GMRES: Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems

National Taiwan Normal University

講師：林文偉

前言

Theorem 7.1 (Implicit Q theorem)

Let $AV_1 = V_1H_1$ and $AV_2 = V_2H_2$, where H_1, H_2 are Hessenberg and V_1, V_2 are unitary with $V_1e_1 = V_2e_1 = q_1$. Then $V_1 = V_2$ and $H_1 = H_2$.

▶ Proof

$$A \begin{bmatrix} v_1 & v_2 & \cdots & v_n \end{bmatrix} = \begin{bmatrix} v_1 & v_2 & \cdots & v_n \end{bmatrix} \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1n} \\ h_{21} & h_{22} & & h_{2n} \\ & \ddots & \ddots & \vdots \\ & & h_{n,n-1} & h_{nn} \end{bmatrix}$$

with

$$v_i^T v_j = \delta_{ij}, \quad i,j = 1, \ldots, n$$

前言

[Arnoldi algorithm]

Input: Given v_1 with $\|v_1\|_2 = 1$;

Output: Arnoldi factorization: $AV_k = V_k H_k + h_{k+1,k} v_{k+1} e_k^T$.

Iterate $j = 1, 2, \cdots$,

compute $h_{ij} = (Av_j, v_i)$ for $i = 1, 2, \cdots, j$,

$\tilde{v}_{j+1} = Av_j - \sum_{i=1}^{j} h_{ij} v_i$,

$h_{j+1,j} = \|\tilde{v}_{j+1}\|_2$,

$v_{j+1} = \tilde{v}_{j+1} / h_{j+1,j}$.

End;

前言

Remark 1

(a) *Let* $V_k = [v_1, \cdots, v_k] \in \mathbb{R}^{n \times k}$ *where* v_j, *for* $j = 1, \ldots, k$, *is generated by Arnoldi algorithm. Then* $H_k \equiv V_k^T A V_k$ *is upper* $k \times k$ *Hessenberg.*

(b) *Arnoldi's original method was a Galerkin method for approximate the eigenvalue of* A *by* H_k.

前言

In order to solve $Ax = b$ by the Galerkin method using $< K_k > \equiv < V_k >$, we seek an approximate solution $x_k = x_0 + z_k$ with $z_k \in K_k = < r_0, Ar_0, \cdots, A^{k-1}r_0 >$ and $r_0 = b - Ax_0$.

Definition 7.2

$\{x_k\}$ is said to be satisfied the Galerkin condition if $r_k \equiv b - Ax_k$ is orthogonal to K_k for each k.

The Galerkin method can be stated as that find

$$x_k = x_0 + z_k \quad \text{with} \quad z_k \in V_k \tag{1}$$

such that

$$(b - Ax_k, v) = 0, \quad \forall\, v \in V_k,$$

前言

which is equivalent to find

$$z_k \equiv V_k y_k \in \mathcal{V}_k \tag{2}$$

such that

$$(r_0 - A z_k, v) = 0, \quad \forall\, v \in \mathcal{V}_k. \tag{3}$$

Substituting (2) into (3), we get

$$V_k^T (r_0 - A V_k y_k) = 0,$$

which implies that

$$y_k = (V_k^T A V_k)^{-1} \|r_0\| e_1. \tag{4}$$

前言

Since V_k is computed by the Arnoldi algorithm with $v_1 = r_0/\|r_0\|$, y_k in (4) can be represented as

$$y_k = H_k^{-1}\|r_0\|e_1.$$

Substituting it into (2) and (1), we get

$$x_k = x_0 + V_k H_k^{-1}\|r_0\|e_1.$$

Using the result that $AV_k = V_k H_k + h_{k+1,k}v_{k+1}e_k^T$, r_k can be reformulated as

$$\begin{aligned} r_k &= b - Ax_k = r_0 - AV_k y_k = r_0 - (V_k H_k + h_{k+1,k}v_{k+1}e_k^T)y_k \\ &= r_0 - V_k\|r_0\|e_1 - h_{k+1,k}e_k^T y_k v_{k+1} = -(h_{k+1,k}e_k^T y_k)v_{k+1}. \end{aligned}$$

一、The generalized minimal residual (GMRES) algorithm

The approximate solution of the form $x_0 + z_k$, which minimizes the residual norm over $z_k \in K_k$, can in principle be obtained by following algorithms:

- The ORTHODIR algorithm of Jea and Young;
- the generalized conjugate residual method (GCR);
- GMRES.

Let

$$V_k = [v_1, \cdots, v_k], \quad \tilde{H}_k = \begin{bmatrix} h_{1,1} & h_{1,2} & \cdots & h_{1,k} \\ h_{2,1} & h_{2,2} & \cdots & h_{2,k} \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & h_{k,k-1} & h_{k,k} \\ 0 & \cdots & 0 & h_{k+1,k} \end{bmatrix} \in \mathbb{R}^{(k+1)\times k}.$$

一、The generalized minimal residual (GMRES) algorithm

By Arnoldi algorithm, we have

$$AV_k = V_{k+1}\tilde{H}_k. \tag{5}$$

To solve the least square problem:

$$\min_{z \in K_k} \|r_o - Az\|_2 = \min_{z \in K_k} \|b - A(x_o + z)\|_2, \tag{6}$$

where $K_k = < r_o, Ar_o, \cdots, A^{k-1}r_o > = < v_1, \cdots, v_k >$ with $v_1 = \frac{r_o}{\|r_o\|_2}$.

一、The generalized minimal residual (GMRES) algorithm

Set $z = V_k y$, the least square problem (6) is equivalent to

$$\min_{y \in \mathbb{R}^k} J(y) = \min_{y \in \mathbb{R}^k} \|\beta v_1 - A V_k y\|_2, \quad \beta = \|r_o\|_2. \tag{7}$$

Using (5), we have

$$J(y) = \|V_{k+1}[\beta e_1 - \tilde{H}_k y]\|_2 = \|\beta e_1 - \tilde{H}_k y\|_2. \tag{8}$$

Hence, the solution of the least square (6) is

$$x_k = x_o + V_k y_k,$$

where y_k minimize the function $J(y)$ defined by (8) over $y \in \mathbb{R}^k$.

一、The generalized minimal residual (GMRES) algorithm

[GMRES algorithm]

Input: choose x_0, compute $r_0 = b - Ax_0$ and $v_1 = r_0/\|r_0\|$;

Output: solution of linear system $Ax = b$.

Iterate $j = 1, 2, \cdots, k$,

compute $h_{ij} = (Av_j, v_i)$ for $i = 1, 2, \cdots, j$,

$\tilde{v}_{j+1} = Av_j - \sum_{i=1}^{j} h_{ij} v_i$,

$h_{j+1,j} = \|\tilde{v}_{j+1}\|_2$,

$v_{j+1} = \tilde{v}_{j+1}/h_{j+1,j}$.

End;

Form the solution:

$x_k = x_0 + V_k y_k$, where y_k minimizes $J(y)$ in (8).

Difficulties: when k is increasing, storage for v_j, like k, the number of multiplications is like $\frac{1}{2}k^2 N$.

一、The generalized minimal residual (GMRES) algorithm

[GMRES(m) algorithm]

Input: choose x_0, compute $r_0 = b - Ax_0$ and $v_1 = r_0/\|r_0\|$;
Output: solution of linear system $Ax = b$.
Iterate $j = 1, 2, \cdots, m$,
compute $h_{ij} = (Av_j, v_i)$ for $i = 1, 2, \cdots, j$,
$\tilde{v}_{j+1} = Av_j - \sum_{i=1}^{j} h_{ij} v_i$,
$h_{j+1,j} = \|\tilde{v}_{j+1}\|_2$,
$v_{j+1} = \tilde{v}_{j+1}/h_{j+1,j}$.
End;
Form the solution:

$$x_m = x_0 + V_m y_m, \text{ where } y_m \text{ minimizes } \| \beta e_1 - \widetilde{H}_m y \| \text{ for } y \in \mathbb{R}^m.$$

Restart: Compute $r_m = b - Ax_m$, if $\|r_m\|$ is small , then stop,
else , Compute $x_0 = x_m$ and $v_1 = r_m/ \| r_m \|$, GoTo Iterate step.

二、Practical Implementation: Consider QR factorization of $\widetilde{H}_k$

Consider the matrix $\widetilde{H}_k$. We want to solve the least squares problem:

$$\min_{y \in \mathbb{R}^k} \| \beta e_1 - \widetilde{H}_k y \|_2$$

Assume Givens rotations F_i , $i = 1 \ldots, j$ such that

$$F_j \cdots F_1 \widetilde{H}_j = F_j \ldots F_1 \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times \end{bmatrix} = \begin{bmatrix} \times & \times & \times & \times \\ & \times & \times & \times \\ & & \times & \times \\ & & & \times \\ & & & \times \end{bmatrix} \equiv R_j \in \mathbb{R}^{(j+1) \times j}.$$

二、Practical Implementation: Consider QR factorization of $\widetilde{H}_K$

In order to obtain R_{j+1} we must start by premultiptying the new column by the previous rotations.

$$\widetilde{H}_{j+1} = \begin{bmatrix} \times & \times & \times & \times & + \\ \times & \times & \times & \times & + \\ 0 & \times & \times & \times & + \\ 0 & 0 & \times & \times & + \\ 0 & 0 & 0 & \times & + \\ \hline 0 & 0 & 0 & 0 & + \end{bmatrix} \Rightarrow F_j \ldots F_i \widetilde{H}_{j+1} = \begin{bmatrix} \times & \times & \times & \times & + \\ & \times & \times & \times & + \\ & & \times & \times & + \\ & & & \times & + \\ & & & 0 & r \\ & & & 0 & h \end{bmatrix}$$

The principal upper $(j+1) \times j$ submatrix is nothing but R_j, and $h := h_{j+2,j+1}$ is not affected by the previous rotations. The next rotation F_{j+1} defined by

$$\begin{cases} c_{j+1} & \equiv & r/(r^2 + h^2)^{1/2}, \\ s_{j+1} & = & -h/(r^2 + h^2)^{1/2}. \end{cases}$$

二、Practical Implementation: Consider QR factorization of $\widetilde{H}_k$

Thus, after k steps of the above process, we have achieved

$$Q_k \widetilde{H}_k = R_k$$

where Q_k is a $(k+1) \times (k+1)$ unitary matrix and

$$J(y) = \| \beta e_1 - \widetilde{H}_k y \| = \| Q_k[\beta e_1 - \widetilde{H}_k y] \| = \| g_k - R_k y \|, \qquad (9)$$

where $g_k \equiv Q_k \beta e_1$. Since the last row of R_k is a zero row, the minimization of (9) is achieved at $y_k = \widetilde{R}_k^{-1} \widetilde{g}_k$, where $\widetilde{R}_k$ and $\widetilde{g}_k$ are removed the last row of R_k and the last component of g_k, respectively.

Proposition 7.3

$\| r_k \| = \| b - Ax_k \| = |$ *The (k+1)-st component of* g_k $|$.

二、Practical Implementation: Consider QR factorization of $\tilde{H}_k$

Proposition 7.4

The solution x_j produced by GMRES at step j is exact which is equivalent to

(i) *The algorithm breaks down at step j,*

(ii) $\tilde{v}_{j+1} = 0$,

(iii) $h_{j+1,j} = 0$,

(iv) *The degree of the minimal polynomial of r_0 is j.*

Corollary 7.5

For an $n \times n$ problem GMRES terminates at most n steps.

This uncommon type of breakdown is sometimes referred to as a "Lucky" breakdown is the context of the Lanczos algorithm.

二、Practical Implementation: Consider QR factorization of $\tilde{H}_k$

Proposition 7.6

Suppose that A *is diagonalizable so that* $A = XDX^{-1}$ *and let*

$$\varepsilon^{(m)} = \min_{p \in P_m, p(0)=1} \max_{\lambda_i \in \sigma(A)} |p(\lambda_i)| .$$

Then

$$\|r_{m+1}\| \leq \kappa(X)\varepsilon^{(m)} \|r_0\| ,$$

where $\kappa(X) = \|X\| \|X^{-1}\|$.

When A is positive real with symmetric part M, it holds that

$$\|r_m\| \leq [1 - \alpha/\beta]^{m/2} \|r_0\| ,$$

where $\alpha = (\lambda_{\min}(M))^2$ and $\beta = \lambda_{\max}(A^T A)$.
This proves the convergence of GMRES(m) for all m, when A is positive real.

二、Practical Implementation: Consider QR factorization of $\tilde{H}_k$

Theorem 7.7

Assume $\lambda_1, \ldots, \lambda_\nu$ of A with positive(negative) real parts and the other eigenvalues enclosed in a circle centered at C with $C > 0$ and have radius R with $C > R$. Then

$$\varepsilon^{(m)} \leq \left[\frac{R}{C}\right]^{m-\nu} \max_{j=\nu+1,\cdots,N} \prod_{i=1}^{\nu} \frac{|\lambda_i - \lambda_j|}{|\lambda_i|} \leq \left[\frac{D}{d}\right]^2 \left[\frac{R}{C}\right]^{m-\nu}$$

where

$$D = \max_{\substack{i=1,\cdots,\nu \\ j=\nu+1,\cdots,N}} |\lambda_i - \lambda_j| \quad \textit{and} \quad d = \min_{i=1,\cdots,\nu} |\lambda_i| .$$

二、Practical Implementation: Consider QR factorization of $\tilde{H}_k$

Proof.

Consider $p(z) = r(z)q(z)$ where $r(z) = (1 - z/\lambda_1)\cdots(1 - z/\lambda_\nu)$ and $q(z)$ arbitrary polynomial of $deg \le m - \nu$ such that $q(0) = 1$. Since $p(0) = 1$ and $p(\lambda_i) = 0$, for $i = 1, \ldots, \nu$, we have

$$\varepsilon^{(m)} \le \max_{j=\nu+1,\cdots,N} |p(\lambda_j)| \le \max_{j=\nu+1,\cdots,N} |r(\lambda_j)| \max_{j=\nu+1,\cdots,N} |q(\lambda_j)| .$$

It is easily seen that

$$\max_{j=\nu+1,\cdots,N} |r(\lambda_j)| = \max_{j=\nu+1,\cdots,N} \prod_{i=1}^{\nu} \frac{|\lambda_i - \lambda_j|}{|\lambda_i|} \le \left[\frac{D}{d}\right]^{\nu} .$$

By maximum principle, the maximum of $|q(z)|$ for $z \in \{\lambda_j\}_{j=\nu+1}^{N}$ is on the circle. Taking $\sigma(z) = [(C - z)/C]^{m-\nu}$ whose maximum on the circle is $(R/C)^{m-\nu}$ yields the desired result. ■

二、Practical Implementation: Consider QR factorization of $\tilde{H}_k$

Corollary 7.8

Under the assumptions of Proposition 7.6 and Theorem 7.7, GMRES(m) converges for any initial x_0 if

$$m > \nu Log\left[\frac{DC}{dR}\kappa(X)^{1/\nu}\right] \Big/ Log\left[\frac{C}{R}\right].$$

三、Appendix

Proof of Implicit Q Theorem

Let

$$A[q_1 \quad q_2 \quad \cdots \quad q_n] = [q_1 \quad q_2 \quad \cdots \quad q_n] \begin{bmatrix} h_{11} & h_{12} & \cdots & \cdots & h_{1n} \\ h_{21} & h_{22} & \ddots & & \vdots \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & h_{n-1,n} \\ 0 & \cdots & 0 & h_{n,n-1} & h_{nn} \end{bmatrix}. \quad (10)$$

三、Appendix

Then we have

$$Aq_1 = h_{11}q_1 + h_{21}q_2. \tag{11}$$

Since $q_1 \perp q_2$, it implies that

$$h_{11} = q_1^* A q_1 / q_1^* q_1.$$

From (11), we get that

$$\tilde{q_2} \equiv h_{21}q_2 = Aq_1 - h_{11}q_1.$$

That is

$$q_2 = \tilde{q_2} / \|\tilde{q_2}\|_2 \quad \text{and} \quad h_{21} = \|\tilde{q_2}\|_2.$$

三、Appendix

Similarly, from (10),

$$Aq_2 = h_{12}q_1 + h_{22}q_2 + h_{32}q_3,$$

where

$$h_{12} = q_1^* A q_2 \quad \text{and} \quad h_{22} = q_2^* A q_2.$$

Let

$$\tilde{q_3} = Aq_2 - h_{12}q_1 + h_{22}q_2.$$

Then

$$q_3 = \tilde{q_3}/\|\tilde{q_3}\|_2 \quad \text{and} \quad h_{32} = \|\tilde{q_3}\|,$$

and so on.

三、Appendix

Therefore, $[q_1, \cdots, q_n]$ are uniquely determined by q_1. Thus, uniqueness holds.
Let $K_n = [v_1, Av_1, \cdots, A^{n-1}v_1]$ with $\|v_1\|_2 = 1$ is nonsingular.
$K_n = U_n R_n$ and $U_n e_1 = v_1$. Then

$$AK_n = K_n C_n = [v_1, Av_1, \cdots, A^{n-1}v_1] \begin{bmatrix} 0 & \cdots & \cdots & 0 & * \\ 1 & \ddots & & \vdots & * \\ 0 & \ddots & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & 0 & \vdots \\ 0 & \cdots & 0 & 1 & * \end{bmatrix}. \quad (12)$$

三、Appendix

Since K_n is nonsingular, (12) implies that

$$A = K_n C_n K_n^{-1} = (U_n R_n) C_n (R_n^{-1} U_n^{-1}).$$

That is

$$A U_n = U_n (R_n C_n R_n^{-1}),$$

where $(R_n C_n R_n^{-1})$ is Hessenberg and $U_n e_1 = v_1$. Because $< U_n >=< K_n >$, find $A V_n = V_n H_n$ by any method with $V_n e_1 = v_1$, then it holds that $V_n = U_n$, i.e., $v_n^{(i)} = u_n^{(i)}$ for $i = 1, \cdots, n$.