TILM3592

Advanced Statistical Learning

Kernelized Linear Discriminant Analysis

NAME
MATR.

1 Introduction

This paper will try to explain the generalization of linear discriminant analysis to a nonlinear, kernelized version. Focus is perhaps not put as much on the classification as on the dimension-reduction part, focusing perhaps more on kernelizing Fisher's linear discriminant analysis so that the resulting projection can be used as a basis for a classifier. It was partly inspired by exercise 12.10, *Kernels and linear discriminant analysis*, in [2].

2 (Fisher) Linear Discriminant Analysis

Consider trying to classify two classes by separating them using a hyperplane, similar to the (linear, soft or hard margin) support vector machine or the logistic regression algorithm. Another approach could be to represent the two classes using their means and covariances. The problem could then take the form of finding a projection of the data such that the spread between the two classes (means) is maximized while the spread within the groups is minimized. This can be likened to principal components analysis where one instead tries to find a projection that maximizes the spread and we will see that this approach is similar in some ways.

To make the previous more concrete, let $\mathbf{X}$ be a $(p \times 1)$ random vector and $\mathbf{w}$ be a $(p \times 1)$ vector representing a linear combination, then consider the covariance and mean of the random variable $\mathbf{w}^\intercal\mathbf{X}$, $\mathbb{C}\mathrm{ov}\left(\mathbf{w}^\intercal\mathbf{X}, \mathbf{w}^\intercal\mathbf{X}\right)$ and $\mathbb{E}\left(\mathbf{w}^\intercal\mathbf{X}\right)$. Since the expectation operator is linear we know that $\mathbb{E}\left(\mathbf{w}^\intercal\mathbf{X}\right) = \mathbf{w}^\intercal\,\mathbb{E}\left(\mathbf{X}\right)$ and since

$$\mathbb{C}\mathrm{ov}\left(\mathbf{X}, \mathbf{X}\right) := \mathbb{E}\left[\left(\mathbf{X} - \mathbb{E}\left[\mathbf{X}\right]\right)\left(\mathbf{X} - \mathbb{E}\left[\mathbf{X}\right]\right)^\intercal\right] = \Sigma_{\mathbf{XX}},$$

we get

$$\begin{aligned}\underline{\mathbb{C}\mathrm{ov}\left(\mathbf{w}^\intercal\mathbf{X}, \mathbf{w}^\intercal\mathbf{X}\right)} &= \mathbb{E}\left[\left(\mathbf{w}^\intercal\mathbf{X} - \mathbb{E}\left[\mathbf{w}^\intercal\mathbf{X}\right]\right)\left(\mathbf{w}^\intercal\mathbf{X} - \mathbb{E}\left[\mathbf{w}^\intercal\mathbf{X}\right]\right)^\intercal\right] \\ &= \mathbb{E}\left[\left(\mathbf{w}^\intercal\mathbf{X} - \mathbf{w}^\intercal\mathbb{E}\left[\mathbf{X}\right]\right)\left(\mathbf{w}^\intercal\mathbf{X} - \mathbf{w}^\intercal\mathbb{E}\left[\mathbf{X}\right]\right)^\intercal\right] \\ &= \mathbb{E}\left[\mathbf{w}^\intercal\left(\mathbf{X} - \mathbb{E}\left[\mathbf{X}\right]\right)\left(\mathbf{X} - \mathbb{E}\left[\mathbf{X}\right]\right)^\intercal\mathbf{w}\right] \\ &= \mathbf{w}^\intercal\mathbb{E}\left[\left(\mathbf{X} - \mathbb{E}\left[\mathbf{X}\right]\right)\left(\mathbf{X} - \mathbb{E}\left[\mathbf{X}\right]\right)^\intercal\right]\mathbf{w} = \underline{\mathbf{w}^\intercal\Sigma_{\mathbf{XX}}\mathbf{w}}.\end{aligned}$$

Just as a sanity check, since $\mathbf{X}$ is $(p \times 1)$ we know that $\mathbb{C}\mathrm{ov}\left(\mathbf{X}, \mathbf{X}\right)$ is a $(p \times p)$ matrix and since $\mathbf{w}^\intercal$ is $(1 \times p)$ the random variable $\mathbf{w}^\intercal\mathbf{X}$ is (1×1) (scalar) the covariance (variance) $\mathbb{C}\mathrm{ov}\left(\mathbf{w}^\intercal\mathbf{X}, \mathbf{w}^\intercal\mathbf{X}\right)$ also has to be scalar. This is true since $\mathbf{w}^\intercal\Sigma_{\mathbf{XX}}\mathbf{w}$ is a $(1 \times p)$ vector times a $(p \times p)$ matrix times a $(p \times 1)$ vector. Thus the result is a (1×1) vector or a scalar which is what we were hoping for.

For the spread of the means or the distance between the means we can look at the quantity $\left\|\mathbb{E}\left(\mathbf{w}^\intercal\mathbf{X}\right) - \mathbb{E}\left(\mathbf{w}^\intercal\mathbf{Y}\right)\right\|_2$ where $\mathbf{X}$ are the random vectors in class 1 and $\mathbf{Y}$ are the random vectors in class 2. The squared distance $\left\|\mathbb{E}\left(\mathbf{w}^\intercal\mathbf{X}\right) - \mathbb{E}\left(\mathbf{w}^\intercal\mathbf{Y}\right)\right\|_2^2$ can be written as

$$\begin{aligned}\underline{\left\|\mathbb{E}\left(\mathbf{w}^\intercal\mathbf{X}\right) - \mathbb{E}\left(\mathbf{w}^\intercal\mathbf{Y}\right)\right\|_2^2} &= \left\langle\mathbb{E}\left(\mathbf{w}^\intercal\mathbf{X}\right) - \mathbb{E}\left(\mathbf{w}^\intercal\mathbf{Y}\right), \mathbb{E}\left(\mathbf{w}^\intercal\mathbf{X}\right) - \mathbb{E}\left(\mathbf{w}^\intercal\mathbf{Y}\right)\right\rangle \\ &= \left(\mathbb{E}\left[\mathbf{w}^\intercal\mathbf{X}\right] - \mathbb{E}\left[\mathbf{w}^\intercal\mathbf{Y}\right]\right)^\intercal\left(\mathbb{E}\left[\mathbf{w}^\intercal\mathbf{X}\right] - \mathbb{E}\left[\mathbf{w}^\intercal\mathbf{Y}\right]\right) \\ &= \left(\mathbf{w}^\intercal\mathbb{E}\left[\mathbf{X}\right] - \mathbf{w}^\intercal\mathbb{E}\left[\mathbf{Y}\right]\right)^\intercal\left(\mathbf{w}^\intercal\mathbb{E}\left[\mathbf{X}\right] - \mathbf{w}^\intercal\mathbb{E}\left[\mathbf{Y}\right]\right) \\ &= \left(\mathbb{E}\left[\mathbf{X}\right] - \mathbb{E}\left[\mathbf{Y}\right]\right)^\intercal\mathbf{w}\mathbf{w}^\intercal\left(\mathbb{E}\left[\mathbf{X}\right] - \mathbb{E}\left[\mathbf{Y}\right]\right)\end{aligned}$$

which is a (1×1) vector so its transpose is the same as itself, so

$$= \mathbf{w}^\intercal\left(\mathbb{E}\left[\mathbf{X}\right] - \mathbb{E}\left[\mathbf{Y}\right]\right)\left(\mathbb{E}\left[\mathbf{X}\right] - \mathbb{E}\left[\mathbf{Y}\right]\right)^\intercal\mathbf{w} = \underline{\mathbf{w}^\intercal S_{\mathbf{XY}}\mathbf{w}}.$$

Since we want to maximize the distance between the projected means while minimizing the covariances of the projected variances this means we want to maximize $\mathbf{w}^\intercal S_{\mathbf{XY}}\mathbf{w}$ while minimizing $\mathbf{w}^\intercal\Sigma_{\mathbf{XX}}\mathbf{w} + \mathbf{w}^\intercal\Sigma_{\mathbf{YY}}\mathbf{w} = \mathbf{w}^\intercal\left(\Sigma_{\mathbf{XX}} + \Sigma_{\mathbf{YY}}\right)\mathbf{w}$ with respect to $\mathbf{w}$. There are two natural ways to do this; either we can maximize the (generalized Rayleigh) quotient $\frac{\mathbf{w}^\intercal S_{\mathbf{XY}}\mathbf{w}}{\mathbf{w}^\intercal(\Sigma_{\mathbf{XX}}+\Sigma_{\mathbf{YY}})\mathbf{w}}$ or we can maximize the difference $\mathbf{w}^\intercal S_{\mathbf{XY}}\mathbf{w} - \mathbf{w}^\intercal\left(\Sigma_{\mathbf{XX}} + \Sigma_{\mathbf{YY}}\right)\mathbf{w}$. We will soon see that these two are equivalent in some way but first we need to add a condition to the optimization problem so that it is well defined: Consider the first optimization problem. For a given $\mathbf{w}$ notice that stretching it by the scalar a won't have an impact on the value of the objective function $\frac{\mathbf{w}^\intercal S_{\mathbf{XY}}\mathbf{w}}{\mathbf{w}^\intercal(\Sigma_{\mathbf{XX}}+\Sigma_{\mathbf{YY}})\mathbf{w}}$ since $\frac{a\mathbf{w}^\intercal S_{\mathbf{XY}}a\mathbf{w}}{a\mathbf{w}^\intercal(\Sigma_{\mathbf{XX}}+\Sigma_{\mathbf{YY}})a\mathbf{w}} = \frac{a^2\mathbf{w}^\intercal S_{\mathbf{XY}}\mathbf{w}}{a^2\mathbf{w}^\intercal(\Sigma_{\mathbf{XX}}+\Sigma_{\mathbf{YY}})\mathbf{w}} = \frac{\mathbf{w}^\intercal S_{\mathbf{XY}}\mathbf{w}}{\mathbf{w}^\intercal(\Sigma_{\mathbf{XX}}+\Sigma_{\mathbf{YY}})\mathbf{w}}$. Thus only the direction of the vector $\mathbf{w}$ matters and we can add the condition $\mathbf{w}^\intercal\left(\Sigma_{\mathbf{XX}} + \Sigma_{\mathbf{YY}}\right)\mathbf{w} = 1$, fixing the denominator at the same time.

The optimization problem that is ultimately solved is then the following:

$$\begin{aligned}\max_{\mathbf{w}} \quad & \mathbf{w}^\intercal S_{\mathbf{XY}}\mathbf{w} \\ \text{such that} \quad & \mathbf{w}^\intercal\left(\Sigma_{\mathbf{XX}} + \Sigma_{\mathbf{YY}}\right)\mathbf{w} = 1.\end{aligned} \tag{1}$$

This is readily solved using the Lagrange multiplier method. The Lagrangian is given by $\mathcal{L}(\mathbf{w}, \lambda) = \mathbf{w}^\intercal S_{\mathbf{XY}} \mathbf{w} - \lambda \left(\mathbf{w}^\intercal \left(\Sigma_{\mathbf{XX}} + \Sigma_{\mathbf{YY}} \right) \mathbf{w} - 1 \right)$ with the derivative

$$\frac{\mathrm{d}\mathcal{L}\left(\mathbf{w}, \lambda\right)}{\mathrm{d}\mathbf{w}} = 2 S_{\mathbf{XY}} \mathbf{w} - 2\lambda \left(\Sigma_{\mathbf{XX}} + \Sigma_{\mathbf{YY}} \right) \mathbf{w}.$$

For $\lambda = 1$ this is the same derivative as the derivative of the second optimization problem considered (with the difference between the projected spread and variances). Setting the derivative equal to zero gives the following relation:

$$S_{\mathbf{XY}} \mathbf{w} = \lambda \left(\Sigma_{\mathbf{XX}} + \Sigma_{\mathbf{YY}} \right) \mathbf{w}.$$

This is called a generalized eigenvalue problem which can be changed into a regular eigenvector problem if $\left(\Sigma_{\mathbf{XX}} + \Sigma_{\mathbf{YY}} \right)$ is non-singular. If this is the case then we can make further progress.

Since $S_{\mathbf{XY}}$ is a product of two vectors it has rank 1 (the basis is one of the vectors and the weights are given by the other vector). Thus the result of the matrix multiplication $\left(\Sigma_{\mathbf{XX}} + \Sigma_{\mathbf{YY}} \right)^{-1} S_{\mathbf{XY}}$ also has at most rank 1 and thus has at most one eigenvalue not equal to one. The solution to the optimization problem is then the eigenvector of the matrix $\left(\Sigma_{\mathbf{XX}} + \Sigma_{\mathbf{YY}} \right)^{-1} S_{\mathbf{XY}}$.

The algorithm usually referred to as linear discriminant analysis can be seen as first finding this projection and then choosing a point on the one-dimensional projection to use as the threshold for classification. This can be done by assuming things about the distribution of the data, for example assuming that the two classes are normally distributed around separate means but with the same covariance matrix. Using the assumptions and incorporating weighting based on the number of observations in each class one can determine a threshold such that the resulting classification is equivalent to the "normal" linear discriminant analysis algorithm.

3 A note on the solution

Usually, the covariance and distance matrices $\Sigma_{\mathbf{XX}}$, $\Sigma_{\mathbf{YY}}$ and $S_{\mathbf{XY}}$ are unknown and thus need to be estimated from some observed data. In other words the matrices depend on the observed data and one can even see that they lie in the space spanned by the observations in some sense.

Consider the maximum likelihood estimate of the expected value of class one ($\mathbb{E}\left(\mathbf{X}\right)$) which is equal to

$$\mathbf{m}_{\mathbf{X}} := \frac{1}{N_{\mathbf{X}}} \sum_{i=1}^{N_{\mathbf{X}}} \mathbf{X}_i$$

which is a linear combination of the $N_{\mathbf{X}}$ observations $\mathbf{X}_i$ and thus lies in the span of the observations. The analogous is true for $\mathbf{m}_{\mathbf{Y}}$, the maximum likelihood estimate of the expected value of class 2.

The covariances can be estimated as

$$\hat{\Sigma}_{\mathbf{XX}} := \frac{1}{N_{\mathbf{X}}} \sum_{i=1}^{N_{\mathbf{X}}} (\mathbf{X}_i - \mathbf{m}_{\mathbf{X}}) (\mathbf{X}_i - \mathbf{m}_{\mathbf{X}})^{\mathsf{T}} \qquad \text{and}$$
$$\hat{\Sigma}_{\mathbf{YY}} := \frac{1}{N_{\mathbf{Y}}} \sum_{i=1}^{N_{\mathbf{Y}}} (\mathbf{Y}_i - \mathbf{m}_{\mathbf{Y}}) (\mathbf{Y}_i - \mathbf{m}_{\mathbf{Y}})^{\mathsf{T}},$$

and the distance between the means as

$$\hat{S}_{\mathbf{XY}} := (\mathbf{m}_{\mathbf{X}} - \mathbf{m}_{\mathbf{Y}}) (\mathbf{m}_{\mathbf{X}} - \mathbf{m}_{\mathbf{Y}})^{\mathsf{T}}.$$

All three of these are matrices where the rows/columns are linear combinations of the observations $\mathbf{X}_i$ and $\mathbf{Y}_i$.

For the solution to the optimization problem, the eigenvector, the same has to be true, i.e. the eigenvector has to be some linear combination of the observations. In other words,

$$\mathbf{w} = \sum_{i=1}^{N_{\mathbf{X}}} \alpha_{\mathbf{X}_i} \mathbf{X}_i + \sum_{i=1}^{N_{\mathbf{Y}}} \alpha_{\mathbf{Y}_i} \mathbf{Y}_i$$

or, by defining $\mathbf{Z}$ as the matrix containing all $N_{\mathbf{X}} + N_{\mathbf{Y}} = N$ observations,

$$= \sum_{i=1}^{N} \alpha_i \mathbf{Z}_i.$$

Consider now a (possibly non-linear) transformation $h : \mathbb{R}^p \longmapsto \mathbb{R}^P$. Suppose we wish to apply the (Fisher) linear discriminant analysis algorithm to this transformed dataset. Defining the vectors

$$\mathbf{m}_{h(\mathbf{X})} := \frac{1}{N_{\mathbf{X}}} \sum_{i=1}^{N_{\mathbf{X}}} h(\mathbf{X}_i) \qquad \text{and}$$
$$\mathbf{m}_{h(\mathbf{Y})} := \frac{1}{N_{\mathbf{Y}}} \sum_{i=1}^{N_{\mathbf{Y}}} h(\mathbf{Y}_i),$$

as well as the matrices

$$\hat{\Sigma}_{h(\mathbf{X})h(\mathbf{X})} := \frac{1}{N_{\mathbf{X}}} \sum_{i=1}^{N_{\mathbf{X}}} \left(h(\mathbf{X}_i) - \mathbf{m}_{h(\mathbf{X})}\right) \left(h(\mathbf{X}_i) - \mathbf{m}_{h(\mathbf{X})}\right)^{\mathsf{T}},$$
$$\hat{\Sigma}_{h(\mathbf{Y})h(\mathbf{Y})} := \frac{1}{N_{\mathbf{Y}}} \sum_{i=1}^{N_{\mathbf{Y}}} \left(h(\mathbf{Y}_i) - \mathbf{m}_{h(\mathbf{Y})}\right) \left(h(\mathbf{Y}_i) - \mathbf{m}_{h(\mathbf{Y})}\right)^{\mathsf{T}} \qquad \text{and}$$
$$\hat{S}_{h(\mathbf{X})h(\mathbf{Y})} := \left(\mathbf{m}_{h(\mathbf{X})} - \mathbf{m}_{h(\mathbf{Y})}\right) \left(\mathbf{m}_{h(\mathbf{X})} - \mathbf{m}_{h(\mathbf{Y})}\right)^{\mathsf{T}}$$

we can try solving the optimization problem

$$\begin{aligned} \max_{\mathbf{w}} \quad & \mathbf{w}^\intercal \hat{S}_{h(\mathbf{X})h(\mathbf{Y})} \mathbf{w} \\ \text{such that} \quad & \mathbf{w}^\intercal \left(\hat{\Sigma}_{h(\mathbf{X})h(\mathbf{X})} + \hat{\Sigma}_{h(\mathbf{Y})h(\mathbf{Y})} \right) \mathbf{w} = 1, \end{aligned} \tag{2}$$

where $\mathbf{w}$ is now a $(1 \times P)$ vector. If P is large this might be difficult, especially if $P > N$ since the matrix $\left(\hat{\Sigma}_{h(\mathbf{X})h(\mathbf{X})} + \hat{\Sigma}_{h(\mathbf{Y})h(\mathbf{Y})} \right) =: W_h$ will be singular and a similar approach as the solution to (1) would not work.

However, what we do know is that if there is a solution $\mathbf{w}$, it has to be of the form

$$\mathbf{w} = \sum_{i=1}^{N} \alpha_i h\left(\mathbf{Z}_i\right).$$

The linear combination of the observed transformed average, $\mathbf{w}^\intercal \mathbf{m}_{h(\mathbf{X})}$, then has to have the form

$$\begin{aligned} \mathbf{w}^\intercal \mathbf{m}_{h(\mathbf{X})} &= \frac{1}{N_\mathbf{X}} \mathbf{w}^\intercal \sum_{i=1}^{N_\mathbf{X}} h\left(\mathbf{X}_i\right) = \frac{1}{N_\mathbf{X}} \sum_{j=1}^{N} \alpha_j h\left(\mathbf{Z}_j\right)^\intercal \sum_{i=1}^{N_\mathbf{X}} h\left(\mathbf{X}_i\right) \\ &= \frac{1}{N_\mathbf{X}} \sum_{j=1}^{N} \sum_{i=1}^{N_\mathbf{X}} \alpha_j h\left(\mathbf{Z}_j\right)^\intercal h\left(\mathbf{X}_i\right) = \frac{1}{N_\mathbf{X}} \sum_{j=1}^{N} \sum_{i=1}^{N_\mathbf{X}} \alpha_j \left\langle h\left(\mathbf{Z}_j\right), h\left(\mathbf{X}_i\right) \right\rangle. \end{aligned}$$

At this point we recognize the inner product between two high-dimensional vectors, $\left\langle h\left(\mathbf{Z}_j\right), h\left(\mathbf{X}_i\right) \right\rangle$, and realize that this can be computed using kernels.

4 Reformulating Fishers Linear Discriminant using Kernels

Kernels are functions that efficiently compute inner products in some high-dimensional (even infinite-dimensional) space. An advantage with using kernels is that when we have some mapping h to a higher dimensional space, the kernel doesn't need to explicitly compute the mapping but instead it implicitly computes in a high-dimensional space (given that the mapping h is such that it gives rise to a kernel). One can also go the other way around and fix a kernel that then determines what mapping one is working in. In general, kernels can always be used instead of inner products.

Continuing with the manipulation of $\mathbf{w}^\intercal \mathbf{m}_{h(\mathbf{X})}$, the inner product $\left\langle h\left(\mathbf{Z}_j\right), h\left(\mathbf{X}_i\right) \right\rangle$ can be replaced by a suitable kernel $k\left(\mathbf{Z}_j, \mathbf{X}_i\right)$, meaning that

$$\underline{\mathbf{w}^\intercal \mathbf{m}_{h(\mathbf{X})}} = \frac{1}{N_\mathbf{X}} \sum_{j=1}^{N} \sum_{i=1}^{N_\mathbf{X}} \alpha_j k\left(\mathbf{Z}_j, \mathbf{X}_i\right)$$

which can be seen as the multiplication

$$= \underline{\boldsymbol{\alpha}^\intercal \mathbf{M}_\mathbf{X}},$$

where $\boldsymbol{\alpha}$ is an $(1 \times N)$ vector of the scalars α_j, $j = 1, \ldots, N$ and $\mathbf{M}_{\mathbf{X}}$ is the $(1 \times N)$ vector of the scalars $\frac{1}{N_{\mathbf{X}}} \sum_{i=1}^{N_{\mathbf{X}}} k(\mathbf{Z}_j, \mathbf{X}_i)$, $j = 1, \ldots, N$. Similarly we get that $\mathbf{w}^\intercal \mathbf{m}_{h(\mathbf{Y})} = \boldsymbol{\alpha}^\intercal \mathbf{M}_{\mathbf{Y}}$.

The spread between the two transformed and projected means is then given by

$$\begin{aligned}
\underline{\left\| \mathbf{w}^\intercal \mathbf{m}_{h(\mathbf{X})} - \mathbf{w}^\intercal \mathbf{m}_{h(\mathbf{Y})} \right\|_2^2} &= \left\| \boldsymbol{\alpha}^\intercal \mathbf{M}_{\mathbf{X}} - \boldsymbol{\alpha}^\intercal \mathbf{M}_{\mathbf{Y}} \right\|_2^2 = \left\langle \boldsymbol{\alpha}^\intercal \mathbf{M}_{\mathbf{X}} - \boldsymbol{\alpha}^\intercal \mathbf{M}_{\mathbf{Y}}, \boldsymbol{\alpha}^\intercal \mathbf{M}_{\mathbf{X}} - \boldsymbol{\alpha}^\intercal \mathbf{M}_{\mathbf{Y}} \right\rangle \\
&= \left(\boldsymbol{\alpha}^\intercal \mathbf{M}_{\mathbf{X}} - \boldsymbol{\alpha}^\intercal \mathbf{M}_{\mathbf{Y}} \right)^\intercal \left(\boldsymbol{\alpha}^\intercal \mathbf{M}_{\mathbf{X}} - \boldsymbol{\alpha}^\intercal \mathbf{M}_{\mathbf{Y}} \right) \\
&= \left(\mathbf{M}_{\mathbf{X}} - \mathbf{M}_{\mathbf{Y}} \right)^\intercal \boldsymbol{\alpha} \boldsymbol{\alpha}^\intercal \left(\mathbf{M}_{\mathbf{X}} - \mathbf{M}_{\mathbf{Y}} \right) \\
&= \boldsymbol{\alpha}^\intercal \left(\mathbf{M}_{\mathbf{X}} - \mathbf{M}_{\mathbf{Y}} \right) \left(\mathbf{M}_{\mathbf{X}} - \mathbf{M}_{\mathbf{Y}} \right)^\intercal \boldsymbol{\alpha} = \underline{\boldsymbol{\alpha}^\intercal S_k \boldsymbol{\alpha}},
\end{aligned}$$

this is the same calculation as the one performed for the original space and S_k is the $(N \times N)$ matrix defined as $S_k := \left(\mathbf{M}_{\mathbf{X}} - \mathbf{M}_{\mathbf{Y}} \right) \left(\mathbf{M}_{\mathbf{X}} - \mathbf{M}_{\mathbf{Y}} \right)^\intercal$.

For the covariance/variance within the groups we first need to take a look at the projection of a transformed variable,

$$\begin{aligned}
\underline{\mathbf{w}^\intercal h(\mathbf{X})} &= \sum_{j=1}^{N} \alpha_j h(\mathbf{Z}_j)^\intercal h(\mathbf{X}) = \sum_{j=1}^{N} \alpha_j \left\langle h(\mathbf{Z}_j), h(\mathbf{X}) \right\rangle \\
&= \sum_{j=1}^{N} \alpha_j k(\mathbf{Z}_j, \mathbf{X}) = \underline{\boldsymbol{\alpha}^\intercal \mathbf{K}_{\mathbf{X}}}.
\end{aligned}$$

The covariance $\underline{\mathbb{C}\mathrm{ov}\left(\mathbf{w}^\intercal h(\mathbf{X}), \mathbf{w}^\intercal h(\mathbf{X}) \right)}$ is then estimated by

$$\begin{aligned}
&\frac{1}{N_x} \sum_{i=1}^{N_x} \left(\left(\mathbf{w}^\intercal h(\mathbf{X}_i) - \mathbf{w}^\intercal \mathbf{m}_{h(\mathbf{X})} \right) \left(\mathbf{w}^\intercal h(\mathbf{X}_i) - \mathbf{w}^\intercal \mathbf{m}_{h(\mathbf{X})} \right)^\intercal \right) \\
&= \frac{1}{N_x} \sum_{i=1}^{N_x} \left(\left(\boldsymbol{\alpha}^\intercal \mathbf{K}_{\mathbf{X}_i} - \boldsymbol{\alpha}^\intercal \mathbf{M}_{\mathbf{X}} \right) \left(\boldsymbol{\alpha}^\intercal \mathbf{K}_{\mathbf{X}_i} - \boldsymbol{\alpha}^\intercal \mathbf{M}_{\mathbf{X}} \right)^\intercal \right) \\
&= \frac{1}{N_x} \sum_{i=1}^{N_x} \left(\boldsymbol{\alpha}^\intercal \left(\mathbf{K}_{\mathbf{X}_i} - \mathbf{M}_{\mathbf{X}} \right) \left(\mathbf{K}_{\mathbf{X}_i} - \mathbf{M}_{\mathbf{X}} \right)^\intercal \boldsymbol{\alpha} \right) \\
&= \boldsymbol{\alpha}^\intercal \left(\frac{1}{N_{\mathbf{X}}} \sum_{i=1}^{N_{\mathbf{X}}} \left(\mathbf{K}_{\mathbf{X}_i} - \mathbf{M}_{\mathbf{X}} \right) \left(\mathbf{K}_{\mathbf{X}_i} - \mathbf{M}_{\mathbf{X}} \right)^\intercal \right) \boldsymbol{\alpha} = \underline{\boldsymbol{\alpha}^\intercal \Sigma_{k\mathbf{X}} \boldsymbol{\alpha}}
\end{aligned}$$

and similarly for class $\mathbf{Y}$ we get the estimate $\boldsymbol{\alpha}^\intercal \Sigma_{k\mathbf{Y}} \boldsymbol{\alpha}$, where the $(N \times N)$ matrix describes the average observed distances from the transformed mean of the observations and is given by $\Sigma_{k\mathbf{Y}} := \left(\frac{1}{N_{\mathbf{Y}}} \sum_{i=1}^{N_{\mathbf{Y}}} \left(\mathbf{K}_{\mathbf{Y}_i} - \mathbf{M}_{\mathbf{Y}} \right) \left(\mathbf{K}_{\mathbf{Y}_i} - \mathbf{M}_{\mathbf{Y}} \right)^\intercal \right)$ (there might be more efficient ways to compute this). The two covariances can be summed, we define this as the matrix $W_k := \Sigma_{k\mathbf{X}} + \Sigma_{k\mathbf{Y}}$.

In this way we have transformed the optimization problem (2) into the following optimization problem:

$$\begin{aligned}
\max_{\boldsymbol{\alpha}} \quad & \boldsymbol{\alpha}^\intercal S_k \boldsymbol{\alpha} \\
\text{such that} \quad & \boldsymbol{\alpha}^\intercal W_k \boldsymbol{\alpha} = 1,
\end{aligned} \tag{3}$$

a problem where we already know that the solution $\boldsymbol{\alpha}$ is given by the only eigenvector of the matrix $W_k^{-1} S_k$.

The projection of a new observation $\mathbf{X}$ is given by $\mathbf{w}^\intercal h(\mathbf{X}) = \boldsymbol{\alpha}\mathbf{K}_\mathbf{X}$. The kernel k or the non-linear transformation h should be chosen using cross-validation. Classification can be done in the projected space using almost any classification method.

4.1 Some notes

In general, the matrix W_k might not be invertible according to [1] since N dimensional covariances are estimated using N observations. The authors suggest replacing W_k by $W_\gamma = W_k + \gamma I$ where I is the $(N \times N)$ identity matrix. For a sufficiently large γ, the resulting matrix will be invertible. This also has the benefit of having a regularizing effect, which is good since the high-dimensional space that the kernel allows for might lead to overfitting.

For the binary classification problem one can solve for the direction of the eigenvector by noting that $\boldsymbol{\alpha} \propto W_k^{-1}(\mathbf{M}_\mathbf{X} - \mathbf{M}_\mathbf{Y})$. Once the direction is known the magnitude does not really matter.

The method can be extended to multiple-classes and then gives as many egienvectors or directions/projections as the number of classes minus 1. Then the estimation of the covariance matrices can be troublesome since all the combinations of the classes have to be taken into account.

5 Conclusion

Three optimization problems were presented and two of them solved in order to explain how the dimensionality reduction step of the linear discriminant classifier can be generalized to a kernelized version. The main steps were to formulate an optimization problem so that the resulting solution is pleasing, then to solve the optimization problem using Lagrange multipliers and to recognize the generalized eigenvalue problem. After the original problem is solved some knowledge of linear algebra and reproducing kernels is used to rework the problem into one where the data is only present in inner products so that they can be switched for kernels. After this one can implicitly work in infinite-dimensional spaces to get a good projection that can be used for classification.

References

[1] S. Mika, G. Rätsch, J. Weston, B. Schölkopf and K.-R. Müller, *Fisher Discriminant Analysis With Kernels.* Neural Networks for Signal Processing IX: Proceedings of the 1999 IEEE Signal Processing Society Workshop, 1999.

[2] T. Hastie, R. Tibshirani and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer, Second Edition, 2009.