Jacobi–type Algorithm for Cosine–Sine Decomposition

Saša Stanko, Sanja Singer

Faculty of Mechanical Engineering and Naval Architecture, University of Zagreb, Croatia

Numerical Analysis and Scientific Computation with Applications (NASCA 2018)

July 2–6, 2018, Kalamata, Greece

This work has been supported in part by
Croatian Science Foundation under the project IP–2014–09–3670.

Hrvatska zaklada za znanost

Outline

- Introduction
- Algorithm for 2×1 CS Decomposition
- Algorithm for 2×2 CS Decomposition
- Numerical experiments
- Conclusion

Introduction

Definition of 2×1 CS Decomposition

- Orthonormal matrix $Q \in \mathbb{R}^{n \times m}$
- 2×1 block structure

$$Q = \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix}, \qquad Q_1 \in \mathbb{R}^{k \times m}, \qquad Q_2 \in \mathbb{R}^{(n-k) \times m}$$

- Assumption $m \leq k$ and $k + m \leq n$
- Matrix Q can be decomposed as

$$Q = \begin{bmatrix} U_1 & 0 \\ 0 & U_2 \end{bmatrix} \begin{bmatrix} C \\ 0 \\ \hline S \\ 0 \end{bmatrix} V^T$$

where U_1, U_2, V are orthogonal, C, S are real, diagonal, non-negative, $C^2 + S^2 = I$

Definition of 2×2 CS Decomposition

- Orthogonal matrix $Q \in \mathbb{R}^{n \times n}$
- 2×2 block structure

$$Q = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix}, \qquad Q_{11} \in \mathbb{R}^{k \times m}$$

- Assumption $m \leq k$, $k + m \leq n$
- Matrix Q can be decomposed as

$$Q = \begin{bmatrix} U_1 & 0 \\ 0 & U_2 \end{bmatrix} \left[\begin{array}{c|ccc} C & -S & 0 & 0 \\ 0 & 0 & I_{k-m} & 0 \\ \hline S & C & 0 & 0 \\ 0 & 0 & 0 & I_{n-k-m} \end{array}\right] \begin{bmatrix} V_1 & 0 \\ 0 & V_2 \end{bmatrix}^T$$

where U_1, U_2, V_1, V_2 are orthogonal, C, S are real, diagonal, non-negative, $C^2 + S^2 = I$

Difficulties in computing CS decomposition

- Naive approach does not work
    - close singular values
    - small singular values

Difficulties in computing CS decomposition

- Naive approach does not work
  - close singular values
  - small singular values

Algorithm

1: Use SVD to compute $Q_1 = U_1 C V^T$
2: Use SVD to compute $Q_2 = U_2 S V^T$

Difficulties in computing CS decomposition

- Naive approach does not work
  - close singular values
  - small singular values

Algorithm

1: Use SVD to compute $Q_2 = U_2 S V^T$
2: Set $X = Q_1 V$
3: Set $C = \text{diag}(\text{cnorm}(X))$
4: Set $U_1 = X C^{-1}$

Difficulties in computing CS decomposition

- Naive approach does not work
  - close singular values
  - small singular values

Algorithm

1: Use SVD to compute $Q_2 = U_2 S V^T$
2: Set $X = Q_1 V$
3: Use QR to compute $X = U_1 R$
4: Set $C = \text{diag}(\text{diag}(R))$

Existing solutions

- SVD plus corrections by Gilbert W. Stewart in 1982
- SVD plus corrections by Charles Van Loan in 1985
- Sketch of 2×2 CS decomposition by Vjeran Hari in 2005
- The first 2×2 CS decomposition by Brian D. Sutton in 2009
- Divide and conquer by Brian D. Sutton in 2013
- Polar decomposition by Evan S. Gawlik et al. in 2018

Algorithm for 2×1 CSD

Jacobi–type algorithm

Any algorithm that can be specified by

- subproblem selection
- pivoting strategy
- stopping criterion

Eigenvalue problem

- Symmetric matrix $A \in \mathbb{R}^{n \times n}$
- Select pivoting element (p, q)
- Solve subproblem

$$\begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix}^T \begin{bmatrix} a_{pp} & a_{pq} \\ a_{qp} & a_{qq} \end{bmatrix} \begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix} = \begin{bmatrix} a'_{pp} & 0 \\ 0 & a'_{qq} \end{bmatrix}$$

by solving

$$\tan(2\varphi) = \frac{2a_{pq}}{a_{pp}^2 - a_{qq}^2}$$

for smaller $\tan\varphi$ and setting

$$\cos\varphi = \frac{1}{\sqrt{1+\tan^2\varphi}} \qquad \text{and} \qquad \sin\varphi = \frac{\tan\varphi}{\sqrt{1+\tan^2\varphi}}$$

- Transform A to

$$A' = J(p,q,\varphi)^T A J(p,q,\varphi)$$

Singular value problem

- Matrix $A \in \mathbb{R}^{n \times m}$
- Eigenvalue problem for $A^T A$
- Select columns p and q
- Solve subproblem

$$\begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix}^T \begin{bmatrix} a_p^T a_p & a_p^T a_q \\ a_q^T a_p & a_q^T a_q \end{bmatrix} \begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix} = \begin{bmatrix} a'_{pp} & 0 \\ 0 & a'_{qq} \end{bmatrix}$$

by solving

$$\tan(2\varphi) = \frac{2a_p^T a_q}{\|a_p\|^2 - \|a_q\|^2}$$

for smaller $\tan\varphi$ and setting

$$\cos\varphi = \frac{1}{\sqrt{1+\tan^2\varphi}} \qquad \text{and} \qquad \sin\varphi = \frac{\tan\varphi}{\sqrt{1+\tan^2\varphi}}$$

- Transform A to

$$A' = AJ(p, q, \varphi)$$

CS Decomposition

- Orthonormal matrix $Q \in \mathbb{R}^{n \times m}$ with 2×1 block structure

$$Q = \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} =: \begin{bmatrix} A \\ B \end{bmatrix}$$

- Select columns p and q
- Solve subproblems

$$\begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix}^T \begin{bmatrix} a_p^T a_p & a_p^T a_q \\ a_q^T a_p & a_q^T a_q \end{bmatrix} \begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix} = \begin{bmatrix} a'_{pp} & 0 \\ 0 & a'_{qq} \end{bmatrix}$$

and

$$\begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix}^T \begin{bmatrix} b_p^T b_p & b_p^T b_q \\ b_q^T b_p & b_q^T b_q \end{bmatrix} \begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix} = \begin{bmatrix} b'_{pp} & 0 \\ 0 & b'_{qq} \end{bmatrix}$$

- Transform Q to

$$Q' = QJ(p, q, \varphi)$$

Which subproblem to solve?

- Choose subproblem with smaller Frobenius norm
- Choose subproblem more sensitive to errors in rotation

Sensitivity

Theorem

Let σ_1, σ_2 be singular values of matrix $\begin{bmatrix} x & y \end{bmatrix}$ and let V be the matrix of its right singular vectors. The angle between columns of the matrix

$$\begin{bmatrix} x' & y' \end{bmatrix} = \begin{bmatrix} x & y \end{bmatrix} V \left(I + \begin{bmatrix} \epsilon_{11} & -\epsilon_{12} \\ \epsilon_{12} & \epsilon_{11} \end{bmatrix} \right), \quad |\epsilon_{11}|, |\epsilon_{12}| \leq \varepsilon$$

satisfies

$$\cos \measuredangle(x', y') = \frac{|\epsilon_{12}|}{1 + 2\epsilon_{11}} \left| \frac{\sigma_1}{\sigma_2} - \frac{\sigma_2}{\sigma_1} \right| + \mathcal{O}(\varepsilon^2)$$

Sensitivity (continued)

- In finite precision arithmetic computed rotation is

$$\hat{V} = \begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix} \left(I + \begin{bmatrix} \epsilon_{11} & -\epsilon_{12} \\ \epsilon_{12} & \epsilon_{11} \end{bmatrix} \right), \quad |\epsilon_{11}|, |\epsilon_{12}| \le \varepsilon$$

- Transformation with $\hat{V}$ instead of V gives

$$\begin{bmatrix} a'_p & a'_q \end{bmatrix} = \begin{bmatrix} a_p & a_q \end{bmatrix} \hat{V} \qquad \text{and} \qquad \begin{bmatrix} b'_p & b'_q \end{bmatrix} = \begin{bmatrix} b_p & b_q \end{bmatrix} \hat{V}$$

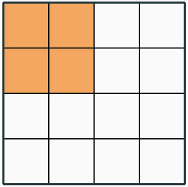
- Theorem implies

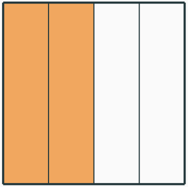
$$\frac{\cos\measuredangle(a'_p, a'_q)}{\cos\measuredangle(b'_p, b'_q)} \approx \tan\theta_1 \tan\theta_2$$

  where $\cos\theta_1$, $\cos\theta_2$ are singular values of matrix $\begin{bmatrix} a_p & a_q \end{bmatrix}$ and $\sin\theta_1$, $\sin\theta_2$ are singular values of matrix $\begin{bmatrix} b_p & b_q \end{bmatrix}$
- Approximate $\tan\theta_1 \tan\theta_2$ by solving both subproblems

Pivoting strategy

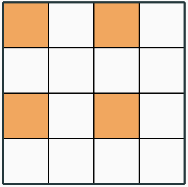
- Pivot is a pair (p, q)
- Sweep is a sequence of all distinct pivots
- Strategies
  - row–cyclic and column–cyclic
  - modulus
- Block and hierarchical versions

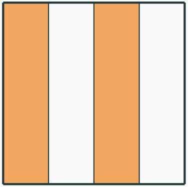




Pivoting strategy

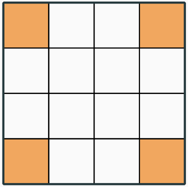
- Pivot is a pair (p, q)
- Sweep is a sequence of all distinct pivots
- Strategies
  - row–cyclic and column–cyclic
  - modulus
- Block and hierarchical versions

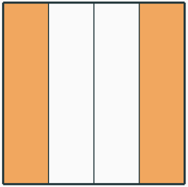




Pivoting strategy

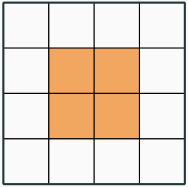
- Pivot is a pair (p, q)
- Sweep is a sequence of all distinct pivots
- Strategies
  - row–cyclic and column–cyclic
  - modulus
- Block and hierarchical versions

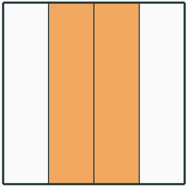




Pivoting strategy

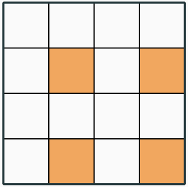
- Pivot is a pair (p, q)
- Sweep is a sequence of all distinct pivots
- Strategies
  - row–cyclic and column–cyclic
  - modulus
- Block and hierarchical versions

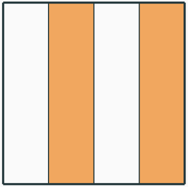




Pivoting strategy

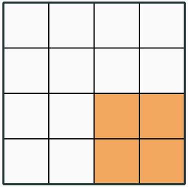
- Pivot is a pair (p, q)
- Sweep is a sequence of all distinct pivots
- Strategies
  - row–cyclic and column–cyclic
  - modulus
- Block and hierarchical versions

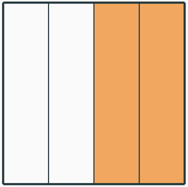




Pivoting strategy

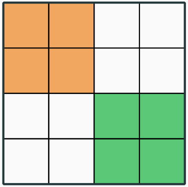
- Pivot is a pair (p, q)
- Sweep is a sequence of all distinct pivots
- Strategies
  - row–cyclic and column–cyclic
  - modulus
- Block and hierarchical versions

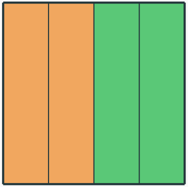




Pivoting strategy

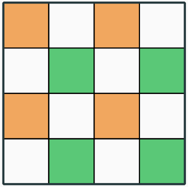
- Pivot is a pair (p, q)
- Sweep is a sequence of all distinct pivots
- Strategies
    - row–cyclic and column–cyclic
    - modulus
- Block and hierarchical versions

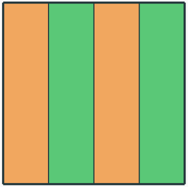




Pivoting strategy

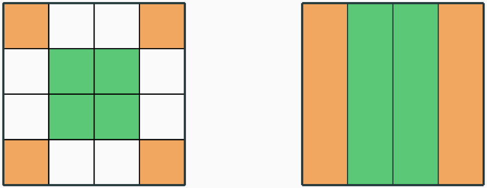
- Pivot is a pair (p, q)
- Sweep is a sequence of all distinct pivots
- Strategies
  - row–cyclic and column–cyclic
  - modulus
- Block and hierarchical versions





Pivoting strategy

- Pivot is a pair (p, q)
- Sweep is a sequence of all distinct pivots
- Strategies
  - row–cyclic and column–cyclic
  - modulus
- Block and hierarchical versions



Stopping criteria

- Skip subproblem if it is almost solved

$$\begin{bmatrix} a_p^T a_p & a_p^T a_q \\ a_q^T a_p & a_q^T a_q \end{bmatrix}$$

  - small angle between columns

$$\frac{|a_p^T a_q|}{\|a_p\| \|a_q\|} < \varepsilon$$

  - computed cosine in rotation equal to 1
- Stop after all pivots in a sweep have been skipped

Final stage

- When stopped, algorithm has produced

$$Q = \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} = \begin{bmatrix} Q_1' \\ Q_2' \end{bmatrix} V^T$$

- Matrix V is obtained by accumulating rotations
- Matrices C and S are obtained by

$$C = \text{diag}(\text{cnorm}(Q_1')) \qquad \text{and} \qquad S = \text{diag}(\text{cnorm}(Q_2'))$$

- Matrices U_1 and U_2 are obtained by

$$U_1 = Q_1' C^{-1} \qquad \text{and} \qquad U_2 = Q_2' S^{-1}$$

- Or by QR factorization of Q_1 and Q_2

Accuracy of computed matrices

- Matrices V, C i S are computed accurately
- Matrices U_1 and U_2 are computed with naive approach
- If stopping criterion in finite precision arithmetic satisfies

$$\left| \left(\frac{a_p}{\|a_p\|} \right)^T \frac{a_q}{\|a_q\|} \right| < \varepsilon$$

then matrices U_1 and U_2 satisfy

$$\|U_1^T U_1 - I\| = \mathcal{O}(\varepsilon) \qquad \text{and} \qquad \|U_2^T U_2 - I\| = \mathcal{O}(\varepsilon)$$

Algorithm for 2×2 CSD

Generalization of previous algorithm

- Orthogonal matrix $Q \in \mathbb{R}^{n \times n}$ with 2×2 block structure

$$Q = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix}$$

- Previous approach leads to two 2×1 CS decompositions
- How to merge results?
- Instead, compute one 2×1 CS decomposition and V_2

Our approach

- The goal is

$$Q = \begin{bmatrix} U_1 & 0 \\ 0 & U_2 \end{bmatrix} \left[\begin{array}{c|ccc} C & -S & 0 & 0 \\ 0 & 0 & I_{k-m} & 0 \\ \hline S & C & 0 & 0 \\ 0 & 0 & 0 & I_{n-k-m} \end{array}\right] \begin{bmatrix} V_1 & 0 \\ 0 & V_2 \end{bmatrix}^T$$

- Start with 2×1 CSD of the first block–column of Q
- Write U_1 and U_2 as

$$U_1 = \begin{bmatrix} U_{11} & U_{12} \end{bmatrix} \qquad \text{and} \qquad U_2 = \begin{bmatrix} U_{21} & U_{22} \end{bmatrix}$$

- Write V_2 as

$$V_2 = \begin{bmatrix} V_{21} & V_{22} & V_{23} \end{bmatrix}$$

- Then

$$V_{21} = -Q_{12}^* U_{11} S^{-1} = Q_{22}^* U_{21} C^{-1}, \quad V_{22} = Q_{12}^* U_{12}, \quad V_{23} = Q_{22}^* U_{22}$$

Computation of V_2

- Decide which expression to use for V_{21}

$$V_{21} = -Q_{12}^* U_{11} S^{-1} = Q_{22}^* U_{21} C^{-1}$$

- Make decision for each column j
- Use first expression if $[S]_{jj} \geq 1/\sqrt{2}$, otherwise use second
- Compare it with Sutton's approach

$$V_2 = -Q_{12}^* U_{11} S + Q_{22}^* U_{21} C$$

- If 2×1 CS decomposition is computed backward stably, so is 2×2 CS decomposition

Numerical experiments

When is algorithm stable?

- Described by Van Loan in 1982, extended by Sutton in 2009
- Assume

$$\|Q^T Q - I\|_2 = \varepsilon$$

- Algorithm is stable if computed matrices satisfy

$$\|U_1^T U_1 - I\|_2 \approx \varepsilon\,, \quad \|U_2^T U_2 - I\|_2 \approx \varepsilon$$

$$\|V_1^T V_1 - I\|_2 \approx \varepsilon\,, \quad \|V_2^T V_2 - I\|_2 \approx \varepsilon$$

$$\begin{bmatrix} U_1 & 0 \\ 0 & U_2 \end{bmatrix}^T Q \begin{bmatrix} V_1 & 0 \\ 0 & V_2 \end{bmatrix} \approx \left[\begin{array}{c|ccc} C & -S & 0 & 0 \\ 0 & 0 & I_{k-m} & 0 \\ \hline S & C & 0 & 0 \\ 0 & 0 & 0 & I_{n-k-m} \end{array}\right]$$

When is algorithm stable?

- Described by Van Loan in 1982, extended by Sutton in 2009
- Assume

$$\|Q^T Q - I\|_2 = \varepsilon$$

- Algorithm is stable if computed matrices satisfy

$$\|U_1^T U_1 - I\|_2 \approx \varepsilon\,, \quad \|U_2^T U_2 - I\|_2 \approx \varepsilon$$

$$\|V_1^T V_1 - I\|_2 \approx \varepsilon\,, \quad \|V_2^T V_2 - I\|_2 \approx \varepsilon$$

$$\left\| U_1^T Q_{11} V_1 - \begin{bmatrix} C \\ 0 \end{bmatrix} \right\|_2 \approx \varepsilon\,, \quad \left\| U_1^T Q_{12} V_2 - \begin{bmatrix} -S & 0 & 0 \\ 0 & I_{k-m} & 0 \end{bmatrix} \right\|_2 \approx \varepsilon$$

$$\left\| U_2^T Q_{21} V_1 - \begin{bmatrix} S \\ 0 \end{bmatrix} \right\|_2 \approx \varepsilon\,, \quad \left\| U_2^T Q_{22} V_2 - \begin{bmatrix} C & 0 & 0 \\ 0 & 0 & I_{n-k-m} \end{bmatrix} \right\|_2 \approx \varepsilon$$

Examples

- First extensive testing was done by Sutton in 2009
- Haar measure

```
[Q, ] = qr(randn(n))
Q = Q * diag(sign(randn(n, 1)))
```

- Clusters of singular values

```
delta = 10.^(-18*rand(n/2+1, 1))
theta = pi/2 * cumsum(delta(1:n_half)) / sum(delta)
C = diag(cos(theta))
S = diag(sin(theta))
```

Results

- Ratio of maximal error and $\|Q^T Q - I\|_2$
- Repeated 50 times for each dimension
- Results for Haar measure

	small norm		more sensitive	
n	max err	mean err	max err	mean err
8	6.3299	1.6650	3.4773	1.7572
16	3.5957	2.1857	4.0737	2.1895
32	4.8882	3.1968	5.9868	3.3542
64	7.5288	5.0814	6.3542	5.2818
128	16.1404	9.7910	16.1378	9.8586
256	28.4062	21.6186	31.5716	21.8330

Results

- Ratio of maximal error and $\|Q^T Q - I\|_2$
- Repeated 50 times for each dimension
- Results for clusters of singular values

	small norm		more sensitive	
n	max err	mean err	max err	mean err
8	2.6013	1.4913	2.7823	1.2905
16	3.1457	2.0416	4.0652	2.2305
32	6.2167	3.7636	5.6218	3.8143
64	9.7981	7.2262	11.0153	7.1827
128	22.3003	14.1390	21.3632	14.1483
256	43.0313	28.1370	47.9561	27.5747

Conclusion

Conclusion and future work

- Conclusion
  - Development of a new algorithm for 2×1 CS decomposition
  - Construction of Jacobi rotations
  - Analysis of stopping criterion
  - Modification of an algorithm for 2×2 CS decomposition
- Future work
  - Generalization to unitary matrices
  - Block and parallel version

Thank you for your attention!