67577 – Intro. to Machine Learning **Fall semester, 2008/9**

Lecture 8: Spectral Analysis I: PCA, LDA, CCA

Lecturer: Amnon Shashua *Scribe: Amnon Shashua* [1]

In this lecture (and the following one) we will focus on *spectral methods* for learning. Today we will focus on dimensionality reduction using Principle Component Analysis (PCA), multi-class learning using Linear Discriminant Analysis (LDA) and Canonical Correlation Analysis (CCA). In the next lecture we will focus on spectral clustering methods.

Dimensionality reduction appears when the dimension of the input vector is very large (imagine pixels in an image, for example) while the coordinate measurements are highly inter-dependent (again, imagine the redundancy present among neighboring pixels in an image). High dimensional data impose computational efficiency challenges and often translate to poor generalization abilities of the learning engine (see lectures on PAC). A dimensionality reduction can also be viewed as a *feature extraction* process where one takes as input a large feature set (the original measurements) and creates from them a much smaller number of new features which are then fed into the learning engine.

In this lecture we will focus on feature extraction from a very specific (and constrained) standpoint. We would be looking for a mixing (linear combination) of the input coordinates such that we obtain a linear projection from R^n to R^q for some $q < n$. In doing so we wish to reduce the redundancy while preserving as much as possible the variance of the data. From a statistical standpoint this is achieved by transforming to a new set of variables, called principal components, which are uncorrelated so that the first few retain most of the variation present in all of the original coordinates. For example, in an image processing application the input images are highly redundant where neighboring pixel values are highly correlated. The purpose of feature extraction would be to transform the input image into a vector of output components with the least redundancy possible. Form a geometric standpoint, this is achieved by finding the "closest" (in least squares sense) linear q-dimensional susbspace to the m sample points S. The new subspace is a lower dimensional "best approximation" to the sample S. These two, equivalent, perspectives on data compression (dimensionality reduction) form the central idea of *principal component analysis* (PCA) which probably the oldest (going back to Pearson 1901) and best known of the techniques of multivariate analysis in statistics. The computation of PCA is very simple and the definition is straightforward, but has a wide variety of different applications, a number of different derivations, quite a number of different terminologies (especially outside the statistical literature) and is the basis for quite a number of variations on the basic technique.

We then extend the variance preserving approach for data representation for *labeled* data sets. We will describe the linear classifier approach (separating hyperplane) form the point of view of looking for a hyperplane such that when the data is projected onto it the separation is maximized (the distance between the class means is maximal) and the data within each class is compact (the variance/spread is minimized). The solution is also produced, just like PCA, by a spectral analysis of the data. This approach goes under the name of Fisher's Linear Discriminant Analysis (LDA).

What is common between PCA and LDA is (i) the use of spectral matrix analysis — i.e.,

[1]

what can you do with eigenvalues and eigenvectors of matrices representing subspaces of the data? (ii) these techniques produce optimal results for *normally distributed* data and are very easy to implement. There is a large variety of uses of spectral analysis in statistical and learning literature including spectral clustering, Multi Dimensional Scaling (MDS) and data modeling in general. Another point to note is that this is the first time in the course where the type of data distribution plays a role in the analysis — the two techniques are defined for any distribution but are optimal only under the Gaussian distribution.

We will also describe a non-linear extension of PCA known as Kernel-PCA, but the focus would be mostly on PCA itself and its analysis from a couple of vantage points: (i) PCA as an optimal reconstruction after a dimension reduction, i.e., data compression, and (ii) PCA for redundancy reduction (decorrelation) of the output components.

8.1 PCA: Statistical Perspective

Let $\mathbf{x}_1, ..., \mathbf{x}_m \in R^n$ be our sample data S of vectors in R^n, arranged as columns of a matrix A. It will be convenient to assume that the data is centered, i.e., $\sum \mathbf{x}_i = 0$. If the data is not centered we can always center it by computing the mean vector $\mu = (1/m)\sum_i \mathbf{x}_i$ and replace the original data sample with the new sample $\mathbf{x}_i - \mu$. In a statistical sense, the coordinates of the vector $\mathbf{x} \in R^n$ are considered as random variables, thus a row in the matrix A is the sample of values of a particular random variable, drawn from some unknown probability distribution, associated with the row position. We wish to find vectors $\mathbf{u}_1, ..., \mathbf{u}_q$ (arranged as columns of a matrix U), where $q \leq min(n, m)$, such that the new feature measurements $\mathbf{y} = U^\top \mathbf{x}$ (who are the result of linear combinations $\mathbf{u}_1^\top \mathbf{x}, ..., \mathbf{u}_q^\top \mathbf{x}$ of the original feature measurements $\mathbf{x}$) have certain desirable properties.

The idea property to seek from the new coordinates $\mathbf{y}$ is statistical independence, i.e., $P(y_1, .., y_q) = P(y_1) \cdots P(y_q)$ which would mean that we have removed the redundancy of the original data $\mathbf{x}$ in the best possible manner. This goal, however, is too much to ask from a linear transformation and instead we would ask for a weaker property to hold: that the pairwise covariance $cov(y_i, y_j) = 0$ vanishes, i.e., that the covariance matrix on the new coordinates is diagonal. A diagonal covariance insures some redundancy removal, but not as good as statistical independence. However, when the data is Normally distributed $P(\mathbf{x}) \sim N(\mu, \Sigma)$ with mean μ and covariance Σ, then the transformation which diagonalizes the covariance matrix also guarantees statistical independence. Among all transformations that de-correlate the data we will seek the one that maximizes the *spread* (variance) of the sample data after being projected onto the new axes vectors.

8.1.1 Maximizing the Variance of Output Coordinates

The property we would like to maximize is that the projection of the sample data on the new axes is as *spread* as possible. To start this analysis, assume $q = 1$, i.e., the n components of the input vector $\mathbf{x}$ are reduced to a single output component $y = \mathbf{u}^\top \mathbf{x}$. We are looking for a single vector $\mathbf{u} \in R^n$ whose direction *maximizes the variance* of the output component y.

Formally, we are looking for a unit vector $\mathbf{u}$ which maximizes $\sum_i (\mathbf{u}^\top \mathbf{x}_i)^2$ (see Appendix A for basic statistical definitions and note that $E[y] = 0$ because $\sum_i \mathbf{u}^\top \mathbf{x}_i = \mathbf{u}_i^\top (\sum_i \mathbf{x}_i) = 0$). In other words, the projected points onto the axis represented by the vector $\mathbf{u}$ are as spread as possible (in a least squares sense). In vector notation, the optimization problem takes the following form:

$$\max_{\mathbf{u}} \frac{1}{2} \|\mathbf{u}^\top A\|^2 \qquad \textit{subject to} \qquad \frac{1}{2} \mathbf{u}^\top \mathbf{u} = 1$$

The Lagrangian of the problem is:

$$L(\mathbf{u}, \lambda) = \frac{1}{2}\mathbf{u}^\top AA^\top \mathbf{u} - \lambda(\frac{1}{2}\mathbf{u}^\top \mathbf{u} - 1)$$

By taking the partial derivative $\partial L/\partial \mathbf{u} = 0$ we obtain the following necessary condition (see Appendix B):

$$AA^\top \mathbf{u} = \lambda \mathbf{u},$$

which tells us that $\mathbf{u}$ is an eigenvector of the $n \times n$ (symmetric and positive definite) matrix $AA^\top$. There are n eigenvectors associated with $AA^\top$ and we can easily convince ourselves that we are looking for the one associated with the maximal eigenvalue: substitute $\lambda\mathbf{u}$ instead of $AA^\top\mathbf{u}$ in the criterion function $\mathbf{u}^\top AA^\top \mathbf{u}$ to obtain $\lambda(\mathbf{u}^\top\mathbf{u}) = \lambda$ and since the eigenvalues must be positive (since $AA^\top$ is positive definite), then the optimum is obtained for the maximal eigenvalue. The leading eigenvector $\mathbf{u}$ of $AA^\top$ is called the *first principal axis* of the data sample represented by the columns of the matrix A, and $y = \mathbf{u}^\top\mathbf{x}$ is called the first *principal component* of the data sample.

For convenience, we denote $\mathbf{u}_1 = \mathbf{u}$ and $\lambda_1 = \lambda$ as the leading eigenvector and eigenvalue of $AA^\top$. Next, we look for $y_2 = \mathbf{u}_2^\top\mathbf{x}$ which is *uncorrelated* with $y_1 = \mathbf{u}_1^\top\mathbf{x}$ and which has maximum variance (and so on for $\mathbf{u}_3, ..., \mathbf{u}_q$). Two random variables are uncorrelated if their covariance vanishes. By definition of covariance (see Appendix A) we obtain:

$$\begin{aligned} Cov(y_1 y_2) &= \sum_i (\mathbf{u}_1^\top \mathbf{x}_i)(\mathbf{u}_2^\top \mathbf{x}_i) = \mathbf{u}_1^\top (\sum_i \mathbf{x}_i \mathbf{x}_i^\top)\mathbf{u}_2 \\ &= \mathbf{u}_1^\top AA^\top \mathbf{u}_2 = \mathbf{u}_2^\top AA^\top \mathbf{u}_1 = \lambda_1 \mathbf{u}_1^\top \mathbf{u}_2 = 0 \end{aligned}$$

We can therefore use the condition $\mathbf{u}_1^\top\mathbf{u}_2 = 0$ to specify zero correlation between y_1, y_2. The functional to be optimized becomes:

$$\max_{\mathbf{u}_2} \frac{1}{2}\|\mathbf{u}_2^\top A\|^2 \qquad \textit{subject to} \qquad \frac{1}{2}\mathbf{u}_2^\top\mathbf{u}_2 = 1, \quad \mathbf{u}_1^\top\mathbf{u}_2 = 0,$$

with the Lagrangian being:

$$L(\mathbf{u}_2, \lambda, \delta) = \frac{1}{2}\mathbf{u}_2^\top AA^\top \mathbf{u}_2 - \lambda(\frac{1}{2}\mathbf{u}_2^\top\mathbf{u}_2 - 1) - \delta\mathbf{u}_1^\top\mathbf{u}_2.$$

By taking the partial derivative with respect to $\mathbf{u}_2$ we obtain the necessary condition:

$$AA^\top\mathbf{u}_2 - \lambda\mathbf{u}_2 - \delta\mathbf{u}_1 = 0.$$

Multiply the equation by $\mathbf{u}_1$ from the left:

$$\mathbf{u}_1^\top AA^\top\mathbf{u}_2 - \lambda\mathbf{u}_1^\top\mathbf{u}_2 - \delta\mathbf{u}_1^\top\mathbf{u}_1 = 0,$$

and noting from above that $\mathbf{u}_1^\top AA^\top\mathbf{u}_2 = \mathbf{u}_1^\top\mathbf{u}_2 = 0$ we obtain $\delta = 0$. As a result we obtain:

$$AA^\top\mathbf{u}_2 = \lambda\mathbf{u}_2,$$

so once more we have that $\lambda, \mathbf{u}_2$ form an eigenvalue/eigenvector pair of $AA^\top$. As before, λ should be as large as possible from the remaining spectral decomposition. By induction, it can be shown that the remaining principal vectors $\mathbf{u}_3, ..., \mathbf{u}_q$ are the decreasing order eigenvactors of $AA^\top$ and the variance of the i'th principal component $y_i = \mathbf{u}_i^\top\mathbf{x}$ is λ_i.

Taken together, the PCA is the solution of the following optimization problem:

$$\max_{\mathbf{u}_1,...,\mathbf{u}_q} \frac{1}{2}\sum_i \|\mathbf{u}_i^\top A\|^2 \qquad \textit{subject to} \quad \mathbf{u}_i^\top \mathbf{u}_i = 1, \quad \mathbf{u}_i^\top \mathbf{u}_j = 0, \quad i \neq j = 1, ..., q.$$

It will be useful for later to write the optimization function in a more concise manner as follows. Let U be the $n \times q$ matrix whose columns are $\mathbf{u}_i$ and $D = diag(\lambda_1, ..., \lambda_q)$ is an $q \times q$ diagonal matrix and $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_q$. Then from above we have that $U^\top U = I$ and $AA^\top U = UD$. Using the fact that $trace(\mathbf{x}\mathbf{y}^\top) = \mathbf{x}^\top \mathbf{y}$, $trace(AB) = trace(BA)$ and $trace(A+B) = trace(A) + trace(B)$ we can convert $\sum_i \|\mathbf{u}_i^\top A\|^2$ to $trace(U^\top AA^\top U)$ as follows:

$$\begin{aligned}\sum_i \mathbf{u}_i^\top AA^\top \mathbf{u}_i &= \sum_i trace(A^\top \mathbf{u}_i \mathbf{u}_i^\top A) = trace(A^\top (\sum_i \mathbf{u}_i \mathbf{u}_i^\top) A) \\ &= trace(A^\top UU^\top A) = trace(U^\top AA^\top U)\end{aligned}$$

Thus, PCA becomes the solution of the following optimization function:

$$\max_{U \in R^{n \times q}} trace(U^\top AA^\top U) \qquad \textit{subject to} \quad U^\top U = I. \tag{8.1}$$

The solution, as saw above, is that $U = [\mathbf{u}_1, ..., \mathbf{u}_q]$ consists of the decreasing order eigenvectors of $AA^\top$. At the optimum, $trace(U^\top AA^\top U)$ is equal to $trace(D)$ which is equal to the sum of eigenvalues $\lambda_1 + ... + \lambda_q$.

It is worthwhile noting that when $q = n$, $UU^\top = U^\top U = I$, and the PCA transform is a change of basis in R^n known as Karhunen-Loeve transform.

To conclude, the PCA transform looks for q orthogonal direction vectors (called the principal axes) such that the projection of input sample vectors onto the principal directions has the maximal spread, or equivalently that the variance of the output coordinates $\mathbf{y} = U^\top \mathbf{x}$ is maximal. The principal directions are the leading (with respect to descending eigenvalues) q eigenvectors of the matrix $AA^\top$. When $q = n$, the principal directions form a basis of R^n with the property of de-correlating the data and maximizing the variance of the coordinates of the sample input vectors.

8.1.2 Decorrelation: Diagonalization of the Covariance Matrix

In the previous section we saw that PCA generates a new coordinate system $\mathbf{y} = U^\top \mathbf{x}$ where the coordinates $y_1, ..., y_q$ of $\mathbf{x}$ in the new system are *uncorrelated*. This means that the covariance matrix over the principle components should be diagonal. In this section we will explore this perspective in more detail.

The covariance matrix $\mathbf{\Sigma}_x$ of the sample data $\mathbf{x}_1, ..., \mathbf{x}_m$ with zero mean is

$$(1/m)\sum_i \mathbf{x}_i \mathbf{x}_i^\top = (1/m)AA^\top,$$

therefore the matrix $AA^\top$ we derived above is a scaled version of the covariance of the sample data (see Appendix A). The scale factor $1/m$ was unimportant in the process above because the eigenvectors are of unit norm, thus any scale of $AA^\top$ would produce the same set of eigenvectors.

The off-diagonal entries of the covariance matrix $\mathbf{\Sigma}_x$ represent the correlation (a measure of statistical dependence) between the i'th and j'th component vectors, i.e., the entries of the input vectors $\mathbf{x}$. The existence of correlations among the components (features) of the input signal is a sign of redundancy, therefore from the point of view of transforming the input representation

into one which is *less* redundant, we would like to find a transformation $\mathbf{y} = U^\top \mathbf{x}$ with an output representation $\mathbf{y}$ which is associated with a diagonal covariance matrix $\mathbf{\Sigma}_y$, i.e., the components of $\mathbf{y}$ are uncorrelated.

Formally, $\mathbf{\Sigma}_y = (1/m)\sum_i \mathbf{y}_i \mathbf{y}_i^\top = (1/m) U^\top A A^\top U$, therefore we wish to find an $n \times q$ matrix for which $U^\top A A^\top U$ is diagonal. If in addition, we would require that the *variance* of the output coordinates is maximized, i.e., $trace(U^\top A A^\top U)$ is maximal (but then we need to constrain the length of the column vectors of U, i.e., set $\|\mathbf{u}_i\| = 1$) then we would get a unique solution for U where the columns are orthonormal and are defined as the first q eigenvectors of the covariance matrix $\mathbf{\Sigma}_x$. This is exactly the optimization problem defined by eqn. (8.1).

We see therefore that PCA "decorrelates" the input data. Decorrelation and statistical independence are not the same thing. If the coordinates are statistically independent then the covariance matrix is diagonal[2], but it does not follow that uncorrelated variables must be statistically independent — covariance is just one measure of dependence. In fact, the covariance is a measure of pairwise dependency only. However, it is a fact that uncorrelated variables are statistically independent if they have a multivariate normal distribution (a Gaussian). In other words, if the sample data $\mathbf{x}$ are drawn from a probability distribution $p(\mathbf{x})$ which has Gaussian form, the PCA transforms the sample data into a statistically independent set of variables $\mathbf{y} = U^\top \mathbf{x}$. The details are explained below.

Recall that a multivariate normal distribution of the random variables $\mathbf{x} = (x_1, ..., x_n)^\top$ is defined as $p(\mathbf{x}) \approx N(\mu, \mathbf{\Sigma})$:

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{n/2}|\mathbf{\Sigma}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\mu)^\top \mathbf{\Sigma}^{-1}(\mathbf{x}-\mu)}.$$

Also recall that a linear combination of the variables produces also a normal distribution $N(U^\top \mu, U^\top \mathbf{\Sigma} U)$:

$$\mathbf{\Sigma}_y = \sum_{\mathbf{y}} (\mathbf{y}-\mu_y)(\mathbf{y}-\mu_y)^\top = \sum_{\mathbf{x}} (U^\top \mathbf{x} - U^\top \mu_x)(U^\top \mathbf{x} - U^\top \mu_x)^\top = U^\top \mathbf{\Sigma}_x U,$$

therefore choose U such that $\mathbf{\Sigma}_y = U^\top \mathbf{\Sigma} U$ is a diagonal matrix $\mathbf{\Sigma}_y = diag(\sigma_1^2, ..., \sigma_n^2)$. We have in that case:

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{n/2}\prod_i \sigma_i} e^{-\frac{1}{2}\sum_i \left(\frac{x_i-\mu_i}{\sigma_i}\right)^2}$$

which can be written as a product of univariate normal distributions $p_{x_i}(x_i)$:

$$p(\mathbf{x}) = \prod_{i=1}^{n} \frac{1}{(2\pi)^{1/2}\sigma_i} e^{-\frac{1}{2}\left(\frac{x_i-\mu_i}{\sigma_i}\right)^2} = \prod_{i=1}^{n} p_{x_i}(x_i),$$

which proves the assertion that decorrelated normally distributed variables are statistically independent.

8.2 PCA: Optimal Reconstruction

A different, yet equivalent, perspective on the PCA transformation is as an optimal reconstruction (in a least squares sense) after a dimension reduction. We are given a sample data as before $\mathbf{x}_1, ..., \mathbf{x}_m$ and we are looking for a *small* number of orthonormal principal vectors $\mathbf{u}_1, ..., \mathbf{u}_q$ where

[2] $\sigma_{xy} = \sum_x \sum_y (x-\mu_x)(y-\mu_y)p(x,y) = \sum_x \sum_y (x-\mu_x)(y-\mu_y)p(x)(p(y) = (\sum_x (x-\mu_x)p(x))(\sum_y (y-\mu_y)p(y)) = 0$

$q < min(n, k)$ which define a q-dimensional linear subspace of R^n which *best* approximate the original input vectors in a least squares sense. In other words, the projection $\hat{\mathbf{x}}_i$ of the sample points $\mathbf{x}_i$ onto the q-dimensional subspace should minimize $\sum_i \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|^2$ over all possible q-dimensional subspaces of R^n.

Let $\mathcal{U}$ be the subspace spanned by the principal vectors (columns of U) and let P be the $n \times n$ projection matrix mapping a point $\mathbf{x} \in R^n$ onto its projection $\hat{\mathbf{x}} \in \mathcal{U}$. From the definition of projection, the vector $\mathbf{x} - \hat{\mathbf{x}}$ must be orthogonal to the subspace $\mathcal{U}$. Let $\mathbf{y} = (y_1, ..., y_q)$ be the coordinates of $\hat{\mathbf{x}}$ with respect to the principal vectors, i.e., $U\mathbf{y} = \hat{\mathbf{x}}$. Then, from orthogonality we have that $(\mathbf{x} - U\mathbf{y})^\top U\mathbf{w} = 0$ for all vectors $\mathbf{w} \in R^n$. Since this is true for all $\mathbf{w}$ then $U^\top U\mathbf{y} - U^\top \mathbf{x} = 0$. Therefore, $\mathbf{y} = (U^\top U)^{-1} U^\top \mathbf{x}$ and as a result the projection matrix P becomes:

$$P = U(U^\top U)^{-1} U^\top,$$

satisfying $P\mathbf{x} = \hat{\mathbf{x}}$. In the case the columns of U are orthonormal, $U^\top U = I$, we have $P = UU^\top$. We are ready now to describe the optimization problem on U: we wish to find an orthonormal set of principal vectors, $U^\top U = I$, such that $\sum_i \|\mathbf{x}_i - UU^\top \mathbf{x}_i\|^2$ is minimized.

Note that $\sum_i \|\mathbf{x}_i - UU^\top \mathbf{x}_i\|^2 = \|A - UU^\top A\|_F^2$ where $\|B\|_F^2 = \sum_{i,j} b_{ij}^2$ is the square *Frobenious* norm of a matrix. The optimal reconstruction problem therefore becomes:

$$\min_U \|A - UU^\top A\|_F^2 \qquad \textit{subject to} \quad U^\top U = I.$$

We will show now that:

$$\text{argmin}_U \|A - UU^\top A\|_F^2 = \text{argmax } trace(U^\top AA^\top U),$$

which shows that the optimal reconstruction problem is solved by PCA (recall Eqn. 8.1).

From the identity $\|B\|_F^2 = trace(BB^\top)$, we have:

$$\|A - UU^\top A\|_F^2 = trace((A - UU^\top A)(A - UU^\top A)^\top).$$

Expanding the right hand side gives us:

$$\begin{aligned} trace((A - UU^\top A)(A - UU^\top A)^\top) &= trace(AA^\top) - trace(AA^\top UU^\top) \\ &- trace(UU^\top AA^\top) + trace(UU^\top AA^\top UU^\top) \end{aligned}$$

The second and third term are equal (commutativity of trace) and is also equal to the 4th term due to commutativity of the trace and $U^\top U = I$. Taken together:

$$\|A - UU^\top A\|_F^2 = trace(AA^\top) - trace(U^\top AA^\top U).$$

To conclude, we have proven that by taking the first q eigenvectors of $AA^\top$ we obtain a linear subspace which is *as close as possible* (in a least squares sense) to the original sample data. Hence, PCA can be viewed as a vehicle for optimal reconstruction after dimension reduction. The optimization problem whose solution is the leading q eigenvectors of $AA^\top$ is described in eqn. 8.1:

$$\max_{U \in R^{n \times q}} trace(U^\top AA^\top U) \qquad \textit{subject to} \quad U^\top U = I.$$

8.3 The Case $n >> m$

Consider the situation where n, the dimension of the input vectors, is relatively large compared to the number of sample vectors m. For example, consider input vectors representing 50×50 sized images of faces, i.e., $n = 2500$, where $m = 100$. In other words, we are looking for a small number of "face templates" (known as "eigenfaces") which approximate well the original set of 100 face images. In this case, $AA^\top$ is very large, 2500×2500, yet the number of non-vanishing eigenvalues cannot be higher than 100. Given that the eigendecomposition process is $O(2500^3)$, the computational burden would be very high. However, it is possible to perform an eigendecomposition on $A^\top A$ (a 100×100 matrix) instead, as shown next.

Let the columns of Q be the first $q < m$ eigenvectors of $A^\top A$, i.e., $A^\top AQ = QD$ where D is diagonal containing the corresponding eigenvalues. After pre-multiplying both sides by A we obtain:

$$AA^\top(AQ) = (AQ)D,$$

from which we conclude that AQ contains the first q eigenvectors (but un-normalized) of $AA^\top$. We have therefore that $U = AQD^{-\frac{1}{2}}$ because:

$$U^\top U = D^{-\frac{1}{2}}Q^\top A^\top AQD^{-\frac{1}{2}} = D^{-\frac{1}{2}}DD^{-\frac{1}{2}} = I,$$

where we used the fact that $Q^\top A^\top AQ = D$. Note that eigenvalues of $A^\top A$ and $AA^\top$ are the same (because $AA^\top(AQD^{-\frac{1}{2}}) = (AQD^{-\frac{1}{2}})D$).

8.4 Kernel PCA

We can take the case $n >> m$ described in the previous section one step further and consider such large values of n which are practically uncomputable — a situation which results when mapping the original input vectors to a high dimensional space: $\phi(\mathbf{x})$ where $\phi : R^n \to \mathcal{F}$ for which $dim(\mathcal{F}) >> n$. For example, $\phi(\mathbf{x})$ representing the d'th order monomials of the coordinates of $\mathbf{x}$, i.e., $dim(\mathcal{F}) = \binom{n+d-1}{d}$ which is exponential in d. The mappings of interest are those which are paired with a non-linear kernel function: $k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^\top \phi(\mathbf{x}')$ (see Lecture 5).

Performing PCA on $A = [\phi(\mathbf{x}_1), ..., \phi(\mathbf{x}_m)]$ is equivalent to finding the non-linear surface in R^n (the nature of the non-linearity depends on the choice of $\phi()$) which best approximates the original sample data $\mathbf{x}_1, ..., \mathbf{x}_m$. The problem is that $AA^\top$ is not computable — however $A^\top A$ is computable because $(A^\top A)_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$.

From the previous section, $U = AQD^{-\frac{1}{2}} = AV$ contains the first q eigenvectors of $AA^\top$ (where Q and D are computable). Since A itself is not computable we cannot represent U explicitly, but we can project a new vector $\phi(\mathbf{x})$ onto the principal directions $\mathbf{u}_1, ..., \mathbf{u}_q$ and obtain the principal components, i.e., the output vector $\mathbf{y} = U^\top \phi(\mathbf{x})$, as follows.

$$\mathbf{y} = U^\top \phi(\mathbf{x}) = V^\top A^\top \phi(\mathbf{x}) = V^\top \begin{pmatrix} k(\mathbf{x}_1, \mathbf{x}) \\ . \\ . \\ . \\ k(\mathbf{x}_m, \mathbf{x}) \end{pmatrix}.$$

Given the principal components (entries of $\mathbf{y} = U^\top \phi(\mathbf{x})$ of $\phi(\mathbf{x})$) we can measure, for example, the *distance* between $\phi(\mathbf{x})$ and the projection $\hat{\phi(\mathbf{x})} = UU^\top \phi(\mathbf{x}) = U\mathbf{y}$ onto the linear subspace spanned

by $\mathbf{u}_1, ..., \mathbf{u}_q$ (without the need to explicitly compute the principal axes $\mathbf{u}_i$), as follows.

$$\begin{aligned}\|\phi(\mathbf{x}) - \hat{\phi(\mathbf{x})}\|^2 &= \phi(\mathbf{x})^\top \phi(\mathbf{x}) + \hat{\phi(\mathbf{x})}^\top \hat{\phi(\mathbf{x})} - 2\phi(\mathbf{x})^\top \hat{\phi(\mathbf{x})} \\ &= k(\mathbf{x},\mathbf{x}) + \mathbf{y}^\top U^\top U\mathbf{y} - 2\phi(\mathbf{x})^\top (UU^\top \phi(\mathbf{x})) \\ &= k(\mathbf{x},\mathbf{x}) - \mathbf{y}^\top \mathbf{y} - 2\mathbf{y}^\top \mathbf{y} \\ &= k(\mathbf{x},\mathbf{x}) - \|\mathbf{y}\|^2\end{aligned}$$

8.5 Fisher's LDA: Basic Idea

We now extend the variance preserving approach for data representation for *labeled* data sets. We will focus on 2-class sets and look for a separating hyperplane:

$$f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + b,$$

such that $\mathbf{x}$ belongs to the first class if $f(\mathbf{x}) > 0$ and $\mathbf{x}$ belongs to the second class if $f(\mathbf{x}) < 0$. In the statistical literature this type of function is called a *linear discriminant function*. The decision boundary is given by the set of points satisfying $f(\mathbf{x}) = 0$ which is a hyperplane. Fisher's (1936) Linear Discriminant Analysis (LDA) is a variance preserving approach for finding a linear discriminant function.

We will then introduce another popular statistical technique called Canonical Correlation Analysis (CCA) for learning the mapping between input and output vectors using the notion "angle" between subspaces.

What is common in the three techniques PCA, LDA and CCA is the use of spectral matrix analysis — i.e., what can you do with eigenvalues and eigenvectors of matrices representing subspaces of the data? These techniques produce optimal results for normally distributed data and are very easy to implement. There is a large variety of uses of spectral analysis in statistical and learning literature including spectral clustering, Multi Dimensional Scaling (MDS) and data modeling in general.

To appreciate the general idea behind Fisher's LDA consider Fig. 8.1. Let the centers of classes one and two be denoted by μ_1 and μ_2 respectively. A linear discriminant function is a projection onto a 1D subspace such that the classes would be separated the most in the 1D subspace. The obvious first step in this kind of analysis is to make sure that the projected centers $\hat{\mu}_1, \hat{\mu}_2$ would be separated as much as possible. We can easily see that the direction of the 1D subspace should be proportional to $\mu_1 - \mu_2$ as follows:

$$(\hat{\mu}_1 - \hat{\mu}_2)^2 = \left(\frac{\mathbf{w}^\top \mu_1}{\|\mathbf{w}\|} - \frac{\mathbf{w}^\top \mu_2}{\|\mathbf{w}\|}\right)^2 = \left(\frac{\mathbf{w}^\top}{\|\mathbf{w}\|}(\mu_1 - \mu_2)\right)^2.$$

The right-hand term is maximized when $\mathbf{w} \approx \mu_1 - \mu_2$. As illustrated in Fig. 8.1, this type of consideration is not sufficient to capture separability in the projected subspace because the spread (variance) of the data points around their centers also play an important role. For example, the horizontal axis in the figure separates the centers better than the vertical axis but on the other hand does a worse job in separating the classes themselves because of the way the data points are spread around their centers. The argument in favor of separating the centers would work if the data points were living in a hyper-sphere around the centers, but will not be sufficient otherwise.

The basic idea behind Fisher's LDA is to consider the sample covariance matrix of the individual classes as well as their centers, in the following way. The optimal 1D projection would that which

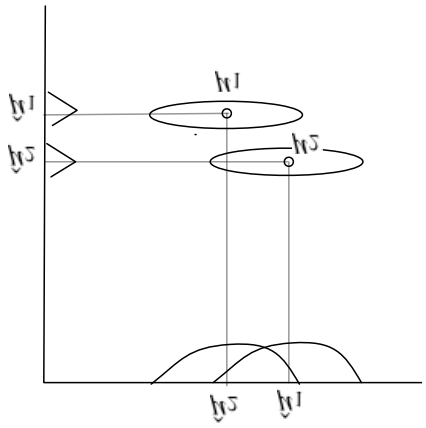

Figure 8.1: Linear discriminant analysis based on class centers alone is not sufficient. Seeking a projection which maximizes the distance between the projected centers will prefer the horizontal axis over the vertical, yet the two classes overlap on the horizontal axis. The projected distance along the vertical axis is smaller yet the classes are better separated. The conclusion is that the sample variance of the two classes must be taken into consideration as well.

maximizes the variance of the projected centers while *minimizes* the variance of the projected data points of each class separately. Mathematically, this idea can be implemented by maximizes the following ratio:

$$\max_{\mathbf{w}} \frac{(\hat{\mu}_1 - \hat{\mu}_2)^2}{s_1^2 + s_2^2},$$

where s_1^2 is the scaled variance of the projected points of the first class:

$$s_1^2 = \sum_{\mathbf{x}_i \in C_1} (\hat{\mathbf{x}}_i - \hat{\mu}_1)^2,$$

and likewise,

$$s_2^2 = \sum_{\mathbf{x}_i \in C_2} (\hat{\mathbf{x}}_i - \hat{\mu}_2)^2,$$

where $\hat{\mathbf{x}} = \frac{\mathbf{w}^\top}{\|\mathbf{w}\|}\mathbf{x}_i + b$.

We will now formalize this approach and derive its solution. We will begin with a general description of a multiclass problem where the sample data points belong to q different classes, and later focus on the case of $q = 2$.

8.6 Fisher's LDA: General Derivation

Let the sample data points S be members of q classes $C_1, ..., C_q$ where the number of points belonging to class C_i is denoted by l_i and the total number of the training set is $l = \sum_i l_i$. Let μ_j denote the center of class C_i and μ denote the center of the complete training set S:

$$\begin{aligned} \mu_j &= \frac{1}{l_j} \sum_{bfx_i \in C_j} \mathbf{x}_i \\ \mu &= \frac{1}{l} \sum_{\mathbf{x}_i \in S} \mathbf{x}_i \end{aligned}$$

Let A_j be the matrix associated with class C_j whose columns consists of the mean shifted data points:

$$A_j = [\mathbf{x}_1 - \mu_j, ..., \mathbf{x}_{l_j} - \mu_j] \quad \mathbf{x}_i \in C_j.$$

Then, $\frac{1}{l_j} A_j A_j^\top$ is the covariance matrix (see Lecture 9) associated with class C_j. Let S_w (where "w" stands for "within") be the sum of the class covariance matrices:

$$S_w = \sum_i^q \frac{1}{l_j} A_j A_j^\top.$$

From the discussion in the previous section, it is $\frac{1}{\|\mathbf{w}\|^2} \mathbf{w}^\top S_w \mathbf{w}$ which we wish to minimize. To see why this is so, note

$$\sum_{\mathbf{x}_i \in C_j} (\hat{\mathbf{x}}_i - \hat{\mu}_j)^2 = \sum_{\mathbf{x}_i \in C_j} \frac{\mathbf{w}^\top (\mathbf{x}_i - \mu_j)^2}{\|\mathbf{w}\|^2} = \frac{1}{\|\mathbf{w}\|^2} \mathbf{w}^\top A_j A_j^\top \mathbf{w}.$$

Let B be the matrix holding the class centers:

$$B = [\mu_1 - \mu, ..., \mu_q - \mu],$$

and let $S_b = \frac{1}{q} BB^\top$ (where "b" stands for "between"). From the discussion above it is $\frac{1}{\|\mathbf{w}\|^2} \mathbf{w}^\top S_b \mathbf{w} = \sum_i (\hat{\mu}_i - \hat{\mu})^2$ which we wish to *maximize*. Taken together, we wish to maximize the ratio (called "Rayleigh's quotient"):

$$\max_{\mathbf{w}} J(\mathbf{w}) = \frac{\mathbf{w}^\top S_b \mathbf{w}}{\mathbf{w}^\top S_w \mathbf{w}}.$$

The necessary condition for optimality is:

$$\frac{\partial J}{\partial \mathbf{w}} = \frac{S_b \mathbf{w}(\mathbf{w}^\top S_w \mathbf{w}) - S_w \mathbf{w}(\mathbf{w}^\top S_b \mathbf{w})}{(\mathbf{w}^\top S_w \mathbf{w})^2} = 0,$$

From which we obtain the generalized eigensystem:

$$S_b \mathbf{w} = J(\mathbf{w}) S_w \mathbf{w}. \tag{8.2}$$

That is, $\mathbf{w}$ is the leading eigenvector of $S_w^{-1} S_b$ (assuming S_w is invertible). The general case of finding q such axes involves finding the leading generalized eigenvectors of (S_b, S_w) — the derivation is out of scope of this lecture. Note that since $S_w^{-1} S_b$ is not symmetric there may be no real-value solution, which is a complication will not pursue further in this course. Instead we will focus now on the 2-class ($q = 2$) setting below.

8.7 Fisher's LDA: 2-class

The general derivation is simplified when there are only two classes. The covariance matrix $BB^\top$ becomes a rank-1 matrix:

$$BB^\top = (\mu_1 - \mu)(\mu_1 - \mu)^\top + (\mu_2 - \mu)(\mu_2 - \mu)^\top = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^\top.$$

As a result, $BB^\top \mathbf{w}$ is a vector in direction $\mu_1 - \mu_2$. Therefore, the solution for $\mathbf{w}$ from eqn. 8.2 is:

$$\mathbf{w} \cong S_w^{-1} (\mu_1 - \mu_2).$$

The decision boundary $\mathbf{w}^\top(\mathbf{x}-\mu)=0$ becomes:

$$\mathbf{x}^\top S_w^{-1}(\mu_1-\mu_2)-\frac{1}{2}(\mu_1+\mu_2)^\top S_w^{-1}(\mu_1-\mu_2)=0. \tag{8.3}$$

This decision boundary will surface again in the course when we consider Bayseian inference. It will be shown that this decision boundary is the Maximum Likelihood solution in the case where the two classes are normally distributed with means μ_1,μ_2 and with the same covariance matrix S_w.

8.8 LDA versus SVM

Both LDA and SVM search for a so called "optimal" linear discriminant function, what is the difference? The heart of the matter lies in the definition of what constitutes a sufficient compact representation of the data. In LDA the assumption is that each class can be represented by its mean vector and its spread (i.e., covariance matrix). This is true for normally distributed data — but not true in general. This means that we should expect that LDA will produce the optimal discriminant linear function when each of the classes are normally distributed.

With SVM, on the other hand, there is no assumption on how the data is distributed. Instead, the emerging result is that the data is represented by the subset of data points which lie on the boundary between the two classes (the so called support vectors). Rather than making a parametric assumption on how the data can be captured (i.e., mean and covariance) the theory shows that the data can be captured by a special subset of points. The tools, as a result, are naturally more complex (quadratic linear programming versus spectral matrix analysis) — but the advantage is that optimality is guaranteed without making assumptions on the distribution of the data (i.e., distribution free). It can be shown that SVM and LDA would produce the same result if the class data is normally distributed.

8.9 Canonical Correlation Analysis

CCA is a technique for learning a mapping $f(\mathbf{x})=\mathbf{y}$ where $\mathbf{x}\in R^k$ and $\mathbf{y}\in R^s$ using the notion of subspace similarity (an extension of the inner product between two vectors) from a training set of $(\mathbf{x}_i,\mathbf{y}_i)$, $i=1,...,n$. Such a mapping, where $\mathbf{y}$ can be any point in R^k as opposed to a discrete set of labels, is often referred to as a "regression" (as opposed to "classification").

Like in PCA and LDA, the approach would be to look for projection axes such that the projection of the input and output vectors on those axes satisfy certain requirements — and like PCA and LDA the tools we would be using is matrix spectral analysis.

It will be convenient to stack our vectors as rows of an input matrix A and output matrix B. Let A be an $n\times k$ matrix whose rows are $\mathbf{x}_1^\top,...,\mathbf{x}_n^\top$ and B is the $n\times s$ matrix whose rows are $\mathbf{y}_1^\top,...,\mathbf{y}_n^\top$. Consider vectors $\mathbf{u}\in R^k$ and $\mathbf{v}\in R^s$ and project the input and output data onto them producing $A\mathbf{u}=(\mathbf{x}_1^\top\mathbf{u},...,\mathbf{x}_n^\top\mathbf{u})$ and $B\mathbf{v}$. The requirement we would like to place on the projection axes is that $A\mathbf{u}\approx B\mathbf{v}$, or in other words that $(A\mathbf{u})^\top(B\mathbf{v})$ is maximal. The requirement therefore is that the projection of the input points onto the $\mathbf{u}$ axis is similar to the projection of the output points onto the $\mathbf{v}$ axis. If we extend this notion to multiple axes $\mathbf{u}_1,...,\mathbf{u}_q$ (not necessarily orthogonal) and $\mathbf{v}_1,...,\mathbf{v}_q$ where $q\leq\min(k,s)$ our requirement becomes that the new coordinates of the input points projected onto the subspace spanned by the $\mathbf{u}$ vectors are *similar* to the new coordinates of the output points projected onto the subspace spanned by the $\mathbf{v}$ vectors. In other words, we

wish to find two q-dimensional subspaces one of R^k and the other of R^s such that the two sets of projected points are as aligned as possible.

CCA goes a step further and makes the assumption that the input/output relationship is solely determined by the relation (angles) between the column spaces of A, B. In other words, the particular columns of A are not really important, what is important is the space U_A spanned by the columns. Since $\mathbf{g} = A\mathbf{u}$ is a point in U_A (a linear combination of the columns of A) and $\mathbf{h} = B\mathbf{v}$ is a point in U_B, then $\mathbf{g}^\top\mathbf{h}$ is the cosine angle, $\cos(\phi)$ between the two axes provided that we normalize the vectors $\mathbf{g}$ and $\mathbf{h}$. If we continue this line of reasoning recursively, we obtain a set of angles $0 \leq \theta_1 \leq ... \leq \theta_q \leq (\pi/2)$, called "principal angles", between the two subspaces uniquely defined as:

$$cos(\theta_j) = \max_{\mathbf{g}\in U_A} \max_{\mathbf{h}\in U_B} \mathbf{g}^\top\mathbf{h} \tag{8.4}$$

subject to:

$$\mathbf{g}^\top\mathbf{g} = \mathbf{h}^\top\mathbf{h} = 1, \quad \mathbf{h}^\top\mathbf{h}_i = 0, \mathbf{g}^\top\mathbf{g}_i = 0, \quad i = 1, ..., j-1$$

As a result, we obtain the following optimization function over axes $\mathbf{u}, \mathbf{v}$:

$$\max_{\mathbf{u},\mathbf{v}} \mathbf{u}^\top A^\top B\mathbf{v} \quad \text{s.t.} \quad \|A\mathbf{u}\|^2 = 1, \quad \|B\mathbf{v}\|^2 = 1.$$

To solve this problem we first perform a "QR" factorization of A and B. A "QR" factorization of a matrix A is a Grahm-Schmidt process resulting in an orthonormal set of vectors arranged as the columns of a matrix Q_A whose column space is equal to the column space of A, and a matrix R_A which contains the coefficients of the linear combination of the columns of Q_A such that $A = Q_A R_A$. Since orthoganilzation is not unique, the Grahm-Schmidt process perfroms the orthogonalization such that R_A is an upper-diagonal matrix. Likewise let $B = Q_B R_B$. Because the column spaces of A and Q_A are the same, then for every $\mathbf{u}$ there exists a $\hat{\mathbf{u}}$ such that $A\mathbf{u} = Q_A\hat{\mathbf{u}}$. Our optimization problem now becomes:

$$\max_{\hat{\mathbf{u}},\hat{\mathbf{v}}} \hat{\mathbf{u}}^\top Q_A^\top Q_B\hat{\mathbf{v}} \quad \text{s.t.} \quad \|\hat{\mathbf{u}}\|^2 = 1, \quad \|\hat{\mathbf{v}}\|^2 = 1.$$

The solution of this problem is when $\hat{\mathbf{u}}$ and $\hat{\mathbf{v}}$ are the leading singular vectors of $Q_A^\top Q_B$. The singular value decomposition (SVD) of any matrix E is a decomposition $E = UDV^\top$ where the columns of U are the leading eigenvectors of $EE^\top$, the rows of $V^\top$ are the leading eigenvectors of $E^\top E$ and D is a diagonal matrix whose entries are the corresponding square eigenvalues (note that the eigenvalues of $EE^\top$ and $E^\top E$ are the same). The SVD decomposition has the property that if we keep only the first q leading eigenvectors then $UDV^\top$ is the closest (in least squares sense) rank q matrix to E.

Therefore, let $\hat{U}D\hat{V}^\top$ be the SVD of $Q_A^\top Q_B$ using the first q eigenvectors. Then, our sought after axes $U = [\mathbf{u}_1, ..., \mathbf{u}_q]$ is simply $R_A^{-1}\hat{U}$ and likewise and the axes $V = [\mathbf{v}_1, ..., \mathbf{v}_q]$ is equal to $R_B^{-1}\hat{V}$. The axes are called "canonical vectors", and the vectors $\mathbf{g}_i = A\mathbf{u}_i$ (mutually orthogonal) are called "variates". The concept of principal angles is due to Jordan in 1875, where Hotelling in 1936 is the first to introduce the recursive definition above.

Given a new vector $\mathbf{x} \in R^k$ the resulting vector $\mathbf{y}$ can be found by solving the linear system $U^\top\mathbf{x} = V^\top\mathbf{y}$ (since our assumption is that in the new basis the coordinates of $\mathbf{x}$ and $\mathbf{y}$ are similar).

To conclude, the relationship between A and B is captured by creating similar variates, i.e., creating subspaces of dimension q such that the projections of the input vectors and the output vectors have similar coordinates. The process for obtaining the two q-dimensional subspaces is by performing a QR factorization of A and B followed by an SVD. Here again the spectral analysis of the input and output data matrices plays a pivoting role in the input/output association.

A Variance, Covariance, etc.

Let X, Y be two random variables and let $f(x, y)$ be some function on $x \in X, y \in Y$, and let $p(x, y)$ be the probability of the event x and y occurring together. The expectation $E[f(x, y)]$ is defined:

$$E[f(x,y)] = \sum_{x \in X} \sum_{y \in Y} f(x,y)p(x,y)$$

. The mean, variance and covariance are defined:

$$\begin{aligned}
\mu_x &= E[X] = \sum_x \sum_y xp(x,y) \\
\mu_y &= E[Y] = \sum_x \sum_y yp(x,y) \\
\sigma_x^2 &= Var[X] = E[(x-\mu_x)^2] = \sum_x \sum_y (x-\mu_x)^2 p(x,y) \\
\sigma_y^2 &= Var[Y] = E[(y-\mu_y)^2] = \sum_x \sum_y (y-\mu_y)^2 p(x,y) \\
\sigma_{xy} &= Cov(XY) = E[(x-\mu_x)(y-\mu_y)] = \sum_x \sum_y (x-\mu_x)(y-\mu_y)p(x,y)
\end{aligned}$$

In vector-matrix notation, let $\mathbf{x}$ represent the n random variables of $X_1, ..., X_n$, i.e., $\mathbf{x} = (x_1, ..., x_n)^\top$ is an instance vector and $p(\mathbf{x})$ is the probability of the instance occurrence. Then the mean is a vector μ and the covariance matrix E are defined:

$$\begin{aligned}
\mu &= \sum_{\mathbf{x} \in \{X_1, ..., X_n\}} \mathbf{x}p(\mathbf{x}) \\
E &= \sum_{\mathbf{x}} (\mathbf{x}-\mu)(\mathbf{x}-\mu)^\top p(\mathbf{x})
\end{aligned}$$

Note that the covariance matrix E is the linear superposition of rank-1 matrices $(\mathbf{x}-\mu)(\mathbf{x}-\mu)^\top$ with coefficients $p(\mathbf{x})$. The diagonal of E containes the variances of the variables $x_1, ..., x_n$. For a uniform distribution and a sample data S consisting of m points, let $A = [\mathbf{x}_1 - \mu, ..., \mathbf{x}_m - \mu]$ be the matrix whose columns consist of the points centered around the mean: $\mu = \frac{1}{m}\sum_i \mathbf{x}_i$. The (sample) covariance matrix is $E = \frac{1}{m}AA^\top$.

B Derivatives of Matrix Operations: Scalar Functions of a Vector

The two most important examples of a scalar function of a vector $\mathbf{x}$ are the linear form $\mathbf{a}^\top\mathbf{x}$ and the quadratic form $\mathbf{x}^\top A\mathbf{x}$ for some square matrix A.

$$\begin{aligned}
d(\mathbf{a}^\top\mathbf{x}) &= \mathbf{a}^\top d\mathbf{x} \\
d(\mathbf{x}^\top A\mathbf{x}) &= (d\mathbf{x})^\top A\mathbf{x} + \mathbf{x}^\top A(d\mathbf{x}) \\
&= \left((d\mathbf{x})^\top A\mathbf{x}\right)^\top + \mathbf{x}^\top A(d\mathbf{x}) \\
&= \mathbf{x}^\top(A + A^\top)d\mathbf{x}
\end{aligned}$$

where the derivative $d(\mathbf{x}^\top A\mathbf{x})$ using the rule of products $d(f \cdot g) = (df) \cdot g + f \cdot (dg)$ where $g = A\mathbf{x}$ and $f = \mathbf{x}^\top$ and noting that $d(A\mathbf{x}) = Ad\mathbf{x}$. Thus, $\frac{d}{d\mathbf{x}}(\mathbf{a}^\top\mathbf{x}) = \mathbf{a}^\top$ and $\frac{d}{d\mathbf{x}}(\mathbf{x}^\top A\mathbf{x})) = \mathbf{x}^\top(A + A^\top)$. If A is symmetric then $\frac{d}{d\mathbf{x}}(\mathbf{x}^\top A\mathbf{x})) = (2A\mathbf{x})^\top$.