2008

Linear Dimensionality Reduction by Maximizing the Chernoff Distance in the Transformed Space

Luis Rueda
University of Windsor

Myriam Herrera

Recommended Citation

Rueda, Luis and Herrera, Myriam. (2008). Linear Dimensionality Reduction by Maximizing the Chernoff Distance in the Transformed Space. *Pattern Recognition*, 41 (10), 3138-3152.
https://scholar.uwindsor.ca/computersciencepub/9

Linear Dimensionality Reduction by Maximizing the Chernoff Distance in the Transformed Space

Luis Rueda[*] **and Myriam Herrera**[†]

Abstract

Linear dimensionality reduction (LDR) techniques are quite important in pattern recognition due to their linear time complexity and simplicity. In this paper, we present a novel LDR technique which, though linear, aims to maximize the Chernoff distance in the transformed space; thus, augmenting the class separability in such a space. We present the corresponding criterion, which is maximized via a gradient-based algorithm, and provide convergence and initialization proofs. We have performed a comprehensive performance analysis of our method combined with two well-known classifiers, linear and quadratic, on synthetic and real-life data, and compared it with other LDR techniques. The results on synthetic and standard real-life datasets show that the proposed criterion outperforms the latter when combined with both linear and quadratic classifiers.

[*] Member of the IEEE. Department of Computer Science, University of Concepción, Edmundo Larenas 215, Concepción, 4030000, Chile. Phone: +56 41 220-4305, Fax: +56 41 222-1770. E-mail: lrueda@inf.udec.cl. Partially supported by the Chilean National Fund for Scientific and Technological Development, FONDECYT grant No. 1060904.

[†] Institute of Informatics, National University of San Juan, Cereceto y Meglioli, San Juan, 5400, Argentina. E-mail: mherrera@iinfo.unsj.edu.ar

1 Introduction

Linear dimensionality reduction (LDR) techniques have been studied for a long time in the field of pattern recognition. They are typically the preferred ones due to their efficiency, and because they are simpler to implement and understand. We assume that we are dealing with two classes, ω_1 and ω_2, which are represented by two normally distributed n-dimensional random vectors, $\mathbf{x}_1 \sim N(\mathbf{m}_1, \mathbf{S}_1)$ and $\mathbf{x}_2 \sim N(\mathbf{m}_2, \mathbf{S}_2)$, and whose *a priori* probabilities are p_1 and p_2 respectively. The aim is to linearly transform $\mathbf{x}_1$ and $\mathbf{x}_2$ into new normally distributed random vectors $\mathbf{y}_1 = \mathbf{A}\mathbf{x}_1$ and $\mathbf{y}_2 = \mathbf{A}\mathbf{x}_2$ of dimension d, $d < n$, using a matrix $\mathbf{A}$ of order $d \times n$, in such a way that the classification error in the transformed space is as small as possible.

1.1 Related Work

Various schemes that yield LDR have been reported in the literature, including the well known *Fisher's discriminant* (FD) approach [5], and its extensions: the *direct* Fisher's discriminant analysis [21], its kernelized version for face recongition [8], the combined principal component analysis (PCA) and linear discriminant analysis (LDA) [19], the kernelized PCA and LDA [18], and a two-dimensional FD-based approach for face recognition [20]. An improvement to the FD approach that decomposes classes into subclasses has been proposed in [10]. Rueda et al. [16] showed that the optimal classifier between two normally distributed classes can be linear even when the covariance matrices *are not equal*. In [15], a new approach to selecting the *best hyperplane classifier* (BHC), which is obtained from the optimal pairwise linear classifier, has been introduced. A computationally intensive method for LDR was proposed in [14], which aims to minimize the classification error in the transformed space and operates by computing (or approximating) the *exact* values for the integrals. This approach, though extremely time consuming, does not guarantees an optimal LDR. Another criterion used for dimensionality reduction is the subclass discriminant analysis [22], which aims to optimally divide the classes into subclasses, and then performs the reduction followed by classification.

We now focus on two LDR approaches which are closely related to our proposed method. Let $\mathbf{S}_W = p_1\mathbf{S}_1 + p_2\mathbf{S}_2$ and $\mathbf{S}_E = (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^t$ be the within-class and between-class scatter matrices respectively. The well-known FD criterion consists of maximizing the Mahalanobis distance between the transformed distributions by finding $\mathbf{A}$ that maximizes the following function [5]:

$$J_{FD}(\mathbf{A}) = tr\left\{(\mathbf{A}\mathbf{S}_W\mathbf{A}^t)^{-1}(\mathbf{A}\mathbf{S}_E\mathbf{A}^t)\right\} . \tag{1}$$

The matrix $\mathbf{A}$ that maximizes (1) is obtained by finding the eigenvalue decomposition of the matrix:

$$\mathbf{S}_{FD} = \mathbf{S}_W^{-1}\mathbf{S}_E \, , \tag{2}$$

and taking the d eigenvectors whose eigenvalues are the largest ones. Since $\mathbf{S}_E$ is of rank one, $\mathbf{S}_W^{-1}\mathbf{S}_E$ is also of rank one. Thus, the eigenvalue decomposition of $\mathbf{S}_W^{-1}\mathbf{S}_E$ leads to only one non-zero eigenvalue, and hence FD can only reduce to dimension $d = 1$.

Loog and Duin have recently proposed a new LDR technique for normally distributed classes [7], namely LD, which takes the Chernoff distance in the original space into consideration to minimize the error rate in the transformed space. They consider the concept of *directed distance matrices*, and a linear transformation in the original space, to finally generalize Fisher's criterion in the transformed space by substituting the between-class scatter matrix for the corresponding directed distance matrix. The LD criterion consists of obtaining the matrix $\mathbf{A}$ that maximizes the function [7]:

$$\begin{aligned} J_{LD_2}(\mathbf{A}) \quad &= tr\left\{(\mathbf{A}\mathbf{S}_W\mathbf{A}^t)^{-1}\right. \\ &\left.\left[\mathbf{A}\mathbf{S}_E\mathbf{A}^t - \mathbf{A}\mathbf{S}_W^{\frac{1}{2}}\frac{p_1\log(\mathbf{S}_W^{-\frac{1}{2}}\mathbf{S}_1\mathbf{S}_W^{-\frac{1}{2}}) + p_2\log(\mathbf{S}_W^{-\frac{1}{2}}\mathbf{S}_2\mathbf{S}_W^{-\frac{1}{2}})}{p_1p_2}\mathbf{S}_W^{\frac{1}{2}}\mathbf{A}^t\right]\right\} \end{aligned} \tag{3}$$

where the logarithm of a matrix $\mathbf{M}$, $\log(\mathbf{M})$, is defined as:

$$\log(\mathbf{M}) \triangleq \mathbf{\Phi} \log(\mathbf{\Lambda})\mathbf{\Phi}^{-1} \,. \tag{4}$$

with $\mathbf{\Phi}$ and $\mathbf{\Lambda}$ representing the eigenvectors and eigenvalues of $\mathbf{M}$.

The solution to this criterion is given by the matrix $\mathbf{A}$ that is composed of the d eigenvectors (whose eigenvalues are the largest ones) of the following matrix:

$$\mathbf{S}_{LD_2} = \mathbf{S}_W^{-1} \left[\mathbf{S}_E - \mathbf{S}_W^{\frac{1}{2}} \frac{p_1 \log(\mathbf{S}_W^{-\frac{1}{2}} \mathbf{S}_1 \mathbf{S}_W^{-\frac{1}{2}}) + p_2 \log(\mathbf{S}_W^{-\frac{1}{2}} \mathbf{S}_2 \mathbf{S}_W^{-\frac{1}{2}})}{p_1 p_2} \mathbf{S}_W^{\frac{1}{2}} \right] \,. \tag{5}$$

The FD criterion discussed above aims to minimize the classification error by maximizing the Mahalanobis distance between distributions, resulting in an optimal criterion (in the Bayesian context) only when the covariance matrices are equal [6], while the LD criterion utilizes, as pointed out above, a directed distance matrix, which is incorporated in Fisher's criterion assuming the within-class scatter matrix is the identity.

1.2 Highlights of the Proposed Criterion

In this paper, we take advantage of the relationship between the probability of classification error of the optimal (in the Bayesian sense) classifier and the Chernoff distance, and propose a new criterion for LDR that aims to maximize the separability of the distributions in the transformed space based on the Chernoff measure. Since we are assuming the original distributions are normal, the distributions in the transformed space are also normal[1]. Thus, the Bayes classifier in the transformed space is quadratic and the classification error (also known as *true* error [5]) does not have a closed-form expression. Let $p(\mathbf{y}|\omega_i)$ be the class-conditional probability that a vector $\mathbf{y} = \mathbf{A}\mathbf{x}$ in the transformed space belongs to class ω_i. The probability of error can be bounded by the Chernoff distance between two distributions as follows [5]:

[1] We note, however, that this assumption is not necessarily true in practice, and that our proposed criterion is still efficient even when data has other distributions, as shown in the empirical result section.

$$\begin{aligned} \Pr[\text{error}] &= \int_{\mathcal{R}_2} p_1 p(\mathbf{y}|\omega_1) d\mathbf{y} + \int_{\mathcal{R}_1} p_2 p(\mathbf{y}|\omega_2) d\mathbf{y} & (6) \\ &\leq p_1^{\beta} p_2^{1-\beta} \int p^{\beta}(\mathbf{y}|\omega_1) p^{1-\beta}(\mathbf{y}|\omega_2) d\mathbf{y} = p_1^{\beta} p_2^{1-\beta} e^{-k(\beta,\mathbf{A})}\,, & (7) \end{aligned}$$

where $\mathcal{R}_1$ and $\mathcal{R}_2$ are the regions in which an object is assigned to class ω_1 or ω_2 respectively. For normally distributed classes, it can be shown that the Chernoff distance is given by [5]:

$$\begin{aligned} k(\beta,\mathbf{A}) &= \frac{\beta(1-\beta)}{2}(\mathbf{Am_1} - \mathbf{Am_2})^t[\beta\mathbf{A}\mathbf{S}_1\mathbf{A}^t + (1-\beta)\mathbf{A}\mathbf{S}_2\mathbf{A}^t]^{-1}(\mathbf{Am_1} - \mathbf{Am_2}) \\ &\quad + \frac{1}{2}\log\frac{|\beta\mathbf{A}\mathbf{S}_1\mathbf{A}^t + (1-\beta)\mathbf{A}\mathbf{S}_2\mathbf{A}^t|}{|\mathbf{A}\mathbf{S}_1\mathbf{A}^t|^{\beta}|\mathbf{A}\mathbf{S}_2\mathbf{A}^t|^{1-\beta}}\,, \qquad (8) \end{aligned}$$

where $\beta \in [0,1]$.

The larger the value of $k(\beta,\mathbf{A})$ is, the smaller the bound for the classification error is, and hence, in this paper, we propose to maximize (8). To clarify this, we note that the FD criterion also aims to maximize the separability between distributions in the transformed space, but coincides with the optimal classifier only when the latter is linear, i.e. when the covariance matrices are coincident, a rare case. As observed above, the LD criterion utilizes the Chernoff distance in its directed distance matrix but in the original space. This criterion, however, does not optimize such a distance in the *transformed* space, as it can be observed in the example given below. A few remarks are discussed prior to the example.

For normally distributed classes, (7) and (8) are useful for approximating the probability of error for the *optimal* (Bayesian) classifier. Since this is not usually the case for real-life data, other factors should be taken into consideration. First, normal distributions are characterized by the first two moments, while it is not (always) the case for real-life data. As pointed out in [5], the Chernoff bound can still be used when normality is not in place; however, it is not as accurate as for normal data. Second, the distribution of the real-life data is usually not known, and it could follow a certain distribution function, not necessarily normal, or even a mixture of distribution functions. Third, it is important to note that the advantages of the Chernoff distance are taken in the context of a quadratic (Bayesian for normal distributions), and hence changing the classifier, the resulting classification will change too. Thus, using a linear or a kernel-based classifier could not have the same effect as the quadratic classifier; however, the empirical results presented later show that still good results presented later are obtained on real-life data.

Consider two normally distributed two-dimensional random vectors, $\mathbf{x}_1 \sim N(\mathbf{m}_1, \mathbf{S}_1)$ and $\mathbf{x}_2 \sim N(\mathbf{m}_2, \mathbf{S}_2)$, where the underlying parameters are: $\mathbf{m}_1 = [0.5001, 0.4947]^t$, $\mathbf{m}_2 = [2.1069, 1.4324]^t$, $\mathbf{S}_1 = [0.8205, 0.4177; 0.4177, 2.8910]$, $\mathbf{S}_2 = [5.1150, -4.3990; -4.3990, 5.7119]$, $p_1 = 0.5479$. Consider also a linear transformation $\mathbf{y} = \mathbf{A}\mathbf{x}$ to the one-dimensional space, i.e. $\mathbf{A}$ is of order 1×2, or equivalently $\mathbf{A}^t$ is a two-dimensional vector. As shown later in the paper, we can just "rotate" $\mathbf{A}^t$ and produce different values for the Chernoff distance in the transformed space, and *only one* value for each angle. Thus, in Figure 1, we plot three different criteria for all possible values of the angle θ between $\mathbf{A}^t$ and $[1, 0]^t$, including $J_F(\mathbf{A})$ computed as in (1), $J_{LD}(\mathbf{A})$ computed as in (33), and the Chernoff distance in the transformed one-dimensional space computed as in (8), where $\beta = p_1$. The probability of classification error in the transformed space, computed as in (6), is also plotted. The dotted vertical lines represent the points at which the three criteria achieve a maximum value, a single maximum for $J_F(\mathbf{A})$ and $J_{LD}(\mathbf{A})$, and two maxima (a local and a global) for $k(\beta, \mathbf{A})$. The solid vertical line represents the point at which Pr[error] is minimum, achieving a value of Pr[error] = 0.2083. To compare the latter with that of the three criteria, we note that the probability of error for the global maximum for $k(\beta, \mathbf{A})$ is 0.2085, only 0.0002 away from the optimal, while the probabilities of error for $J_F(\mathbf{A})$ and $J_{LD}(\mathbf{A})$ are 0.3417 and 0.3616 respectively, 0.1334 and 0.1533 away from the optimal. We also noticed that the probability of error for the local maximum of $k(\beta, \mathbf{A})$ is 0.3325, which is even smaller than those of $J_F(\mathbf{A})$ and $J_{LD}(\mathbf{A})$. As we will also show later, we note that maximizing the criterion $J_F(\mathbf{A})$ or $J_{LD}(\mathbf{A})$ does not necessarily imply maximizing the Chernoff distance in the transformed

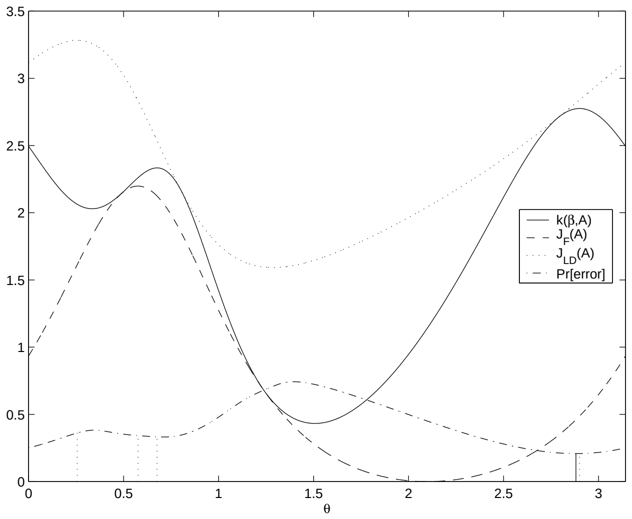


Figure 1: Plot of three different dimensionality reduction criteria, namely Fisher's, Loog-Duin's, and the Chernoff distance in the transformed space, for a two-dimensional to one-dimensional reduction example. The probability of error, computed as in (6), is also plotted. The x-axis represents the different angles of the transformation vector $\mathbf{A}$.

space (as our criterion aims to), and hence minimizing the classification error. Also, we observe that the $k(\beta, \mathbf{A})$ function has more than one peak and so, as shown later, it is not possible to find a closed-form expression for the optimal solution.

2 The Proposed LDR: Two-class Case

The criterion that we propose aims to maximize the Chernoff distance between the transformed random vectors, as in (8). Note that the function k in (8) has two parameters, β and $\mathbf{A}$, which have to be optimized. While for a given transformation matrix $\mathbf{A}$, β takes different values in $[0, 1]$, here, we consider the heuristic adopted in [7], i.e. $p_1 = \beta$ and $p_2 = 1 - \beta$ as a way for "weighting" the respective covariance matrices in the Chernoff distance. Considering different values of β is a problem that deserves further investigation.

2.1 The Criterion

Since after the transformation, new random vectors of the form $\mathbf{y}_1 \sim N(\mathbf{A}\mathbf{m}_1; \mathbf{A}\mathbf{S}_1\mathbf{A}^t)$ and $\mathbf{y}_2 \sim N(\mathbf{A}\mathbf{m}_2; \mathbf{A}\mathbf{S}_2\mathbf{A}^t)$ are obtained, the aim is to find the matrix $\mathbf{A}$ that maximizes:

$$J^*_{c_{12}}(\mathbf{A}) = p_1 p_2 (\mathbf{Am}_1 - \mathbf{Am}_2)^t [\mathbf{AS}_W \mathbf{A}^t]^{-1} (\mathbf{Am}_1 - \mathbf{Am}_2) + \log \left(\frac{|\mathbf{AS}_W \mathbf{A}^t|}{|\mathbf{AS}_1 \mathbf{A}|^{p_1} |\mathbf{AS}_2 \mathbf{A}|^{p_2}} \right) , \quad (9)$$

where $\mathbf{S}_W = p_1 \mathbf{S}_1 + p_2 \mathbf{S}_2$, and the logarithm of a matrix $\mathbf{M}$, $\log(\mathbf{M})$, is defined as in (4). Using this definition, it follows that $\log |\mathbf{B}| = tr\{\log(\mathbf{B})\}$. Also, since $tr\{\mathbf{BCD}\} = tr\{\mathbf{DBC}\}$, we have:

$$\begin{aligned} & (\mathbf{Am}_1 - \mathbf{Am}_2)^t [\mathbf{AS}_W \mathbf{A}^t]^{-1} (\mathbf{Am}_1 - \mathbf{Am}_2) && (10) \\ = \; & tr\{(\mathbf{Am}_1 - \mathbf{Am}_2)^t [\mathbf{AS}_W \mathbf{A}^t]^{-1} (\mathbf{Am}_1 - \mathbf{Am}_2)\} && (11) \\ = \; & tr\{(\mathbf{Am}_1 - \mathbf{Am}_2)(\mathbf{Am}_1 - \mathbf{Am}_2)^t [\mathbf{AS}_W \mathbf{A}^t]^{-1}\} && (12) \\ = \; & tr\{\mathbf{AS}_E \mathbf{A}^t (\mathbf{AS}_W \mathbf{A}^t)^{-1}\} \, , && (13) \end{aligned}$$

where $\mathbf{S}_E = (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^t$. In this way, maximizing (9) is equivalent to maximizing:

$$J^*_{c_{12}}(\mathbf{A}) = tr \left\{ p_1 p_2 \mathbf{AS}_E \mathbf{A}^t (\mathbf{AS}_W \mathbf{A}^t)^{-1} + \log(\mathbf{AS}_W \mathbf{A}^t) - p_1 \log(\mathbf{AS}_1 \mathbf{A}^t) - p_2 \log(\mathbf{AS}_2 \mathbf{A}^t) \right\} \quad (14)$$

Note that for any value of $J^*_{c_{12}}(\mathbf{A})$, where the rows of $\mathbf{A}$ are linearly independent, there exists an orthogonal matrix $\mathbf{Q}$ such that the Chernoff distance $J^*_{c_{12}}(\mathbf{Q})$ is the same as that of $\mathbf{A}$. That is, the solution is always found in a compact set $\{\mathbf{Q} : \mathbf{QQ}^t = \mathbf{I}_d\}$, and thus:

$$max_{\{\mathbf{A}\}} J^*_{c_{12}}(\mathbf{A}) = max_{\{\mathbf{Q}:\mathbf{QQ}^t = \mathbf{I}_d\}} J^*_{c_{12}}(\mathbf{Q}) \quad (15)$$

This follows by decomposing $\mathbf{A}^t = \mathbf{RQ}$, where $\mathbf{R}$ is of order $d \times d$ and lower triangular, and $\mathbf{Q}$ is of order $d \times n$, such that $\mathbf{QQ}^t = \mathbf{I}_d$ (see Appendix A.1).

The relationship between the proposed criterion and two well-known criteria, FD and

LD, follows. When $\mathbf{S}_1 = \mathbf{S}_2$, it is true that $J_{FD} = J_{LD_2}$. That is, FD will only lead to a linear, optimal (in the Bayesian sense) classifier only when the covariance matrices are coincident (although there are other cases in which the optimal classifier could be given in terms of a pair of linear functions, and which are not discussed here – cf. [16]).

The relationship between the proposed and the LD criteria is not straightforward – we thus analyze it for particular cases only. We assume that $\mathbf{S}_1$ and $\mathbf{S}_2$ are diagonal, and that $\mathbf{A}$ is a $d \times n$ matrix with its d rows orthogonal to each other, i.e. $\mathbf{A}\mathbf{A}^t = \mathbf{I}_d$. Assume also that, pre and post-multiplying by $\mathbf{S}_W^{\frac{1}{2}}$, so that $\mathbf{S}_W = p_1\mathbf{S}_1 + p_2\mathbf{S}_2 = \mathbf{I}_n$, it can be shown that (see Appendix A.2) $J_{LD_2}(\mathbf{A}) = J^*_{c_{12}}(\mathbf{A})$ only when the transformation matrix is of the form $\mathbf{A} = [[0, \ldots, 0, 1_{i_1}, 0, \ldots, 0]^t \, [0, \ldots, 0, 1_{i_2}, 0, \ldots, 0]^t \, \ldots \, [0, \ldots, 0, 1_{i_d}, 0, \ldots, 0]^t]^t$, where $i_j \neq i_k$, $j, k = 1, \ldots, d$. Additionally, analyzing the first order necessary conditions, it follows that $\nabla J_{LD_2}(\mathbf{A}) = \nabla J^*_{C_{12}}(\mathbf{A})$ only if $\mathbf{A}(\log \mathbf{S}_1 + \log \mathbf{S}_2) = (\mathbf{A}\mathbf{S}_1\mathbf{A}^t)^{-1}\mathbf{A}\mathbf{S}_1 + (\mathbf{A}\mathbf{S}_2\mathbf{A}^t)^{-1}\mathbf{A}\mathbf{S}_2 - 2\mathbf{A}$. This is not necessarily true, except in very restricted cases. Suppose for example, and without loss of generality, that $\mathbf{A}$ is $1 \times d$ and the 1 is at the first postion. Since $\mathbf{A}^t$ is an eigenvector of $\mathbf{S}_{LD_2}$, it implies that $(m_{11} - m_{21})^2 = p_1 \log \lambda_{11} + p_2 \log \lambda_{21}$. That is, the square difference between the first components of the means will have to be the same as the weighted sum of the logs of the first eigenvalues of $\mathbf{S}_1$ and $\mathbf{S}_2$.

From the above analysis (and Appendix A.2), we conclude the following. The special case in which both J_{LD_2} and $J^*_{c_{12}}$ coincide, is when $\mathbf{A}$ has a row with exactly one 1 and the rest of the components 0, and the 1s are at different columns. $\mathbf{A}$ contains the eigenvectors of S_{LD_2}, and which also coincide with the optimal solution to $J^*_{c_{12}}$. This implies that, first, the projection is carried out onto an orthogonal subspace whose basis is canonical, and that basis will represent a few features of the original space. Second, all features, except those at positions $i_1, i_2, \ldots, i_d$ will be excluded in the transformed space, thus, transforming the projection scheme into a *feature selection* method rather than a *linear dimensionality reduction* method. Third, we seek for a combination of features so that we take advantage of all features from the original space (and not some of them) towards maximizing the Chernoff distance in the transformed space.

2.2 The Algorithm and its Convergence

In order to maximize $J^*_{c_{12}}$, we propose the following algorithm, which is based on the gradient method (how difficult is to find a direct solution is discussed in Appendix A.3). The learning rate, one of the parameters to the algorithm is obtained by maximizing the objective function in the direction of the gradient. The first task to do is to find the gradient matrix using the corresponding operator, ∇, in the following manner:

$$\begin{aligned}\nabla J^*_{c_{12}}(\mathbf{A}) = \frac{\partial J^*_{c_{12}}}{\partial \mathbf{A}} &= 2p_1p_2\left[\mathbf{S}_E\mathbf{A}^t(\mathbf{A}\mathbf{S}_W\mathbf{A}^t)^{-1} - \mathbf{S}_W\mathbf{A}^t(\mathbf{A}\mathbf{S}_W\mathbf{A}^t)^{-1}(\mathbf{A}\mathbf{S}_E\mathbf{A}^t)(\mathbf{A}\mathbf{S}_W\mathbf{A}^t)^{-1}\right]^t \\ &\quad +2\left[\mathbf{S}_W\mathbf{A}^t(\mathbf{A}\mathbf{S}_W\mathbf{A}^t)^{-1} - p_1\mathbf{S}_1\mathbf{A}^t(\mathbf{A}\mathbf{S}_1\mathbf{A}^t)^{-1} - p_2\mathbf{S}_2\mathbf{A}^t(\mathbf{A}\mathbf{S}_2\mathbf{A}^t)^{-1}\right]^t \quad (16)\end{aligned}$$

The formal procedure that maximizes $J^*_{c_{12}}$ is shown in Algorithm **Chernoff_LDA_Two** given below. The algorithm receives as a parameter, a threshold, τ, which indicates when the search will stop. Also, by (29), once $\mathbf{A}$ is obtained, there always exists an orthogonal matrix $\mathbf{Q}$ such that $\mathbf{A}$ can be decomposed into $\mathbf{RQ}$. An additional step is then introduced in the algorithm, which decomposes $\mathbf{A}$ into $\mathbf{RQ}$, and utilizes $\mathbf{Q}$ in the next step. Note that this regularization could be avoided, and hence the algorithm will also converge. However, we include it for the following reasons. First, the set of all matrices of order $d \times n$ along with its usual topology, the set $\{\mathbf{Q} : \mathbf{Q}\mathbf{Q}^t = \mathbf{I}_d\}$ is a compact set; then, any continuous function achieves its maximum in that compact set. Second, we have empirically found that searching for a solution in the compact set runs faster than searching in the whole set. Third, initialization of the secant method, as seen later, is easier as initial values can be chosen in the compact set, i.e. by angle and not by value. Fourth, note that optimizing $J^*_{c_{12}}$ could be stated as a constrained optimization problem. However, the constraint $\mathbf{Q}\mathbf{Q}^t = \mathbf{I}_d$ imposes adding a multiplier. Due to the iterative nature of the solution, an extra step will be required to find the corresponding multiplier, which is, in our case, avoided by imposing the RQ decomposition.

```
Algorithm Chernoff_LDA_Two
    Input: Threshold $\tau$
    begin
        $\mathbf{A}^{(0)} \leftarrow \max_{\mathbf{A}}\{J^*_{c_{12}}(\mathbf{A}_{FD}), J^*_{c_{12}}(\mathbf{A}_{LD})\}$  // Max. of Fisher's and Loog-Duin's methods
        $k \leftarrow 0$
        repeat
            $\eta_k \leftarrow max_{\eta>0}\phi_{k_{12}}(\eta)$
            $\mathbf{B} \leftarrow \mathbf{A}^{(k)} + \eta_k \nabla J^*_{c_{12}}(\mathbf{A}^{(k)})$
            Decompose $\mathbf{B}$ into $\mathbf{R}$ and $\mathbf{Q}$
            $\mathbf{A}^{(k+1)} \leftarrow \mathbf{Q}$
            $k \leftarrow k+1$
        until $|J^*_{c_{12}}(\mathbf{A}^{(k-1)}) - J^*_{c_{12}}(\mathbf{A}^{(k)})| < \tau$
        return $\mathbf{A}^{(k)}$, $J^*_{c_{12}}(\mathbf{A}^{(k)})$
    end
```

It is not difficult to see that Algorithm **Chernoff_LDA_Two** converges. The convergence argument is a generalization of that of the gradient algorithm given in [3]. While that proof is for vectors only, i.e. for reducing to dimension one, our case deals with matrices of order $d \times n$. It, thus follows that (see Appendix A.4), if $\{\mathbf{A}^{(k)}\}_{k=1}^{\infty}$ is the sequence of matrices generated by Algorithm **Chernoff_LDA_Two**, then if $\nabla J^*_{c_{12}}(\mathbf{A}^{(k)}) \neq 0$, $J^*_{c_{12}}(\mathbf{A}^{(k)}) < J^*_{c_{12}}(\mathbf{A}^{(k+1)})$. Otherwise, the algorithm terminates.

Algorithm **Chernoff_LDA_Two** needs a learning rate, η_k, which when small, convergence is slower but more likely, while when η_k is large, convergence is faster but less likely. However, when η_k is chosen carefully as in the algorithm, i.e. $\eta_k \leftarrow max_{\eta>0}\phi_{k_{12}}(\eta)$, the algorithm always converges. There are many ways of computing η_k, one of them being the expression that maximizes the value of $J^*_{c_{12}}$ in the next step [3]. Consider the following function of η:

$$\phi_{k_{12}}(\eta) = J^*_{c_{12}}(\mathbf{A}^{(k)} + \eta \nabla J^*_{c_{12}}(\mathbf{A}^{(k)}))\,. \tag{17}$$

The secant method can be used to optimize $\phi_{k_{12}}(\eta)$. Starting from initial values of $\eta^{(0)}$ and $\eta^{(1)}$, at step $j+1$, η is updated as follows:

$$\eta^{(j+1)} = \eta^{(j)} + \frac{\eta^{(j)} - \eta^{(j-1)}}{\frac{d\phi_{k_{12}}}{d\eta}(\eta^{(j)}) - \frac{d\phi_{k_{12}}}{d\eta}(\eta^{(j-1)})} \frac{d\phi_{k_{12}}}{d\eta}(\eta^{(j)})\,, \tag{18}$$

where $\frac{d\phi_{k_{12}}}{d\eta}$ is obtained by using Equation (43). This procedure is repeated until the difference between $\eta^{(j-1)}$ and $\eta^{(j)}$ is as small as desired. One of the initial values of η is $\eta_0 = 0$ and the other value of η_1 resulting from the angle difference between $\mathbf{A}$ at step k and the matrix obtained by adding the latter and the product between the learning rate and the gradient matrix, as follows (see Appendix A.5):

$$\eta_1 = \frac{d^2\epsilon - d}{tr\{\mathbf{A}^{(k)}[\nabla J^*_{c_{12}}(\mathbf{A}^{(k)})]^t\}}\,, \tag{19}$$

where $\epsilon = \cos\theta$, and θ is the angle between $\mathbf{A}^{(k)}$ and $[\mathbf{A}^{(k)} + \eta_k \nabla J^*_{c_{12}}(\mathbf{A}^{(k)})]$ with these two matrices residing in a hypersphere of radius d.

Geometrically speaking, since $\|\mathbf{A}\|_F$ is a norm that satisfies the properties of a metric, we can ensure that there exists a matrix norm $\|\mathbf{A}\|$ *induced* or *compatible* in $\mathbb{R}^n$, such that for any $\mathbf{A} \neq 0$, $\|\mathbf{A}\| = \sqrt{\lambda_1}$ holds, where λ_1 is the largest eigenvalue of $\mathbf{A}$ [3, pp.33]. Then, since that eigenvalue is $\lambda_1 = 1$, the matrix norm induced results in $\left\|\mathbf{A}^{(k)}\right\| = \left\|\mathbf{A}^{(k+1)}\right\| = 1$. In this way, we ensure that the rows of $\mathbf{A}^{(k)}$ and $\mathbf{A}^{(k+1)}$ reside in the same hypersphere in $\mathbb{R}^n$, whose radius is unity. Then, since those rows are linearly independent, they *could* be "rotated" independently using a vector $\boldsymbol{\eta}$ of dimension d. However, Algorithm **Chernoff_LDA_Two** uses a scalar instead of a vector. For this reason, the "rotation" can be seen on a hypersphere of radius d and all the rows of $\mathbf{A}$ are rotated using the same scalar. As an example, if we choose $\hat{\theta} = \pi/180$, and suppose that $\mathbf{A}^{(k)}$ is of order $1 \times n$, i.e. a vector in $\mathbb{R}^n$, we obtain a value of $\epsilon \approx 0.9998$. Thus the variation between $\mathbf{A}^{(k)}$ and $\mathbf{A}^{(k+1)}$ is one degree, where, obviously, the value of η_1 depends also on $\mathbf{A}^{(k)}$ and $\nabla J^*_{c12}(\mathbf{A}^{(k)})$. Note that we are considering that $\mathbf{A}^{(k)}$ is the matrix $\mathbf{Q}$, which is orthogonal and obtained by means of the RQ decomposition.

To conclude this section, we note that we could use a vector $\boldsymbol{\eta}$ to update the matrix $\mathbf{A}^{(k)}$, instead of a scalar. This would change the direction of each row in $\mathbf{A}$ independently, since these rows are linearly independent and compose a basis in $\mathbb{R}^d$. Thus, each of these rows would be "rotated" by using a different scalar η_i, where the d scalars compose the vector $\boldsymbol{\eta}$. Studying this problem is one of the possible extensions of our work presented here in this paper.

3 Multi-class Case

For the multi-class problem we assume that we are dealing with k classes, $\omega_1, \ldots, \omega_k$, whose *a priori* probabilities are given by $p_1, \ldots, p_k$, and which are represented by k n-dimensional normally distributed random vectors, $\mathbf{x}_1 \sim N(\mathbf{m}_1; \mathbf{S}_1), \ldots, \mathbf{x}_k \sim N(\mathbf{m}_k; \mathbf{S}_k)$. For the FD criterion, the following definitions are used: $\mathbf{S}_E = \sum_{i=1}^{k} p_i(\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^t$, where $\mathbf{m} = \sum_{i=1}^{k} p_i \mathbf{m}_i$, and $\mathbf{S}_W = \sum_{i=1}^{k} p_i \mathbf{S}_i$. Then, the FD approach aims to find a matrix $\mathbf{A}$ that maximizes the criterion function given in (1), and which is obtained by finding the d eigenvalues (whose eigenvectors are the largest ones) of the matrix given in (2).

The LD criterion for the multi-class problem aims to find the $d \times n$ transformation matrix $\mathbf{A}$ that maximizes the following function [7]:

$$\begin{aligned} J_{LD}(\mathbf{A}) \;=\; & \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} p_i p_j tr \Big\{ (\mathbf{A}\mathbf{S}_W\mathbf{A}^t)^{-1} \mathbf{A}\mathbf{S}_W^{\frac{1}{2}} \\ & \Big[(\mathbf{S}_W^{-\frac{1}{2}}\mathbf{S}_{ij}\mathbf{S}_W^{-\frac{1}{2}})^{-\frac{1}{2}} \mathbf{S}_W^{-\frac{1}{2}} \mathbf{S}_{E_{ij}} \mathbf{S}_W^{-\frac{1}{2}} (\mathbf{S}_W^{-\frac{1}{2}}\mathbf{S}_{ij}\mathbf{S}_W^{-\frac{1}{2}})^{-\frac{1}{2}} + \frac{1}{\pi_i \pi_j} \Big(\log(\mathbf{S}_W^{-\frac{1}{2}}\mathbf{S}_{ij}\mathbf{S}_W^{-\frac{1}{2}}) \\ & - \pi_i \log(\mathbf{S}_W^{-\frac{1}{2}}\mathbf{S}_i\mathbf{S}_W^{-\frac{1}{2}}) - \pi_j \log(\mathbf{S}_W^{-\frac{1}{2}}\mathbf{S}_j\mathbf{S}_W^{-\frac{1}{2}}) \Big) \Big] \mathbf{S}_W^{\frac{1}{2}} \mathbf{A}^t \Big\} , \end{aligned} \tag{20}$$

where $\mathbf{S}_{E_{ij}} = (\mathbf{m}_i - \mathbf{m}_j)(\mathbf{m}_i - \mathbf{m}_j)^t$, $\pi_i = \frac{p_i}{p_i + p_j}$, $\pi_j = \frac{p_j}{p_i + p_j}$, and $\mathbf{S}_{ij} = \pi_i \mathbf{S}_i + \pi_j \mathbf{S}_j$. The multi-class LD criterion is maximized as it is done for the two-dimensional case, by finding the matrix $\mathbf{A}$ composed of the d eigenvectors (whose eigenvalues are the largest ones) of the following matrix:

$$\begin{aligned}\mathbf{S}_{LD} &= \sum_{i=1}^{k-1}\sum_{j=i+1}^{k} p_i p_j \mathbf{S}_W^{-1}\mathbf{S}_W^{\frac{1}{2}}\left[(\mathbf{S}_W^{-\frac{1}{2}}\mathbf{S}_{ij}\mathbf{S}_W^{-\frac{1}{2}})^{-\frac{1}{2}}\mathbf{S}_W^{-\frac{1}{2}}\mathbf{S}_{E_{ij}}\mathbf{S}_W^{-\frac{1}{2}}(\mathbf{S}_W^{-\frac{1}{2}}\mathbf{S}_{ij}\mathbf{S}_W^{-\frac{1}{2}})^{-\frac{1}{2}}\right.\\ &\left.+\frac{1}{\pi_i\pi_j}\left(\log(\mathbf{S}_W^{-\frac{1}{2}}\mathbf{S}_{ij}\mathbf{S}_W^{-\frac{1}{2}}) - \pi_i\log(\mathbf{S}_W^{-\frac{1}{2}}\mathbf{S}_i\mathbf{S}_W^{-\frac{1}{2}}) - \pi_j\log(\mathbf{S}_W^{-\frac{1}{2}}\mathbf{S}_j\mathbf{S}_W^{-\frac{1}{2}})\right)\right]\mathbf{S}_W^{\frac{1}{2}}\,, \qquad (21)\end{aligned}$$

Our multi-class criterion is not straightforward as the Chernoff distance is not defined for more than two distributions. This is also the case of other classifiers, such as the well-known support vector machines or kernel-based classifiers, for which majority votes of $k(k-1)/2$ decisions are among the most widely-used schemes [1], as opposed to other schemes like one-against-all or all-at-once, which suffer the problem of yielding unclassifiable regions [5]. In our case, however, it is natural to maximize the *weighted* sum of pairwise Chernoff distances between classes ω_i and ω_j, for all $i = 1, \ldots, k-1$, $j = i, \ldots, k$. The "weights" used for the pairwise class criterion are given by the normalized joint prior probabilities between classes ω_i and ω_j, $\pi_i\pi_j$. The criterion, thus, consists of finding the optimal transformation $\mathbf{Ax}$, where $\mathbf{A}$ is a matrix of order $d \times n$ that maximizes the function:

$$J_c^*(\mathbf{A}) = \sum_{i=1}^{k-1}\sum_{j=i+1}^{k} J_{c_{ij}}^*(\mathbf{A})\,, \qquad (22)$$

with:

$$J_{c_{ij}}^*(\mathbf{A}) = tr\left\{\pi_i\pi_j(\mathbf{AS}_{W_{ij}}\mathbf{A}^t)^{-1}\mathbf{AS}_{E_{ij}}\mathbf{A}^t + \log(\mathbf{AS}_{W_{ij}}\mathbf{A}^t) - \pi_i\log(\mathbf{AS}_i\mathbf{A}^t) - \pi_j\log(\mathbf{AS}_j\mathbf{A}^t)\right\} \qquad (23)$$

The gradient matrix, given by the first-order necessary condition, is the following:

$$\nabla J_c^*(\mathbf{A}) = \frac{\partial}{\partial\mathbf{A}}\sum_{i=1}^{k-1}\sum_{j=i+1}^{k} J_{c_{ij}}^*(\mathbf{A}) = \sum_{i=1}^{k-1}\sum_{j=i+1}^{k}\nabla J_{c_{ij}}^*(\mathbf{A})\,, \qquad (24)$$

where

$$\begin{aligned}\nabla J^*_{c_{ij}}(\mathbf{A}) = 2\pi_i\pi_j \left[\mathbf{S}_{E_{ij}}\mathbf{A}^t(\mathbf{A}\mathbf{S}_{W_{ij}}\mathbf{A}^t)^{-1} - \mathbf{S}_{W_{ij}}\mathbf{A}^t(\mathbf{A}\mathbf{S}_{W_{ij}}\mathbf{A}^t)^{-1}(\mathbf{A}\mathbf{S}_{E_{ij}}\mathbf{A}^t)(\mathbf{A}\mathbf{S}_{W_{ij}}\mathbf{A}^t)^{-1}\right]^t \\ +2\left[\mathbf{S}_{W_{ij}}\mathbf{A}^t(\mathbf{A}\mathbf{S}_{W_{ij}}\mathbf{A}^t)^{-1} - \pi_i\mathbf{S}_i\mathbf{A}^t(\mathbf{A}\mathbf{S}_i\mathbf{A}^t)^{-1} - \pi_j\mathbf{S}_j\mathbf{A}^t(\mathbf{A}\mathbf{S}_j\mathbf{A}^t)^{-1}\right]^t\end{aligned} \quad (25)$$

In order to find the matrix $\mathbf{A}$ that maximizes $J^*_c(\mathbf{A})$, we use the same algorithm as for the two-class case. The convergence proofs and initialization procedures are also a natural extension of the two-class case. In the experimental section, we show some empirical results for the multi-class criterion proposed here, which shows the advantages of optimizing the Chernoff distance in the transformed space.

4 Empirical Results

In order to evaluate the classification performance of the proposed criterion, we present an empirical analysis of the classification accuracy and the Chernoff distance in the transformed space on synthetic and real-life data. Three LDR techniques are compared, namely FD and LD as discussed in Section 1, and the proposed method, as presented in Sections 2 and 3, namely RH. In order to analyze the classification power of the LDR techniques, two classifiers are used in the transformed space, the linear and quadratic classifiers.

4.1 Synthetic Data

The tests on synthetic data involve ten different datasets of dimensions $n = 10, 20, \ldots, 100$ each with two randomly generated normally distributed classes. The two classes of each dataset, ω_1 and ω_2, are then fully specified by their parameters, $\mathbf{m}_1$, $\mathbf{m}_2$, $\mathbf{S}_1$ and $\mathbf{S}_2$. Each element of the means, $\mathbf{m}_1$ and $\mathbf{m}_2$, was generated by following distributions $\mathrm{U}[0, b/n]$ and $\mathrm{U}[b/n, 2b/n]$, where b was set to 10. Dividing by n makes sure that the classification task is not easier when increasing the dimension. The eigenvalues of the covariances, $\mathbf{S}_1$ and

n	FD+Q	d^*	LD+Q	d^*	RH+Q	d^*	FD+L	d^*	LD+L	d^*	RH+L	d^*
10	0.28653	1	0.05314*	9	0.05323	9	0.28979	1	0.28882*	6	0.28883	9
20	0.22255	1	0.01968	18	0.01958*	18	0.22700	1	0.22018	3	0.21878*	4
30	0.15119	1	0.00269*	24	0.00269*	24	0.18218*	1	0.18248	27	0.18248	27
40	0.28725	1	0.00660	36	0.00657*	36	0.29784	1	0.29537	8	0.29466*	6
50	0.37045	1	0.00549*	49	0.00549*	49	0.39616*	1	0.39745	1	0.39745	1
60	0.32076	1	0.00068*	56	0.00068*	56	0.32292	1	0.31603	21	0.31525*	23
70	0.38187	1	0.00001*	28	0.00001*	28	0.38196	1	0.38191*	30	0.38191*	30
80	0.32314	1	0.00000*	37	0.00000*	37	0.34298	1	0.33417	23	0.33408*	25
90	0.32474	1	0.00000*	30	0.00000*	30	0.32636	1	0.32474*	1	0.32474*	1
100	0.19861	1	0.00000*	31	0.00000*	31	0.27859*	1	0.27873	78	0.27872	72

Table 1: Error Rates for the three LDR methods, FD, LD and RH, where the samples are projected onto the d^*-dimensional space with d^* giving the lowest error rate for $d = 1, \ldots, n-1$.

$\mathbf{S}_2$, were randomly generated as U$[0, b]$, and the corresponding eigenvectors from a random matrix in U$(0, b/n)$ followed by a QR decomposition, taking the orthogonal matrix $\mathbf{Q}$. This ensures that the covariances are positive and definite. A linear transformation using $\mathbf{S}_1^{-\frac{1}{2}}$ was applied, obtaining covariances $\mathbf{I}$ and $\mathbf{S}_1^{-\frac{1}{2}}\mathbf{S}_2\mathbf{S}_1^{-\frac{1}{2}}$ respectively, followed by a subsequent linear transformation using $\mathbf{\Phi}_2$, which contains the eigenvectors of $\mathbf{S}_1^{-\frac{1}{2}}\mathbf{S}_2\mathbf{S}_1^{-\frac{1}{2}}$. After all the transformations, the underlying covariance matrices resulted in $\mathbf{I}$ and $\mathbf{\Lambda}_2$. We also randomly generated p_1 as a U$[0.3, 0.7]$ and assigned $p_2 = 1 - p_1$. We trained three LDR techniques, FD, LD and RH using these parameters, and for each dataset we generated 100,000 samples for testing purposes. For each dataset, we found the corresponding transformation matrix $\mathbf{A}$ for each dimension $d = 1, \ldots, n-1$. After the linear transformation is performed we have tested two classifiers: the linear (L) classifier, which is obtained by averaging the covariances matrices in the transformed space, and the quadratic (Q) classifier which is the one that minimizes the probability of classification error assuming that the parameters in the transformed normally distributed data are given by $\mathbf{A}\mathbf{m}_i$ and $\mathbf{A}\mathbf{S}_i\mathbf{A}^t$ [5].

The minimum error rates obtained for each individual classifier for synthetic data are shown in Table 1. The first column represents the dimension of each datset. The next columns correspond to the error rate and the *best* dimension d^* for the three LDR methods

n	FD	LD	RH
10	0.001	0.042	0.610
20	0.001	0.234	2.483
30	0.002	0.603	5.721
40	0.002	1.395	8.442
50	0.003	2.794	13.524
60	0.003	5.965	77.313
70	0.004	9.790	92.080
80	0.004	15.010	339.722
90	0.004	23.696	400.014
100	0.005	33.150	463.859

Table 2: Execution times for the training phase for the three LDR methods, FD, LD and RH, run on synthetic data.

and for each classifier, quadratic and linear. The '*' symbol beside the error rate indicates that the lowest among the three methods, FD, LD and RH, was obtained. Note that for FD, $d^* = 1$, since, as pointed out earlier, the solution matrix contains only one non-zero eigenvalue. We observe that for the quadratic classifier LD and RH outperformed FD for all the datasets. Also, LD and RH jointly achieved the minimum error rate for seven datasets, while RH obtained the best error rate in nine out of ten datasets. For the linear classifier, again, LD and RH outperformed FD, and also RH achieved the lowest error rate in six out of ten datasets, outperforming LD. In all dimensions, the LDR techniques coupled with the quadratic classifier performed better than the LDR with the linear classifier. This is due to the fact that the data used in the experiments obey the normal distribution. We also observe that RH performed better than LD and FD, when coupled with both linear and quadratic classifiers. Note that RH+Q achieved the lowest classification error in a dimension lower than that of the original space. From these observations we conclude that the best classification is due to: (i) the effect of reducing dimensions with RH, and (ii) the use of a quadratic classifier in a lower dimension, and hence justifying the use of our proposed LDR technique. Table 2 lists the cpu time (in seconds) taken by each of the LDR techniques, for the training phase only, since the classification using either the quadratic or linear classifier takes the same amount of time for each LDR, independently of the training phase. FD, as expected, takes fractions of seconds in all cases as they only reduce to dimension one. Also, LD is much faster than RH, since the latter has to perform a gradient search; however, that search is speeded up using the secant method for finding the best learning parameter. We also note that LD and RH, yet slow, they both have to search over all dimensions, e.g. for $n = 100$, they search for all dimensions from 1 to 99. This time could be reduced drastically if reducing to lower dimensions, say, starting from one and up to a point in which the error rate stabilizes or increases.

When comparing the performance of LD and RH, they both achieve similar error rates, while RH is much (5 to 20 times) slower. These results are included as an indication on how (i) LD and RH outperform FD, and (ii) observe the relationship between the time spent by RH in comparison with LD. Against FD there is no point of comparison, since FD reduces to dimension one and its performance in terms of error is quite poor. RH, however, performs much better than LD (and FD) in real-life data (despite its higher running time), as discussed in the next subsection. Finally we note that the time spent (and compared) by the three methods is critical in the learning phase only, while in the classification stage, the speed will remain constant for both LD and RH.

4.2 Two-class Real-life Data

As in the experiments on synthetic data, to test the LDR method proposed here, and to compare it with others, we also performed a few simulations on real life data which involve 39 two-class, d-dimensional datasets drawn from the UCI machine learning repository [12]. Originally, seven datasets were of two classes, and the others were multi-class, from which we extracted pairs of classes. We assumed the classes are normally distributed, and so the mean and covariance were obtained for each class, and the prior probabilities were estimated as $p_i = n_i/(n_i + n_j)$, where n_i and n_j are the numbers of samples for classes ω_i and ω_j respectively. We trained the quadratic (Q) and linear (L) classifiers on the data transformed by the LDR methods in discussion, namely FD, LD, and RH, and obtained the mean of the error rate for a ten-fold cross-validation experiment. The results for the best value of d, where $d = 1, \dots, n$ with n the dimension of the original space, are shown in Table 3. The first column indicates the name of the dataset and the pair of classes separated by "," (when classes are not given, it means the problem itself is two-class), where the name of the dataset is as follows: W = Wisconsin breast cancer, B = Bupa liver, P = Pima, D = Wisconsin diagnostic breast cancer, C = Cleveland heart-disease, S = SPECTF heart, I = Iris, T = Thyroid, G = Glass, N = Wine, J = Japanese vowels, L = Letter, E = Pendigits, and O = Ionosphere. The other columns represent the error rates[2] as in Table 1. The error rate marked with '*' represents the lowest (optimal) one out of the three LDR, and for the same classifier, e.g. one '*' for quadratic and another '*' for linear. In bold are the error rates that are not significantly different from the optimal, where this significance is obtained from a signed rank test [13] with a significance level of 0.01. The dimension to the right of the '/' indicates that there is a lower dimension for which the error rate is not significantly different from the optimal in that dimension.

For the quadratic classifier, RH obtained the lowest error rate in 29 out of 39 cases. Compared to the other techniques, FD and LD yielded the lowest error rate both in 10 cases. This behavior is also observed for the linear classifier, in which RH wins in 25 cases, while FD and LD obtained the lowest error rates in 9 and 21 cases respectively. It then follows that RH obtained the lowest error rate in more cases than the others. In addition to this, RH shows an excellent performance for obtaining an error rate not significantly different from the optimal. This occurs in 34 out of 39 cases for RH combined with the quadratic classifier, and in 35 out of 39 cases when RH is coupled with the linear classifier.

A more in-depth analysis on the dimension in which each LDR yielded the lowest error rate reinforces the superiority of RH over the other two techniques. RH+Q yields the lowest error rate for dimension *one* in 9 cases. Also, in 17 cases, RH+Q leads to an error rate that is not significantly different from the optimal in dimension *one*. Another point to highlight is that, in most of the cases (32 out of 39), RH+Q achieves the best results in dimensions

[2] To enhance the visualization of the results, we have omitted some pairs that give zero-error classification for all cases. These are I,1,2; I,1,3; T,2,3; G,1,5; N,1,3; J,6,7. Also, since FD always reduces to dimension one, we omit the column d^*.

Dataset	FD+Q	LD+Q	d^*	RH+Q	d^*	FD+L	LD+L	d^*	RH+L	d^*
W	**0.03075**	**0.02783***	1	**0.03075**	1	**0.03962**	**0.03815***	6/5	**0.03962**	1
B	0.36201	0.38857	4	**0.35361***	1	**0.30991**	**0.33016**	5	**0.30126***	5/4
P	**0.22643***	0.25126	2	**0.22643***	1	**0.22903***	**0.23038**	7/5	**0.22903***	1
D	**0.03152***	0.04026	27	**0.03152***	1	0.04207	**0.02988***	20/18	**0.03678**	28
C	**0.16494**	0.16827	11	**0.16137***	11/7	0.16160	**0.15839**	8	**0.14482***	5
S	0.24777	0.04558	41	**0.04281***	36/26	0.23364	**0.17637***	19/11	**0.18037**	15/11
I,2,3	**0.05000**	**0.03000***	1	**0.04000**	2/1	**0.03000***	**0.04000**	1	**0.03000***	1
T,1,2	0.02163	**0.01081**	4/3	**0.00526***	3/2	0.05935	**0.03274**	4	**0.02719***	4
T,1,3	**0.02222***	0.02777	2	0.02777	2	**0.03888**	**0.02777***	4/1	**0.02777***	4/1
G,1,2	**0.31000***	0.39761	7	0.39761	8	**0.28190***	0.29571	8	**0.28904**	7
G,1,3	0.22361	0.20416	1	**0.11250***	8/1	0.22361	0.20416	1	**0.16111***	8/1
G,1,7	**0.02000***	0.04000	8	**0.02000***	1	**0.04000**	**0.03000***	1	**0.04000**	1
G,2,3	**0.15861**	0.21333	8	**0.15361***	8/6	**0.15861***	0.16722	4	0.16611	8
G,2,5	**0.10972**	**0.09833***	7/1	**0.09833***	6/1	**0.09972**	**0.08833***	7/1	**0.08833***	6/1
G,2,7	**0.02727***	0.06363	7	**0.02727***	1	0.04636	0.03727	8	**0.01818***	8/6
G,3,5	**0.00000***	**0.00000***	1	**0.00000***	1	**0.02500**	**0.00000***	6/2	**0.00000***	7/1
G,3,7	**0.06000**	**0.02000***	2	0.04000	4	**0.06000***	**0.06000***	1	**0.06000***	1
G,5,7	**0.05000***	0.07000	4	**0.05000***	1	0.05000	0.05000	8	**0.02500***	2
N,1,2	0.00714	0.00769	6	**0.00000***	6	**0.00769**	**0.00714***	11/9	**0.00769**	1
N,2,3	0.01666	0.01666	3	**0.00833***	7	0.01666	**0.00833***	12	0.01666	1
J,1,2	**0.00143***	0.00526	3	**0.00143***	1	**0.00143***	**0.00143***	11/7	**0.00143***	1
J,1,3	**0.00037***	0.00110	7	**0.00037***	1	**0.00110***	**0.00110***	11/9	**0.00110***	1
J,4,5	0.00751	**0.00177***	7	0.00486	3	0.00441	**0.00088***	9	0.00486	1
J,8,9	**0.06680**	**0.05130***	11/8	**0.05289**	6/1	**0.06947**	**0.07160**	11/8	**0.06840***	8/1
L,C,G	0.08354	0.05109	15	**0.04708***	10	0.08354	0.08490	12	**0.08157***	6
L,D,O	0.03340	**0.01540**	15	**0.01477***	10/8	**0.03278**	**0.03021***	14	**0.03277**	12/3
L,J,T	0.00974	0.00452	10	**0.00387***	8/1	**0.00974**	**0.00974**	15	**0.00908***	10/8
L,K,R	0.09887	**0.04140***	12	**0.04208**	10/9	**0.09620**	**0.09552**	13/12	**0.09420***	1
L,M,N	0.03175	0.01584	13	**0.01459***	13/8	**0.03493**	**0.03303***	13/12	**0.03493**	1
L,O,Q	**0.04559***	0.05728	11	**0.04625**	1	0.04623	0.05013	11	**0.04558***	9/5
L,P,R	0.02050	0.01217	9	**0.01024***	9	**0.02243**	**0.02178***	7	**0.02242**	6/1
L,U,V	0.01074	0.00759	15	**0.00696***	9	0.01201	**0.01138***	10	**0.01138***	9/1
L,V,W	0.02705	0.02704	15	**0.02243***	10	0.02970	0.03103	13	**0.02838***	5
E,1,2	0.00305	0.00131	10	**0.00087***	10	**0.00655***	**0.00655***	10	**0.00655***	1
E,3,4	**0.00227**	**0.00227**	1	**0.00227***	8/1	**0.00227***	**0.00227***	1	**0.00227***	1
E,5,6	0.00137	0.00045	6	**0.00000***	8	**0.00182**	0.00228	11	**0.00182***	13
E,7,8	0.00091	**0.00045***	3	**0.00045***	3	0.00091	**0.00045***	1	0.00091	1
E,9,10	0.01135	**0.00047***	12	0.00094	12	0.01230	**0.00993**	11	**0.00851***	6
O	0.15685	0.08536	8	**0.08243***	8	0.14827	**0.12613***	25/14	**0.12859**	25/1

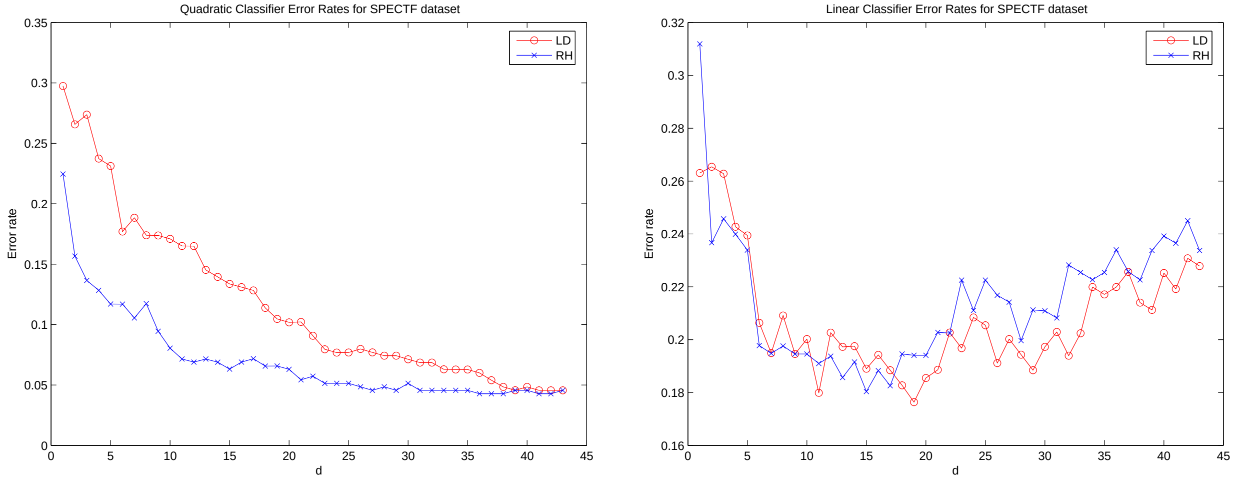
Table 3: Error rates for the two-class datasets drawn from the UCI repository.

lower than or equal to those of LD+Q; in 15 cases, RH+Q yields to the lowest error rate in dimensions lower than or equal to *five*.

This demonstrates that RH outperforms the other two techniques in three aspects: (i) it yields the lowest error rates in more cases than the others, (ii) it leads to error rates which are not significantly different from the optimal in most of the cases, and (iii) it gives better results than the other techniques while reducing to even lower dimensions than LD, and hence speeding up the classification stage.

To show the results from a different perspective, and to analyze the classifiers on different dimensions $d = 1, \ldots, n-1$, we plotted the error rate of the SPECTF dataset for all values of d, and for two methods, LD and RH. FD was excluded, since as pointed out earlier, the data can only be transformed to dimension one. The corresponding plots for the quadratic classifier and the linear classifier are depicted in Figure 2. For the quadratic classifier, the error rate (in general) decreases as the dimension d of the new space increases. Also, in this case, RH clearly leads to a lower error rate than LD, while both converge to similar error rates for values of d close to n. This reflects the fact that as the Chernoff distance in the transformed space increases, the error rate of the quadratic classifier decreases. Note that this behavior is more appropriate for the Bayesian (quadratic for normal distributions) classifier, and not for other classifiers, such as the linear one, as explained below. It would be an interesting problem to investigate the behavior of applying other classifiers to the result of the LDR, e.g. other nonlinear or kernel-like classifiers.

For the linear classifier, the behavior is different, in the sense that the the error rate starts decreasing to a certain point, to increase again after $d = 20$, while in most of the cases, RH leads to error rates comparable to those of LD. Note also that the linear classifier is taken by averaging the covariances, leading to an optimal classifier (in the Bayesian sense) only when the covariances are equal, situation that is not very common to occur in real-life data. This behavior is as expected, i.e. the error rate decreases quickly and stabilizes or even increases (for the linear classifier, in this case) as d becomes larger. This is advantageous for our case, as we can start reducing to dimensions 1,2, ..., until obtaining a reasonable error rate, the



(a) Quadratic classifier.

(b) Linear classifier.

Figure 2: Error rates for different dimensions on the SPECTF dataset.

error rate starts to increase, or the gain is not significant at all.

The plot of the Chernoff distance for different values of $d = 1, \ldots, n-1$, for RH and LD, and for the SPECTF dataset is depicted in Figure 3. It is clear that the Chernoff distance in the transformed space (y-axis), which is computed as in (34), increases as the dimension d of the transformed space increases, leading to RH producing higher Chernoff distances than LD. This, again, shows that since RH seeks for maximizing the Chernoff distance in the transformed space, it is more likely to lead to the lowest error rate, when using a quadratic classifier, in the transformed space. This corroborates the superiority of RH over LD and FD, as shown in Table 3. Additionally, the Chernoff distance for RH increases much quicker than that of LD in lower dimensions, which implies that the desired error rate (e.g. low enough) can be achieved in lower dimensions.

4.3 Multi-class Real-life Data

The datasets involved in the experiments, again, taken from the UCI Machine Learning Repository, are Iris plants, Pendigits, Thyrod gland, Wine, Glass identification, and Vowel context. In order to avoid ill-conditioned covariance matrices, we have applied principal component analysis (PCA) to Glass and reduced the data from dimension nine to eight, and

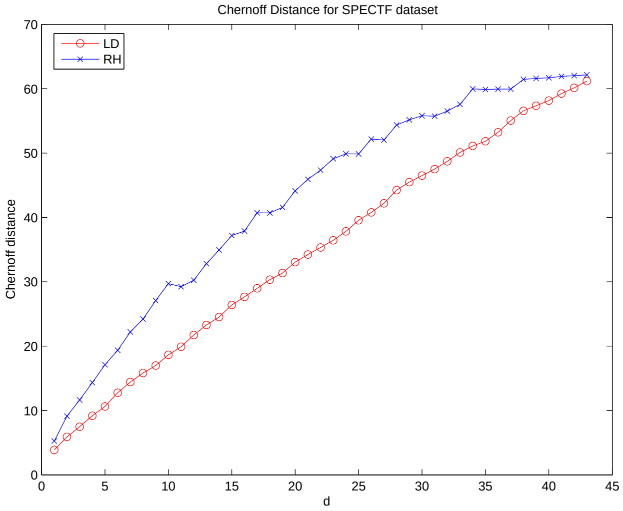


Figure 3: Chernoff distance for the SPECTF dataset.

removed class '6' to apply the 10-fold cross validation method. As in the two-class case, we trained the three LDR techniques, namely FD, LD and RH, followed by a quadratic or linear classifier, in a 10-fold cross-validation experiment. The average classification errors are given in Table 4, in which d^* indicates the dimension that yields the lowest error rate. For each classifier, quadratic and linear, the LDR method(s) that produce(s) the lowest error rate(s) is(are) marked with a '*', and the error rates which are not significantly different from the optimal are in bold. For the quadratic classifier, we note that the RH method yields lower error rate in four times, while FD and LD reach the best error rate in two times. For the linear classifier, both FD and LD lead to the lowest error rate four times each, while RH does it in three times. This is as expected, since RH aims to maximize the Chernoff distance in the transformed space, which is related to the Bayesian quadratic classifier, but not necessarily to the linear classifier. Also, the error rates obtained using RH and the quadratic classifier are in all cases (except in Iris and Glass) much smaller than the corresponding rate for the linear classifier, independently of the LDR technique coupled with the latter. For example, in Vowel, the error rate of RH+Q is more than 2% lower than that of LD+Q, and more than 7% lower than that of FD+Q. Regarding the significance test, RH coupled with the quadratic classifier leads to error rates not significantly different from the optimal in all cases, and for the linear classifier in all except one case. This demonstrates the effectiveness

Dataset	FD+Q	d^*	LD+Q	d^*	RH+Q	d^*	FD+L	d^*	LD+L	d^*	RH+L	d^*
Iris	0.02666	1	**0.02000***	1	**0.02000***	1	**0.02000***	1	**0.02000***	1	**0.02000***	1
Pendigit	0.04931	9	**0.02319**	15	**0.02228***	14	**0.12399***	9	**0.12909**	15	**0.13009**	15
Thyroid	**0.03290***	1	0.04220	1	**0.03744**	4	**0.09350***	1	**0.09350***	4	**0.09350***	1
Wine	0.01111	2	**0.00555***	2	**0.00555***	2	0.02225	2	**0.01111***	5	**0.01637**	2
Glass	**0.43330***	2	**0.44680**	4	**0.44957**	4	**0.33870***	4	0.39677	6	0.41062	6
Vowel	0.37878	9	0.32222	6	**0.30404***	6	0.46565	6	**0.44444***	2	**0.44444***	2

Table 4: Average error rates obtained from the three LDR techniques coupled with quadratic and linear classifiers on the multi-class datasets drawn from the UCI machine learning repository.

of the proposed LDR method in achieving the lowest error rates for multi-class datasets.

4.4 Protein Interaction Data

To analyze the performance of the LDR methods on a real-life application, we tested them on protein interaction prediction. Protein interactions are crucial in understanding cell processes and biological functions. The interaction prediction problem has been studied for quite a few years and the idea is to predict the type of complex or interaction sites. Our experiments centered on identifying protein complexes of two types, transient and obligate [11]. The dataset used includes 212 *transient* complexes and 115 *obligate* complexes [11]. The features for each complex represent the interaction energies: solvation and electrostatic. These features were calculated by following the approach and programs given in [2], which outputs the 20 residues in the two protein chains[3] that provide the maximum and minimum energy values contributing to the binding energy of the interaction. Energy values and residue numbers are provided for chains A, B, and AB. The residue numbers are not included in the results shown below, since they do not improve the classification accuracy at all (this was observed in the experiments performed). Also, some of the energy values are linear combinations or sums of other values. For this reason, we have compiled three different datasets that include (i) solvation and electrostatic energy values for chains A, B and AB,

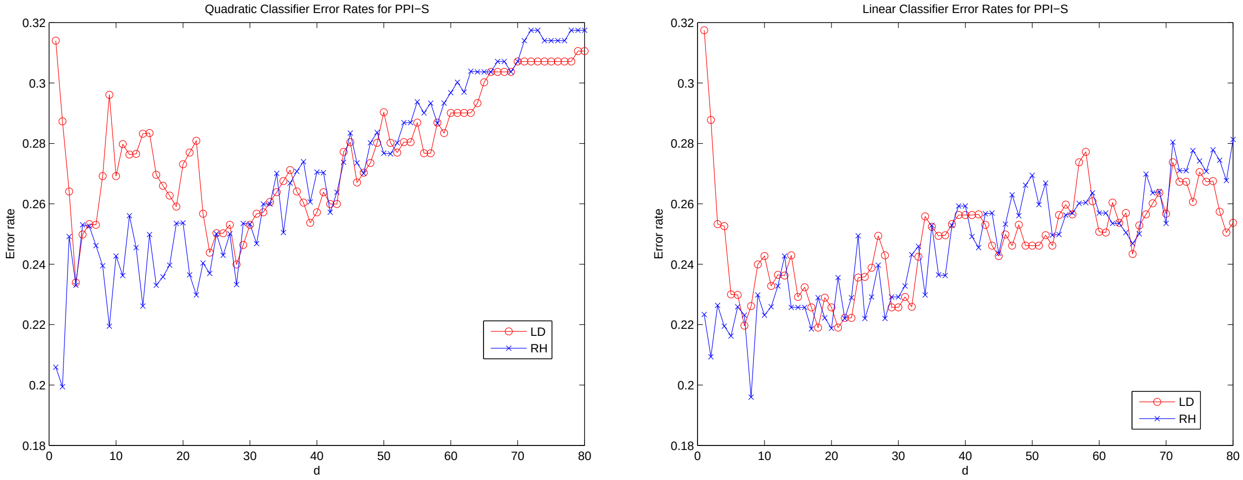
[3] Two chains per complex were taken. In case in which more than two chains were present they were merged into two chains and renamed A and B. For example, for complex 1l9j C:HLM, two chains were considered: A = C and B = HLM.

Dataset	n	LD+Q	d^*	RH+Q	d^*	LD+L	d^*	RH+L	d^*
PPI-SE	160	0.279755	56	**0.266185***	51	**0.215795**	19	**0.215573***	25
PPI-E	120	**0.320690**	68	**0.317464***	68	**0.239266***	30	**0.239266***	30
PPI-S	120	0.233927	4	**0.199444***	2	0.219021	21	**0.195996***	8

Table 5: Error Rates for two LDR methods, LD and RH, applied to the protein interaction dataset, where d^* indicates the lowest error rate for $d = 1, \ldots, n-1$.

(ii) electrostatic values for chains A and B, and (iii) solvation values for chains A and B. The datasets are named PPI-SE, PPI-E, and PPI-S, and contain 160, 120 and 120 features respectively. Two LDR methods, LD and RH, were trained, followed by a quadratic or linear classifier, in a 10-fold cross-validation experiment, as in Section 4.2. Note that results for the FD method are not included, since it yields an error rate of more than 30% for the three datasets. The classification errors and the best dimensions d^* for each LDR method and classifier are shown in Table 5. Again, in bold are the values that are not significantly different from the optimal. Considering solvation and electrostatic energies, i.e. dataset PPI-SE, RH yields a lower error rate than LD when combined with the quadratic classifier, and both yield similar error rates for the linear classifier. For electrostatic energies, i.e. dataset PPI-E, RH also leads to lower error rate than LD for the quadratic classifier, but the error rates for both methods are higher than those on PPI-SE. The best results were obtained on the PPI-S dataset, that is, when using solvation energies only. RH yielded lower error rates than LD for both quadratic and linear classifiers, where the difference for this case between the two LDR methods is above 3% and 2% respectively. It is also worth mentioning that the best error rates obtained by RH correspond to lower dimensions than those of the best errors yielded by LD.

To visually analyze the results on the PPI-S dataset (the best case of the three datasets), the error rates are plotted in Figure 4. The x-axis corresponds to the reduced dimension, while the y-axis represents the error rates for both LD and RH. The error rates start to decrease for lower dimensions, to stabilize at some point, and finally, increase with the dimension. A noteworthy point to observe is that RH leads to substantially lower error



(a) Quadratic classifier.

(b) Linear classifier.

Figure 4: Error rates for dimensions $d = 1, \ldots, 80$, obtained after applying LD and RH to the PPI-S dataset.

rates for lower dimensions, for the first 25 and 15 dimensions for the quadratic and linear classifiers. The average error differences between LD and RH were computed for the first 20 dimensions for both classifiers, resulting in 3.25% and 1.74% for the quadratic and linear classifiers respectively. For larger dimensions, the difference is not significant at all, but the errors are much higher than for lower dimensions. This demonstrates that RH leads to better dimensionality reduction in even lower dimensions, and hence reducing the complexity of the classification phase.

5 Conclusion

We have introduced a new criterion for linear dimensionality reduction (LDR), which, unlike previous approaches such as Fisher's and Loog-Duin's, aims to maximize the Chernoff distance in the transformed space. We have derived the corresponding criteria, and provided proofs for the convergence of the optimizing gradient-based algorithms. Additionally, we have shown that for any input parameters there always exists an orthogonal matrix that optimizes the proposed criterion. Based on this result, we have also provided and proved an angle-based initialization criterion for the secant method used as an intermediate step in the main algorithms.

We have tested the proposed LDR criterion, RH, on synthetic and real-life datasets from the UCI machine learning repository, and compared the results with other two LDR criteria, namely FD and LD, all of these coupled with both a quadratic and a linear classifier. The empirical results show the superiority of RH over the existing FD and LD criteria, mainly when the techniques are coupled with the *quadratic* classifier, demonstrating the importance of maximizing the Chernoff distance in the transformed space for such a classifier. We have also included a test on protein interaction classification, which shows that RH yields to lower error rates than LD.

One of the possible extensions for this work is to use a vector $\boldsymbol{\eta}$ to update the matrix $\mathbf{A}^{(k)}$, instead of a scalar. In this way, the direction of each row in $\mathbf{A}$ would change independently, and hence each of the rows would be "rotated" by using a different scalar η_i. We are also planning to investigate the use of other optimization techniques for our approach in order to avoid local optima. A second problem to investigate involves the use of a parameter β in optimizing $k(\beta, \mathbf{a})$, as opposed to the heuristic $\beta = p_1$. Finally, the application of the proposed LDR technique to face recognition is an interesting problem to investigate, as quite a few approaches have been proposed [9].

Acknowledgments: The authors would like to express their sincere gratitude to the Referees for their valuable comments and their dedication to improve the quality of this paper. This research has been supported by the Chilean National Council for Technological and Scientific Research, FONDECYT grant No. 1060904, and the Institute of Informatics, National University of San Juan, Argentina.

References

[1] S. Abe. *Support Vector Machines for Pattern Classification.* Springer, 2005.

[2] C. Camacho and C. Zhang. FastContact: Rapid Estimate of Contact and Binding Free Energies. *Bioinformatics*, 21(10):2534–2536, 2005.

[3] E. Chong and S. Zak. *An Introduction to Optimization*. John Wiley and Sons, Inc., New York, NY, 2nd edition, 2001.

[4] D. Harville. *Matriz Algebra from a Statisticians Perspective*. Springer-Verlag, New York, 1997.

[5] R. Duda, P. Hart, and D. Stork. *Pattern Classification*. John Wiley and Sons, Inc., New York, NY, 2nd edition, 2000.

[6] M. Herrera and R. Leiva. Generalización de la Distancia de Mahalanobis para el Análisis Discriminante Lineal en Poblaciones con Matrices de Covarianza Desiguales. *Revista de la Sociedad Argentina de Estadística*, 3(1-2):64–86, 1999.

[7] M. Loog and P.W. Duin. Linear Dimensionality Reduction via a Heteroscedastic Extension of LDA: The Chernoff Criterion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):732–739, 2004.

[8] J. Lu, K. Plataniotis, and A. Venetsanopoulos. Face Recognition Using Kernel Direct Discriminant Analysis Algorithms. *IEEE Trans. on Neural Networks*, 14(1):117–126, 2003.

[9] J. Lu, K. Plataniotis, A. Venetsanopoulos, and S. Li. Ensemble-based Discriminant Learning with Boosting for Face Recognition. *IEEE Trans. on Neural Networks*, 17(1):166–178, 2006.

[10] A. Martinez and M. Zhu. Where Are Linear Feature Extraction Methods Applicable? *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(12):1934–1944, 2005.

[11] J. Mintseris and Z. Weng. Structure, Function, and Evolution of Transient and Obligate Protein-protein Interactions. *Proceedings of the National Academy of Sciences, USA*, 102(31):10930–10935, 2005.

[12] D. Newman, S. Hettich, C. Blake, and C. Merz. UCI repository of machine learning databases, 1998. University of California, Irvine, Dept. of Computer Science.

[13] J. Rice. *Mathematical Statistics and Data Analysis.* Belmont: Duxbury Press, second edition, 1995.

[14] M. Rohl and C. Weihs. Optimal vs. classical linear dimension reduction. In *Information Age, Studies in Classification, Data Analysis, and Knowledge Organization*, pages 252–259. Springer, 1999.

[15] L. Rueda. Selecting the Best Hyperplane in the Framework of Optimal Pairwise Linear Classifiers. *Pattern Recognition Letters*, 25(2):49–62, 2004.

[16] L. Rueda and B. J. Oommen. On Optimal Pairwise Linear Classifiers for Normal Distributions: The Two-Dimensional Case. *IEEE Transations on Pattern Analysis and Machine Intelligence*, 24(2):274–280, 2002.

[17] D. Stinson. *Cyptography : Theory and Practice.* CRC Press, second edition, 2002.

[18] J. Yang, A. Frangi, J.Y. Yang, D. Zhang, and J. Zhong. KPCA Plus LDA: A Complete Kernel Fisher Discriminant Framework for Feature Extraction and Recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(2):230–244, 2005.

[19] J. Yang and J. Y. Yang. Why Can LDA Be Performed in PCA Transformed Space? *Pattern Recognition*, 36(2):563–566, 2003.

[20] J. Yang, D. Zhang, X. Yong, and J.Y. Yang. Two-dimensional Discriminant Transform for Face Recognition. *Pattern Recognition*, 38(7):1125–1129, 2005.

[21] H. Yu and J. Yang. A Direct LDA Algorithm for High-dimensional Data with Application to Face Recognition. *Pattern Recognition*, 34(10):2067–2070, 2001.

[22] M. Zhu and A. Martinez. Subclass Discriminant Analysis. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28(8):1262–1273, 2006.

Appendix A

A.1. Compactness of the Solution Set

Let $\mathbf{A}$ be any real $d \times n$ matrix, $d \leq n$, whose rows are linearly independent, and $J^*_{c_{12}}(\mathbf{A})$ be defined as in (34). Then, it follows that $max_{\{\mathbf{Q}:\mathbf{Q}\mathbf{Q}^t=\mathbf{I}_d\}} J^*_{c_{12}}(\mathbf{Q}) \leq max_{\{\mathbf{A}\}} J^*_{c_{12}}(\mathbf{A})$.

If we apply the QR decomposition to the matrix $\mathbf{A}^t$, which is full-row rank, we can ensure that there exist unique matrices $\mathbf{Q}_1$ of order $n \times d$ whose columns are orthogonal, and $\mathbf{R}_1$ of order $d \times d$ which is upper triangular with real positive elements in its diagonal, in such a way that[4] $\mathbf{A}^t = \mathbf{Q}_1\mathbf{R}_1$, or $\mathbf{A} = (\mathbf{Q}_1\mathbf{R}_1)^t = \mathbf{R}_1^t\mathbf{Q}_1^t = \mathbf{R}\mathbf{Q}$, where $\mathbf{R}$ is of order $d \times d$ and lower triangular, and $\mathbf{Q}$ is of order $d \times n$, such that $\mathbf{Q}\mathbf{Q}^t = \mathbf{I}_d$. Then, we have:

$$\begin{aligned}
J^*_{c_{12}}(\mathbf{A}) &= p_1p_2(\mathbf{R}\mathbf{Q}\mathbf{m}_1 - \mathbf{R}\mathbf{Q}\mathbf{m}_2)^t[\mathbf{R}\mathbf{Q}\mathbf{S}_W\mathbf{Q}^t\mathbf{R}^t]^{-1}(\mathbf{R}\mathbf{Q}\mathbf{m}_1 - \mathbf{R}\mathbf{Q}\mathbf{m}_2) \\
&\quad + \log\left(\frac{|\mathbf{R}\mathbf{Q}\mathbf{S}_W\mathbf{Q}^t\mathbf{R}^t|}{|\mathbf{R}\mathbf{Q}\mathbf{S}_1\mathbf{Q}^t\mathbf{R}^t|^{p_1}|\mathbf{R}\mathbf{Q}\mathbf{S}_2\mathbf{Q}^t\mathbf{R}^t|^{p_2}}\right) && (26) \\
&= p_1p_2\,[\mathbf{R}(\mathbf{Q}\mathbf{m}_1 - \mathbf{Q}\mathbf{m}_2)]^t\,[\mathbf{R}^t]^{-1}[\mathbf{Q}\mathbf{S}_W\mathbf{Q}^t]^{-1}\mathbf{R}^{-1}[\mathbf{R}(\mathbf{Q}\mathbf{m}_1 - \mathbf{Q}\mathbf{m}_2)] \\
&\quad + \log\left(\frac{|\mathbf{R}||\mathbf{Q}\mathbf{S}_W\mathbf{Q}^t||\mathbf{R}^t|}{[|\mathbf{R}||\mathbf{Q}\mathbf{S}_1\mathbf{Q}^t||\mathbf{R}^t|]^{p_1}[|\mathbf{R}||\mathbf{Q}\mathbf{S}_2\mathbf{Q}^t||\mathbf{R}^t|]^{p_2}}\right) && (27) \\
&= p_1p_2(\mathbf{Q}\mathbf{m}_1 - \mathbf{Q}\mathbf{m}_2)^t\mathbf{R}^t[\mathbf{R}^t]^{-1}[\mathbf{Q}\mathbf{S}_W\mathbf{Q}^t]^{-1}\mathbf{R}^{-1}\mathbf{R}(\mathbf{Q}\mathbf{m}_1 - \mathbf{Q}\mathbf{m}_2) \\
&\quad + \log\left(\frac{2|\mathbf{R}||\mathbf{Q}\mathbf{S}_W\mathbf{Q}^t|}{(2|\mathbf{R}|)^{p_1}|\mathbf{Q}\mathbf{S}_1\mathbf{Q}^t|^{p_1}(2|\mathbf{R}|)^{p_2}|\mathbf{Q}\mathbf{S}_2\mathbf{Q}^t|^{p_2}}\right) && (28) \\
&= p_1p_2(\mathbf{Q}\mathbf{m}_1 - \mathbf{Q}\mathbf{m}_2)^t[\mathbf{Q}\mathbf{S}_W\mathbf{Q}^t]^{-1}(\mathbf{Q}\mathbf{m}_1 - \mathbf{Q}\mathbf{m}_2) \\
&\quad + \log\left(\frac{|\mathbf{Q}\mathbf{S}_W\mathbf{Q}^t|}{|\mathbf{Q}\mathbf{S}_1\mathbf{Q}^t|^{p_1}|\mathbf{Q}\mathbf{S}_2\mathbf{Q}^t|^{p_2}}\right). && (29)
\end{aligned}$$

Since the determinant of a matrix and its transpose are the same, and $p_1 + p_2 = 1$, then $2|\mathbf{R}|$ and $(2|\mathbf{R}|)^{p_1}(2|\mathbf{R}|)^{p_2}$ cancel out, resulting in $J^*_c(\mathbf{A}) = J^*_c(\mathbf{Q})$, or equivalently:

$$max_{\{\mathbf{A}\}} J^*_{c_{12}}(\mathbf{A}) = max_{\{\mathbf{Q}:\mathbf{Q}\mathbf{Q}^t=\mathbf{I}_d\}} J^*_{c_{12}}(\mathbf{Q})\,. \qquad (30)$$

[4] The upper triangular matrix $\mathbf{R}_1$ is obtained from the coefficients of the iterative expressions of the Gram-Schmidt orthogonalization process [4].

A.2. Relationship to the LD Criterion

To see the relationship between J_{LD_2} and $J^*_{c_{12}}$, we, first, assume that $\mathbf{S}_1$ and $\mathbf{S}_2$ are diagonal, and that $\mathbf{A}$ is a $d \times n$ matrix with its d rows orthogonal to each other, i.e. $\mathbf{A}\mathbf{A}^t = \mathbf{I}_d$. Also, we assume that, pre and post-multiplying by $\mathbf{S}_W^{\frac{1}{2}}$, we have $\mathbf{S}_W = p_1\mathbf{S}_1 + p_2\mathbf{S}_2 = \mathbf{I}_n$. Then:

$$J_{LD_2}(\mathbf{A}) = tr\{p_1p_2\mathbf{A}\mathbf{S}_E\mathbf{A}^t - p_1\mathbf{A}\log(\mathbf{S}_1)\mathbf{A}^t - p_2\mathbf{A}\log(\mathbf{S}_2)\mathbf{A}^t\}\text{ , and} \tag{31}$$

$$J^*_{c_{12}}(\mathbf{A}) = tr\{p_1p_2\mathbf{A}\mathbf{S}_E\mathbf{A}^t - p_1\log(\mathbf{A}\mathbf{S}_1\mathbf{A}^t) - p_2\log(\mathbf{A}\mathbf{S}_2\mathbf{A}^t)\}\,. \tag{32}$$

Suppose now (losing generality, but aiming to analyze a particular case) that $p_1 = p_2 = 1/2$, and A is of order $1 \times d$. Then, we have:

$$J_{LD_2}(\mathbf{A}) = tr\{1/2\mathbf{A}\mathbf{S}_E\mathbf{A}^t - \mathbf{A}[\log(\mathbf{S}_1) + \log(\mathbf{S}_2)]\mathbf{A}^t\}\text{ , and} \tag{33}$$

$$J^*_{c_{12}}(\mathbf{A}) = tr\{1/2\mathbf{A}\mathbf{S}_E\mathbf{A}^t - [\log(\mathbf{A}\mathbf{S}_1\mathbf{A}^t) + \log(\mathbf{A}\mathbf{S}_2\mathbf{A}^t)]\}\,. \tag{34}$$

The eigenvalue decomposition of $\mathbf{A}\mathbf{S}_1\mathbf{A}^t$ is of order 1×1. If there was a matrix that satisfies $J^*_{c_{12}}(A) = J_{LD_2}(\mathbf{A})$, we would have:

$$tr[\log(\mathbf{A}\mathbf{S}_1\mathbf{A}^t) + \log(\mathbf{A}\mathbf{S}_2\mathbf{A}^t)] = tr[\mathbf{A}\log(\mathbf{S}_1)\mathbf{A}^t + \mathbf{A}\log(\mathbf{S}_2)\mathbf{A}^t]\,. \tag{35}$$

or equivalently,

$$tr[\log(\mathbf{A}\mathbf{S}_1\mathbf{A}^t)] + tr[\log(\mathbf{A}\mathbf{S}_2\mathbf{A}^t)] = tr[\mathbf{A}\log(\mathbf{S}_1)\mathbf{A}^t] + tr[\mathbf{A}\log(\mathbf{S}_2)\mathbf{A}^t]\,. \tag{36}$$

Since $\mathbf{A} = [a_1, \ldots, a_n]$ is of order $1 \times n$ and $\mathbf{S}_1 = diag(s_{11}, s_{12}, \ldots, s_{1n})$ is diagonal, we have $tr[\log(\mathbf{A}\mathbf{S}_1\mathbf{A}^t)] = \log(a_1^2 s_{11}) + \log(a_2^2 s_{12}) + \ldots + \log(a_n^2 s_{1n}) = \log\left(\sum_{i=1}^n a_i^2 s_{1i}\right)$. Also, $tr[\mathbf{A}\log(\mathbf{S}_1)\mathbf{A}^t] = \sum_{i=1}^n a_i^2 \log s_{1i}$.

Since $\mathbf{A}$ is orthogonal, it is true that $\sum_{i=1}^n a_i^2 = 1$. Additionally, logarithm is a concave "$\cap$" function in $(0,\infty)$. Then, Jensen's inequality [17] ensures that $\sum_{i=1}^{n} a_i^2 \log s_{1i} \leq \log\left(\sum_{i=1}^{n} a_i^2 s_{1i}\right)$, with equality if: (i) all s_{1i} are equal, or (ii) only one a_i is 1 and all a_j, $j \neq i$, are equal to zero.

We can do the same with $\mathbf{S}_2$, leading to $tr[\log(\mathbf{A}\mathbf{S}_1\mathbf{A}^t)+\log(\mathbf{A}\mathbf{S}_2\mathbf{A}^t)] \leq tr[\mathbf{A}\log(\mathbf{S}_1)\mathbf{A}^t + \mathbf{A}\log(\mathbf{S}_2)\mathbf{A}^t]$. Moreover, this is also true for $p_1 \neq p_2$, and hence $tr[p_1\log(\mathbf{A}\mathbf{S}_1\mathbf{A}^t)+p_2\log(\mathbf{A}\mathbf{S}_2\mathbf{A}^t)] \leq tr[p_1\mathbf{A}\log(\mathbf{S}_1)\mathbf{A}^t + p_2\mathbf{A}\log(\mathbf{S}_2)\mathbf{A}^t]$ and $J_{LD_2}(\mathbf{A}) \neq J_{c_{12}}^*(\mathbf{A})$.

We then conclude that both criteria are different, except for special cases: when the covariances are equal or when the transformation matrix is of the form $\mathbf{A} = [0,\ldots,0,1_i,0,\ldots,0]$. As this is very restrictive, in general both criteria lead to different solutions. Note also that this could also be generalized for $d > 1$ by an inductive argument on d, resulting in a matrix of the form $\mathbf{A} = [[0,\ldots,0,1_{i_1},0,\ldots,0]^t\,[0,\ldots,0,1_{i_2},0,\ldots,0]^t\,\ldots\,[0,\ldots,0,1_{i_d},0,\ldots,0]^t]^t$, where $i_j \neq i_k$, $j,k = 1,\ldots,d$.

To reinforce this hypothesis, we analyze the first order necessary conditions. Suppose that $\mathbf{S}_1$ and $\mathbf{S}_2$ are diagonal in such a way that $\mathbf{S}_W = p_1\mathbf{S}_1 + p_2\mathbf{S}_2 = \mathbf{I}_n$. Also, since the maximum resides in a compact set, for any matrix $\mathbf{A}$ of order $d \times n$ such that $\mathbf{A}\mathbf{A}^t = \mathbf{I}_d$ the gradients reduce to:

$$\nabla J_{LD_2}(\mathbf{A}) = 2p_1p_2[\mathbf{S}_E\mathbf{A}^t - \mathbf{A}^t(\mathbf{A}\mathbf{S}_E\mathbf{A}^t)]^t - 2\mathbf{A}(p_1\log\mathbf{S}_1 + p_2\log\mathbf{S}_2)\,, \text{and} \tag{37}$$

$$\nabla J_{c_{12}}^*(\mathbf{A}) = 2p_1p_2[\mathbf{S}_E\mathbf{A}^t - \mathbf{A}^t(\mathbf{A}\mathbf{S}_E\mathbf{A}^t)]^t + 2[\mathbf{A}^t - p_1\mathbf{S}_1\mathbf{A}^t(\mathbf{A}\mathbf{S}_1\mathbf{A}^t)^{-1} - p_2\mathbf{S}_2\mathbf{A}^t(\mathbf{A}\mathbf{S}_2\mathbf{A}^t)^{-1}]^t\,. \tag{38}$$

The solution matrix $\mathbf{A}$ for LD reduces to the d eigenvectors of S_{LD_2} that satisfy the condition $\mathbf{A}\mathbf{A}^t = \mathbf{I}_d$. Also, $\nabla J_{LD_2}(\mathbf{A}) = 0$, then from the above expressions, we have that $2p_1p_2[\mathbf{S}_E\mathbf{A}^t - \mathbf{A}^t(\mathbf{A}\mathbf{S}_E\mathbf{A}^t)]^t - 2\mathbf{A}(p_1\log\mathbf{S}_1 + p_2\log\mathbf{S}_2) = 0$, and $\mathbf{A}\mathbf{S}_E - (\mathbf{A}\mathbf{S}_E\mathbf{A}^t)\mathbf{A} = 2\mathbf{A}(\log\mathbf{S}_1 + \log\mathbf{S}_2)$. If this was the solution matrix for the RH criterion, then it would satisfy $\nabla J_{c_{12}}^*(\mathbf{A}) = 0$, and hence $2p_1p_2[\mathbf{S}_E\mathbf{A}^t - \mathbf{A}^t(\mathbf{A}\mathbf{S}_E\mathbf{A}^t)]^t + 2[\mathbf{A}^t - p_1\mathbf{S}_1\mathbf{A}^t(\mathbf{A}\mathbf{S}_1A^t)^{-1} -$

$p_2\mathbf{S}_2\mathbf{A}^t(\mathbf{A}\mathbf{S}_2\mathbf{A}^t)^{-1}]^t = 0$. Then, it is true that $\frac{1}{2}[\mathbf{S}_E\mathbf{A}^t - \mathbf{A}^t(\mathbf{A}\mathbf{S}_E\mathbf{A}^t)]^t + 2[\mathbf{A}^t - \frac{1}{2}(\mathbf{S}_1\mathbf{A}^t(\mathbf{A}\mathbf{S}_1\mathbf{A}^t)^{-1} + \mathbf{S}_2\mathbf{A}^t(\mathbf{A}\mathbf{S}_2\mathbf{A}^t)^{-1})]^t = 0$, which implies that $\mathbf{A}\mathbf{S}_E - (\mathbf{A}\mathbf{S}_E\mathbf{A}^t)\mathbf{A} = 2[(\mathbf{A}\mathbf{S}_1\mathbf{A}^t)^{-1}\mathbf{A}\mathbf{S}_1 + (\mathbf{A}\mathbf{S}_2\mathbf{A}^t)^{-1}\mathbf{A}\mathbf{S}_2] - 4\mathbf{A}$.

Taking both expressions in an equality, we have that $2\mathbf{A}(\log\mathbf{S}_1 + \log\mathbf{S}_2) = 2[(\mathbf{A}\mathbf{S}_1\mathbf{A}^t)^{-1}\mathbf{A}\mathbf{S}_1 + (\mathbf{A}\mathbf{S}_2\mathbf{A}^t)^{-1}\mathbf{A}\mathbf{S}_2] - 4\mathbf{A}$. This implies that $\mathbf{A}(\log\mathbf{S}_1 + \log\mathbf{S}_2) = (\mathbf{A}\mathbf{S}_1\mathbf{A}^t)^{-1}\mathbf{A}\mathbf{S}_1 + (\mathbf{A}\mathbf{S}_2\mathbf{A}^t)^{-1}\mathbf{A}\mathbf{S}_2 - 2\mathbf{A}$, which is not necessarily true, except in exceptional cases.

A.3. A Direct Solution for η

As discussed previously, we emphasize that it is quite important to efficiently obtain the value of η that maximizes the function $\phi_{k_{12}}(\eta)$ given in (17). Thus, finding a direct solution for the first-order necessary condition for this function would be the best option; however, we now show that this seems not to be possible. We know that the first derivative of the corresponding expression with respect to η results in the following expression:

$$\frac{d\phi_{k_{12}}}{d\eta}(\eta) = [\nabla J^*_{c_{12}}(\mathbf{A}^{(k)} + \eta\nabla J^*_{c_{12}}(\mathbf{A}^{(k)}))] \cdot \nabla J^*_{c_{12}}(\mathbf{A}^{(k)}) = \mathbf{0}\,. \tag{39}$$

Now, taking Equation (16), and doing some simple algebraic manipulations, we obtain:

$$\begin{aligned}\nabla J^*_{c_{12}}(\mathbf{A}) = \frac{\partial J^*_{c_{12}}}{\partial \mathbf{A}} &= p_1 p_2\left[\mathbf{S}_E\mathbf{A}^t - \mathbf{S}_W\mathbf{A}^t(\mathbf{A}\mathbf{S}_W\mathbf{A}^t)^{-1}(\mathbf{A}\mathbf{S}_E\mathbf{A}^t)\right]^t \\ &\quad + \left[\mathbf{S}_W\mathbf{A}^t - p_1\mathbf{S}_1\mathbf{A}^t(\mathbf{A}\mathbf{S}_1\mathbf{A}^t)^{-1}(\mathbf{A}\mathbf{S}_W\mathbf{A}^t) - p_2\mathbf{S}_2\mathbf{A}^t(\mathbf{A}\mathbf{S}_2\mathbf{A}^t)^{-1}(\mathbf{A}\mathbf{S}_W\mathbf{A}^t)\right]^t \\ &= \mathbf{0}\,. \end{aligned} \tag{40}$$

Substituting $\mathbf{A}$ for $\mathbf{A} + \eta\mathbf{G}$, where $\mathbf{G} = \nabla J^*_c(\mathbf{A}^{(k)})$ in Equation (40), and using the resulting expression in (39), we obtain the following formula:

$$
\begin{aligned}
&\Big\{p_1p_2\{\mathbf{S}_E\mathbf{A}^t + \mathbf{S}_E\eta\mathbf{G}^t - \mathbf{S}_W\mathbf{A}^t\left[(\mathbf{A}+\eta\mathbf{G})\mathbf{S}_W(\mathbf{A}+\eta\mathbf{G})^t\right]^{-1}\left[(\mathbf{A}+\eta\mathbf{G})\mathbf{S}_E(\mathbf{A}+\eta\mathbf{G})^t\right] \\
&-\eta\mathbf{S}_W\mathbf{G}^t\left[(\mathbf{A}+\eta\mathbf{G})\mathbf{S}_W(\mathbf{A}+\eta\mathbf{G})^t\right]^{-1}\left[(\mathbf{A}+\eta\mathbf{G})\mathbf{S}_E(\mathbf{A}+\eta\mathbf{G})^t\right]\} \\
&+\mathbf{S}_W\mathbf{A}^t + \eta\mathbf{S}_W\mathbf{G}^t \\
&-p_1\mathbf{S}_1\mathbf{A}^t\left[(\mathbf{A}+\eta\mathbf{G})\mathbf{S}_1(\mathbf{A}+\eta\mathbf{G})^t\right]^{-1}\left[(\mathbf{A}+\eta\mathbf{G})\mathbf{S}_W(\mathbf{A}+\eta\mathbf{G})^t\right] \\
&-p_1\eta\mathbf{S}_1\mathbf{G}^t\left[(\mathbf{A}+\eta\mathbf{G})\mathbf{S}_1(\mathbf{A}+\eta\mathbf{G})^t\right]^{-1}\left[(\mathbf{A}+\eta\mathbf{G})\mathbf{S}_W(\mathbf{A}+\eta\mathbf{G})^t\right] \\
&-p_2\mathbf{S}_2\mathbf{A}^t\left[(\mathbf{A}+\eta\mathbf{G})\mathbf{S}_2(\mathbf{A}+\eta\mathbf{G})^t\right]^{-1}\left[(\mathbf{A}+\eta\mathbf{G})\mathbf{S}_W(\mathbf{A}+\eta\mathbf{G})^t\right] \\
&-p_2\eta\mathbf{S}_2\mathbf{G}^t\left[(\mathbf{A}+\eta\mathbf{G})\mathbf{S}_2(\mathbf{A}+\eta\mathbf{G})^t\right]^{-1}\left[(\mathbf{A}+\eta\mathbf{G})\mathbf{S}_W(\mathbf{A}+\eta\mathbf{G})^t\right]\Big\}^t \cdot \mathbf{G} = \mathbf{0}\,. \qquad (41)
\end{aligned}
$$

In order to obtain the value of η that satisfies the first order necessary condition we would have to isolate η in Equation (41), which as can be seen is quite intricate. Say, we have quadratic equations in η, and in some cases the inverse of these. This demonstrates that, at least in most of the cases, a unique solution does not exist. Also, we do not even know if the inverse of $[(\mathbf{A}+\eta\mathbf{G})\mathbf{S}_W(\mathbf{A}+\eta\mathbf{G})]$ exists. To summarize, we observe that obtaining a direct solution for η seems not to be possible, justifying the iterative solution based on the secant method as proposed in Section 2.

A.4. Convergence

Theorem 1. *Let $\{\mathbf{A}^{(k)}\}_{k=1}^{\infty}$ be the sequence of matrices generated by Algorithm* **Chernoff_LDA_Two**. *If $\nabla J^*_{c_{12}}(\mathbf{A}^{(k)}) \neq 0$, then $J^*_{c_{12}}(\mathbf{A}^{(k)}) < J^*_{c_{12}}(\mathbf{A}^{(k+1)})$. Otherwise, the algorithm terminates.*

Proof. Consider the function $\phi_{k_{12}}(\eta)$ as defined in (17). We have that for any $\eta > 0$, $\phi_{k_{12}}(\eta) \geq \phi_{k_{12}}(\eta_k)$ holds.

We observe now that $\phi_{k_{12}}$ is the following decomposition of functions:

$$\eta \to \mathbf{A}^{(k)} + \eta\nabla J^*_{c_{12}}(\mathbf{A}^{(k)}) \to J^*_{c_{12}}(\mathbf{A}^{(k)} + \eta\nabla J^*_{c_{12}}(\mathbf{A}^{(k)}))\,, \qquad (42)$$

and that there exists an isomorphism between the matrix space of order $d \times n$, $d \leq n$, with its inner product $\mathbf{B} \cdot \mathbf{C} = tr\{\mathbf{B} \cdot \mathbf{C}^t\}$, and the vector space of dimension dn with its usual inner product.

Let us compute this derivative:

$$\frac{d\phi_{k_{12}}}{d\eta}(\eta) = [\nabla J^*_{c_{12}}(\mathbf{A}^{(k)} + \eta \nabla J^*_{c_{12}}(\mathbf{A}^{(k)}))] \cdot \nabla J^*_{c_{12}}(\mathbf{A}^{(k)})\,, \tag{43}$$

obtaining, for $\eta = 0$, the following expression:

$$\frac{d\phi_{k_{12}}}{d\eta}(0) = [\nabla J^*_{c_{12}}(\mathbf{A}^{(k)} + 0\nabla J^*_{c_{12}}(\mathbf{A}^{(k)}))] \cdot \nabla J^*_{c_{12}}(\mathbf{A}^{(k)}) = \parallel \nabla J^*_{c_{12}}(\mathbf{A}^{(k)}) \parallel^2_F > 0\,, \tag{44}$$

where $\parallel \mathbf{B} \parallel^2_F$ is the inner product $\mathbf{B} \cdot \mathbf{B} = tr\{\mathbf{B} \cdot \mathbf{B}^t\}$, also known as the Frobenius norm [3], which always results in a nonnegative value.

If $\nabla J^*_{c_{12}}(\mathbf{A}^k) \neq 0$, $\frac{d\phi_{k_{12}}}{d\eta}(0) > 0$, then, there exists an environment near 0 in which the function $\phi_{k_{12}}$ is monotonically increasing. Thus, we can make sure that there exists $\overline{\eta} > 0$ such that for all $\eta \in (0, \overline{\eta}]$, we have $\phi_{k_{12}}(0) < \phi_{k_{12}}(\eta)$. Using the latter inequality and Equation (17), we obtain the following equality:

$$J^*_{c_{12}}(\mathbf{A}^{(k)}) = \phi_{k_{12}}(0) < \phi_{k_{12}}(\overline{\eta}) \leq \phi_{k_{12}}(\eta_k) = J^*_{c_{12}}(\mathbf{A}^{(k+1)})\,. \tag{45}$$

Note that if $\nabla J^*_{c_{12}}(\mathbf{A}^{(k)}) = 0$, we have that $\mathbf{A}^{(k)} = \mathbf{A}^{(k+1)}$, and hence if $\tau > 0$, $|J^*_{c_{12}}(\mathbf{A}^{(k+1)}) - J^*_{c_{12}}(\mathbf{A}^{(k)})| < \tau$, the algorithm terminates. □

A.5. Initialization

We already know that $\mathbf{A}^{(k)}$ is an orthogonal matrix of order $d \times n$. By virtue of (29), the algorithm allows to ensure that $[\mathbf{A}^{(k)}][\mathbf{A}^{(k)}]^t = [\mathbf{A}^{(k+1)}][\mathbf{A}^{(k+1)}]^t = \mathbf{I}_d$. Thus, we have that $\left\|\mathbf{A}^{(k)}\right\|_F = \left\|\mathbf{A}^{(k+1)}\right\|_F = d$. This indicates that both matrices are located near the environment of zero (null matrices) of radius d in the matrix space. Therefore, we have that the magnitude is preserved and only the direction of $\mathbf{A}$ is changed, which is measured by the angle[5] between $\mathbf{A}^{(k)}$ and $\mathbf{A}^{(k+1)}$. Note that $J^*_{c_{12}}(\mathbf{A}) = J^*_{c_{12}}(-\mathbf{A})$, and so it follows that a maximum of $J^*_{c_{12}}(\mathbf{A})$ resides in half of the environment of radius d. We arbitrarily choose the angle difference between $\mathbf{A}^{(k)}$ and $[\mathbf{A}^{(k)} + \eta_1 \nabla J^*_{c_{12}}(\mathbf{A}^{(k)})]$. Let θ be the angle between $\mathbf{A}^{(k)}$ and $[\mathbf{A}^{(k)} + \eta_k \nabla J^*_{c_{12}}(\mathbf{A}^{(k)})]$. Then, we have that:

$$\begin{aligned} \cos\theta &= \frac{tr\{[\mathbf{A}^{(k)}][(\mathbf{A}^{(k)} + \eta_k \nabla J^*_{c_{12}}(\mathbf{A}^{(k)}))]^t\}}{\|\mathbf{A}^{(k)}\|_F \left\|\mathbf{A}^{(k)} + \eta_k \nabla J^*_{c_{12}}(\mathbf{A}^{(k)})\right\|_F} && (46) \\ &= \frac{d + \eta_k tr\{[\mathbf{A}^{(k)}][\nabla J^*_{c_{12}}(\mathbf{A}^{(k)})]^t\}}{d\left\|\mathbf{A}^{(k)} + \eta_k \nabla J^*_{c_{12}}(\mathbf{A}^{(k)})\right\|_F} && (47) \end{aligned}$$

Since $\left\|\mathbf{A}^{(k)}\right\|_F = d$ and $\left\|\mathbf{A}^{(k+1)})\right\|_F = \left\|\mathbf{A}^{(k)} + \eta_k \nabla J^*_{c_{12}}(\mathbf{A}^{(k)}))\right\|_F = d$, we can write (47) in the following manner:

$$\cos\theta = \frac{1}{d} + \frac{\eta_k tr\{[\mathbf{A}^{(k)}][\nabla J^*_{c_{12}}(\mathbf{A}^{(k)})]^t\}}{d^2} \qquad (48)$$

Also, we know that $\cos\theta \leq 1$, and hence we choose a value of $\hat{\theta} \to 0$. This implies that $\cos\hat{\theta} \to 1$, as $\cos\hat{\theta} = \epsilon$, then (48) leads to (19) .

[5] As in [4, pp. 60-61], the angle between two nonnull matrices $\mathbf{A}$ and $\mathbf{B}$ of order $d \times n$ is defined as $\cos\theta = \frac{\mathbf{A}\cdot\mathbf{B}}{\|A\|\|B\|}$, where $\mathbf{A}\cdot\mathbf{B} = tr\{\mathbf{A}\mathbf{B}^t\}$.