-科特斯公式，需要积分区间端点上的函数值；而开牛顿-科特斯公式不使用在区间端点上的函数值

中点法则

$$\int_{x_0}^{x_1} f(x)dx = hf(w) + \frac{h^3}{24}f''(c)$$
$$\text{其中} h = (x_1 - x_0), w\text{是中点} x_0 + \frac{h}{2}, c\text{在} x_0\text{和} x_1\text{之间}$$

复合重点法则

$$\int_b^a f(x)dx = h\sum_{i=1}^{m} f(w_i) + \frac{(b-a)h^2}{24}f''(c)$$

$$其中 h = \frac{b-a}{m}, c在a和b之间, w_i是[a,b]中m个相等子区间的中点$$

特征值与奇异值

线性变换

如果从线性变换的角度看待矩阵，矩阵作用在向量上的效果有两种：

- 旋转

考虑向量旋转矩阵(**单位正交矩阵**)，改变向量方向，但不改变大小

$$A = \begin{bmatrix} cos\theta & sin\theta \\ -sin\theta & cos\theta \end{bmatrix}$$

当$\theta = 0$时，A为单位矩阵，$Ax = Ex = x$，向量x方向不变

当$\theta = 45^\circ$时，Ax即向量x旋转45°

- 伸缩

考虑**对角矩阵**，改变向量大小，但不改变方向。与之有关的是**特征向量**与**特征值**

$$A = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_n \end{bmatrix}$$

对角线上的$\lambda_1, \lambda_2, \dots, \lambda_n$即为特征值，代表各个方向伸缩的比例

特征值与特征向量

N维非零向量x是$N \times N$的矩阵A的特征向量，当且仅当下式成立：

$$Ax = \lambda x$$

其中x称为特征向量，λ为一标量，称为x对应的特征值。即特征向量被施以线性变换A只会使向量伸长或缩短而其方向不被改变

对于一个矩阵A，有一组特征向量；再将这组向量进行正交化单位化，也就是我们学过的Gram-Schmidt正交化，就能得到一组正交单位向量。

特征值分解，就是将矩阵A分解为如下方式：

$$A = P\Lambda P^{-1}$$

P为矩阵A的特征向量组成的矩阵，Λ为对角矩阵，对角线上的元素就是特征值

特征值分解可以得到特征值与特征向量，

- 特征值表示的是这个特征到底有多重要，即变化大小
- 特征向量表示这个特征是什么，可以将每一个特征向量理解为一个线性的子空间，即变化方向

幂迭代方法

对于矩阵A的特征值$\lambda_1,\lambda_2,...,\lambda_n$，最大的特征值称为**占优特征向量**

幂迭代

对于矩阵A的特征值$\lambda_1,\lambda_2,...,\lambda_n$，及其对应的特征向量$x_1,x_2,...,x_n$

选取一个随机向量α，将其表示为特征向量的线性组合:

$$\alpha = k_1x_1 + k_2x_2 + \cdots + k_nx_n$$

乘上矩阵A对向量α进行线性变换(利用公式$A = P\Lambda P^{-1}$):

$$\begin{aligned}A\alpha = A\left[\,x_1,x_2,\dots,x_n\,\right]\begin{bmatrix}k_1\\k_2\\\vdots\\k_n\end{bmatrix} = AP\begin{bmatrix}k_1\\k_2\\\vdots\\k_n\end{bmatrix} = P\Lambda\begin{bmatrix}k_1\\k_2\\\vdots\\k_n\end{bmatrix}\\ = \left[\,x_1,x_2,\dots,x_n\,\right]\begin{bmatrix}\lambda_1\\\lambda_2\\\vdots\\\lambda_n\end{bmatrix}\begin{bmatrix}k_1\\k_2\\\vdots\\k_n\end{bmatrix}\\ = k_1\lambda_1x_1 + k_2\lambda_2x_2 + \cdots + k_n\lambda_nx_n\end{aligned}$$

$$\begin{gathered}A^2a = k_1\lambda_1^2x_1 + k_2\lambda_2^2x_2 + \cdots + k_n\lambda_n^2x_n\\ \vdots\\ A^ma = k_1\lambda_1^mx_1 + k_2\lambda_2^mx_2 + \cdots + k_n\lambda_n^mx_n\end{gathered}$$

当$m\to\infty$时，改写该式:

$$\frac{A^ma}{\lambda_1^m} = k_1x_1 + k_2(\frac{\lambda_2}{\lambda_1})^mx_2 + \cdots + k_n(\frac{\lambda_n}{\lambda_1})^mx_n$$

可知，对于最大特征值λ_1，该式收敛到x_1的某个倍数。通过多次迭代，我们可以找到矩阵A的占优特征向量

幂迭代局限于求解最大(绝对值最大)的特征值，我们可以利用这个性质，找到其他特征向量

- 最小特征向量：对矩阵的逆矩阵A^{-1}进行幂迭代，对所得到的特征值b计算倒数b^{-1}，就得到矩阵A的最小特征值

  A的逆矩阵A^{-1}的特征值为$\lambda_1^{-1},\lambda_2^{-1},...,\lambda_n^{-1}$，特征向量与$A$相同
- 其他特征向量：对矩阵的转移矩阵$(A-sI)^{-1}$进行幂迭代，得到最大特征值b，则$\lambda = b^{-1}+s$为矩阵A在s附近的特征值

  A的转移矩阵$A-sI$的特征值为$\lambda_1-s,\lambda_2-s,...,\lambda_n-s$，特征向量和矩阵$A$的特征向量相同

Rayleigh商

已知矩阵A和近似特征向量，找到特征值的最优估计。考虑特征值方程

$$x\lambda = Ax$$

通过法线方程找到最小二乘解

$$x^T x\lambda = x^T Ax$$
$$\lambda = \frac{x^T Ax}{x^T x}$$

即**Rayleigh**商，对归一化的特征向量使用Rayleigh商可以得到特征值近似

QR算法