(Arnoldi, 1951)

Implicitly Restarted Arnoldi method. (SORENSEN 1992) Implicit application of polynomial filters in a k-step Arnoldi method)

Let initial vector $q = c_1 v_1 + \cdots + c_k v_k$

where $v_1, v_2, \ldots, v_k$ are eigenvectors

and $k << n$.

Then Arnoldi iteration terminates in k steps or fewer.

and $\text{span}\{q_1, \ldots, q_k\} = \text{span}\{v_1, \ldots, v_k\}$

Thus ~~if~~ we would like to ~~we~~ get hold of a vector q that is linear combination of a few eigenvectors.

Let A be semi-simple.

and $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n|$

and $|\lambda_k| > |\lambda_{k+1}|$

~~Suppose~~ we would like to compute $\lambda_1, \ldots, \lambda_k$.

For arbitrary $q \in \mathbb{R}^n$

$$q = c_1 v_1 + \cdots + c_n v_n.$$

(components along all directions)

Our goal is to come up with a vector $\hat{q} = \hat{c}_1 v_1 + \cdots + \hat{c}_n v_n$

s.t. $\hat{c}_{k+1} \approx \hat{c}_{k+2} \approx \cdots \approx \hat{c}_n = 0$

Such a choice can be got hold off by taking $\hat{q} = p(A) q$

where $p(\cancel{A})$ is some polynomial.

$$\hat{q} = c_1 p(\lambda_1) v_1 + \cdots + c_n p(\lambda_n) v_n$$

If p is selected s.t. $p(\lambda_1) \cdots p(\lambda_k)$ are large compared to $p(\lambda_{k+1}) \cdots p(\lambda_n)$ then it is good.

IRA. Part I

We want K eigenvalues of A.

~~Do m iterations of~~

Perform m Arnoldi iterations.
where $m = K + j$, K and j are comparable.

Start with q. After m steps we get

$$A Q_m = Q_m H_m + q_{m+1} h_{m+1,m} e_m^T$$

Compute eigenvalues of H_m. (Ritz values of A) associated to colspan$\{q_1, \ldots, q_m\}$.
$\mu_1, \ldots, \mu_m$
in order $|\mu_1| \geq |\mu_2| \geq \ldots \geq |\mu_m|$.

$\mu_1, \ldots, \mu_K$ are estimates of K dominant eigenvalues of A.

whereas $\mu_{K+1}, \ldots, \mu_m$ approximate remain ones we are not interested in.

Part II.

Perform QR algorithm on H_m with shifts $\nu_1, \ldots, \nu_j$. Using $\nu_1 = \mu_{K+1}, \ldots, \nu_j = \mu_m$ as shifts.

QR with Shifts.

(i) $H_m - \nu_1 I_m = V_1 R_1$

$R_1 V_1 + \nu_1 I_m = {}^1H_m$

${}^1H_m - \nu_2 I_m = V_2 R_2$

$R_2 V_2 + \nu_2 I_m = {}^2H_m$

$\vdots$

${}^{j-1}H_m - \nu_j I_m = V_j R_j$

$R_j V_j + \nu_j I_m = {}^jH_m$

Repeat ~~until~~ convergence.

~~End~~. $V_m = V_1 V_2 \cdots V_j$

$R_m = R_1 R_2 \cdots R_j$

~~$\hat{H}_m = \hat{V}_m^T H_m V_m$~~ (Ex. 6.2.36)

Effectively. $\hat{H}_m = {}^jH_m = V_m^T H_m V_m$

$$(H_m - \nu_1 I)(H_m - \nu_2 I)\cdots(H_m - \nu_j I) = V_m R_m$$

$$p(H_m) = V_m R_m.$$

where $p(z) = (z-\nu_1)(z-\nu_2)\cdots(z-\nu_j)$

~~Update~~ Let $\hat{Q}_m = Q_m V_m$

$\hat{q}_1 = \hat{Q}_m(:,1)$

Next IRA iteration starts with

$\hat{q}_1$ for m steps. But we don't

need to ~~recompute~~ $\cancel{Q_m}$, $\cancel{H_m}$.

start Arnoldi iteration from scratch.

~~We can use~~ $\cancel{Q_m}$

$$A Q_m = Q_m H_m + q_{m+1} h_{m+1,m} e_m^T$$

$$A Q_m V_m = Q_m H_m V_m + q_{m+1} h_{m+1,m} e_m^T V_m$$

$$A \hat{Q}_m = \hat{Q}_m \hat{H}_m + q_{m+1} h_{m+1,m} e_m^T V_m.$$

(solve from Watkins 6.4.21) to show $e_m^T V_m$ has first $m-j-1$ as zeroes.

and $m-j$ th entry is non-zero.

$$\begin{array}{ccccccc} [\,0 & 0 & \cdots & 0 & \beta & \underbrace{* \cdots *}_{\text{last } j \text{ entries}}\,] \\ 1 & 2 & & m-j-1 & m-j & \end{array}$$

If we drop $*$ then we get

a vector of the form βe_K^T where $K = m-j$

Drop last j columns from

$$A \hat{Q}_m^p = \hat{Q}_m \hat{H}_m + q_{m+1} h_{m+1,m} e_m^T V_m.$$

We get

$$A \hat{Q}_K = \hat{Q}_K \hat{H}_{K+1,K} + \underbrace{\check{q}_{K+1} \check{h}_{K+1,K} e_K^T + q_{m+1} h_{m+1,m} \beta e_K^T}$$

Let $\hat{q}_{K+1} = \gamma \left(\check{q}_{K+1} \check{h}_{K+1,K} + q_{m+1} h_{m+1,m} \beta \right).$

γ is s.t $\|\hat{q}_{K+1}\|_2 = 1.$

Then.

$$A \hat{Q}_k = \hat{Q}_k \hat{H}_k + \hat{q}_{k+1} \hat{h}_{k+1,k} e_k^T$$

where $\hat{h}_{k+1,k} = \frac{1}{\gamma}$.

$\hat{Q}_k$ ~~is the~~ Columns are exactly those vectors which we would ~~[illegible]~~ get if we started Arnoldi method from $\hat{q}_1$.

Implicitly restarting.
perform remaining ~~[illegible]~~ j steps and build Q_m, H_m again.

For Symmetric A

Lanczos iteration.

Implicitly restarted Lanczos method.