Roweis Discriminant Analysis: A Generalized Subspace Learning Method

Benyamin Ghojogh, Fakhri Karray, Mark Crowley

Abstract—We present a new method which generalizes subspace learning based on eigenvalue and generalized eigenvalue problems. This method, Roweis Discriminant Analysis (RDA), is named after Sam Roweis to whom the field of subspace learning owes significantly. RDA is a family of infinite number of algorithms where Principal Component Analysis (PCA), Supervised PCA (SPCA), and Fisher Discriminant Analysis (FDA) are special cases. One of the extreme special cases, which we name Double Supervised Discriminant Analysis (DSDA), uses the labels twice; it is novel and has not appeared elsewhere. We propose a dual for RDA for some special cases. We also propose kernel RDA, generalizing kernel PCA, kernel SPCA, and kernel FDA, using both dual RDA and representation theory. Our theoretical analysis explains previously known facts such as why SPCA can use regression but FDA cannot, why PCA and SPCA have duals but FDA does not, why kernel PCA and kernel SPCA use kernel trick but kernel FDA does not, and why PCA is the best linear method for reconstruction. Roweisfaces and kernel Roweisfaces are also proposed generalizing eigenfaces, Fisherfaces, supervised eigenfaces, and their kernel variants. We also report experiments showing the effectiveness of RDA and kernel RDA on some benchmark datasets.

Index Terms—Roweis Discriminant Analysis (RDA), Generalized subspace learning, Principal Component Analysis (PCA), Fisher Discriminant Analysis (FDA), Supervised Principal Component Analysis (SPCA), Double Supervised Discriminant Analysis (DSDA).

I. INTRODUCTION

SUBSPACE and manifold learning, also referred to as representation learning [1], [2], are very useful in machine learning and pattern analysis for feature extraction, data visualization, and dimensionality reduction [3]. The subspace of data is a lower-dimensional space than the input space of data which can appropriately represent the data with the smallest possible representation error. In other words, the data usually exist on a submanifold or subspace with a lower intrinsic dimensionality [4], [5].

The submanifold of data can be either linear or nonlinear. Different linear and nonlinear methods have been proposed in the literature for subspace and manifold learning. Principal Component Analysis (PCA) [6], first proposed in [7], was one of the first methods in linear subspace learning. The Fisher Discriminant Analysis (FDA) [8], first proposed in [9], was one of the first linear supervised subspace learning methods. Both PCA and FDA were based on the scatter of data. Metric Multi-Dimensional Scaling (MDS) [10] was a linear method which tried to preserve the similarity of the data points. In later approaches after MDS, the cost function in MDS was changed to preserving the distances of points [11], yielding to Sammon mapping [12]. Sammon mapping can probably be considered as the first nonlinear subspace learning method. Another approach to handle nonlinearity of data is to modify the data instead of changing the linear algorithm. That was the perspective of kernel PCA [13], [14] which used the dual of PCA and the kernel trick [15] to pull the data to the feature space hoping that it becomes roughly linear in that space. Kernel FDA [16], [17] was also proposed for handling the nonlinear data in a supervised manner. However, it did not use the kernel trick (which we will explain why in this paper) but used representation theory [18] instead. Recently, deep FDA [19], [20] was proposed which uses a least squares approach [21], [22].

After Sammon mapping, there was not any important method, which is actually nonlinear without changing the data like kernel PCA/FDA, until Isomap [23] and Locally Linear Embedding (LLE) [24]. The former used geodesic distance rather than Euclidean distance in the kernel of MDS [25], [26] and the latter reconstructed every point by its k-Nearest Neighbors (k-NN) to locally fit the data [27]. Stochastic Neighbor Embedding (SNE) [28] was a probabilistic approach to manifold learning where the probability of a point being neighbor of others was tried to be preserved. While the Gaussian distribution was used in SNE, the Student-t distribution was used in the embedded space in t-SNE method in order to tackle the crowding problem [29]. Gradually, deep learning became popular where the Deep Belief Network (DBN) [30] was proposed in order to learn the latent subspace of data in an undercomplete autoencoder. One of the most recent important subspace learning methods was Supervised PCA (SPCA) [31]. The SPCA made use of the empirical estimation of the Hilbert-Schmidt Independence Criterion (HSIC) [32]. The HSIC computes the dependence of two random variables by calculating their correlation in the feature space. The SPCA tried to maximize the dependence of projected data and the labels in order to use the information of labels for better embedding. Recently, supervised random projection [33] used low-rank kernel approximation in the formulation of SPCA for better efficiency.

In this paper, we propose a generalized subspace learning method, named Roweis Discriminant Analysis (RDA), named after Sam Roweis (1972–2010) who contributed significantly to subspace and manifold learning. He proposed many important methods in subspace/manifold learning including LLE [24], [27], SNE [28], and metric learning by class collapsing [34]. The proposed RDA is a family of infinite number of subspace learning methods including PCA, FDA, and
SPCA which have been proposed in the literature. It is a
generalized method and contains many methods which use
linear projection into a lower dimensional subspace. The main
contributions of our paper can be listed as the following:

1) proposing RDA as a generalized subspace learning
   method based on eigenvalue and generalized eigenvalue
   problems,
2) proposing Double Supervised Discriminant Analysis
   (DSDA), which is not yet proposed in the literature to
   the best of our knowledge, as one of the extreme cases
   in RDA,
3) proposing RDA in the feature space to have kernel RDA
   which generalizes kernel PCA, kernel SPCA, and kernel
   FDA,
4) explaining the reasons behind some of the characteristics
   of PCA, SPCA, and FDA such as (I) why SPCA can be
   used for both classification and regression but FDA is
   only for classification, (II) why PCA and SPCA have
dual methods but FDA does not have a dual, (III) why kernel
PCA and kernel SPCA can use kernel trick but kernel FDA
uses representation theory rather than kernel trick, (IV) why PCA
is the best linear method
for reconstruction,
5) proposing Roweisfaces and kernel Roweisfaces for
   demonstrating the generalizability of the RDA approach
   for the eigenfaces [35], [36], kernel eigenfaces [37],
   Fisherfaces [38], kernel Fisherfaces [39] and supervised
   eigenfaces [31], [40].

The remainder of the paper is organized as follows. Section
II introduces projection into a subspace and the reconstruction
of data after projection. Two general forms of optimization
for subspace learning are also introduced. In Section III, the
theory of PCA, FDA, and SPCA is briefly reviewed. The
proposed RDA is explained in Section IV. A dual is also
proposed for the RDA in Section V but for some special
cases. Section VI explains two types of kernel RDA, one based
on kernel trick and another using representation theory. The
experimental results are reported in Section VII where the
Roweisfaces are also proposed. Section VIII summarizes and
concludes the article.

II. SUBSPACE AND PROJECTION

Let \( n, n_d, d, c, n_j, x_i, x_i(j), \) and \( x_{i,j} \) denote the training
sample size, test (out-of-sample) sample size, dimensionality
of data, number of classes, sample size of the j-th class, the
i-th training instance, the \( i \)-th training instance in the j-th class,
and the \( i \)-th test instance, respectively. We stack the \( n \) training
data points column-wise in a matrix \( X = [x_1, \ldots, x_n] \in \mathbb{R}^{d \times n} \) and similarly for the \( n_d \) out-of-sample data points as
\( \tilde{X} = [\tilde{x}_1, \ldots, \tilde{x}_n] \in \mathbb{R}^{d \times n}. \)

Let \( U \in \mathbb{R}^{d \times d} \) be the projection matrix whose columns
\( \{u_j\}_{j=1}^d \) are the projection directions spanning the desired
subspace so the subspace is the column-space of \( U \). If we trunc-
ate the projection matrix to have \( \mathbb{R}^{d \times p} \ni U = [u_1, \ldots, u_p], \) the
subspace is spanned by \( p \) projection directions and it will be
a \( p \) dimensional subspace where \( p \leq d \). We want the
projection directions to be orthonormal to capture different
information; therefore:

\[
U^\top U = I,
\]

where \( I \) is the identity matrix.

The projection of training data into the subspace and its
reconstruction after the projection are [4]:

\[
\mathbb{R}^{p \times n} \ni \bar{X} := U^\top X, \tag{2}
\]

\[
\mathbb{R}^{d \times n} \ni \hat{X} := UU^\top X = UX, \tag{3}
\]

respectively, where \( \bar{X} = [\bar{x}_1, \ldots, \bar{x}_n] \) and \( \hat{X} = [\hat{x}_1, \ldots, \hat{x}_n] \).

The projection and reconstruction of the out-of-sample data are:

\[
\mathbb{R}^{p \times n_d} \ni \bar{X}_t := U^\top X_t, \tag{4}
\]

\[
\mathbb{R}^{d \times n_d} \ni \hat{X}_t := UU^\top X_t = U\tilde{X}_t, \tag{5}
\]

respectively, where \( \bar{X}_t = [\bar{x}_t, \ldots, \bar{x}_{t,n}] \) and \( \hat{X}_t = [\hat{x}_t, \ldots, \hat{x}_{t,n}] \). Note that in subspace learning, it is recom-
meded to center the data, either training or out-of-sample,
using the training mean. If we do that, the training mean should
be added back to the reconstructed data. If the mean of training
data is:

\[
\mu := \frac{1}{n} \sum_{i=1}^n x_i, \tag{6}
\]

the centered training data are:

\[
\mathbb{R}^{d \times n} \ni \tilde{X} := XH = X - \mu, \tag{7}
\]

where \( \tilde{X} = [\tilde{x}_1, \ldots, \tilde{x}_n] = [x_1 - \mu, \ldots, x_n - \mu] \) and:

\[
\mathbb{R}^{n \times n} \ni H := I - (1/n)11^\top, \tag{8}
\]

is the centering matrix. The covariance matrix, or the total
scatter, is defined as:

\[
\mathbb{R}^{n \times n} \ni S_T := \sum_{i=1}^n (x_i - \mu)(x_i - \mu)^\top \tag{9}
\]

\[
= \tilde{X}\tilde{X}^\top \overset{(7)}{=} XHHX^\top = XHX^\top,
\]

where it is noticed that the centering matrix is symmetric and
idempotent.

If we project the centered training data, we have:

\[
||\tilde{X}||_F^2 = ||UU^\top \tilde{X}||_F^2 = \text{tr}(U^\top S_T U), \tag{10}
\]

where \( ||.||_F \) and \( \text{tr}(.) \) denote the Frobenius norm and trace
of matrix, respectively. Hence, \( \text{tr}(U^\top S U) \) is the squared
length of the reconstruction. This term can also be interpreted
as the variance of the projected data according to quadratic
characteristic of variance. We want the variance of projection
to be maximized where the projection matrix is orthogonal;
otherwise, the optimization problem is ill-defined. If we con-
sider a general scatter of data, \( S \in \mathbb{R}^{d \times d} \), the optimization
is:

\[
\text{maximize} \quad \text{tr}(U^\top S U), \tag{11}
\]

subject to \( U^\top U = I, \)
where the constraint makes the problem well-defined. The Lagrangian [41] of the problem is:
\[
\mathcal{L} = \text{tr}(U^T S U) - \text{tr}(\Lambda^T (U^T U - I)),
\]
where \( \Lambda \in \mathbb{R}^{p \times p} \) is a diagonal matrix \( \text{diag}([\lambda_1, \ldots, \lambda_p]^T) \) including the Lagrange multipliers. Setting the derivative of Lagrangian to zero gives:
\[
\frac{\partial \mathcal{L}}{\partial U} = 2SU - 2UA \xrightarrow{\text{set}} 0 \implies SU = U\Lambda,
\]
which is the eigenvalue problem for \( S \) where the columns of \( U \) and the diagonal of \( \Lambda \) are the eigenvectors and eigenvalues of \( S \), respectively [42]. The eigenvectors and eigenvalues are sorted from the leading (largest eigenvalue) to the trailing (smallest eigenvalue) because we are maximizing in the optimization problem (the reason lies in the second order condition).

We can also have two types of scatters, e.g., \( S_1 \in \mathbb{R}^{n \times n} \) and \( S_2 \in \mathbb{R}^{n \times n} \). As the scatter matrix is symmetric and positive semi-definite, we can decompose it as:
\[
S_2 = \Psi_S \Omega_S \Psi_S^\top = \Psi_S \Omega_S^{(1/2)} \Omega_S^{-(1/2)} \Psi_S^\top = \Delta^\top \Delta,
\]
where \((a)\) is because of Singular Value Decomposition (SVD) and \((b)\) is for \( \mathbb{R}^{n \times n} \ni \Delta := \Omega_S^{(1/2)} \Psi_S^\top \Delta \). The \( \Delta U \) can be interpreted as manipulated (or rotated) projection directions. We want the manipulated projection matrix to be orthogonal; thus:
\[
(\Delta U)^\top (\Delta U) = U^T S_2 U \xrightarrow{\text{set}} I.
\]
In this case, the optimization is expressed as:
\[
\begin{align*}
\text{maximize} & \quad \text{tr}(U^T S_1 U), \\
\text{subject to} & \quad U^T S_2 U = I,
\end{align*}
\]
and the constraint makes the problem well-defined. The Lagrangian [41] of the problem is:
\[
\mathcal{L} = \text{tr}(U^T S_1 U) - \text{tr}(\Lambda^T (U^T S_2 U - I)),
\]
where \( \Lambda \) is a diagonal matrix which includes the Lagrange multipliers. Setting the derivative of Lagrangian to zero gives:
\[
\frac{\partial \mathcal{L}}{\partial U} = 2S_1 U - 2S_2 U \Lambda \xrightarrow{\text{set}} 0 \implies S_1 U = S_2 U\Lambda,
\]
which is the generalized eigenvalue problem \( (S_1, S_2) \) where the columns of \( U \) and the diagonal of \( \Lambda \) are the eigenvectors and eigenvalues, respectively [42]. The eigenvectors and eigenvalues are again sorted from the leading to the trailing because of maximization.

III. PCA, FDA, AND SPCA

The optimization problem in PCA [7], [6], [40] is expressed as:
\[
\begin{align*}
\text{maximize} & \quad \text{tr}(U^T S_T U), \\
\text{subject to} & \quad U^T U = I,
\end{align*}
\]
where \( S_T \in \mathbb{R}^{n \times n} \) is the total scatter defined in Eq. (9). The solution to Eq. (17) is the eigenvalue problem for \( S_T \) according to Eq. (12). Thus, the PCA directions are the eigenvectors of the total scatter.

The FDA [9], [8] maximizes the Fisher criterion [43], [44]:
\[
\begin{align*}
\text{maximize} & \quad \text{tr}(U^T S_B U), \\
\text{subject to} & \quad U^T S_W U = I,
\end{align*}
\]
where the \( S_B \) and \( S_W \) are the between and within scatters, respectively, defined as:
\[
\begin{align*}
\mathbb{R}^{d \times d} & \ni S_B := \sum_{j=1}^{c} n_j (\mu_j - \mu)(\mu_j - \mu)^\top, \\
\mathbb{R}^{d \times d} & \ni S_W := \sum_{j=1}^{c} \sum_{i=1}^{n_j} (x_i(j) - \mu_j)(x_i(j) - \mu_j)^\top,
\end{align*}
\]
where the mean of the \( j \)-th class is:
\[
\mathbb{R}^d \ni \mu_j := \frac{1}{n_j} \sum_{i=1}^{n_j} x_i(j).
\]
The total scatter can be considered as the summation of the between and within scatters [21], [47]:
\[
S_T = S_B + S_W \implies S_B = S_T - S_W.
\]
Therefore, the Fisher criterion can be written as [47]:
\[
\begin{align*}
f_F(U) &= \frac{d_T(U)}{d_W(U)} - 1 := \frac{\text{tr}(U^T S_B U)}{\text{tr}(U^T S_W U)} - 1.
\end{align*}
\]
The \(-1\) is a constant and can be dropped in the optimization problem because the variable \( U \) and not the objective is the goal; therefore, the optimization in FDA can now be expressed as:
\[
\begin{align*}
\text{maximize} & \quad \text{tr}(U^T S_B U), \\
\text{subject to} & \quad U^T S_W U = I.
\end{align*}
\]
Hence, the FDA directions can be obtained by the generalized eigenvalue problem \( (S_T, S_W) \) [47]. Note that some articles, such as [21], [22], [19], solve the generalized eigenvalue problem \( (S_B, S_T) \) by considering another version of the Fisher criterion which is \( \text{tr}(U^T S_B U)/\text{tr}(U^T S_T U) \). This criterion is obtained if we consider minimization of the inverse of Eq. (18), use Eq. (23) for \( S_W \), drop the constant \(-1\), and convert minimization to maximization by inverting the criterion again.

Comparing the Eqs. (17) and (25) shows that PCA captures the orthonormal directions with the maximum variance of data; however, FDA has the same goal but also it requires the manipulated directions to be orthonormal. This manipulation is done by the within scatter which makes sense because the within scatters make use of the class labels. This comparison gives a hint for the connection between PCA and FDA.
The SPCA [31] makes use of the empirical estimation of the Hilbert-Schmidt Independence Criterion (HSIC) [32]:

\[
\text{HSIC} := \frac{1}{(n-1)^2} \text{tr}(K_1 H K_2 H),
\]

where \(K_1\) and \(K_2\) are the kernels over the first and second random variable. The idea of HSIC is to measure the dependence of two random variables by calculating the correlation of their pulled values to the Hilbert space. The SPCA uses HSIC for the projected data \(U^\top X\) and the labels \(Y\) and maximizes the dependence of them in order to make use of the labels. It uses the linear kernel for the projected data \(K_1 = (U^\top X)(U^\top X) = X^\top U U^\top X\) and an arbitrary valid kernel for the labels \(K_2 = K_y\). Therefore, the scaled Eq. (26) in SPCA is:

\[
\text{tr}(X^\top U U^\top X HK_y H) \overset{(a)}{=} \text{tr}(U^\top XHK_yHX^\top U),
\]

where \((a)\) is because of the cyclic property of the trace. The optimization problem in SPCA is expressed as:

\[
\begin{align*}
\text{maximize} & \quad \text{tr}(U^\top XHK_yHX^\top U), \\
\text{subject to} & \quad U^\top U = I,
\end{align*}
\]

where \(K_y\) is the kernel matrix over the labels of data, either for classification or regression. The solution to Eq. (28) is the eigenvalue problem for \(XHK_yHX^\top\). Note that this term, restricted to a linear kernel for \(K_y\), is used as the between scatter in [22], [19], [20] hinting for a connection between SPCA and FDA if we compare the objectives in Eqs. (19) and (28).

PCA, FDA, and SPCA, which were explained, are several fundamental subspace learning methods which are different in terms of whether/how to use the labels. Comparing Eqs. (17), (25), and (28) and considering the general forms in Eqs. (11) and (15) show that these methods belong to a family of methods based on eigenvalue and generalized eigenvalue problems. This gave us a motivation to propose a generalized subspace learning method, named RDA, as a family of methods including PCA, FDA, and SPCA.

IV. ROWEIS DISCRIMINANT ANALYSIS (RDA)

A. Methodology

RDA aims at maximizing the \(\text{tr}(U^\top R_1 U)\) interpreted as the squared length of the reconstruction or the scatter of projection (see Eq. (10)), while requiring the manipulated projection directions to be orthonormal (see Eq. (14)). Therefore, the optimization of RDA is formalized as:

\[
\begin{align*}
\text{maximize} & \quad \text{tr}(U^\top R_1 U), \\
\text{subject to} & \quad U^\top R_2 U = I,
\end{align*}
\]

where \(R_1\) and \(R_2\) are the first and second Roweis matrices defined as:

\[
\begin{align*}
\mathbb{R}^{d\times d} & \ni R_1 := XHPHX^\top, \\
\mathbb{R}^{d\times d} & \ni R_2 := r_2 S_W + (1 - r_2) I,
\end{align*}
\]

respectively, where:

\[
\mathbb{R}^{n\times n} \ni P := r_1 K_y + (1 - r_1) I.
\]

The \(r_1 \in [0, 1]\) and \(r_2 \in [0, 1]\) are the first and second Roweis factors. Note that \(HPH\) in Eq. (30) is double-centering the matrix \(P\).

The solution to Eq. (29) is the generalized eigenvalue problem \((R_1, R_2)\) according to Eq. (16). Therefore, the RDA directions are the eigenvectors of this generalized eigenvalue problem. The RDA directions are sorted from the leading to trailing eigenvalues because of the maximization in Eq. (29). The optimization of the RDA can be interpreted in another way using what we define as the Roweis criterion. We want to maximize this criterion:

\[
\begin{align*}
\text{maximize} & \quad f_R(U) := \frac{d_{R_1}(U)}{d_{R_2}(U)} := \frac{\text{tr}(U^\top R_1 U)}{\text{tr}(U^\top R_2 U)}, \quad (33)
\end{align*}
\]

As in FDA, the Roweis criterion is a generalized Rayleigh-Ritz Quotient [45]; thus, the optimization in Eq. (33) is equivalent to Eq. (29). It is noteworthy that if we consider only a one-dimensional RDA subspace, the Roweis criterion is:

\[
\begin{align*}
&\text{maximize} \quad u^\top R_1 u, \\
&\text{subject to} \quad u^\top R_2 u = 1.
\end{align*}
\]

B. The Special Cases of RDA & the Roweis Map

Consider the Eqs. (30), (31), and (32). We know that the range of both \(r_1\) and \(r_2\) is \([0, 1]\). If we consider the extreme cases of \(r_1\) and \(r_2\), we find an interesting relationship:

\[
\begin{align*}
&\begin{align*}
& r_1 = 0, \quad r_2 = 0 \implies \text{RDA} \equiv \text{PCA}, \\
& r_1 = 0, \quad r_2 = 1 \implies \text{RDA} \equiv \text{FDA}, \\
& r_1 = 1, \quad r_2 = 0 \implies \text{RDA} \equiv \text{SPCA},
\end{align*}
& (36, 37, 38)
\end{align*}
\]

by comparing Eq. (29) with Eqs. (17), (25), and (28) and noticing the Eq. (9). We see that PCA, FDA, and SPCA are all special cases of RDA.

In fact, RDA is a family of infinite number of algorithms for subspace learning. By choosing any number for the \(r_1\) and \(r_2\) in the range \([0, 1]\), RDA gives us a new algorithm for learning the subspace of data. We define a map, named Roweis map, which includes the infinite number of special cases of RDA where three of its corners are PCA, FDA, and SPCA. The rows and columns of the Roweis map are the values of \(r_1\) and \(r_2\), respectively. Figure 1-a shows this map.

An interesting thing about RDA is that one of the four cases of \(r_1\) and \(r_2\) is not yet proposed in the literature to the best of our knowledge. If \(r_1 = 1\) and \(r_2 = 1\), the Eq. (29) becomes:

\[
\begin{align*}
&\begin{align*}
& \text{maximize} \quad \text{tr}(U^\top XHK_yHX^\top U), \\
& \text{subject to} \quad U^\top S_W U = I,
\end{align*}
& (39)
\end{align*}
\]
whose solution is the generalized eigenvalue problem \((XHX, S_W)\) according to Eq. (16). As can be seen, this optimization uses the labels twice, once in the kernel over the labels and once in the within scatter. Therefore, we name this special subspace learning method the Double Supervised Discriminant Analysis (DSDA).

C. Properties of the Roweis Matrices and Factors

1) Properties of the Roweis Factors: In Eq. (32), if \(r_1 = 1\), we are fully using the kernel over labels and if \(r_1 = 0\), it reduces to the identity matrix. So, we have:

\[
P = \begin{cases} 
K_y & \text{if } r_1 = 1, \\
I & \text{if } r_1 = 0.
\end{cases}
\]  

On the other hand, in Eq. (31), if \(r_2 = 1\), we are fully using the within scatter using the labels and if \(r_1 = 0\), it reduces to the identity matrix. Hence:

\[
R_2 = \begin{cases} 
S_W & \text{if } r_2 = 1, \\
I & \text{if } r_2 = 0.
\end{cases}
\]  

Therefore, we can conclude that if the Roweis factor is one, we fully use the labels as a supervised method and if the Roweis factor is zero, we are not using the labels at all. Therefore, the Roweis factor is a measure of using labels or being supervised. As we have two Roweis factors, we define:

\[
[0,1] \ni s := (r_1 + r_2)/2,
\]  

as the supervision level which is a planar function depicted in Fig. 2. Note that the extremes \(s = 0\) and \(s = 1\) refer to the unsupervised and fully (double) supervised subspace learning, respectively. Recall that the Roweis map includes the different subspace learning methods from unsupervised PCA (top-left corner of map) to the DSDA (bottom-right corner of map). The value \(s = 0.5\) can be interpreted as using the labels once (as in both FDA and SPCA).

It is noteworthy that if \(r_2 = 0\), we have \(R_2 = I\) according to Eq. (41). Therefore, according to Eqs. (29) and (40), the optimization of RDA merely includes \(K_y\) and not \(S_W\) regarding the labels. On the other hand, the \(K_y\) is a soft measure of similarity between the labels while the \(S_W\) uses the labels strictly for knowing which data instance belongs to which class. It shows that if we use \(S_W (r_2 \neq 0)\), the labels must be for classes in classification; however, if using solely \(K_y (r_2 = 0)\), the labels can be either for classification or regression. This sheds light to why SPCA (with \(r_2 = 0\)) can be used for both classification and regression tasks [31] but FDA (with \(r_2 = 1\)) is only for classification.

2) Properties of the Roweis Matrices: The Roweis matrices, \(R_1\) and \(R_2\), are positive semi-definite. The reason is that the within scatter, the kernel matrix, and the identity matrix are positive semi-definite and \(r_1, r_2 \in [0, 1]\), so:

\[
\begin{align*}
r_1 K_y \succeq 0, (1-r_1)I \succeq 0 \implies P \succeq 0, R_1 \succeq 0, \quad (43) \\
r_2 S_W \succeq 0, (1-r_2)I \succeq 0 \implies R_2 \succeq 0. \quad (44)
\end{align*}
\]

The Roweis matrices are also symmetric because the within scatter, the kernel matrix, and the identity matrix are symmetric; hence:

\[
R_1 = R_1^T, \quad R_2 = R_2^T, \quad P = P^T. \quad (45)
\]

It is also worth mentioning that according to the definitions of \(R_1\), \(R_2\), and \(P\), the \(P\) and the second Roweis matrices have the essence of kernel and scatter matrices, respectively, while the first Roweis matrix has the mixture essence of scatter and kernel matrices. According to Eq. (27), the first Roweis matrix is the term used in the HSIC over the projected data and the labels.

Now, we analyze the rank of the Roweis matrices. We have \(\mathbb{R}^{n \times n} \ni K_y := \Phi(Y)^\top \Phi(Y)\) where \(\Phi(Y) \in \mathbb{R}^t\) is the pulled \(Y\) to the feature space [15]. We usually have \(t \gg \ell\).
Hence, $\text{rank}(K_y) \leq \min(n, t)$. According to Eq. (32) and the subadditivity property of the rank:

$$\text{rank}(P) \leq \text{rank}(K_y) + \text{rank}(I_n) \leq \min(n, t) + n. \quad (46)$$

Note that in the extreme cases $r_1 = 0$ and $r_1 = 1$, the rank of $P$ is $n$ and $\leq \min(n, t)$, respectively. According to Eq. (30):

$$\text{rank}(R_1) \leq \min(\text{rank}(X) + \text{rank}(H) + \text{rank}(P)) \leq \min(d, n - 1, t) = \min(d, n - 1), \quad (47)$$

where the $-1$ is because of subtracting the mean in centering the matrix (e.g., consider the case with only one data instance). In general, the rank of a covariance (scatter) matrix over the $d$-dimensional data with sample size $n$ is at most $\min(d, n - 1)$. The $d$ is because the covariance matrix is a $d \times d$ matrix, the $n$ is because we iterate over $n$ data instances for calculating the covariance matrix, and the $-1$ is again because of subtracting the mean. Hence, according to Eq. (21), we have $\text{rank}(S_W) \leq \min(d, n - 1)$. According to Eq. (31) and the subadditivity property of the rank:

$$\text{rank}(R_2) \leq \text{rank}(S_W) + \text{rank}(I_d) \leq \min(d, n - 1) + d. \quad (48)$$

In the extreme cases $r_2 = 0$ and $r_2 = 1$, the rank of $R_2$ is $d$ and $\leq \min(d, n - 1)$, respectively.

\section*{D. Dimensionality of the RDA Subspace}

There exist a rigorous solution for the generalized eigenvalue problem $(R_1, R_2)$ [42]. However, we can solve this problem as:

$$R_1 U = R_2 U A \quad \Rightarrow \quad R_2^{-1} R_1 U = U A \quad \Rightarrow \quad U = \text{eig}(R_2^{-1} R_1), \quad (49)$$

where $\text{eig}(.)$ stacks the eigenvectors column-wise. If $R_2$ is singular, we strengthen its diagonal:

$$U = \text{eig}\left((R_2 + \varepsilon I)^{-1} R_1\right), \quad (50)$$

where $\varepsilon$ is a very small positive number, large enough to make $R_2$ full rank. In the literature, this approach is known as regularized discriminant analysis [48].

According to Eqs. (47) and (48), the rank of $R_2^{-1} R_1$ in Eq. (49) is:

$$\text{rank}(R_2^{-1} R_1) \leq \min(\text{rank}(R_2^{-1}), \text{rank}(R_1)) \leq \min(d, n - 1). \quad (51)$$

Therefore, the $p$, which is the dimensionality of the RDA subspace, is at most $\min(d, n - 1)$. The $\min(d, n - 1)$ leading eigenvectors are considered as the RDA directions and the rest of eigenvectors are invalid and ignored for having zero or very small eigenvalues.

It is noteworthy that according to Eq. (20), we have $\text{rank}(S_B) \leq \min(d, c - 1) = c - 1$. If Eq. (19) is used for FDA, we have $p \leq c - 1$ in FDA. In RDA, for the values $r_1 = 0, r_2 = 1$, RDA is reduced to FDA when the Eq. (25) is used; hence, the dimensionality of FDA as a special case of RDA is at most $\min(d, n - 1)$ and not $c - 1$.

\subsection*{E. Robust RDA}

According to Eq. (48), the rank of $R_2 \in \mathbb{R}^{d \times d}$ is at most $\min(d, n - 1) + d$. In the extreme case $r_2 = 1$, its rank is at most $\min(d, n - 1)$. Thus, for the case $r_2$ is very close to one, i.e., $r_2 \approx 1$, if $n - 1 < d$, the $R_2$ is singular. This happens when the dimensionality of data is huge but the sample size is small. In this case, we face a problem if we use the Eq. (49), for solving the generalized eigenvalue problem $(R_1, R_2)$. Two possible approaches to tackle this problem are the pseudo-inverse of $R_2$ [49] and strengthening the diagonal of $R_2$ [48], [19]. We can also propose the Robust RDA (RRDA) inspired by robust FDA [50], [51] to be robust against singularity. We emphasize that RRDA is useful if $r_2 \approx 1$, $n - 1 < d$, where we want to use Eq. (49) for solving the generalized eigenvalue problem.

In RRDA, the $R_2$ is decomposed using eigenvalue decomposition: $R_1 = \Phi \Lambda \Phi^\top$ where $\Phi$ and $\Lambda = \text{diag}([\lambda_1, \ldots, \lambda_d]^\top)$ include the eigenvectors and eigenvalues of $R_1$, respectively. The eigenvalues are sorted as $\lambda_1 \geq \cdots \geq \lambda_d$ and the eigenvectors are sorted accordingly. If $R_1$ is close to singularity, the first $d'$ eigenvalues are valid and the rest $(d - d')$ eigenvalues are either very small or zero. The appropriate $d'$ is obtained as:

$$d' := \arg \min_m \left(\frac{\sum_{j=1}^{m} \lambda_j}{\sum_{k=1}^{d} \lambda_k} \geq 0.98\right), \quad (52)$$

where 0.98 is a selected number close to 1. The $(d-d')$ invalid eigenvalues are replaced with $\lambda_s$:

$$\mathbb{R}^{d \times d} \ni \Lambda' := \text{diag}([\lambda_1, \ldots, \lambda_{d'}, \lambda_s, \ldots, \lambda_s]^\top), \quad (53)$$

where [50]:

$$\lambda_s := \frac{1}{d - d'} \sum_{j=d+1}^{d} \lambda_j. \quad (54)$$

Hence, the $R_2$ is replaced with $R_2'$:

$$\mathbb{R}^{d \times d} \ni R_2' := \Phi^\top \Lambda' \Phi, \quad (55)$$

and the robust RDA directions are the eigenvectors of the generalized eigenvalue problem $(R_1, R_2')$.

\section*{V. DUAL RDA for $r_2 = 0$}

The dual RDA exists for $r_2 = 0$ for the reason explained later in this section. This statement clarifies why, in the literature, PCA and SPCA (with $r_2 = 0$) have their dual methods but FDA (with $r_2 = 1$) does not have a dual.

As the matrix $K_y$ is symmetric and positive semi-definite, we can decompose it as:

$$K_y^{(a)} = \Psi_K \Omega_K \Psi_K^\top = \Psi_K \Omega_K^{(1/2)} \Omega_K^{(1/2)} \Psi_K^\top = \Upsilon \Upsilon^\top, \quad (56)$$

where $(a)$ is because of SVD and $(b)$ is for:

$$\mathbb{R}^{n \times n} \ni \Upsilon := \Psi_K \Omega_K^{(1/2)}. \quad (57)$$
Thus, the $R_1$ can be decomposed as:

$$ R_1 = XH(r_1K_y + (1 - r_1)I)H^T $$

$$ (a) = r_1XHYY^TH^TX + (1 - r_1)XHH^TX $$

$$ (b) = r_1QQ^T + (1 - r_1)XX^T $$

$$ (c) = r_1QQ^T + (1 - r_1)S_T $$

where $(a)$ is because of Eq. (56) and $H$ is symmetric, $(b)$ is because of Eq. (7) and:

$$ \mathbb{R}^{d \times n} \ni Q := XHY $$

and $(c)$ is because the $QQ^T$ and $XX^T$ are both symmetric and in the same form so their summation is also symmetric (recall Eq. (45)) and in the same form. We can obtain Eq. (58) in another way, too: as $R_1$ is symmetric and positive definite (see Eq. (43)), we can decompose it:

$$ R_1 = \Psi_R \Omega_R \Psi_R^T = \Psi_R \Omega_R^{(1/2)} \Omega_R^{(1/2)} \Psi_R^T $$

where $(a)$ is because of SVD and $(b)$ is for:

$$ W := \Psi_R \Omega_R^{(1/2)}. $$

For the reason that will be explained at the end of this section, we use the incomplete SVD in $(a)$ for $R_1$ where $\Psi_R \in \mathbb{R}^{d \times k}$, $\Omega_R \in \mathbb{R}^{k \times k}$, $\Omega_R := \min(d,n) [52]$. Therefore, we have $W \in \mathbb{R}^{d \times k}$. If we had used the complete SVD, we would have $W \in \mathbb{R}^{d \times d}$.

If $r_2 = 0$, we have $R_2 = I$ according to Eq. (31). In this case, the Eq. (29) becomes similar to Eq. (11) and thus the solution which is the generalized eigenvalue problem $(R_1, R_2)$ gets reduced to the eigenvalue problem for $R_1$. According to the properties of SVD, the matrix of left singular vectors of $W$ is equivalent to the matrix of eigenvectors of $WWW^T = R_1$. Therefore, we can use incomplete SVD for $W$:

$$ \mathbb{R}^{d \times k} \ni W = U \Sigma V^T, $$

where the columns of $U \in \mathbb{R}^{d \times k}$ and $V \in \mathbb{R}^{d \times k}$ are the left and right singular vectors (i.e., the eigenvectors of $WWW^T$ and $W^TW$), respectively. The diagonal entries of $\Sigma \in \mathbb{R}^{k \times k}$ are the singular value of $W$, i.e., the square root of eigenvalues of $WWW^T$ or $W^TW$.

The dual RDA exists only for $r_2 = 0$ because in Eq. (62), the $U$ is an orthogonal matrix so $U^TU = I$. This implies that the constraint in Eq. (29) should be $U^TU = I$ which means $R_2 = I$. Recall that this is the reason for FDA (with $r_2 \neq 0$) not having a dual.

We obtain $U$ from Eq. (62):

$$ WV \overset{(62)}{\Rightarrow} U \Sigma V^T \Rightarrow U = WV \Sigma^{-1}, $$

where the orthogonality of $V$ is noticed. The projection of data $X$ in dual RDA is:

$$ \tilde{X} \overset{(2)}{=} U^TX \overset{(63)}{=} (WV \Sigma^{-1})^TX = \Sigma^{-T}V^TW^TX $$

$$ \overset{(61)}{=} \Sigma^{-1}V^T \Omega_R^{(1/2)} \Psi_R^T X. $$

Note that $\Sigma$ is symmetric. Similarly, out-of-sample projection in dual RDA is:

$$ \tilde{X}_t = \Sigma^{-1}V^T \Omega_R^{(1/2)} \Psi_R^T X_t. $$

The reconstruction of training data in dual RDA is:

$$ \tilde{X} \overset{(3)}{=} UU^TX = U \tilde{X} $$

$$ \overset{(a)}{=} WV \Sigma^{-1} \Sigma^{-1}V^T \Omega_R^{(1/2)} \Psi_R^T X $$

$$ \overset{(61)}{=} \Psi_R \Omega_R^{(1/2)} V \Sigma^{-2} V^T \Omega_R^{(1/2)} \Psi_R^T X, $$

where $(a)$ is because of Eqs. (63) and (64). Similarly, reconstruction of out-of-sample data in dual RDA is:

$$ \tilde{X}_t = \Psi_R \Omega_R^{(1/2)} V \Sigma^{-2} V^T \Omega_R^{(1/2)} \Psi_R^T X_t. $$

To summarize, in RDA, the projection of training and out-of-sample data is Eq. (2) and (4), respectively. In RDA, the reconstruction of training and out-of-sample data is Eq. (3) and (5), respectively. However, in dual RDA, the projection of training and out-of-sample data is Eq. (64) and (65), respectively. In dual RDA, the reconstruction of training and out-of-sample data is Eq. (66) and (67), respectively.

Note that in RDA and dual RDA, we can truncate $U$, $\Sigma$, and $V$, respectively, to $U \in \mathbb{R}^{d \times k}$, $\Sigma \in \mathbb{R}^{p \times p}$, and $V \in \mathbb{R}^{d \times n}$ in order to have a $p$-dimensional subspace ($p \leq d$). For determining the appropriate $p$ in RDA, the scree plot [53] or the ratio $\lambda_j / \sum_{k=1}^d \lambda_k$ [54] can be used, where $\lambda_j$ is the $j$-th largest eigenvalue.

The dual RDA is very useful especially if the dimensionality of data is much greater than the sample size of data, i.e., $d \gg n$. In this case $W \in \mathbb{R}^{d \times k} = \mathbb{R}^{d \times n}$ (62). Accordin to Eq. (62), $U$ and $V$ are the eigenvectors of $WWW^T \in \mathbb{R}^{d \times n}$ and $W^TW \in \mathbb{R}^{n \times n}$, respectively. As $n < d$ in this case, the computation of eigenvectors of $W^TW$ is much faster and needs less storage than $WWW^T$. Therefore, in this case, we use the eigenvector decomposition:

$$ W^TW = V \Sigma^2 V^T, $$

rather than using Eq. (62). Note that in dual RDA, we do not require $U$ but only $\Sigma$ and $V$. Hence, if $n \ll d$, it is recommended to use dual RDA which is more efficient than RDA in terms of speed of calculation and storage.

VI. KERNEL RDA

Let $\phi : \mathcal{X} \rightarrow \mathcal{H}$ be the pulling function mapping the data $x \in \mathcal{X}$ to the feature space $\mathcal{H}$. In other words, $x \mapsto \phi(x)$. Let $t$ denote the dimensionality of the feature space, i.e., $\phi(x) \in \mathbb{R}^t$ while $x \in \mathbb{R}^d$. Note that we usually have $t \gg d$. The kernel over two vectors $x_1$ and $x_2$ is the inner product of their pulled data $[15]$:

$$ \mathbb{R} \ni k(x_1, x_2) := \phi(x_1)^T \phi(x_2). $$

We can have two types of kernel RDA, one using the dual RDA and kernel trick and the other one using the representation theory [18]. The former, which makes use of the inner product of data instances, holds for only two special cases but the latter works for the entire range of the Roweis map.
A. Kernel RDA Using Dual RDA for Special Cases

If \( r_2 = 0 \), we have the dual RDA and can use it for working out the kernel RDA using the kernel trick. However, the dual RDA is useful for the kernel trick in the two cases of \( r_1 = 0 \) and \( r_1 = 1 \). The reason is that in these cases, the inner product of data points appear enabling us to use the kernel trick. Therefore, we can use the dual RDA for kernel trick in \( r_1 = r_2 = 0 \) (i.e., PCA) and \( r_1 = 0, r_2 = 1 \) (i.e., SPCA). This explains why, in the literature, PCA and SPCA (with \( r_2 = 0 \)) have their kernel methods using kernel trick (or their dual) but kernel FDA (with \( r_1 = 0, r_2 = 1 \)) uses representation theory and not the kernel trick.

1) Special Case of Kernel RDA: Kernel PCA: If \( r_1 = 0 \), we have \( \mathbf{R}_1 = S_T = \tilde{X} \tilde{X}^\mathsf{T} \) according to Eq. (58). In this case, \( W = \tilde{X} = XH \). So, we decompose \( \tilde{X} \) in Eq. (62). In this case, RDA is reduced to PCA where the data should be centered. Hence, in reconstruction, we add the training mean back. Therefore, the Eqs. (64), (65), (66), and (67) are modified to:

\[
\tilde{X} = U \Sigma V^\mathsf{T} \quad \Rightarrow \quad U^\mathsf{T} \tilde{X} = \Sigma V^\mathsf{T} \quad \Rightarrow \quad \tilde{X} = \Sigma V^\mathsf{T},
\]

(70)

\[
\tilde{X}_t = \Sigma^{-1} V^\mathsf{T} \tilde{X}^\mathsf{T} \tilde{X}_t,
\]

(71)

\[
\tilde{X} = U \Sigma V^\mathsf{T} \quad \Rightarrow \quad U = \tilde{X} V \Sigma^{-1}
\]

\[
 \Rightarrow \tilde{X} = U \Sigma V^\mathsf{T} + \mu,
\]

(72)

\[
\tilde{X}_t = \tilde{X} V \Sigma^{-2} V^\mathsf{T} \tilde{X}^\mathsf{T} \tilde{X}_t + \mu,
\]

respectively, where \( \mu \) is because \( U \) and \( V \) are orthogonal matrices and \( \tilde{X}_t \) is the centered out-of-sample data using the training mean:

\[
\tilde{X}_t := X_t - \mu.
\]

(74)

In the kernel RDA for \( r_1 = r_2 = 0 \) (i.e., kernel PCA [13], [14]), the incomplete SVD is applied on the \( \Phi(X) := \Phi(X)H \) which is the centered data in the feature space. We use the Eqs. (70), (71), (72), and (73) and replace the inner products by the kernels:

\[
\Phi(X) = U \Sigma V^\mathsf{T}
\]

(75)

\[
\Phi(\tilde{X}) = \Sigma^{-1} V^\mathsf{T} \Phi(X) \Phi(X) = \Sigma^{-1} V^\mathsf{T} \tilde{X}_t,
\]

(76)

\[
\Phi(\tilde{X}) = \Phi(X) V \Sigma^\mathsf{T} + \mu,
\]

(77)

\[
\Phi(\tilde{X}) = \Phi(X) V \Sigma^{-2} V^\mathsf{T} \Phi(X) \Phi(X) + \mu
\]

(78)

where \( \Phi(X) \) is the centered out-of-sample data in the feature space using the mean of the pulled training data and \( \mu \) is because the double-centered kernel over the training and out-of-sample data is calculated as:

\[
\tilde{K}_t := \Phi(X)^\mathsf{T} \Phi(X) = \Phi(X)^\mathsf{T} \Phi(X) = K_x 1_{n \times n_t} = K_x 1_{n \times n_t}.
\]

\[
\tilde{K}_t := \Phi(X)^\mathsf{T} \Phi(X) = \Phi(X)^\mathsf{T} \Phi(X) = K_x 1_{n \times n_t},
\]

(79)

where \( \tilde{K}_t := \Phi(X)^\mathsf{T} \Phi(X) \) and \( \tilde{K}_x := \Phi(X)^\mathsf{T} \Phi(X) \). For the proof of Eq. (79), refer to the appendices in [14] and [40]. In Eqs. (77) and (78), there exists \( \Phi(X) \)

which is not necessarily available; therefore, in kernel RDA with \( r_1 = r_2 = 0 \) (kernel PCA), we cannot reconstruct.

In practice, the \( \Phi(X) \) is not available for its SVD decomposition; therefore, the \( V \) and \( \Sigma \) are found by eigenvalue problem for the double-centered training kernel, \( K_x := HH_x^\mathsf{T} \Phi(X)^\mathsf{T} \Phi(X) \):

\[
\tilde{K}_t = V \Sigma^2.
\]

(80)

2) Special Case of Kernel RDA: Kernel SPCA: If \( r_1 = 1 \), we have \( \mathbf{R}_1 = QQ^\mathsf{T} \) according to Eq. (58). In this case, \( W = Q = XH \). So, we decompose \( Q \) in Eq. (62). In this case, RDA is reduced to SPCA. Therefore, the Eqs. (64), (65), (66), and (67) are modified to:

\[
\tilde{X} = \Sigma^{-1} V^\mathsf{T} \Upsilon^\mathsf{T} \tilde{X}^\mathsf{T} \tilde{X}_t,
\]

(81)

\[
\tilde{X}_t = \Sigma^{-1} V^\mathsf{T} \Upsilon^\mathsf{T} \tilde{X}^\mathsf{T} \tilde{X}_t,
\]

(82)

\[
\tilde{X} = XHY \Sigma^{-2} V^\mathsf{T} \Upsilon^\mathsf{T} \tilde{X}^\mathsf{T} \tilde{X}_t,
\]

(83)

\[
\tilde{X}_t = XHY \Sigma^{-2} V^\mathsf{T} \Upsilon^\mathsf{T} \tilde{X}^\mathsf{T} \tilde{X}_t.
\]

(84)

In the kernel RDA for \( r_1 = 1, r_2 = 0 \) (i.e., kernel SPCA [31]), the incomplete SVD is applied on the \( \Phi(X) \). We replace the inner products by the kernels in the Eqs. (81), (82), (83), and (84):

\[
\Phi(\tilde{X}) = \Sigma^{-1} V^\mathsf{T} \Upsilon^\mathsf{T} HK_x,
\]

(85)

\[
\Phi(\tilde{X}_t) = \Sigma^{-1} V^\mathsf{T} \Upsilon^\mathsf{T} HK_t,
\]

(86)

\[
\Phi(\tilde{X}) = \Phi(X) H \Sigma^{-2} V^\mathsf{T} \Upsilon^\mathsf{T} HK_x,
\]

(87)

\[
\Phi(\tilde{X}_t) = \Phi(X) H \Sigma^{-2} V^\mathsf{T} \Upsilon^\mathsf{T} HK_t.
\]

(88)

In Eqs. (87) and (88), the term \( \Phi(X) \) exists which is not necessarily available so reconstruction cannot be done in kernel RDA with \( r_1 = 1, r_2 = 0 \) (i.e., kernel SPCA). Note that, in practice, the \( V \) and \( \Sigma \) are found by the eigenvalue problem \( K_x V = \Sigma^2 V \).

B. Direct Kernel RDA

1) Methodology: According to the representation theory [18], any pulled solution (direction) \( \phi(u) \in \mathcal{H} \) must lie in the span of all the training vectors pulled to \( \mathcal{H} \), i.e., \( \Phi(X) = [\phi(x_1), \ldots, \phi(x_n)] \in \mathbb{R}^t \times n \). Hence:

\[
\mathbb{R}^t \ni \phi(u) = \sum_{i=1}^n \theta_i \phi(x_i) = \Phi(X) \theta,
\]

(89)

where \( \mathbb{R}^n \ni \theta = [\theta_1, \ldots, \theta_n] \) is the unknown vector of coefficients, and \( \phi(u) \in \mathbb{R}^t \) is the pulled RDA direction to the feature space. The pulled directions can be put together in \( \mathbb{R}^t \times p \ni \Phi(U) = [\phi(u_1), \ldots, \phi(u_p)] \):

\[
\mathbb{R}^t \times p \ni \Phi(U) = \Phi(X) \Theta,
\]

(90)

where \( \mathbb{R}^n \ni \Theta = [\theta_1, \ldots, \theta_p] \).
According to Eqs. (33), (91), and (99), the Roweis criterion is equivalent to:

$$f(\Theta) := \frac{\text{tr}(\Theta^\top M \Theta)}{\text{tr}(\Theta^\top L \Theta)}.\quad (100)$$

Maximizing this generalized Rayleigh-Ritz Quotient [45] is equivalent to:

$$\max \Theta \quad \text{subject to} \quad \Theta^\top L \Theta = I,$$

whose solution is the generalized eigenvalue problem \((M, L)\) according to Eq. (16). Therefore, the direct kernel RDA directions are the eigenvectors of this generalized eigenvalue problem. The directions are sorted from the leading to trailing eigenvalues because of the maximization in Eq. (101). In kernel RDA, the directions are \(n\)-dimensional while in RDA, we had \(d\)-dimensional directions.

In kernel RDA, the projection and reconstruction of the training and out-of-sample data are:

$$\hat{X} = \Phi(U)^\top \Phi(X) := \Theta^\top K_x,\quad (102)$$

$$\hat{X} = \Phi(U) \Phi(U)^\top \Phi(X) := \Phi(X) \Theta \Theta^\top K_x,\quad (103)$$

$$\hat{X}_t = \Theta^\top K_t, \quad \hat{X}_t = \Phi(X) \Theta \Theta^\top K_t,\quad (104)$$

where \(\Phi(X)\) existing in the reconstructions are not necessarily available so we do not have reconstruction in kernel RDA.

2) Properties of \(M, N, L\): According to Eqs. (93), (98), and (32), we have:

$$M = \begin{cases} K_x H K_x & \text{if } r_1 = 1, \\
K_x H K_x & \text{if } r_1 = 0, \end{cases}\quad (105)$$

$$L = \begin{cases} N \sum_{j=1}^c K_j H_j K_j^\top & \text{if } r_2 = 1, \\
K_x & \text{if } r_2 = 0, \end{cases}\quad (106)$$

where it is noticed that \(H\) is idempotent. We see that if \(r_1 = 1\) and \(r_2 = 1\), the labels are used in calculation of \(M\) and \(N\), respectively. Also, if \(r_1 = 0\) and \(r_2 = 0\), the labels are not used in them at all. Thus, the Roweis factors are measures of using labels also in kernel RDA. The Eq. (42) can again be used as level of supervision. Moreover, according to Eqs. (105), (106), and (97), the matrices \(M, N, L\) are symmetric.

About the ranks of these matrices, we have \(\text{rank}(P) \leq \min(n, t) + n\) according to Eq. (46). As \(\mathbb{R}^{n \times n} \ni K_x = \Phi(X)^\top \Phi(X)\), we have \(\text{rank}(K_x) \leq \min(n, t)\). According to Eq. (93):

$$\text{rank}(M) \leq \min(\text{rank}(K_x) + \text{rank}(H) + \text{rank}(P)) \leq \min(n, n - 1, t), \quad (107)$$

where the \(-1\) is because of the centering the matrix and \((a)\) is because \(t\) is often large. In both the extreme cases \(r_1 = 0\) and \(r_2 = 1\), the rank of \(M\) is again at most \(n - 1\). According to Eq. (97), we have \(\text{rank}(N) \leq \min(n, c - 1)\) because it has \(c\) iterations each of which includes the centring matrix. According to Eq. (98) and the subadditivity property of the rank:

$$\text{rank}(L) \leq \text{rank}(N) + \text{rank}(K_x) \leq \min(n, c) - 1.\quad (108)$$

In the extreme cases \(r_2 = 0\) and \(r_2 = 1\), the rank of \(L\) is at most \(\min(n, t) = n\) and \(\min(n, n - 1)\), respectively.

3) Dimensionality of the Direct Kernel RDA Subspace: Similar to Eq. (49), we cal solve Eq. (101) as \(\Theta = \text{eig}(L^{-1} M)\) where the numeric hack of Eq. (50) may also be required. According to Eqs. (107) and (108), we have:

$$\text{rank}(L^{-1} M) \leq \min(\text{rank}(L^{-1}), \text{rank}(M)) \leq \min(n, c - 1).\quad (109)$$
Therefore, the dimensionality of the kernel RDA subspace is \( p \leq \min(n, c) - 1 \), restricted by rank of \( L \), where the \( \min(n-1, c-1) \) leading eigenvectors are considered as the kernel RDA directions and the rest of eigenvectors are invalid with very small eigenvalues. Notice that in most datasets, \( c < n \) so we usually have \( p \leq c - 1 \). Recall that this upperbound also exists on the dimensionality of the FDA and kernel FDA subspaces [8], [16], [46].

4) Special Cases of Kernel RDA: The Roweis map can have two layers, one for the input space and another for the feature space. The top layer is the bottom layer pulled to the feature space (see Fig. 1-b). Therefore, the four corners of Roweis map on the feature space can be considered as kernel PCA, kernel FDA, kernel FDA, and kernel DSDA. The whole map includes the kernel methods of an infinite number of subspace learning algorithms. Kernel PCA [13], [14], kernel FDA [16], [17], and kernel SPCA [31] are already proposed in the literature but the kernel method of the other algorithms in the Roweis map, such as kernel DSDA, are new.

As mentioned before, the methods in the Roweis map along \( r_2 = 0 \), including PCA and SPCA, have two kinds of kernelization which are using kernel trick (or dual of each method) and representation theory. This explains why there exist two types of kernel SPCA [31]. The kernel PCA using the kernel trick is already proposed [13], [14] while the kernel PCA using representation theory is proposed here for the first time to the best of our knowledge. The direct kernel RDA is reduced to exactly the existing direct kernel SPCA [31] for \( r_1 = 1, r_2 = 0 \).

VII. EXPERIMENTS
A. Visualization: Synthetic Nonlinear Datasets

One of the applications of subspace learning is data visualization. For the first experiment, we created two synthetic datasets which are nonlinear. The two datasets are binary XOR and concentric rings having two dimensions and two classes. The sample size in every dataset is 400 where 70% and 30% of the data are used for training and out-of-sample (test), respectively. The datasets are shown in Fig. 3.

As these datasets are highly nonlinear, RDA does not separate the classes as well as kernel RDA. Therefore, the first two dimensions of the embedding in nine special cases of kernel RDA (with Radial Basis Function (RBF) kernel for \( K_x \)) are illustrated in Fig. 3. Note that in all the classification experiments in this paper, we used the Kronecker delta kernel for \( K_y \) [31]. As this figure shows, kernel PCA does not perform as well as kernel SPCA, kernel FDA, and kernel DSDA. Overall, the larger the Roweis factors get, the better the two classes are separated which is expected because the supervision level is increased. By sweeping \( r_2 = 0 \rightarrow 1 \), the two classes are almost collapsed into two one dimensional lines because, according to Eq. (109), the rank of kernel RDA is restricted by \( c-1 = 1 \) when \( r_2 = 1 \) but this restriction does not exist for \( r_2 = 0 \). Another interpretation is because of taking the within scatter into account when \( r_2 \) is closer to one so the classes are collapsed. Moreover, these figures show that the RDA is capable of handling out-of-sample data well enough.

B. ROWEISFACES: THE GHOST FACES IN RDA

For the next experiment, we used the AT&T face dataset [55] including 40 subjects each having 10 images with different expressions and poses. The data were standardized to have zero mean and unit variance. We divided the dataset into two classes of images having and not having eye glasses.

1) The Facial Eigenvectors: We trained nine special cases, \( r_1, r_2 \in \{0, 0.5, 1\} \), of RDA and kernel RDA using the facial dataset. We name the facial eigenvectors, or ghost faces, in RDA as ROWEISFACES. The existing special cases of ROWEISFACES in the literature are eigenspaces \( (r_1 = r_2 = 0) \) [35], [36], Fisherfaces \( (r_1 = 0, r_2 = 1) \) [38], and supervised eigenspaces \( (r_1 = 1, r_2 = 0) \) [31], [40]. For \( r_1 = r_2 = 1 \) in Roweisfaces, we use the name double supervised eigenspaces. We name facial embedding using kernel RDA as kernel Roweisfaces whose existing special cases are kernel eigenspaces \( (r_1 = r_2 = 0) \) [37], kernel Fisherfaces \( (r_1 = 0, r_2 = 1) \) [39], kernel supervised eigenspaces \( (r_1 = 1, r_2 = 0) \) [31], [40]. For \( r_1 = r_2 = 1 \) in kernel Roweisfaces, we use the name kernel double supervised eigenspaces.

The eigenvectors in kernel RDA are \( n \)-dimensional and not \( d \)-dimensional so we can show the ghost faces only in Roweisfaces and not kernel Roweisfaces. The trained Roweisfaces are shown in Fig. 4 for the special cases. The PCA case has captured different features such as eyes, hair, lips, nose, and face border. However, the more we consider the labels by increasing \( r_1 \) and \( r_2 \), the more features related to eyes and cheeks are extracted because of the more discrimination of having or not having glasses. Increasing \( r_1 \) tends to extract more features like Haar wavelet features [56] which are useful for face feature detection (see Viola-Jones face detector [57]). Increasing \( r_2 \), however, fades out the irrelevant features leaving merely the eyes which are important. The double supervised eigenspaces have a mixture of Haar features and fading out unimportant features.

2) The Projections: The top two dimensions of projection of the facial images into the RDA and kernel RDA (with RBF kernel for \( K_x \)) subspaces are shown in Fig. 5. As expected, kernel RDA separates the classes better that RDA. Also, the larger the supervision level, the better the separation. For the same reasons explained for Fig. 3, the two classes are collapsed into one-dimensional lines for \( r_2 = 1 \) in kernel RDA.

3) The Reconstructions: As explained before, the data cannot be reconstructed in kernel RDA but it can be done in RDA. Some reconstructed images for the facial dataset in Roweisfaces are shown in Fig. 6. The quality of reconstruction falls down as \( r_2 \) is increased. We explain the reason in the following. According to Eq. (3), the reconstruction error for \( XA \) is \( ||XA - UU^T XA||_F^2 \) where \( A \) is a symmetric matrix. Minimizing this error where the bases are orthonormal is:

\[
\begin{align*}
\text{minimize} & \quad ||XA - UU^T XA||_F^2 \\
\text{subject to} & \quad U^T U = I,
\end{align*}
\] (110)

whose Lagrangian is simplified to \( L = \text{tr}(A^2 X^T X - A^2 X^T UU^T) \) noticing the constraint. Setting the derivative of Lagrangian to zero results in \( XA^2 X^T U = UA \) which is...
the eigenvalue problem for $X A^2 X^\top$. Comparing this to the solution of Eq. (29) and noticing Eq. (30), shows we can have $A^2 = HPH$, $r_2 = 0$, and $R_2 = I$ for minimization of reconstruction error. Hence, the best setting for reconstruction is to have $r_2 = 0$. In addition, if $r_1 = 0$, the objective in Eq. (110) becomes the error between $X$ and $\hat{X} = UU^\top X$ (see Eq. (3)) which is the reconstruction error of centered data. This explains why PCA (with $r_2 = 0$) is the best linear method for
C. RDA for Classification

To experiment on a classification task, we used a subset of the MNIST handwritten digit dataset [58] with 5000 training and 1000 test images. Note that since RDA uses a type of subspace learning based on eigenvalue and generalized eigenvalue problems, it cannot handle very large datasets. Prior works which make use of eigenvalue decomposition, such as [31], [59], have similar constraints. The data were standardized. We used 1-Nearest Neighbor (1-NN) classifier for the projected data instances because it shows the structure of embedded data. The error rates of the classification for some special cases of RDA and kernel RDA (with RBF kernel for \( K_x \)) are shown in Fig. 7 where we sweep over the dimensionality of subspace. As expected, in most cases, the kernel RDA performed better than RDA. For RDA, the DSDA and the case \( r_1 = r_2 = 0.5 \) were better than PCA and FDA which makes sense because DSDA has full supervision level. Also, kernel DSDA and the kernel method for \( r_1 = r_2 = 0.5 \) performed much better than kernel PCA and kernel SPCA.

D. RDA for Regression

According to [60] and inspired by [31], we used three synthetic benchmarks which are for evaluating regression tasks in subspace learning. Let \( X_{i,} \) and \( X_{i,} \) denote the \( i \)-th column (instance) and \( j \)-th row (feature) of \( X \), respectively. In the benchmark 1, \( X \in \mathbb{R}^{4 \times n} \) and the vector of labels is \( y = X_{1,:}/(0.5 + (X_{2,:} + 1.5)^2) + (1 + X_{3,:})^2 + 0.5 \varepsilon \) where \( \varepsilon \sim \mathcal{N}(0, 1) \) is the Gaussian additive noise and \( X_{i,:} \sim \mathcal{N}(0, I) \). In benchmark 2, \( X \in \mathbb{R}^{4 \times n} \) and \( y = \sin^2(\pi X_{2,:} + 1) + 0.5 \varepsilon \) where \( X_{i,:} \) is uniformly distributed on the set \([0, 1] \setminus \{x \in \mathbb{R}^4 | x_j \leq 0.7, \forall j \in \{1, 2, 3, 4\}\} \). The distribution of this benchmark is not elliptical. In benchmark 3, \( X \in \mathbb{R}^{10 \times n} \) and \( y = 0.5(X_{1,:})^2 \varepsilon \) where the noise is multiplicative here and \( X_{i,:} \sim \mathcal{N}(0, I) \). Inspired by [31], we drew 50 samples of size 100 from these benchmark distributions from each of which we took 70% and 30% of data for training and test, respectively. We used linear regression and the top two features of the projected data. We used RBF kernel for both \( K_x \) and \( K_y \) (note that the delta kernel was useful for classification task).

The average root mean squared errors of regression over the 50 created datasets are reported in Table I where only the cases with \( r_2 = 0 \) are used in RDA and kernel RDA because as explained before, regression only applies to \( r_1 = 0 \) in the Roweis map. As expected, often, the greater \( r_1 \) had better performance for the sake of larger supervision level. Moreover,
was also proposed using two methods of kernel trick and representation theory. Applying RDA and kernel RDA on facial dataset gave us Roweisfaces and kernel Roweisfaces generalizing eigenfaces, Fisherfaces, kernel eigenfaces, kernel Fisherfaces, and supervised eigenfaces. We also demonstrated some cases where DSDA, the novel method derived from the bottom-right corner of the Roweis map, provides superior preliminary results.

Our analysis of the theory of RDA and kernel RDA shed light to some facts about some of the existing subspace learning methods: (1) We showed that if \( r_2 = 0 \), the labels can be for either classification or regression. This explains why SPCA (with \( r_2 = 0 \)) can be used for both classification and regression while FDA (with \( r_2 = 1 \)) is only for classification task. (2) We showed that if \( r_2 = 0 \), the method has a dual which explains why PCA and SPCA (with \( r_2 = 0 \)) have their dual methods but FDA (with \( r_2 = 1 \)) does not have a dual. (3) If \( r_2 = 0 \) we have the dual required for the kernel trick. We also showed that only for \( r_1 = 0 \) and \( r_1 = 1 \), the inner product of the data points appear in the dual method. This explains why PCA (with \( r_1 = r_2 = 0 \)) and SPCA (with \( r_1 = 0, r_2 = 0 \)) have their kernel methods using kernel trick but kernel FDA (with \( r_1 = 0, r_2 = 1 \)) uses representation theory but not kernel trick. (4) If \( r_2 = 0 \), we showed that the reconstruction error can be minimized. This explains why PCA (with \( r_2 = 0 \)) is the best linear method for reconstruction error.

### Appendix A

**Derivation of Eq. (94)**

The mean of the \( j \)-th class, Eq. (22), in the feature space is:

\[
\mathbb{R}^l \ni \phi(\mu_j) := \frac{1}{n_j} \sum_{i=1}^{n_j} \phi(x_i^{(j)}).
\]  

We have:

\[
\begin{align*}
\phi(u)^\top \Phi(S_W) \phi(u) &= \left( \sum_{\ell=1}^{c} \sum_{i=1}^{n_j} \phi(\mu_j) \right) \left( \sum_{j=1}^{c} \sum_{i=1}^{n_j} (\phi(x_i^{(j)}) - \phi(\mu_j))^\top \left( \sum_{k=1}^{n} \theta_k \phi(x_k) \right) 
\end{align*}
\]

\[
\begin{align*}
&\sum_{j=1}^{c} \sum_{\ell=1}^{c} \sum_{i=1}^{n_j} \sum_{k=1}^{n} (\theta_\ell \phi(x_\ell)^\top (\phi(x_i^{(j)}) - \phi(\mu_j))^\top \theta_k \phi(x_k)) 
&= \sum_{\ell=1}^{c} \sum_{i=1}^{n_j} \frac{n_j}{n} \sum_{k=1}^{n} (\phi(x_i^{(j)}) - \phi(\mu_j))^\top \theta_k \phi(x_k) 
&= \left( \phi(x_i^{(j)}) - \phi(\mu_j) \right) ^\top \theta_k \phi(x_k)
\end{align*}
\]

In most cases, kernel RDA performs better than or similar to RDA.

### VIII. Summary and Conclusion

In this paper, we proposed RDA which generalized subspace learning including PCA, SPCA, and FDA. One of the extreme special cases is not yet in the literature and we named it Double Supervised Discriminant Analysis (DSDA). Dual RDA for some special cases was proposed here, too. Kernel RDA
\[
\begin{align*}
(69) & \quad \sum_{j=1}^{c} \sum_{\ell=1}^{n} \sum_{k=1}^{n_j} \sum_{i=1}^{n_j} \sum_{k=1}^{n_j} \sum_{i=1}^{n_j} \\
\quad & \left( \theta_k k(x_\ell, x_i) - \frac{1}{n_j} \sum_{e=1}^{n_j} \theta_k k(x_\ell, x_e) \right) \\
\quad & \left( \theta_k k(x_i, x_k) - \frac{1}{n_j} \sum_{z=1}^{n_j} \theta_k k(x_z, x_k) \right)
\end{align*}
\]
\[
(b) \quad \sum_{j=1}^{c} \sum_{\ell=1}^{n} \sum_{i=1}^{n_j} \sum_{k=1}^{n_j} \sum_{i=1}^{n_j} \sum_{k=1}^{n_j} \\
\quad & \left( \theta_k k(x_\ell, x_i) - \frac{1}{n_j} \sum_{e=1}^{n_j} \theta_k k(x_\ell, x_e) \right) \\
\quad & \left( \theta_k k(x_i, x_k) - \frac{1}{n_j} \sum_{z=1}^{n_j} \theta_k k(x_z, x_k) \right)
\]
\[
\begin{align*}
& = \sum_{j=1}^{c} \sum_{\ell=1}^{n} \sum_{k=1}^{n_j} \sum_{i=1}^{n_j} \sum_{k=1}^{n_j} \sum_{i=1}^{n_j} \\
& \left( \theta_k \theta_k k(x_\ell, x_i) \right) \\
& \left( \theta_k \theta_k k(x_i, x_k) \right) \\
& - \frac{2 \theta_k \theta_k}{n_j} \sum_{e=1}^{n_j} k(x_\ell, x_e) k(x_k, x_e) \\
& + \frac{\theta_k \theta_k}{n_j^2} \sum_{e=1}^{n_j} \sum_{z=1}^{n_j} k(x_\ell, x_e) k(x_k, x_z) k(x_i, x_z)
\end{align*}
\]
\[
= \sum_{j=1}^{c} \sum_{\ell=1}^{n} \sum_{k=1}^{n_j} \sum_{i=1}^{n_j} \sum_{k=1}^{n_j} \sum_{i=1}^{n_j} \\
\left( \theta_k \theta_k k(x_\ell, x_i) \right) \\
\left( \theta_k \theta_k k(x_i, x_k) \right) \\
- \frac{\theta_k \theta_k}{n_j} \sum_{e=1}^{n_j} k(x_\ell, x_e) k(x_k, x_e) \\
+ \frac{\theta_k \theta_k}{n_j^2} \sum_{e=1}^{n_j} \sum_{z=1}^{n_j} k(x_\ell, x_e) k(x_k, x_z) k(x_i, x_z)
\]
\[
(96) \quad \sum_{j=1}^{c} \left( \theta^\top K_j K_j^\top \theta - \theta^\top K_j \frac{1}{n_j} 11^\top K_j^\top \theta \right)
\]
\[
(97) \quad \sum_{j=1}^{c} \theta^\top K_j (I - \frac{1}{n_j} 11^\top) K_j^\top \theta
\]
\[
(98) \quad \sum_{j=1}^{c} \theta^\top K_j H_j K_j^\top \theta = \theta^\top \left( \sum_{j=1}^{c} K_j H_j K_j^\top \right) \theta
\]
\[
(99) \quad \theta^\top N \theta
\]

where \((a)\) is because of Eqs. (21) and (89) and \((b)\) is because \(k(x_1, x_2) = k(x_2, x_1) \in \mathbb{R}\). Q.E.D.
