Krylov Subspace Methods for the Eigenvalue problem

Presented by: Sanjeev Kumar

Applications

- We need only few eigen (singular) pairs, and matrices can be large and sparse
  - Solving homogeneous system of linear equations A x = 0. Solution is given by right singular vector of A corresponding to smallest singular value
  - Principal component analysis

    We are interested in eigen pairs corresponding to few largest eigenvalues
  - Discretization of Partial differential equation
  - Spectral image segmentation

References:

- Bai, Z., Demmel, J., Dongarra, J., Ruhe, A., and van der Vorst, H., editors. 2000. Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide, SIAM, Philadelphia, PA . http://www.cs.utk.edu/~dongarra/etemplates/book.html
- D.C. Sorensen, Implicitly Restarted Arnoldi/Lanczos Methods for Large Scale Eigenvalue Calculations http://techreports.larc.nasa.gov/icase/1996/icase-1996-40.pdf
- C.D. Meyer, Matrix Analysis and Applied Linear Algebra, SIAM http://www.matrixanalysis.com/DownloadChapters.html
- Marco Latini, The QR algorithm past and present http://www.acm.caltech.edu/~mlatini/research/presentation-**qr**-feb04.pdf
- J. Shi and J. Malik. "Normalized Cuts and Image Segmentation", IEEE PAMI, Vol. 22, No. 8, August 2000

Review: Eigenvalue and Eigenvector

- If $A x = \lambda x$
  - where,
    - $A \in \mathbb{R}^{n \times n}$
    - x : vector
    - λ : scalar
  - Then,
    - λ : eigenvalue
    - x : eigenvector
    - (λ, x) : eigen pair

Review: Eigenvalue decomposition (EVD)

- $A V = V D$
- $V = [x_1, x_2, \cdots, x_n]$
  - x_i's are eigenvectors
- $D = \text{diag}(\lambda_1, \lambda_2, \cdots, \lambda_n)$
  - λ_i's are eigenvalues
- In Matlab: [V, D] = eig(A)

Review: Characteristic Equation

- Eigenvalues are roots of the polynomial equation

  $$\det(A - \lambda \mathcal{I}) = 0$$

  - $\mathcal{I}$: n $\times$ n Identity matrix
  - det(.) : determinant of the matrix
  - Polynomial equation of degree n

Review: Companion Matrix

- Roots of a polynomial equation

  $x^n + \alpha_{n-1}x^{n-1} + \cdots + \alpha_1 x + \alpha_0 = 0$

  are given by eigenvalues of the matrix

$$\begin{pmatrix} 0 & 0 & \cdots & 0 & \alpha_0 \\ 1 & 0 & \cdots & 0 & \alpha_1 \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & \cdots & 1 & 0 & \alpha_{n-1} \\ 0 & 0 & \cdots & 1 & \alpha_{n-1} \end{pmatrix}_{n \times n}$$

Review: Abel-Rufini's Theorem

- Theorem
  - There are no algebraic formulae for roots of a general polynomial with degree greater than 4
- Consequence
  - As opposed to solving linear system of equations, iteration is the only way for eigenvalue computations for a general matrix

Review: Eigenvector Expansion

- (λ_i, x_i) are eigen pairs of matrix A
- Let us express any vector v as linear combination of eigenvectors,
  - $v = c_1 x_1 + \cdots + c_n x_n$
- Result of successive multiplication by A can be represented as,
  - $A v = \lambda_1 c_1 x_1 + \cdots + \lambda_n c_n x_n$
  - $(A^j) v = \lambda_1^j c_1 x_1 + \cdots + \lambda_n^j c_n x_n$
  - Useful later

Problem Statement

- Given a matrix $A \in \mathbb{R}^{n \times n}$, find k eigen pairs corresponding to eigenvalues with properties such as
  - Largest (or smallest) absolute value
  - Largest (or smallest) real part
  - Nearest to given scalar μ

Power Iteration

- Basic iteration step: $v_{j+1} = \frac{Av_j}{\|Av_j\|}$
- Analysis
  - (λ_i, x_i) : eigen pair of A, arranged in decreasing order of abs(λ_i)

$$\begin{aligned} v_0 &= \sum_{i=1}^{n} c_i x_i \\ \Rightarrow v_j &= k \sum_{i=1}^{n} \left(\frac{\lambda_i}{\lambda_1}\right)^j c_i x_i \\ \Rightarrow v_\infty &= x_1 \end{aligned}$$

Power Iteration, cont.

- Eigenvalue estimate : $(\widehat{\lambda_1})_j = \frac{v_j^T A v_j}{v_j^T v_j}$
- Convergence rate (eigenvector): $\left(\frac{\lambda_2}{\lambda_1}\right)$
- Disadvantages:
  - Very slow convergence if $\lambda_1 \approxeq \lambda_2$
  - Cannot find complex eigenvalues
  - Only finds largest eigenvalue

Spectral Transformation

- $A \in \mathbb{C}^{n \times n}$ has eigen pair (λ, x)
- p(τ) and q(τ) are polynomials in τ
- Polynomial transformation
  - p(A) has eigen pair (p(λ), x)
- Rational transformation
  - $[q(A)]^{-1}p(A)$ has eigen pair (p(λ)/q(λ), x)
- Shift-Invert
  - $(A-\mu \mathcal{I})^{-1}$ has eigen pair (1/(λ-μ), x)

Inverse Iteration

- Use a prior estimate of eigenvalue in shift-invert transformation

$$B = (A - \mu I)^{-1}$$
$$v_{j+1} = \frac{Bv_j}{\|Bv_j\|}$$

- Due to ill-conditioning, linear solver preferred to inverse
- Pre-factorize to keep per iteration complexity low

Rayleigh Quotient Iteration

- Use current estimate of eigenvalue as shift
- Advantages
  - Faster convergence: quadratic in general and cubic for hermitian problem
- Disadvantages
  - Per iteration complexity high

Subspace Iteration

- Used for finding multiple eigenvalues simultaneously.
- Generalization of power iteration to multiple vectors.
- Need better normalization than individually normalizing each vector otherwise every vector will converge to v_1

Subspace Iteration

- Start with $Q_0 \in \mathbb{C}^{n \times k}$ whose columns are orthonormal
- Iteration steps
  - $Z_j = A\, Q_{j-1}$
  - Orthonormalize $Z_j = X_j\, R_j$
    - Columns of X_j are orthonormal
    - R_j is upper triangular
  - $Q_j = X_j$
  - Test for convergence
- Convergence rate = $\left(\frac{\lambda_{k+1}}{\lambda_k}\right)$

Upper Hessenberg Matrix

- Upper Hessenberg Matrix
  - H(i,j) = 0 for i>(j+1)
  - Hermitian $\Rightarrow$ Tri-diagonal
- A $\rightarrow$ H
  - Householder reduction
  - Givens rotation
  - Both are $O(n^3)$ in general

$$\begin{pmatrix} * & * & * & * & * \\ & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \end{pmatrix}$$

Schur's Triangularization Theorem

- $\forall A \in \mathbb{C}^{n \times n}$, $\exists$ Q, R,

such that

  - Q is a unitary matrix (not unique)
  - R is an upper triangular matrix (not unique)
  - A Q = Q R
  - Diagonal elements of R are eigenvalues of A
- $A \rightarrow R \rightarrow \{ \lambda \}$
  - 2nd step is trivial but 1st step is very difficult
- $A \rightarrow H \rightarrow \{ \lambda \}$
  - In practice, we go from A to upper hessenberg H

QR Iteration

- Iteration steps:
  - $A_k = Q_k R_k$
  - $A_{k+1} = R_k Q_k$
- Every iteration is similarity and hence preserves eigenvalues
  - $A_{k+1} = Q_k^T A_k Q_k$
- If converges then converges to $A_\infty = R$
- Doesn't converge for matrices with complex or negative eigenvalues

Explicitly shifted QR iteration

- Convert A to upper hessenberg H using similarity transformation
- Iteration steps
  - $H_k - \alpha_k \mathcal{I} = Q_k R_k$
  - $H_k = R_k Q_k + \alpha_k \mathcal{I}$
- Efficiency considerations
  - Householder reduction is efficient for $A \rightarrow H$
  - QR factorization of H can be done efficiently using Givens rotation
- Related to inverse power iteration with shift α_k

Implicitly shifted QR iteration

- Combine two complex conjugate shifts
- Can handle complex eigenvalues and eigenvectors using real arithmetic, thus increasing efficiency

Definitions

- For $A \in \mathbb{C}^{n \times n}$ and $0 \neq b \in \mathbb{C}^{n \times 1}$,
  - $\{ b, Ab, A^2b, \cdots, A^{j-1}b \}$: Krylov sequence
  - $\mathcal{K}_j = \text{span}\{ b, Ab, \cdots, A^{j-1}b \}$: Krylov subspace
  - $K_{n \times j} = (b \mid Ab \mid \cdots \mid A^{j-1}b)$: Krylov matrix
    - $\mathcal{K}_i = \text{Range}(K_{n \times j})$

Krylov Subspace

- Let A have n distinct eigenvalues $\lambda_1, \cdots, \lambda_n$, with orthonormal eigenvectors $x_1, \cdots, x_n$ which form an orthonormal basis for $\mathbb{R}^n$.
- For any vector $b \in \mathbb{R}^n$, $b = c_1 x_1 + \cdots + c_n x_n$
- Let's analyze the structure of Krylov matrix, $K_{n \times j} = (b \mid Ab \mid \cdots \mid A^{j-1}b)$

Krylov Subspace

$$K_{n\times j} = [c_1 x_1 | \cdots | c_n x_n]_{n\times n} \begin{pmatrix} 1 & \lambda_1 & \cdots & \lambda_1^{j-1} \\ 1 & \lambda_2 & \cdots & \lambda_2^{j-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \lambda_n & \cdots & \lambda_n^{j-1} \end{pmatrix}_{n\times j}$$

- If $c_i \neq 0 \; \forall$ i,
  - Rank($K_{n \times j}$) = min(j, n)
- If number of non-zero c_i's is m,
  - Rank($K_{n \times j}$) = min(j, m)

Basis for Krylov Subspace

- Krylov sequence forms a basis for Krylov subspace but it is ill-conditioned.
- Better to work with an orthonormal basis.
- Lanczos algorithm builds an orthonormal basis for Krylov subspace for hermitian matrices.
- Arnoldi algorithm generalizes this to non-hermitian matrices.

Ritz value and Ritz vector

- Let $Q_k = [q_1, \cdots, q_k]$ be orthonormal basis for Krylov subspace $\mathcal{K}_k$.
- $H = Q^H A Q$ is projection of operator A onto $\mathcal{K}_k$
- Let (λ, y) be an eigen pair of H.
- λ is called Ritz value of A and provides an approximation for eigenvalue of A.
- $x = Q\, y$ is called Ritz vector of A and provides an approximation for eigenvector of A.

Lanczos Tridiagonalization Algorithm

$$A[q_1, \cdots, q_j, \cdots, q_n] = [q_1, \cdots, q_j, \cdots, q_n] \begin{pmatrix} \alpha_1 & \beta_1 & & & \\ \beta_1 & \alpha_2 & \beta_2 & & \\ & \beta_2 & \alpha_3 & \cdots & \\ & & \cdots & \cdots & \beta_{n-1} \\ & & & \beta_{n-1} & \alpha_n \end{pmatrix}$$

$$Aq_j = \beta_{j-1} q_{j-1} + \alpha_j q_j + \beta_j q_{j+1}$$

$$\beta_j q_{j+1} = v_j = Aq_j - \beta_{j-1} q_{j-1} - \alpha_j q_j$$

- Start with $q_0 = 0$ and $q_1 = v_1$ (start vector)
- Update α_j, β_j, q_{j+1}, for j=1,2, $\cdots$, n, until β is non-zero

Lanczos Tridiagonalization Algorithm

- After k iteration steps,

$$AQ_k = Q_k H_k + \beta_k q_{k+1} e_k^T$$

- $Q_k = [\, q_1, \cdots, q_k \,]$
- H_k = k $\times$ k tridiagonal
- Eigen pairs of H_k are Ritz pairs, and provide an approximation for eigen pairs of A
- If $\beta_k = 0$ for k<n, iteration terminates before n steps.

Lanczos Tridiagonalization Algorithm

- Advantages
  - Every iteration step has just one matrix-vector multiplication, hence linear complexity for sufficiently sparse matrices
- Disadvantages
  - In finite precision arithmetic q_i's may not be orthogonal. Cost of insuring orthogonality can be huge as number of iteration increases. This is the motivation behind all restarting schemes.

Arnoldi Iteration

- Like Lanczos but now we have an upper hessenberg H, instead of tridiagonal.

$$A[q_1, \cdots, q_j, \cdots, q_n] = [q_1, \cdots, q_j, \cdots, q_n] \begin{pmatrix} h_{1,1} & h_{1,2} & h_{1,3} & \cdots & h_{1,n} \\ h_{2,1} & h_{2,2} & h_{2,3} & \cdots & h_{2,n} \\ & h_{3,2} & h_{3,3} & \cdots & \vdots \\ & & \ddots & \ddots & \vdots \\ & & & h_{n-1,n} & h_{n,n} \end{pmatrix}$$

$$h_{j+1,j} q_{j+1} = A q_j - \sum_{i=1}^{j} h_{i,j} q_i$$

Arnoldi Iteration

- After k iteration steps, we get k-step Arnoldi factorization of A

$$AQ_k = Q_k H_k + \beta_k q_{k+1} e_k^T$$

- Ritz pairs of A can be computed from H_k and checked for convergence
- Maximum number of steps is selected as, m = k + p

Restarting the Arnoldi Method

- Need for Restarting
  - Quadratic increase in cost of maintaining orthogonality of basis vectors with increasing number of iterations
- Explicit Restart
  - Restart the whole process (m steps) with new initial vector
- Implicit Restart
  - Compress eigen information of interest and repeat last p steps.

Explicitly restarted Arnoldi Iteration

- Start with vector v_1
- Compute m=k+p step Arnoldi factorization
- Compute Ritz estimates for eigenvalues
- Stop if convergence has been achieved
- Separate eigenvalue estimates into groups of wanted and unwanted
- Compute better starting vector w_1 using obtained information
- Restart m step Arnoldi factorization using w_1

Explicitly Restarted Arnoldi Iteration

- How to compute better starting vector ?
  - Build a polynomial transformation $\psi(A)$ to damp unwanted components from eigenvector expansion of v_1 and set $w_1 = \psi(A)\, v_1$
  - Set w_1 = linear combination of Ritz vectors

Implicitly restarted Arnoldi Iteration

- Start with vector v_1
- Compute m=k+p step Arnoldi factorization
- Compute Ritz-estimates for eigenvalues
- Stop if convergence has been achieved
- Select unwanted eigenvalues
- Using unwanted eigenvalues as shifts, apply p steps of implicitly shifted QR iteration to compress eigen information of interest into k dimensional subspace
- Repeat last p steps of Arnoldi factorization

Implicitly Restarted Arnoldi Iteration

- Advantages
  - Uses O(nk) space for computing k eigen pairs
  - For small k and sufficiently sparse matrix, each iteration costs only O(n) time.
  - Number of iterations for convergence is only slightly larger than Arnoldi iteration without restart

Singular Value Decomposition (SVD)

- $A = U S V^T$
- $A \in \mathbb{R}^{m \times n}$
  - As opposed to eigenvalue decomposition, SVD is defined for rectangular matrices as well
- $U = [u_1, \cdots, u_m]$
  - u_i 's are left singular vectors
- $V = [v_1, \cdots, v_n]$
  - v_i 's are right singular vectors
- $S \in \mathbb{R}^{m \times n}$ (same size as A)
  - Singular values on diagonal, rest zero
- In Matlab: [U, S, V] = svd(A)

Singular Value Decomposition (SVD)

- Singular values of A $\leftrightarrow$ eigenvalues of A^TA or AA^T
- Right singular vectors of A $\leftrightarrow$ eigenvectors of A^TA
- Left singular vectors $\leftrightarrow$ eigenvectors of AA^T
- A^TA and AA^T are not computed explicitly. $A^TAx \equiv A^T(Ax)$

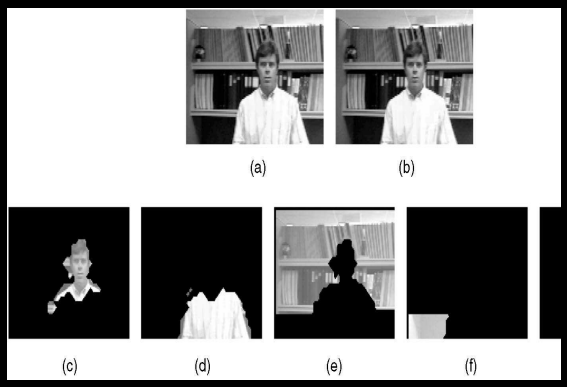
Some experiments in Matlab

- eigs (svds):
  - finds k eigen (singular) values with specified property and corresponding eigen (singular) vectors. Uses implicitly restarted Arnoldi iteration via ARPACK routines.
- Demo:
  - grouping of points using normalized cut

Normalized Cuts and Image Segmentation

- Affinity matrix

$$w_{ij} = e^{\frac{-\|F(i)-F(j)\|_2^2}{\sigma_I^2}} * \begin{cases} e^{\frac{-\|X(i)-X(j)\|_2^2}{\sigma_X^2}} & \text{if } \|X(i)-X(j)\|_2 < r \\ 0 & \text{otherwise.} \end{cases}$$



- Eigenvalue problem

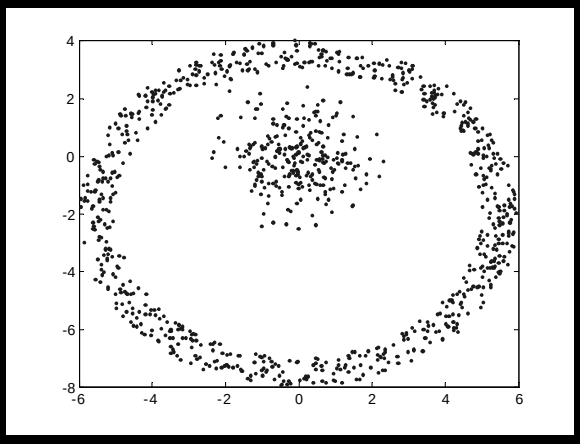
$$\mathbf{D}^{-\frac{1}{2}}(\mathbf{D}-\mathbf{W})\mathbf{D}^{-\frac{1}{2}}\boldsymbol{x} = \lambda \boldsymbol{x}.$$

Where, D = Diag(Sum(W))

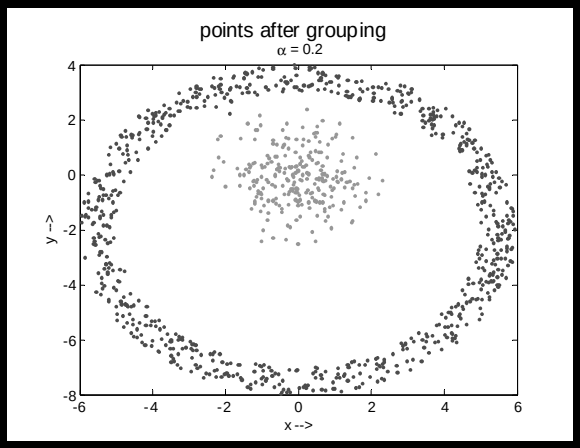
- Second smallest eigenvector used for segmentation

Matlab Demo

Before grouping



After grouping



For 1000 points, eigs : eig = 1.8 s : 22.4 s

Conclusion

- Krylov subspace methods are very suitable for finding few eigen (singular) pairs of interest.
- By using the matrix only in the form of matrix-vector product, they allow for very efficient use of special structures present in the matrix e.g. sparseness.
- Implicitly Restarted Arnoldi Iteration is the most time and space efficient method for computing few eigen pairs for large sparse matrices.

Extra Slides

- Krylov
- Lanczos
- Arnoldi
- Linear System of equations and Krylov subspace
- Conditioning of Matrices

Krylov

- Aleksei Nikolaevich Krylov (1863–1945) showed in 1931 how to use sequences of the form $\{\mathbf{b}, \mathbf{Ab}, \mathbf{A}^2\mathbf{b}, \ldots\}$ to construct the characteristic polynomial of a matrix. Krylov was a Russian applied mathematician whose scientific interests arose from his early training in naval science that involved the theories of buoyancy, stability, rolling and pitching, vibrations, and compass theories. Krylov served as the director of the Physics–Mathematics Institute of the Soviet Academy of Sciences from 1927 until 1932, and in 1943 he was awarded a "state prize" for his work on compass theory. Krylov was made a "hero of socialist labor," and he is one of a few mathematicians to have a lunar feature named in his honor—on the moon there is the "Crater Krylov."

Lanczos

- Cornelius Lanczos (1893–1974) was born Kornel Lowy in Budapest, Hungary, to Jewish parents, but he changed his name to avoid trouble during the dangerous times preceding World War II. After receiving his doctorate from the University of Budapest in 1921, Lanczos moved to Germany where he became Einstein's assistant in Berlin in 1928. After coming home to Germany from a visit to Purdue University in Lafayette, Indiana, in 1931, Lanczos decided that the political climate in Germany was unacceptable, and he returned to Purdue in 1932 to continue his work in mathematical physics. The development of electronic computers stimulated Lanczos's interest in numerical analysis, and this led to positions at the Boeing Company in Seattle and at the Institute for Numerical Analysis of the National Bureau of Standards in Los Angeles. When senator Joseph R. McCarthy led a crusade against communism in the 1950s, Lanczos again felt threatened, so he left the United States to accept an offer from the famous Nobel physicist Erwin Schr¨odinger (1887–1961) to head the Theoretical Physics Department at the Dublin Institute for Advanced Study in Ireland where Lanczos returned to his first love-the theory of relativity.

Arnoldi

- Walter Edwin Arnoldi (1917–1995) was an American engineer who published this technique in 1951, not far from the time that Lanczos's algorithm emerged. Arnoldi received his undergraduate degree in mechanical engineering from Stevens Institute of Technology, Hoboken, New Jersey, in 1937 and his MS degree at Harvard University in 1939. He spent his career working as an engineer in the Hamilton Standard Division of the United Aircraft Corporation where he eventually became the division's chief researcher. He retired in 1977. While his research concerned mechanical and aerodynamic properties of aircraft and aerospace structures, Arnoldi's name is kept alive by his orthogonalization procedure.

Linear system of equations and Krylov subspace

- Solution of $Ax = b$ lies in the Krylov subspace

  span { b, Ab, ⋯ }
- Generalized Minimum RESidual (GMRES) method iteratively builds Krylov subspace of increasing dimension till sufficiently accurate solution has been obtained.

Conditioning of Matrices

- Notion of conditioning depends on problem.
  - For linear systems we ideally want equal eigenvalues
  - For eigenvalue problem we want large separation between wanted and unwanted eigenvalues
  - If several eigenvalues are almost equal, finding individual eigenvectors is ill conditioned. Better to find invariant subspace associated with that cluster of eigenvalues.