Tensor Decompositions and Compression

AWM Workshop
Joint Meetings 2006

Carla D. Moravitz Martin

carlam@math.cornell.edu

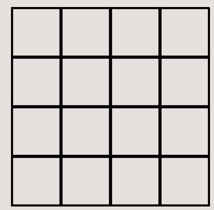
Department of Mathematics

Cornell University

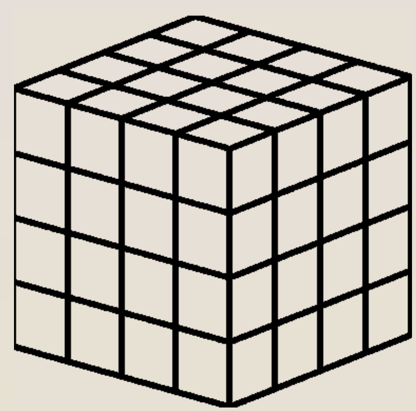
January 15, 2006

What are Tensors?

- Second-order tensor $A = (a_{ij}) \in \mathbb{R}^{n_1 \times n_2}$



- Third-order tensor $\mathcal{A} = (a_{ijk}) \in \mathbb{R}^{n_1 \times n_2 \times n_3}$



- p^{th}-order tensor $\mathcal{A} = (a_{i_1 i_2 \ldots i_p}) \in \mathbb{R}^{n_1 \times \cdots \times n_p}$

Leading Applications

- Chemometrics
- Psychometrics
- Computer Image Recognition
- Chromatography
- Other applications using multiway data analysis (e.g., acoustics, phylogenetics)

Motivation: A two-way decomposition

Suppose $U \in \mathbb{R}^{m\times m}$, $V \in \mathbb{R}^{n\times n}$ are orthogonal, and $\Sigma = U^T AV$, then

$$\begin{aligned} A = U\Sigma V^T &= \sum_{i=1}^{m}\sum_{j=1}^{n} \sigma_{ij} u_i v_j^T \\ &= \sum_{i=1}^{m}\sum_{j=1}^{n} \sigma_{ij} (u_i \circ v_j) \end{aligned}$$

where $u_i = U(:,i)$, $v_j = V(:,j)$

SVD Representation

Can choose $U \in \mathbb{R}^{m\times m}$, $V \in \mathbb{R}^{n\times n}$ orthogonal so $\Sigma = U^T A V = \mathrm{diag}(\sigma_1, \ldots, \sigma_r)$:

$$A = U\Sigma V^T = \sum_{i=1}^{r} \sigma_i u_i v_i^T = \sum_{i=1}^{r} \sigma_i (u_i \circ v_i)$$

where $u_i = U(:, i)$, $v_i = V(:, i)$ and $\mathrm{rank}(A) = r$.

Reduction to diagonal form and rank-revealing properties make the SVD appealing to many applications.

Why we care about Compression: Image Compression Example

Mars landscape

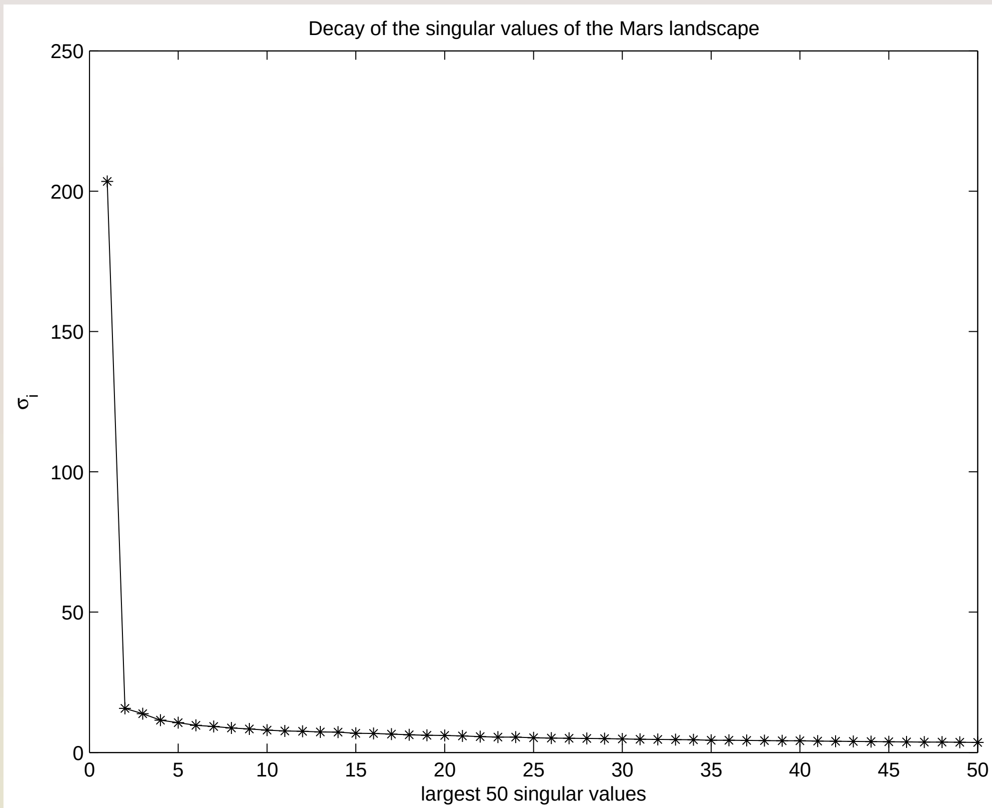
Photo Credit: NASA/JPL/Cornell, 2004

A 400×400 pixel image means sending $160,000$ pixels back to earth!

Singular Values of Mars Image

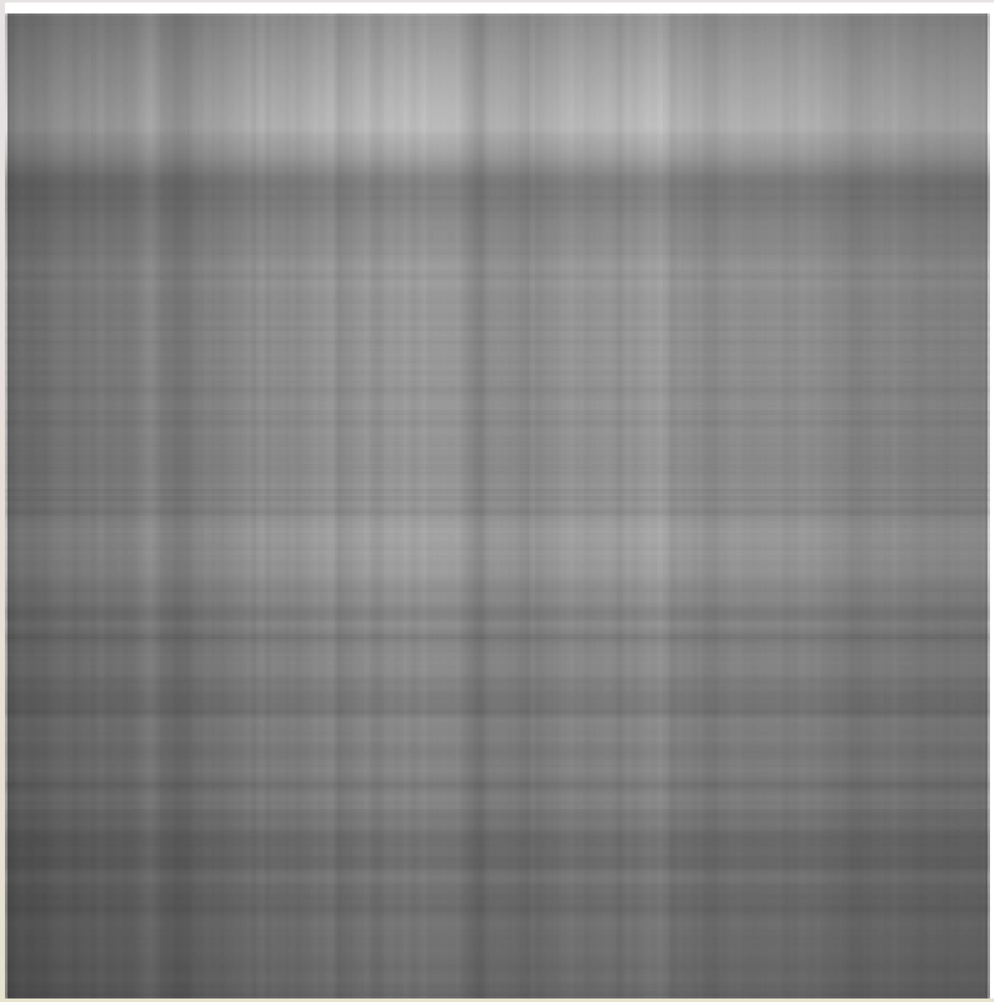
Stored image $\longrightarrow A \in \mathbb{R}^{400\times 400}$

$\text{Rank}(A) = 400$



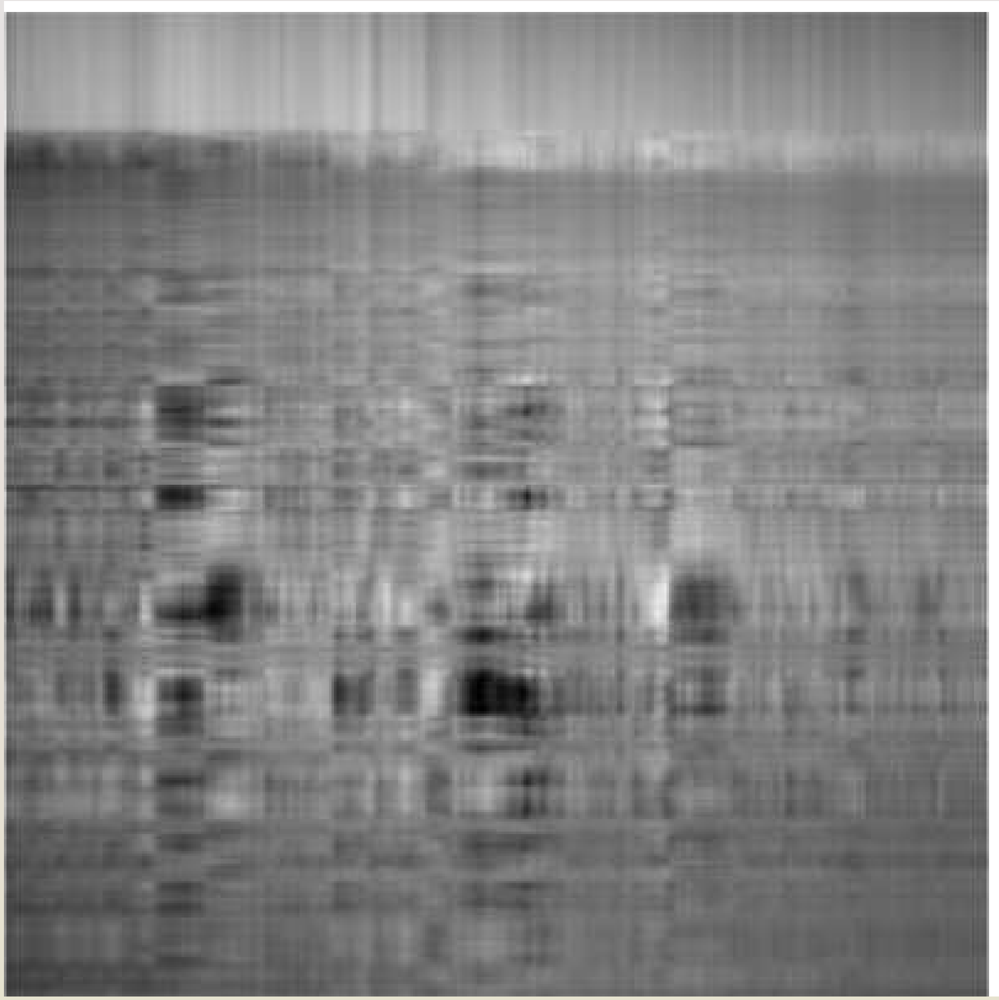
Plot of the Singular Values

Low Rank Approximations to Mars Image



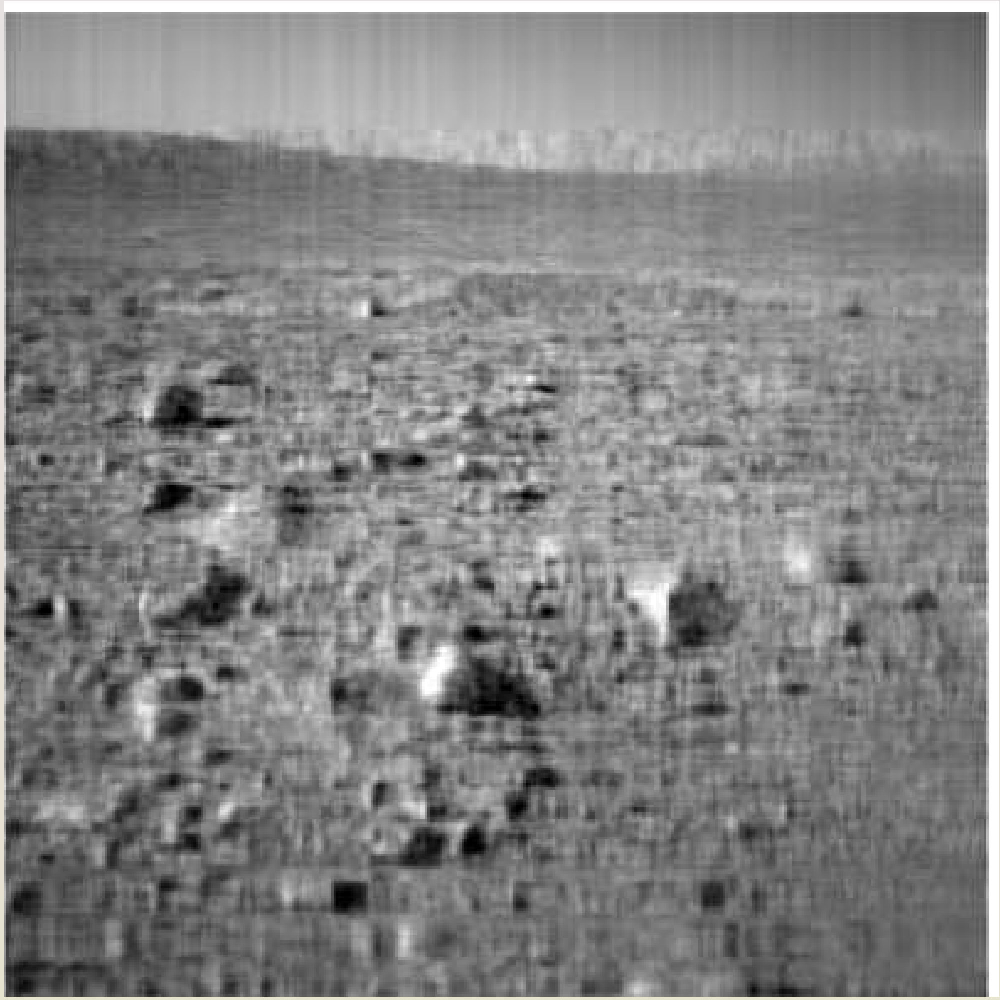
Rank-1 approximation (0.5% of data)

Low Rank Approximations to Mars Image



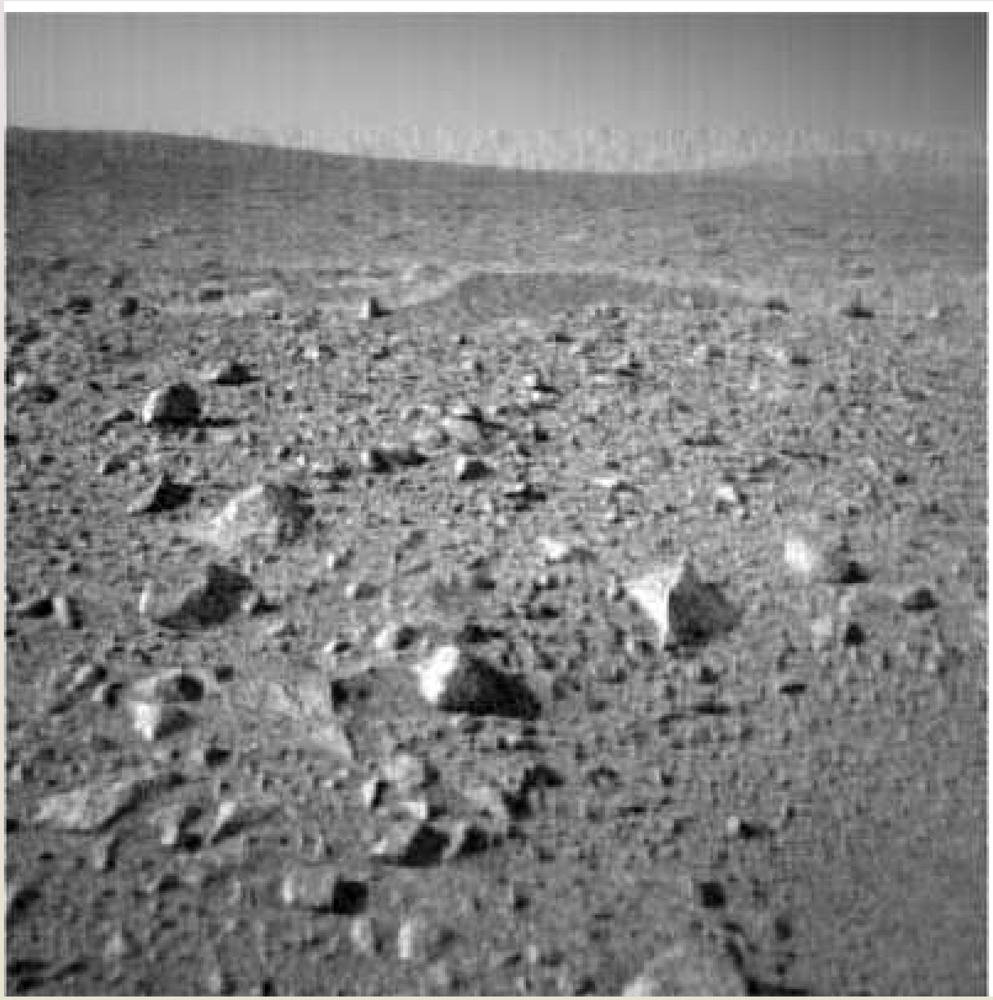
Rank-5 approximation (2.5% of data)

Low Rank Approximations to Mars Image



Rank-20 approximation (10% of data)

Low Rank Approximations to Mars Image



Rank-50 approximation (25% of data)

Low Rank Approximations to Mars Image

Rank-100 approximation (50% of data)

Low Rank Approximations to Mars Image

Rank-130 approximation (65% of data)

Low Rank Approximations to Mars Image

True Image

Tensor Decompositions

Let $\mathcal{A} \in \mathbb{R}^{m \times n \times p}$

Goal: To find "interesting" $U \in \mathbb{R}^{m \times m}$, $V \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{p \times p}$, and $\Sigma = (\sigma_{ijk}) \in \mathbb{R}^{m \times n \times p}$ such that

$$\mathcal{A} \;=\; \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{p} \sigma_{ijk} (u_i \circ v_j \circ w_k)$$

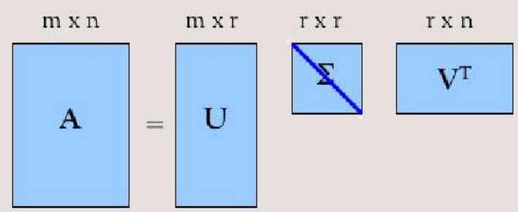
where $u_i = U(:, i)$, $v_j = V(:, j)$, $w_k = W(:, k)$

Useful Tensor Decompositions

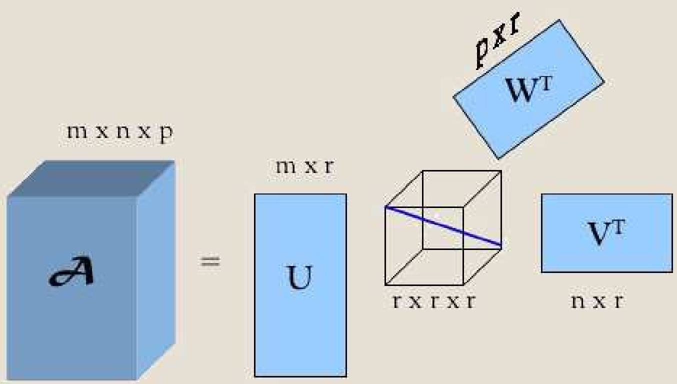
$$\mathcal{A} \quad = \quad \sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{k=1}^{p}\sigma_{ijk}(u_i \circ v_j \circ w_k)$$

1. U, V, W orthogonal
2. $\Sigma = (\sigma_{ijk})$ "compressed"
3. Rank revealing - what is rank?

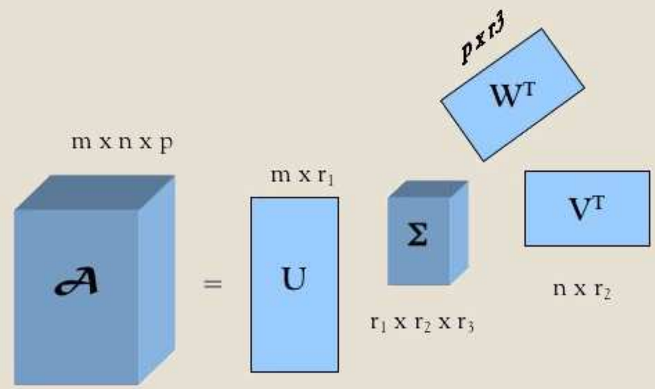
Orthogonal *or* Diagonal for Tensors



Case 1: Diagonal Σ



Case 2: Orthogonal U, V, W



General Tensor Rank (diagonalizing)

Tensor rank of $\mathcal{A} \in \mathbb{R}^{m \times n \times p}$ is the minimum number of rank-1 tensors that sum to $\mathcal{A}$ in linear combination.

If a tensor $\mathcal{A}$ has a minimal representation as

$$\mathcal{A} = \sum_{i=1}^{r} \sigma_i (u_i \circ v_i \circ w_i),$$

then $\operatorname{rank}(\mathcal{A}) = r$.

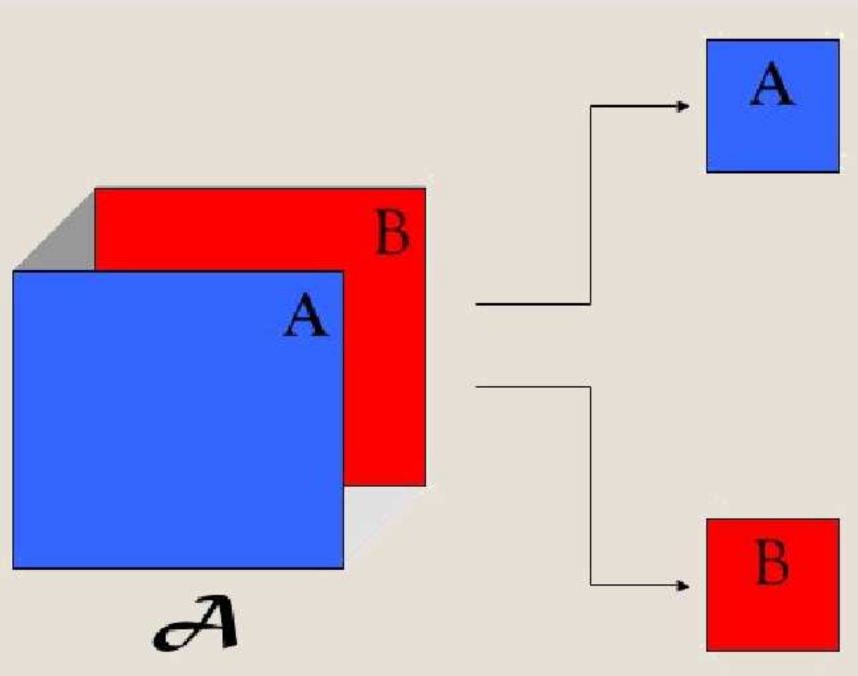
Tensor Rank is complicated ...

- Formula for the maximum possible rank of a tensor does not exist (as far as we know...)
- No known method to compute the "minimum" tensor decomposition
- Minimum tensor representation not necessarily orthogonal (Denis and Dhorne, 1989)
- A tensor over $\mathbb{R}$ may have a different rank then the same tensor considered over $\mathbb{C}$ (Kruskal, 1989)
- Set of rank-deficient tensors has positive volume (Kruskal, 1989)

Relating Tensor Rank to Eigenvalue Decompositions

Generalized Eigenvalue Problem

Let $\mathcal{A} \in \mathbb{R}^{n \times n \times 2}$.



Solve

$$Ax = \lambda Bx$$

Results: Connections between Rank and Generalized Eigenvalues

Let $A, B \in \mathbb{R}^{2\times 2}$. Then,

- if the generalized eigenvalues of A and B are real with a full set of eigenvectors, then the resulting tensor has maximum rank of two

- if the generalized eigenvalues of A and B are complex conjugates, then the resulting tensor is rank-three

- if the generalized eigenvalues of A and B are repeated with dependent eigenvectors, then the resulting tensor is rank-three

Results for $n \times n \times 2$ Tensors

If $A, B \in \mathbb{R}^{n \times n}$ and at least one of A and B is invertible with a full set of generalized eigenvectors, then the maximum tensor rank is

$$n + k$$

where k is the number of complex conjugate generalized eigenpairs.

Conclusion:

$$\text{rank} \quad \leq \quad \lfloor 3n/2 \rfloor$$

Compressing Tensors

(Orthogonal/SVD-like Ideas)

Goal of Compression

In the representation

$$\mathcal{A} = \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n}\sigma_{ijk}(u_i \circ v_j \circ w_k),$$

can we choose orthogonal $U, V, W \in \mathbb{R}^{n\times n}$ so that $\Sigma = (\sigma_{ijk})$ is "compressed"?

(i.e., Σ has most of its "mass" in a relatively small number of σ_{ijk})

Jacobi-Compression Algorithm

- Extension of Jacobi SVD Algorithm for matrices
- Computes tensor representations of $2 \times 2 \times 2$ subtensors
- Maximizes $\displaystyle\sum_{i=1}^{n} \sigma_{iii}^2 \quad \text{or} \quad \sum_{i=1}^{n} \sigma_{iii}$

Review of Jacobi SVD Algorithm

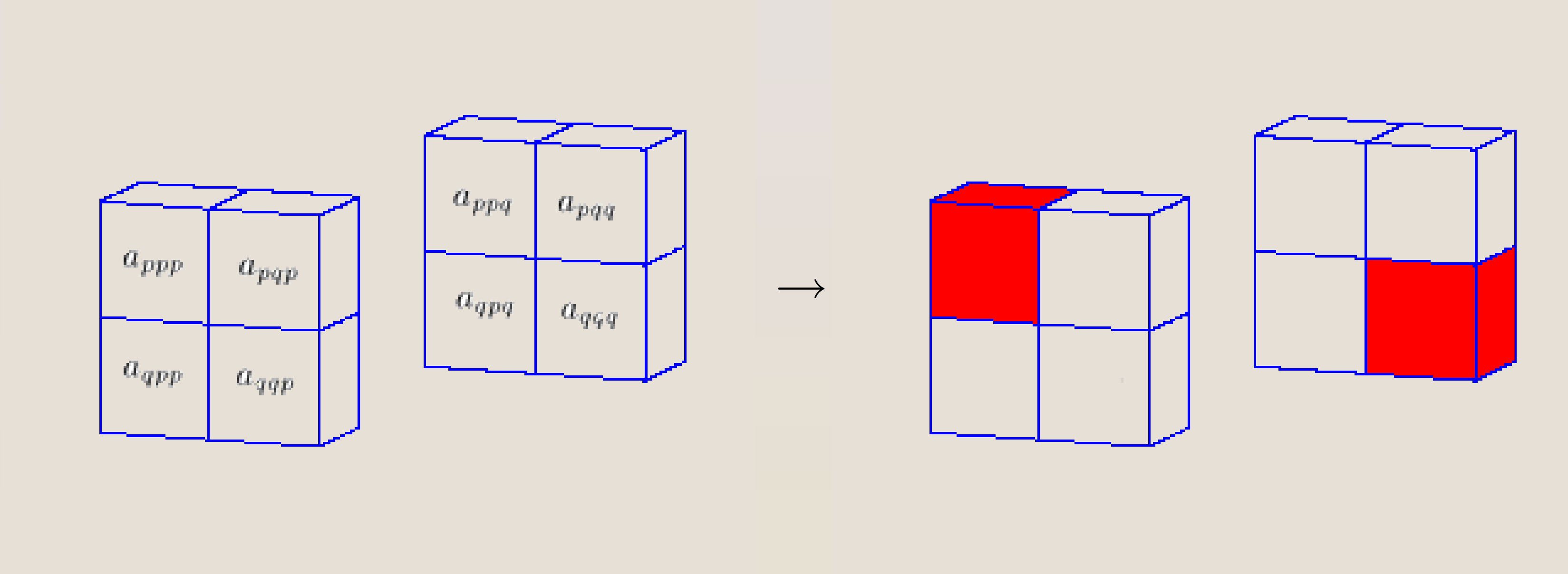
- Computes SVD of 2×2 submatrices of $A \in \mathbb{R}^{n \times n}$
- Pick a (p, q) pair and compute (c_1, s_1), (c_2, s_2) so that

$$\begin{bmatrix} c_1 & s_1 \\ -s_1 & c_1 \end{bmatrix}^T \begin{bmatrix} a_{pp} & a_{pq} \\ a_{qp} & a_{qq} \end{bmatrix} \begin{bmatrix} c_2 & s_2 \\ -s_2 & c_2 \end{bmatrix} = \begin{bmatrix} \sigma_p & 0 \\ 0 & \sigma_q \end{bmatrix}$$

- Update affected portions of A

Jacobi-Compression Algorithm

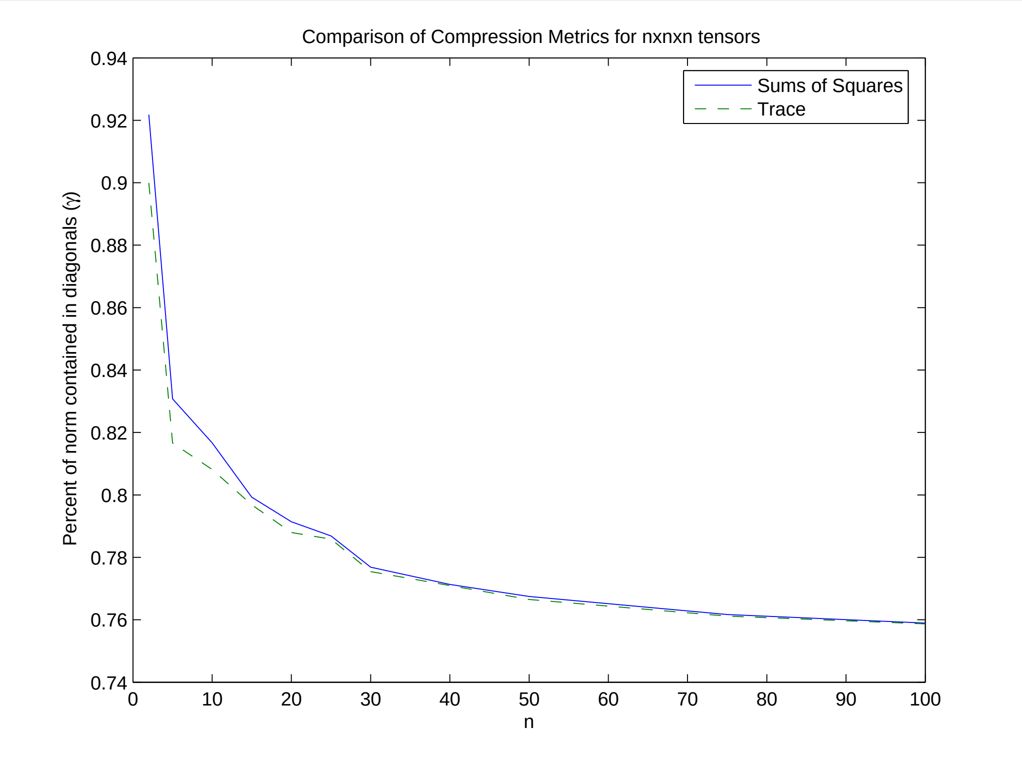
Works with (p, q) pairs of $2 \times 2 \times 2$ subtensors of $\mathcal{A} \in \mathbb{R}^{n \times n \times n}$:



Faster convergence if an *iteration* involves three sweeps with different cube orientations

`Jacobi-Compress` Compression Results

Compression for different n: $\left(\dfrac{\sum \sigma_{iii}^2}{\sum \sigma_{ijk}^2}\right)$



Conclusions

- Tensor decompositions used in analyzing multi-way data
- Applications need to drive the algorithms
- Tensor rank complicated, but provide insight by computing exactly for $n \times n \times 2$ tensors
- Can bypass the rank problem in some applications by "compressing" tensors
- Many open research problems and opportunities for contributions

Carla D. Moravitz Martin
Department of Mathematics, Cornell University
carlam@math.cornell.edu