Notes on Implementation of Component Analysis Techniques

Dr. Stefanos Zafeiriou

January 2015

1 Computing Principal Component Analysis

Assume that we have a matrix of centered data observations

$$\mathbf{X} = [\mathbf{x}_1 - \boldsymbol{\mu}, \ldots, \mathbf{x}_N - \boldsymbol{\mu}] \tag{1}$$

where $\boldsymbol{\mu}$ denotes the mean vector. $\mathbf{X}$ has size $F \times N$, where F is the number of dimensions and N is the number of observations. Their covariance matrix is given by

$$\mathbf{S_t} = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^T = \frac{1}{N}\mathbf{X}\mathbf{X}^T \tag{2}$$

In Principal Component Analysis (PCA), we aim to maximize the variance of each dimension by maximizing

$$\begin{aligned} \mathbf{W_0} = \arg\max_{\mathbf{W}} \quad & \mathrm{tr}(\mathbf{W}^T\mathbf{S_t}\mathbf{W}) \\ \text{subject to} \quad & \mathbf{W}^T\mathbf{W} = \mathbf{I} \end{aligned} \tag{3}$$

The solution of Eq. 3 can be derived by solving

$$\mathbf{S_t}\mathbf{W} = \mathbf{W}\boldsymbol{\Lambda} \tag{4}$$

Thus, we need to perform eigenanalysis on $\mathbf{S_t}$. If we want to keep d principal components, the computational cost of the above operation is $\mathcal{O}(dF^2)$. If F is large, this computation can be quite expensive.

Lemma 1
Let us assume that $\mathbf{B} = \mathbf{X}\mathbf{X}^T$ and $\mathbf{C} = \mathbf{X}^T\mathbf{X}$. It can be proven that $\mathbf{B}$ and $\mathbf{C}$ have the same positive eigenvalues $\boldsymbol{\Lambda}$ and, assuming that $N < F$, then the eigenvectors $\mathbf{U}$ of $\mathbf{B}$ and the eigenvectors $\mathbf{V}$ of $\mathbf{C}$ are related as $\mathbf{U} = \mathbf{X}\mathbf{V}\boldsymbol{\Lambda}^{-\frac{1}{2}}$.

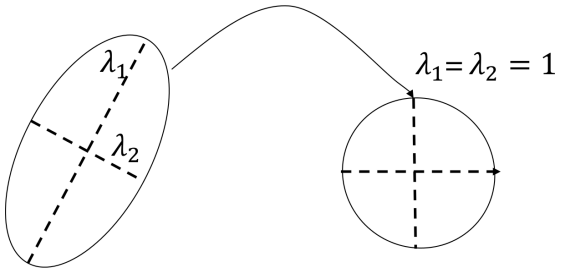

Figure 1: Example of data whitening using the PCA projection matrix $\mathbf{W} = \mathbf{U}\mathbf{\Lambda}^{-\frac{1}{2}}$.

Using Lemma 1 we can compute the eigenvectors $\mathbf{U}$ of $\mathbf{S_t}$ in $\mathcal{O}(N^3)$. The eigenanalysis of $\mathbf{X}^T\mathbf{X}$ is denoted by

$$\mathbf{X}^T\mathbf{X} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T \tag{5}$$

where $\mathbf{V}$ is a $N \times (N-1)$ matrix with the eigenvectors as columns and $\mathbf{\Lambda}$ is a $(N-1) \times (N-1)$ diagonal matrix with the eigenvalues. Given that $\mathbf{V}^T\mathbf{V} = \mathbf{I}$ and $\mathbf{V}\mathbf{V}^T \neq \mathbf{I}$ we have

$$\left.\begin{array}{l}\mathbf{X}^T\mathbf{X} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T \\ \mathbf{U} = \mathbf{X}\mathbf{V}\mathbf{\Lambda}^{-\frac{1}{2}}\end{array}\right\} \Rightarrow \mathbf{U}^T\mathbf{X}\mathbf{X}^T\mathbf{U} = \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{V}^T\mathbf{X}^T\mathbf{X}\mathbf{X}^T\mathbf{X}\mathbf{V}\mathbf{\Lambda}^{-\frac{1}{2}} =$$

$$= \mathbf{\Lambda}^{-\frac{1}{2}}\underbrace{\mathbf{V}^T\mathbf{V}}_{\mathbf{I}}\mathbf{\Lambda}\underbrace{\mathbf{V}^T\mathbf{V}}_{\mathbf{I}}\mathbf{\Lambda}\underbrace{\mathbf{V}^T\mathbf{V}}_{\mathbf{I}}\mathbf{\Lambda}^{-\frac{1}{2}} = \tag{6}$$

$$= \mathbf{\Lambda}$$

The pseudocode for computing PCA is

Algorithm 1 Principal Component Analysis

1: **procedure** PCA
2: Compute dot product matrix: $\mathbf{X}^T\mathbf{X} = \sum_{i=1}^{N}(\mathbf{x}_i - \boldsymbol{\mu})^T(\mathbf{x}_i - \boldsymbol{\mu})$
3: Eigenanalysis: $\mathbf{X}^T\mathbf{X} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$
4: Compute eigenvectors: $\mathbf{U} = \mathbf{X}\mathbf{V}\mathbf{\Lambda}^{-\frac{1}{2}}$
5: Keep specific number of first components: $\mathbf{U_d} = [\mathbf{u}_1, \ldots, \mathbf{u}_d]$
6: Compute d features: $\mathbf{Y} = \mathbf{U_d}^T\mathbf{X}$

Now, the covariance matrix of $\mathbf{Y}$ is

$$\mathbf{Y}\mathbf{Y}^T = \mathbf{U}^T\mathbf{X}\mathbf{X}^T\mathbf{U} = \mathbf{\Lambda}$$

The final solution of Eq. 3 is given as the projection matrix

$$\mathbf{W} = \mathbf{U}\mathbf{\Lambda}^{-\frac{1}{2}} \tag{7}$$

which normalizes the data to have unit variance (Figure 1). This procedure is called whitening (or sphereing).

2 Computing Linear Discriminant Analysis

As explained before (Section 1), PCA finds the principal components that maximize the data variance without taking into account the class labels. In contrast to this, Linear Discriminant Analysis (LDA) computes the linear directions that maximize the separation between multiple classes. This is mathematically expressed as maximizing

$$\begin{aligned} \mathbf{W_0} = \arg\max_{\mathbf{W}} \quad & \mathrm{tr}(\mathbf{W}^T \mathbf{S_b} \mathbf{W}) \\ \text{subject to} \quad & \mathbf{W}^T \mathbf{S_w} \mathbf{W} = \mathbf{I} \end{aligned} \tag{8}$$

Assume that we have C number of classes, denoted by $\mathbf{c}_i = [\mathbf{x}_1, \ldots, \mathbf{x}_{N_{\mathbf{c}_i}}], i = 1, \ldots, C$, where each $\mathbf{x}_j$ has F dimensions and $\boldsymbol{\mu}(\mathbf{c}_i)$ is the mean vector of the class $\mathbf{c}_i$. Thus, the overall data matrix is $\mathbf{X} = [\mathbf{c}_1, \ldots, \mathbf{c}_C]$ with size $F \times N$ ($N = \sum_{i=1}^{C} N_{\mathbf{c}_i}$) and $\boldsymbol{\mu}$ is the overall mean (mean of means). $\mathbf{S_w}$ is the within-class scatter matrix

$$\mathbf{S_w} = \sum_{j=1}^{C} \mathbf{S}_j = \sum_{j=1}^{C} \sum_{\mathbf{x}_i \in \mathbf{c}_j} (\mathbf{x}_i - \boldsymbol{\mu}(\mathbf{c}_j))(\mathbf{x}_i - \boldsymbol{\mu}(\mathbf{c}_j))^T \tag{9}$$

that has $\mathrm{rank}\,(\mathbf{S_w}) = \min(F, N - C)$. Moreover, $\mathbf{S_b}$ is the between-class scatter matrix

$$\mathbf{S_b} = \sum_{j=1}^{C} N_{\mathbf{c}_j} (\boldsymbol{\mu}(\mathbf{c}_j) - \boldsymbol{\mu})(\boldsymbol{\mu}(\mathbf{c}_j) - \boldsymbol{\mu})^T \tag{10}$$

that has $\mathrm{rank}\,(\mathbf{S_b}) = \min(F, C - 1)$. The solution of Eq. 8 is given from the generalized eigenvalue problem

$$\mathbf{S_b} \mathbf{W} = \mathbf{S_w} \mathbf{W} \boldsymbol{\Lambda} \tag{11}$$

thus $\mathbf{W_0}$ corresponds to the eigenvectors of ${\mathbf{S_w}}^{-1} \mathbf{S_b}$ that have the largest eigenvalues. In order to deal with the singularity of $\mathbf{S_w}$, we can do the following steps:

1. Perform PCA on our data matrix $\mathbf{X}$ to reduce the dimensions to $N - C$ using the eigenvectors $\mathbf{U}$
2. Solve LDA on this reduced spaceand get $\mathbf{Q}$ that has $C - 1$ columns.
3. Compute the total transform as $\mathbf{W} = \mathbf{U}\mathbf{Q}$.

Unfortunately, if you follow the above procedure is possible that important information is solved. In the following, we show how the components of LDA can be computed by applying a simultaneous diagonalization procedure.

Properties
The scatter matrices have some interesting properties. Let us denote

$$\mathbf{M} = \begin{bmatrix} \mathbf{E}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{E}_2 & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{E}_C \end{bmatrix} = \mathrm{diag}\,\{\mathbf{E}_1, \mathbf{E}_2, \ldots, \mathbf{E}_C\} \tag{12}$$

where

$$\mathbf{E}_i = \begin{bmatrix} \frac{1}{N_{\mathbf{c}_i}} & \cdots & \frac{1}{N_{\mathbf{c}_i}} \\ \vdots & \ddots & \vdots \\ \frac{1}{N_{\mathbf{c}_i}} & \cdots & \frac{1}{N_{\mathbf{c}_i}} \end{bmatrix}_{N_{\mathbf{c}_i} \times N_{\mathbf{c}_i}} \tag{13}$$

Note that $\mathbf{M}$ is idempotent, thus $\mathbf{MM} = \mathbf{M}$. Given that the data covariance matrix is $\mathbf{S_t} = \mathbf{X}\mathbf{X}^T$, the between-class scatter matrix can be written as

$$\mathbf{S_b} = \mathbf{XMMX}^T = \mathbf{XMX}^T \tag{14}$$

and the within-class scatter matrix as

$$\mathbf{S_w} = \underbrace{\mathbf{X}\mathbf{X}^T}_{\mathbf{S_t}} - \underbrace{\mathbf{X}\mathbf{M}\mathbf{X}^T}_{\mathbf{S_b}} = \mathbf{X}(\mathbf{I} - \mathbf{M})\mathbf{X}^T \tag{15}$$

Thus, we have that $\mathbf{S_t} = \mathbf{S_w} + \mathbf{S_b}$. Note that since $\mathbf{M}$ is idempotent, $\mathbf{I} - \mathbf{M}$ is also idempotent.

Given the above properties, the objective function of Eq. 8 can be expressed as

$$\begin{aligned} \mathbf{W_0} = \arg\max_{\mathbf{W}} \quad & \mathrm{tr}(\mathbf{W}^T\mathbf{XMMX}^T\mathbf{W}) \\ \text{subject to} \quad & \mathbf{W}^T\mathbf{X}(\mathbf{I} - \mathbf{M})(\mathbf{I} - \mathbf{M})\mathbf{X}^T\mathbf{W} = \mathbf{I} \end{aligned} \tag{16}$$

The optimization of this problem involves a procedure called Simultaneous Diagonalization. Let's assume that the final transform matrix has the form

$$\mathbf{W} = \mathbf{UQ} \tag{17}$$

We aim to find the matrix $\mathbf{U}$ that diagonalizes $\mathbf{S_w} = \mathbf{X}(\mathbf{I} - \mathbf{M})(\mathbf{I} - \mathbf{M})\mathbf{X}^T$. This practically means that, given the constraint of Eq. 16, we want

$$\begin{aligned} &\mathbf{W}^T\mathbf{X}(\mathbf{I} - \mathbf{M})(\mathbf{I} - \mathbf{M})\mathbf{X}^T\mathbf{W} = \mathbf{I} \Rightarrow \\ \Rightarrow &\mathbf{Q}^T \underbrace{\mathbf{U}^T\mathbf{X}(\mathbf{I} - \mathbf{M})(\mathbf{I} - \mathbf{M})\mathbf{X}^T\mathbf{U}}_{\mathbf{I}} \mathbf{Q} = \mathbf{I} \end{aligned} \tag{18}$$

Consequently, using Eqs. 17 and 18, the objective function of Eq. 16 can be further expressed as

$$\begin{aligned} \mathbf{Q_0} = \arg\max_{\mathbf{Q}} \quad & \mathrm{tr}(\mathbf{Q}^T\mathbf{U}^T\mathbf{XMMX}^T\mathbf{UQ}) \\ \text{subject to} \quad & \mathbf{Q}^T\mathbf{Q} = \mathbf{I} \end{aligned} \tag{19}$$

where the constraint $\mathbf{W}^T\mathbf{X}(\mathbf{I} - \mathbf{M})(\mathbf{I} - \mathbf{M})\mathbf{X}^T\mathbf{W} = \mathbf{I}$ now has the form $\mathbf{Q}^T\mathbf{Q} = \mathbf{I}$.

Lemma 2
Assume the matrix $\mathbf{X}(\mathbf{I} - \mathbf{M})(\mathbf{I} - \mathbf{M})\mathbf{X}^T = \mathbf{X_w}{\mathbf{X_w}}^T$, where $\mathbf{X_w}$ is the $F \times N$ matrix $\mathbf{X_w} = \mathbf{X}(\mathbf{I} - \mathbf{M})$. By performing eigenanalysis on ${\mathbf{X_w}}^T\mathbf{X_w}$ as ${\mathbf{X_w}}^T\mathbf{X_w} =$

$\mathbf{V_w}\mathbf{\Lambda}\mathbf{V_w}^T$, we get $N-C$ positive eigenvalues, thus $\mathbf{V_w}$ is a $N \times (N-C)$ matrix.

The optimization problem of Eq. 19 can be solved in two steps

1. Find $\mathbf{U}$ such that $\mathbf{U}^T\mathbf{X}(\mathbf{I}-\mathbf{M})(\mathbf{I}-\mathbf{M})\mathbf{X}^T\mathbf{U} = \mathbf{I}$. By applying Lemma 2, we get $\mathbf{U} = \mathbf{X_w}\mathbf{V_w}\mathbf{\Lambda_w}^{-1}$. Note that $\mathbf{U}$ has size $F \times (N-C)$.

2. Find $\mathbf{Q_0}$. By denoting
$$\tilde{\mathbf{X}}_\mathbf{b} = \mathbf{U}^T\mathbf{X}\mathbf{M}$$
the $(N-C) \times N$ matrix of projected class means, Eq. 19 becomes
$$\begin{aligned} \mathbf{Q_0} = \arg\max_{\mathbf{Q}} \quad & \mathrm{tr}(\mathbf{Q}^T\tilde{\mathbf{X}}_\mathbf{b}\tilde{\mathbf{X}}_\mathbf{b}^T\mathbf{Q}) \\ \text{subject to} \quad & \mathbf{Q}^T\mathbf{Q} = \mathbf{I} \end{aligned} \tag{20}$$
which is equivalent to applying PCA on the matrix of projected class means. The final $\mathbf{Q_0}$ is a matrix with columns the d eigenvectors of $\tilde{\mathbf{X}}_\mathbf{b}\tilde{\mathbf{X}}_\mathbf{b}^T$ that correspond to the d largest eigenvalues ($d \leq C-1$).

The final projection matrix is given by

$$\mathbf{W_0} = \mathbf{Q_0}\mathbf{U} \tag{21}$$

Based on the above, the pseudocode for computing LDA is

Algorithm 2 Linear Discriminant Analysis

1: **procedure** LDA
2: Find eigenvectors of $\mathbf{S_w}$ that correspond to non-zero eigenvalues (usually $N-C$), i.e. $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_{N-C}]$ by performing eigen-analysis to $(\mathbf{I}-\mathbf{M})\mathbf{X}^T\mathbf{X}(\mathbf{I}-\mathbf{M}) = \mathbf{V_w}\mathbf{\Lambda_w}\mathbf{V_w}^T$ and computing $\mathbf{U} = \mathbf{X}(\mathbf{I}-\mathbf{M})\mathbf{V_w}\mathbf{\Lambda_w}^{-1}$ (performing whitening on $\mathbf{S}_w$).
3: Project the data as $\tilde{\mathbf{X}}_\mathbf{b} = \mathbf{U}^T\mathbf{X}\mathbf{M}$.
4: Perform PCA on $\tilde{\mathbf{X}}_\mathbf{b}$ to find $\mathbf{Q}$ (i.e., compute the eigenanalysis of $\tilde{\mathbf{X}}_\mathbf{b}\tilde{\mathbf{X}}_\mathbf{b}^T = \mathbf{Q}\mathbf{\Lambda_b}\mathbf{Q}^T$).
5: The total transform is $\mathbf{W} = \mathbf{U}\mathbf{Q}$

Locality preserving projections can be computed in a similar fashion. The first step is to perform whitening of $\mathbf{X}\mathbf{D}\mathbf{X}^T$. We do so by applying Lemma 1 and performing eigenanalysis of $\mathbf{D}^{1/2}\mathbf{X}^T\mathbf{X}\mathbf{D}^{1/2} = \mathbf{V_p}\mathbf{\Lambda_p}\mathbf{V_p}^T$. Then, the whitening transform is given by $\mathbf{U}^T = \mathbf{X}\mathbf{D}^{1/2}\mathbf{\Lambda_p}^{-1/2}$. The next step is to project the data as $\tilde{\mathbf{X}}_\mathbf{p} = \mathbf{U}^T\mathbf{X}$ and find the eigenvectors $\mathbf{Q}$ of $\tilde{\mathbf{X}}_\mathbf{p}(\mathbf{D}-\mathbf{S})\tilde{\mathbf{X}}_\mathbf{p}^T$ [1] that correspond to the lowest (but non-zero eigenvalues). Then the total transform would be $\mathbf{W} = \mathbf{U}\mathbf{Q}$.

[1] Where $\mathbf{S}$ is the connectivity matrix defined in the slides