Multi-Linear Algebraic Eigendecompositions and Their Application in Data Science

Randy C. Hoover[†], Kyle Caudle[*] and Karen S. Braman[*]

[†]Department of Electrical and Computer Engineering
[*]Department of Mathematics and Computer Science
South Dakota School of Mines and Technology
Rapid City, SD 57701, USA

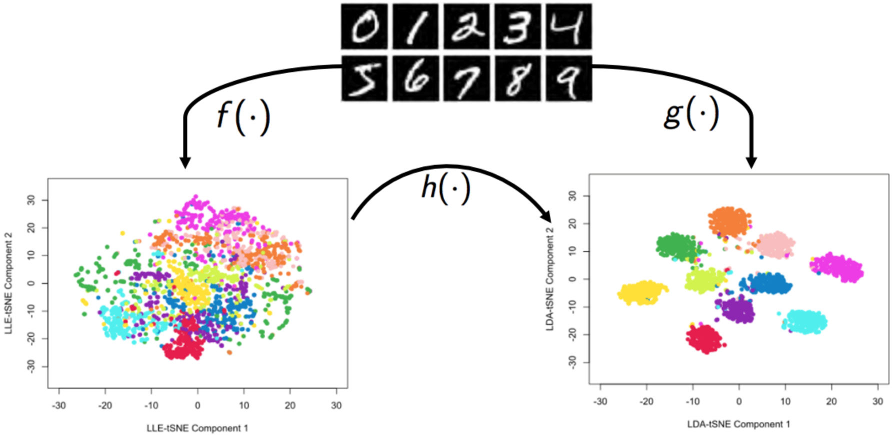
February 5, 2019

SOUTH DAKOTA SCHOOL OF MINES & TECHNOLOGY

Outline of today's talk

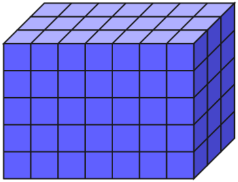
1. Motivational slide
2. Introduction to tensors and multi-linear subspace learning (MSL)
3. Mathematical foundations of tensors and their application to MSL
    1. Tensor-based Eigenvalue Decomposition
4. Analysis of MSL applied to image sequences
5. Multi-class classification via Multi-Linear Discriminant Analysis
6. Conclusions and open research directions

Data Analysis: Dimensionality Reduction, Clustering, and Classification (What are we trying to do?)



What do we mean by Tensor: *n*-mode or *n*-way array

$$\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_j} \quad \leftarrow \quad j\text{-th order tensor}$$



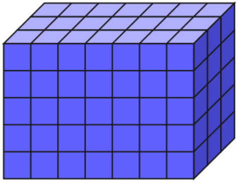
$\mathcal{A} \in \mathbb{R}^{5 \times 7 \times 3}$

Relation to Linear Algebra:

- Vectors are 1−mode tensors
- Matrices are 2−mode tensors

What do we mean by Tensor: n-mode or n-way array

$$\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_j} \quad \leftarrow \quad j\text{-th order tensor}$$



$\mathcal{A} \in \mathbb{R}^{5 \times 7 \times 3}$

Today's Talk:

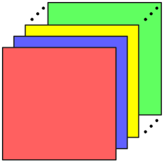
- Tensor products
- High order eigenvalue decomposition
- Tensor-based Discriminant Analysis (MLDA)
- Multi-class classification

Where do tensors arise?

Many application areas:

- Machine Learning/Data Analysis
  - Computer Vision
  - Pattern Recognition
  - Anomoly Detection
  - Multi-Modal Data Fusion
- Robotics
  - Sensor Fusion
  - Volumetric Range Data
  - 3-D Mapping
- Medical/Health
  - Computed tomography
  - Magnetic resonance
- Geospatial
  - Multi-spectral satellite imagery
- Control Theory
  - Swarm Dynamics
- etc......

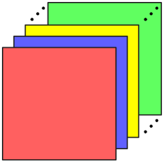
For $\mathcal{A} \in \mathbb{R}^{\ell \times m \times n}$,



$$\xrightarrow{\texttt{MatVec}} \begin{bmatrix} \mathcal{A}_1 \\ \mathcal{A}_2 \\ \mathcal{A}_3 \\ \vdots \\ \mathcal{A}_n \end{bmatrix} \in \mathbb{R}^{\ell n \times m}$$

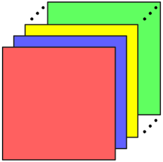
Mathematical foundations: MatVec and fold

For $\mathcal{A} \in \mathbb{R}^{\ell \times m \times n}$,



$$\xrightarrow{\texttt{MatVec}} \begin{bmatrix} \mathcal{A}_1 \\ \mathcal{A}_2 \\ \mathcal{A}_3 \\ \vdots \\ \mathcal{A}_n \end{bmatrix} \in \mathbb{R}^{\ell n \times m}$$

$$\begin{bmatrix} \mathcal{A}_1 \\ \mathcal{A}_2 \\ \mathcal{A}_3 \\ \vdots \\ \mathcal{A}_n \end{bmatrix} \xrightarrow{\texttt{fold}}$$



$$\in \mathbb{R}^{\ell \times m \times n}$$

Mathematical foundations: Block Circulant Matrices

For $\mathcal{A} \in \mathbb{R}^{\ell \times m \times n}$, define

$$\texttt{circ}\,(\mathcal{A}) \equiv \begin{bmatrix} \mathcal{A}_1 & \mathcal{A}_n & \mathcal{A}_{n-1} & \cdots & \mathcal{A}_2 \\ \mathcal{A}_2 & \mathcal{A}_1 & \mathcal{A}_n & \cdots & \mathcal{A}_3 \\ \mathcal{A}_3 & \mathcal{A}_2 & \mathcal{A}_1 & \cdots & \mathcal{A}_4 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathcal{A}_n & \mathcal{A}_{n-1} & \mathcal{A}_{n-2} & \cdots & \mathcal{A}_1 \end{bmatrix}$$

$$\texttt{circ}(\mathcal{A}) \in \mathbb{R}^{\ell n \times mn}$$

Mathematical foundations: Tensor Multiplication or t-product (Kilmer, Martin and Perrone)

Def: Tensor Product

Let $\mathcal{A}$ be $\ell \times p \times n$ and $\mathcal{B}$ be $p \times m \times n$. Then the t-product $\mathcal{A} * \mathcal{B}$ is the $\ell \times m \times n$ tensor

$$\mathcal{A} * \mathcal{B} = \texttt{fold}\left(\texttt{circ}(\mathcal{A}) \cdot \texttt{MatVec}(\mathcal{B})\right).$$

example

Suppose $\mathcal{A} \in \mathbb{R}^{\ell \times p \times 3}$ and $\mathcal{B} \in \mathbb{R}^{p \times m \times 3}$. Then

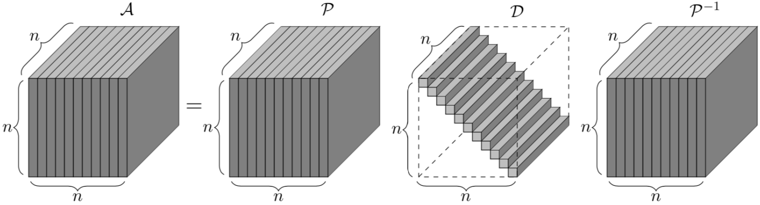
$$\mathcal{A} * \mathcal{B} = \texttt{fold}\left(\begin{bmatrix} \mathcal{A}_1 & \mathcal{A}_3 & \mathcal{A}_2 \\ \mathcal{A}_2 & \mathcal{A}_1 & \mathcal{A}_3 \\ \mathcal{A}_3 & \mathcal{A}_2 & \mathcal{A}_1 \end{bmatrix} \begin{bmatrix} \mathcal{B}_1 \\ \mathcal{B}_2 \\ \mathcal{B}_3 \end{bmatrix} \right) \in \mathbb{R}^{\ell \times m \times 3}.$$

Tensor-based Eigenvalue Decomposition (t-eig) - (Braman)

Def: Tensor Eigenvalue Decomposition

Given $\mathcal{A} \in \mathbb{R}^{n \times n \times n}$, there exists $n \times n \times n$ tensor $\mathcal{P}$ and an $n \times n \times n$ f-diagonal tensor $\mathcal{D}$ such that

$$\mathcal{A} = \mathcal{P} * \mathcal{D} * \mathcal{P}^{-1} \implies \mathcal{A} * \mathcal{P} = \mathcal{D} * \mathcal{P} \implies \mathcal{A} * \mathcal{P}_j = \mathcal{P}_j \mathbf{d}_j$$



Why is this useful?

Multi-class classification via Multi-Linear Discriminant Analysis (MLDA)

- Construct a multi-class data tensor:

$$\mathcal{X} = [\mathcal{X}_1, \mathcal{X}_2, \cdots, \mathcal{X}_C] \in \mathbb{R}^{n \times q \times n}$$

  - $\mathcal{X}_i$ is a tensor corresponding to class i
  - each lateral slice of $\mathcal{X}$ is an $n \times n$ "sample" (image)
  - q is the total number of samples
  - C is the number of distinct classes

Why is this useful?

Multi-class classification via Multi-Linear Discriminant Analysis (MLDA)

- Construct a multi-class data tensor:

$$\mathcal{X} = [\mathcal{X}_1, \mathcal{X}_2, \cdots, \mathcal{X}_C] \in \mathbb{R}^{n \times q \times n}$$

  - $\mathcal{X}_i$ is a tensor corresponding to class i
  - each lateral slice of $\mathcal{X}$ is an $n \times n$ "sample" (image)
  - q is the total number of samples
  - C is the number of distinct classes

- Goal:
  Maximize between-class separation/minimize within-class separation!

SOUTH DAKOTA SCHOOL OF MINES & TECHNOLOGY

Construct the within-class scatter tensor

- Account for the total "scatter/spread" within each class:

$$\mathcal{S}_W = \sum_{i=1}^{C} \sum_{\mathcal{X}_j \in c_i} (\mathcal{X}_j - \mathcal{M}_i) * (\mathcal{X}_j - \mathcal{M}_i)^T$$

where

- $\mathcal{X}_j \in \mathbb{R}^{n \times n}$ is the j^{th} lateral slice of $\mathcal{X}$
- $\mathcal{M}_i \in \mathbb{R}^{n \times 1 \times n} = \frac{1}{N_i} \sum_{\mathcal{X}_j \in c_i} \mathcal{X}_j$ is the mean tensor of class i
  - N_i is the total number of data samples in each class, i.e., $i = 1, 2, \ldots, C$

SOUTH DAKOTA SCHOOL OF MINES & TECHNOLOGY

Construct the between-class scatter tensor

- Account for the total “scatter/spread” across all classes:

$$\mathcal{S}_B = \sum_{i=1}^{C} N_i(\mathcal{M}_i - \mathcal{M}) * (\mathcal{M}_i - \mathcal{M})^T$$

where

  - $\mathcal{M}$ is the mean of all data samples

Optimal subspace computation

- The projection tensor $\mathcal{U}$ is computed by maximizing the ratio of "determinants" between $\mathcal{S}_W$ and $\mathcal{S}_B$ in the projection space:

$$\underset{\mathcal{U}}{\operatorname{argmax}} \frac{|\mathcal{U}^T * \mathcal{S}_B * \mathcal{U}|}{|\mathcal{U}^T * \mathcal{S}_W * \mathcal{U}|}$$

- Reformulating as a multi-linear Lagrangian, $\mathcal{U}$ can be computed by solving the generalized tensor eigenvalue problem

$$\mathcal{S}_B * \mathcal{U}_p = \boldsymbol{\lambda}_p * \mathcal{S}_W * \mathcal{U}_p$$

where

  - $\mathcal{U} = [\mathcal{U}_1, \mathcal{U}_2, \ldots, \mathcal{U}_p] \in \mathbb{R}^{n \times p \times n}$ are the eigenmatrices corresponding to the p largest eigentuples $\boldsymbol{\lambda}_p \in \mathbb{R}^{1 \times 1 \times n}$

Optimal subspace computation

- Advantages:
  1. Don't need to worry about the "small sample size" problem encountered in matrix LDA $\implies$ $\mathcal{S}_W$ is invertable!

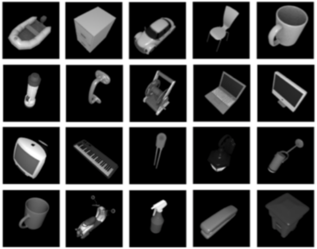
$$\left(\mathcal{S}_W^{-1} * \mathcal{S}_B\right) * \mathcal{U}_p = \boldsymbol{\lambda}_p * \mathcal{U}_p$$

  2. Accounts for the "temporal" AND spatial correlation in the image
- $\mathcal{U}_p$ can be computed using t-eig defined previously!
  - There are at most $C-1$ nonzero eigentuples $\implies$ the subspace has at most dimension $C-1$

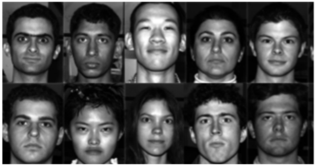
How well does it work?

- Three different data sets for experimental analysis:
  - MNIST: Handwritten Digits
  - Ray-Traced
    - 128 images captured as object rotates
  - Faces: Yale-B Extended (variation in illumination)

(a)



(b)



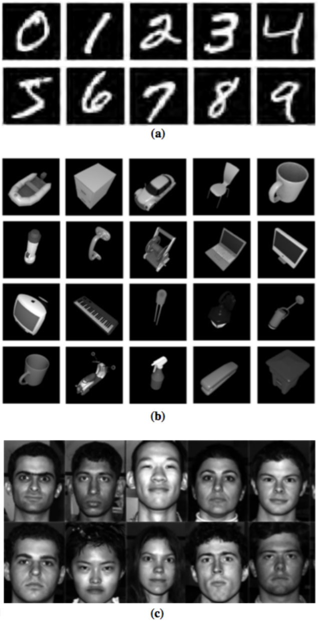
(c)

How well does it work?

	MNIST	Ray-Traced	Yale B
n	28	50	40
q	60,000	2560	1520
N_i	1000	128	40
C	10	20	38

Recall:

- n is the sample (image) size $n \times n$
- q is the total number of samples
- N_i is the number of samples in class i
- C is the number of total classes



How well does it work?

- Classification accuracy:
  - Subspace dimension is $p = C - 1$
  - Project new query image into the subspace and perform:
    - NN: Nearest neighbor search
    - NC: Nearest center search

	MNIST	Ray-Traced	extended Yale B
# of Test Samples	10,000	1280	722
LDA (NN):	85.1%	99.6%	56.8% (w/PCA: 97.8%)
LDA (NC):	84.7%	97.9%	55.2% (w/PCA: 96.2%)
MLDA (NN):	**93.2%**	**100%**	**95.7%**
MLDA (NC):	**91.4%**	**99.2%**	**94.8%**

Summary

- New approach to multi-class classification
  - Tensor Eigenvalue Decomposition
  - Multi-linear LDA (MLDA):
    - Multi-class classification
  - Experimental results:
    - Hand written digits (MNIST)
    - Objects (ray-traced)
    - Faces (Yale-B)
  - Outperforms traditional Matrix LDA
  - Overcomes "small sample size" issues

SOUTH DAKOTA
SCHOOL OF MINES
& TECHNOLOGY

Future Directions

- Compare current MLDA with other "tensor-based" approaches
  - CONDECOMP/PARAFAC
  - Tucker decompositions (HOSVD)
- Investigate multi-linear extension to other linear and non-linear subspace learning models:
  - CCA, ICA, LLE, ISOMAP, Compressed Sensing, etc.
- Hybrid machine learning methods:
  - Multi-linear subspace projections as a pre-cursor to decision trees, deep neural nets, long-short term memory networks, etc.

Thank you for your time!

Questions?