Krylov Subspace Methods

$$A \in \mathbb{R}^{n \times n}, r \in \mathbb{R}^n$$

If A sparse or specially structured, so easy to compute

$$Ar, A(Ar), \ldots$$

i.e. $$r, Ar, A^2 r, \ldots$$

are easy to compute, then Krylov subspaces

$$\mathcal{K}_k(A, r) = \text{ span } \{r, Ar, \ldots, A^{k-1} r\}$$

are convenient nested vector subspaces (exercise: check).

Note $$y \in \mathcal{K}_k(A, r) \Leftrightarrow y = q_{k-1}(A) r_0$$

where $q_{k-1} \in \Pi_{k-1}$ (real polynomials of degree $\leq k - 1$.)

If solving $Ax = b$, (A invertible), guess x_0, $r_0 = b - Ax_0$, then look for

$$x_k \in x_0 + \mathcal{K}_k(A, r_0) \quad , r_k = b - Ax_k,\ k = 1, 2, \ldots$$

$$\begin{aligned} & & x_k &\in x_0 + \mathcal{K}_k(A, r_0) \\ \Leftrightarrow & & x_k &= x_0 + q_{k-1}(A) r_0 \\ \Leftrightarrow & x - x_k &= x - x_0 - q_{k-1}(A) r_0 \\ \Leftrightarrow & \underbrace{A(x - x_k)}_{\substack{b - Ax_k \\ \| \\ r_k}} &= \underbrace{A(x - x_0)}_{r_0} - A\, q_{k-1}(A) r_0 \end{aligned}$$

i.e. $\quad r_k = p_k(A) r_0 \quad , p_k \in \Pi_k \ , p_k(0) = 1$

$\Leftrightarrow A^{-1} r_k = p_k(A) A^{-1} r_0$ i.e. $e_k = p_k(A) e_0, e_k = x - x_k$

Krylov Subspace Methods

Most common Krylov subspace methods are characterised by

$$r_k = p_k(A) r_0 \quad , p_k \in \Pi_k \ , p_k(0) = 1$$

AND some optimality condition

e.g. $\|r_k\|_2$ should be minimal over $x_k \in x_0 + \mathcal{K}_k(A, r_0)$.

$$\mathcal{K}_k(A, r_0) = \text{ span } \{r_0, A r_0, \ldots, A^{k-1} r_0\}$$

First step is however to compute a good basis for $\mathcal{K}_k(A, r_0)$ since $A^k r_0$ tends to point in a single direction:

If $r_0 = \sum \alpha_i z_i, Az_i = \lambda_i z_i$ and if $|\lambda_1| > |\lambda_j|, j \neq 1$ then

$$\begin{aligned} A^k r_0 &= \sum_{i=1}^{n} \alpha_i A^k z_i = \sum_{i=1}^{n} \alpha_i \lambda_i^k z_i \\ &= \lambda_1^k \left[\alpha_1 z_1 + \sum_{i=2}^{n} \alpha_i \underbrace{\left(\frac{\lambda_i}{\lambda_1}\right)^k}_{\to 0 \text{ as } k \to \infty} z_i \right] \end{aligned}$$

Note also $\|A^k r_0\| \to 0$ or ∞ depending on whether $|\lambda_1| < 1$ or $|\lambda_1| > 1$

Arnoldi's method:

guess x_0 , $r_0 = b - Ax_0$, $v_1 = r_0/\|r_0\|_2$

for $l = 1, 2, \ldots$

$w = Av_l$

for $j = 1, \ldots, l$

$h_{jl} = v_j^T w$

$w = w - h_{jl} v_j$

end

$h_{l+1,l} = \|w\|_2$

$v_{l+1} = w/h_{l+1,l}$

end

is a way of generating an orthonormal basis $\{v_1, v_2, \ldots, v_k\}$ for $\mathcal{K}_k(A, r_0)$

In matrix form we can write Arnoldi as

$$\boldsymbol{AV_k = V_k H_k + h_{k+1,k}} \begin{bmatrix} | & | & & | \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{v_{k+1}} \\ | & | & & | \end{bmatrix} = \boldsymbol{V_{k+1}\hat{H}_k}$$

where

$$\boldsymbol{V_k} = \begin{bmatrix} | & | & & | \\ \boldsymbol{v_1} & \boldsymbol{v_2} & \cdots & \boldsymbol{v_k} \\ | & | & & | \end{bmatrix}, \quad \text{has orthogonal columns,}$$

$$\boldsymbol{H_k} = \begin{bmatrix} \boldsymbol{h_{11}} & \boldsymbol{h_{12}} & \cdots & \cdots & \boldsymbol{h_{1k}} \\ \boldsymbol{h_{21}} & \boldsymbol{h_{22}} & \cdots & \cdots & \boldsymbol{h_{2k}} \\ \boldsymbol{0} & \boldsymbol{h_{32}} & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \boldsymbol{0} & \cdots & \boldsymbol{0} & \boldsymbol{h_{k,k-1}} & \boldsymbol{h_{kk}} \end{bmatrix} \in \mathbb{R}^{\boldsymbol{k}\times\boldsymbol{k}}$$

is upper Hessenberg

and $\hat{H}_k = \begin{bmatrix} h_{11} & h_{12} & \cdots & \cdots & h_{1k} \\ h_{21} & h_{22} & \cdots & \cdots & h_{2k} \\ 0 & h_{32} & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & h_{k,k-1} & h_{kk} \\ 0 & \cdots & 0 & 0 & h_{k+1,k} \end{bmatrix} \in \mathbb{R}^{(k+1)\times k}$

$= \begin{bmatrix} & H_k & & & \\ 0 & \cdots & 0 & 0 & h_{k+1,k} \end{bmatrix} \in \mathbb{R}^{(k+1)\times k}$

Note: $V_k \in \mathbb{R}^{n\times k}$ has orthogonal columns
$\Rightarrow V_k^T V_k = I \in \mathbb{R}^{k\times k}$ so

$$AV_k = V_k H_k + h_{k+1,k} \begin{bmatrix} | & | & & | \\ 0 & 0 & \cdots & v_{k+1} \\ | & | & & | \end{bmatrix} \Rightarrow V_k^T A V_k = H_k$$

(Exercise: check)

Now

$$x_k \in x_0 + \mathcal{K}_k(A, r_0) \Leftrightarrow x_k = x_0 + V_k y$$

for some $y \in \mathbb{R}^k$ since $\{v_1 \dots v_k\}$ is an orthonormal basis for $\mathcal{K}_k(A, r_0)$. Also

$$\begin{aligned} x_k = x_0 + V_k y \quad \Leftrightarrow \qquad x - x_k &= x - x_0 - V_k y \\ A(x - x_k) &= A(x - x_0) - AV_k y \\ \text{i.e.} \qquad r_k &= r_0 - AV_k y \end{aligned}$$

so $\|r_k\|_2$ is minimal $\Leftrightarrow$ $y \in \mathbb{R}^k$ is such that $\|r_0 - AV_k y\|_2$ is minimal.

$\|r_k\|_2$ is minimal $\Leftrightarrow$ $y \in \mathbb{R}^k$ is such that $\|r_0 - AV_k y\|_2$ is minimal.

But $r_0 = \|r_0\| v_1 = \|r_0\| V_k e_1 = \|r_0\| V_{k+1} e_1$,
$e_1^T = [1, 0, \ldots, 0]$

and by above $AV_k = V_{k+1} \hat{H}_k$ so

$$\|r_k\|_2 \text{ is min } \Leftrightarrow \left\| V_{k+1}(\|r_0\| e_1 - \hat{H}_k y) \right\|_2 \text{ is min}$$

but $V_{k+1}^T V_{k+1} = I \in \mathbb{R}^{(k+1)\times(k+1)}$ as $\{v_1, \ldots, v_{k+1}\}$

are orthonormal

so required vector y is that which minimises the linear least squares problem

$$\left\| \|r_0\| e_1 - \hat{H}_k y \right\|_2$$

Required vector y is that which minimises the linear least squares problem

$$\left\| \|r_0\| e_1 - \hat{H}_k y \right\|_2$$

$\Rightarrow$ need QR factorisation of the rectangular Hessenberg matrix $\hat{H}_k \in \mathbb{R}^{k+1 \times k}$
which can be achieved by one additional Givens rotation for each k since $\hat{H}_k$ is built up by appending the last column for each k.

(see Exercises)

This is the basis of the GMRES algorithm
(Generalised Minimal Residual Method)

x_0 , $r_0 = b - Ax_0$, $v_1 = r_0/\|r_0\|$

For $k = 1, 2, \ldots$

do step k of the Arnoldi algorithm

($\Rightarrow$ have $v_1, v_2 \ldots \underbrace{v_{k+1}}_{\text{new}}$ and $\underbrace{\hat{H}_k}_{\text{last column new}}$)

solve the Hessenberg linear least squares problem

$y =$ arg min $\left\| \|r_0\| e_1 - \hat{H}_k y \right\|_2$

$x_k = x_0 + V_k y$

end

As before the Hessenberg least squares problem is solved by QR factorisation of $\hat{H}_k$ using $k+1$ Givens rotations, but since $\hat{H}_k$ is the same as $\hat{H}_{k-1}$ except for one additional row and column this can be implemented as only 1 Givens rotation for each k. (see exercises)
Notes:

1. $\|r_k\|_2$ is the linear least squares error in the Hessenberg least squares problem.
2. x_k only needs to be calculated if $\|r_k\|$ satisfies the stopping criterion $\|r_k\| \leq$ TOL.
3. work at k^{th} GMRES iteration is $O(k^2)$ for the least squares solution + 1 matrix vector product + vector operations. So for a sparse matrix with $O(1)$ entries per row work $\simeq O(k^2 n) \Rightarrow$ cheap method if only relatively few iterations (k) are needed.

Notes (continued);

4. GMRES gets expensive in storage of $v_1 \ldots v_k$ and the orthogonalisation computation if k gets too large, so sometimes is restarted: do a fixed number l of GMRES iterations then reset $r_0 \leftarrow r_l$ and repeat. This is GMRES (l) (which is not guaranteed to work!). Unrestarted GMRES is often called FULL GMRES.
5. If $A \in \mathbb{R}^{n \times n}$ then if GMRES does not stop before n steps, $\{v_1, \ldots, v_n\}$ is an orthonormal basis for $\mathbb{R}^n$ $\Rightarrow x_k = x$ because $\|r_k\|_2$ is minimimal for $x_k \in x_0 + \mathcal{K}_n(A, r_0)$ ie. for $x_k \in \mathbb{R}^n$.
6. Also if continues for n steps $AV_n = V_n H_n$, ie. $A = V_n H_n V_n^T$, $V_n, H_n \in \mathbb{R}^{n \times n}$. So A has been reduced by orthogonal similarity transform to Hessenberg form.

Convergence of GMRES

$$r_k = p_k(A)r_0 \text{ with } \|r_k\| \text{ minimal}$$

$\Rightarrow$ GMRES implicitly finds $p_k \in \Pi_k \ , \ p_k(0) = 1$ such that $\|p_k(A)r_0\|_2$ is minimal.

If A is diagonalisable

$$A = X\Lambda X^{-1} \Rightarrow p_k(A) = Xp_k(\Lambda)X^{-1}$$

$$A = X\Lambda X^{-1} \Rightarrow p_k(A) = X p_k(\Lambda) X^{-1}$$

implies

$$\|r_k\|_2 = \|p_k(A) r_0\|_2 = \|X p_k(\Lambda) X^{-1} r_0\|$$

$$\leq \min_{p \in \Pi_k, p(0)=1} \|X\|_2 \|X^{-1}\|_2 \, \|p(\Lambda)\|_2 \, \|r_0\|_2$$

or similar to before

$$\frac{\|r_k\|_2}{\|r_0\|_2} \leq \|X\|_2 \|X^{-1}\|_2 \min_{p \in \Pi_k, p(0)=1} \max_{\lambda_j} |p(\lambda_j)|$$

λ_j $\uparrow$ eigenvalues of A

—-$(\star)$

Comments:

1. $\|X\|_2\|X^{-1}\|_2 = \kappa(X)$ is a number independent of k: if it is large $\Rightarrow$ $(\star)$ is not very useful convergence estimate. If it is of moderate size (not known in practice!) then fast convergence if $\exists p \in \Pi_k \ , \ p(0) = 1$ such that $p(\lambda_j)$ small for all eigenvalues λ_j of A.
2. Other GMRES convergence bounds exist, but so far non is descriptive over a range of problems.
3. the expense of GMRES for k large has led to development of algorithms for non-symmetric matrices with fixed work per iteration, but these necessarily must loose any optimality property in general.

However: If $A = A^T$ then GMRES has fixed work per step: To see this we have

$$V_k^T A V_k = H_k$$

but $A = A^T$ so LHS is symmetric $\Rightarrow$ H_k is symmetric and Hessenberg $\Rightarrow$ tridiagonal

i.e. Arnoldi's algorithm would calculate lots of zeros in this case!

In fact the symmetric Lanczos algorithm is used: