Semi-definite and Second Order Cone Programming Seminar
Fall 2012
Final Project - Part I. (Eigenvalue Decomposition)

Instructor: Farid Alizadeh
Scribe: Wang Yao, Gyorgy Matyasfalvi

12/03/2012

1 Eigen Value Decomposition

1.1 The Eigenvalue Problem

The difficulty of the eigenvalue problem arises form the consideration that any polynomial root-finding problem can be stated as an eigenvalue problem. We know by Abel that no formula exists for expressing the roots of an arbitrary polynomial. From this we can deduce that any algorithm for finding the eigenvalues has to be iterative. Therefore the goal of an eigenvalue solver should be to produce sequences of numbers that converge rapidly toward eigenvalues [2]. Before we start discussing the specifics of an eigenvalue algorithm let's mention a couple techniques and concepts that we will use in devising an algorithm.

1.2 Reduced QR factorization [2]

The column spaces of matrix $A \in \mathbb{R}^{m \times m}$ are

$$\langle \mathbf{a}_1 \rangle \subseteq \langle \mathbf{a}_1, \mathbf{a}_2 \rangle \subseteq \langle \mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3 \rangle \subseteq \dots$$

The idea of QR factorization is the construction of a sequence of orthonormal vectors $\mathbf{q}_1, \mathbf{q}_2, \dots$ that successively span the column spaces of A such that:

$$A = QR$$

In the following form:

$$\left(\begin{array}{c|c|c|c} & & & \\ a_1 & a_2 & \dots & a_n \\ & & & \end{array}\right) = \left(\begin{array}{c|c|c|c} & & & \\ q_1 & q_2 & \dots & q_n \\ & & & \end{array}\right) \begin{pmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ & r_{22} & & \\ & & \ddots & \vdots \\ & & & r_{nn} \end{pmatrix}$$

Given $a_1, a_2, \dots$ we can construct vectors $q_1, q_2, \dots$ and entries r_{ij} by the process known as *Gram-Schmidt orthogonalization.*

1.3 Power iteration simultaneous iteration

The concept of *power iteration* tells us that the sequence:

$$\frac{x}{\|x\|}, \frac{Ax}{\|Ax\|}, \frac{A^2x}{\|A^2x\|}, \frac{A^3x}{\|A^3x\|}, \dots \tag{1}$$

converges under certain assumptions to an eigenvector corresponding to the largest eigenvalue of A in absolute value.

1.4 QR Factorization of a Krylov Matrix - Lanczos procedure

Let us denote by $\mathcal{K}_n$ the so called *Krylov subspaces* generated by A and b:

$$\mathcal{K}_n = \langle b, Ab, \dots, A^{n-1}b \rangle \tag{2}$$

Let's denote by $K_n \in \mathbb{R}^{m \times n}$ the so called *Krylov matrix*:

$$K_n = \left(\begin{array}{c|c|c|c} & & & \\ b & Ab & \dots & A^{n-1}b \\ & & & \end{array}\right) \tag{3}$$

It can be shown that if A is diagonalizable:

$$\dim \mathcal{K}_n = \min\{m, n\} \tag{4}$$

All symmetric matrices are normal and since diagonalizable. The Lanczos procedure finds the QR factorization of K_n.

Data: $A \in \mathbb{R}^{m\times m}$ Symmetric
Result: $Q_k = (q_1, \ldots, q_k)$ Orthonormal and $T_k \in \mathbb{R}^{k\times k}$ Tridiagonal
Initialization: $r_0 = q_1$; $\beta_0 = 1$; $q_0 = 0$; $l = 0$;
while $\beta_l \neq 0$ **do**
  $q_{l+1} = \frac{r_l}{\beta_l}$; $l = l + 1$; $\alpha_l = q_l^T A q_l$;
  $r_l = A q_l - \alpha_l q_l - \beta_{l-1} q_{l-1}$;
  $\beta_l = \|r_l\|_2$;
end

Algorithm 1: Lanczos Procedure [1]

The algorithm stops if $\beta_l = 0$ i.e. if it has found an invariant subspace. The vectors $\{q_1, \ldots, q_k\}$ form an invariant subspace of A. The Ritz values and Ritz vector of A in this space are obtained by means of the $k \times k$ eigenvalue problem:

$$T_k y = Q_k^T A Q_k y = \vartheta_k y \tag{5}$$

Claim 1 *If* (ϑ_k, y) *is an eigenpair of* T_k *then* $(\vartheta_k, Q_k y)$ *is an eigenpair of* A.

Proof: We have that (ϑ_k, y) is the solution to:

$$T_k y = Q_k^T A Q_k y = \vartheta_k y \tag{6}$$

Since $\{q_1, \ldots, q_k\}$ form an invariant subspace of A we can say that:

$$\begin{aligned}
A Q_k y &= Q_k x \\
Q_k^T A Q_k y &= Q_k^T Q_k x \\
Q_k^T A Q_k y &= I x \\
Q_k^T A Q_k y &= x \\
&\Downarrow \\
x &= \vartheta_k y \\
&\Downarrow \\
A Q_k y &= \vartheta_k Q_k y
\end{aligned}$$

■

Pros:

- The cost of a single iteration step does not depend on the index of the iteration (matrix-vector multiplication and $7m$ flops).
- The eigenvalues of T_k can be computed efficiently e.g. the tridiagonal QR algorithm in $O(k^2)$ flops. The cost of computing the eigenvalues of T_k are in general negligible compared with the cost of forming $A q_l$.

Cons:

- Storing $q_1, \ldots, q_2$ is expensive.
- Quick termination resulting in a "small" invariant subspace i.e. our subspace is far from optimal.

Turns out that the extremal eigenvalues and corresponding eigenvectors are well approximated after a few iterations [3].

$$|\vartheta_j - \lambda| \leq \epsilon \tag{7}$$

where ϵ is tiny for $j << k$, an we can get an estimation of the error from the following Lemma [3]:

Lemma 1 *Let $A \in \mathbb{F}^{m\times m}$ be Hermitian. Let $\vartheta \in \mathbb{R}$ and $x \in \mathbb{F}^n$ with $x \neq 0$ be arbitrary. Set $\tau := \frac{\|(A-\vartheta I)x\|}{\|x\|}$. Then there is an eigenvalue of A in the interval $[\vartheta - \tau, \vartheta + \tau]$.*

According to Lemma 1. there exists a λ of A such that:

$$|\lambda - \vartheta_j| \leq \beta_j |s_{ji}| \tag{8}$$

where s_{ji} is the last element of the eigenvector matrix S_j of T_j.

1.5 Improvements [3]

However due to floating point arithmetic loss of orthogonality among Ritz vectors is introduced and as a result one could end up computing multiple copies of certain eigenvalues. To overcome this issue partial re-orthogonalization is introduced in the algorithm. A further issue with the Lanczos algorithm is that it cannot detect the multiplicity of the eigenvalues it computes. The Block Lanczos algorithm is capable of determining multiplicities of eigenvalues up to the block size.

1.6 Block Lanczos

The idea of the Block Lanczos procedure is that instead of a single vector q_1 we shall build our "'big" Krylov subspace from a matrix $X_1 \in \mathbb{R}^{m\times d}$ of mutually orthogonal vectors the following way:

$$\mathcal{K}_{kd}(X_1) = span\{X_1, AX_1, \ldots, A^{k-1}X_1\} \tag{9}$$

The Block Lanczos procedure will deliver $\tilde{Q}_k = [X_1\ X_2\ \ldots\ X_k]$ consisting of a basis of $\mathcal{K}_{kd}$ which will lead to the following projection:

$$\tilde{T}_k = \tilde{Q}_k^T A \tilde{Q}_k \tag{10}$$

where $\tilde{T}_k$ is a block tridiagonal matrix of the following form:

$$\tilde{T}_k = \begin{pmatrix} M_1 & B_1^T & \dots & 0 \\ B_1 & M_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & B_{k-1}^T \\ 0 & \dots & B_{k-1} & M_k \end{pmatrix} \tag{11}$$

If B_is are upper triangular $\tilde{T}_k$ is a band matrix with bandwidth $2d+1$ [**?**].

Data: $A \in \mathbb{R}^{m\times m}$ Symmetric, $X_1 \in \mathbb{R}^{m\times d}$ Orthogonal, k
Result: $\tilde{Q}_k = (X_1, \dots, X_k)$ and $\tilde{T}_k$ Block-Tridiagonal
Initialization: $M_1 = X_1^T A X_1$; $B_0 = 0$;
for $l = 1 : k-1$ **do**
  $R_l = AX_l - X_l M_l - X_{l-1} B_{l-1}^T$;
  $(X_{l+1}, B_l) = qr(R_l)$ (The QR decomposition) ;
  $M_{l+1} = X_{l+1}^T A X_{l+1}$;
end

Algorithm 2: Block Lanczos Procedure [1]

Once we obtain our $\tilde{T}_k$ we solve the eigenvalue problem:

$$\tilde{T}_k \mathbf{y} = \vartheta \mathbf{y} \tag{12}$$

To get the Ritz pair $(\vartheta, \tilde{Q}_k \mathbf{y})$

Data: $A \in \mathbb{R}^{m\times m}$ Symmetric, $X_1 \in \mathbb{R}^{m\times d}$ Orthogonal, k
Result: $X = \tilde{Q}_k U_k(:, 1:d)$, $\sum = \Lambda(1:d, 1:d)$
Initialization: $M_1 = X_1^T A X_1$; $B_0 = 0$;
1. Compute $\tilde{Q}_k$ and $\tilde{T}_k$ by Algorithm 2. ;
2. Compute EVD of $\tilde{T}_k : \tilde{T}_k = U_k \Lambda_k U_k^T$, where the eigenvalues on the diagonal of Λ are in a decreasing order. ;

Algorithm 3: Block Lanczos for EVD [1]

1.7 Conclusion

In [1] they claim that extensive experimental results indicate that the BLWS makes its host algorithm at least twice faster.

References

[1] Lin, Z., Wei, S., *Accelerating Iterations Involving Eigenvalue or Singular Value Decomposition by Block Lanczos with Warm Start*, Microsoft Technical Report: MSR-TR-2010-162, 2010.

[2] Bau, D. III, Trefethen, N. L., *Numerical Linear Algebra* p. 194, SIAM, 1997, Philadelphia.

[3] Arbenz, P., Kressner, D., *Lecture Notes on Solving Large Scale Eigenvalue Problems*, http://people.inf.ethz.ch/arbenz/ewp/Lnotes/lsevp2010.pdf , 2012.