Tracking dominant symmetric matrices (including a new matrix factorization)

Paul Van Dooren, U.C.Louvain, Belgium
Nicola Mastronardi, CNR Bari, Italy

Large Graphs and Networks
UCL Université catholique de Louvain

Eurasip lecture 2, Bari, August 2012

Goal

We want to approximate an $N \times N$ indefinite symmetric matrix H by a rank r factorization

$$H \approx UMU^T$$

where $U^T U = I_r$ and M is allowed to be indefinite as well

The optimal approximation is known

- $\Lambda(M)$ contains the r largest eigenvalues of H (in modulus)
- $\mathrm{Im}(U)$ is a basis for the corresponding eigenspace

Goal

We want to approximate an $N \times N$ indefinite symmetric matrix H by a rank r factorization

$$H \approx UMU^T$$

where $U^T U = I_r$ and M is allowed to be indefinite as well

The optimal approximation is known

- $\Lambda(M)$ contains the r largest eigenvalues of H (in modulus)
- $\mathrm{Im}(U)$ is a basis for the corresponding eigenspace

The complexity of this problem is $\mathcal{O}(N^3)$ for a dense matrix H

Goal

We want to approximate an $N \times N$ indefinite symmetric matrix H by a rank r factorization

$$H \approx UMU^T$$

where $U^T U = I_r$ and M is allowed to be indefinite as well

The optimal approximation is known

- $\Lambda(M)$ contains the r largest eigenvalues of H (in modulus)
- $\mathrm{Im}(U)$ is a basis for the corresponding eigenspace

The complexity of this problem is $\mathcal{O}(N^3)$ for a dense matrix H

Goal :
Iterative procedure for large dense matrices in $\mathcal{O}(Nnr)$ complexity

Goal

We want to approximate an $N \times N$ indefinite symmetric matrix H by a rank r factorization

$$H \approx UMU^T$$

where $U^T U = I_r$ and M is allowed to be indefinite as well

The optimal approximation is known

- $\Lambda(M)$ contains the r largest eigenvalues of H (in modulus)
- $\mathrm{Im}(U)$ is a basis for the corresponding eigenspace

The complexity of this problem is $\mathcal{O}(N^3)$ for a dense matrix H

Goal :
Iterative procedure for large dense matrices in $\mathcal{O}(Nnr)$ complexity
($r \ll N$ is the rank of M and $n \leq N$ is the window size of the method)

"Sweeping" through H

We show only the columns and rows of U and H that are involved

$$\begin{bmatrix} \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \end{bmatrix} \quad \begin{bmatrix} \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \end{bmatrix}$$

Rank $r = 3$ and we will use window size $n = 6$

step 1

$$\begin{bmatrix} \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ & & \\ & & \\ & & \\ & & \end{bmatrix} \quad \begin{bmatrix} \times & \times & \times & \times & \times & \times & & & & \\ \times & \times & \times & \times & \times & \times & & & & \\ \times & \times & \times & \times & \times & \times & & & & \\ \times & \times & \times & \times & \times & \times & & & & \\ \times & \times & \times & \times & \times & \times & & & & \\ \times & \times & \times & \times & \times & \times & & & & \\ & & & & & & & & & \\ & & & & & & & & & \\ & & & & & & & & & \\ & & & & & & & & & \end{bmatrix}$$

Start with leading $n \times n$ subproblem ($n = 6$ is the window size)

step 2

$$
\begin{bmatrix}
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
 & & \\
 & & \\
 & & \\
 & &
\end{bmatrix}
\begin{bmatrix}
\times & \times & \times & \times & \times & \times & \color{blue}{\times} & & & \\
\times & \times & \times & \times & \times & \times & \color{blue}{\times} & & & \\
\times & \times & \times & \times & \times & \times & \color{blue}{\times} & & & \\
\times & \times & \times & \times & \times & \times & \color{blue}{\times} & & & \\
\times & \times & \times & \times & \times & \times & \color{blue}{\times} & & & \\
\times & \times & \times & \times & \times & \times & \color{blue}{\times} & & & \\
\color{blue}{\times} & \color{blue}{\times} & \color{blue}{\times} & \color{blue}{\times} & \color{blue}{\times} & \color{blue}{\times} & \color{blue}{\times} & & & \\
 & & & & & & & & & \\
 & & & & & & & & & \\
 & & & & & & & & &
\end{bmatrix}
$$

Expand

step 2

$$
\begin{bmatrix}
\times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times \\
 & & & & \\
 & & & & \\
 & & & &
\end{bmatrix}
\begin{bmatrix}
\times & \times & \times & \times & \times & \times & \times & & & \\
\times & \times & \times & \times & \times & \times & \times & & & \\
\times & \times & \times & \times & \times & \times & \times & & & \\
\times & \times & \times & \times & \times & \times & \times & & & \\
\times & \times & \times & \times & \times & \times & \times & & & \\
\times & \times & \times & \times & \times & \times & \times & & & \\
\times & \times & \times & \times & \times & \times & \times & & & \\
 & & & & & & & & & \\
 & & & & & & & & & \\
 & & & & & & & & &
\end{bmatrix}
$$

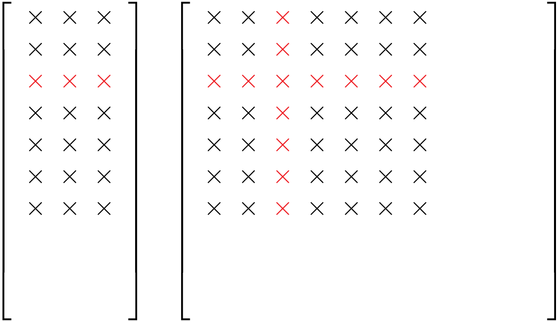
Expand also U (rank increases by 2)

step 3

$$\begin{bmatrix} \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ & & \\ & & \end{bmatrix} \begin{bmatrix} \times & \times & \times & \times & \times & \times & \times & & & \\ \times & \times & \times & \times & \times & \times & \times & & & \\ \times & \times & \times & \times & \times & \times & \times & & & \\ \times & \times & \times & \times & \times & \times & \times & & & \\ \times & \times & \times & \times & \times & \times & \times & & & \\ \times & \times & \times & \times & \times & \times & \times & & & \\ \times & \times & \times & \times & \times & \times & \times & & & \\ & & & & & & & & & \\ & & & & & & & & & \end{bmatrix}$$

Downdate rank ...

step 4



... and also downsize window

step 4

$$
\begin{bmatrix}
\times & \times & \times \\
\times & \times & \times \\
 & & \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
 & & \\
 & & \\
 & &
\end{bmatrix}
\begin{bmatrix}
\times & \times & & \times & \times & \times & \times & & & \\
\times & \times & & \times & \times & \times & \times & & & \\
 & & & & & & & & & \\
\times & \times & & \times & \times & \times & \times & & & \\
\times & \times & & \times & \times & \times & \times & & & \\
\times & \times & & \times & \times & \times & \times & & & \\
\times & \times & & \times & \times & \times & \times & & & \\
 & & & & & & & & & \\
 & & & & & & & & & \\
 & & & & & & & & &
\end{bmatrix}
$$

step 5

$$
\begin{bmatrix}
\times & \times & \times \\
\times & \times & \times \\
 & & \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
 & & \\
 & & \\
 & &
\end{bmatrix}
\begin{bmatrix}
\times & \times & & \times & \times & \times & \times & \times & & \\
\times & \times & & \times & \times & \times & \times & \times & & \\
 & & & & & & & & & \\
\times & \times & & \times & \times & \times & \times & \times & & \\
\times & \times & & \times & \times & \times & \times & \times & & \\
\times & \times & & \times & \times & \times & \times & \times & & \\
\times & \times & & \times & \times & \times & \times & \times & & \\
\times & \times & & \times & \times & \times & \times & \times & & \\
 & & & & & & & & & \\
 & & & & & & & & &
\end{bmatrix}
$$

Expand

step 6

$$
\begin{bmatrix}
\times & \times & \times \\
\times & \times & \times \\
 & & \\
\times & \times & \times \\
\color{red}\times & \color{red}\times & \color{red}\times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
 & & \\
 & &
\end{bmatrix}
\begin{bmatrix}
\times & \times & & \times & \color{red}\times & \times & \times & \times & & \\
\times & \times & & \times & \color{red}\times & \times & \times & \times & & \\
 & & & & & & & & & \\
\times & \times & & \times & \color{red}\times & \times & \times & \times & & \\
\color{red}\times & \color{red}\times & & \color{red}\times & \color{red}\times & \color{red}\times & \color{red}\times & \color{red}\times & & \\
\times & \times & & \times & \color{red}\times & \times & \times & \times & & \\
\times & \times & & \times & \color{red}\times & \times & \times & \times & & \\
\times & \times & & \times & \color{red}\times & \times & \times & \times & & \\
 & & & & & & & & & \\
 & & & & & & & & &
\end{bmatrix}
$$

Downdate and downsize

step 7

$$
\begin{bmatrix}
\times & \times & \times \\
\times & \times & \times \\
 & & \\
\times & \times & \times \\
 & & \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
 & & \\
 & &
\end{bmatrix}
\begin{bmatrix}
\times & \times & & \times & & \times & \times & \times & & \\
\times & \times & & \times & & \times & \times & \times & & \\
 & & & & & & & & & \\
\times & \times & & \times & & \times & \times & \times & & \\
 & & & & & & & & & \\
\times & \times & & \times & & \times & \times & \times & & \\
\times & \times & & \times & & \times & \times & \times & & \\
\times & \times & & \times & & \times & \times & \times & & \\
 & & & & & & & & & \\
 & & & & & & & & &
\end{bmatrix}
$$

step 8

$$
\begin{bmatrix}
\times & \times & \times \\
\times & \times & \times \\
 & & \\
\times & \times & \times \\
 & & \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
 & & \\
 & &
\end{bmatrix}
\begin{bmatrix}
\times & \times & & \times & & \times & \times & \times & \times & \\
\times & \times & & \times & & \times & \times & \times & \times & \\
 & & & & & & & & & \\
\times & \times & & \times & & \times & \times & \times & \times & \\
 & & & & & & & & & \\
\times & \times & & \times & & \times & \times & \times & \times & \\
\times & \times & & \times & & \times & \times & \times & \times & \\
\times & \times & & \times & & \times & \times & \times & \times & \\
\times & \times & & \times & & \times & \times & \times & \times & \\
 & & & & & & & & &
\end{bmatrix}
$$

Expand

step 9

$$
\begin{bmatrix}
\color{red}\times & \color{red}\times & \color{red}\times \\
\times & \times & \times \\
 & & \\
\times & \times & \times \\
 & & \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
 & &
\end{bmatrix}
\begin{bmatrix}
\color{red}\times & \color{red}\times & & \color{red}\times & & \color{red}\times & \color{red}\times & \color{red}\times & \color{red}\times & \\
\color{red}\times & \times & & \times & & \times & \times & \times & \times & \\
 & & & & & & & & & \\
\color{red}\times & \times & & \times & & \times & \times & \times & \times & \\
 & & & & & & & & & \\
\color{red}\times & \times & & \times & & \times & \times & \times & \times & \\
\color{red}\times & \times & & \times & & \times & \times & \times & \times & \\
\color{red}\times & \times & & \times & & \times & \times & \times & \times & \\
\color{red}\times & \times & & \times & & \times & \times & \times & \times & \\
 & & & & & & & & &
\end{bmatrix}
$$

Downdate and downsize

step 10

$$\begin{bmatrix} & & \\ \times & \times & \times \\ & & \\ \times & \times & \times \\ & & \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ & & \end{bmatrix} \begin{bmatrix} & & & & & & & & & \\ & \times & & \times & & \times & \times & \times & \times & \\ & & & & & & & & & \\ & \times & & \times & & \times & \times & \times & \times & \\ & & & & & & & & & \\ & \times & & \times & & \times & \times & \times & \times & \\ & \times & & \times & & \times & \times & \times & \times & \\ & \times & & \times & & \times & \times & \times & \times & \\ & \times & & \times & & \times & \times & \times & \times & \\ & & & & & & & & & \end{bmatrix}$$

step 11

$$
\begin{bmatrix}
\times & \times & \times \\
 & & \\
\times & \times & \times \\
 & & \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times
\end{bmatrix}
\begin{bmatrix}
\times & & \times & & \times & \times & \times & \times & \times \\
 & & & & & & & & \\
\times & & \times & & \times & \times & \times & \times & \times \\
 & & & & & & & & \\
\times & & \times & & \times & \times & \times & \times & \times \\
\times & & \times & & \times & \times & \times & \times & \times \\
\times & & \times & & \times & \times & \times & \times & \times \\
\times & & \times & & \times & \times & \times & \times & \times \\
\times & & \times & & \times & \times & \times & \times & \times
\end{bmatrix}
$$

Expand

step 12

$$
\begin{bmatrix}
 & & \\
\times & \times & \times \\
 & & \\
\times & \times & \times \\
 & & \\
{\color{red}\times} & {\color{red}\times} & {\color{red}\times} \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times
\end{bmatrix}
\begin{bmatrix}
 & & & & & & & & & \\
 & \times & & \times & & {\color{red}\times} & \times & \times & \times & \times \\
 & & & & & & & & & \\
 & \times & & \times & & {\color{red}\times} & \times & \times & \times & \times \\
 & & & & & & & & & \\
 & {\color{red}\times} & & {\color{red}\times} & & {\color{red}\times} & {\color{red}\times} & {\color{red}\times} & {\color{red}\times} & {\color{red}\times} \\
 & \times & & \times & & {\color{red}\times} & \times & \times & \times & \times \\
 & \times & & \times & & {\color{red}\times} & \times & \times & \times & \times \\
 & \times & & \times & & {\color{red}\times} & \times & \times & \times & \times \\
 & \times & & \times & & {\color{red}\times} & \times & \times & \times & \times
\end{bmatrix}
$$

Downdate and downsize

step 13

$$
\begin{bmatrix}
 & & \\
\times & \times & \times \\
 & & \\
\times & \times & \times \\
 & & \\
 & & \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times
\end{bmatrix}
\begin{bmatrix}
 & & & & & & & & & \\
 & \times & & \times & & & \times & \times & \times & \times \\
 & & & & & & & & & \\
 & \times & & \times & & & \times & \times & \times & \times \\
 & & & & & & & & & \\
 & & & & & & & & & \\
 & \times & & \times & & & \times & \times & \times & \times \\
 & \times & & \times & & & \times & \times & \times & \times \\
 & \times & & \times & & & \times & \times & \times & \times \\
 & \times & & \times & & & \times & \times & \times & \times
\end{bmatrix}
$$

Aim

Applications

- Tracking indefinite matrix problems
- Updating in sequential quadratic programming
- Updating saddle point problems

Aim

Applications

- Tracking indefinite matrix problems
- Updating in sequential quadratic programming
- Updating saddle point problems

Tools

- Updating increases $r \to r + 2$ and $n \to n + 1$
- Downdating restores the rank r (reduce by 2 again)
- Downsizing restores the window size n (optional)

The competitors

The following factorizations are frequently used :

The competitors

The following factorizations are frequently used :

- $M = VT_{ri}V^T$ (tridiagonalization)

The competitors

The following factorizations are frequently used :

- $M = VT_{ri}V^T$ (tridiagonalization)
- $M = LD_{bl}L^T$ (block diagonal)

The competitors

The following factorizations are frequently used :

- $M = VT_{ri}V^T$ (tridiagonalization)
- $M = LD_{bl}L^T$ (block diagonal)

but they suffer from the following disadvantages :

The competitors

The following factorizations are frequently used :

- $M = VT_{ri}V^T$ (tridiagonalization)
- $M = LD_{bl}L^T$ (block diagonal)

but they suffer from the following disadvantages :

- updating bordered tridiagonal factorizations is expensive because of the updating of V

The competitors

The following factorizations are frequently used :

- $M = VT_{ri}V^T$ (tridiagonalization)
- $M = LD_{bl}L^T$ (block diagonal)

but they suffer from the following disadvantages :

- updating bordered tridiagonal factorizations is expensive because of the updating of V
- deflating eigenvalues from a block diagonal is expensive because of the updating of D_{bl}

An anti-triangular decomposition

An anti-triangular decomposition

Theorem

Every $n \times n$ symmetric matrix H has a factorization $H = UMU^T$ where

$$M = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & Y^T \\ 0 & 0 & X & Z^T \\ 0 & Y & Z & W \end{bmatrix} \begin{matrix} \}n_0 \\ \}n_1 \\ \}n_2 \\ \}n_1 \end{matrix}, \quad X = eLL^T$$

where $U^T U = I_n$, $e = \pm 1$, $Y \in \mathbb{R}^{n_1 \times n_1}$, $L \in \mathbb{R}^{n_2 \times n_2}$ and $r = 2n_1 + n_2$

An anti-triangular decomposition

Theorem

Every $n \times n$ symmetric matrix H has a factorization $H = UMU^T$ where

$$M = \begin{array}{c} \left[\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & Y^T \\ 0 & 0 & X & Z^T \\ 0 & Y & Z & W \end{array}\right] \end{array} \begin{array}{l} \}n_0 \\ \}n_1 \\ \}n_2 \\ \}n_1 \end{array}, \quad X = eLL^T$$

where $U^TU = I_n$, $e = \pm 1$, $Y \in \mathbb{R}^{n_1 \times n_1}$, $L \in \mathbb{R}^{n_2 \times n_2}$ and $r = 2n_1 + n_2$

$$\text{Ex}: \quad T = \left[\begin{array}{c|cc|ccc|cc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & \times \\ 0 & 0 & 0 & 0 & 0 & 0 & \times & \times \\ \hline 0 & 0 & 0 & \times & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times & \times & \times \\ \hline 0 & 0 & \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times & \times & \times \end{array}\right]$$

Properties

$$M = \begin{array}{c} \left[\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & Y^T \\ 0 & 0 & eLL^T & Z^T \\ 0 & Y & Z & W \end{array}\right] \end{array} \begin{array}{l} \}n_0 \\ \}n_1 \\ \}n_2 \\ \}n_1 \end{array} , \quad e = \pm 1$$

- $r = n_1 + n_2 + n_1$

Properties

$$M = \begin{array}{c} \left[\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & Y^T \\ 0 & 0 & eLL^T & Z^T \\ 0 & Y & Z & W \end{array}\right] \begin{array}{l} \}n_0 \\ \}n_1 \\ \}n_2 \\ \}n_1 \end{array} \end{array}, \qquad e = \pm 1$$

- $r = n_1 + n_2 + n_1$
- L is full rank n_2 and can be chosen lower-triangular

Properties

$$M = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & Y^T \\ 0 & 0 & eLL^T & Z^T \\ 0 & Y & Z & W \end{bmatrix} \begin{matrix} \}n_0 \\ \}n_1 \\ \}n_2 \\ \}n_1 \end{matrix} , \qquad e = \pm 1$$

- $r = n_1 + n_2 + n_1$
- L is full rank n_2 and can be chosen lower-triangular
- Y is full rank n_1 and can be chosen anti-triangular

Properties

$$M = \begin{array}{c} \left[\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & Y^T \\ 0 & 0 & eLL^T & Z^T \\ 0 & Y & Z & W \end{array}\right] \end{array} \begin{array}{l} \}n_0 \\ \}n_1 \\ \}n_2 \\ \}n_1 \end{array}, \quad e = \pm 1$$

- $r = n_1 + n_2 + n_1$
- L is full rank n_2 and can be chosen lower-triangular
- Y is full rank n_1 and can be chosen anti-triangular
- $InM_+ = (n_1 + n_2, n_1, n - r)$ and $InM_- = (n_1, n_1 + n_2, n - r)$

Properties

$$M = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & Y^T \\ 0 & 0 & eLL^T & Z^T \\ 0 & Y & Z & W \end{bmatrix} \begin{array}{l} \}n_0 \\ \}n_1 \\ \}n_2 \\ \}n_1 \end{array} , \quad e = \pm 1$$

- $r = n_1 + n_2 + n_1$
- L is full rank n_2 and can be chosen lower-triangular
- Y is full rank n_1 and can be chosen anti-triangular
- $InM_+ = (n_1 + n_2, n_1, n - r)$ and $InM_- = (n_1, n_1 + n_2, n - r)$
- U is rectangular when removing zero rows/columns in M

Properties

$$M = \begin{array}{c} \left[\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & Y^T \\ 0 & 0 & eLL^T & Z^T \\ 0 & Y & Z & W \end{array}\right] \begin{array}{l} \}n_0 \\ \}n_1 \\ \}n_2 \\ \}n_1 \end{array} \end{array}, \quad e = \pm 1$$

- $r = n_1 + n_2 + n_1$
- L is full rank n_2 and can be chosen lower-triangular
- Y is full rank n_1 and can be chosen anti-triangular
- $InM_+ = (n_1 + n_2, n_1, n - r)$ and $InM_- = (n_1, n_1 + n_2, n - r)$
- U is rectangular when removing zero rows/columns in M
- The decomposition of a dense matrix requires $\mathcal{O}(n^3)$ flops

Properties

$$M = \begin{array}{c} \left[\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & Y^T \\ 0 & 0 & eLL^T & Z^T \\ 0 & Y & Z & W \end{array} \right] \end{array} \begin{array}{l} \}n_0 \\ \}n_1 \\ \}n_2 \\ \}n_1 \end{array}, \quad e = \pm 1$$

- $r = n_1 + n_2 + n_1$
- L is full rank n_2 and can be chosen lower-triangular
- Y is full rank n_1 and can be chosen anti-triangular
- $InM_+ = (n_1 + n_2, n_1, n - r)$ and $InM_- = (n_1, n_1 + n_2, n - r)$
- U is rectangular when removing zero rows/columns in M
- The decomposition of a dense matrix requires $\mathcal{O}(n^3)$ flops

For $\begin{bmatrix} 0 & B^T \\ B & A \end{bmatrix}$ one just requires a QR of B to get a (permuted) M

Proof (for e=1)

Let $$U^T H U = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & Y^T \\ 0 & 0 & LL^T & Z^T \\ 0 & Y & Z & W \end{bmatrix}, \quad U = [U_1|U_2|U_3|U_4]$$ then

Proof (for e=1)

$$\text{Let} \quad U^T H U = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & Y^T \\ 0 & 0 & LL^T & Z^T \\ 0 & Y & Z & W \end{bmatrix}, \quad U = [U_1|U_2|U_3|U_4] \quad \text{then}$$

- $[U_1]$ spans the *kernel* of H

Proof (for e=1)

Let $$U^T H U = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & Y^T \\ 0 & 0 & LL^T & Z^T \\ 0 & Y & Z & W \end{bmatrix}, \quad U = [U_1|U_2|U_3|U_4]$$ then

- $[U_1]$ spans the *kernel* of H
- $[U_1|U_2]$ spans a *maximal neutral* subspace of H

Proof (for e=1)

Let $$U^T H U = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & Y^T \\ 0 & 0 & LL^T & Z^T \\ 0 & Y & Z & W \end{bmatrix}, \quad U = [U_1|U_2|U_3|U_4]$$ then

- $[U_1]$ spans the *kernel* of H
- $[U_1|U_2]$ spans a *maximal neutral* subspace of H
- $[U_1|U_2|U_3]$ spans a *maximal non-negative* subspace of H

Proof (for e=1)

$$\text{Let} \quad U^T H U = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & Y^T \\ 0 & 0 & LL^T & Z^T \\ 0 & Y & Z & W \end{bmatrix}, \quad U = [U_1|U_2|U_3|U_4] \quad \text{then}$$

- $[U_1]$ spans the *kernel* of H
- $[U_1|U_2]$ spans a *maximal neutral* subspace of H
- $[U_1|U_2|U_3]$ spans a *maximal non-negative* subspace of H
- $[U_1|U_2|U_3|U_4]$ spans $\Re^n$

Proof (for e=1)

Let $$U^T H U = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & Y^T \\ 0 & 0 & LL^T & Z^T \\ 0 & Y & Z & W \end{bmatrix}, \quad U = [U_1|U_2|U_3|U_4]$$ then

- $[U_1]$ spans the *kernel* of H
- $[U_1|U_2]$ spans a *maximal neutral* subspace of H
- $[U_1|U_2|U_3]$ spans a *maximal non-negative* subspace of H
- $[U_1|U_2|U_3|U_4]$ spans $\Re^n$

The maximal neutral subspace is not unique but

Proof (for e=1)

Let $$U^T H U = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & Y^T \\ 0 & 0 & LL^T & Z^T \\ 0 & Y & Z & W \end{bmatrix}, \quad U = [U_1|U_2|U_3|U_4]$$ then

- $[U_1]$ spans the *kernel* of H
- $[U_1|U_2]$ spans a *maximal neutral* subspace of H
- $[U_1|U_2|U_3]$ spans a *maximal non-negative* subspace of H
- $[U_1|U_2|U_3|U_4]$ spans $\Re^n$

The maximal neutral subspace is not unique but

the maximal non-negative subspace containing it, is unique

(see Gohberg-Lancaster-Rodman)

Some examples

Let $H = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$

then neutral vectors are e.g. given by $u_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}, \quad \begin{bmatrix} 0 \\ a \\ b \\ \sqrt{a^2 + 2b^2} \end{bmatrix}$

Some examples

Let $H = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$

then neutral vectors are e.g. given by $u_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}, \quad \begin{bmatrix} 0 \\ a \\ b \\ \sqrt{a^2 + 2b^2} \end{bmatrix}$

In case 1 we have $M = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 2 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$

Some examples

Let $H = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$

then neutral vectors are e.g. given by $u_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}, \quad \begin{bmatrix} 0 \\ a \\ b \\ \sqrt{a^2 + 2b^2} \end{bmatrix}$

In case 1 we have $M = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 2 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$

When all positive eigenvalues are equal and all negative eigenvalues are equal, M is unique and essentially diagonal !

Rank r bordering problem

Rank r bordering problem

Let $H \approx UM_rU^T$, find a rank r approximation of the bordered matrix

$$\hat{H} = \begin{bmatrix} H & b \\ b^T & \alpha \end{bmatrix}$$

Rank r bordering problem

Let $H \approx UM_rU^T$, find a rank r approximation of the bordered matrix

$$\hat{H} = \begin{bmatrix} H & b \\ b^T & \alpha \end{bmatrix}$$

We thus need an optimal rank r approximation of the rank $r+2$ matrix

$$\tilde{H} = \begin{bmatrix} UM_rU^T & b \\ b^T & \alpha \end{bmatrix}.$$

Rank r bordering problem

Let $H \approx U M_r U^T$, find a rank r approximation of the bordered matrix

$$\hat{H} = \left[\begin{array}{cc} H & b \\ b^T & \alpha \end{array} \right]$$

We thus need an optimal rank r approximation of the rank $r+2$ matrix

$$\tilde{H} = \left[\begin{array}{cc} U M_r U^T & b \\ b^T & \alpha \end{array} \right].$$

One easily obtains the rank $r+2$ factorization $\tilde{H} = \tilde{U}\tilde{M}\tilde{U}^T$ where

$$\tilde{U} := \left[\begin{array}{cc|c} u_\perp & U & \\ \hline & & 1 \end{array} \right], \quad \tilde{M} := \left[\begin{array}{c|c|c} 0 & 0 & \rho \\ \hline 0 & M_r & r \\ \hline \rho & r^T & \alpha \end{array} \right], \quad b = Ur + \rho u_\perp$$

comes from a Gram Schmidt orthogonalization of $[U|b]$

Updating + downdating problem

Notice that up to a symmetric permutation, $\tilde{T}$ is anti-triangular :

$$\tilde{T} := \begin{bmatrix} 0 & 0 & 0 & 0 & \rho \\ 0 & 0 & 0 & S & r_s \\ 0 & 0 & eLL^T & W & r_w \\ 0 & S^T & W^T & G & r_g \\ \rho & r_s^T & r_w^T & r_g^T & \alpha \end{bmatrix}$$

where the matrix size n_1 increased by 1 (and the rank of $\tilde{T}$ by 2)

We need to "chop off" $\tilde{T}$'s 2 eigenvalues of smallest modulus : λ_1, λ_2
Inverse iteration (a few solves) will yield good approximations of v_1, v_2

Updating + downdating problem

Notice that up to a symmetric permutation, $\tilde{T}$ is anti-triangular :

$$\tilde{T} := \begin{bmatrix} 0 & 0 & 0 & 0 & \rho \\ 0 & 0 & 0 & S & r_s \\ 0 & 0 & eLL^T & W & r_w \\ 0 & S^T & W^T & G & r_g \\ \rho & r_s^T & r_w^T & r_g^T & \alpha \end{bmatrix}$$

where the matrix size n_1 increased by 1 (and the rank of $\tilde{T}$ by 2)

We need to "chop off" $\tilde{T}$'s 2 eigenvalues of smallest modulus : λ_1, λ_2
Inverse iteration (a few solves) will yield good approximations of v_1, v_2

Deflating them out amounts to rotating v_i to e_i and updating $\tilde{T}$ and $\tilde{U}$
(we know the new values of n_1 and n_2 from the signs of λ_1 and λ_2)

Updating + downdating problem

Notice that up to a symmetric permutation, $\tilde{T}$ is anti-triangular :

$$\tilde{T} := \begin{bmatrix} 0 & 0 & 0 & 0 & \rho \\ 0 & 0 & 0 & S & r_s \\ 0 & 0 & eLL^T & W & r_w \\ 0 & S^T & W^T & G & r_g \\ \rho & r_s^T & r_w^T & r_g^T & \alpha \end{bmatrix}$$

where the matrix size n_1 increased by 1 (and the rank of $\tilde{T}$ by 2)

We need to "chop off" $\tilde{T}$'s 2 eigenvalues of smallest modulus : λ_1, λ_2
Inverse iteration (a few solves) will yield good approximations of v_1, v_2

Deflating them out amounts to rotating v_i to e_i and updating $\tilde{T}$ and $\tilde{U}$
(we know the new values of n_1 and n_2 from the signs of λ_1 and λ_2)

If $\rho = 0$ the work to deflate the 0 eigenvalues is trivial

Updating + downdating problem

Notice that up to a symmetric permutation, $\tilde{T}$ is anti-triangular :

$$\tilde{T} := \begin{bmatrix} 0 & 0 & 0 & 0 & \rho \\ 0 & 0 & 0 & S & r_s \\ 0 & 0 & eLL^T & W & r_w \\ 0 & S^T & W^T & G & r_g \\ \rho & r_s^T & r_w^T & r_g^T & \alpha \end{bmatrix}$$

where the matrix size n_1 increased by 1 (and the rank of $\tilde{T}$ by 2)

We need to "chop off" $\tilde{T}$'s 2 eigenvalues of smallest modulus : λ_1, λ_2
Inverse iteration (a few solves) will yield good approximations of v_1, v_2

Deflating them out amounts to rotating v_i to e_i and updating $\tilde{T}$ and $\tilde{U}$
(we know the new values of n_1 and n_2 from the signs of λ_1 and λ_2)

If $\rho = 0$ the work to deflate the 0 eigenvalues is trivial
All of these steps can be performed in $\mathcal{O}(Nr)$ flops

Downsizing problem

- Let H_v be an orthogonal (Householder) transformation such that

$$H_v \mathbf{v} = \upsilon \; e_1, \qquad \upsilon = \mp \|\mathbf{v}\|_2, \quad \tilde{U} = \left[\begin{array}{c} \mathbf{v}^T \\ V \end{array} \right]$$

Downsizing problem

- Let H_v be an orthogonal (Householder) transformation such that

$$H_v \mathbf{v} = \upsilon\, e_1, \qquad \upsilon = \mp \|\mathbf{v}\|_2, \quad \tilde{U} = \begin{bmatrix} \mathbf{v}^T \\ V \end{bmatrix}$$

- Then

$$\tilde{U} H_v = \left[\begin{array}{c|ccc} \upsilon & 0 & \cdots & 0 \\ \hline \multicolumn{4}{c}{V H_v} \end{array} \right]$$

Downsizing problem

- Let H_v be an orthogonal (Householder) transformation such that

$$H_v \mathbf{v} = \upsilon \; e_1, \qquad \upsilon = \mp\|\mathbf{v}\|_2, \quad \tilde{U} = \begin{bmatrix} \mathbf{v}^T \\ V \end{bmatrix}$$

- Then

$$\tilde{U} H_v = \left[\begin{array}{c|ccc} \upsilon & 0 & \cdots & 0 \\ \hline \multicolumn{4}{c}{V H_v} \end{array} \right]$$

- To recover orthonormality of VH_v, divide its first column by $\tau := \sqrt{1 - \upsilon^2}$ and multiply the first column and row of $H_v \tilde{T} H_v$ by τ

Downsizing problem

- Let H_v be an orthogonal (Householder) transformation such that

$$H_v \mathbf{v} = \upsilon \, e_1, \qquad \upsilon = \mp\|\mathbf{v}\|_2, \quad \tilde{U} = \begin{bmatrix} \mathbf{v}^T \\ V \end{bmatrix}$$

- Then

$$\tilde{U} H_v = \left[\begin{array}{c|ccc} \upsilon & 0 & \cdots & 0 \\ \hline \multicolumn{4}{c}{V H_v} \end{array} \right]$$

- To recover orthonormality of VH_v, divide its first column by $\tau := \sqrt{1 - \upsilon^2}$ and multiply the first column and row of $H_v \tilde{T} H_v$ by τ
- The batman factorization of the scaled $\tilde{T}$ then has to be restored

Downsizing problem

- Let H_v be an orthogonal (Householder) transformation such that

$$H_v \mathbf{v} = \upsilon \, e_1, \qquad \upsilon = \mp \|\mathbf{v}\|_2, \quad \tilde{U} = \begin{bmatrix} \mathbf{v}^T \\ V \end{bmatrix}$$

- Then

$$\tilde{U} H_v = \left[\begin{array}{c|ccc} \upsilon & 0 & \cdots & 0 \\ \hline \multicolumn{4}{c}{V H_v} \end{array} \right]$$

- To recover orthonormality of $V H_v$, divide its first column by $\tau := \sqrt{1 - \upsilon^2}$ and multiply the first column and row of $H_v \tilde{T} H_v$ by τ
- The batman factorization of the scaled $\tilde{T}$ then has to be restored
- Any row of $\tilde{U}$ can be chosen to be removed this way

Downsizing problem

- Let H_v be an orthogonal (Householder) transformation such that

$$H_v \mathbf{v} = \upsilon \, e_1, \qquad \upsilon = \mp \|\mathbf{v}\|_2, \quad \tilde{U} = \begin{bmatrix} \mathbf{v}^T \\ V \end{bmatrix}$$

- Then

$$\tilde{U} H_v = \left[\begin{array}{c|ccc} \upsilon & 0 & \cdots & 0 \\ \hline \multicolumn{4}{c}{V H_v} \end{array} \right]$$

- To recover orthonormality of $V H_v$, divide its first column by $\tau := \sqrt{1 - \upsilon^2}$ and multiply the first column and row of $H_v \tilde{T} H_v$ by τ
- The batman factorization of the scaled $\tilde{T}$ then has to be restored
- Any row of $\tilde{U}$ can be chosen to be removed this way
- Choose the row of least norm to get a better approximation

Example (positive semi-definite)

The following matrix has rank 3

$$F(i,j) = \sum_{k=1}^{3} \exp\left(-\frac{(i-\mu_k)^2 + (j-\mu_k)^2}{2\sigma_k}\right), \qquad i,j = 1,\ldots,100,$$

with

$$\mu = \left[\begin{array}{ccc} 4 & 18 & 76 \end{array}\right], \quad \sigma = \left[\begin{array}{ccc} 10 & 20 & 5 \end{array}\right].$$

Let $F = Q\Lambda Q^T$ be its spectral decomposition and let $\tilde{\Delta} \in \mathbb{R}^{100\times 100}$ be a matrix of random numbers generated by the `matlab` function `randn`, and define $\Delta = \tilde{\Delta}/\|\tilde{\Delta}\|_2$. Consider the matrix

$$H = F + \varepsilon \Delta\Delta^T, \qquad \varepsilon = 1.0e-5$$

Example

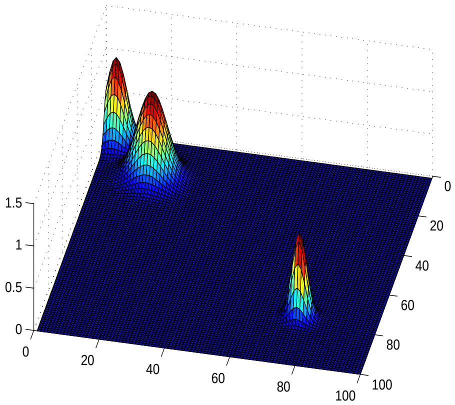


Figure: Graph of the size of the entries of the matrix H.

Example

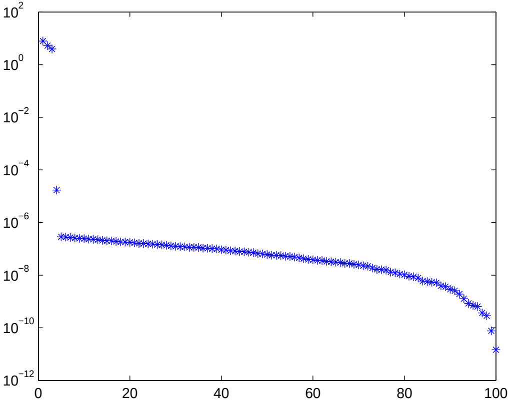


Figure: Distribution of the eigenvalues of the matrix H in logarithmic scale.

Example

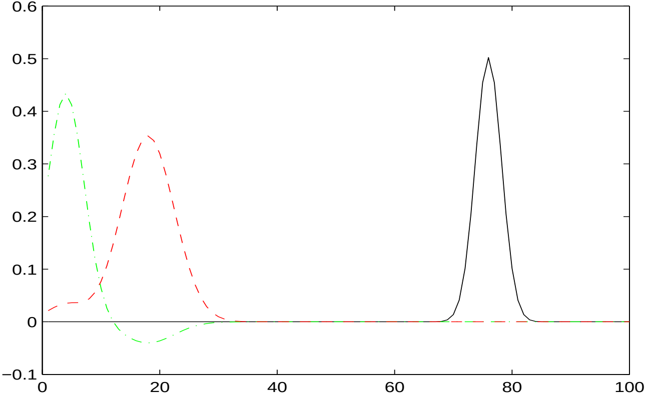


Figure: Plot of the three dominant eigenvectors of H

Example

Largest three eigenvalues of the matrix H (first column)

Largest three eigenvalues computed with our procedure (next cols) with rank $r = 3$ and window $n = 30, 40, 50$

λ_i	$\mu_i, n = 30$	$\mu_i, n = 40$	$\mu_i, n = 50$
7.949478e0	7.375113e0	7.820407e0	7.947127e0
5.261405e0	5.255163e0	5.260243e0	5.261384e0
3.963329e0	3.948244e0	3.963213e0	3.963329e0

Example

Largest three eigenvalues of the matrix H (first column)

Largest three eigenvalues computed with our procedure (next cols) with rank $r = 3$ and window $n = 30, 40, 50$

λ_i	$\mu_i, n = 30$	$\mu_i, n = 40$	$\mu_i, n = 50$
7.949478e0	7.375113e0	7.820407e0	7.947127e0
5.261405e0	5.255163e0	5.260243e0	5.261384e0
3.963329e0	3.948244e0	3.963213e0	3.963329e0

Reconstruction of dominant eigenvalues is quite good

Example

Largest three eigenvalues of the matrix H (first column)

Largest three eigenvalues computed with our procedure (next cols) with rank $r = 3$ and window $n = 30, 40, 50$

λ_i	$\mu_i, n = 30$	$\mu_i, n = 40$	$\mu_i, n = 50$
7.949478e0	7.375113e0	7.820407e0	7.947127e0
5.261405e0	5.255163e0	5.260243e0	5.261384e0
3.963329e0	3.948244e0	3.963213e0	3.963329e0

Reconstruction of dominant eigenvalues is quite good

The quality is improving with increasing window size n

Examples

We will consider a few examples of full decompositions
i.e. no downdating or downsizing

Ex1: $A = \mathrm{randn}(100); A = A + A'; \mathrm{Inertia}(A) = (50, 0, 50)$
Ex2: $A = \mathrm{rand}(100); A = A + A'; \mathrm{Inertia}(A) = (48, 0, 52)$
Ex3: A low rank
Ex4: A diagonal dominant

In each of the case we had good backward stability

$\frac{\|QMQ^T - A\|_2}{\|A\|_2}$	$\frac{\|LU - PA\|_2}{\|A\|_2}$	$\frac{\|QR - A\|_2}{\|A\|_2}$
2.49e-15	7.27e-16	1.12e-15

See movies

Random example

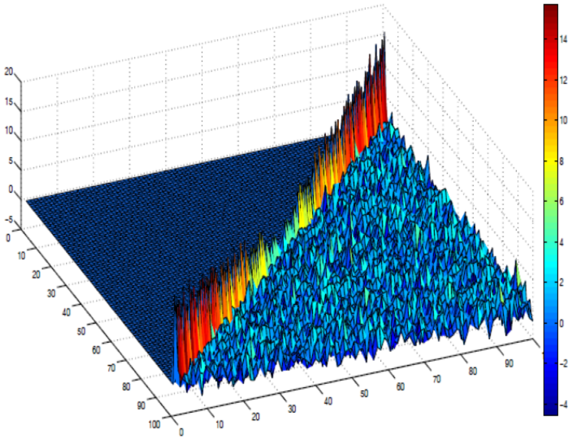


Figure: A random matrix $A = B + B^T$ with B generated by `randn(100,100)`

Random example

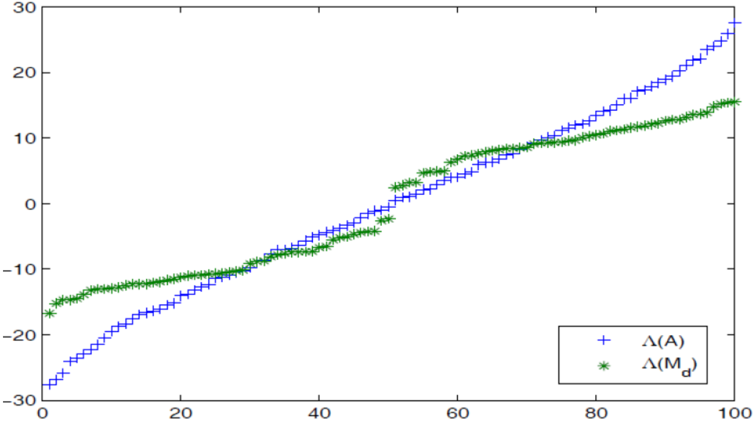


Figure: Eigenvalues and estimates based on the diagonal and anti-diagonal

Clustered example

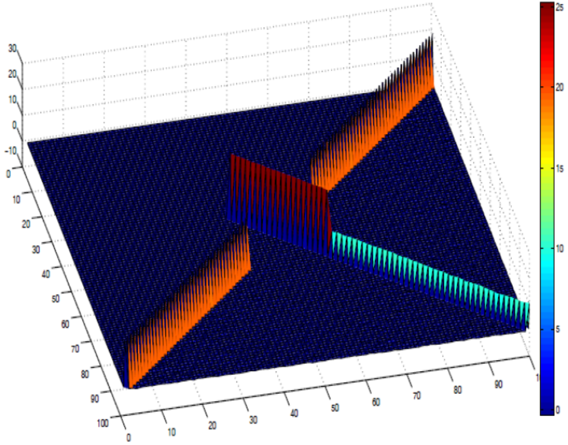


Figure: Matrix with a cluster of positive and a cluster of negative eigenvalues, generated by $A = U\mathit{diag}\left(-60I_{40}, 40I_{60}\right)U^{T} + E$ with $\|E\| \leq 1$

Clustered example

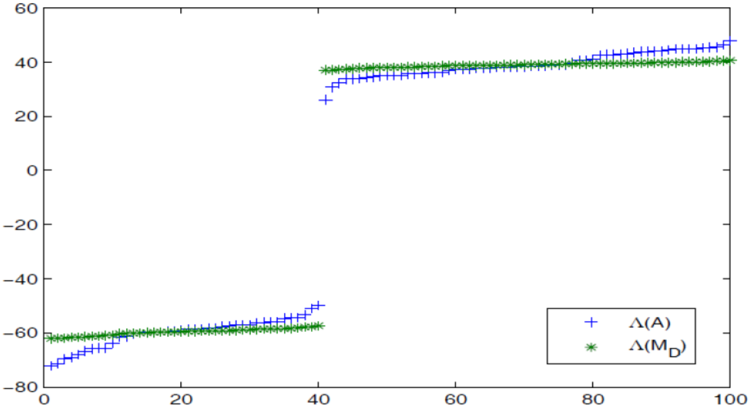


Figure: Eigenvalues and estimates based on the diagonal and anti-diagonal

Conclusions

We presented

- An efficient algorithm for computing incrementally the dominant eigenspace of a symmetric matrix

Conclusions

We presented

- An efficient algorithm for computing incrementally the dominant eigenspace of a symmetric matrix
- The overall complexity of the incremental updating technique to compute an $N \times r$ basis matrix U for the dominant eigenspace of H, is reduced to $\mathcal{O}(N^2 r)$

Conclusions

We presented

- An efficient algorithm for computing incrementally the dominant eigenspace of a symmetric matrix
- The overall complexity of the incremental updating technique to compute an $N \times r$ basis matrix U for the dominant eigenspace of H, is reduced to $\mathcal{O}(N^2 r)$
- When using both incremental updating and downsizing to compute the dominant eigenspace of $\hat{H}_n$ (an $n \times n$ principal submatrix of H_N), the complexity is reduced to $\mathcal{O}(Nnr)$.

References

Chahlaoui, Gallivan, Van Dooren, *An incremental method for computing dominant singular subspaces*, SIMAX, 2001

Hoegaerts, De Lathauwer, Goethals, Suykens, Vandewalle, De Moor, *Efficiently updating and tracking the dominant kernel principal components* Neural Networks, 2007.

Mastronardi, Tyrtishnikov, Van Dooren, *A fast algorithm for updating and downsizing the dominant kernel principal components*, SIMAX, 2010

Baker, Gallivan, Van Dooren, *Low-rank incrmenetal methods for computing dominant singular subspaces*, accepted, Lin. Alg. Appl., 2011

Mastronardi, Van Dooren, *Anti-triangular factorizations of symmetric matrices*, in preparation, 2011