Multilinear Image Representation and Statistical Learning Models
http://virtual01.lncc.br/∼giraldi/Tutorial2016/

Tiene A. Filisbino

Gilson Antonio Giraldi

Carlos Eduardo Thomaz

Laboratório Nacional de Computação Científica-LNCC/MCTI

Centro Universitário da FEI

October 4th, 2016

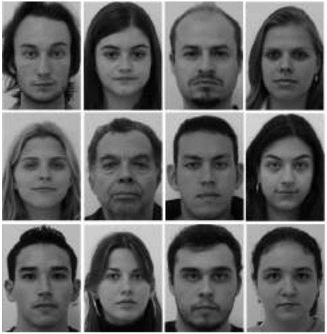
Topics

1. Introduction to Multilinear Data Representation
2. Dimensionality Reduction in Tensor Spaces.
    1. Generalized Matrix and Tensor Product Approaches
3. Subspace Learning in Tensor Spaces: General Problem
    1. Multilinear Principal Component Analysis (MPCA) and Variants
4. Discriminant Analysis and Recognition in Multilinear Spaces.
5. Application: FEI Database Analysis.
    1. Implementation Details
    2. Ranking and Understanding Tensor Components
    3. Recognition Rates in Gender Experiments
    4. Tensor Components and Reconstruction
6. Perspectives.
7. Conclusions and Final Remarks.

Introduction

Mathematical Representation and Analysis of Image Databases

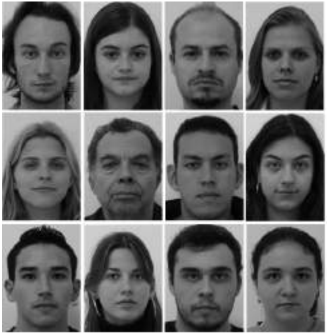
- Redundancy
- High-dimensional spaces



Introduction

Mathematical Representation and Analysis of Image Databases

- Redundancy
- High-dimensional spaces



- Linear Techniques:PCA
- Kernel Methods
- Multilinear Image Representation

Introduction

$\boxed{m - dimensional}$ $\boxed{m' - dimensional}$

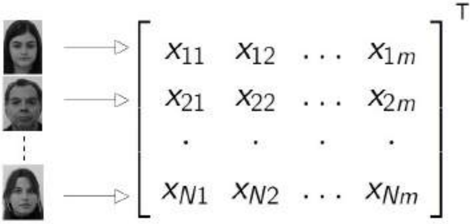
$$m' << m$$

$\Rightarrow$ Linear Dimensionality Reduction

$\Rightarrow$ Multilinear Dimensionality Reduction

Introduction

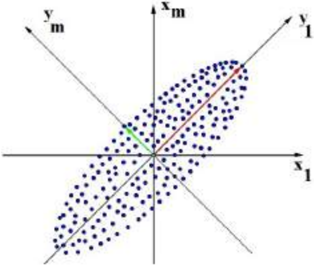
Linear Techniques:PCA

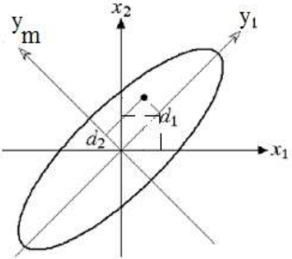


$$P_{pca} = \arg\min_{P} \sum_{i=1}^{N} ||P.I_{m'}.P^T.\widetilde{x}_i - \widetilde{x}_i||^2.$$

1. Consider a gray scale $m_1 \times m_2$ image as a high dimensional vector in $\Re^m$ space, where $m = m_1 \cdot m_2$.
2. $\overline{x} = \frac{1}{N} \sum_{i=1}^{N} x_i$.
3. $\widetilde{x}_i = x_i - \overline{x}$.
4. $S = \sum_{i=1}^{N} \widetilde{x}_i.\widetilde{x}_i^T$.
5. $P^T S P = \Lambda$.

Introduction
Details about PCA.





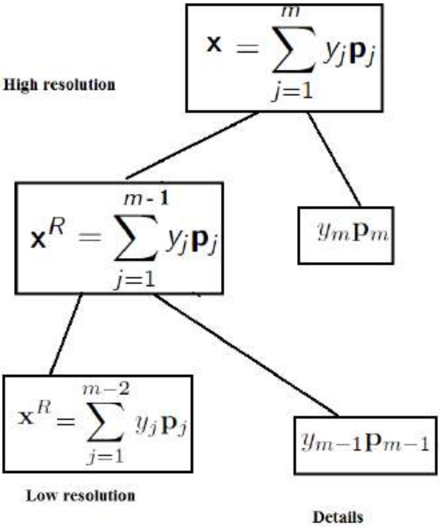
$$\mathbf{y} = P^T \mathbf{x},$$

$$PI_{m\prime}\mathbf{y} = PI_{m\prime}P^T\mathbf{x},$$

$$\mathbf{x}^R = PI_{m\prime}\mathbf{y} = \sum_{j=1}^{m\prime} y_j \mathbf{p}_j,$$

Image $I \in \mathbb{R}^{m_1 \times m_2}$ then:
$\mathbf{p}_j \in \mathbb{R}^{m_1 \cdot m_2}$.

Naive Multiresolution using PCA.



Limitations of PCA.

- Dimension of rows and column spaces in PCA system
- Dimensionality reduction means only truncate expression:

$$\mathbf{x} = \sum_{j=1}^{m} y_j \mathbf{p}_j,$$

- Dimension of covariance matrix S may be high
- Small sample size problem:
    - We do not need to decompose S
    - Limited number of dimensions

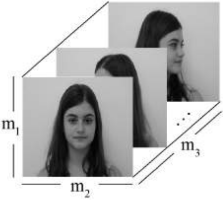
Multidimensional Image Data Representation

Figure: Samples from FEI database. http://fei.edu.br/∼cet/facedatabase.html

Introduction

Multiliner Techniques: Tensor Representation

- Generalization of linear technique to dimensionality reduction.
- Generalized matrix for data representation: named tensor.
- Multilinear techniques as:



Images in gray scale: third order tensor

1. *Multilinear Principal Components Analysis (MPCA)[Lu et al., 2008].*
2. *MPCA variants [Lu et al., 2009, Panagakis et al., 2010].*
3. *Concurrent Subspace Analysis (CSA) [Xu et al., 2008]*

Applications for Multiliner Techniques

- Face transfer: use one face to animate another one [Vlasic et al., 2005],
- Face recognition/reconstruction under multiple viewpoints [Filisbino et al., 2013b, Jia and Gong, 2005],
- Video content representation and retrieval [Zhou et al., 2012, Liu et al., 2008],
- Gait recognition [Lu et al., 2008],
- Visualization and computer graphics [Pajarola et al., 2013].

Applications: Face transfer: use one face to animate another one

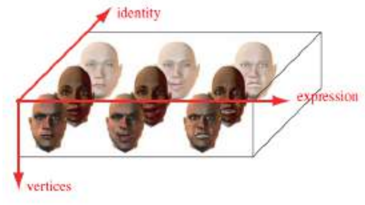


Figure: Pose, identity, expressions, visemes (speech-related mouth articulations). Source [Vlasic et al., 2005]

Applications: Face reconstruction under multiple viewpoints

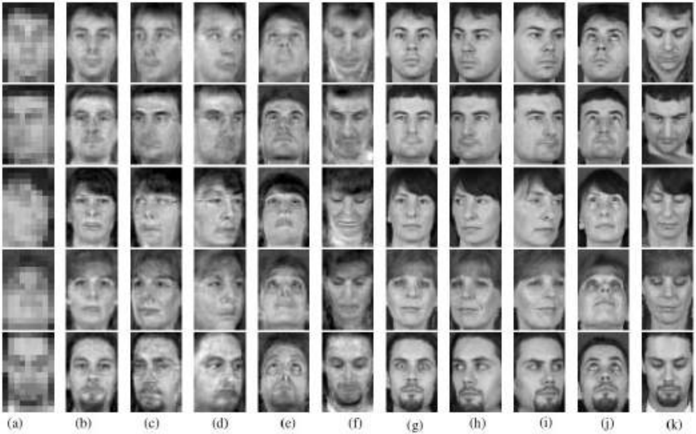


Figure: (a) Low-resolution 14 $\times$ 9. (b)-(f) Reconstruction in 56 $\times$ 36. (g)-(k) Ground truth. (source [Jia and Gong, 2005])

Applications: Video content representation and retrieval

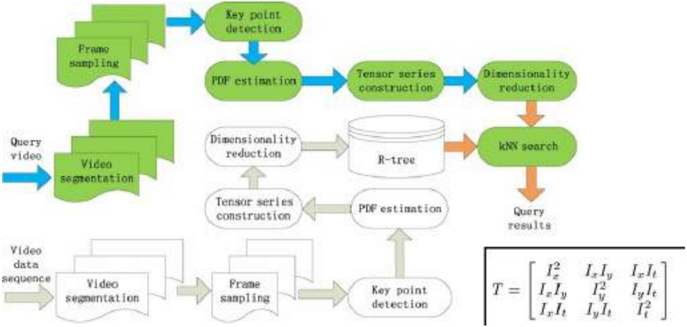


Figure: Find clips that are identical in content to a query (source [Zhou et al., 2012])

Applications: Gait Recognition

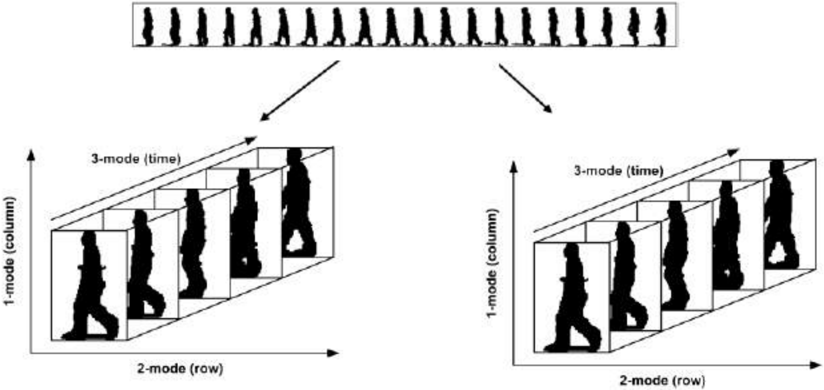


Figure: Third-order tensor representing a gait silhouette sequence (source [Lu et al., 2008]).

Applications: Multiresolution Volume Visualization

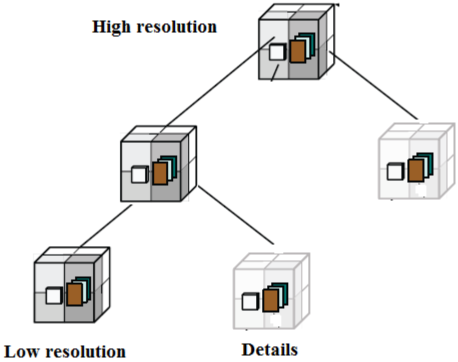


Figure: Multiscale tensor reconstruction for visualization (source [Suter et al., 2011])

Tensor Product Spaces

- Vector spaces V_1 and V_2; $dim\left(V_1\right) = n$ and $dim\left(V_2\right) = m$,
- Basis: $\{\mathbf{e}_1^1, \mathbf{e}_1^2, \mathbf{e}_1^3, ..., \mathbf{e}_1^n\}$ and $\{\mathbf{e}_2^1, \mathbf{e}_2^2, \mathbf{e}_2^3, ..., \mathbf{e}_2^m\}$,
- Tensor product $V_1 \otimes V_2$:

1. Dimension:

$$dim\left(V_1 \otimes V_2\right) = n.m, \tag{1}$$

2. Basis:

$$V_1 \otimes V_2$$
$$= span\left\{\mathbf{e}_1^i \otimes \mathbf{e}_2^j; 1 \leq i \leq n, 1 \leq j \leq m\right\}, \tag{2}$$

3. Given $\mathbf{v} = \sum_{i=1}^{n} v_i \mathbf{e}_1^i$ and $\mathbf{u} = \sum_{j=1}^{m} u_j \mathbf{e}_2^j$:

$$\mathbf{v} \otimes \mathbf{u} = \sum_{i=1}^{n} \sum_{j=1}^{m} v_i u_j \mathbf{e}_1^i \otimes \mathbf{e}_2^j. \tag{3}$$

Generalizing Tensor Product Spaces

- Given $V_1, V_2, \ldots, V_n$, with $dim\,(V_i) = m_i$, and $\{\mathbf{e}_i^1, \mathbf{e}_i^2, ..., \mathbf{e}_i^{m_i}\}$ a basis for V_i , then:

$$V_1 \otimes V_2 \otimes \ldots \otimes V_n$$
$$= span\left\{\mathbf{e}_1^{i_1} \otimes \mathbf{e}_2^{i_2} \otimes \ldots \otimes \mathbf{e}_n^{i_n};\ \mathbf{e}_k^{i_k} \in V_k\right\}, \qquad (4)$$

- A tensor $\mathbf{X}$ of order n is an element $\mathbf{X} \in V_1 \otimes V_2 \otimes \ldots \otimes V_n$:

$$\mathbf{X} = \sum_{i_1, i_2, \cdots, i_n} \mathbf{X}_{i_1, i_2, \cdots, i_n} \mathbf{e}_1^{i_1} \otimes \mathbf{e}_2^{i_2} \otimes \cdots \otimes \mathbf{e}_n^{i_n}. \qquad (5)$$

Changing Tensor Representation

- Let $V_i = \mathbb{R}^{m_i}$ and new basis:

$$\widetilde{B} = \left\{ \widetilde{\mathbf{e}}_1^{i_1} \otimes \widetilde{\mathbf{e}}_2^{i_2} \otimes \cdots \otimes \widetilde{\mathbf{e}}_n^{i_n}, \quad \widetilde{\mathbf{e}}_k^{i_k} \in \mathbb{R}^{m_k} \right\}, \tag{6}$$

- Basis change matrices $R^k \in \mathbb{R}^{m_k \times m_k}$, defined by:

$$\mathbf{e}_k^{i_k} = \sum_{j_k=1}^{m_k} R_{i_k j_k}^k \widetilde{\mathbf{e}}_k^{j_k}, \tag{7}$$

where $k = 1, 2, \cdots, n$ and $i_k = 1, 2, \cdots, m_k$.

- New tensor representation:

$$\mathbf{X} = \sum_{j_1, j_2, \cdots, j_n} \widetilde{\mathbf{X}}_{j_1, j_2, \cdots, j_n} \widetilde{\mathbf{e}}_1^{j_1} \otimes \widetilde{\mathbf{e}}_2^{j_2} \cdots \otimes \widetilde{\mathbf{e}}_n^{j_n}, \tag{8}$$

with:

$$\widetilde{\mathbf{X}}_{j_1, j_2, \cdots, j_n} = \sum_{i_1, i_2, \cdots, i_n} \mathbf{X}_{i_1, i_2, \cdots, i_n} R_{i_1 j_1}^1 R_{i_2 j_2}^2 \ldots . R_{i_n j_n}^n .$$

Dimensionality Reduction in Tensor Spaces

- Projection matrices $U^k \in \mathbb{R}^{m_k \times m\prime_k}$, as follows:

$$U^k_{i_k j_k} = R^k_{i_k j_k}, \quad i_k = 1, 2, \cdots, m_k; \quad j_k = 1, 2, \cdots, m\prime_k, \tag{9}$$

with $k = 1, 2, \cdots, n$, $m\prime_k \leq m_k$.

- Reduced representation:

$$\mathbf{Y} = \sum_{j_1, j_2, \cdots, j_n = 1}^{m\prime_1, \cdots, m\prime_n} \mathbf{Y}_{j_1, j_2, \cdots, j_n} \widetilde{\mathbf{e}}_1^{j_1} \otimes \widetilde{\mathbf{e}}_2^{j_2} \cdots \otimes \widetilde{\mathbf{e}}_n^{j_n}, \tag{10}$$

where:

$$\mathbf{Y}_{j_1, j_2, \cdots, j_n} = \sum_{i_1, i_2, \cdots, i_n = 1}^{m_1, \cdots, m_n} \mathbf{X}_{i_1, i_2, \cdots, i_n} U^1_{i_1 j_1} U^2_{i_2 j_2} \ldots U^n_{i_n j_n}. \tag{11}$$

Tensor Product Formalism

- Advantage: clarify tensor operations
- Disadvantage: uncomfortable expressions. Example:

$$\mathbf{Y} = \sum_{j_1, j_2, \cdots, j_n = 1}^{m\prime_1, \cdots, m\prime_n} \mathbf{Y}_{j_1, j_2, \cdots, j_n} \widetilde{\mathbf{e}}_1^{j_1} \otimes \widetilde{\mathbf{e}}_2^{j_2} \cdots \otimes \widetilde{\mathbf{e}}_n^{j_n}, \tag{12}$$

- Solution: Generalized matrix approach:

$$\mathbf{Y} = \left[\mathbf{Y}_{j_1, j_2, \cdots, j_n}\right]_{j_1, j_2, \cdots, j_n = 1}^{m\prime_1, \cdots, m\prime_n} \tag{13}$$

- Multilinear functions/algebra and generalized matrices

Generalized Matrix Approach

- A tensor **X** in expression (5) is just a generalized matrix $\mathbf{X} \in \mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n}$,
- The mode-k product of tensor $\mathbf{X} \in \mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n}$ with the matrix $A \in \mathbb{R}^{m\prime_k \times m_k}$ is given by:

$$(\mathbf{X} \times_k A)_{i_1,\ldots,i_{k-1},i,i_{k+1},\ldots,i_n}$$

$$= \sum_{j=1}^{m_k} \mathbf{X}_{i_1,\cdots,.i_{k-1},j,i_{k+1},\cdots i_n} A_{i,j}, \quad i = 1, 2, \ldots, m\prime_k. \qquad (14)$$

Generalized Matrix Approach

- Dimensionality Reduction: We can compute expression (11) as:

$$\mathbf{Y}_{j_1,j_2,\cdots,j_n} = \left(\mathbf{X} \times_1 U^{1^T} \times_2 U^{2^T} \ldots \times_n U^{n^T}\right)_{j_1,j_2,\cdots,j_n}, \tag{15}$$

  or, in a compact form [Filisbino et al., 2013a]:

$$\mathbf{Y} = \mathbf{X} \times_1 U^{1^T} \times_2 U^{2^T} \ldots \times_n U^{n^T}. \tag{16}$$

- Reconstruction:
  - $$\mathbf{X}^R = \mathbf{Y} \times_1 U^1 \ldots \times_n U^n, \tag{17}$$
  - $$\mathbf{X}^R = \mathbf{X} \times_1 U^1 U^{1^T} \ldots \times_n U^n U^{n^T}, \tag{18}$$

Multilinear Spaces: Definitions

$\Rightarrow$ A tensor of order n is just a generalized matrix $\mathbf{X} \in \mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n}$.

Then:

1. There is an isomorphism between $\mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n}$ and $\mathbb{R}^{m_1 \cdot m_2 \cdots m_n}$.
2. The internal product between two tensors $\mathbf{X} \in \mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n}$ and $\mathbf{Y} \in \mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n}$ is defined by:

$$\langle \mathbf{X}, \mathbf{Y} \rangle = \sum_{i_1=1,\ldots,i_n=1}^{m_1,\ldots,m_n} \mathbf{X}_{i_1,..,i_n} \mathbf{Y}_{i_1,..,i_n}$$

3. The Frobenius norm of a tensor is given by the expression: $\| \mathbf{X} \| = \sqrt{\langle \mathbf{X}, \mathbf{X} \rangle}$, and the distance between tensors $\mathbf{X}$ and $\mathbf{Y}$ is computed by:

$$D(\mathbf{X}, \mathbf{Y}) = \| \mathbf{X} - \mathbf{Y} \| .$$

Multilinear Spaces: Terminology

1. Tensor of order n;
2. Mode k
3. Tensor components

$$\widetilde{\mathbf{e}}_1^{j_1} \otimes \widetilde{\mathbf{e}}_2^{j_2} \cdots \otimes \widetilde{\mathbf{e}}_n^{j_n} \tag{19}$$

4. Core Tensor $\mathbf{Y}$:

$$\mathbf{X}^R = \mathbf{Y} \times_1 U^1 \ldots \times_n U^n, \tag{20}$$

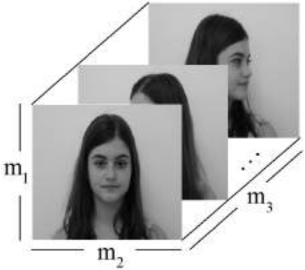
5. Tensor Fields
6. Multilinear versus Tensor
7. Component space: V_i in expression

$$V_1 \otimes V_2 \otimes \ldots \otimes V_n$$

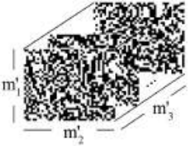
or $\mathbb{R}^{m_i}$ in expression

$$\mathbb{R}^{m_1} \otimes \mathbb{R}^{m_2} \otimes \ldots \otimes \mathbb{R}^{m_n} \equiv \mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n}$$

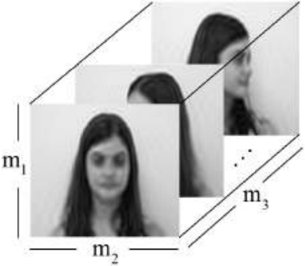
Multilinear Dimensionality Reduction



Original image: $\mathbb{R}^{m_1 \times m_2 \times m_3}$



Projected images: $\mathbb{R}^{m_1' \times m_2' \times m_3'}$



Reconstruction visualization: $\mathbb{R}^{m_1 \times m_2 \times m_3}$

Problem: Optimality criteria to seek for suitable matrices U^1, U^2,...,U^n.

Concurrent Subspace Analysis (CSA)

- Database: $D = \{\mathbf{X}_i \in \mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n},\ i = 1, 2, \ldots, N\}$
- Least square error minimization criterium:

$$(U^j|_{j=1}^n) = \arg \min_{U^j|_{j=1}^n} \sum_{i=1}^{N} ||\mathbf{X}_i \times_1 U^1 {U^1}^T \ldots \times_n U^n {U^n}^T - \mathbf{X}_i||^2$$

We can re-write CSA problem as follows:

$$P^{\star} = \arg\min_{P} \sum_{i=1}^{N} \left\| P P^T x_i^v - x_i^v \right\|^2, \quad (21)$$

$$\textit{subject to} :$$

$$P = U^n \otimes U^{n-1} \otimes \ldots \otimes U^1, \quad (22)$$

so, CSA is a constrained version of PCA [Filisbino et al., 2013a].

Analysis of Constrained PCA

- Simplified Version:

$$
\begin{aligned}
P^{\star} &= \arg\min_{P} \sum_{i=1}^{N} \left\| P I_m P^T x_i^v - x_i^v \right\|^2, \quad (23)\\
\textit{subject to} &: \\
P &= C \otimes D, \quad (24)
\end{aligned}
$$

- A simple manipulation shows that we must maximize [Filisbino et al., 2013a]:

$$
\widetilde{J_m} = Tr\left[I_m P^T R P \right],
$$

Analysis of Constrained PCA

- Lagrange multipliers:

$$\widetilde{J_m} = Tr\left[I_m \left(C \otimes D \right)^T R \left(C \otimes D \right) \right]$$

$$+ Tr\left[I_m \left(I - \left(C \otimes D \right)^T \left(C \otimes D \right) \right) M \right],$$

- Gateux derivative respect to the C [Filisbino et al., 2013a]:

$$\lim_{\tau \to 0} \frac{\widetilde{J_m}\left(C + \tau H \right) - \widetilde{J_m}\left(C \right)}{\tau} = 0,$$

- If $C \in \mathbb{R}^{s \times s}$:

$$N_{equations} = s^2 + s + 1,$$

equations against $2s^2$ variables

Mode-k Flattening and CSA

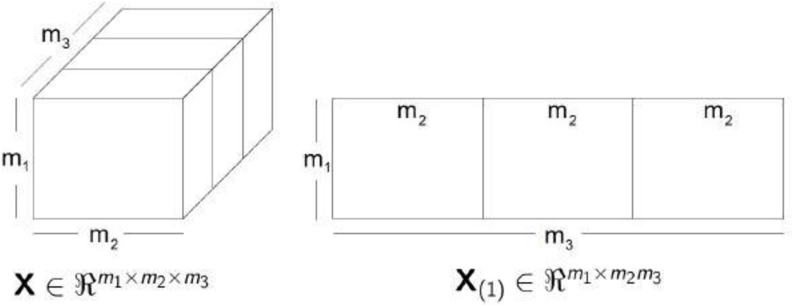


Figure: Mode-k Flattening to convert tensor into matrix

CSA Fundamental Theorem

Theorem

If $(U^1, ..., U^{k-1}, U^{k+1}, ..., U^n)$ *are known then the matrix* U^k *ideal is composed by the* $m\prime_k$ *principal eigenvectors of the covariance matrix*

$$C^{(k)} = \sum_{i=1}^{N} X_{i(k)}^{k} X_{i(k)}^{k^T},$$

where $X_{i(k)}^{k}$ *is the matrix generated through the mode-k flattening of the* $\mathbf{X}_i^k$*; that is:*

$$X_{i(k)}^{k} \Longleftarrow_k \mathbf{X}_i^k,$$

and

$$\mathbf{X}_i^k = \mathbf{X}_i \times_1 U^{1^T} ... \times_{k-1} U^{k-1^T} \times_{k+1} U^{k+1^T} ... \times_n U^{n^T}.$$

Multilinear Principal Component Analysis (MPCA)

- Let us consider a database:

$$D = \left\{ \mathbf{X}_i \in \mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n},\ i = 1, 2, \ldots, N \right\}. \tag{25}$$

- Variance maximization [Lu et al., 2008]:

$$\left(U^j|_{j=1}^n\right) = \arg \max_{U^j|_{j=1}^n} \frac{1}{N} \sum_{i=1}^{N} ||\mathbf{Y}_i - \overline{\mathbf{Y}}||^2, \tag{26}$$

where $\mathbf{Y}_i$ is given by expression (16) and $\overline{\mathbf{Y}}$ is the mean tensor computed by:

$$\overline{\mathbf{Y}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{Y}_i. \tag{27}$$

Multilinear Principal Component Analysis

- Theorem: Let $U^k \in \mathbb{R}^{m_k \times m\prime_k}$ $i = 1, 2, \ldots, n$, be the solution to (26). Then, given the projection matrices $U^1, ..., U^{k-1}, U^{k+1}, ..., U^n$, the matrix U^k consists of the $m\prime_k$ principal eigenvectors of the matrix:

$$\Phi^{(k)} = \sum_{i=1}^{N} (X_{i(k)} - \overline{X}_{(k)}) U_{\Phi^{(k)}} . U_{\Phi^{(k)}}^T . (X_{i(k)} - \overline{X}_{(k)})^T, \qquad (28)$$

where $X_{i(k)}$ and $\overline{X}_{(k)}$ are the mode-k flattening of sample tensor X_i and of the global mean $\overline{X}$, respectively, and:

$$U_{\Phi^{(k)}} = U^{k+1} \otimes U^{k+2} \otimes ... \otimes U^n \otimes U^1 \otimes U^2 \otimes ... \otimes U^{k-1}. \qquad (29)$$

Proof of MPCA Theorem [Lu et al., 2008]

Properties: $||\mathbf{X}|| = ||X_{(k)}||$, $||A||^2 = trace(AA^T)$ and, if $\mathbf{S} = \mathbf{X} \times_1 U^1 \times_2 U^2 ... \times_n U^n$ then $S_{(k)} = U^k X_{(k)} (U_{\Phi^{(k)}})^T$, where $U_{\Phi^{(k)}}$ is given by (29),

$$\Psi = \sum_{i=1}^{N} ||\mathbf{Y}_i - \overline{\mathbf{Y}}||^2 = \sum_{i=1}^{N} || \left(\mathbf{X}_i - \overline{\mathbf{X}}\right) \times_1 U^{1^T} \times_2 U^{2^T} ... \times_n U^{n^T} ||^2$$

$$= \sum_{i=1}^{N} ||U^{k^T} \left(X_{i(k)} - \overline{X}_{(k)}\right) U_{\Phi^{(k)}}||^2$$

Proof of MPCA Theorem (Continue)

$$= \sum_{i=1}^{N} trace\left({U^{k}}^{T} \left(X_{i(k)} - \overline{X}_{(k)}\right) U_{\Phi^{(k)}} \cdot U_{\Phi^{(k)}}^{T} \left(X_{i(k)} - \overline{X}_{(k)}\right)^{T} U^{k} \right)$$

$$trace\left({U^{k}}^{T} \sum_{i=1}^{N} \left[\left(X_{i(k)} - \overline{X}_{(k)}\right) U_{\Phi^{(k)}} \cdot U_{\Phi^{(k)}}^{T} \left(X_{i(k)} - \overline{X}_{(k)}\right)^{T} \right] U^{k} \right)$$

$$= trace\left({U^{k}}^{T} \Phi^{(k)} U^{k} \right)$$

$$\Phi^{(k)} = \sum_{i=1}^{N} (X_{i(k)} - \overline{X}_{(k)}) U_{\Phi^{(k)}} . U_{\Phi^{(k)}}^{T} . (X_{i(k)} - \overline{X}_{(k)})^{T},$$

CSA/MPCA Equivalence

Theorem

If CSA receives centered input samples and $m_k = m\prime_k$, $k = 1, 2, \cdots, n$, then the obtained projection matrices $U^1, ..., U^n$ are equal to the ones generated by MPCA.

Proof. [Filisbino et al., 2015]

MPCA and Traditional PCA

Considering tensors of order $n = 1$. Input samples are vectors $x_i \in \mathbb{R}^{m_1}$, $i = 1, 2, \ldots, N$ and, in this case, we have only one projection matrix $U^1 \in \mathbb{R}^{m_1 \times m\prime_1}$. If we consider full projection $(m\prime_1 = m_1)$ then U_1 is an orthogonal matrix. Also, $\widetilde{y}_i = \widetilde{x}_i \times_1 U^1 = {U^1}^T \widetilde{x}_i$, $\widetilde{X}_{i(k)} \to \widetilde{x}_i$ once we have just one mode $k = 1$ in this case. Therefore, the matrix $\Phi^{(k)}$ assumes the form:

$$\Phi^{(k)} = \Phi^{(1)} = \sum_{i=1}^{N} \widetilde{x}_i \widetilde{x}_i^T = \sum_{i=1}^{N} (x_i - \overline{x})(x_i - \overline{x})^T ,$$

which is the covariance matrix S of the PCA.

Tensor versus Linear Subspace Learning

- PCA.
  - Vectorize the samples to get vectors $v \in \mathbb{R}^{m_1 \cdot m_2 \dots m_n}$
  - Covariance matrix $C \in \mathbb{R}^{m_1 \cdot m_2 \dots m_n \times m_1 \cdot m_2 \dots m_n}$
- Small sample size problems: $N \ll m_1 \cdot m_2 \dots m_n$
  - PCA:
  - Number of PCA components is $(N - 1)$ or less
  - Efficient methods for PCA computation
- MPCA:
- Covariance matrices $C^k \in \mathbb{R}^{m_k \times m_k}$
- In general $N \cdot \amalg_{i \neq k} m_i \gg m_k$
- Flattening operation generates a number of $N \cdot \amalg_{i \neq k} m_i$ samples for $\Phi^{(k)}$ computation

Subspace Learning: Uncorrelated MPCA

Let the tensor-to-vector decomposition:

$$\mathbf{Y}_i = \sum_{p=1}^{P} y_{i_p} \widetilde{\mathbf{e}}_p^1 \otimes \widetilde{\mathbf{e}}_p^2 \cdots \otimes \widetilde{\mathbf{e}}_p^n, \quad P < \prod_{k=1}^{n} m_k \tag{30}$$

Let:

$$\overline{y}_p = \frac{1}{N} \sum_i y_{i_p}, \quad \mathbf{g}_p = \begin{pmatrix} y_{1_p} & y_{2_p} & \ldots & y_{N_p} \end{pmatrix}^T.$$

Then:

$$\left(\widetilde{\mathbf{e}}_p^1, \widetilde{\mathbf{e}}_p^2, \ldots, \widetilde{\mathbf{e}}_p^n\right) = \arg\max \sum_{i=1}^{N} \left(y_{i_p} - \overline{y}_p\right)^2, \tag{31}$$

subject to:

$$\widetilde{\mathbf{e}}_p^j \cdot \widetilde{\mathbf{e}}_p^j = 1, \quad and \quad \frac{\mathbf{g}_p \cdot \mathbf{g}_q}{||\mathbf{g}_p||.||\mathbf{g}_q||} = \delta_{pq}.$$

Subspace Learning: Non-Negative MPCA

Problem

$$\left(U^j\big|_{j=1}^n\right) = \arg\max_{U^j|_{j=1}^n} \frac{1}{N}\sum_{i=1}^{N} ||\mathbf{Y}_i - \overline{\mathbf{Y}}||^2, \tag{32}$$

subject to $U^j \geq 0$.
Solution [Panagakis et al., 2010]:

$$\left(U^j\big|_{j=1}^n\right) = \arg\max_{U^j|_{j=1}^n} \frac{1}{N}\sum_{i=1}^{N} ||\mathbf{Y}_i - \overline{\mathbf{Y}}||^2, \tag{33}$$

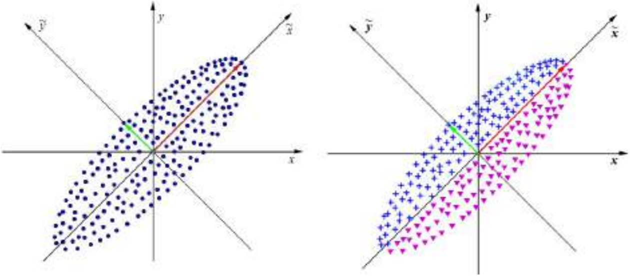
subject to:

$$U^j \in Gr\left(m_j, m\prime_j\right) \quad and \quad U^j \geq 0,$$

where $Gr\left(m_j, m\prime_j\right)$ is the set of $m\prime_j$-dimensional linear subspaces of $\mathbb{R}^{m_j}$, termed the Grassmann manifold.

Discriminant Analysis and Statistical Learning Approaches

Dimensionality Reduction × Discriminant Analysis



(a) (b)

Figure: (a) Scatter plot and principal directions. (b) The same population but distinguishing patterns plus (+) and triangle (▼).

Discriminant Analysis and Classification

⇒ Find discriminate directions to separate sample groups
⇒ Classification approaches.

Supervised statistical learning methods like:

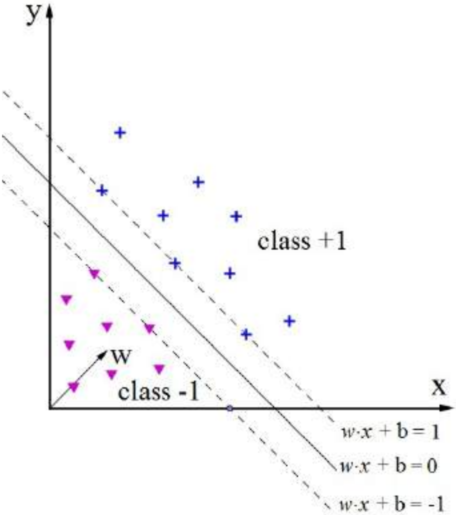
- Support Vector Machine (SVM)
- Discriminant Analsysis:

Linear: Linear Discriminant Analysis (LDA).
Multilinear: Fisher criterion [Lu et al., 2008].

Support Vector Machine

SVM

- $f(x) = (x \cdot w^{svm}) + b = 0$
- $w^{svm} = \sum_{i=1}^{N} \alpha_i y_i x_i$



Linear Discriminant Analysis

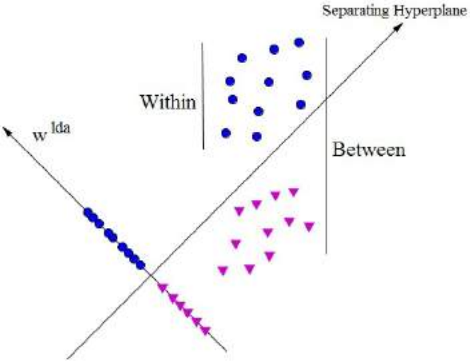
LDA

- $W^{lda} = \arg\max_{W} \frac{|W^T S_b W|}{|W^T S_w W|}$.
- S_b is the between-class matrix.
- S_w is the within-class matrix.
- W_{lda} is eigenvectors of $S_w^{-1} S_b$.

$\Rightarrow S_b = \sum_{i=1}^{g} N_i(\overline{x}_i - \overline{x})(\overline{x}_i - \overline{x})^T$

$\Rightarrow S_w = \sum_{i=1}^{g} \sum_{j=1}^{N_i} (x_{i,j} - \overline{x}_i)(x_{i,j} - \overline{x}_i)^T$

Linear Discriminant Analysis



Ranking Tensor Components

- Total scatter tensor
- Estimating Spectral Structure of Data
- Tensor Discriminant Principal Component Analysis
- Fisher Discriminability Criterion

Estimating Variances

- There is no a closed-form solution for subspace learning problems in tensor spaces.
- Total scatter tensor defined by [Lu et al., 2008]:

$$\mathbf{\Psi}_{j_1,j_2,\cdots,j_n} = \sum_{i=1}^{N} \frac{\left(\mathbf{Y}_{i;j_1,j_2,\cdots,j_n} - \overline{\mathbf{Y}}_{j_1,j_2,\cdots,j_n}\right)^2}{N}, \tag{34}$$

- Rank the tensor components by sorting:

$$E = \left\{\mathbf{\Psi}_{j_1,j_2,\cdots,j_n},\ j_k = 1, 2, \cdots, m\prime_k\right\}. \tag{35}$$

Spectral Structure of MPCA/CSA Subspaces

- Each component subspace:

$$\left\{\widetilde{\mathbf{e}}_k^{j_k}, \quad j_k = 1, 2, \cdots, m\prime_k\right\}, \quad k = 1, 2, \cdots, n,$$

has associated eigenvalues:

$$\left\{\lambda_{j_k}^{k}, \quad j_k = 1, 2, \cdots, m\prime_k\right\}, \quad k = 1, 2, \cdots, n,$$

- The data distribution in each subspace:

$$\mathbf{v}_k = \sum_{j_k=1}^{m\prime_k} \lambda_{j_k}^{k} \widetilde{\mathbf{e}}_k^{j_k}, \quad k = 1, 2, \cdots, n,$$

.

Spectral Structure of MPCA Subspaces

- Variance explained by the element of basis $\widetilde{B}$:

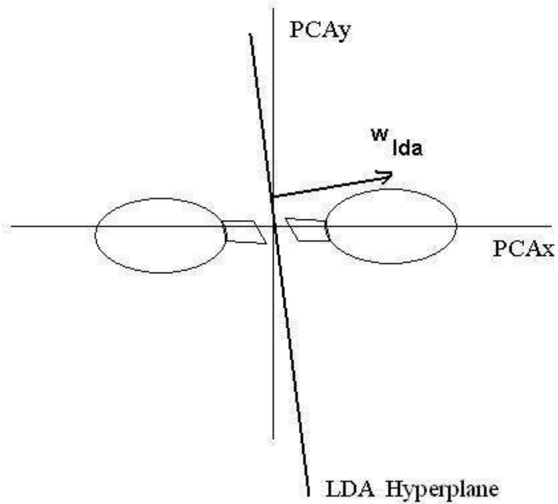
$$\mathbf{v}_1 \otimes \mathbf{v}_2 \otimes \cdots \otimes \mathbf{v}_n$$

$$= \sum_{j_1, j_2, \cdots, j_n} \lambda_{j_1}^1 \lambda_{j_2}^2 \cdots \lambda_{j_n}^n \widetilde{\mathbf{e}}_1^{j_1} \otimes \widetilde{\mathbf{e}}_2^{j_2} \otimes \cdots \otimes \widetilde{\mathbf{e}}_n^{j_n}. \qquad (36)$$

- Consequently, we can rank the MPCA/CSA tensor components by sorting:

$$E = \left\{ \lambda_{j_1, j_2, \ldots, j_n} = \lambda_{j_1}^1 \lambda_{j_2}^2 \cdots \lambda_{j_n}^n, \; j_k = 1, 2, \cdots, m\prime_k \right\}. \qquad (37)$$

Geometry Behind Discriminant Principal Components [Thomaz and Giraldi, 2010]



Tensor Discriminant Principal Components - TDPCA

Tensor discriminant principal components analysis (TDPCA) [Thomaz and Giraldi, 2010, Filisbino et al., 2015].

Steps:

1. $\{(\mathbf{X}_i, l_i); \mathbf{X}_i \in \mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n}, l_i \in \{-1, 1\}, i = 1, 2, ..., N\}$
2. Dimensionality reduction using the MPCA/CSA subspaces: $\mathbf{Y}_i \in \mathbb{R}^{m\prime_1 \times m\prime_2 \times \ldots \times m\prime_n}$.
3. Linear classifier is estimated using $\mathbf{Y_i}$ and labels.

Separating hyperplane is defined through a discriminant tensor $\mathbf{W} \in \mathbb{R}^{m\prime_1 \times m\prime_2 \times \ldots \times m\prime_n}$

We select the first principal MPCA/CSA components the ones with the highest discriminant weights, that is, $P_{classifier} = [p_1, p_2, ..., p_m]$, corresponding to the largest discriminant weights $|w_1| \geq |w_2| \geq \ldots \geq |w_m|$

Fisher criterion [Lu et al., 2008]

$$W^{Fisher}_{j_1,j_2,\cdots,j_n} = \frac{\sum_{c=1}^{C} N_c \cdot \left(\overline{\mathbf{Y}}_{c;j_1,j_2,\cdots,j_n} - \overline{\mathbf{Y}}_{j_1,j_2,\cdots,j_n}\right)^2}{\sum_{i=1}^{N} \left(\mathbf{Y}_{i;j_1,j_2,\cdots,j_n} - \overline{\mathbf{Y}}_{c_i;j_1,j_2,\cdots,j_n}\right)^2} \quad (38)$$

where,

- C is the number of classes;
- N_c is the number of elements of class c;
- $\overline{\mathbf{Y}}_c$ is the average tensor of the samples belonging to class c;
- $\overline{\mathbf{Y}}$ is the average tensor of all the samples;
- $\overline{\mathbf{Y}}_{c_i}$ is the average of the class corresponding to the *ith* tensor.

Justification of Fisher criterion for Tensors [Filisbino et al., 2015]

In [Yan et al., 2005] Fisher criterion is implemented by:

$$\left(U^j|_{j=1}^n\right) = \arg\max_{U^j|_{j=1}^n} \frac{\sum_{c=1}^{C} N_c \cdot ||\overline{\mathbf{X}}_c \times_1 U^1 \ldots \times_n U^n - \overline{\mathbf{X}} \times_1 U^1 \ldots \times_n U^n||^2}{\sum_{i=1}^{N} ||\mathbf{X}_i \times_1 U^1 \ldots \times_n U^n - \overline{\mathbf{X}}_{c_i} \times_1 U^1 \ldots \times_n U^n||^2}, \tag{39}$$

We can rewrite expression (39) as:

$$\left(U^j|_{j=1}^n\right) = \arg\max_{U^j|_{j=1}^n} \frac{\sum_{c=1}^{C} N_c \cdot ||\sum_{j_1,j_2,\cdots,j_n} \left(\overline{\mathbf{Y}}_{c;j_1,j_2,\cdots,j_n} - \overline{\mathbf{Y}}_{j_1,j_2,\cdots,j_n}\right) \widetilde{\mathbf{e}}_1^{j_1} \otimes \widetilde{\mathbf{e}}_2^{j_2} \cdots \otimes \widetilde{\mathbf{e}}_n^{j_n}||^2}{\sum_{i=1}^{N} ||\sum_{j_1,j_2,\cdots,j_n} \left(\mathbf{Y}_{i;j_1,j_2,\cdots,j_n} - \overline{\mathbf{Y}}_{c_i;j_1,j_2,\cdots,j_n}\right) \widetilde{\mathbf{e}}_1^{j_1} \otimes \widetilde{\mathbf{e}}_2^{j_2} \cdots \otimes \widetilde{\mathbf{e}}_n^{j_n}||^2}, \tag{40}$$

Justification for Fisher criterion for Tensors

By considering Frobenius norm:

$$\left(U^j\big|_{j=1}^{n}\right) = \arg\max_{U^j|_{j=1}^{n}} \frac{\sum_{j_1,j_2,\cdots,j_n}\left(\sum_{c=1}^{C} N_c \cdot \left(\overline{\mathbf{Y}}_{c;j_1,j_2,\cdots,j_n} - \overline{\mathbf{Y}}_{j_1,j_2,\cdots,j_n}\right)^2\right)}{\sum_{j_1,j_2,\cdots,j_n}\left(\sum_{i=1}^{N}\left(\mathbf{Y}_{i;j_1,j_2,\cdots,j_n} - \overline{\mathbf{Y}}_{c_i;j_1,j_2,\cdots,j_n}\right)^2\right)}, \tag{41}$$

We can postulate that the larger is the value of $\Gamma_{j_1,j_2,\cdots,j_n}$ computed by:

$$W^{Fisher}_{j_1,j_2,\cdots,j_n} = \frac{\sum_{c=1}^{C} N_c \cdot \left(\overline{\mathbf{Y}}_{c;j_1,j_2,\cdots,j_n} - \overline{\mathbf{Y}}_{j_1,j_2,\cdots,j_n}\right)^2}{\sum_{i=1}^{N}\left(\mathbf{Y}_{i;j_1,j_2,\cdots,j_n} - \overline{\mathbf{Y}}_{c_i;j_1,j_2,\cdots,j_n}\right)^2}, \tag{42}$$

then more discriminant is the tensor component $\widetilde{\mathbf{e}}_1^{j_1} \otimes \widetilde{\mathbf{e}}_2^{j_2} \cdots \otimes \widetilde{\mathbf{e}}_n^{j_n}$ for samples classification.

Experimental Results

Faces Database

Fei Database:

Figure: FEI face database sample. http://fei.edu.br/∼cet/facedatabase.html

Experimental Results: Ranking MPCA/CSA Components.

Gender Experiments

$$\text{Database: } (\mathbf{X}_i, l_i) \in \mathbb{R}^{480\times 640\times 11}, l_i \in \{-1, 1\}$$

↓

$$U^1, U^2, U^3$$

↓

$$\text{Base } \widetilde{B} : S_1 = \mathbb{R}^{479\times 639\times 11}; S_2 = \mathbb{R}^{33\times 42\times 9}$$

↓

Ranking

↓

$$\widehat{B}_1, \widehat{B}_2, \ldots, \widehat{B}_{2000}; \widehat{B}_j \subset \widetilde{B}$$

↓

$$\widehat{Y}_{t,1}, \widehat{Y}_{t,2}, \ldots, \widehat{Y}_{t,2000}$$

Implementation Details

System

- Processor: Intel-Core-I7
- Cores: 6 - Threads: 12
- RAM: $12GB$
- $S_1 = \mathbb{R}^{479\times639\times11} > 12GB \Rightarrow \text{Time} = 3h \Rightarrow T_{max} = 3$
- $S_2 = \mathbb{R}^{33\times42\times9} = 7GB \Rightarrow \text{Time} = 5' \Rightarrow T_{max} = 4$

Implementation Details

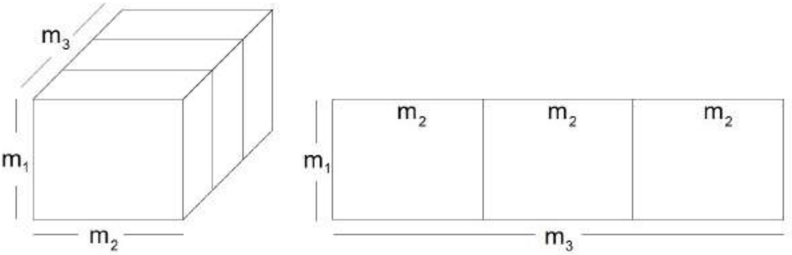
- Install the Matlab Tensor Toolbox.
  - http://www.sandia.gov/ tgkolda/TensorToolbox/

Principal Functions

- *tensor*(X): transform in tensor the generalize matrix X.
- *tenmat*(X, n): flattening in dimension n.
- *ttm*(X, U, n): mode-k product.

Examples

- $tenmat(\mathbf{X}, 1)$

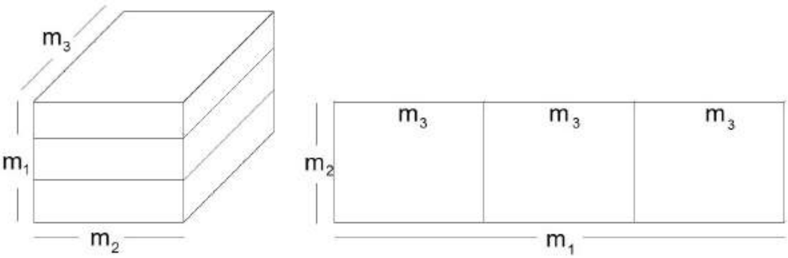


$\mathbf{X} \in \Re^{m_1 \times m_2 \times m_3}$ $\mathbf{X}_{(1)} \in \Re^{m_1 \times m_2 m_3}$

$$X_{(1)} = \left[\begin{array}{ccc|ccc} x_{(111)} & x_{(121)} & x_{(131)} & x_{(112)} & x_{(122)} & x_{(132)} \\ x_{(211)} & x_{(221)} & x_{(231)} & x_{(212)} & x_{(222)} & x_{(232)} \\ x_{(311)} & x_{(321)} & x_{(331)} & x_{(312)} & x_{(322)} & x_{(332)} \end{array} \right]$$

Examples

- *tenmat*$(\mathbf{X}, 2)$

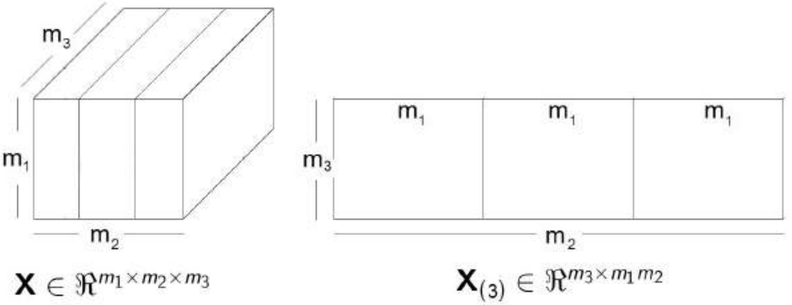


$\mathbf{X} \in \Re^{m_1 \times m_2 \times m_3}$ $\quad$ $\mathbf{X}_{(2)} \in \Re^{m_2 \times m_1 m_3}$

$$X_{(2)} = \left[\begin{array}{ccc|ccc} x(111) & x(211) & x(311) & x(112) & x(212) & x(312) \\ x(121) & x(221) & x(321) & x(122) & x(222) & x(322) \\ x(131) & x(231) & x331) & x(132) & x(232) & x(332) \end{array} \right]$$

Examples

- $tenmat(\mathbf{X}, 3)$



$$X_{(2)} = \left[\begin{array}{ccc|ccc|ccc} x_{(111)} & x_{(211)} & x_{(311)} & x_{(121)} & x_{(211)} & x_{(321)} & x_{(131)} & x_{(231)} & x_{(331)} \\ x_{(112)} & x_{(222)} & x_{(322)} & x_{(122)} & x_{(222)} & x_{(322)} & x_{(132)} & x_{(232)} & x_{(332)} \end{array} \right]$$

Examples

Let the Tensor $\mathbf{X} \in \mathbb{R}^{m_1 \times m_2 \times m_3}$ and the matrix $U \in \mathbb{R}^{m\prime_k \times m_k}$.
The function,

- $ttm(\mathbf{X}, U, n) = (\mathbf{X} \times_n A)$

$$\Rightarrow ttm(\mathbf{X}, U, 1) = (\mathbf{X} \times_1 U)_{m_1', m_2, m_3} = \sum_{j=1}^{m_1} \mathbf{X}_{j, m_2, m_3} . U_{m_1', j}$$

$$\Rightarrow ttm(\mathbf{X}, U, 2) = (\mathbf{X} \times_2 U)_{m_1, m_2', m_3} = \sum_{j=1}^{m_2} \mathbf{X}_{m_2, j, m_3} . U_{m_2', j}$$

$$\Rightarrow ttm(\mathbf{X}, U, 3) = (\mathbf{X} \times_3 U)_{m_1, m_2, m_3'} = \sum_{j=1}^{m_3} \mathbf{X}_{m_1, m_2, j} . U_{m_3', j}$$

Matricized version

$$\mathbf{Y} = (\mathbf{X} \times_n U) \Longleftrightarrow Y_{(n)} = UX_{(n)}$$

Input: Samples $\left\{ \mathbf{X}_i \in \mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n},\ i = 1, \ldots, N \right\}$; $\boxed{tensor(\mathbf{X}_i)}$

Preprocessing: Center the input samples as $\{ \widetilde{\mathbf{X}}_i = \mathbf{X}_i - \overline{\mathbf{X}} \}$

Initialization: Eigen-decomposition of $\Phi^{(k)*} = \sum_{i=1}^{N} \widetilde{\mathbf{X}}_{i(k)} \cdot \widetilde{\mathbf{X}}_{i(k)}^{T}$, set U_0^k as the most significant $m_k^{'}$ eigenvectors , for $k = 1, \ldots, n$. $\boxed{\widetilde{\mathbf{X}}_{i(k)} \longleftarrow tenmat(\widetilde{\mathbf{X}}_i, k)}$

Local optimization:

Compute $\widetilde{\mathbf{Y}}_i = \widetilde{\mathbf{X}}_i \times_1 U_0^{1^T} \ldots \times_n U_0^{n^T}, i = 1, \ldots, N$; $\boxed{\mathbf{Y} = ttm(\widetilde{\mathbf{X}}, Us, 1..n)}$

Compute $\Upsilon_0 = \sum_{i=1}^{N} ||\widetilde{\mathbf{Y}}_i||_F^2$;

for $t = 1, \ldots$ to T_{max} **do**

  for $k = 1, \ldots$ to n **do**

    Set the matrix U_t^k to consist of the $m_k^{'}$ leading eigenvectors of $\Phi^{(k)}$, defined in expression (28);

$$\boxed{U_t^k \longleftarrow \Phi^k = A_{(k)} \cdot A_{(k)}^T \longleftarrow tenmat(\mathbf{A}^k, k) = ttm(\widetilde{\mathbf{X}}, Us, -k)}$$

  end for

  Compute $\widetilde{\mathbf{Y}}_i, i = 1, \ldots, N$ and Υ_t;

  if $|\Upsilon_t - \Upsilon_{t-1}| < \eta$ **then**

    break; $\boxed{\eta = 0.001}$

  end if

end for

Output: Projection matrices $U^k = U_t^k, k = 1, \ldots, n$.

Estimating MPCA Subspaces Dimensions

The reduced dimensions $m_k^{'}$, $k = 1, 2, \ldots, n$ must be specified in advance or determined by some heuristic. In [Lu et al., 2008] it is proposed to compute these values in order to satisfy the criterium:

$$\frac{\sum_{i_k=1}^{m_k'} \lambda_{i_k(k)}}{\sum_{i_k=1}^{m_k} \lambda_{i_k(k)}} > \Omega \tag{43}$$

where Ω is a threshold to be specified by the user and $\lambda_{i_k(k)}$ is the $i_k th$ eigenvalue of $\Phi^{(k)*}$ ($\Omega = 0.95$).

CSA Algorihm

Input: Samples $\{\mathbf{X}_i \in \mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n},\ i = 1, ..., N\}$; dimensions $m_k^{'}$; $k = 1, \cdots, n$.
Initialization of U_0^k by truncating the number of columns of the identity matrix;
for $t = 1, ...$ to T_{max} **do**
    for $k = 1, ...$ to n **do**
        Mode-k tensor products:
        $\mathbf{X}_i^k = \mathbf{X}_i \times_1 {U_t^1}^T \ldots \times_{k-1} {U_t^{(k-1)}}^T \times_{k+1} {U_{t-1}^{(k+1)}}^T \ldots \times_n {U_{t-1}^n}^T$
        Mode-k flattening:
        $\mathbf{X}_i^k$ for the matrix X_i^k: $X_i^k \Longleftarrow_k \mathbf{X}_i^k$
        Covariance matrix computation:
        $C^k : C^k = \sum_{i=1}^N X_i^k {X_i^k}^T$
        Compute the first $m_k^{'}$ leading eigenvectors of C^k,
        $C^k U_k^t = U_k^t \Lambda^k$, which constitute the column vectors of U_k^t
    end for
    if $(t > 2$ and $Tr[abs({U_t^k}^T U_{t-1}^k)]/m_k^{'} > (1 - \epsilon), k = 1, ..., n)$ **then**
        break; $\boxed{\epsilon = 0.001}$
    end if
end for
Output the matrices $U_k = U_k^t, k = 1, ..., n$.

Review: Ranking Tensor Components

- Statistical Variance

$$E = \left\{ \mathbf{\Psi}_{j_1, j_2, \cdots, j_n}, \; j_k = 1, 2, \cdots, m\prime_k \right\}. \qquad (44)$$

- Spectral Structure

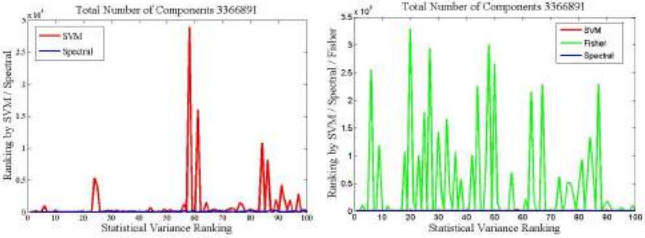
$$E = \left\{ \lambda_{j_1, j_2, \ldots, j_n} = \lambda^1_{j_1} \lambda^2_{j_2} \cdots \lambda^n_{j_n}, \; j_k = 1, 2, \cdots, m\prime_k \right\}. \qquad (45)$$

- Fisher criterion

$$W^{Fisher}_{j_1, j_2, \cdots, j_n} = \frac{\sum_{c=1}^{C} N_c \cdot \left(\overline{\mathbf{Y}}_{c; j_1, j_2, \cdots, j_n} - \overline{\mathbf{Y}}_{j_1, j_2, \cdots, j_n} \right)^2}{\sum_{i=1}^{N} \left(\mathbf{Y}_{i; j_1, j_2, \cdots, j_n} - \overline{\mathbf{Y}}_{c_i; j_1, j_2, \cdots, j_n} \right)^2} \qquad (46)$$

- TDPCA: Largest discriminant weights $|w_1| \geq |w_2| \geq \ldots \geq |w_m|$

Understanding Tensor Components



(a) (b)

Figure: Subspace S_1^{mpca} for FEI database. Ranking using: (a) Statistical variance (horizontal axis), TDPCA-SVM (red) and spectral variance (blue). (b) Including the ranking by Fisher criterion.

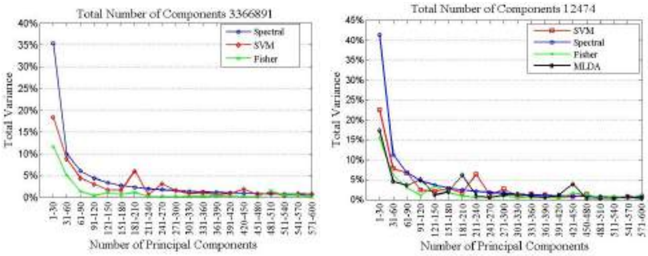
Understanding Tensor Components

In order to quantify the amount of variance of the principal components, the proportion of total variance information λ described by the k^{th} tensor principal component can be calculated as follows:

$$\lambda_k = \frac{\lambda_k}{\sum_{j=1}^{m} \lambda_j}, \quad k = 1, 2, ..., m = \prod_{i=1}^{n} m\prime_i, \tag{47}$$

where $\{\lambda_1, \lambda_2, ..., \lambda_m\}$ are the estimated variances.

Understanding Tensor Components: Total Variance



(a) (b)

Figure: (a) Amount of total variance explained by the 600 S_1^{mpca} most expressive tensor components selected by spectral variance, TDPCA-SVM, and Fisher criteria. (b) Amount of total variance using the 600 most expressive tensor components of S_2^{mpca}, including also the TDPCA-MLDA components.

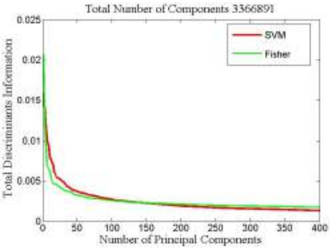
Understanding Tensor Components

In order to quantify the discriminant power of the principal components, the proportion of total discriminant information t described by the k^{th} tensor principal component can be calculated as follows:

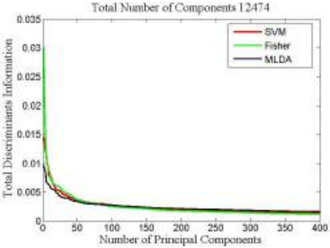
$$t_k = \frac{|\sigma_k|}{\sum_{j=1}^{m} |\sigma_j|}, \quad k = 1, 2, ..., m, \tag{48}$$

where m is the subspace dimension and $[\sigma_1, \sigma_2, ..., \sigma_m]$ are the weights computed using the separating hyperplanes.

Experimental Results: Total Discriminant



S_1^{mpca}



S_2^{mpca}

Recognition Rates

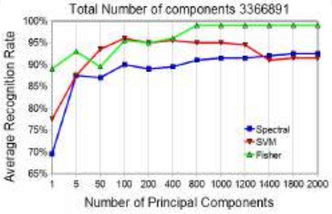
We have compared the effectiveness of the tensor principal components ranked on recognition tasks using:

- k-fold cross validation.
- Computing the Mahalanobis distance from $\mathbf{Y}_t$ to $\overline{\mathbf{Y}}_i$.

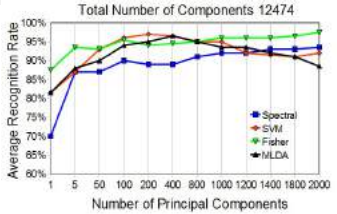
$$d_i^k(\mathbf{Y}_t) = \sum_{j=1}^{k} \frac{1}{\lambda_j}(\mathbf{Y}_{t;j} - \overline{\mathbf{Y}}_{i;j})^2 \tag{49}$$

where, $\widehat{\mathbf{Y}}_t$ is test observation and $\overline{\widehat{\mathbf{Y}}}_i$ class mean, with λ_j is the corresponding spectral variance and k is the number of tensor principal components retained.

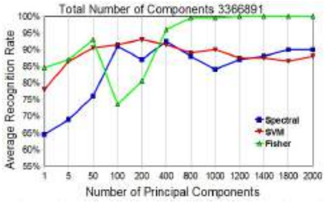
Recognition Rates for Gender (FEI Database)



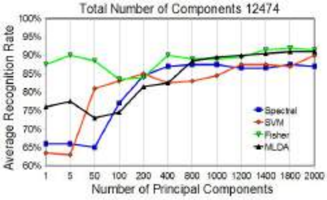
S_1^{mpca}



S_2^{mpca}



S_1^{csa}



S_2^{csa}

Projection in subspace $\widehat{B}_2$

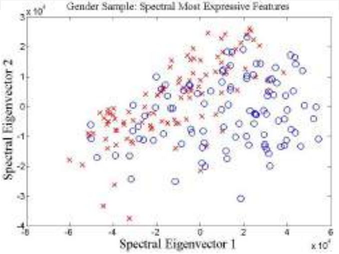


Figure: S_2^{mpca}

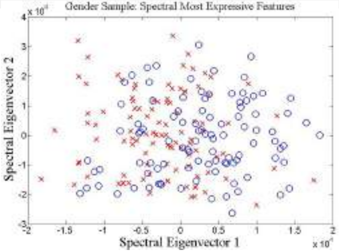


Figure: S_2^{csa}

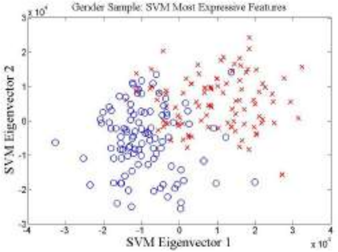


Figure: S_2^{mpca}

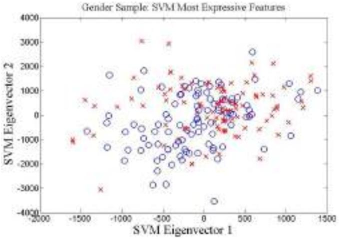


Figure: S_2^{csa}

Reconstruction

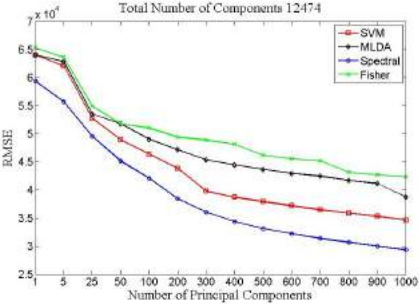
The reconstruction error, which is quantified through the root mean squared error (RMSE), computed as follows:

$$RMSE\left(\widehat{B}\right) = \sqrt{\frac{\sum_{i=1}^{N} ||\mathbf{X}_i^R - \mathbf{X}_i||^2}{N}}, \tag{50}$$

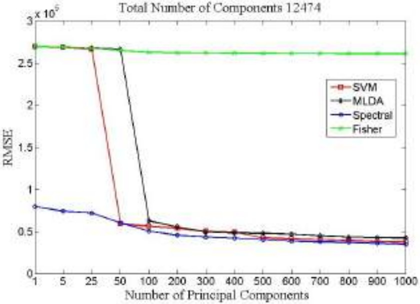
where $\widehat{B}$ is the subspace for projection.

Multidimensional Image Data Representation

RMSE



S_2^{mpca}



S_2^{csa}

Reconstruction from Subspace $\widehat{B}_{100}$

MPCA:




Fisher | TDPCA-MLDA | Spectral | TDPCA-SVM

CSA:




Fisher | TDPCA-MLDA | Spectral | TDPCA-SVM

Computational Complexity: Asymptotic Analysis

Assuming $m_1 = m_2 = \ldots m_n = m$. One iteration of MPCA

- Formation of the matrix $\Phi^{(k)}$: $O\left(N \cdot n \cdot m^{(n+1)}\right)$,
- Eigen-decomposition: $O\left(m^3\right)$
- Computation of projection $\widetilde{\mathbf{Y}}_i$: $O\left(n \cdot m^{(n+1)}\right)$
- Computation Complexity for MPCA: $O\left(N \cdot n \cdot m^{(n+1)}\right)$

Computational Complexity of Ranking Techniques

- Spectral: $O\left(\prod_{i=1}^{n} m\prime_i\right)$
- TDPCA-MLDA: $O(min(N, \prod_{i=1}^{n} m\prime_i) \cdot \prod_{i=1}^{n} m\prime_i))$
- TDPCA-SVM: $O(max(N, \prod_{i=1}^{n} m\prime_i) \cdot N^2)$
- Fisher: $O\left(N \cdot \prod_{i=1}^{n} m\prime_i\right)$

Issues in Multilinear Applications

- Issues in Multilinear Applications
- Many problems involving tensors are NP-hard. Ex: Finding the best rank-1 tensor decomposition [Hillar and Lim, 2013];
- A problem Π is NP-hard if a polynomial-time algorithm for Π would imply a polynomial-time algorithm for every problem in NP. A problem is NP-complete if it is both NP-hard and an element of NP.
- Memory/CPU requirements
- Reconstruction Artifacts
- Tensors in differentiable manifolds
- Incorporating prior information in MPCA

Perspectives: Manifold Learning and Tensor Fields

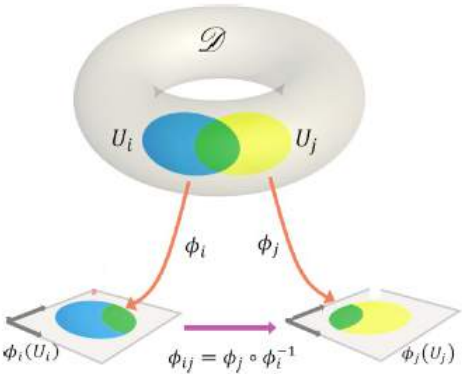


Figure: Manifold charting.

Perspectives: Manifold Learning and Tensor Fields

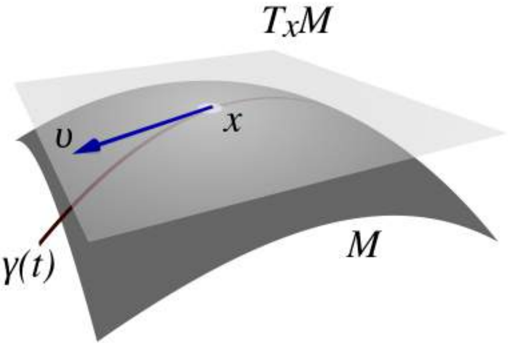


Figure: Tangent vector and tangent space.

Perspectives: Manifold Learning and Tensor Fields

- Tensor field
- Given subspaces $T_p^i(\mathcal{M}) \subset T_p(\mathcal{M})$, with $dim\left(T_p^i(\mathcal{M})\right) = m_i$, $i = 1, 2, \cdots, n$,
- The tensor product: $T_p^1(\mathcal{M}) \otimes T_p^2(\mathcal{M}) \otimes \cdots \otimes T_p^n(\mathcal{M})$,
- Individual basis $\left\{\mathbf{e}_k^{i_k}(p), i_k = 1, 2, \cdots, m_k\right\} \subset T_p^k(\mathcal{M})$;
- A basis for the vector space $T_p^1(\mathcal{M}) \otimes T_p^2(\mathcal{M}) \otimes \cdots \otimes T_p^n(\mathcal{M})$ is the set:

$$\left\{\mathbf{e}_1^{i_1}(p) \otimes \mathbf{e}_2^{i_2}(p) \otimes \cdots \otimes \mathbf{e}_n^{i_n}(p), \quad \mathbf{e}_k^{i_k}(p) \in T_p^k(\mathcal{M})\right\}. \tag{51}$$

- A tensor $\mathbf{X}$ of order n in $p \in \mathcal{M}$:

$$\mathbf{X}(p) = \sum_{i_1, i_2, \cdots, i_n} \mathbf{X}_{i_1, i_2, \cdots, i_n}(p)\, \mathbf{e}_1^{i_1}(p) \otimes \mathbf{e}_2^{i_2}(p) \otimes \cdots \otimes \mathbf{e}_n^{i_n}(p). \tag{52}$$

Perspectives: Manifold Learning and Tensor Fields

The application of the above concepts for data analysis depends on the following issues:

- Manifold learning to build the local coordinate systems $\{(U_\alpha, \varphi_\alpha)\}_{\alpha \in I}$, for $\mathcal{M}$;
- Discrete tensor field computation $\mathbf{X}(p_i)$, $i = 1, 2, \cdots, N$;
- Local subspace learning technique to perform dimensionality reduction to compute the discrete tensor field $\mathbf{Y}(p_i)$, $i = 1, 2, \cdots, N$, given by:

$$\mathbf{Y}(p_i) = \left(\mathbf{X} \times_1 \mathbf{U}^{1^T} \times_2 \mathbf{U}^{2^T} \ldots \times_n \mathbf{U}^{n^T}\right)(p_i), \tag{53}$$

Perspectives: Application of Spatial Weighting Maps

Leonardo da Vinci's Advice to Artists (E. Kelen, 1990)

- **"If Nature had a fixed model for the proportions of the face everyone would look alike and it would be impossible to tell them apart; but she has <u>varied</u> the pattern in such a way that although there is an all but universal standard as to size, one clearly distinguishes one face from another."**

→ Holistic cognition task composed of configural (global) and featural (local) sources of information.

Is the frontal face below of a male subject?

Is the frontal face below of a male subject?

Is the frontal face below of a male subject?

Yes. It is a male.

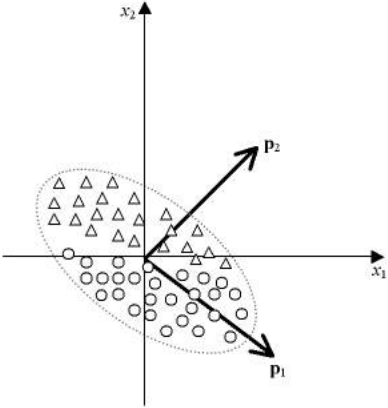
Priori information (from left to right): Statistical, cognitive and both.

How do we accomplish this process of coding and decoding faces?

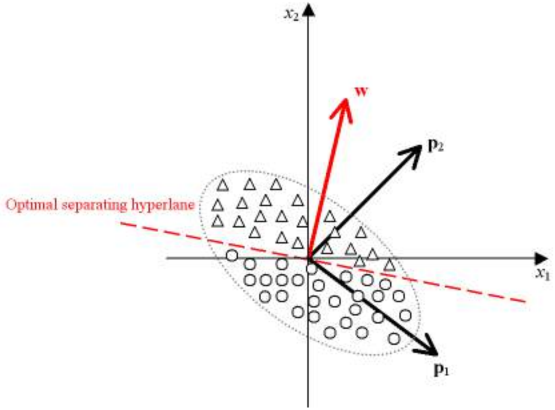
We will present a feature extraction approach of determining priori-driven dimensions along which face images vary that might be useful to understand the way faces are processed.

Goal: Combine variance with prior knowledge and analyze all features simultaneously.

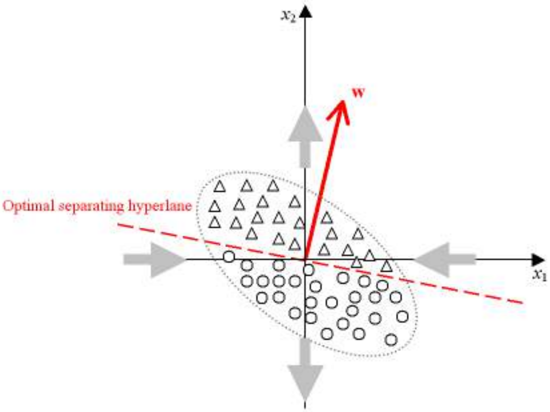
Geometric idea:



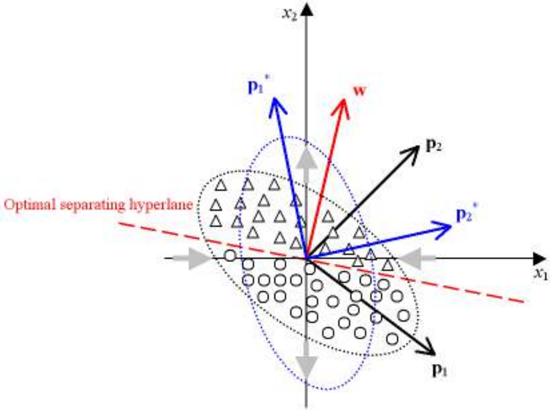
Geometric idea: (cont.)



Geometric idea: (cont.)



Geometric idea: (cont.)



Mathematically,

Let an $N \times n$ data matrix X be composed of N face images with n pixels, that is, $X = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)^T$. Let this data matrix X have covariance matrix

$$S = \frac{1}{(N-1)} \sum_{i=1}^{N} (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T,$$

where $\bar{\mathbf{x}}$ is the grand mean vector of X given by

$$\bar{\mathbf{x}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_i = (\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_n).$$

Mathematically, (cont.)

Let an $N \times n$ data matrix X be composed of N face images with n pixels, that is, $X = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)^T$. Let this data matrix X have covariance matrix

$$S = \frac{1}{(N-1)} \sum_{i=1}^{N} (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T,$$

where $\bar{\mathbf{x}}$ is the grand mean vector of X given by

$$\bar{\mathbf{x}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_i = (\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_n).$$

→ Variable deviations from the mean have the same weight. That is, all the n variables are equally important.

Mathematically, (cont.)

The well-known Pearson's sample correlation coefficient between the j^{th} and k^{th} pixels can be defined as follows:

$$
\begin{aligned}
r_{jk} &= \frac{s_{jk}}{\sqrt{s_j}\sqrt{s_k}} \\
&= \frac{\sum_{i=1}^{N}(x_{ij}-\bar{x}_j)(x_{ik}-\bar{x}_k)}{\sqrt{\sum_{i=1}^{N}(x_{ij}-\bar{x}_j)^2}\sqrt{\sum_{i=1}^{N}(x_{ik}-\bar{x}_k)^2}},
\end{aligned}
$$

for $j = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, n$. Analogously we can describe a priori-driven sample covariance s^*_{jk} between the j^{th} and k^{th} variables by

$$
\begin{aligned}
s^*_{jk} &= (\sqrt{w_j}\sqrt{w_k})s_{jk} \\
&= \sum_{i=1}^{N} \sqrt{w_j}(x_{ij}-\bar{x}_j)\sqrt{w_k}(x_{ik}-\bar{x}_k).
\end{aligned}
$$

Mathematically, (cont.)

The spatial weighting vector (or spatial attention map)

$$\mathbf{w} = \left[w_1, w_2, \ldots, w_n\right]^T$$

is such that $w_j \geq 0$ and $\sum_{j=1}^{n} w_j = 1$, where each w_j measures the information power of the j^{th} pixel. Thus, when n pixels are observed on N samples, the weighted sample covariance matrix S^* can be described by

$$S^* = \left\{s^*_{jk}\right\} = \left\{\sum_{i=1}^{N} \sqrt{w_j}(x_{ij} - \bar{x}_j)\sqrt{w_k}(x_{ik} - \bar{x}_k)\right\}.$$

Mathematically, (cont.)

Let S^* have respectively P^* and Λ^* eigenvector and eigenvalue matrices, as follows:

$$P^{*T} S^* P^* = \Lambda^*.$$

The set of m ($m \leq n$) eigenvectors of S^*, that is,

$$P^* = [\mathbf{p}_1^*, \mathbf{p}_2^*, \ldots, \mathbf{p}_m^*],$$

which corresponds to the $\underline{m}$ largest eigenvalues, defines the orthonormal coordinate system for the data matrix X called priori-driven principal components.

Incorporating priori-driven information (algorithm)
Spatially weighted PCA

1. Calculate a spatial weighting vector $\mathbf{w} = [w_1, w_2, \ldots, w_n]^T$;
2. Normalize $\mathbf{w}$: Replace w_j with $\frac{|w_j|}{\sum_{j=1}^{n} |w_j|}$;
3. Standardize all n variables, replacing x_{ij} with $z_{ij} = x_{ij} - \bar{x}_j$;
4. Spatially weigh up all the standardized z_{ij}: $z^*_{ij} = z_{ij}\sqrt{w_j}$;
5. Calculate P^*, the m largest eigenvectors of $(Z^*)^T Z^*$.

Experiments

Two-group separation tasks (frontal faces):

- Gender (female versus male)
- Facial expression (smiling versus non-smiling)

Database (2D): FEI (400 images, 200 subjects)

Note: All the face images have been converted to grayscale, pre-aligned and cropped to 128x128 pixels in size.

Experiments (cont.)

Statistical prior information has been described simply by the leading eigenvector of the between-scatter matrix S_b

$$S_b = \sum_{i=1}^{g} N_i(\bar{\mathbf{x}}_i - \bar{\mathbf{x}})(\bar{\mathbf{x}}_i - \bar{\mathbf{x}})^T,$$

$\rightarrow$ 1st order statistical differences.

Experiments (cont.)

Cognitive prior information has been described by (absolute) gaze duration heatmap means using Tobii TX300 eye tracker and the following settings:

- Binocular tracking
- Data sampling rate of 300Hz
- Minimum fixation duration of 60ms
- Maximum dispersion threshold of 0.5 degrees

Each stimulus task begins with a calibration procedure implemented in the Tobii Studio software to ensure accurate tracking of the eye gaze.

Experiments (cont.)

Stimuli composed of 60 images distributed equally among the tasks with the following display settings:

- Face images enlarged to 512x512 pixels
- Black background and stimuli centralized
- Central fixation cross in between stimuli
- Presentation at a distance of 60cm
- 23in 1920x1080 widescreen monitor

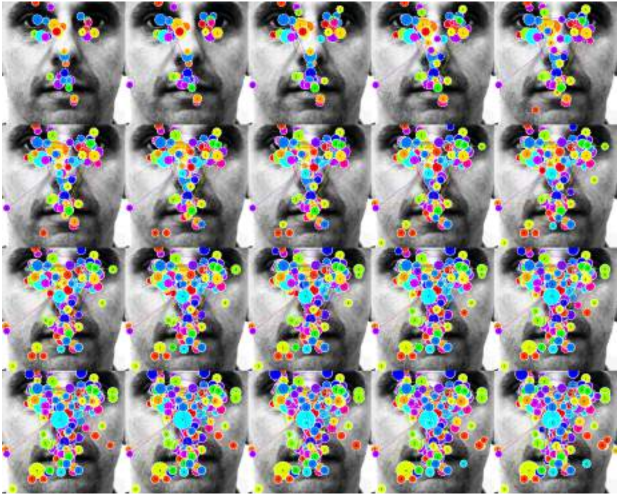
All face stimuli are presented for 3 seconds followed by a question that requires a response in relation to the task (male/female, smiling/non-smiling).

Experiments (cont.)

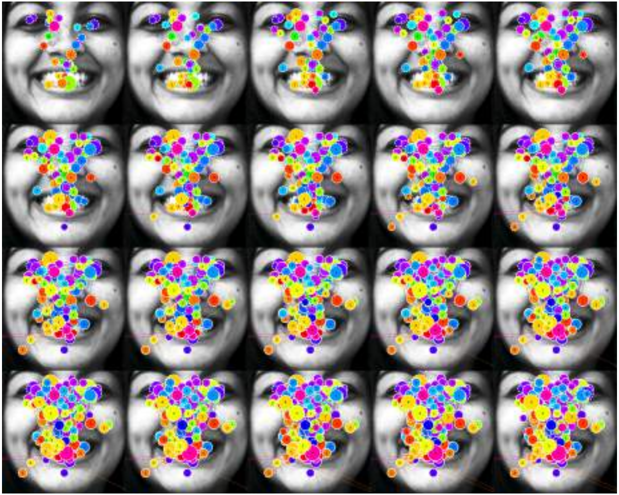
A number of 44 adults (26 men and 18 women) participated in these experiments. All participants:

- are Brazilian students or staff at FEI
- have normal or correct vision
- provided written informed consent
- Presentation at a distance of 60cm

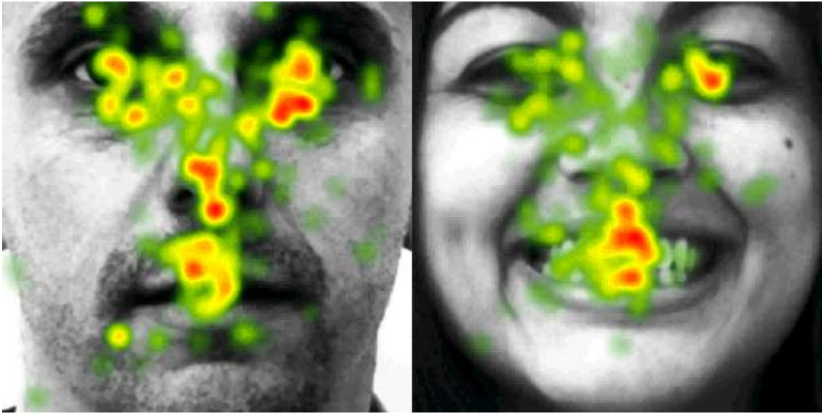
Gender experiments/results (gazeplot, 3 seconds)



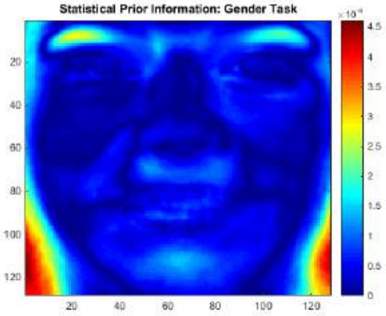
Facial expression experiments/results (gazeplot, 3 seconds)



Gender and facial expression corresponding heatmaps

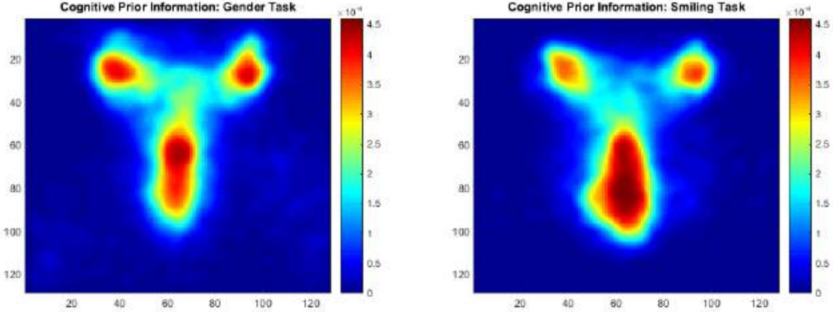


Results (statistical prior information)



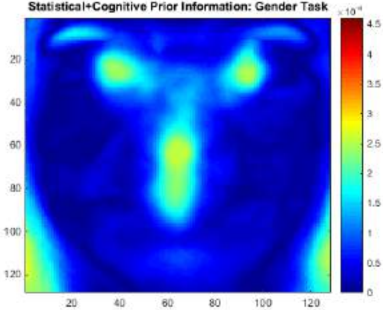
Note: Differences are essentially local.

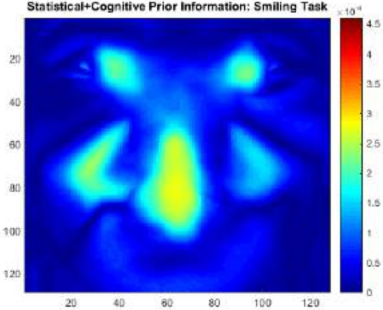
Results (cognitive prior information)



Note: Differences are essentially global.

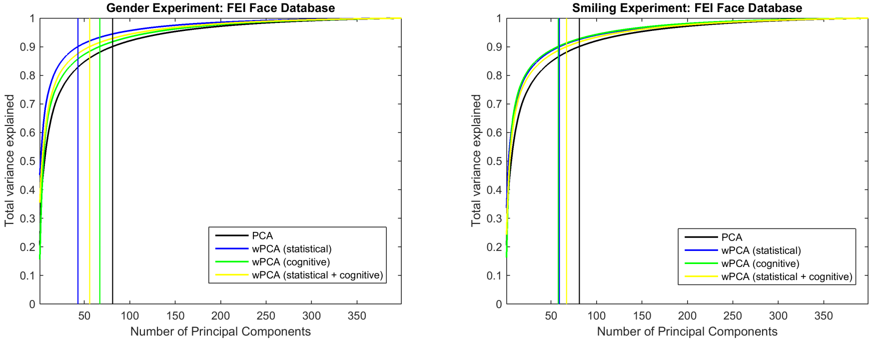
Results (both prior information)





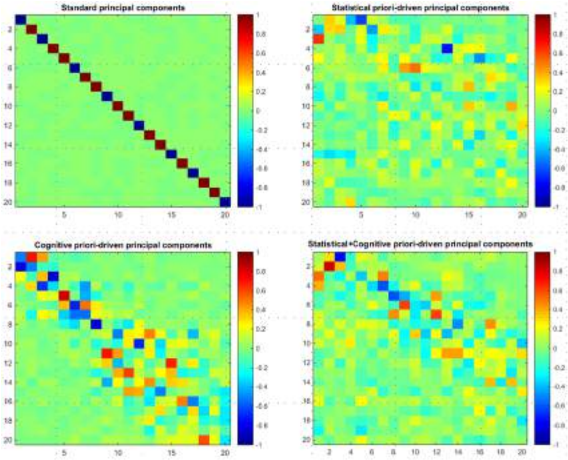
Note: Global and local differences.

Results (dimensionality reduction)



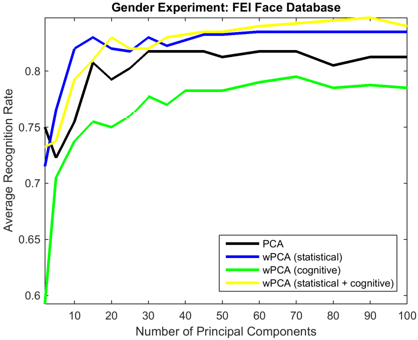
Note: Less priori-driven components than standard ones.

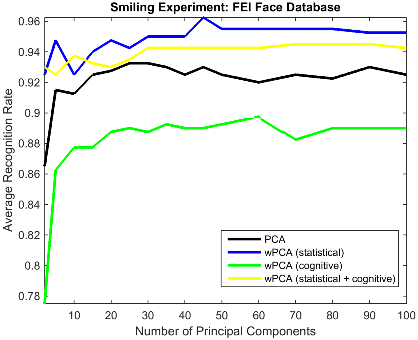
Results (inner product matrices of the top 20 components)



Note: Changes in the information retained and the ordering of pcas.

Results (10-fold cross validation, nearest neighbor)





Note: Cognitive prior information alone disappointing. But if combined with statistical one could improve the discriminant power of the components.

Thus,

How do we accomplish this process of coding and decoding faces?

"... it is not really the perception of likeness for which we are originally programmed, but the noticing of unlikeness, the departure from the norm which stands out and sticks in the mind." (Grombrich, E. H., 1972)

$\rightarrow$ Priori-driven variance might be a way forward.

Perspectives: High Performance and NN-Factoring

- High Performance Solutions:
  - Reduce the inter processor communication in distributed memory algorithms [Austin et al., 2015],
  - Intermediate data explosion: computation of matrix $\Phi^{(k)}$, fast memory access in mode-k flattening
  - Out-of-Core solutions [Suter et al., 2011]
  - Quantum Computers: Could they be effective for tensor problems [Hillar and Lim, 2013]?
- Visualization Artifacts: Non-Negative factorization

Conclusions and Final Remarks

- Tensor in pattern recognition:
  - Dimensionality reduction,
  - Ranking tensor components
  - Classification
  - Reconstruction
- Generalized Matrix and Tensor Product Approaches
- Tensor computation is not matrix computation with additional subscripts
  - Tensors are geometric objects
  - Manifold learning
- Incorporation of prior knowledge to steer the data mining tasks
- High Performance Requirements

Austin, W., Ballard, G., and Kolda, T. G. (2015).
Parallel tensor compression for large-scale scientific data.
CoRR, abs/1510.06689.

Filisbino, T., Giraldi, G., and Thomaz, C. (2013a).
Defining and sorting tensor components for face image analysis.
Technical report, National Laboratory for Scientific Computing.

Filisbino, T., Giraldi, G., and Thomaz, C. (2013b).
Ranking methods for tensor components analysis and their application to face images.
In *Graphics, Patterns and Images (SIBGRAPI), 2013 26th SIBGRAPI - Conference on*, pages 312–319.

Filisbino, T. A., Giraldi, G. A., and Thomaz, C. E. (2015).
Comparing ranking methods for tensor components in multilinear and concurrent subspace analysis with applications in face images.
Int. J. Image Graphics, 15(1).

Hillar, C. J. and Lim, L.-H. (2013).

Most tensor problems are np-hard.
J. ACM, 60(6):45:1–45:39.

Jia, K. and Gong, S. (2005).
Multi-modal tensor face for simultaneous super-resolution and recognition.
In *ICCV*, pages 1683–1690. IEEE Computer Society.

Liu, Y., Wu, F., Zhuang, Y., and Xiao, J. (2008).
Active post-refined multimodality video semantic concept detection with tensor representation.
In *Proc. of the 16th ACM Inter. Conf. on Multimedia*, pages 91–100.

Lu, H., Plataniotis, K., and Venetsanopoulos, A. (2008).
Mpca: Multilinear principal component analysis of tensor objects.
Neural Networks, IEEE Trans. on, 19(1):18–39.

Lu, H., Plataniotis, K. N., and Venetsanopoulos, A. N. (2009).
Uncorrelated multilinear principal component analysis for unsupervised multilinear subspace learning.

Neural Networks, IEEE Trans. on, 20(11):1820–1836.

Pajarola, R., Suter, S. K., and Ruiters, R. (2013).
Tensor approximation in visualization and computer graphics.
In *Eurographics 2013 - Tutorials*, number t6, Girona, Spain. Eurographics Association.

Panagakis, Y., Kotropoulos, C., and Arce, G. (2010).
Non-negative multilinear principal component analysis of auditory temporal modulations for music genre classification.
Audio, Speech, and Language Processing, IEEE Trans. on, 18(3):576–588.

Suter, S. K., GuitiÃ¡n, J. A. I., Marton, F., Agus, M., Elsener, A., Zollikofer, C. P. E., Gopi, M., Gobbetti, E., and Pajarola, R. (2011).
Interactive multiscale tensor reconstruction for multiresolution volume visualization.
IEEE Trans. Vis. Comput. Graph., 17(12):2135–2143.

Thomaz, C. E. and Giraldi, G. A. (2010).

A new ranking method for principal components analysis and its application to face image analysis.
Image Vision Comput., 28(6):902–913.

Vlasic, D., Brand, M., Pfister, H., and Popović, J. (2005).
Face transfer with multilinear models.
ACM Trans. Graph., 24:426–433.

Xu, D., Yan, S., Zhang, L., Lin, S., Zhang, H.-J., and Huang, T. (2008).
Reconstruction and recognition of tensor-based objects with concurrent subspaces analysis.
Circuits and Systems for Video Technology, IEEE Trans. on, 18(1):36–47.

Yan, S., Xu, D., Yang, Q., Zhang, L., Tang, X., and Zhang, H.-J. (2005).
Discriminant analysis with tensor representation.
In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 526–532 vol. 1.

Zhou, X., Chen, L., and Zhou, X. (2012).
Structure tensor series-based large scale near-duplicate video retrieval.
Multimedia, IEEE Trans. on, 14(4):1220–1233.