LEC 5: Two Dimensional Linear Discriminant Analysis

Dr. Guangliang Chen

March 8, 2016

Outline

- Last time: LDA/QDA (classification)
- Today: 2DLDA (dimensionality reduction)
- HW3b and Midterm Project 3

What is 2DLDA?

Both PCA and FDA require first vectorizing the images. The output is also in vector form.

2DLDA considers images as two-dimensional signals and works with matrices directly (no vectorization needed). The output will still be images (but smaller).

2DLDA has the advantage of preserving information along both dimensions (i.e., rows and columns).

How does 2DLDA work?

2DLDA transforms $r \times c$ images to smaller $r' \times c'$ images.

Let $\mathbf{X} \in \mathbb{R}^{r \times c}$ be a given image. The transformation is defined by two matrices with orthonormal columns, $\mathbf{L} \in \mathbb{R}^{r \times r'}$ and $\mathbf{R} \in \mathbb{R}^{c \times c'}$:

$$\mathbf{Y} = \mathbf{L}^T \mathbf{X} \mathbf{R} \in \mathbb{R}^{r' \times c'}.$$

Like FDA, 2DLDA finds the best transformations $\mathbf{L}, \mathbf{R}$ by preserving the most discriminatory information in the projection space

$$\max_{\mathbf{L},\mathbf{R}} \frac{\text{between-class scatter}}{\text{within-class scatter}}$$

Notation

Let $\mathbf{X}_i \in \mathbb{R}^{r\times c}, 1 \le i \le n$ be the ith image in the training set, which consist of k classes $\Pi_1, \ldots, \Pi_k$.

Let

$$\mathbf{M}_i = \frac{1}{n_i} \sum_{\mathbf{X}\in\Pi_i} \mathbf{X}$$

be the (matrix) mean of class i, and

$$\mathbf{M} = \frac{1}{n} \sum_{1\le i\le k} \sum_{\mathbf{X}\in\Pi_i} \mathbf{X} = \frac{1}{n} \sum_i n_i \mathbf{M}_i$$

the global mean.

Matrix norms

To define within-class and between-class scatters we need to introduce matrix norms. For any matrix $\mathbf{A}$ (not necessarily square), define

- Frobenius norm:

$$\|\mathbf{A}\|_{\mathrm{F}} = \sqrt{\sum_{i,j} a_{ij}^2}$$

  It can be shown that $\|\mathbf{A}\|_{\mathrm{F}} = \sqrt{\sum_i \sigma_i^2}$

- Spectral norm:

$$\|\mathbf{A}\|_2 = \sigma_1 \quad \text{(largest singular value)}$$

See Instructor's lecture notes on SVD for more detail.

Defining within-class and between-class scatters

In the *given* image space

- Within-class scatter:

$$s_w^2 = \sum_i \sum_{\mathbf{X} \in \Pi_i} \|\mathbf{X} - \mathbf{M}_i\|_\mathrm{F}^2$$

- Between-class scatter:

$$s_b^2 = \sum_i n_i \|\mathbf{M}_i - \mathbf{M}\|_\mathrm{F}^2$$

Defining within-class and between-class scatter

In the *transformed* space

- Within-class scatter:

$$\begin{aligned}\tilde{s}_w^2 &= \sum_i \sum_{\mathbf{X}\in\Pi_i} \|\mathbf{L}^T\mathbf{X}\mathbf{R} - \mathbf{L}^T\mathbf{M}_i\mathbf{R}\|_\mathrm{F}^2 \\ &= \sum_i \sum_{\mathbf{X}\in\Pi_i} \|\mathbf{L}^T(\mathbf{X} - \mathbf{M}_i)\mathbf{R}\|_\mathrm{F}^2\end{aligned}$$

- Between-class scatter:

$$\tilde{s}_b^2 = \sum_i n_i \|\mathbf{L}^T(\mathbf{M}_i - \mathbf{M})\mathbf{R}\|_\mathrm{F}^2$$

The mathematical formulation of 2DLDA

Solve

$$\max_{\mathbf{L},\mathbf{R}} \frac{\sum_i n_i \|\mathbf{L}^T(\mathbf{M}_i - \mathbf{M})\mathbf{R}\|_{\mathrm{F}}^2}{\sum_i \sum_{\mathbf{X}\in\Pi_i} \|\mathbf{L}^T(\mathbf{X} - \mathbf{M}_i)\mathbf{R}\|_{\mathrm{F}}^2}$$

where $\mathbf{L} \in \mathbb{R}^{r\times r'}, \mathbf{R} \in \mathbb{R}^{c\times c'}$ are tall matrices with orthonormal columns.

Note. The projected images will be given by

$$\mathbf{Y}_i = \mathbf{L}^T\mathbf{X}_i\mathbf{R} \in \mathbb{R}^{r'\times c'}, \quad \forall\, i$$

Matrix trace and it properties

The *trace* of a square matrix is the sum of its diagonals: $\operatorname{trace}(\mathbf{A}) = \sum_i a_{ii}$.

It has the following properties:

- Linearity (for matrices of same size): $\operatorname{trace}\left(\sum_i \alpha_i \mathbf{A}_i\right) = \sum_i \alpha_i \operatorname{trace}(\mathbf{A}_i)$
- Trace-commutativity:

$$\operatorname{trace}(\mathbf{AB}) = \operatorname{trace}(\mathbf{BA}) \quad \text{(whenever both are defined)}$$

- Relation to Frobenius norm

$$\|\mathbf{A}\|_{\mathrm{F}}^2 = \operatorname{trace}(\mathbf{A}\mathbf{A}^T) = \operatorname{trace}(\mathbf{A}^T\mathbf{A})$$

Rewriting the problem

Using the trace properties we first rewrite the within-class scatter as follows

$$\begin{aligned}
&\sum_i \sum_{\mathbf{X}\in\Pi_i} \|\mathbf{L}^T(\mathbf{X}-\mathbf{M}_i)\mathbf{R}\|_{\mathrm{F}}^2 \\
&= \sum_i \sum_{\mathbf{X}\in\Pi_i} \mathrm{trace}\left(\mathbf{L}^T(\mathbf{X}-\mathbf{M}_i)\mathbf{R}\mathbf{R}^T(\mathbf{X}-\mathbf{M}_i)^T\mathbf{L}\right) \\
&= \mathrm{trace}\left(\sum_i \sum_{\mathbf{X}\in\Pi_i} \mathbf{L}^T(\mathbf{X}-\mathbf{M}_i)\mathbf{R}\mathbf{R}^T(\mathbf{X}-\mathbf{M}_i)^T\mathbf{L}\right)
\end{aligned}$$

Note that $\mathbf{L}^T$ and $\mathbf{L}$ may be factored out of the double summation (but still within the trace operator).

Similarly, for the between-class scatter,

$$
\begin{aligned}
&\sum_i n_i \|\mathbf{L}^T(\mathbf{M}_i - \mathbf{M})\mathbf{R}\|_{\mathrm{F}}^2 \\
&= \operatorname{trace}\left(\sum_i n_i \mathbf{L}^T(\mathbf{M}_i - \mathbf{M})\mathbf{R}\mathbf{R}^T(\mathbf{M}_i - \mathbf{M})^T\mathbf{L}\right)
\end{aligned}
$$

The 2DLDA problem now becomes

$$
\max_{\mathbf{L},\mathbf{R}} \frac{\operatorname{trace}\left(\sum_i n_i \mathbf{L}^T(\mathbf{M}_i - \mathbf{M})\mathbf{R}\mathbf{R}^T(\mathbf{M}_i - \mathbf{M})^T\mathbf{L}\right)}{\operatorname{trace}\left(\sum_i \sum_{\mathbf{X}\in\Pi_i} \mathbf{L}^T(\mathbf{X} - \mathbf{M}_i)\mathbf{R}\mathbf{R}^T(\mathbf{X} - \mathbf{M}_i)^T\mathbf{L}\right)}
$$

Solving the problem

The joint optimization problem over $\mathbf{L}, \mathbf{R}$ is very difficult to solve.

We consider a special case when $\mathbf{R}$ is given. The problem reduces to

$$\max_{\mathbf{L}} \frac{\operatorname{trace}\left(\mathbf{L}^T \mathbf{S}_b^{\mathbf{R}} \mathbf{L}\right)}{\operatorname{trace}\left(\mathbf{L}^T \mathbf{S}_w^{\mathbf{R}} \mathbf{L}\right)}$$

where

$$\mathbf{S}_w^{\mathbf{R}} = \sum_i \sum_{\mathbf{X} \in \Pi_i} (\mathbf{X} - \mathbf{M}_i)\mathbf{R}\mathbf{R}^T (\mathbf{X} - \mathbf{M}_i)^T \in \mathbb{R}^{r \times r}$$

$$\mathbf{S}_b^{\mathbf{R}} = \sum_i n_i (\mathbf{M}_i - \mathbf{M})\mathbf{R}\mathbf{R}^T (\mathbf{M}_i - \mathbf{M})^T \in \mathbb{R}^{r \times r}$$

The solution of this reduced problem is often approximated by the first c' generalized eigenvectors

$$\mathbf{S}_b^{\mathbf{R}}\,\mathbf{l}_i = \lambda_i\,\mathbf{S}_w^{\mathbf{R}}\,\mathbf{l}_i$$

or eigenvectors

$$\left(\mathbf{S}_w^{\mathbf{R}}\right)^{-1}\mathbf{S}_b^{\mathbf{R}}\,\mathbf{l}_i = \lambda_i\mathbf{l}_i$$

Remarks:

- Both matrices $\mathbf{S}_w^{\mathbf{R}}, \mathbf{S}_b^{\mathbf{R}}$ have the size of $r \times r$, and thus are much smaller than their counterparts in FDA which have a size of $d \times d$ with $d = rc$. Therefore, this problem is much easier to solve numerically.

- In general $\mathbf{S}_w^{\mathbf{R}}$ is nonsingular, so the singularity issue with FDA does not exist in 2DLDA.

Similarly, if $\mathbf{L}$ is given to us, then the problem maybe written as

$$\max_{\mathbf{R}} \frac{\operatorname{trace}\left(\sum_i n_i \mathbf{R}^T(\mathbf{M}_i - \mathbf{M})^T \mathbf{L}\mathbf{L}^T(\mathbf{M}_i - \mathbf{M})\mathbf{R}\right)}{\operatorname{trace}\left(\sum_i \sum_{\mathbf{X}\in\Pi_i} \mathbf{R}^T(\mathbf{X} - \mathbf{M}_i)^T \mathbf{L}\mathbf{L}^T(\mathbf{X} - \mathbf{M}_i)\mathbf{R}\right)} = \frac{\operatorname{trace}\left(\mathbf{R}^T \mathbf{S}_b^{\mathbf{L}} \mathbf{R}\right)}{\operatorname{trace}\left(\mathbf{R}^T \mathbf{S}_w^{\mathbf{L}} \mathbf{R}\right)}$$

where

$$\mathbf{S}_w^{\mathbf{L}} = \sum_i \sum_{\mathbf{X}\in\Pi_i} (\mathbf{X} - \mathbf{M}_i)^T \mathbf{L}\mathbf{L}^T(\mathbf{X} - \mathbf{M}_i) \in \mathbb{R}^{c\times c},$$
$$\mathbf{S}_b^{\mathbf{L}} = \sum_i n_i(\mathbf{M}_i - \mathbf{M})^T \mathbf{L}\mathbf{L}^T(\mathbf{M}_i - \mathbf{M}) \in \mathbb{R}^{c\times c}.$$

The approximate solution is given by the first few eigenvectors of $\left(\mathbf{S}_w^{\mathbf{L}}\right)^{-1} \mathbf{S}_b^{\mathbf{L}} \in \mathbb{R}^{c\times c}$.

Algorithm for 2DLDA

The previous discussions motivate us to solve the 2DLDA problem using an iterative procedure:

1. Initialize $\mathbf{R} = \begin{pmatrix} \mathbf{I}_{c' \times c'} \\ \mathbf{0}_{(c-c') \times c'} \end{pmatrix} \in \mathbb{R}^{c \times c'}$
2. Iterative until convergence:
   - $\mathbf{L} \longleftarrow$ top r' eigenvectors of $\left(\mathbf{S}_w^{\mathbf{R}}\right)^{-1} \mathbf{S}_b^{\mathbf{R}}$
   - $\mathbf{R} \longleftarrow$ top c' eigenvectors of $\left(\mathbf{S}_w^{\mathbf{L}}\right)^{-1} \mathbf{S}_b^{\mathbf{L}}$
3. Return final versions of $\mathbf{L}$ and $\mathbf{R}$

MATLAB code for 2DLDA

2DLDA is not implemented in MATLAB.

However, there is a toolbox available at MATLAB File Exchange:

`http://www.mathworks.com/matlabcentral/fileexchange/20174-2dlda-pk-lda-for-feature-extraction`

The function to use is

[R, L] = iterative2DLDA(trainImages, trainLabels+1, 10, 10, 28, 28)

% Columns are images
% Labels must start at 1

Ways of using 2DLDA

Like FDA, 2DLDA is a supervised dimensionality reduction methods. It has the following usage:

- 2DLDA + a classifier (e.g., kNN, $k$means, LDA/QDA, Naive Bayes)
- 2DLDA + FDA + a classifier

Comparison between FDA and 2DLDA

Both are supervised methods aiming to preserve discriminatory information.

- 2DLDA is more flexible (can project data down to any size $r' \times c'$)
- 2DLDA does not have the singularity issue (no PCA needed)
- 2DLDA is harder to solve (as it has two matrices to choose, so that we can only use alternating optimization) but individual linear algebra problems are much easier to solve (as the scatter matrices are smaller)

HW3b (due Friday noon, March 18)

First use 2DLDA to transform the MNIST handwritten digits to a smaller size (9×9, or a better pair that you find for the particular classifiers used below), and then do the following.

4. Perform kNN classification with three different distance metrics: Euclidean, city block and cosine of the angle. You may use a fixed $k = 3$ as before, but if you have more time, try different values of k (e.g., from 1 to 10). Plot the results in a bar graph (if single k) or as three curves (if multiple k). How does 2DLDA compare with PCA in terms of dimensionality reduction for kNN?

5. Apply LDA to the images transformed by 2DLDA. How does this combination perform (by comparing with PCA + LDA)?

Midterm project 3: 2DLDA

Interested students please discuss with me your ideas.

Further learning

- Symmetric 2DLDA
  http://ranger.uta.edu/~chqding/papers/Symmetric2DLDA.pdf

- Nonparametric Discriminant Analysis (NDA)
  http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4775283