非对称特征值问题计算方法

邓建松

2018 年 11 月 28 日

问题分析

- 求一个矩阵的特征值的问题实质上是求一个多项式的根的问题

问题分析

- 求一个矩阵的特征值的问题实质上是求一个多项式的根的问题
- 五阶及五阶以上的多项式的根一般不能用有限次代数运算求得

问题分析

- 求一个矩阵的特征值的问题实质上是求一个多项式的根的问题
- 五阶及五阶以上的多项式的根一般不能用有限次代数运算求得
- 所以矩阵特征值的计算方法本质上都是迭代的

问题分析

- 求一个矩阵的特征值的问题实质上是求一个多项式的根的问题
- 五阶及五阶以上的多项式的根一般不能用有限次代数运算求得
- 所以矩阵特征值的计算方法本质上都是迭代的
- 已有不少非常成熟的数值方法计算矩阵的全部或部分特征值和特征向量

问题分析

- 求一个矩阵的特征值的问题实质上是求一个多项式的根的问题
- 五阶及五阶以上的多项式的根一般不能用有限次代数运算求得
- 所以矩阵特征值的计算方法本质上都是迭代的
- 已有不少非常成熟的数值方法计算矩阵的全部或部分特征值和特征向量
- 本节只是介绍几种最常用的基本方法

基本概念

- 设$A \in \mathbb{C}^{n\times n}$, 复数$\lambda$称为$A$的一个特征值是指存在非零向量$x \in \mathbb{C}^n$使得$Ax = \lambda x$

基本概念

- 设$A \in \mathbb{C}^{n\times n}$, 复数$\lambda$称为$A$的一个特征值是指存在非零向量$x \in \mathbb{C}^n$使得$Ax = \lambda x$
- 此时x称做A的属于λ的一个特征向量

基本概念

- 设$A \in \mathbb{C}^{n\times n}$, 复数$\lambda$称为$A$的一个特征值是指存在非零向量$x \in \mathbb{C}^n$使得$Ax = \lambda x$
- 此时x称做A的属于λ的一个特征向量
- λ是A的一个特征向量当且仅当$\det(\lambda I - A) = 0$

基本概念

- 设$A \in \mathbb{C}^{n\times n}$, 复数$\lambda$称为$A$的一个特征值是指存在非零向量$x \in \mathbb{C}^n$使得$Ax = \lambda x$
- 此时x称做A的属于λ的一个特征向量
- λ是A的一个特征向量当且仅当$\det(\lambda I - A) = 0$
- 多项式$p_A(\lambda) = \det(\lambda I - A)$称为$A$的特征多项式

特征多项式

- $p_A(\lambda)$是一个首一的n次多项式

特征多项式

- $p_A(\lambda)$是一个首一的n次多项式
- 由代数基本定理知$p_A(\lambda)$有n个根，即A有n个特征值

特征多项式

- $p_A(\lambda)$是一个首一的n次多项式
- 由代数基本定理知$p_A(\lambda)$有n个根，即A有n个特征值
- 记A的特征值全体为$\lambda(A)$, 称之为A的谱集

特征多项式

- $p_A(\lambda)$是一个首一的n次多项式
- 由代数基本定理知$p_A(\lambda)$有n个根，即A有n个特征值
- 记A的特征值全体为$\lambda(A)$, 称之为A的谱集
- 假设$p_A(\lambda)=\prod_{j=1}^{r}(\lambda-\lambda_j)^{n_j}$, 其中$n_1+\cdots+n_r=n$, $\lambda_i\neq\lambda_j$ $(i\neq j)$, 则称n_i为λ_i的代数重数, 而称$m_i=n-\mathrm{rank}(\lambda_i I-A)$为$\lambda_i$的几何重数

- 显然$m_i \leqslant n_i$

- 显然$m_i \leqslant n_i$
- 如果$n_i = 1$, 则称λ_i为A的一个单特征值；否则称λ_i是A的一个重特征值

- 显然$m_i \leqslant n_i$
- 如果$n_i = 1$, 则称λ_i为A的一个单特征值；否则称λ_i是A的一个重特征值
- 如果$n_i = m_i$, 则称λ_i为A的一个半单特征值

- 显然$m_i \leqslant n_i$
- 如果$n_i = 1$, 则称λ_i为A的一个单特征值；否则称λ_i是A的一个重特征值
- 如果$n_i = m_i$, 则称λ_i为A的一个半单特征值
- 如果A的所有特征值都是半单的，则称A是非亏损的

- 显然$m_i \leqslant n_i$
- 如果$n_i = 1$, 则称λ_i为A的一个单特征值；否则称λ_i是A的一个重特征值
- 如果$n_i = m_i$, 则称λ_i为A的一个半单特征值
- 如果A的所有特征值都是半单的，则称A是非亏损的
- A是非亏损的当且仅当A有n个线性无关的特征向量，即A是可对角化的

相似变换

- 设$A, B \in \mathbb{C}^{n\times n}$. 若存在非奇异阵$X \in \mathbb{C}^{n\times n}$使得$B = XAX^{-1}$, 则称$A$与$B$是相似的，而上述变换称为相似变换

相似变换

- 设$A, B \in \mathbb{C}^{n\times n}$. 若存在非奇异阵$X \in \mathbb{C}^{n\times n}$使得$B = XAX^{-1}$, 则称$A$与$B$是相似的，而上述变换称为相似变换
- 若A与B相似，则它们有相同的特征值，而且x是A的一个特征向量当且仅当$y = Xx$是B 的一个特征向量

相似变换

- 设$A, B \in \mathbb{C}^{n\times n}$. 若存在非奇异阵$X \in \mathbb{C}^{n\times n}$使得$B = XAX^{-1}$, 则称$A$与$B$是相似的，而上述变换称为相似变换
- 若A与B相似，则它们有相同的特征值，而且x是A的一个特征向量当且仅当$y = Xx$是B 的一个特征向量
- 矩阵特征值问题的基本思想：把给定矩阵相似变换为易于计算特征值和特征向量的矩阵

Jordan分解定理

- 设$A \in \mathbb{C}^{n\times n}$有$r$个互不相同的特征值$\lambda_1, \ldots, \lambda_r$，其（代数）重数分别为$n(\lambda_1), \ldots, n(\lambda_r)$, 则存在非奇异矩阵$P \in \mathbb{C}^{n\times n}$

$$P^{-1}AP = \mathrm{diag}(J(\lambda_1), \ldots, J(\lambda_r))$$

其中

$$J(\lambda_i) = \mathrm{diag}(J_1(\lambda_i), \ldots, J_{k_i}(\lambda_i)) \in \mathbb{C}^{n(\lambda_i)\times n(\lambda_i)}$$

Jordan块

$$J_j(\lambda_i) = \begin{pmatrix} \lambda_i & 1 & & \\ & \lambda_i & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_i \end{pmatrix} \in \mathbb{C}^{n_j(\lambda_i)\times n_j(\lambda_i)}$$

其中

$$n_1(\lambda_i) + \cdots + n_{k_i}(\lambda_i) = n(\lambda_i)$$

Schur分解定理

- 设$A \in \mathbb{C}^{n\times n}$，则存在酉矩阵$U \in \mathbb{C}^{n\times n}$ 使得

$$U^* A U = T$$

其中T是上三角阵。适当选取U可使T的对角元按任意指定的顺序排列

Schur分解定理

- 设$A \in \mathbb{C}^{n\times n}$，则存在酉矩阵$U \in \mathbb{C}^{n\times n}$ 使得

$$U^* A U = T$$

  其中T是上三角阵。适当选取U可使T的对角元按任意指定的顺序排列

- 特征值问题求解的QR方法就是基于这一定理而设计的

Gerschgorin圆盘定理

- 设$A = (a_{ij}) \in \mathbb{C}^{n\times n}$，令

$$G_i(A) = \left\{ z \in \mathbb{C} : |z - a_{ii}| \leqslant \sum_{j\neq i} |a_{ij}| \right\}$$

则有

$$\lambda(A) = \bigcup_{j=1}^{n} G_j(A)$$

左、右特征向量

- 设λ是矩阵$A \in \mathbb{C}^{n\times n}$的一个特征值

左、右特征向量

- 设λ是矩阵$A \in \mathbb{C}^{n\times n}$的一个特征值
- 则关于$x \in \mathbb{C}^n$的方程$Ax = \lambda x$有解，关于$y \in \mathbb{C}^n$的方程$y^T A = \lambda y^T$也有解

左、右特征向量

- 设λ是矩阵$A \in \mathbb{C}^{n\times n}$的一个特征值
- 则关于$x \in \mathbb{C}^n$的方程$Ax = \lambda x$有解，关于$y \in \mathbb{C}^n$的方程$y^T A = \lambda y^T$也有解
- 显然x是前面所定义的相应于λ的特征向量

左、右特征向量

- 设λ是矩阵$A \in \mathbb{C}^{n\times n}$的一个特征值
- 则关于$x \in \mathbb{C}^n$的方程$Ax = \lambda x$有解，关于$y \in \mathbb{C}^n$的方程$y^T A = \lambda y^T$也有解
- 显然x是前面所定义的相应于λ的特征向量
- 我们称x是相应于λ的右特征向量；而y是相应于λ的左特征向量

左、右特征向量

- 设λ是矩阵$A \in \mathbb{C}^{n\times n}$的一个特征值
- 则关于$x \in \mathbb{C}^n$的方程$Ax = \lambda x$有解，关于$y \in \mathbb{C}^n$的方程$y^T A = \lambda y^T$也有解
- 显然x是前面所定义的相应于λ的特征向量
- 我们称x是相应于λ的右特征向量；而y是相应于λ的左特征向量
- y是矩阵A^T相应于λ的右特征向量

左、右特征向量

- 设x_i是A对应于特征值λ_i的右特征向量，y_j是A对应于特征值λ_j的左特征向量

左、右特征向量

- 设x_i是A对应于特征值λ_i的右特征向量，y_j是A对应于特征值λ_j的左特征向量
- 若$\lambda_i \neq \lambda_j$，则必有$y_j^T x_i = 0$. 注意：即使是在复数域上进行运算，这里也只是进行矩阵转置，没有共轭(对于非零复向量x, 可能有$x^T x = 0$)

左、右特征向量

- 设x_i是A对应于特征值λ_i的右特征向量，y_j是A对应于特征值λ_j的左特征向量
- 若$\lambda_i \neq \lambda_j$，则必有$y_j^T x_i = 0$. 注意：即使是在复数域上进行运算，这里也只是进行矩阵转置，没有共轭(对于非零复向量x, 可能有$x^T x = 0$)
- 实际上，在$Ax_i = \lambda_i x_i$上左乘y_j^T；在$y_j^T A = \lambda_j y_j^T$上右乘$x_i$, 然后两式相减，即得

$$(\lambda_i - \lambda_j) y_j^T x_i = 0$$

左、右特征向量

- 设x_i是A对应于特征值λ_i的右特征向量，y_j是A对应于特征值λ_j的左特征向量
- 若$\lambda_i \neq \lambda_j$，则必有$y_j^T x_i = 0$. 注意：即使是在复数域上进行运算，这里也只是进行矩阵转置，没有共轭(对于非零复向量x, 可能有$x^T x = 0$)
- 实际上，在$Ax_i = \lambda_i x_i$上左乘y_j^T; 在$y_j^T A = \lambda_j y_j^T$上右乘$x_i$, 然后两式相减，即得

$$(\lambda_i - \lambda_j) y_j^T x_i = 0$$

- 若$\lambda_i = \lambda_j$，结论如何？

多项式的友阵

- 给定首一n次多项式

$$p(x) = x^n + p_{n-1}x^{n-1} + \cdots + p_1 x + p_0$$

多项式的友阵

- 给定首一n次多项式

$$p(x) = x^n + p_{n-1}x^{n-1} + \cdots + p_1 x + p_0$$

- 容易验证矩阵

$$C = \begin{pmatrix} -p_{n-1} & -p_{n-2} & \cdots & -p_1 & -p_0 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{pmatrix}$$

的特征多项式就是$p(x)$

友阵

- C称为$p(x)$的友阵

友阵

- C称为$p(x)$的友阵
- 那么C的Jordan标准型是什么？

友阵

非对称特征值问题计算方法

邓建松

基本概念与性质

幂法

反幂法

QR方法

基本迭代和收敛性

实Schur标准形

上Hessenberg化

带原点位移的QR迭代

双重步位移的QR迭代

隐式QR算法

- C称为$p(x)$的友阵
- 那么C的Jordan标准型是什么？
- 或者问：当矩阵A的特征多项式是$p(x)$时，何时A与C是相似的？

友阵

- C称为$p(x)$的友阵
- 那么C的Jordan标准型是什么？
- 或者问：当矩阵A的特征多项式是$p(x)$时，何时A与C是相似的？
- 结论：若对应于不同特征值只有一个Jordan块，或者说只有一个特征向量时，矩阵A相似于C（参考：J.H. Wilkinson的著作【代数特征值问题】）

特征值和特值向量敏感性分析

- 从数值计算的角度来看，我们应首先弄清要计算的特征值和特征向量是否是病态的，即矩阵元素的小变化，是否会引起所关心的特征值和特征向量的巨大变化

特征值和特值向量敏感性分析

- 从数值计算的角度来看，我们应首先弄清要计算的特征值和特征向量是否是病态的，即矩阵元素的小变化，是否会引起所关心的特征值和特征向量的巨大变化
- 对一般矩阵而言，这一问题非常复杂

特征值和特值向量敏感性分析

- 从数值计算的角度来看，我们应首先弄清要计算的特征值和特征向量是否是病态的，即矩阵元素的小变化，是否会引起所关心的特征值和特征向量的巨大变化
- 对一般矩阵而言，这一问题非常复杂
- 我们这里只介绍一个简单而又非常重要的结果

- 假设λ是A的一个单特征值，x是相应的特征向量，而且$\|x\|_2 = 1$

- 假设λ是A的一个单特征值，x是相应的特征向量，而且$\|x\|_2 = 1$
- 令$U = (x, U_2) \in \mathbb{C}^{n\times n}$是酉阵，则我们有

$$U^* A U = \begin{pmatrix} \lambda & x^* A U_2 \\ 0 & A_2 \end{pmatrix}$$

- 假设λ是A的一个单特征值，x是相应的特征向量，而且$\|x\|_2 = 1$
- 令$U = (x, U_2) \in \mathbb{C}^{n\times n}$是酉阵，则我们有

$$U^* AU = \begin{pmatrix} \lambda & x^* AU_2 \\ 0 & A_2 \end{pmatrix}$$

- 由于λ是A的单特征值，所以

$$\delta = \min_{\mu\in\lambda(A_2)} |\lambda - \mu| > 0$$

- 定义$\Sigma^{\perp} = U_2(\lambda I - A_2)^{-1} U_2^*$

- 定义$\Sigma^\perp = U_2(\lambda I - A_2)^{-1}U_2^*$
- 取y为A的属于λ的左特征向量(即$y^TA = \lambda y^T$)

- 定义$\Sigma^{\perp} = U_2(\lambda I - A_2)^{-1} U_2^*$
- 取y为A的属于λ的左特征向量(即$y^T A = \lambda y^T$)
- 由于λ是单特征值，所以$y^T x \neq 0$. (为什么？) 从而可以取y使得$y^T x = 1$

- 定义$\Sigma^{\perp} = U_2(\lambda I - A_2)^{-1}U_2^*$
- 取y为A的属于λ的左特征向量(即$y^TA = \lambda y^T$)
- 由于λ是单特征值，所以$y^Tx \neq 0$. (为什么？)
  从而可以取y使得$y^Tx = 1$
    - 利用Jordan标准形考虑特征向量的形式

- 定义$\Sigma^{\perp} = U_2(\lambda I - A_2)^{-1}U_2^*$
- 取y为A的属于λ的左特征向量(即$y^TA = \lambda y^T$)
- 由于λ是单特征值，所以$y^Tx \neq 0$. (为什么？)
  从而可以取y使得$y^Tx = 1$
  - 利用Jordan标准形考虑特征向量的形式
- 给矩阵A以微小的扰动使其变为$\tilde{A}$,
  记$\varepsilon = \|\tilde{A} - A\|_2$, 则存在$\tilde{A}$的一个特征值$\tilde{\lambda}$和对应的特征向量$\tilde{x}$, 使得(证明略)

  $$|\tilde{\lambda}-\lambda| \leqslant \|y\|_2\varepsilon+O(\varepsilon^2), \|\tilde{x}-x\|_2 \leqslant \|\Sigma^{\perp}\|_2\varepsilon+O(\varepsilon^2)$$

条件数

- 上式表明λ和x的敏感性分别与$\|y\|_2$和$\|\Sigma^{\perp}\|_2$的大小有关

条件数

- 上式表明λ和x的敏感性分别与$\|y\|_2$和$\|\Sigma^{\perp}\|_2$的大小有关
- 我们分别称$\|y\|_2$和$\|\Sigma^{\perp}\|_2$为特征值λ和特征向量x的条件数. 记做

$$\operatorname{cond}(\lambda) = \|y\|_2, \quad \operatorname{cond}(x) = \|\Sigma^{\perp}\|_2$$

幂法：初步假设

- 幂法是计算矩阵模最大特征值和对应特征向量的一种迭代方法

幂法：初步假设

- 幂法是计算矩阵模最大特征值和对应特征向量的一种迭代方法
- 为了说明基本想法，我们先假定$A \in \mathbb{C}^{n\times n}$可对角化，即$A$有如下分解

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

其中$\Lambda = \mathrm{diag}(\lambda_1, \dots, \lambda_n)$,
$X = (x_1, \dots, x_n) \in \mathbb{C}^{n\times n}$ 非奇异

幂法：初步假设

- 幂法是计算矩阵模最大特征值和对应特征向量的一种迭代方法
- 为了说明基本想法，我们先假定$A \in \mathbb{C}^{n\times n}$可对角化，即$A$有如下分解

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

其中$\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$,
$X = (x_1, \ldots, x_n) \in \mathbb{C}^{n\times n}$ 非奇异

- 再假定$|\lambda_1| > |\lambda_2| \geqslant \cdots \geqslant |\lambda_n|$

开始迭代

- 任取一向量$u_0 \in \mathbb{C}^n$

开始迭代

- 任取一向量$u_0 \in \mathbb{C}^n$
- 由于X的列向量构成$\mathbb{C}^n$的一组基，所以有

$$u_0 = \sum_{j=1}^{n} \alpha_i x_i, \quad \alpha_i \in \mathbb{C}$$

开始迭代

- 任取一向量$u_0 \in \mathbb{C}^n$
- 由于X的列向量构成$\mathbb{C}^n$的一组基，所以有

$$u_0 = \sum_{j=1}^{n} \alpha_i x_i, \quad \alpha_i \in \mathbb{C}$$

- 从而我们有

$$\begin{aligned} A^k u_0 &= \sum_{j=1}^{n} \alpha_j A^k x_j = \sum_{j=1}^{n} \alpha_j \lambda_j^k x_j \\ &= \lambda_1^k \left(\alpha_1 x_1 + \sum_{j=2}^{n} \alpha_j \left(\frac{\lambda_j}{\lambda_1} \right)^k x_j \right) \end{aligned}$$

- 由此即知

$$\lim_{k\to\infty} \frac{A^k u_0}{\lambda_1^k} = \alpha_1 x_1$$

- 由此即知
$$\lim_{k\to\infty}\frac{A^k u_0}{\lambda_1^k}=\alpha_1 x_1$$
- 这表明当$\alpha_1\neq 0$且k充分大时，向量$u_k=A^k u_0/\lambda_1^k$就是A的一个很好的近似特征向量

实际应用的难处

- 我们事先并不知道A的特征值λ_1

实际应用的难处

- 我们事先并不知道A的特征值λ_1
- 对充分大的k，A^k的计算工作量太大

实际应用的难处

- 我们事先并不知道A的特征值λ_1
- 对充分大的k, A^k的计算工作量太大
- 因此直接应用上式计算A的近似特征值有困难

变通

- 在上式中λ_1^k只是改变向量的长度，但不改方向，因此我们不必用λ_1^k约化$A^k u_0$

变通

- 在上式中λ_1^k只是改变向量的长度，但不改方向，因此我们不必用λ_1^k约化$A^k u_0$
- 但这种约化也是必要的，以防止溢出

变通

- 在上式中λ_1^k只是改变向量的长度，但不改方向，因此我们不必用λ_1^k约化$A^k u_0$
- 但这种约化也是必要的，以防止溢出
- 在计算$A^k u_0$时也不必先算出A^k，只需迭代地进行就可以了

幂法的迭代格式

- 任意取定$u_0 \in \mathbb{C}^n$为初始向量，通常要求$\|u_0\|_\infty = 1$; $k \leftarrow 1$

幂法的迭代格式

- 任意取定$u_0 \in \mathbb{C}^n$为初始向量，通常要求$\|u_0\|_\infty = 1$; $k \leftarrow 1$
- $y_k = Au_{k-1}$

幂法的迭代格式

- 任意取定$u_0 \in \mathbb{C}^n$为初始向量，通常要求$\|u_0\|_\infty = 1$; $k \leftarrow 1$
- $y_k = Au_{k-1}$
- $\mu_k = \zeta_j^{(k)}$, $\zeta_j^{(k)}$是y_k的模最大分量

幂法的迭代格式

- 任意取定$u_0 \in \mathbb{C}^n$为初始向量，通常要求$\|u_0\|_\infty = 1$; $k \leftarrow 1$
- $y_k = Au_{k-1}$
- $\mu_k = \zeta_j^{(k)}$, $\zeta_j^{(k)}$是y_k的模最大分量
- $u_k = y_k/\mu_k$

幂法的迭代格式

- 任意取定$u_0 \in \mathbb{C}^n$为初始向量，通常要求$\|u_0\|_\infty = 1$; $k \leftarrow 1$
- $y_k = Au_{k-1}$
- $\mu_k = \zeta_j^{(k)}$, $\zeta_j^{(k)}$是y_k的模最大分量
- $u_k = y_k/\mu_k$
- $k \leftarrow k+1$, 返回第二步

幂法的迭代格式

- 任意取定$u_0 \in \mathbb{C}^n$为初始向量，通常要求$\|u_0\|_\infty = 1$; $k \leftarrow 1$
- $y_k = Au_{k-1}$
- $\mu_k = \zeta_j^{(k)}$, $\zeta_j^{(k)}$是y_k的模最大分量
- $u_k = y_k / \mu_k$
- $k \leftarrow k + 1$, 返回第二步
- 这一迭代方法称做**幂法**

幂法的收敛性定理

定理

设$A \in \mathbb{C}^{n\times n}$有$r$个互不相同的特征值，且满足$|\lambda_1| > |\lambda_2| \geqslant \cdots \geqslant |\lambda_r|$, 而且$\lambda_1$是半单的。如果初始向量$u_0$在$\lambda_1$的特征子空间上的投影不为零，则由幂法产生的向量序列$\{u_k\}$收敛到λ_1的一个特征向量x_1（极限平行于x_0在特征子空间上投影向量），而且$\{\mu_k\}$收敛到λ_1

收敛定理的证明

- 根据假设，A的Jordan分解为

$$A = X \operatorname{diag}(J_1, \ldots, J_r) X^{-1}$$

其中$X \in \mathbb{C}^{n\times n}$非奇异，$J_i$是由属于$\lambda_i$的Jordan块构成的上三角阵，$n_1 + \cdots + n_r = n$. 由于$\lambda_1$是半单的，所以$J_1 = \lambda_1 I_{n_1}$

收敛定理的证明

- 根据假设，A的Jordan分解为

$$A = X\,\mathrm{diag}(J_1, \ldots, J_r)X^{-1}$$

其中$X \in \mathbb{C}^{n\times n}$非奇异，$J_i$是由属于$\lambda_i$的Jordan块构成的上三角阵，$n_1 + \cdots + n_r = n$. 由于$\lambda_1$是半单的，所以$J_1 = \lambda_1 I_{n_1}$

- 令$y = X^{-1}u_0$, 并将y与X分块：

$$y = (y_1^T, y_2^T, \ldots, y_{n_r}^T)^T,$$

$$X = (X_1, X_2, \ldots, X_r)$$

其中y_i^T有n_i个元素，X_i有n_i列

- 从而我们有

$$
\begin{aligned}
A^k u_0 &= X \operatorname{diag}(J_1^k, \ldots, J_r^k) X^{-1} u_0 \\
&= X_1 J_1^k y_1 + X_2 J_2^k y_2 + \cdots + X_r J_r^k y_r \\
&= \lambda_1^k X_1 y_1 + X_2 J_2^k y_2 + \cdots + X_r J_r^k y_r \\
&= \lambda_1^k \left(X_1 y_1 + \sum_{j=2}^{r} X_j \left(\frac{J_j}{\lambda_1} \right)^k y_j \right)
\end{aligned}
$$

- 注意到$\lambda_1^{-1}J_i$的谱半径为$\rho(\lambda_1^{-1}J_i) = |\lambda_i|/|\lambda_1| < 1$, $i = 2, \ldots, r$, 所以我们有

$$\lim_{k\to\infty} \frac{1}{\lambda_1^k} A^k u_0 = X_1 y_1$$

- 注意到$\lambda_1^{-1}J_i$的谱半径为$\rho(\lambda_1^{-1}J_i) = |\lambda_i|/|\lambda_1| < 1$, $i = 2, \ldots, r$, 所以我们有
$$\lim_{k\to\infty} \frac{1}{\lambda_1^k} A^k u_0 = X_1 y_1$$
- 根据假设（u_0在λ_1的特征子空间上的投影不为零），我们有$X_1 y_1 \neq 0$

- 注意到$\lambda_1^{-1}J_i$的谱半径为$\rho(\lambda_1^{-1}J_i) = |\lambda_i|/|\lambda_1| < 1$, $i = 2, \ldots, r$, 所以我们有
$$\lim_{k\to\infty} \frac{1}{\lambda_1^k} A^k u_0 = X_1 y_1$$
- 根据假设（u_0在λ_1的特征子空间上的投影不为零），我们有$X_1 y_1 \neq 0$
  - $y = X^{-1}u_0 \Rightarrow u_0 = Xy$, 所以$y_1$即为$u_0$在$\lambda_1$的特征子空间上的组合系数，基向量就是$X_1$中各列向量，从而$X_1 y_1$就是$u_0$在这个特征子空间中的投影向量

- 由幂法产生的$\{u_k\}$满足

$$u_k = \frac{Au_{k-1}}{\mu_k} = \frac{A^k u_0}{\mu_k \mu_{k-1} \cdots \mu_1}$$

- 由幂法产生的$\{u_k\}$满足
$$u_k = \frac{Au_{k-1}}{\mu_k} = \frac{A^k u_0}{\mu_k \mu_{k-1} \cdots \mu_1}$$
- $\|u_k\|_\infty = 1$，即u_k至少有一个分量为1，所以$\zeta_k = \mu_k \mu_{k-1} \cdots \mu_1$必为$A^k u_0$的一个模最大分量

- 由幂法产生的$\{u_k\}$满足
$$u_k = \frac{Au_{k-1}}{\mu_k} = \frac{A^k u_0}{\mu_k \mu_{k-1} \cdots \mu_1}$$
- $\|u_k\|_\infty = 1$, 即u_k至少有一个分量为1, 所以$\zeta_k = \mu_k \mu_{k-1} \cdots \mu_1$必为$A^k u_0$的一个模最大分量
- 从而ζ_k / λ_1^k就是$A^k u_0 / \lambda_1^k$的一个模最大分量

- 由幂法产生的$\{u_k\}$满足

$$u_k = \frac{Au_{k-1}}{\mu_k} = \frac{A^k u_0}{\mu_k \mu_{k-1} \cdots \mu_1}$$

- $\|u_k\|_\infty = 1$, 即u_k至少有一个分量为1, 所以$\zeta_k = \mu_k \mu_{k-1} \cdots \mu_1$必为$A^k u_0$的一个模最大分量
- 从而ζ_k / λ_1^k就是$A^k u_0 / \lambda_1^k$的一个模最大分量
- 这样由前述极限等式我们知下述极限存在：

$$\zeta = \lim_{k\to\infty} \frac{\zeta_k}{\lambda_1^k}$$

- 这样我们就有

$$\lim_{k\to\infty} u_k = \lim_{k\to\infty} \frac{A^k u_0}{\zeta_k} = \lim_{k\to\infty} \frac{A^k u_0/\lambda_1^k}{\zeta_k/\lambda_1^k} = \frac{X_1 y_1}{\zeta} = x_1$$

- 这样我们就有

$$\lim_{k\to\infty} u_k = \lim_{k\to\infty} \frac{A^k u_0}{\zeta_k} = \lim_{k\to\infty} \frac{A^k u_0/\lambda_1^k}{\zeta_k/\lambda_1^k} = \frac{X_1 y_1}{\zeta} = x_1$$

- 由于$\tilde{y} = (y_1^T, 0, \ldots, 0)^T$是$\mathrm{diag}(J_1, \ldots, J_r)$相应于$\lambda_1$的特征向量，所以$X_1 y_1 = X\tilde{y}$是$A$相应于$\lambda_1$的特征向量，这就说明$x_1$是属于$\lambda_1$的一个特征向量

- 这样我们就有

$$\lim_{k\to\infty} u_k = \lim_{k\to\infty} \frac{A^k u_0}{\zeta_k} = \lim_{k\to\infty} \frac{A^k u_0/\lambda_1^k}{\zeta_k/\lambda_1^k} = \frac{X_1 y_1}{\zeta} = x_1$$

- 由于$\tilde{y} = (y_1^T, 0, \ldots, 0)^T$是$\mathrm{diag}(J_1, \ldots, J_r)$相应于$\lambda_1$的特征向量，所以$X_1 y_1 = X\tilde{y}$是$A$相应于$\lambda_1$的特征向量，这就说明$x_1$是属于$\lambda_1$的一个特征向量
- 由$Au_{k-1} = \mu_k u_k$, $x_1 \neq 0$可知$\mu_k \to \lambda_1$

注解

- 若定理条件不满足，则由幂法产生的序列的收敛性分析变得非常复杂

注解

- 若定理条件不满足，则由幂法产生的序列的收敛性分析变得非常复杂
- 这时$\{u_k\}$可能有若干个收敛于不同向量的子列

注解

- 若定理条件不满足，则由幂法产生的序列的收敛性分析变得非常复杂
- 这时$\{u_k\}$可能有若干个收敛于不同向量的子列
- 例：$A = XDX^{-1}$, $D = \mathrm{diag}(3, 2, 1, -3)$,

$$X = \begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & -1 & 0 \\ 1 & 2 & 1 & 1 \\ -1 & 0 & 0 & 1 \end{pmatrix}$$

收敛速度分析

- 幂法的收敛速度主要取决于$|\lambda_2|/|\lambda_1|$的大小

收敛速度分析

- 幂法的收敛速度主要取决于$|\lambda_2|/|\lambda_1|$的大小
- 这个数总是小于1的，它越小收敛也就越快

收敛速度分析

- 幂法的收敛速度主要取决于$|\lambda_2|/|\lambda_1|$的大小
- 这个数总是小于1的，它越小收敛也就越快
- 为了加快收敛，我们可以采用位移的方法，即对$A-\mu I$应用幂法，这里选取恰当的μ，使$A-\mu I$的模最大特征值与其它特征值之模的距离更大

收敛速度分析

- 幂法的收敛速度主要取决于$|\lambda_2|/|\lambda_1|$的大小
- 这个数总是小于1的，它越小收敛也就越快
- 为了加快收敛，我们可以采用位移的方法，即对$A-\mu I$应用幂法，这里选取恰当的μ，使$A-\mu I$的模最大特征值与其它特征值之模的距离更大
- 对前例可取$\mu=3$，从而收敛到-3对应的特征向量

求模第二大的特征值和对应特征向量

- 为了求模第二大的特征值和对应的特征向量，那么直接迭代是不行的

求模第二大的特征值和对应特征向量

- 为了求模第二大的特征值和对应的特征向量，那么直接迭代是不行的
- 需要对原矩阵进行降阶：即在知道了λ_1和x_1的前提下，把矩阵降低一阶，使它只包含A 的特征值$\lambda_2, \ldots, \lambda_n$，这一方法通常称为收缩技巧

求模第二大的特征值和对应特征向量

- 为了求模第二大的特征值和对应的特征向量，那么直接迭代是不行的
- 需要对原矩阵进行降阶：即在知道了λ_1和x_1的前提下，把矩阵降低一阶，使它只包含A 的特征值$\lambda_2, \ldots, \lambda_n$，这一方法通常称为收缩技巧
- 最简单实用的收缩技巧是利用正交变换

基于正交变换的收缩技巧

- 假设$Ax_1 = \lambda_1 x_1$

基于正交变换的收缩技巧

- 假设$Ax_1 = \lambda_1 x_1$
- 取酉矩阵P使得$Px_1 = \alpha e_1$ (可用Householder变换实现)

基于正交变换的收缩技巧

- 假设$Ax_1 = \lambda_1 x_1$
- 取酉矩阵P使得$Px_1 = \alpha e_1$ (可用Householder变换实现)
- 从而$PAx_1 = \lambda_1 Px_1 = \lambda_1 \alpha e_1$

基于正交变换的收缩技巧

- 假设$Ax_1 = \lambda_1 x_1$
- 取酉矩阵P使得$Px_1 = \alpha e_1$ (可用Householder变换实现)
- 从而$PAx_1 = \lambda_1 Px_1 = \lambda_1 \alpha e_1$
- 而$x_1 = \alpha P^* e_1$, 所以$PAP^* e_1 = \lambda_1 e_1$, 即

$$PAP^* = \begin{pmatrix} \lambda_1 & * \\ 0 & B_1 \end{pmatrix}$$

基于正交变换的收缩技巧

- 假设$Ax_1 = \lambda_1 x_1$
- 取酉矩阵P使得$Px_1 = \alpha e_1$ (可用Householder变换实现)
- 从而$PAx_1 = \lambda_1 Px_1 = \lambda_1 \alpha e_1$
- 而$x_1 = \alpha P^* e_1$, 所以$PAP^* e_1 = \lambda_1 e_1$, 即
$$PAP^* = \begin{pmatrix} \lambda_1 & * \\ 0 & B_1 \end{pmatrix}$$
- $n-1$阶矩阵B_1的特征值就是$\lambda_2, \ldots, \lambda_n$，对其应用幂法即可

注解

- 幂法的计算严重依赖于矩阵特征值的分布情况，因此实际用起来很不方便，特别是不适用于自动计算

注解

- 幂法的计算严重依赖于矩阵特征值的分布情况，因此实际用起来很不方便，特别是不适用于自动计算
- 当矩阵阶数非常高，无法利用其他更有效的算法时才用幂法计算少数几个模最大的特征值和相应的特征向量

注解

- 幂法的计算严重依赖于矩阵特征值的分布情况，因此实际用起来很不方便，特别是不适用于自动计算
- 当矩阵阶数非常高，无法利用其他更有效的算法时才用幂法计算少数几个模最大的特征值和相应的特征向量
- 基于幂法可以诱导出一些更有效的算法

反幂法

- 反幂法也称为反迭代法，它对A^{-1}应用幂法，从而求出A的模最小特征值和对应的特征向量

反幂法

- 反幂法也称为反迭代法，它对A^{-1}应用幂法，从而求出A的模最小特征值和对应的特征向量
- 其基本格式为

$$
\begin{aligned}
Ay_k &= z_{k-1},\\
\mu_k &= \zeta_i,\ \zeta_i\text{是}y_k\text{的模最大分量}\\
z_k &= y_k/\mu_k
\end{aligned}
$$

迭代结果

- 若A的特征值为$|\lambda_n| < |\lambda_{n-1}| \leqslant \cdots \leqslant |\lambda_1|$，则$\{z_k\}$收敛到$A$的对应于$\lambda_n$的一个特征向量，而$\{\mu_k\}$收敛于$\lambda_n^{-1}$

迭代结果

- 若A的特征值为$|\lambda_n| < |\lambda_{n-1}| \leqslant \cdots \leqslant |\lambda_1|$，则$\{z_k\}$收敛到$A$的对应于$\lambda_n$的一个特征向量，而$\{\mu_k\}$收敛于$\lambda_n^{-1}$
- 收敛速度由$|\lambda_n|/|\lambda_{n-1}|$的大小决定

应用场合

- 实际应用中，反幂法主要是用来求特征向量的，是在用某种方法求得A的某个特征值λ_i的近似值$\tilde{\lambda}_i$之后，应用反幂法于$A-\tilde{\lambda}_i I$上

应用场合

- 实际应用中，反幂法主要是用来求特征向量的，是在用某种方法求得A的某个特征值λ_i的近似值$\tilde{\lambda}_i$之后，应用反幂法于$A-\tilde{\lambda}_i I$上
- 即实际计算中常用的是带位移的反幂法，此时迭代格式为：

$$
\begin{aligned}
(A-\mu I)v_k &= z_{k-1},\\
z_k &= v_k/\|v\|_2
\end{aligned}
$$

格式分析

- 反幂法每迭代一次就需要解一个线性方程组，这要比幂法的运算量大得多

格式分析

- 反幂法每迭代一次就需要解一个线性方程组，这要比幂法的运算量大得多
- 由于方程组的系数矩阵不随k的变化而变化，所以可以事先对它进行列主元的LU分解，然后每次迭代就只需要解两个三角形方程组即可

格式分析

- 反幂法每迭代一次就需要解一个线性方程组，这要比幂法的运算量大得多
- 由于方程组的系数矩阵不随k的变化而变化，所以可以事先对它进行列主元的LU分解，然后每次迭代就只需要解两个三角形方程组即可
- 这里采用$\|\cdot\|_2$进行规范化，只是为了下面的分析方便。实际应用中完全可以采用$\|\cdot\|_\infty$进行规范化

收敛速度分析

- 迭代收敛速度取决于$|\lambda_1 - \mu|/|\lambda_2 - \mu|$的大小

收敛速度分析

- 迭代收敛速度取决于$|\lambda_1 - \mu|/|\lambda_2 - \mu|$的大小
- 从收敛速度的角度来看，μ取得越靠近A的某个特征值越好

收敛速度分析

- 迭代收敛速度取决于$|\lambda_1 - \mu|/|\lambda_2 - \mu|$的大小
- 从收敛速度的角度来看，μ取得越靠近A的某个特征值越好
- 但是当μ与A的某个特征值很靠近时，$A - \mu I$就与一个奇异矩阵很靠近，迭代时就需要求解一个非常病态的方程组

收敛速度分析

- 迭代收敛速度取决于$|\lambda_1-\mu|/|\lambda_2-\mu|$的大小
- 从收敛速度的角度来看，μ取得越靠近A的某个特征值越好
- 但是当μ与A的某个特征值很靠近时，$A-\mu I$就与一个奇异矩阵很靠近，迭代时就需要求解一个非常病态的方程组
- 实际计算的经验和理论分析的结果表明：$A-\mu I$的病态性并不影响其收敛速度，而且当μ与A的某个特征值很靠近时，常常只迭代一次就可以了

现象分析

- 设λ是A的一个单特征值，x是属于λ的单位特征向量

现象分析

- 设λ是A的一个单特征值，x是属于λ的单位特征向量
- 假定带位移的迭代格式中位移μ与λ十分靠近，且x是良态的，即$\mathrm{cond}(x)$不太大

现象分析

- 设λ是A的一个单特征值，x是属于λ的单位特征向量
- 假定带位移的迭代格式中位移μ与λ十分靠近，且x是良态的，即$\mathrm{cond}(x)$不太大
  - 设(x, U_2)是酉阵

现象分析

- 设λ是A的一个单特征值，x是属于λ的单位特征向量
- 假定带位移的迭代格式中位移μ与λ十分靠近，且x是良态的，即$\mathrm{cond}(x)$不太大
  - 设(x, U_2)是酉阵
  - 由条件数的定义知

$$\begin{aligned}\mathrm{cond}(x) &= \|U_2(\lambda I - A_2)^{-1}U_2^*\|_2 \\ &= \|(\lambda I - A_2)^{-1}U_2^*\|_2\end{aligned}$$

其中$A_2 = U_2^* A U_2$

- 假定给定z_0之后，我们用列主元Gauss消去法求解迭代格式中的线性方程组$(A-\mu I)v_1 = z_0$，得到计算解$\hat{v}_1$

- 假定给定z_0之后，我们用列主元Gauss消去法求解迭代格式中的线性方程组$(A-\mu I)v_1=z_0$，得到计算解$\hat{v}_1$
- 由Gauss消去法的误差分析知$\hat{v}_1$满足

$$(A-\mu I-E)\hat{v}_1=z_0$$

其中E与$A-\mu I$和z_0有关，但$\|E\|_2$有一致的上界，通常差不多就是机器精度

- 记$e=\hat{v}_1-v_1=(A-\mu I)^{-1}E\hat{v}_1$, 并将$e$分解为
$$e=x_1+x_2$$
其中$x_1\in\mathrm{span}\{x\}$, $x_2\in\mathrm{span}\{x\}^\perp$, 则存在$\alpha\in\mathbb{C}$和$y\in\mathbb{C}^{n-1}$使得
$$x_1=\alpha x,\quad x_2=U_2y$$

- 另一方面，我们有

$$A - \mu I = (x, U_2) \begin{pmatrix} \lambda - \mu & x^* A U_2 \\ 0 & A_2 - \mu I \end{pmatrix} \begin{pmatrix} x^* \\ U_2^* \end{pmatrix}$$

- 另一方面，我们有

$$A - \mu I = (x, U_2) \begin{pmatrix} \lambda - \mu & x^* A U_2 \\ 0 & A_2 - \mu I \end{pmatrix} \begin{pmatrix} x^* \\ U_2^* \end{pmatrix}$$

- 因而

$$\begin{aligned} & (A - \mu I)^{-1} \\ = & (x, U_2) \begin{pmatrix} \dfrac{1}{\lambda - \mu} & \dfrac{-1}{\lambda - \mu} x^* A U_2 (A_2 - \mu I)^{-1} \\ 0 & (A_2 - \mu I)^{-1} \end{pmatrix} \begin{pmatrix} x^* \\ U_2^* \end{pmatrix} \end{aligned}$$

- 注意到$x^*xx^* = x^*$, $x^*U_2 = 0$

- 注意到$x^*xx^* = x^*$, $x^*U_2 = 0$
- 这样我们就有

$$\begin{aligned}(A-\mu I)^{-1} &= \frac{xx^*}{\lambda-\mu}(I - AU_2(A_2-\mu I)^{-1}U_2^*) \\ &\quad + U_2(A_2-\mu I)^{-1}U_2^*\end{aligned}$$

$$\begin{aligned}\alpha = x^*e &= x^*(A-\mu I)^{-1}E\hat{v}_1 \\ &= \frac{x^*}{\lambda-\mu}(I - AU_2(A_2-\mu I)^{-1}U_2^*)E\hat{v}_1 \\ y = U_2^*e &= U_2^*(A-\mu I)^{-1}E\hat{v}_1 \\ &= (A_2-\mu I)^{-1}U_2^*E\hat{v}_1\end{aligned}$$

- 注意到

$$(A_2-\mu I)^{-1} = (I+(\lambda-\mu)(A_2-\lambda I)^{-1})^{-1}(A_2-\lambda I)^{-1}$$

- 注意到

$$(A_2-\mu I)^{-1}=(I+(\lambda-\mu)(A_2-\lambda I)^{-1})^{-1}(A_2-\lambda I)^{-1}$$

- 所以当λ与μ很近时

$$s=\|(A_2-\mu I)^{-1}U_2^*\|_2\approx\|(A_2-\lambda I)^{-1}U_2^*\|_2=\mathrm{cond}(x$$

- 注意到

$$(A_2-\mu I)^{-1}=(I+(\lambda-\mu)(A_2-\lambda I)^{-1})^{-1}(A_2-\lambda I)^{-1}$$

- 所以当λ与μ很近时

$$s=\|(A_2-\mu I)^{-1}U_2^*\|_2\approx\|(A_2-\lambda I)^{-1}U_2^*\|_2=\mathrm{cond}(x$$

- 因此当x良态时，s也不会太大，于是

$$\|x_2\|_2=\|y\|_2\leqslant s\|E\hat{v}_1\|_2$$

就是一个不太大的量

- 另一方面，

$$|\alpha| \leqslant \frac{1}{|\lambda| - |\mu|}(1 + \|A\|_2 s)\|E\hat{v}_1\|_2$$

却是一个很大的量

- 另一方面，
$$|\alpha| \leqslant \frac{1}{|\lambda| - |\mu|}(1 + \|A\|_2 s)\|E\hat{v}_1\|_2$$
却是一个很大的量
- 这就是说求解方程组所引起的误差主要对其解在特征子空间span$\{x\}$上投影的长度有影响

- 另一方面，
$$|\alpha| \leqslant \frac{1}{|\lambda| - |\mu|}(1 + \|A\|_2 s)\|E\hat{v}_1\|_2$$
却是一个很大的量
- 这就是说求解方程组所引起的误差主要对其解在特征子空间span$\{x\}$上投影的长度有影响
- 而这对于我们要计算λ的近似特征向量是十分有利的

- 另一方面，
$$|\alpha| \leqslant \frac{1}{|\lambda| - |\mu|}(1 + \|A\|_2 s)\|E\hat{v}_1\|_2$$
却是一个很大的量
- 这就是说求解方程组所引起的误差主要对其解在特征子空间span$\{x\}$上投影的长度有影响
- 而这对于我们要计算λ的近似特征向量是十分有利的
- 因为我们关心的主要是所得向量的方向而不是它的长度

- 上面的分析实质上也表明，当μ靠近λ且x良态时，只需应用一次反幂法就可得到λ的较好近似特征向量

- 上面的分析实质上也表明，当μ靠近λ且x良态时，只需应用一次反幂法就可得到λ的较好近似特征向量
- 为此我们引进一些概念

达到机器精度

- 设机器精度为**u**

达到机器精度

- 设机器精度为$\mathbf{u}$
- 对于给定的$\mu \in \mathbb{C}$, 如果存在$E \in \mathbb{C}^{n\times n}$使得

$$\det(A+E-\mu I)=0, \|E\|_2=O(\mathbf{u})$$

我们称μ是A的一个达到机器精度的近似特征值

达到机器精度

- 设机器精度为$\mathbf{u}$
- 对于给定的$\mu \in \mathbb{C}$, 如果存在$E \in \mathbb{C}^{n\times n}$使得

$$\det(A+E-\mu I)=0, \|E\|_2=O(\mathbf{u})$$

  我们称μ是A的一个达到机器精度的近似特征值
- 对于一个给定的$x \in \mathbb{C}^n$, 如果存在$F \in \mathbb{C}^{n\times n}$满足$\|F\|_2=O(\mathbf{u})$, 使得$x$是$A+F$的特征向量，我们称$x$是$A$的一个达到机器精度的近似特征向量

两者的关系

- 若μ是A的一个达到机器精度的近似特征值，则存在$E \in \mathbb{C}^{n\times n}$ 满足$\|E\|_2 = O(\mathbf{u})$使得$(A + E - \mu I)y = 0$有非零解

两者的关系

- 若μ是A的一个达到机器精度的近似特征值，则存在$E \in \mathbb{C}^{n\times n}$满足$\|E\|_2 = O(\mathbf{u})$使得$(A+E-\mu I)y=0$有非零解
- 设$y \in \mathbb{C}^n$满足$(A+E-\mu I)y=0$, $\|y\|_2=1$, $\|E\|_2 = O(\mathbf{u})$, 则我们有

$$(A+E)y = \mu y$$

即y是A的一个达到机器精度的近似特征向量

- 若我们在迭代格式中取$z_0 = (A - \mu I)y$, 那么在精确计算的前提下，只需迭代一次就可得到A的达到机器精度的特征向量

- 若我们在迭代格式中取$z_0 = (A - \mu I)y$, 那么在精确计算的前提下，只需迭代一次就可得到A的达到机器精度的特征向量
- 在实际计算时我们无法按这种方式取初始向量，但这说明了在恰当选取初始向量之后，反幂法具有“一次迭代”性

λ病态时的情形

- 还需要指出的是，当λ比较病态时，利用反幂法再进行迭代，一般不会得到更好的近似特征向量

λ病态时的情形

- 还需要指出的是，当λ比较病态时，利用反幂法再进行迭代，一般不会得到更好的近似特征向量
- 例
$$A = \begin{pmatrix} 1 & 1 \\ 10^{-10} & 1 \end{pmatrix}$$
其特征值$\lambda_1 = 0.99999$, $\lambda_2 = 1.00001$, 对应的特征向量$x_1 = (1, -10^{-5})^T$, $x_2 = (1, 10^{-5})^T$.
取$\mu = 1$, $z_0 = (0, 1)^T$,两次迭代结果更差

初始向量的选取方法

- 利用随机数发生子程序随机地选取向量，规范化后作为初始向量

初始向量的选取方法

- 利用随机数发生子程序随机地选取向量，规范化后作为初始向量
- 半次迭代法：首先对$A-\mu I$进行LU分解，得到$A-\mu I=LU$, 第一次迭代为$LUv_1=z_0$. 选$z_0=Le$，其中$e=(1,\ldots,1)^T$, 则为了求v_1只需求解$Uv_1=e$

QR方法简介

- QR方法是自计算机问世以来矩阵计算的重大进展之一，也是目前计算一般矩阵的全部特征值和特征向量的最有效方法之一

QR方法简介

- QR方法是自计算机问世以来矩阵计算的重大进展之一，也是目前计算一般矩阵的全部特征值和特征向量的最有效方法之一
- QR方法是一种迭代方法，它利用正交相似变换把矩阵逐步约化为上三角阵或者拟上三角阵

QR方法简介

- QR方法是自计算机问世以来矩阵计算的重大进展之一，也是目前计算一般矩阵的全部特征值和特征向量的最有效方法之一
- QR方法是一种迭代方法，它利用正交相似变换把矩阵逐步约化为上三角阵或者拟上三角阵
- 基本收敛速度是二次的。当原矩阵是实对称时，可达到三次收敛

The top 10 algorithms of 1900s

- Metropolis Algorithm for Monte Carlo

The top 10 algorithms of 1900s

- Metropolis Algorithm for Monte Carlo
- Simplex Method for Linear Programming

The top 10 algorithms of 1900s

- Metropolis Algorithm for Monte Carlo
- Simplex Method for Linear Programming
- Krylov Subspace Iteration Methods

The top 10 algorithms of 1900s

- Metropolis Algorithm for Monte Carlo
- Simplex Method for Linear Programming
- Krylov Subspace Iteration Methods
- The Decompositional Approach to Matrix Computations

The top 10 algorithms of 1900s

- Metropolis Algorithm for Monte Carlo
- Simplex Method for Linear Programming
- Krylov Subspace Iteration Methods
- The Decompositional Approach to Matrix Computations
- The Fortran Optimizing Compiler

The top 10 algorithms of 1900s

- Metropolis Algorithm for Monte Carlo
- Simplex Method for Linear Programming
- Krylov Subspace Iteration Methods
- The Decompositional Approach to Matrix Computations
- The Fortran Optimizing Compiler
- QR Algorithm for Computing Eigenvalues

The top 10 algorithms of 1900s

- Metropolis Algorithm for Monte Carlo
- Simplex Method for Linear Programming
- Krylov Subspace Iteration Methods
- The Decompositional Approach to Matrix Computations
- The Fortran Optimizing Compiler
- QR Algorithm for Computing Eigenvalues
- Quicksort Algorithm for Sorting

The top 10 algorithms of 1900s

- Metropolis Algorithm for Monte Carlo
- Simplex Method for Linear Programming
- Krylov Subspace Iteration Methods
- The Decompositional Approach to Matrix Computations
- The Fortran Optimizing Compiler
- QR Algorithm for Computing Eigenvalues
- Quicksort Algorithm for Sorting
- Fast Fourier Transform

The top 10 algorithms of 1900s

- Metropolis Algorithm for Monte Carlo
- Simplex Method for Linear Programming
- Krylov Subspace Iteration Methods
- The Decompositional Approach to Matrix Computations
- The Fortran Optimizing Compiler
- QR Algorithm for Computing Eigenvalues
- Quicksort Algorithm for Sorting
- Fast Fourier Transform
- Integer Relation Detection

The top 10 algorithms of 1900s

- Metropolis Algorithm for Monte Carlo
- Simplex Method for Linear Programming
- Krylov Subspace Iteration Methods
- The Decompositional Approach to Matrix Computations
- The Fortran Optimizing Compiler
- QR Algorithm for Computing Eigenvalues
- Quicksort Algorithm for Sorting
- Fast Fourier Transform
- Integer Relation Detection
- Fast Multipole Method

历史

- J.G.F. Francis与V.N. Kublanovskaya在上世纪50年代独立发展出QR方法：基于QR分解

历史

- J.G.F. Francis与V.N. Kublanovskaya在上世纪50年代独立发展出QR方法：基于QR分解
- 经过几十年的发展，QR方法的现代版本称为隐式QR算法

历史

- J.G.F. Francis与V.N. Kublanovskaya在上世纪50年代独立发展出QR方法：基于QR分解
- 经过几十年的发展，QR方法的现代版本称为隐式QR算法
- 但在隐式QR算法中没有显式地进行QR分解，因此有人建议称之为Francis算法

历史

- J.G.F. Francis与V.N. Kublanovskaya在上世纪50年代独立发展出QR方法：基于QR分解
- 经过几十年的发展，QR方法的现代版本称为隐式QR算法
- 但在隐式QR算法中没有显式地进行QR分解，因此有人建议称之为Francis算法
- 在QR算法之前，曾出现过LR算法，它是基于LU分解的。LR算法是由H. Rutishauser在1950年代发展的

QR方法的基本迭代

- 对给定的$A_0 = A \in \mathbb{C}^{n\times n}$，QR方法的基本迭代格式如下：

$$A_{m-1} = Q_m R_m, \quad (A_{m-1}\text{的QR分解})$$
$$A_m = R_m Q_m$$

这里Q_m为酉阵，R_m为上三角阵

QR方法的基本迭代

- 对给定的$A_0 = A \in \mathbb{C}^{n\times n}$，QR方法的基本迭代格式如下：

$$A_{m-1} = Q_m R_m, \quad (A_{m-1}\text{的QR分解})$$
$$A_m = R_m Q_m$$

这里Q_m为酉阵，R_m为上三角阵

- 课本例6.4.1和其它两个例子的程序演示

QR方法的基本迭代

- 对给定的$A_0 = A \in \mathbb{C}^{n\times n}$，QR方法的基本迭代格式如下：

$$
\begin{aligned}
A_{m-1} &= Q_m R_m, \quad (A_{m-1}\text{的QR分解}) \\
A_m &= R_m Q_m
\end{aligned}
$$

  这里Q_m为酉阵，R_m为上三角阵

- 课本例6.4.1和其它两个例子的程序演示
- 为了下面理论分析方便，我们暂且要求R_m的对角元都是非负的

迭代过程分析

- 根据迭代格式，我们有$A_m = Q_m^* A_{m-1} Q_m$, 所以矩阵序列$\{A_m\}$中每一个矩阵都与A相似

迭代过程分析

- 根据迭代格式，我们有$A_m = Q_m^* A_{m-1} Q_m$，所以矩阵序列$\{A_m\}$中每一个矩阵都与A相似
- 从而有$A_m = \tilde{Q}_m^* A \tilde{Q}_m$，其中$\tilde{Q}_m = Q_1 Q_2 \cdots Q_m$

迭代过程分析

- 根据迭代格式，我们有$A_m = Q_m^* A_{m-1} Q_m$，所以矩阵序列$\{A_m\}$中每一个矩阵都与A相似
- 从而有$A_m = \tilde{Q}_m^* A \tilde{Q}_m$，其中$\tilde{Q}_m = Q_1 Q_2 \cdots Q_m$
- 即$A\tilde{Q}_m = \tilde{Q}_m A_m = \tilde{Q}_m Q_{m+1} R_{m+1}$

迭代过程分析

- 根据迭代格式，我们有$A_m = Q_m^* A_{m-1} Q_m$，所以矩阵序列$\{A_m\}$中每一个矩阵都与A相似
- 从而有$A_m = \tilde{Q}_m^* A \tilde{Q}_m$，其中$\tilde{Q}_m = Q_1 Q_2 \cdots Q_m$
- 即$A\tilde{Q}_m = \tilde{Q}_m A_m = \tilde{Q}_m Q_{m+1} R_{m+1}$
- 定义$\tilde{R}_k = R_k R_{k-1} \cdots R_1$，我们有

$$\tilde{Q}_{m+1} \tilde{R}_{m+1} = A \tilde{Q}_m \tilde{R}_m$$

迭代过程分析

- 根据迭代格式，我们有$A_m = Q_m^* A_{m-1} Q_m$，所以矩阵序列$\{A_m\}$中每一个矩阵都与A相似
- 从而有$A_m = \tilde{Q}_m^* A \tilde{Q}_m$，其中$\tilde{Q}_m = Q_1 Q_2 \cdots Q_m$
- 即$A\tilde{Q}_m = \tilde{Q}_m A_m = \tilde{Q}_m Q_{m+1} R_{m+1}$
- 定义$\tilde{R}_k = R_k R_{k-1} \cdots R_1$，我们有

$$\tilde{Q}_{m+1} \tilde{R}_{m+1} = A \tilde{Q}_m \tilde{R}_m$$

- 由此可归纳证明$A^m = \tilde{Q}_m \tilde{R}_m$，这是$A^m$的QR分解

QR方法与幂法的关系

- 记$\tilde{R}_m$的元素为γ_{ij}，$\tilde{Q}_m$的第一列为$q_1^{(m)}$，则由$A^m = \tilde{Q}_m \tilde{R}_m$可得

$$A^m e_1 = \gamma_{11} q_1^{(m)}$$

QR方法与幂法的关系

- 记$\tilde{R}_m$的元素为γ_{ij}，$\tilde{Q}_m$的第一列为$q_1^{(m)}$，则由$A^m = \tilde{Q}_m \tilde{R}_m$可得

$$A^m e_1 = \gamma_{11} q_1^{(m)}$$

- 所以$q_1^{(m)}$可以看作是对A用e_1作初始向量的幂法所得到的向量

QR方法与幂法的关系

- 记$\tilde{R}_m$的元素为γ_{ij}, $\tilde{Q}_m$的第一列为$q_1^{(m)}$, 则由$A^m = \tilde{Q}_m \tilde{R}_m$可得

$$A^m e_1 = \gamma_{11} q_1^{(m)}$$

- 所以$q_1^{(m)}$可以看作是对A用e_1作初始向量的幂法所得到的向量
- 若A的模最大特征值λ_1与其他特征值分离，那么$q_1^{(m)}$将收敛到A的一个属于λ_1的特征向量

A_m下三角元素趋于0

定理

设$A \in \mathbb{C}^{n\times n}$的$n$个特征值满足

$$|\lambda_1| > |\lambda_2| > \cdots > |\lambda_n| > 0$$

并设n阶方阵Y的第i行是A对应于λ_i的左特征向量。如果Y有LU分解，则由QR方法产生的矩阵A_m的对角线以下的元素趋向于0，同时第i个对角元趋向于λ_i，$i = 1, 2, \ldots, n$

定理证明框架

- A可对角化，从而A^m也可对角化

定理证明框架

- A可对角化，从而A^m也可对角化
- 基于条件，构造A^m的QR分解，其中各元素在$m \to \infty$时的极限形状已知

定理证明框架

- A可对角化，从而A^m也可对角化
- 基于条件，构造A^m的QR分解，其中各元素在$m \to \infty$时的极限形状已知
- 利用$A^m = \tilde{Q}_m \tilde{R}_m$和QR分解的唯一性得到$\tilde{Q}_m$和$\tilde{R}_m$，其在$m \to \infty$时的极限形状已知

定理证明框架

- A可对角化，从而A^m也可对角化
- 基于条件，构造A^m的QR分解，其中各元素在$m \to \infty$时的极限形状已知
- 利用$A^m = \tilde{Q}_m \tilde{R}_m$和QR分解的唯一性得到$\tilde{Q}_m$和$\tilde{R}_m$，其在$m \to \infty$时的极限形状已知
- 利用$A_m = \tilde{Q}_m^* A \tilde{Q}_m$可证$A_m$下三角元素趋向于零

定理证明

- 令$X = Y^{-1}$, $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$, 则有$A = X\Lambda Y$

定理证明

- 令$X = Y^{-1}$, $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$, 则有$A = X\Lambda Y$
- 设Y的LU分解为$Y = LU$, 其中L是单位下三角阵，U是上三角阵

定理证明

- 令$X=Y^{-1}$, $\Lambda=\mathrm{diag}(\lambda_1,\dots,\lambda_n)$, 则有$A=X\Lambda Y$
- 设Y的LU分解为$Y=LU$, 其中L是单位下三角阵，U是上三角阵
- 从而我们有

$$\begin{aligned}A^m = X\Lambda^m Y = X\Lambda^m LU &= X(\Lambda^m L\Lambda^{-m})\Lambda^m U\\ &= X(I+E_m)\Lambda^m U\end{aligned}$$

构造A^m的QR分解

- 由于L是单位下三角阵，而$|\lambda_i| < |\lambda_j|(i > j)$, 所以有$\lim_{m\to\infty} E_m = 0$

构造A^m的QR分解

- 由于L是单位下三角阵，而$|\lambda_i| < |\lambda_j|(\ i > j)$, 所以有$\lim\limits_{m\to\infty} E_m = 0$
- 令X的QR分解为$X = QR$. 由于X非奇异，所以可要求R的对角元全是正数。这样我们有

$$A^m = QR(I + E_m)\Lambda^m U = Q(I + RE_mR^{-1})R\Lambda^m U$$

构造A^m的QR分解

- 由于L是单位下三角阵，而$|\lambda_i| < |\lambda_j|(\ i > j)$, 所以有$\lim_{m\to\infty} E_m = 0$
- 令X的QR分解为$X = QR$. 由于X非奇异，所以可要求R的对角元全是正数。这样我们有

$$A^m = QR(I + E_m)\Lambda^m U = Q(I + RE_mR^{-1})R\Lambda^m U$$

- 当m充分大时，$I + RE_mR^{-1}$非奇异，故可取它的QR分解$I + RE_mR^{-1} = \hat{Q}_m\hat{R}_m$, 其中$\hat{R}_m$的对角元均正数

- 由$E_m \to 0\ (m \to \infty)$可知

$$\lim_{m\to\infty} \hat{Q}_m = \lim_{m\to\infty} \hat{R}_m = I$$

- 由$E_m \to 0$ $(m \to \infty)$可知

$$\lim_{m\to\infty} \hat{Q}_m = \lim_{m\to\infty} \hat{R}_m = I$$

- 至此我们有$A^m = (Q\hat{Q}_m)(\hat{R}_m R \Lambda^m U)$

- 由$E_m \to 0\ (m \to \infty)$可知
$$\lim_{m\to\infty} \hat{Q}_m = \lim_{m\to\infty} \hat{R}_m = I$$
- 至此我们有$A^m = (Q\hat{Q}_m)(\hat{R}_m R \Lambda^m U)$
  - 这是A^m的一个QR分解，只是对角元可能不是正数

- 由$E_m \to 0\ (m \to \infty)$可知

$$\lim_{m\to\infty} \hat{Q}_m = \lim_{m\to\infty} \hat{R}_m = I$$

- 至此我们有$A^m = (Q\hat{Q}_m)(\hat{R}_m R \Lambda^m U)$
  - 这是A^m的一个QR分解，只是对角元可能不是正数
- 为校正这一点，设u_{ii}是U的对角元，定义

$$D_1 = \mathrm{diag}\left(\frac{\lambda_1}{|\lambda_1|}, \dots, \frac{\lambda_n}{|\lambda_n|}\right)$$

$$D_2 = \mathrm{diag}\left(\frac{u_{11}}{|u_{11}|}, \dots, \frac{u_{nn}}{|u_{nn}|}\right)$$

- 这样我们就有

$$A^m = (Q\hat{Q}_m D_1^m D_2)(D_2^{-1} D_1^{-m} \hat{R}_m R \Lambda^m U)$$

- 这样我们就有

$$A^m = (Q\hat{Q}_m D_1^m D_2)(D_2^{-1} D_1^{-m} \hat{R}_m R \Lambda^m U)$$

- 由于$A^m = \tilde{Q}_m \tilde{R}_m$, 所以根据QR分解的唯一性，我们有

$$\tilde{Q}_m = Q\hat{Q}_m D_1^m D_2, \tilde{R}_m = D_2^{-1} D_1^{-m} \hat{R}_m R \Lambda^m U$$

- 这样我们就有

$$A^m = (Q\hat{Q}_m D_1^m D_2)(D_2^{-1} D_1^{-m} \hat{R}_m R \Lambda^m U)$$

- 由于$A^m = \tilde{Q}_m \tilde{R}_m$，所以根据QR分解的唯一性，我们有

$$\tilde{Q}_m = Q\hat{Q}_m D_1^m D_2, \tilde{R}_m = D_2^{-1} D_1^{-m} \hat{R}_m R \Lambda^m U$$

- 所以有

$$A_m = \left(D_2^* (D_1^*)^m \hat{Q}_m^* Q^*\right) A \left(Q\hat{Q}_m D_1^m D_2\right)$$

- 而注意到$A = X\Lambda Y = X\Lambda X^{-1} = QR\Lambda R^{-1}Q^*$

- 而注意到$A = X\Lambda Y = X\Lambda X^{-1} = QR\Lambda R^{-1}Q^*$
- 从而有

$$A_m = D_2^*(D_1^*)^m \hat{Q}_m^* R\Lambda R^{-1}\hat{Q}_m D_1^m D_2$$

- 而注意到$A = X\Lambda Y = X\Lambda X^{-1} = QR\Lambda R^{-1}Q^*$
- 从而有

$$A_m = D_2^*(D_1^*)^m \hat{Q}_m^* R\Lambda R^{-1} \hat{Q}_m D_1^m D_2$$

- 这里除$\hat{Q}_m$外，其它矩阵都是上三角阵，而$\hat{Q}_m \to I$ $(m \to \infty)$, 所以A_m的下三角元趋向于零

- 而注意到$A = X\Lambda Y = X\Lambda X^{-1} = QR\Lambda R^{-1}Q^*$
- 从而有

$$A_m = D_2^*(D_1^*)^m \hat{Q}_m^* R\Lambda R^{-1}\hat{Q}_m D_1^m D_2$$

- 这里除$\hat{Q}_m$外，其它矩阵都是上三角阵，而$\hat{Q}_m \to I$ $(m \to \infty)$, 所以A_m的下三角元趋向于零
- 从而对角元趋向于A的第i个特征值

只用到实数运算的QR方法

- 实际应用中所遇到的大量特征值问题都是关于实矩阵的，因此我们希望设计出只涉及到实数运算的QR迭代

只用到实数运算的QR方法

- 实际应用中所遇到的大量特征值问题都是关于实矩阵的，因此我们希望设计出只涉及到实数运算的QR迭代
- 如此的迭代格式为

$$A_{m-1} = Q_m R_m$$

$$A_m = R_m Q_m$$

其中Q_m是正交矩阵，R_m是上三角阵

只用到实数运算的QR方法

- 实际应用中所遇到的大量特征值问题都是关于实矩阵的，因此我们希望设计出只涉及到实数运算的QR迭代
- 如此的迭代格式为

$$A_{m-1} = Q_m R_m$$
$$A_m = R_m Q_m$$

其中Q_m是正交矩阵，R_m是上三角阵

- 但由于复共轭特征值的存在，因此我们不能期望迭代格式产生的A_m仍趋向于一个上三角阵

实Schur分解

定理

设$A \in \mathbb{R}^{n\times n}$，则存在正交矩阵$Q \in \mathbb{R}^{n\times n}$使得

$$Q^TAQ = \begin{pmatrix} R_{11} & R_{12} & \cdots & R_{1m} \\ & R_{22} & \cdots & R_{2m} \\ & & \ddots & \vdots \\ & & & R_{mm} \end{pmatrix}$$

其中R_{ii}或者是一个实数，或者是一个具有一对复共轭特征值的2阶方阵

- 实Schur分解定理中右边的拟上三角阵称为A的实Schur标准形

- 实Schur分解定理中右边的拟上三角阵称为A的实Schur标准形
- 类似可证明实QR迭代格式产生的A_k应逼近于A的实Schur标准形

- 实Schur分解定理中右边的拟上三角阵称为A的实Schur标准形
- 类似可证明实QR迭代格式产生的A_k应逼近于A的实Schur标准形
- 但目前这个版本的QR方法在实用性方面没有竞争力，因为:

- 实Schur分解定理中右边的拟上三角阵称为A的实Schur标准形
- 类似可证明实QR迭代格式产生的A_k应逼近于A的实Schur标准形
- 但目前这个版本的QR方法在实用性方面没有竞争力，因为：
    - 每次迭代的运算量太大

- 实Schur分解定理中右边的拟上三角阵称为A的实Schur标准形
- 类似可证明实QR迭代格式产生的A_k应逼近于A的实Schur标准形
- 但目前这个版本的QR方法在实用性方面没有竞争力，因为：
  - 每次迭代的运算量太大
  - 收敛速度太慢

把QR方法改造成实用算法

- 把A相似到一种特殊矩阵，其QR分解计算简单，而且对QR迭代仍保持矩阵的特殊性不变

把QR方法改造成实用算法

- 把A相似到一种特殊矩阵，其QR分解计算简单，而且对QR迭代仍保持矩阵的特殊性不变
  - 上Hessenberg矩阵

把QR方法改造成实用算法

- 把A相似到一种特殊矩阵，其QR分解计算简单，而且对QR迭代仍保持矩阵的特殊性不变
    - 上Hessenberg矩阵
- 引用原点位移方法

把QR方法改造成实用算法

- 把A相似到一种特殊矩阵，其QR分解计算简单，而且对QR迭代仍保持矩阵的特殊性不变
  - 上Hessenberg矩阵
- 引用原点位移方法
- 只采用实数运算，处理共轭复特征值情况

把QR方法改造成实用算法

- 把A相似到一种特殊矩阵，其QR分解计算简单，而且对QR迭代仍保持矩阵的特殊性不变
  - 上Hessenberg矩阵
- 引用原点位移方法
- 只采用实数运算，处理共轭复特征值情况
  - 双重步位移

把QR方法改造成实用算法

- 把A相似到一种特殊矩阵，其QR分解计算简单，而且对QR迭代仍保持矩阵的特殊性不变
  - 上Hessenberg矩阵
- 引用原点位移方法
- 只采用实数运算，处理共轭复特征值情况
  - 双重步位移
- 实用算法：隐式QR算法

把QR方法改造成实用算法

- 把A相似到一种特殊矩阵，其QR分解计算简单，而且对QR迭代仍保持矩阵的特殊性不变
  - 上Hessenberg矩阵
- 引用原点位移方法
- 只采用实数运算，处理共轭复特征值情况
  - 双重步位移
- 实用算法：隐式QR算法
  - 平均两次QR迭代就可以分离出一个特征值或2×2子矩阵，计算量为$O(n^3)$

相似约化到准上三角阵

- 为了减少每次迭代的运算量，我们先把原矩阵A经相似变换约化为一个准上三角阵

相似约化到准上三角阵

- 为了减少每次迭代的运算量，我们先把原矩阵A经相似变换约化为一个准上三角阵
- 为此，我们先看一下基于Householder变换的相似变换可以得到什么

相似约化到准上三角阵

- 为了减少每次迭代的运算量，我们先把原矩阵A经相似变换约化为一个准上三角阵
- 为此，我们先看一下基于Householder变换的相似变换可以得到什么
- 第一步我们自然选取Householder变换H_1，使得H_1A的第一列有尽可能多的零元素

相似约化到准上三角阵

- 为了减少每次迭代的运算量，我们先把原矩阵A经相似变换约化为一个准上三角阵
- 为此，我们先看一下基于Householder变换的相似变换可以得到什么
- 第一步我们自然选取Householder变换H_1，使得H_1A的第一列有尽可能多的零元素
- 最多可以得到$n-1$个零。这能做到么？

相似变换的妥协

- 在对$A=(a_{ij})$左乘H_1进行行变换之后，还需要对A右乘H_1进行列变换

相似变换的妥协

- 在对$A=(a_{ij})$左乘H_1进行行变换之后，还需要对A右乘H_1进行列变换
- 为了保证已在H_1A中的第一列所出现的零元素不至于因右乘H_1被破坏，我们选取H_1具有如下形式：

$$H_1 = \begin{pmatrix} 1 & 0 \\ 0 & \tilde{H}_1 \end{pmatrix}$$

$\tilde{H}_1$的构造

- 如此我们有

$$H_1AH_1 = \begin{pmatrix} a_{11} & a_2^T\tilde{H}_1 \\ \tilde{H}_1a_1 & \tilde{H}_1A_{22}\tilde{H}_1 \end{pmatrix}$$

其中$a_1^T = (a_{21}, a_{31}, \ldots, a_{n1})$,
$a_2^T = (a_{12}, a_{13}, \ldots, a_{1n})$, A_{22}是A右下角的$n-1$阶主子阵

$\tilde{H}_1$的构造

- 如此我们有

$$H_1AH_1 = \begin{pmatrix} a_{11} & a_2^T\tilde{H}_1 \\ \tilde{H}_1a_1 & \tilde{H}_1A_{22}\tilde{H}_1 \end{pmatrix}$$

其中$a_1^T = (a_{21}, a_{31}, \ldots, a_{n1})$, $a_2^T = (a_{12}, a_{13}, \ldots, a_{1n})$, A_{22}是A右下角的$n-1$阶主子阵

- 所以Householder变换$\tilde{H}_1$的最佳选择应该使得$\tilde{H}_1a_1 = pe_1$

- 如此H_1AH_1的第一列中除开始两个元素外，余下的$n-2$个元素为零

- 如此H_1AH_1的第一列中除开始两个元素外，余下的$n-2$个元素为零
- 类似对后面各列进行处理，我们可以找到$n-2$个Householder变换$H_1, \ldots, H_{n-2}$使得

$$H_{n-2} \cdots H_1 A H_1 \cdots H_{n-2} = H$$

其中$H = (h_{ij})$满足$h_{ij} = 0$, $i > j + 1$, 这样的矩阵称为上Hessenberg矩阵

上Hessenberg分解

- 现在令

$$Q_0 = H_1 H_2 \cdots H_{n-2}$$

则有

$$Q_0^T A Q_0 = H$$

上Hessenberg分解

- 现在令
$$Q_0 = H_1 H_2 \cdots H_{n-2}$$
则有
$$Q_0^T A Q_0 = H$$
- 一般称如此分解式为A的**上Hessenberg分解**

上Hessenberg分解

- 现在令
$$Q_0 = H_1 H_2 \cdots H_{n-2}$$
则有
$$Q_0^T A Q_0 = H$$
- 一般称如此分解式为A的上Hessenberg分解
- 算法运算量为$10n^3/3$; 如果要记录Q, 还需要增加运算量$4n^3/3$

上Hessenberg分解

- 现在令
$$Q_0 = H_1 H_2 \cdots H_{n-2}$$
则有
$$Q_0^T A Q_0 = H$$
- 一般称如此分解式为A的上Hessenberg分解
- 算法运算量为$10n^3/3$; 如果要记录Q, 还需要增加运算量$4n^3/3$
- 算法6.4.1的程序演示

误差分析

- 可以证明：如此算法得到的上Hessenberg矩阵$\hat{H}$满足

$$\hat{H} = Q^T(A+E)Q$$

其中Q是正交矩阵，$\|E\|_F \leqslant cn^2\|A\|_F \mathbf{u}$

误差分析

- 可以证明：如此算法得到的上Hessenberg矩阵$\hat{H}$满足
$$\hat{H} = Q^T(A+E)Q$$
其中Q是正交矩阵，$\|E\|_F \leqslant cn^2\|A\|_F\mathbf{u}$
- 我们也可以采用Givens变换(运算量增加)或者列主元的Gauss消去法将A约化为上Hessenberg矩阵（运算量少，但稳定性较差）

唯一性定理

定理

设A有如下两个上Hessenberg分解：

$$U^TAU = H, \quad V^TAV = G$$

其中$U = (u_1, \dots, u_n)$和$V = (v_1, \dots, v_n)$是n阶正交矩阵，$H = (h_{ij})$和$G = (g_{ij})$是上Hessenberg矩阵。若$u_1 = v_1$，而且H的次对角元$h_{i+1,i}$均不为零，则存在对角元均为1或-1的对角阵D，使得$U = VD$，$H = DGD$

定理证明

采用归纳法证明

- 假定对某个m已证$u_j = \varepsilon_j v_j$, $j = 1, \ldots, m$, 其中$\varepsilon_1 = 1$, $\varepsilon_j = 1$或-1, $j = 2, \ldots, m$. 下面证明存在$\varepsilon_{m+1} = 1$或-1使得$u_{m+1} = \varepsilon_{m+1} v_{m+1}$

定理证明

采用归纳法证明

- 假定对某个m已证$u_j = \varepsilon_j v_j$, $j = 1, \ldots, m$, 其中$\varepsilon_1 = 1$, $\varepsilon_j = 1$或-1, $j = 2, \ldots, m$. 下面证明存在$\varepsilon_{m+1} = 1$或-1使得$u_{m+1} = \varepsilon_{m+1} v_{m+1}$
- 根据上Hessenberg分解式，我们有

$$AU = UH, \quad AV = VG$$

U, V的关系反应到H, G上

- 比例两个矩阵等式的第m列，我们有

$$Au_m = h_{1m}u_1 + \cdots + h_{mm}u_m + h_{m+1,m}u_{m+1}$$

$$Av_m = g_{1m}v_1 + \cdots + g_{mm}v_m + g_{m+1,m}v_{m+1}$$

U, V的关系反应到H, G上

- 比例两个矩阵等式的第m列，我们有

$$Au_m = h_{1m}u_1 + \cdots + h_{mm}u_m + h_{m+1,m}u_{m+1}$$

$$Av_m = g_{1m}v_1 + \cdots + g_{mm}v_m + g_{m+1,m}v_{m+1}$$

- 在上式两边分别左乘u_i^T和v_i^T，可得

$$h_{im} = u_i^T Au_m, \quad g_{im} = v_i^T Av_m, i = 1, \ldots, m$$

U, V的关系反应到H, G上

- 比例两个矩阵等式的第m列，我们有

$$Au_m = h_{1m}u_1 + \cdots + h_{mm}u_m + h_{m+1,m}u_{m+1}$$
$$Av_m = g_{1m}v_1 + \cdots + g_{mm}v_m + g_{m+1,m}v_{m+1}$$

- 在上式两边分别左乘u_i^T和v_i^T，可得

$$h_{im} = u_i^T Au_m, \quad g_{im} = v_i^T Av_m, i = 1, \ldots, m$$

- 根据归纳假设，可得$h_{im} = \varepsilon_i \varepsilon_m g_{im}$，$i = 1, \ldots, m$

第$m+1$列的关系

- 从而我们有

$$\begin{aligned} h_{m+1,m}u_{m+1} &= \varepsilon_m(Av_m - \varepsilon_1^2 g_{1m}v_1 - \cdots - \varepsilon_m^2 g_{mm}v_m) \\ &= \varepsilon_m(Av_m - g_{1m}v_1 - \cdots - g_{mm}v_m) \\ &= \varepsilon_m g_{m+1,m}v_{m+1} \end{aligned}$$

第$m+1$列的关系

- 从而我们有

$$\begin{aligned} h_{m+1,m}u_{m+1} &= \varepsilon_m(Av_m - \varepsilon_1^2 g_{1m}v_1 - \cdots - \varepsilon_m^2 g_{mm}v_m) \\ &= \varepsilon_m(Av_m - g_{1m}v_1 - \cdots - g_{mm}v_m) \\ &= \varepsilon_m g_{m+1,m}v_{m+1} \end{aligned}$$

- 由此即得$|h_{m+1,m}| = |g_{m+1,m}|$

第$m+1$列的关系

- 从而我们有

$$\begin{aligned} h_{m+1,m}u_{m+1} &= \varepsilon_m(Av_m - \varepsilon_1^2 g_{1m}v_1 - \cdots - \varepsilon_m^2 g_{mm}v_m) \\ &= \varepsilon_m(Av_m - g_{1m}v_1 - \cdots - g_{mm}v_m) \\ &= \varepsilon_m g_{m+1,m}v_{m+1} \end{aligned}$$

- 由此即得$|h_{m+1,m}| = |g_{m+1,m}|$
- 而$h_{m+1,m} \neq 0$, 所以我们有$u_{m+1} = \varepsilon_{m+1}v_{m+1}$, 其中$\varepsilon_{m+1} = 1$或$-1$

变换矩阵的第一列

- 根据前面给出的分解方法，我们知道

$$H_1 = \begin{pmatrix} 1 & 0 \\ 0 & \tilde{H}_1 \end{pmatrix}$$

变换矩阵的第一列

- 根据前面给出的分解方法，我们知道

$$H_1 = \begin{pmatrix} 1 & 0 \\ 0 & \tilde{H}_1 \end{pmatrix}$$

- 所以最终的变换矩阵也具有形式

$$H = \begin{pmatrix} 1 & 0 \\ 0 & \tilde{H} \end{pmatrix}$$

变换矩阵的第一列

- 根据前面给出的分解方法，我们知道

$$H_1 = \begin{pmatrix} 1 & 0 \\ 0 & \tilde{H}_1 \end{pmatrix}$$

- 所以最终的变换矩阵也具有形式

$$H = \begin{pmatrix} 1 & 0 \\ 0 & \tilde{H} \end{pmatrix}$$

- 实际上我们可以把H的第一列取为任意单位向量

变换矩阵的第一列

- 根据前面给出的分解方法，我们知道

$$H_1 = \begin{pmatrix} 1 & 0 \\ 0 & \tilde{H}_1 \end{pmatrix}$$

- 所以最终的变换矩阵也具有形式

$$H = \begin{pmatrix} 1 & 0 \\ 0 & \tilde{H} \end{pmatrix}$$

- 实际上我们可以把H的第一列取为任意单位向量
  - 根据H的第一列确定一个正交阵Q

变换矩阵的第一列

- 根据前面给出的分解方法，我们知道
$$H_1 = \begin{pmatrix} 1 & 0 \\ 0 & \tilde{H}_1 \end{pmatrix}$$
- 所以最终的变换矩阵也具有形式
$$H = \begin{pmatrix} 1 & 0 \\ 0 & \tilde{H} \end{pmatrix}$$
- 实际上我们可以把H的第一列取为任意单位向量
  - 根据H的第一列确定一个正交阵Q
  - 那么对Q^TAQ进行前述上Hessenberg化，设变换矩阵为$\tilde{H}$, $\tilde{H}$具有上述形式，那么$H = Q\tilde{H}$ 的第一列就是所指定的单位向量

不可约上Hessenerg矩阵

- 一个上Hessenberg矩阵的下次对角元均不为零，则称为**不可约的**

不可约上Hessenerg矩阵

- 一个上Hessenberg矩阵的下次对角元均不为零，则称为不可约的
- 之所以如此定义，如果有一个下次对角元为零，那么可以分别考虑上下子阵的特征值问题

不可约上Hessenerg矩阵

- 一个上Hessenberg矩阵的下次对角元均不为零，则称为不可约的
- 之所以如此定义，如果有一个下次对角元为零，那么可以分别考虑上下子阵的特征值问题
- 唯一性定理表明：如果$Q^TAQ = H$为不可约的上Hessenberg矩阵，则Q和H在相差一个正负号的意义下完全由Q的第一列确定

不可约上Hesseneng矩阵

- 一个上Hessenberg矩阵的下次对角元均不为零，则称为不可约的
- 之所以如此定义，如果有一个下次对角元为零，那么可以分别考虑上下子阵的特征值问题
- 唯一性定理表明：如果$Q^TAQ = H$为不可约的上Hessenberg矩阵，则Q和H在相差一个正负号的意义下完全由Q的第一列确定
- 这是得以建立隐式QR方法的关键所在

上Hessenberg矩阵的QR分解

- 假设$H \in \mathbb{R}^{n\times n}$是上Hessenberg矩阵

上Hessenberg矩阵的QR分解

- 假设$H \in \mathbb{R}^{n\times n}$是上Hessenberg矩阵
- H的QR分解可以通过$n-1$个平面旋转变换完成

$$P_{n-1,n}P_{n-2,n-1}\cdots P_{12}H = R$$

上Hessenberg矩阵的QR分解

- 假设$H \in \mathbb{R}^{n\times n}$是上Hessenberg矩阵
- H的QR分解可以通过$n-1$个平面旋转变换完成

$$P_{n-1,n}P_{n-2,n-1}\cdots P_{12}H = R$$

- 令$Q = (P_{n-1,n}P_{n-2,n-1}\cdots P_{12})^T$，则$H = QR$就是$H$的QR分解

上Hessenberg矩阵的QR分解

- 假设$H \in \mathbb{R}^{n\times n}$是上Hessenberg矩阵
- H的QR分解可以通过$n-1$个平面旋转变换完成

$$P_{n-1,n}P_{n-2,n-1}\cdots P_{12}H = R$$

- 令$Q = (P_{n-1,n}P_{n-2,n-1}\cdots P_{12})^T$，则$H = QR$就是$H$的QR分解
- 为了完成一次QR迭代，还需要计算$\tilde{H} = RQ = RP_{12}^TP_{23}^T\cdots P_{n-1,n}^T$

RQ的计算结果

- 由于P_{12}是$(1,2)$坐标平面内的旋转变换，因此RP_{12}^T仅有前两列与R不同，而RP_{12}^T的前两列是由R的前两列的线性组合构成，R为上三角阵，所以RP_{12}^T的第一个下次对角元非零

RQ的计算结果

- 由于P_{12}是$(1,2)$坐标平面内的旋转变换，因此RP_{12}^T仅有前两列与R不同，而RP_{12}^T的前两列是由R的前两列的线性组合构成，R为上三角阵，所以RP_{12}^T的第一个下次对角元非零
- 因此RQ仍是一个上Hessenberg矩阵

RQ的计算结果

- 由于P_{12}是$(1,2)$坐标平面内的旋转变换，因此RP_{12}^T仅有前两列与R不同，而RP_{12}^T的前两列是由R的前两列的线性组合构成，R为上三角阵，所以RP_{12}^T的第一个下次对角元非零
- 因此RQ仍是一个上Hessenberg矩阵
- 上Hessenberg矩阵经一次QR迭代后还是上Hessenberg矩阵，计算运算量为$O(n^2)$

QR分解开始消失

- 从一个不可约上Hessenberg矩阵H_k出发，经过一次QR迭代，得到新的上Hessenberg矩阵H_{k+1}，

$$H_{k+1} = Q_k^T H_k Q_k$$

QR分解开始消失

- 从一个不可约上Hessenberg矩阵H_k出发，经过一次QR迭代，得到新的上Hessenberg矩阵H_{k+1}，

$$H_{k+1} = Q_k^T H_k Q_k$$

- 这里的QR迭代实际上就是连续$n-1$次平面旋转操作，不同于标准的QR分解

QR分解开始消失

- 从一个不可约上Hessenberg矩阵H_k出发，经过一次QR迭代，得到新的上Hessenberg矩阵H_{k+1}，

$$H_{k+1} = Q_k^T H_k Q_k$$

- 这里的QR迭代实际上就是连续$n-1$次平面旋转操作，不同于标准的QR分解
- 回忆：上Hessenberg矩阵分解的唯一性

QR分解开始消失

- 从一个不可约上Hessenberg矩阵H_k出发，经过一次QR迭代，得到新的上Hessenberg矩阵H_{k+1}，

$$H_{k+1} = Q_k^T H_k Q_k$$

- 这里的QR迭代实际上就是连续$n-1$次平面旋转操作，不同于标准的QR分解
- 回忆：上Hessenberg矩阵分解的唯一性
- 知道了Q_k的第一列，就几乎完全确定了整个矩阵Q_k

加速收敛：原点位移

- 基本的QR方法是线性收敛的

加速收敛：原点位移

- 基本的QR方法是线性收敛的
- 为了加速收敛，我们引进原点位移

加速收敛：原点位移

- 基本的QR方法是线性收敛的
- 为了加速收敛，我们引进原点位移
- 带原点位移的QR迭代格式如下：

$$H_m - \mu_m I = Q_m R_m$$
$$H_{m+1} = R_m Q_m + \mu_m I$$

其中$H_0 = H$为给定的上Hessenberg矩阵

位移的选取

- H_m为上Hessenberg矩阵，故其最后一行仅有两个非零元素$h_{n,n-1}^{(m)}$，$h_{nn}^{(m)}$

位移的选取

- H_m为上Hessenberg矩阵，故其最后一行仅有两个非零元素$h_{n,n-1}^{(m)}$，$h_{nn}^{(m)}$
- 若QR算法收敛，则当m充分大时$h_{n,n-1}^{(m)}$应很小，因而$h_{nn}^{(m)}$接近于H的一个特征值

位移的选取

- H_m为上Hessenberg矩阵，故其最后一行仅有两个非零元素$h^{(m)}_{n,n-1}$，$h^{(m)}_{nn}$
- 若QR算法收敛，则当m充分大时$h^{(m)}_{n,n-1}$应很小，因而$h^{(m)}_{nn}$接近于H的一个特征值
- 因此我们可以取$\mu_m = h^{(m)}_{nn}$

位移的选取

- H_m为上Hessenberg矩阵，故其最后一行仅有两个非零元素$h_{n,n-1}^{(m)}$，$h_{nn}^{(m)}$
- 若QR算法收敛，则当m充分大时$h_{n,n-1}^{(m)}$应很小，因而$h_{nn}^{(m)}$接近于H的一个特征值
- 因此我们可以取$\mu_m = h_{nn}^{(m)}$
- 可以证明：若$h_{n,n-1}^{(m)} = \varepsilon$很小，则一次带原点位移的QR迭代后，$h_{n,n-1}^{(m+1)} = O(\varepsilon^2)$, 即收敛速度从线性收敛加速到二次收敛

位移的选取

- H_m为上Hessenberg矩阵，故其最后一行仅有两个非零元素$h_{n,n-1}^{(m)}$，$h_{nn}^{(m)}$
- 若QR算法收敛，则当m充分大时$h_{n,n-1}^{(m)}$应很小，因而$h_{nn}^{(m)}$接近于H的一个特征值
- 因此我们可以取$\mu_m = h_{nn}^{(m)}$
- 可以证明：若$h_{n,n-1}^{(m)} = \varepsilon$很小，则一次带原点位移的QR迭代后，$h_{n,n-1}^{(m+1)} = O(\varepsilon^2)$, 即收敛速度从线性收敛加速到二次收敛
- 课本例6.4.2的程序演示

双重步位移

- 单步原点位移的QR迭代具有严重的缺点：
  若A具有复共轭特值，则实位移一般并不能起到加速的作用

双重步位移

- 单步原点位移的QR迭代具有严重的缺点：若A具有复共轭特征值，则实位移一般并不能起到加速的作用
- 为此，我们引入双重步位移的QR迭代，基本想法是把两步带原点位移的QR迭代合并为一步，以避免复数运算

迭代格式

- 设$A \in \mathbb{R}^{n\times n}$, 考察如下的迭代格式:

$$
\begin{aligned}
H_1 &= Q_0^T A Q_0 && \text{上Hessenberg分解} \\
H_k - \mu_k I &= Q_k R_k && \text{QR分解} \\
H_{k+1} &= R_k Q_k + \mu_k I && k = 1, 2, \ldots
\end{aligned}
$$

迭代格式

- 设$A \in \mathbb{R}^{n\times n}$，考察如下的迭代格式：

$$
\begin{aligned}
H_1 &= Q_0^T A Q_0 && \text{上Hessenberg分解} \\
H_k - \mu_k I &= Q_k R_k && \text{QR分解} \\
H_{k+1} &= R_k Q_k + \mu_k I && k = 1, 2, \ldots
\end{aligned}
$$

- 假设迭代格式中出现的上Hessenberg矩阵都是不可约的。否则，可分别对沿对角线的上下两个子矩阵进行QR迭代

复共轭特征值的处理

- 假设H_k的尾部2×2子矩阵

$$S_k = \begin{pmatrix} h_{mm}^{(k)} & h_{mn}^{(k)} \\ h_{nm}^{(k)} & h_{nn}^{(k)} \end{pmatrix}, \quad m = n-1$$

有一对复共轭特征值γ_1和γ_2。这时我们不能期望$h_{nn}^{(k)}$最终收敛于A的某个特征值，而且此时取$\mu_k = h_{nn}^{(k)}$也完全没有加速效果

复共轭特征值的处理

- 假设H_k的尾部2×2子矩阵

$$S_k = \begin{pmatrix} h_{mm}^{(k)} & h_{mn}^{(k)} \\ h_{nm}^{(k)} & h_{nn}^{(k)} \end{pmatrix}, \quad m = n - 1$$

  有一对复共轭特征值γ_1和γ_2。这时我们不能期望$h_{nn}^{(k)}$最终收敛于A的某个特征值，而且此时取$\mu_k = h_{nn}^{(k)}$也完全没有加速效果

- 为了加速，我们应当取γ_1或γ_2作位移，但这样就会涉及到复数运算

连续两次位移

- 为了避免复数运算，我们计划用γ_1和γ_2连续作两次位移，即进行如下迭代：

$$
\begin{aligned}
H_k - \gamma_1 I &= U_1 R_1, & G_1 &= R_1 U_1 + \gamma_1 I \\
G_1 - \gamma_2 I &= U_2 R_2, & H_{k+1} &= R_2 U_2 + \gamma_2 I
\end{aligned}
$$

连续两次位移

- 为了避免复数运算，我们计划用γ_1和γ_2连续作两次位移，即进行如下迭代：

$$H_k - \gamma_1 I = U_1 R_1, \qquad G_1 = R_1 U_1 + \gamma_1 I$$

$$G_1 - \gamma_2 I = U_2 R_2, \qquad H_{k+1} = R_2 U_2 + \gamma_2 I$$

- 由此格式，记$M = (H_k - \gamma_1 I)(H_k - \gamma_2 I)$, $Q = U_1 U_2$, $R = R_2 R_1$, 则

$$M = QR, \quad H_{k+1} = Q^* H_k Q$$

验算

$$
\begin{aligned}
QR &= U_1U_2R_2R_1 = U_1(G_1 - \gamma_2 I)R_1 \\
&= U_1(R_1U_1 + (\gamma_1 - \gamma_2)I)R_1 \\
&= U_1R_1(U_1R_1 + (\gamma_1 - \gamma_2)I) \\
&= (H_k - \gamma_1 I)(H_k - \gamma_2 I) = M \\
U_1G_1 &= U_1(R_1U_1 + \gamma_1 I) = (U_1R_1 + \gamma_1 I)U_1 = H_kU_1 \\
U_2H_{k+1} &= G_1U_2
\end{aligned}
$$

实矩阵M

- 根据M的定义，我们有

$$M = H_k^2 - sH_k + tI$$

其中$s = \gamma_1 + \gamma_2 = h_{mm}^{(k)} + h_{nn}^{(k)} \in \mathbb{R}$,

$t = \gamma_1\gamma_2 = \det S_k \in \mathbb{R}$

实矩阵M

- 根据M的定义，我们有

$$M = H_k^2 - sH_k + tI$$

其中$s = \gamma_1 + \gamma_2 = h_{mm}^{(k)} + h_{nn}^{(k)} \in \mathbb{R}$,
$t = \gamma_1\gamma_2 = \det S_k \in \mathbb{R}$

- 所以M是实矩阵

实矩阵M

- 根据M的定义，我们有
$$M = H_k^2 - sH_k + tI$$
其中$s = \gamma_1 + \gamma_2 = h_{mm}^{(k)} + h_{nn}^{(k)} \in \mathbb{R}$,
$t = \gamma_1\gamma_2 = \det S_k \in \mathbb{R}$
- 所以M是实矩阵
- 根据QR分解的性质，即使γ_1和γ_2均不是H_k的特征值，并假定在计算过程中R_1和R_2 的对角元均取为正数，那么Q也是实矩阵

实矩阵M

- 根据M的定义，我们有

$$M = H_k^2 - sH_k + tI$$

其中$s = \gamma_1 + \gamma_2 = h_{mm}^{(k)} + h_{nn}^{(k)} \in \mathbb{R}$,
$t = \gamma_1\gamma_2 = \det S_k \in \mathbb{R}$

- 所以M是实矩阵
- 根据QR分解的性质，即使γ_1和γ_2均不是H_k的特征值，并假定在计算过程中R_1和R_2 的对角元均取为正数，那么Q也是实矩阵
- 所以H_{k+1}也是实矩阵

初步结论

- 在没有误差的情况下，用γ_1和γ_2连续作两次位移进行QR迭代产生的H_{k+1}仍是实的上Hessenberg矩阵

初步结论

- 在没有误差的情况下，用γ_1和γ_2连续作两次位移进行QR迭代产生的H_{k+1}仍是实的上Hessenberg矩阵
- 但是，实际计算时，由于舍入误差的影响，如此得到的H_{k+1}一般并不是实矩阵

初步结论

- 在没有误差的情况下，用γ_1和γ_2连续作两次位移进行QR迭代产生的H_{k+1}仍是实的上Hessenberg矩阵
- 但是，实际计算时，由于舍入误差的影响，如此得到的H_{k+1}一般并不是实矩阵
- 为了确保得到的H_{k+1}仍是实矩阵，我们对迭代格式进行修改

基于M的迭代

- 修改后的迭代格式如下：

基于M的迭代

- 修改后的迭代格式如下：
  1. 计算$M = H_k^2 - sH_k + tI$

基于M的迭代

- 修改后的迭代格式如下：
  1. 计算$M = H_k^2 - sH_k + tI$
  2. 计算M的QR分解$M = QR$;

基于M的迭代

- 修改后的迭代格式如下：
  1. 计算$M = H_k^2 - sH_k + tI$
  2. 计算M的QR分解$M = QR$;
  3. 计算$H_{k+1} = Q^T H_k Q$

基于M的迭代

- 修改后的迭代格式如下：
  1. 计算$M = H_k^2 - sH_k + tI$
  2. 计算M的QR分解$M = QR$;
  3. 计算$H_{k+1} = Q^T H_k Q$
- M的下带宽是2，即次对角元和次次对角元非零

基于M的迭代

- 修改后的迭代格式如下：
  1. 计算$M = H_k^2 - sH_k + tI$
  2. 计算M的QR分解$M = QR$;
  3. 计算$H_{k+1} = Q^T H_k Q$
- M的下带宽是2，即次对角元和次次对角元非零
- 如此计算第一步形成M的运算量就是$O(n^3)$

降低运算量的想法

- 修改后的迭代格式运算量比较大

降低运算量的想法

- 修改后的迭代格式运算量比较大
- 想法：已有结论告诉我们，无论采用何种方法求正交矩阵$\tilde{Q}$使得$\tilde{Q}^T H_k \tilde{Q} = \tilde{H}_{k+1}$为上Hessenberg矩阵，只要保证$\tilde{Q}$的第一列与$Q$的第一列相同，则$\tilde{H}_{k+1}$就与$H_{k+1}$本质上是一样的

降低运算量的想法

- 修改后的迭代格式运算量比较大
- 想法：已有结论告诉我们，无论采用何种方法求正交矩阵$\tilde{Q}$使得$\tilde{Q}^T H_k \tilde{Q} = \tilde{H}_{k+1}$为上Hessenberg矩阵，只要保证$\tilde{Q}$的第一列与$Q$的第一列相同，则$\tilde{H}_{k+1}$就与$H_{k+1}$本质上是一样的
- 而这要求H_{k+1}是不可约的

H_k与H_{k+1}不可约性的关系

定理

若H_k是不可约的上Hessenberg矩阵，且γ_1和γ_2均非H_k的特征值，则H_{k+1} 也是不可约的上Hessenberg矩阵

定理证明

采用反证法

- 记$H_{k+1} = (\tilde{h}_{ij})$，并假定存在$r$ $(1 \leqslant r \leqslant n-1)$使得$\tilde{h}_{r+1,r} = 0$，而$\tilde{h}_{i+1,i} \neq 0$ $(i = 1, 2, \ldots, r-1)$

定理证明

采用反证法

- 记$H_{k+1} = (\tilde{h}_{ij})$, 并假定存在$r$ $(1 \leqslant r \leqslant n-1)$使得$\tilde{h}_{r+1,r} = 0$, 而$\tilde{h}_{i+1,i} \neq 0$ $(i = 1, 2, \ldots, r-1)$
- 比较等式$H_k Q = QH_{k+1}$两边矩阵的前r列，我们有

$$H_k q_j = \tilde{h}_{1j} q_1 + \cdots + \tilde{h}_{jj} q_j + \tilde{h}_{j+1,j} q_{j+1},$$
$$j = 1, \ldots, r-1$$
$$H_k q_r = \tilde{h}_{1r} q_1 + \tilde{h}_{2r} q_2 + \cdots + \tilde{h}_{rr} q_r$$

- 由此考虑$r+1$个向量$q_1, H_k q_1, \ldots, H_k^r q_1$，它们均可以表示为$q_1, \ldots, q_r$的线性组合，因此存在不全为零的$\alpha_i$，

$$\left(\sum_{i=0}^{r} \alpha_i H_k^i\right) q_1 = 0$$

- 由此考虑$r+1$个向量$q_1, H_k q_1, \ldots, H_k^r q_1$, 它们均可以表示为$q_1, \ldots, q_r$的线性组合，因此存在不全为零的$\alpha_i$,

$$\left(\sum_{i=0}^{r} \alpha_i H_k^i\right) q_1 = 0$$

- 注意到q_r只出现在$H_k^{r-1} q_1$和$H_k^r q_1$中，因此必有$\alpha_r \neq 0$; 否则所有系数都是零

- 由$M = QR$可得$q_1 = \dfrac{1}{r_{11}} Me_1$. 将其代入到上式，并注意$M$也是$H_k$的多项式，我们有$My = 0$, 其中

$$y = \left(\sum_{i=0}^{r} \alpha_i H_k^i \right) e_1$$

- 由$M=QR$可得$q_1=\dfrac{1}{r_{11}}Me_1$. 将其代入到上式，并注意$M$也是$H_k$的多项式，我们有$My=0$, 其中

$$y=\left(\sum_{i=0}^{r}\alpha_i H_k^i\right)e_1$$

- 记$H_k=(h_{ij})$, 直接计算可知y的第$r+1$个分量为

$$\alpha_r h_{21}h_{32}\cdots h_{r+1,r}\neq 0$$

这就是说方程组$My=0$有非零解，这与γ_1和γ_2均非H_k的特征值，从而M非奇异矛盾

实现从H_k到H_{k+1}的新方法

- 由于Q的第一列是M的第一列单位化得到的

实现从H_k到H_{k+1}的新方法

- 由于Q的第一列是M的第一列单位化得到的
- 由$M = H_k^2 - sH_k + tI$可知M的第一列为

$$Me_1 = (\xi_1, \xi_2, \xi_3, 0, \dots, 0)^T$$

其中

$$\xi_1 = (h_{11}^{(k)})^2 + h_{12}^{(k)} h_{21}^{(k)} - s h_{11}^{(k)} + t$$
$$\xi_2 = h_{21}^{(k)} (h_{11}^{(k)} + h_{22}^{(k)} - s)$$
$$\xi_3 = h_{21}^{(k)} h_{32}^{(k)}$$

- 在M的QR分解中，如果Householder变换P_0把Me_1变为αe_1，那么P_0的第一列与Me_1共线

- 在M的QR分解中，如果Householder变换P_0把Me_1变为αe_1，那么P_0的第一列与Me_1共线
    - 利用Householder变换的定义与意义，可直接证明（作为练习）

- 在M的QR分解中，如果Householder变换P_0把Me_1变为αe_1，那么P_0的第一列与Me_1共线
    - 利用Householder变换的定义与意义，可直接证明（作为练习）
- 这就是说P_0的第一列就可以作为Q的第一列

- 在M的QR分解中，如果Householder变换P_0把Me_1变为αe_1，那么P_0的第一列与Me_1共线
    - 利用Householder变换的定义与意义，可直接证明（作为练习）
- 这就是说P_0的第一列就可以作为Q的第一列
- 根据Householder变换的理论，可以确定P_0的具体表示

P_0的表示

- P_0可以按下述方式确定：

$$P_0 = \mathrm{diag}(\tilde{P}_0, I_{n-3})$$

其中

$$\tilde{P}_0 = I_3 - \beta v v^T, \qquad v = (\xi_1 - \alpha, \xi_2, \xi_3)^T,$$

$$\alpha = \sqrt{\xi_1^2 + \xi_2^2 + \xi_3^2}, \quad \beta = 2/(v^T v)$$

- 现令$B = P_0 H_k P_0$, 则我们只要找到第一列为e_1的正交矩阵$\tilde{Q}$, 使得$\tilde{Q}^T B \tilde{Q} = \tilde{H}$为上Hessenberg矩阵，那么$\tilde{H}$就是所期望的$H_{k+1}$

- 现令$B = P_0 H_k P_0$, 则我们只要找到第一列为e_1的正交矩阵$\tilde{Q}$, 使得$\tilde{Q}^T B \tilde{Q} = \tilde{H}$为上Hessenberg矩阵，那么$\tilde{H}$就是所期望的$H_{k+1}$
- 根据前面给出的约化一个矩阵为上Hessenberg矩阵的方法以及上Hessenberg矩阵的唯一性可知，这是很容易做到的

- 现令$B = P_0 H_k P_0$，则我们只要找到第一列为e_1的正交矩阵$\tilde{Q}$，使得$\tilde{Q}^T B \tilde{Q} = \tilde{H}$为上Hessenberg矩阵，那么$\tilde{H}$就是所期望的$H_{k+1}$
- 根据前面给出的约化一个矩阵为上Hessenberg矩阵的方法以及上Hessenberg矩阵的唯一性可知，这是很容易做到的
- 而且算法的运算量只是$O(n^2)$

Francis算法

- 事实上，由于用P_0对H_k进行相似变换为B，只是改变了H的前三列和前三行，因此B比上Hessenberg矩阵只是多了三个可能的非零元素

Francis算法

- 事实上，由于用P_0对H_k进行相似变换为B，只是改变了H的前三列和前三行，因此B比上Hessenberg矩阵只是多了三个可能的非零元素
- 由此我们可以构造Householder变换P_1把第一列多余的两个非零元素消去

Francis算法

- 事实上，由于用P_0对H_k进行相似变换为B，只是改变了H的前三列和前三行，因此B比上Hessenberg矩阵只是多了三个可能的非零元素
- 由此我们可以构造Householder变换P_1把第一列多余的两个非零元素消去
- 逐步递推下去，就可以把B化为上Hessenberg矩阵

Francis算法

- 事实上，由于用P_0对H_k进行相似变换为B，只是改变了H的前三列和前三行，因此B比上Hessenberg矩阵只是多了三个可能的非零元素
- 由此我们可以构造Householder变换P_1把第一列多余的两个非零元素消去
- 逐步递推下去，就可以把B化为上Hessenberg矩阵
- 由此给出的就是著名的Francis双重步位移的QR迭代算法

实用算法

- 前面的讨论解决了用QR方法求给定实矩阵的实Schur标准形的几个关键问题

实用算法

- 前面的讨论解决了用QR方法求给定实矩阵的实Schur标准形的几个关键问题
- 为了得到一个实用的算法，我们还需要给出一个有效的判定准则，确定迭代过程中所产生的上Hessneberg矩阵的次对角元何时可以忽略不计

实用算法

- 前面的讨论解决了用QR方法求给定实矩阵的实Schur标准形的几个关键问题
- 为了得到一个实用的算法，我们还需要给出一个有效的判定准则，确定迭代过程中所产生的上Hessneberg矩阵的次对角元何时可以忽略不计
- 一种简单而实用的准则是：当

$$|h_{i+1,i}| \leqslant (|h_{ii}| + |h_{i+1,i+1}|)\mathbf{u}$$

时，就将$h_{i+1,i}$看做零

隐式QR算法

- 把前面所有分析综合在一起，就是隐式QR算法

隐式QR算法

- 把前面所有分析综合在一起，就是隐式QR算法
- 该算法计算给定的n阶实矩阵A的实Schur分解：$Q^TAQ = T$, 其中Q是正交矩阵，T为拟上三角阵，即对角块为1×1或2×2方阵的块上三角阵，而且每个2×2的对角块必有一对复共轭特征值

算法注解

- 实际计算的统计表明，这一算法每分离出一个1×1或2×2子矩阵平均需要2次QR迭代

算法注解

- 实际计算的统计表明，这一算法每分离出一个1×1或2×2子矩阵平均需要2次QR迭代
- 只计算特征值，运算量平均为$10n^3$；如果还需要Q，总运算量平均约为$25n^3$

算法注解

- 实际计算的统计表明，这一算法每分离出一个1×1或2×2子矩阵平均需要2次QR迭代
- 只计算特征值，运算量平均为$10n^3$；如果还需要Q，总运算量平均约为$25n^3$
- 算法是稳定的