Functional analysis techniques to improve similarity matrices in discrimination problems

Javier González\textsuperscript{a,}\textsuperscript{*}, Alberto Muñoz\textsuperscript{b}

\textsuperscript{a} Bernoulli Institute of Mathematics and Computer Science, University of Groningen, Nijenborg 9, 9747 AG, Groningen, Netherlands
\textsuperscript{b} Department of Statistics, Universidad Carlos III de Madrid, c/J Madrid 126, 28903 Getafe, Spain

\begin{abstract}
In classification problems an appropriate choice of the data similarity measure is a key step to guarantee the success of discrimination procedures. In this work, we propose a general methodology to transform the available data similarity matrix, incorporating the data labels, to improve the performance of discrimination procedures. We will focus on the case when the similarity matrix is asymmetric. We study the precise connection between similarity matrices and integral operators that will allow the evaluation of the transformed matrix on test points. The proposed methodology is used in several simulated and real experiments where the performance of several discrimination techniques is improved.
\end{abstract}

\section{Introduction}
In classification problems (with two classes) the data are usually given as a sample \( s_n = \{(x_i, y_i)\}_{i=1}^n \) where \( x_i \in X \) (some subset of \( \mathbb{R}^p \)) and \( y_i \in \{-1, 1\} \) are the labels. In other cases the available information is a similarity (or distance) matrix \( S \in \mathbb{R}^{n \times n} \) and the data labels. Distances and dissimilarities can be easily transformed into similarities (see for instance [9]).

Denote by \( X \) the matrix whose rows are the vectors \( x_1, \ldots, x_n \). Classification algorithms will use either the sample \( s_n \) (via the matrix \( XX^T \)) or the similarity matrix \( S \) (that will play the role of \( XX^T \) in most statistical discrimination algorithms such as Linear Discriminant Analysis or Flexible Discriminant Analysis) to build a discriminant rule.

When only \( S \) is available and it is positive definite, we can recover Euclidean data coordinates from \( S \) by using metric Multidimensional Scaling (MDS). Note that, given a positive semi-definite \( S \), there exists an Euclidean representation of the data points \( z_i \in \mathbb{R}^q \) (\( q \) being the rank of \( S \)) such that \( S = ZZ^T \) (where \( z_i \) are the rows of \( Z \)). Then the covariance data matrix of this representation will be, assuming centered data, \( S_c = Z^T Z \).

Sometimes the similarity matrix \( S \) is not symmetric. Some examples arise in micro-array data analysis [28], Text Retrieval problems [20,19] and Time Series analysis [1]. Given the lack of symmetry we cannot use spectral techniques (like MDS) neither to obtain Euclidean coordinates nor to find discrimination functions. In this case we need to find a symmetric and positive semi-definite matrix \( S^* \) close to \( S \), that will be used as input for classification algorithms.

In this paper, we will use the data labels to build \( S^* \). Hence we need a device to evaluate \( S^* \) for test data points (for which labels are unknown). To this aim we will use a Functional Data Analysis (FDA) approach. In FDA the concept of vector (finite
2. Estimating integral operators from similarity matrices

In this section, we study the connection between similarity/distance matrices and Hilbert–Schmidt integral operators. Let \( C(X) \) be the Banach space of continuous functions in \( X \) (\( X \) a compact set) with the norm \( \| f \|_\infty = \sup_{x \in X} |f(x)| \). Let \( L^2_+(X) \) be the space of square integrable functions in \( X \) where \( v \) is a Borel measure. Let \( \mathbb{K} : X \times X \rightarrow \mathbb{R} \) be a continuous function. In this case the (linear) map \( \mathbb{K} : L^2_+(X) \rightarrow C(X) \) defined by the operator

\[
(l_v f)(x) = \int K(x, t)f(t)d_v(t),
\]

is well defined and the function \( K \) is named the kernel of \( \mathbb{K} \). Several properties of \( \mathbb{K} \) can be derived from the properties of \( K \). For instance, if \( K \) is continuous then \( \mathbb{K} \) is compact and \( \| \mathbb{K} \| = \sqrt{\mathbb{K}(x, t)} \sup_{x, t \in X} |K(x, t)| \) where \( \mathbb{K}(x, t) \) is the measure of \( X \) [5, 6]. In the sequel we will exclusively concentrate on continuous, symmetric and positive definite kernels which are known as Mercer’s kernels [22, 3]. Then \( \mathbb{K} \) is self-adjoint, positive, compact and the Spectral Theorem applies [13]: There exists a countable sequence of eigenvalues \( \lambda_i \in \mathbb{R} \) and corresponding eigenfunctions \( \phi_i \) (\( i \geq 1 \)) of \( \mathbb{K} \). By Mercer’s theorem [22, 13] the function \( K \) can be expressed as \( K(x, t) = \sum_{i=1}^{\infty} \lambda_i \phi_i(x)\phi_i(t) \), where the convergence is absolute (for each \( x, t \in X \times X \)) and uniform (on \( X \times X \)).

Next we show the relationship between similarity/distance functions and integral operators to conclude that the natural operators corresponding to similarity (distances) matrices are integral operators.

**Proposition 1.** Let \( X \) be a compact set.

1. Consider an integral operator \( \mathbb{K} : L^2_+(X) \rightarrow C(X) \) with associated Mercer kernel function \( K(x, t) = \sum_{i=1}^{\infty} \lambda_i \phi_i(x)\phi_i(t) \).
   Define the map \( \Phi : X \rightarrow \ell^2 \) by \( \Phi(x) = (\sqrt{\lambda_i} \phi_i(x))_{i \in \mathbb{N}} \) (where \( \ell^2 \) is the linear space of all square summable sequences). Define a function \( d_K : X \times X \rightarrow \mathbb{R}^+ \) by

\[
d_K(x, t) = \| \Phi(x) - \Phi(t) \| = \left( \sum_{i \in \mathbb{N}} 2^{i/2} \lambda_i \phi_i(x) - \phi_i(t) \right)^{1/2},
\]

where \( \| \cdot \| \) represents the Euclidean norm in \( \ell^2 \). If \( \Phi \) is injective then the function \( d_K \) induced by \( \mathbb{K} \) on \( X \) is a metric and \( \Phi \) is an isometric mapping between \( (X, d_K) \) and \((\ell^2, d)\).

2. Consider a random sample \( s_n = \{x_1, \ldots, x_n\} \subset X \) and let \( S \) denote a symmetric and positive semi-definite \( n \times n \) similarity matrix, where \( (S)_{ij} \) represents the similarity between \( x_i \) and \( x_j \). Then there exists an integral operator \( \mathbb{K}\mathbb{S} \) such that

\[
K_{s_n} = S.
\]

That is \( K(x, y) = (S)_{ij} \forall x_i, x_j \in s_n \), where \( K \) is the Mercer kernel associated to \( \mathbb{K}\mathbb{S} \).

**Proposition 1** represents a manner to relate linear integral operators associated to Mercer kernels to similarity matrices. Given a symmetric and positive semi-definite similarity matrix \( S \), **Proposition 1.2** guarantees the existence of a Mercer kernel function

\[
K(x, y) = \sum_{j=1}^{n} \lambda_j \phi_j(x)\phi_j(y).
\]
where \( l_i > 0 \) and \( \{\varphi_1, \varphi_2, \ldots, \varphi_n\} \) is an orthogonal set of continuous functions in \( X \) such that \( K^S|_{S^n} = S \). That is, the evaluations of \( K^S \) on the sample correspond to the entries of \( S \). Of course, the basis \( \{\varphi_1, \varphi_2, \ldots, \varphi_n\} \) in the kernel expansion (4) is not necessarily unique and the question of how to choose an ‘appropriate’ basis arises. A natural choice is to consider the eigenfunctions of \( L^K \). However, these eigenfunctions are unknown since \( K^S \) is not available; fortunately they can be estimated: consider the spectral decomposition of \( S \), given by \( S = \sum_{j=1}^{n} l_j \varphi_j \) where \( (l_j, \varphi_j) \) are the pairs of eigenvalues and eigenvectors of \( S \). The next theorem guarantees that the vectors \( \varphi_j \) for \( j = 1, \ldots, n \) converge to the true eigenfunctions \( \phi_j \) of \( L^K \) when the sample size \( n \) increases.

**Theorem 1.** Let \( L_K \) be the integral operator associated to a kernel function \( K : X \times X \rightarrow \mathbb{R} \). Let \( \nu(X) \) be a Borel measure on \( X \) and \( v \) the empirical measure defined by \( v_n(X) = \frac{1}{n} \sum_{j=1}^{n} \delta(x_i) \) where \( I_X \) is the indicator function in \( X \). Let \( (I_K^n f)(x) = \int K(x, t)f(t)dt \) be the corresponding empirical integral operator, that is:

\[
(I_K^n f)(x) = \frac{1}{n} \sum_{i=1}^{n} K(x, x_i)f(x_i). \tag{5}
\]

Define \( \kappa = \sup_{x \in X} |K(x, x)| \) and let \( s_n = \{x_1, \ldots, x_n\} \) be a random sample independently drawn from \( \nu \) and \( K(x_i, x_j) = (K|_{s_n})_{ij} \) the resulting data kernel matrix components. Let \( \{\lambda_j, \phi_j\} \) denote the pairs of eigenvalues and eigenfunctions of \( L_K \). Then the following statements hold:

1. **Convergence of** \( I_K^n \) **to** \( L_K \): Let \( H_K \) be the RKHS associated to \( K \) and \( HS(H_K) \) the Hilbert space of Hilbert–Schmidt operators on \( H_K \). Then with a confidence of \( 1 - \delta \):

\[
\|I_K^n - L_K\|_{HS} \leq \frac{4\kappa^2 \log(2/\delta)}{\sqrt{n}}, \tag{6}
\]

where \( \|L_K\|_{HS} = \|K\|_{L_2^2(X)} \).

2. **Convergence of the eigenvalues of** \( I_K^n \) **to the eigenvalues of** \( L_K \): with a confidence of \( 1 - \delta \):

\[
\sup_{j \geq 1} (\lambda_j - \hat{\lambda}_j) \leq \frac{4\kappa^2 \log(2/\delta)}{\sqrt{n}}, \tag{7}
\]

where the \( \{\lambda_j\} \) and \( \{\hat{\lambda}_j\} \) are the sets of eigenvalues of \( L_K \) and \( I_K^n \) respectively.

3. **Convergence of the eigenfunctions of** \( I_K^n \) **to the eigenfunctions of** \( L_K \): For \( r = \min(\lambda_{j-1} - \lambda_j, \lambda_j - \lambda_{j+1}) > 0, \delta \in (0, 1) \) and \( \forall n \in \mathbb{N} \) then \( \frac{4\kappa^2 \log(2/\delta)}{\sqrt{n}} \leq \frac{1}{2} \) and with confidence \( 1 - \delta \):

\[
\left\| \frac{\phi_j}{\sqrt{\lambda_j}} - \hat{\phi}_j \right\| \leq \frac{16\kappa^2 \log(2/\delta)}{r \sqrt{n}}, \tag{8}
\]

where

\[
\hat{\phi}_j(x) = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} K(x, x_i)\nu_{ji} \tag{9}
\]

is the \( j \)-th eigenfunction of \( I_K^n \) associated to the \( j \)-th eigenvalue \( \hat{\lambda}_j \). The pairs \( (l_j, \varphi_j) \) are the eigenvalues and eigenvectors of \( \frac{1}{n}K|_{s_n} \) and \( \nu_{ji} \) is the \( i \)-th entry of the vector \( \nu_j \).

**Theorem 1** adapts to our problem several results from [26]. Statements 1 and 2 can be deduced from **Propositions 1** and 2 in that paper. Statement 1 represents a particular case of Theorem 2 in [26] (with non normalized kernels) that can also be deduced from the previous propositions. The next proposition shows that Eq. (9) provides a particular manner to find a basis (to construct the kernel function) such that **Proposition 1.2** holds.

**Proposition 2.** Let \( L_K \) be the integral operator associated to a kernel function \( K : X \times X \rightarrow \mathbb{R} \). Let \( \nu \) be a Borel measure on \( X \), and \( s_n = \{x_1, \ldots, x_n\} \) a random sample independently drawn from \( \nu \). Let \( L_K^n \) be its associated empirical operator defined as in **Theorem 1** and \( (K|_{s_n})_{ik} = K(x_i, x_k) \) the elements of the kernel matrix associated to \( K \) and \( s_n \). Then

\[
\sum_{j=1}^{n} \hat{\lambda}_j \hat{\phi}_j(x_i) \hat{\phi}_j(x_k) = (K|_{s_n})_{ik}, \tag{10}
\]

for any \( x_i, x_k \) in \( s_n \) where \( \hat{\lambda}_j \) are the eigenvalues of \( \frac{1}{n}K|_{s_n} \) and \( \hat{\phi}_j \) are the eigenfunctions of \( I_K^n \) given in Eq. (9).
3. Using the labels to improve similarity matrices in discrimination problems

Consider a discrimination problem with two classes. Denote by \( s_n = \{x_1, \ldots, x_n\} \) a sample where \( x_i \in X \) (some subset of \( \mathbb{R}^p \)). Consider \( S \in \mathbb{R}^{p \times n} \) a symmetric positive semi-definite similarity matrix for \( s_i \) and let \( \{y_1, \ldots, y_n\} \) be the corresponding class labels where \( y_i \in \{-1, 1\} \). The goal of this section is to obtain a new similarity matrix \( S' \) from \( S \), incorporating the information provided by \( \{y_1, \ldots, y_n\} \).

Such an \( S' \) can be obtained by means of LDA: Consider the decomposition \( S = ZZ^T \) where \( Z \in \mathbb{R}^{n \times q} \) is a matrix whose rows provide Euclidean coordinates for data points in \( \mathbb{R}^q \) (Z can be obtained, for example, by MDS). LDA finds the projection direction (or directions if the multiclass classification problem is considered) such that for each \( z_i \), the distance between the projected centroids of the two classes is maximum. Such direction is calculated as the first generalized eigenvector of two matrices that depend on the data labels: the “between class variability matrix” (generally denoted by \( S_b \)) and the “intra-class variability matrix” (\( S_w \)). Denote by \( W \in \mathbb{R}^{q \times q} \) the column vector containing such direction (first eigenvector of \( S_w^{-1}S_b \)). Then the squared Euclidean distance between two projected data \( z_iW \) and \( z_jW \) is \( d_{ij}^2 = \|z_iW - z_jW\|^2 = (z_i - z_j)A(z_i - z_j)^T \) where \( A = WW^T \). Hence the transformed matrix \( S^* \) for LDA is

\[
S^*_{\text{LDA}} = ZAZ^T. \tag{11}
\]

Unfortunately, the original \( S \) does not need to be neither symmetric nor positive definite. The following approach will be able to cope with this case.

**Definition 1.** Let \( y = (y_1, \ldots, y_n)^T \) where \( y_i \in \{-1, 1\} \) be the data labels of a classification problem. The \( S \) between-groups separation criterion for a general similarity matrix \( \tilde{S} \in \mathbb{R}^{n \times n} \) is defined as

\[
G(\tilde{S}) = \frac{1}{n} \sum_{i,j=1}^{n} (\tilde{S})_{ij} y_i y_j. \tag{12}
\]

Note that if two data points \( i \) and \( j \) belong to the same class then \( y_i y_j > 0 \) and \( y_i y_j < 0 \) otherwise. Therefore \( G(\tilde{S}) \) will be large if similarities \( \tilde{S}_{ij} \) are large for data points in the same class and small for data points in different classes. The most extreme case is reached when \( (\tilde{S})_{ij} \) equals some positive given constant for \( i, j \) in the same class and negative otherwise. Hence \( G(\tilde{S}) \) can be arbitrarily maximized by replacing \( \tilde{S} \) by a matrix \( S^*(x) = \alpha S_y \) where \( \alpha > 0 \) and \( S_y = yy^T \) for \( y = (y_1, \ldots, y_n)^T \). In particular it is straightforward to check that \( G[S^*(x)] = \alpha n \) and therefore \( \lim_{\alpha \to \infty} G[S^*(x)] = \infty \).

The distance induced by the (symmetric and positive semi-definite) similarity matrix \( S^*(x) = \alpha S_y \) between any two points \( x_i, x_j \) in \( s_n \) is given by

\[
d^{S^*(x)}(x_i, x_j) = \alpha (S_y)_{ij} + \alpha (S_y)_{ji} - 2\alpha (S_y)_{ij} = \begin{cases} d_{ij}^2 = 0 & \text{if } y_i = y_j \\ d_{ij}^2 = 4\alpha & \text{if } y_i \neq y_j. \end{cases} \tag{13}
\]

This distance \( d^{S^*(x)} \) projects data of the same class to the same point located at distance \( 4\alpha \) of the concentration point corresponding to the other class. Obviously the choice \( S^* = S^*(x) \) will produce a serious overfitting problem. See Appendix A for a justification of this over-fitting in the context of regularized classifiers. To prevent the possibility that the optimization of expression (12) leads to such a spurious solution we add to it a penalty term as follows.

**Definition 2.** Let \( S \in \mathbb{R}^{n \times n} \) be a similarity matrix and let \( \{y_1, \ldots, y_n\} \) be the data labels where \( y_i \in \{-1, 1\} \). The \( S \)-regularized between-groups separation criterion for a general similarity matrix \( \tilde{S} \in \mathbb{R}^{n \times n} \) is defined as

\[
G_{\lambda}(\tilde{S}) = \frac{1}{n} \sum_{i,j=1}^{n} (\tilde{S})_{ij} y_i y_j - \lambda \sum_{i,j=1}^{n} (\tilde{S}_{ij} - (S)_{ij})^2, \tag{14}
\]

where \( \lambda > 0 \).

In expression (14) the penalty term accounts for the deviations of each component of \( \tilde{S} \) from the elements of the original similarity matrix \( S \). The optimal value of parameter \( \lambda \) can be obtained by cross validation. The matrix that maximizes expression (14) is obtained in the following proposition.

**Proposition 3.** The matrix \( S^* \) that maximizes the \( S \)-regularized between-groups separation criterion in (14) with respect to \( \tilde{S} \) is given by

\[
S^* = S + \tau S_y, \tag{15}
\]

where \( \tau = 1/(2n\lambda) \).
The resulting matrix $S^*$ is therefore a linear combination (of positive weights $1$ and $\tau$) of the original similarity $S$ and $S_y$ and hence it will be positive semi-definite [4] if $S$ is. To be able to use $S^*$ as input of classification algorithms, only remains to give a method to calculate $(S^*)_{ij}$ when either $x_i$ or $x_j$ are unlabeled data points. The solution to this problem will be given in the next section by the extension of $S^*$ to its operator form using the results in Section 2.

The criterion in expression (12) admits an interesting interpretation in the case $S$ is symmetric and positive semi-definite. Consider the decomposition $S = ZZ^T$ where $Z \in \mathbb{R}^{n \times q}$ is a matrix whose rows provide Euclidean coordinates for data points in $\mathbb{R}^q$ ($Z$ can be obtained, for example, by MDS). Denote by $\tilde{y}_{ki} = 1$ if $z_i$ belongs to class $k$ and $0$ otherwise ($k \in \{1, 2\}$). If labels were unknown we could perform a cluster analysis by minimizing the k-means error function

$$1 \frac{1}{n} \sum_{i=1}^{n} \sum_{j \neq i} \tilde{y}_{ki}(z_i - m_k)^T(z_i - m_k).$$

(16)

Note that the variables to optimize in this case are the centers $m_1$, $m_2$. The obtained clusters would ideally correspond to the classes in our discrimination problem. In a classification problem, the labels $\tilde{y}_{ki}$ are given and therefore the centroids $m_1$, $m_2$ can be considered as constants. We can maximize expression (16) to obtain modified $z_i$ data coordinates (and thus indirectly modify $S = ZZ^T$) in such a way that the two data class groups (represented via the modified $S$) will be more separated. By concentrating points of each class around their respective centroids we are indirectly maximizing the margin between the classes within a 'kernel method' approach [14, 23].

A simple calculation shows that minimizing expression (16) in the $z_i$'s is equivalent to maximize

$$\frac{1}{n} \sum_{i,j=1}^{n} (S)_{ij} \left[ \sum_{k=1}^{n} \frac{\tilde{y}_{ki}\tilde{y}_{kj}}{n_k} \right],$$

(17)

with respect to the entries of the matrix $(S)_{ij}$, where $n_k$ is the number of data points belonging to cluster $k$ and $S = ZZ^T$. In this way we are using the K-means criterion to obtain a modified similarity matrix $S^*$ that increases the margin between the classes. Besides, notice that maximizing expression (17) also increases the intra-groups variability of the classes and therefore pursuit-projection classification methods that use this criterion (for instance, Fisher Discriminant Analysis) will expectedly increase their performance.

4. Application: asymmetric proximities in classification problems

Consider a two class discrimination problem. Denote by $s_n = \{x_1, \ldots, x_n\}$ the available data sample where $x_i \in X$ (some subset of $\mathbb{R}^p$) and $y_1, \ldots, y_n$ represent the labels where $y_i \in \{-1, 1\}$. Let $r_i = \{x_{n+1}, \ldots, x_{n+t}\}$ be a test sample of unlabeled data points and let $S$ be the $(n + t) \times (n + t)$ matrix where $(S)_{ij}$ is the similarity between any $x_i$ and $x_j$ in $s_n \cup r_i$. Through this section we will assume that $S$ is an asymmetric similarity matrix, that is $(S)_{ij} \neq (S)_{ji}$ for $i, j = 1, \ldots, n + t$. In the sequel $S^{[n]}$ denotes the similarity matrix consisting of the first $n$ rows and columns of $S$.

4.1. Obtaining the modified $S^*$ matrix from the asymmetric similarity matrix $S$ and the labels

The asymmetric matrix $S \in \mathbb{R}^{(n + t) \times (n + t)}$ can be written as $S = \frac{1}{2}(S + S^T) + \frac{1}{2}(S - S^T)$ where $\frac{1}{2}(S + S^T)$ is symmetric and $\frac{1}{2}(S - S^T)$ is skew-symmetric. Alternatively we can consider the triangular decomposition of $S$: let $S_1$ and $S_2$ be the two symmetric matrices built from the upper and lower triangular parts of $S$ respectively. That is, we consider that $(S_1)_{ij} = (S_2)_{ji} = (S)_{ij}$ whereas $(S_2)_{ij} = (S_2)_{ji} = (S)_{ji}$. It is straightforward to check that $\frac{1}{2}(S_1 + S_2) = \frac{1}{2}(S + S^T)$ and thus $\frac{1}{2}(S_1 + S_2)$ can be taken as the symmetric part of $S$.

Denote by $S^{[n]}_1$ and $S^{[n]}_2$ the two matrices consisting of the first $n$ rows and columns of $S_1$ and $S_2$. $S^*$ will be the solution to the $(S^{[n]}_1 + S^{[n]}_2)/2$-regularized between-groups separation criterion in (14):

$$S^*_{\text{BGP}} = \frac{1}{2}(S^{[n]}_1 + S^{[n]}_2) + \tau S_y.$$ 

(18)

The matrix $S^*_{\text{BGP}}$ in Eq. (18) does not need to be positive semi-definite since $S^{[n]}_1$ and $S^{[n]}_2$ are not necessarily positive semi-definite either. To convert $S^*_{\text{BGP}}$ into a positive semi-definite we will project it onto the convex cone of positive semi-definite matrices (PSD) of size $n$ defined by

$$M^+_n = \{M = M^T \in \mathbb{R}^{n \times n} : M \succeq 0\},$$

(19)

where $M \succeq 0$ means that $M$ is semi-positive-definite. To this aim we use the Orthogonal projection (using the Frobenius norm) of $S^*_{\text{BGP}}$ onto $M^+_n$ [12]

$$P(S^*_{\text{BGP}}) = \sum_{j=1}^{n} \max(0, l_j) v_j v_j^T,$$

(20)
where \( \mathbf{v} \) are the eigenvectors of \( S_{\text{MSGP}}^{*} \), and \( l \) its corresponding eigenvalues (some of them could be negative). Matrix \( \Pi (S_{\text{MSGP}}^{*}) \) is usually known as the positive part of \( S_{\text{MSGP}}^{*} \). Since \( \Pi (S_{\text{MSGP}}^{*}) \) is symmetric and positive semi-definite we can use it to calculate, via MDS, the data embedding onto some Euclidean space.

A possible alternative to symmetrize \( S \) is to project individually \( S_{1} \) and \( S_{2} \) onto the cone of positive semi-definite matrices and then to choose \( \alpha \) in the following linear combination

\[
S_{\text{RSP}}^{*} = \alpha \Pi (S_{1}) + (1 - \alpha) \Pi (S_{2}),
\]

where \( \alpha \in [0, 1] \), such that the similarity between \( S_{\text{RSP}}^{*} \) and \( S \) is maximized. This can be done using semi-definite programming [15]. In contrast to \( \Pi (S_{\text{RSP}}^{*}) \), \( S_{\text{RSP}}^{*} \) can be evaluated for both labeled and unlabeled data points.

In the next section, we detail how to estimate an integral operator associated to \( \Pi (S_{\text{RSP}}^{*}) \) to evaluate the symmetrized similarity on unlabeled data points. Comparison of both symmetrization procedures will be shown in the experimental section.

4.2. Estimating an integral operator from asymmetric similarity matrices

In this section, we address the problem of estimating a kernel function \( K^{*} \) such that \( K^{*}|_{\infty} \approx \Pi (S_{\text{RSP}}^{*}) \) able to extend \( \Pi (S_{\text{RSP}}^{*}) \) to test points in \( \mathcal{R} \) (that is unlabeled points whose labels are unknown). By Mercer’s theorem the unknown kernel \( K^{*} \) admits an expansion

\[
K^{*}(\mathbf{x}, \mathbf{t}) = \sum_{h=1}^{\infty} \lambda_{h} \phi_{h}(\mathbf{x}) \phi_{h}(\mathbf{t}) \quad \forall \mathbf{x}, \mathbf{t} \in \mathcal{X},
\]

where \( \lambda_{h} \) and \( \phi_{h} \) are respectively the eigenvalues and eigenfunctions of its associated integral operator \( L_{K^{*}} \). We propose an estimator \( \hat{K}^{*} \) of \( K^{*} \) by replacing \( \lambda_{h} \) and \( \phi_{h} \) in expression (22) by appropriate estimators \( \hat{\lambda}_{h} \) and \( \hat{\phi}_{h} \). Next we detail how to calculate such estimators.

1. Estimation of the eigenvalues of \( K^{*} \): Let \( d = \text{rank}(\Pi (S_{\text{RSP}}^{*})) \) and consider the matrix decomposition \( \Pi (S_{\text{RSP}}^{*}) = \mathbf{V} \mathbf{D} \mathbf{V}^{T} \) where \( \mathbf{V} = [\mathbf{v}_{1}, \ldots, \mathbf{v}_{d}] \) is the matrix whose columns are the eigenvectors of \( \Pi (S_{\text{RSP}}^{*}) \) and \( \mathbf{D} = \text{diag}(l_{1}, \ldots, l_{d}) \) is the diagonal matrix containing the eigenvalues. Following Theorem 1, we estimate each \( \lambda_{h} \) of \( K^{*} \) by

\[
\hat{\lambda}_{h} = \frac{l_{h}}{n}, \quad h = 1, \ldots, d,
\]

and it is therefore guaranteed that \( \hat{\lambda}_{h} \to \lambda_{h} \) when \( n \to \infty \).

2. Estimation of the eigenfunctions of \( K^{*} \): in contrast to the previous case we cannot use directly Eq. (9) in Theorem 1 to produce an estimator of each \( \hat{\phi}_{h} \) since the evaluations of \( K^{*} \) in points in \( \mathcal{R} \) are not available. Alternatively we will define some set of eigenfunctions that can be also evaluated in \( \mathcal{R} \).

Define \( S_{\text{av}} = (\Pi (S_{1} + S_{2})/2) \) where \( \Pi \) is the matrix projection method described in Section 4.1 and let \( \mathbf{D} \) and \( \mathbf{W} = [\mathbf{w}_{1}, \ldots, \mathbf{w}_{d_{av}}] \) be the matrices of eigenvalues and eigenvectors of \( S_{\text{av}}^{[n]} \) respectively where \( d_{av} \) is the rank of \( S_{\text{av}}^{[n]} \). By Proposition 1.2 there exists a Mercer kernel \( K_{\text{av}} \) such that \( K_{av}(\mathbf{x}, \mathbf{x}) = (S_{av})_{ij} \), for all points \( \mathbf{x}, \mathbf{x} \in \mathcal{S}_{n} \cup \mathcal{R} \). Denote by \( \{\hat{\phi}_{1}, \ldots, \hat{\phi}_{d_{av}}\} \) the set of eigenfunctions of \( K_{\text{av}} \) estimated using Eq. (9). Note that such eigenfunctions are available for points in \( \mathcal{R} \) using Eq. (9) (the entries of the matrix \( S_{\text{av}} \) are available for points both in \( \mathcal{S}_{n} \) and \( \mathcal{R} \)). In this work, we propose to estimate each \( \hat{\phi}_{h} \) in Eq. (22) by a linear combination of the set of functions \( \{\hat{\phi}_{1}, \ldots, \hat{\phi}_{d_{av}}\} \). That is

\[
\hat{\phi}_{h}(\mathbf{x}) = \sum_{j=1}^{d_{av}} c_{j} \hat{\phi}_{j}(\mathbf{x}),
\]

for any \( \mathbf{x} \in \mathcal{S}_{n} \cup \mathcal{R} \) and for \( h = 1, \ldots, d \). The criteria to fix the weights in expression (24) will be to take the weights of the orthogonal projection of each eigenvector \( \mathbf{v}_{h} \) of \( \Pi (S_{\text{RSP}}^{*}) \) onto the span of eigenvectors of \( S_{\text{av}}^{[n]} \). Given the relationship between eigenvectors and eigenfunctions stated in Theorem 1, the weights \( c_{j} \) estimated this way will approximate the weights of the orthogonal projection of each eigenfunction \( \phi_{h} \) onto \( \text{span}\{\phi_{1}, \ldots, \phi_{d_{av}}\} \) and it will guarantee reasonable estimators of the true eigenfunctions of \( K^{*} \).

In the sequel we will denote by \( \mathbf{c}_{h} = (c_{h,1}, \ldots, c_{h,d_{av}})^{T} \) the vector whose components are the weights of the linear combination in expression (24) and by \( \mathbf{C} = [\mathbf{c}_{1}, \ldots, \mathbf{c}_{d}] \) the matrix containing such values for the first \( d \) eigenfunctions of \( K^{*} \). Finally we define each \( \mathbf{c}_{h} = (c_{h,1}, \ldots, c_{h,d_{av}}) \) in Eq. (24) by

\[
\mathbf{c}_{h} = \arg \min \| \mathbf{v}_{h} - \mathbf{W} \mathbf{c}_{h} \|^{2}, \quad h = 1, \ldots, d.
\]

It is straightforward to check that, in matrix form, a solution of the \( d \) ordinary least squares problems in Eq. (25) (given by the columns of matrix \( \mathbf{C} \)) can be calculated in one step by

\[
\mathbf{C} = (\mathbf{W}^{T} \mathbf{W})^{-1} \mathbf{W}^{T} \mathbf{V}.
\]

Note that \( \mathbf{C} = \mathbf{W}^{T} \mathbf{V} \) if all the eigenvalues of \( S^{[n]} \) are all distinct.
Finally, the estimator of $K^*$ is obtained by replacing each $\lambda_h$ by $l_h/n$ and each $\varphi_h$ by the estimator given in Eq. (24) in expression (22). It is guaranteed that $\hat{K}^*$ is a kernel since the set $\{\varphi_h\}$ are continuous functions on the compact domain $X$ (they are linear combinations of continuous functions) and all the $l_h$ are positive.

To conclude this section we study the ability of the proposed methodology to approximate $\Pi(S_{\text{RSP}}^n)$ with $\hat{K}^*|_{s_n}$. When $(S_1^n + S_2^n)/2$ is positive semi-definite we bound the difference $\|\Pi(S_{\text{RSP}}^n) - \hat{K}^*|_{s_n}\|_F^2$ as it is stated in the following proposition.

**Proposition 4.** For $S_1^n$, $S_2^n$ and $\Pi(S_{\text{RSP}}^n)$ defined as in Section 4.1 and assuming that $(S_1^n + S_2^n)/2$ is positive definite it holds:

$$\|\Pi(S_{\text{RSP}}^n) - \hat{K}^*|_{s_n}\|_F^2 \leq 2\tau(n - \hat{y}^T\hat{y}),$$

where $\hat{y}$ is the orthogonal projection of the vector $y$ onto $\text{span}\{w_1, \ldots, w_{d_w}\}$ where $w_i (i = 1, \ldots, d_w)$ are defined in point 2 of Section 4.2. Here $\|A\|_F = \sum_{ij} a_{ij}$ denotes the Frobenius norm of the matrix $A$.

By Proposition 4, $\Pi(S_{\text{RSP}}^n) = \hat{K}^*|_{s_n}$ in two cases: first when we do not use the labels to define $\Pi(S_{\text{RSP}}^n)$ (that is when $\tau = 0$). Second when $W$ is full rank (in this case $d_w = n$ and $\{w_1, \ldots, w_{d_w}\}$ is a basis of $\mathbb{R}^n$). Hence we have that $y \in \text{span}\{w_1, \ldots, w_{d_w}\}$ so that $y^T\hat{y} = n$ (since $y$ and $\hat{y}$ are equals). Regarding the general case, when $n \rightarrow \infty$ then $\Pi(S_{\text{RSP}}^n) - \hat{K}^*|_{s_n} \rightarrow \Pi(S_{\text{RSP}}^n) - (K^*|_{s_n}) = \text{cte.}$ For $i, j = 1, \ldots, n$. Given that $y^T\hat{y} \leq n$ (see the proof of Proposition 4 for details) then by Proposition 4

$$\left(\Pi(S_{\text{RSP}}^n) - \hat{K}^*|_{s_n}\right)_{ij} \rightarrow 0$$

when $n \rightarrow \infty$ and thus each $\|\Pi(S_{\text{RSP}}^n) - \hat{K}^*|_{s_n}\|_F \rightarrow 0$.

### 4.3. Evaluating $\hat{K}^*$ in training and test points

In order to evaluate $\hat{K}^*$ at any pair of points (points in $s_n \cup r_t$) let us define the matrix

$$\Phi = S_{\text{nv}}^{[n+t,n]}W(D')^{-1},$$

where $S_{\text{nv}}^{[n+t,n]}$ are updated in the first $n + t$ rows and $n$ columns of $S_{\text{nv}}$. Using Eq. (9) it is straightforward to check that $(\Phi)_{ij} = \sqrt{-\lambda_i}(x_i)$ for $i = 1, \ldots, n + t, j = 1, \ldots, d_w$ where $\lambda_i$ is the $i$-th eigenvalue of $K_{\text{nv}}$. Finally, rewriting in matrix form the evaluations of expression (22) on $s_n \cup r_t$ it is straightforward to check that

$$\hat{K}^*|_{s_n \cup r_t} = (\Phi C)D(\Phi C)^T.$$  

Defining $Z^* = \Phi CD^{1/2}$ we have a d-dimensional Euclidean data representation for the data set which corresponds to the modified similarity matrix $S^*$.

### 4.4. Computational complexity of the method

The reconstruction of the kernel matrix involves the solution of $d$ linear systems of equations, the product matrix in (28) and the diagonalization of two matrices $n \times n$. This last step dominates the complexity being this complexity $O(n^4)$.

### 5. Applications and simulation study

In this section, we test our procedure in several scenarios. The first two examples correspond to experiments with real databases where asymmetry plays a relevant role. In Section 5.3, we include a simulation example to study some properties of the proposed method like the effect of the symmetrization and the influence of the parameter $\tau$ in (15) in the final classification results.

#### 5.1. Term classification in text data bases

In this section, we illustrate the performance of the proposed method to transform asymmetric similarities in a real term classification problem. To this aim we use the 20-Newsgroup data set which is a collection of approximately 20,000 newsgroup documents partitioned across 20 different topics [16]. This document collection has become a standard benchmark in text classification problems. In this experiment we focus on two particular topics: “Religion-Christian” and “Politics-Guns”. This choice is interesting because of the considerable overlap between the two classes. This 2-Newsgroup contains 1144 documents with a dictionary of 2564 words. To run the experiments we select a random sample of 1000 words of these two topics and we use for the experiment those that appear in at least 10 documents.
Define \( x_{ij} = 1 \) if the \( i \)-th term appears in the document \( j \)-th and 0 otherwise. Let \( X \) be the resulting \( 253 \times 1135 \) binary data text data matrix. Each term is assigned to one of the two classes ("Religion-Christian" and "Politics-Guns") by voting on the classes of documents in which the term appears.

Denote by \( |x_i| \) the number of documents indexed by the \( i \)-th term and by \( |x_i \wedge x_j| \) the number of documents indexed by both terms \( i \) and \( j \). In this experiment we consider the following asymmetric term similarity measure:

\[
s_{ij} = \frac{|x_i \wedge x_j|}{|x_i|} = \frac{\sum_k \min(x_{ik}, x_{jk})}{\sum_k x_{ik}}. \tag{29}
\]

Similarity \( s_{ij} \) measures the degree in which topic represented by term \( i \) is a subset of topic represented by term \( j \). Note that there is an implicit hierarchy associated to the use of \( s_{ij} \). For instance consider a text data base for which \( x_i = "Statistics" \) and \( x_j = "Bayesian" \). Suppose that the term "Statistics" appears in 100 documents while the term "Bayesian" is present only in 10. Assume also that in these 10 documents the term "Statistics" also occurs. Then the vector representing \( x_j \) will have one's at every position where \( x_j \) does and, therefore, \( |x_i \wedge x_j| = |x_j| \). Thus we obtain that \( s_{ij} = |x_j|/|x_j| = 10/100 = 0.1 \) while \( s_{ji} = |x_j|/|x_j| = 1 \).

The goal is to solve the term classification problem (labels defined above). To this aim we randomly select 80% of the data for training (sample \( s_n \) according the notation in Section 4) and the remaining 20% for testing (subsample \( r_i \)). Then we apply several classification procedures (described below) and obtain the classification errors as the averaged errors of 30 replications of the previous process.

The next step is to calculate the matrix \( S_{\text{RSP}} \) using (18). We assume that \( \tau > 0 \) and we fix its optimal value in a range of 15 equidistant values in the interval \([0.0001, 1]\) using a 5-fold cross validation strategy for each training sample. Note that \( S_{\text{RSP}} \) is defined only for \( s_n \). Next we calculate \( \Pi(S_{\text{RSP}}) \) using (20) (projection of \( S_{\text{RSP}} \) onto the cone of positive semi-definite matrices). The final step is to estimate the functional extension of the matrix \( \Pi(S_{\text{RSP}}) \) to the operator \( L_K \), such that \( K^*|_{s_n} \approx \Pi(S_{\text{RSP}}) \) which is done using the methodology proposed in Section 4.2.

It is natural to ask if \( K^*|_{s_n} \), the functional extension of the original asymmetric similarity matrix \( S \), will perform better than the natural symmetrization \( \overline{S} = (S_1 + S_2)/2 \) or even than the symmetric matrices \( S_1 \) and \( S_2 \). Note that these matrices are defined for both training and test points so we do not need to estimate their corresponding integral operator. However they are not necessarily positive semi-definite. Hence we project them using \( \Pi \) defined in Eq. (20) before applying the discrimination procedures.

It is also natural to ask what would happen if we ignore the existence of asymmetric similarity measures and use the covariance matrix \( X^T X \), that correspond to the choice of Euclidean distances for rows/terms of \( X \). In this case we run the classification procedures case for a slight modification of \( X \): attending the standard practice in Information Retrieval we consider the tf/ (non-binary) representation of terms given by [18]:

\[
x_{ij} = \frac{\text{times the term } i \text{-th appears in the document } j \text{-th}}{\text{times the term } i \text{-th appears in the data base}}.
\]

We also compare our proposal to the natural alternative proposed by Lanckriet et al. [15] described at the end of Section 4.1, that is to use Semi-Definite-Programming to symmetrize \( S \). See expression (11) for details.

The last alternative we will consider to symmetrize \( S \) is to use \( S_{\text{LDA}} \) defined in Eq. (11). Given that \( S \) does not need to be positive semi-definite we will replace it by \( \Pi(\overline{S}) \) (the projection of \( \overline{S} \) onto the cone of positive semi-definite matrices) to calculate the matrix \( W \) in expression (11). The discrimination procedures we will use to test the performance of the different symmetrization methods are: Support Vector Machines (SVMs) [27,23,14], Flexible Discriminant Analysis [10] and \( K \)-nearest neighbor. The three methods can accept the above proposed similarity matrices as input. Regarding the eligible parameters of the methods, the penalization term of the SVM is fixed to 100 in all the experiments. For the \( K \)-nn algorithm, we fix \( K = 5 \). The choice of these parameters is not crucial for our purposes since our goal here is to compare the performance of the representations rather than obtain optimal classification errors in this data set. Of course, if optimal errors want to be obtained for each classification technique a proper selection of the parameters can be performed.

We summarize the classification results of this experiment in Table 1 (top part). The worst performance is obtained for the single similarity measures \( s_{ij} \) and \( s_{ji}^* \) (results in the first two columns). This seems to indicate that combining will be a better strategy in general. The proposed similarity \( S_{\text{RSP}} \) obtains the best classification results with the three discrimination procedures. To test this hypothesis statistically we apply a paired one-tail T-test for the averaged errors (\( H_1 : \mu_d > 0 \), where \( \mu_d \) is the average of the differences between the errors using \( \overline{S} \) and \( S_{\text{RSP}} \)). We obtained a \( p \)-value of 0.0358 which seems to indicate that \( S_{\text{RSP}} \) performs better.

In Fig. 1 we illustrate one of the reasons of the good behavior of the similarity \( S_{\text{RSP}} \). Fig. 1(a) shows the representation of terms (using the two first components) by Multidimensional Scaling using the Euclidean distance and the tf representation. There is a strong overlap between the two term classes. Fig. 1(b) shows the corresponding MDS projection using is the MDS of the terms using the corresponding \( S_{\text{RSP}} \) (\( \tau = 0.0005 \)). The separation of both classes is now apparent.
Table 1
Averaged errors (30 runs) for the three classification techniques with different input matrices. Standard deviations are shown in parenthesis. In bold, the best result of each row is marked.

<table>
<thead>
<tr>
<th></th>
<th>$S_1$</th>
<th>$S_2$</th>
<th>$\overline{S}$</th>
<th>$S_{RSP}$</th>
<th>$S_{ZRP}$</th>
<th>$S_{LDA}$</th>
<th>$tf$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Textual data</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SVM</td>
<td>24.25</td>
<td>24.84</td>
<td>22.94</td>
<td><strong>18.43</strong></td>
<td>19.93</td>
<td>22.55</td>
<td>21.63</td>
</tr>
<tr>
<td></td>
<td>(1.10)</td>
<td>(1.18)</td>
<td>(0.65)</td>
<td>(0.81)</td>
<td>(0.79)</td>
<td>(0.99)</td>
<td>(0.79)</td>
</tr>
<tr>
<td>FDA</td>
<td>25.62</td>
<td>25.49</td>
<td>20.46</td>
<td><strong>20.39</strong></td>
<td>23.01</td>
<td>22.55</td>
<td>34.12</td>
</tr>
<tr>
<td></td>
<td>(1.00)</td>
<td>(0.99)</td>
<td>(1.15)</td>
<td>(1.08)</td>
<td>(0.99)</td>
<td>(0.97)</td>
<td>(0.73)</td>
</tr>
<tr>
<td>5-nn</td>
<td>20.46</td>
<td>20.98</td>
<td>15.95</td>
<td><strong>14.64</strong></td>
<td>18.30</td>
<td>23.53</td>
<td>40.48</td>
</tr>
<tr>
<td></td>
<td>(0.80)</td>
<td>(1.03)</td>
<td>(0.69)</td>
<td>(0.80)</td>
<td>(0.81)</td>
<td>(0.90)</td>
<td>(0.91)</td>
</tr>
<tr>
<td><strong>Microarray data</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SVM</td>
<td>19.67</td>
<td>19.33</td>
<td>18.83</td>
<td><strong>19.00</strong></td>
<td>19.50</td>
<td>27.00</td>
<td>33.17</td>
</tr>
<tr>
<td></td>
<td>(1.29)</td>
<td>(1.26)</td>
<td>(1.58)</td>
<td>(1.56)</td>
<td>(1.41)</td>
<td>(1.77)</td>
<td>(1.35)</td>
</tr>
<tr>
<td>FDA</td>
<td>35.00</td>
<td>37.00</td>
<td>34.33</td>
<td><strong>25.50</strong></td>
<td>33.83</td>
<td>28.00</td>
<td>40.00</td>
</tr>
<tr>
<td></td>
<td>(1.93)</td>
<td>(1.60)</td>
<td>(1.74)</td>
<td>(2.12)</td>
<td>(1.47)</td>
<td>(1.84)</td>
<td>(1.31)</td>
</tr>
<tr>
<td>5-nn</td>
<td>35.67</td>
<td>35.83</td>
<td>35.83</td>
<td><strong>26.50</strong></td>
<td>35.17</td>
<td>27.33</td>
<td>35.67</td>
</tr>
<tr>
<td></td>
<td>(1.64)</td>
<td>(1.83)</td>
<td>(1.70)</td>
<td>(2.08)</td>
<td>(1.93)</td>
<td>(1.84)</td>
<td>(1.51)</td>
</tr>
</tbody>
</table>

Fig. 1. Comparison of two terms representations using Multidimensional Scaling.

5.2. Classification of micro-array data

A genetic expression data base is usually a collection of DNA micro-array experiments where each column represents an experiment and each row a different gene. Generally, there are thousand rows and a few experiments. Each component of the data matrix measures the expression level of each gene in the target relative to each reference sample (experiment).

In this example we work with a Human Microarray Cancer data set [11]. The data correspond to 64 samples where the levels of expression of 6830 genes were measured. In this experiment we focus exclusively on the 16 samples labeled as “renal” or “colon” cancers. The range of the original data was from $-6$ to 6 measuring the expression levels of each gene. These values are recoded to 1 for expressed genes (positive values) and 0 for non expressed genes (negative values).

The use of asymmetric similarities in this example is justified. In Fig. 2 we show two plots containing the histograms of the norms of the words in the database described in Section 5.1 and the histogram of the norms of the genes of this experiment. As it is apparent, both histograms show a very similar shape that verifies Zipf’s law [21]. Just a few words (or genes) have a very large norm whereas for most of them the norm value is small. The behavior of the distribution of the norms of the terms in Fig. 2(a) has been previously shown to indicate an asymmetric relationship between terms [21]. In the same spirit, it does it for the genes in this example. See [7] for further details concerning Zipf’s law in genetics. Therefore, there exists a correspondence between micro-array and textual data: the genes play the role of the terms and the sample plays the role of the document. Hence it makes sense to use the asymmetric similarity $s_q$ defined in Eq. (29) to analyze this type of data.

In this experiment we randomly select a total of 100 genes. The label for each gene is assigned by a voting to the two classes “renal” or “colon. We estimate their similarity matrix using expression (29) and apply the same comparative scheme of the previous example.
Classification results are presented in Table 1 (bottom part). Results are coherent with those obtained in the previous example which confirms that the proposed method is the best in providing a similarity matrix that uniformly improves the performance of the different discrimination procedures tested. Only the SVM when the matrix $S$ is used as input slightly improves the proposed method but the difference is not significant. On the other hand, it always improves the SDP procedure and the matrix $S_{LDA}$. The tf representation of the genes is outperformed with the three classification techniques when the asymmetry of the problem is considered.

5.3. Impact of symmetrization and impact of use of the labels in the definition of $S_{RSP}$

In this section, we study the impact using expression (15) when it is considered to transform an original asymmetric similarity matrix. To this aim, we run a simulated experiment using the data described in Section 5.1. We consider a range of different values of $\tau$ in the interval $[0, 0.15]$ and we study the test errors of the three classification algorithms detailed in Section 5.1 (SVM, FDA and 5-nn) in such range. Then, we compare them with the errors provided by the techniques when no asymmetric similarities are considered (the tf representation of the terms). To perform this experiment we used 80% of the data to train the classifiers and the remaining 20% to calculate the errors. We calculate the average errors as the average of 30 replications of the experiment.

In Fig. 3 we show the average errors for the three techniques using the matrix $S_{RSP}$ in expression (15) for different values of $\tau$ (in black) and using the tf representation (horizontal lines in gray). Regarding the effect of the labels in the definition of $S_{RSP}$, for the FDA and 5-nn algorithms the error curve decreases with $\tau$ until the optimal error is reached to later increase again. These results show the positive influence of considering the labels in the definition of the similarity. The existence of an optimal $\tau$ for which the error is minimized is apparent for both techniques. Also, the results obtained using $S_{RSP}$ always improve the results obtained with the tf representation even for a none optimal selection of $\tau$. Regarding the results obtained
Fig. 4. Theoretical versus the estimated evaluations of $K^*$ in the test sample.

with the SVM, this method already finds competitive results with the tf representation and the margin of improvement is small. However, for values of $\tau$ smaller than 0.4, the use of $S_{RSP}^*$ has the effect of increasing the performance of the tested discrimination algorithms. Again, this justifies the use of an asymmetric similarity and the use of the labels in this example.

5.4. Quality of the approximation of the integral operator

To conclude the experimental section we study the quality of the approximation of the matrix $\Pi(S_{RSP}^*)$ by the operator $L_{K^*}$ for finite samples. That is, we empirically study the error bound in Proposition 4.

In order to illustrate the precision of our methodology to extend $\Pi(S_{RSP}^*)$ to test points, we use the textual data set described in Section 5.1. We split (randomly) the data set in a training sample $s_t$ made up of the 80% of the data and a test sample $r_t$ made up of the remaining 20%. Then entries of the matrix $\hat{K}^*|_{r_t}$ (the estimated kernel evaluated on test points) are compared to their theoretical values (values obtained using the true values of the labels in Eq. (18)). We study such comparative for a range of values of $\tau$ between 0 and 0.15.

In Fig. 4(d) we show, in terms of $\tau$, the average error for the entries of the true matrix $\hat{K}^*|_{r_t}$ for $\tau$ for 30 runs of the experiment. According to Proposition 4, as soon as the value of $\tau$ increases the approximation of the theoretical entries of the matrix is worse since the labels for test data are unknown. However, as we have already shown, to include the labels in the definition of the similarity of the problem improve the classification results.

In Plots 4(a)–(c) we show the scatter plot of the theoretical versus the estimated test values of $\Pi(S_{RSP}^*)$ for $\tau = 0.001, 0.025, 0.1$. The three plots exhibit a strong correlation (0.99, 0.97 and 0.90 respectively) between the theoretical...
and the estimated entries of the matrices assessing the quality of the functional approximation for points in \( r_i \) (points non labeled). Note that, as we may expect, the approximation is better for the smaller \( r \) what agrees again with Proposition 4.

6. Discussion

In this work, we have proposed several techniques to build similarity matrices useful in classification problems. Additionally we have studied their connection with certain type of integral operators by using a Functional Analysis approach. This relationship allows to generalize the concept of matrix to the concept of the integral operator likewise the concept of the vector is related to the concept of the function by using tools from functional analysis.

We have paid a special attention to problems where the data similarity matrix is asymmetric. In such cases the proposed approach allows us to define similarity matrices using directly the labels of the problem. We have shown the utility of this strategy in the experimental section where the errors of a set of classification algorithms were always minimized (among a battery of alternatives) when our proposed similarity was used as input of the discrimination procedures. For the future we aim to use our methodology to analyze other type problems where asymmetry plays an important role like problems in times series of DNA sequences.

Acknowledgments

This work is an improved version of one of the chapters of the dissertation of Dr. J. González defended in July 2010 at Carlos III University of Madrid and developed under the supervision of Prof. A. Muñoz. We thank the support of the projects, Nos. MEC-2007/04438/00, DGULM-2008/00059/00 and SBC-EMA-435065. We also thank the two anonymous reviewers of this work that contributed to improve significantly the original manuscript.

Appendix A. Overfitting produced by the matrix \( S^*(\alpha) \) in classification problems

Consider a discrimination problem with two classes. Denote by \( s_n = \{x_1, \ldots, x_n\} \) a sample where \( x_i \in X \) (some subset of \( \mathbb{R}^p \)) and let \( \{y_1, \ldots, y_n\} \) be the corresponding class labels where \( y_i \in \{-1, 1\} \). Let \( S^* \) be some positive semi-definite similarity matrix. Then there exists an Euclidean representation of the data points \( z_i \in \mathbb{R}^q \) (being \( q \) the rank of \( S \)) such that \( S = Z Z^T \) where \( z_i \) are the rows of \( Z \).

The goal of this section is to show that replacing \( S^* \) by the similarity matrix \( S^*(\alpha) = \alpha y y^T \) where \( \alpha > 0 \) and \( y = (y_1, \ldots, y_n)^T \), results in a serious overfitting when is used in regularized classifiers.

To solve the classification problem we will estimate a hyperplane \( D^*(z) = z w^* = 0 \) where \( z \in \mathbb{R}^q \) and where \( w^* = (w_1^*, \ldots, w_q^*)^T \) is the normal vector to the hyperplane \( D^* \). The classification rule will be assigned to class 1 any data point \( z_0 \in \mathbb{R}^q \) such that \( D^* (z_0) > 0 \) and to class \(-1\) otherwise. In our next reasoning we will assume that the solution to the classification problem is estimated using a regularization approach. In particular, we define \( w^* \) as the optimizer of the following problem:

\[
\begin{align*}
\text{arg min}_w & \quad \{Zw - y\}^T (Zw - y) + \lambda ||w||^2 \\
\end{align*}
\]

where \( \lambda \geq 0 \) and \( w = (w_1, \ldots, w_q)^T \). The first term penalizes deviations of each \( z_i w \) from the label values \( y_i \) while the second term penalizes large values of the norm of \( w \). Note that the quadratic loss function used in expression (30) can be replaced by any other convex loss function generalizing this process to other regularized classifiers.

The solution to the ridge regression problem stated in expression (30) is given by \( w^* = (Z^T Z + \lambda I_q)^{-1} Z^T y \) where \( I_q \) is the identity matrix of dimension \( q \times q \). In particular if we consider \( S^*(\alpha) \) as the data similarity matrix we obtain that \( S^*(\alpha) = Z Z^T \) where now \( Z = \sqrt{\alpha} y \) being \( q \) (the dimension of the representation using MDS) equal to one in this particular case. In addition, the optimal value of \( w \) in (30) is now given by

\[
\begin{align*}
aw^* &= (\alpha y y^T + \lambda)^{-1} \sqrt{\alpha} y y^T = \frac{n \sqrt{\alpha}}{n \alpha + \lambda},
\end{align*}
\]

and for any training point \( z_i \in s_n \) we have that

\[
D^*(z_i) = (w^*)^T z_i = \frac{n \sqrt{\alpha}}{n \alpha + \lambda} \sqrt{\alpha} y_i = \frac{n \alpha}{n \alpha + \lambda} y_i.
\]

Since \( n \alpha / (n \alpha + \lambda) > 0 \) it is straightforward that

\[
D^*(z_i) \begin{cases} > 0 & \text{if } y_i = 1 \\ < 0 & \text{if } y_i = -1, \end{cases}
\]

independently of the value of \( \lambda \). Thus using \( S^*(\alpha) \) as data similarity we always obtain a perfect classification rule for the training sample independently of the Bayes error of the problem. This will produce a clear problem of overfitting problem if \( D^*(z) \) is used to classify test points.
Appendix B. Proofs

Proof of Proposition 1. We prove Proposition 1 as follows:

1. By Mercer’s theorem for every \( x \in X \), \( \sum \lambda_j \phi_j^2(x) \) converges to \( K(x, x) \) that is \( \Phi(x) \in L^2 \). Given that \( K(x, t) = \sum_{j=1}^{\infty} \lambda_j \phi_j(x) \phi_j(t) \) for all \( x, t \in X \) the map \( \Phi : X \rightarrow L^2 \) given by \( x \mapsto (\sqrt{\lambda_j} \phi_j(x))_{j \in \mathbb{N}} \) satisfies \( K(x, t) = \langle \Phi(x), \Phi(t) \rangle \).

Thus \( K \) acts as a dot product in the embedding (the image of the map \( \Phi \)) induced by the eigenfunctions of the operator \( L_K \). Given \( x, t \in X \), by Eq. (2) the squared Euclidean distance between two points in the image of \( \phi \) is given by:

\[
    d^2(\Phi(x), \Phi(t)) = \langle \Phi(x), \Phi(x) \rangle + \langle \Phi(t), \Phi(t) \rangle - 2\langle \Phi(x), \Phi(t) \rangle = K(x, x) + K(t, t) - 2K(x, t).
\]

By Eq. (2) it is trivial to check that for all \( x, t, y \in X \) the function \( d_k \) satisfies: \( d_k(x, t) \geq 0 \), \( d_k(x, t) = d_k(t, x) \) and \( d_k(x, t) \leq d_k(x, y) + d_k(y, t) \). In addition since \( \Phi \) is injective \( d_k(x, t) = 0 \) if and only if \( x = t \) and \( d_k : X \times X \rightarrow \mathbb{R}^+ \). Therefore, every integral operator \( L_k \) induces a dissimilarity function \( d_k \) on \( X \) that will be a distance if the mapping \( \Phi \) is injective.

Now consider the metric spaces \((X, d_k)\) and \((L^2, d)\). Since for all \( x, t \in X \), \( \Phi \) is an injective mapping from \((X, d_k)\) to \((L^2, d)\) and \( d_k(x, t) = d(\Phi(x), \Phi(t)) \), we conclude that \( \Phi \) is an isometric mapping.

2. The second part of the proposition is proven as follows. Let \((l_j, v_j)\) for \( j = 1, \ldots, d \) denote the pairs of eigenvalues of eigenvectors of \( S \). Then \( S = \sum_{j=1}^{d} l_j v_j v_j^T \) where \( d = \text{rank}(S) \).

For each \( j = 1, \ldots, d \), consider the set of points \( \{(x_1, v_{1j}), \ldots, (x_n, v_{nj})\} \subset X \times \mathbb{R} \) where \( v_{kj} \) refers to the \( k \)-th entry of the vector \( v_j \). Consider interpolating polynomials \( p_j : X \rightarrow \mathbb{R} \) for \( j = 1, \ldots, d \) such that \( p_j(x_i) = v_{ji} \) (see [8,17] for a review of polynomial interpolation in several dimensions). Take \( v_j = p_j \) and define \( K^* \) by:

\[
    K^*(x, y) = \sum_{j=1}^{d} l_j \phi_j(x) \phi_j(y).
\]

\( K^* \) is obviously symmetric, continuous, and positive definite (the \( l_j \) are the eigenvalues of \( S \), a positive definite matrix). Thus \( K \) is a Mercer Kernel since \( \{\psi_1, \ldots, \psi_d\} \) is a finite set of continuous functions in \( X \) (See [24] for conditions on \( \psi \)'s to be \( K^* \) a kernel).

By construction of \( K^* \), \( K^*|_{s_n} = S \), that is \( K^*(x_i, x_j) = (S)_{ij} \) for all \( x_i, x_j \in s_n \) what concludes the proof. \( \square \)

Proof of Proposition 2. Let \( \{\hat{\lambda}_j, \hat{\phi}_j\} \) for \( j = 1, \ldots, n \) the pairs of eigenvalues and eigenfunctions of \( L^\alpha_k \). Then \( (L^\alpha_k \hat{\phi}_j)(x) = \hat{\lambda}_j \hat{\phi}_j(x) \) and thus

\[
    (L^\alpha_k \hat{\phi}_j)(x) = \frac{1}{n} \sum_{i=1}^{n} K(x_i, x) \hat{\phi}_j(x_i).
\]

Then if \( l_1, \ldots, l_n \) is the spectrum of \( \frac{1}{n} K|_{s_n} \), a natural estimator of the eigenvalues of \( L^\alpha_k \) is given by

\[
    \hat{\lambda}_j = l_j.
\]

In addition, the estimated eigenfunctions \( \hat{\phi}_j \) given in Eq. (9) (essentially the Nyström formula [2]) give rise the sample embedding (up to the factor \( \sqrt{n} \)) when evaluated in the sample: Let \( x_k \) be a sample point; then

\[
    \hat{\phi}_j(x_k) = \frac{1}{l_j \sqrt{n}} \sum_{i=1}^{n} K(x_i, x_k) v_{ji} = \frac{1}{l_j \sqrt{n}} n l_j v_{jk} = \sqrt{n} v_{jk}.
\]

Consider now the expansion given by \( \sum_{j=1}^{n} \hat{\lambda}_j \hat{\phi}_j(x) \hat{\phi}_j(t) \) the kernel function associated to \( L^\alpha_k \) where \( \hat{\lambda}_j \) and \( \hat{\phi}_j \) are obtained in Eqs. (9) and (35). Then applying the result in Eq. (36):

\[
    \sum_{j=1}^{n} \hat{\lambda}_j \hat{\phi}_j(x_k) \hat{\phi}_j(x_k) = n \sum_{j=1}^{n} l_j v_{ji} v_{jk} = (K|_{s_n})_{ik},
\]

for any \( x_i, x_k \) in \( s_n \). \( \square \)

Proof of Proposition 3. In order to maximize \( G_{ij}[\hat{S}] \) we take the partial derivative for each \( \hat{S}_{ij} \). Then

\[
    \frac{\partial G_{ij}}{\partial \hat{S}_{ij}} = \frac{1}{n} y_j y_i - 2 \lambda \left( \hat{S}_{ij} - S_{ij} \right)
\]
for $i, j = 1, \ldots, n$. Setting the previous partial derivatives to zero yields a linear system whose unique solution is
\[
S^*_y = S_y + \frac{1}{(2n\lambda)}y_j y_i.
\] (39)

for $i, j = 1, \ldots, n$. To check if $S^*$ is a maximum or a minimum we evaluate the Hessian matrix of $G_1[S]$ on $S^*$. Such matrix is the $n \times n$ diagonal matrix
\[
H(S^*) = \begin{pmatrix}
-2\lambda & 0 & \cdots & 0 \\
0 & -2\lambda & \cdots & 0 \\
\vdots & \ddots & \ddots & \vdots \\
0 & 0 & \cdots & -2\lambda
\end{pmatrix}
\]
which is always negative definite for any $\lambda > 0$. Hence, the matrix $S^* = S + \tau S_y$ where $\tau = 1/(2n\lambda)$ is a maximum of Eq. (14).

Proof of Proposition 4. By hypothesis we assume that $\Pi(S^*) = (S_1 + S_2)/2 + \tau S_y$. In order to estimate $\hat{K}^*$ define
\[
\hat{K}^* = \hat{K}_{av} + \hat{K}_y
\]
where $\hat{K}_{av}$ and $\hat{K}_y$ are the kernel functions estimated using the methodology proposed in Section 4.2 where now the matrices $(S_1 + S_2)/2$ and $S_y$ play the role of $\Pi(S^*)$ respectively.

Our aim is to give a bound for $\|\Pi(S^*) - \hat{K}^*|_{\|_F}$ using the previous decompositions of $\Pi(S^*)$ and $\hat{K}^*$ we have that:
\[
\|\Pi(S^*) - \hat{K}^*|_{\|_F} = \|\Pi(S_1 + S_2)/2 + \tau S_y - \hat{K}_{av}|_{\|_F} + \|\hat{K}_y|_{\|_F}. (40)
\]
To bound this error we first check that $(S_1 + S_2)/2 = \hat{K}_{av}|_{\|_F}$ as follows:
\[
\hat{K}_{av}|_{\|_F} = \sum_{h=1}^{d} \frac{n}{d} \left( \sum_{j=1}^{d} (w^T_j w_h) \sqrt{n}w_j \right) \left( \sum_{j=1}^{d} (w^T_j w_h) \sqrt{n}w_j \right)^T
\]
\[
= \sum_{h=1}^{d} l_h (w^T_h w_h) (w^T_h w_h)
\]
\[
= \sum_{h=1}^{d} l_h w_h w_h^T = (S_1 + S_2)/2,
\]
where we have applied that $w^T_h w_h = 1$ for $h = 1, \ldots, d$ and that $\phi_j(x_i) = \sqrt{n}w_{ji}$ (see Proposition 2 for details). Hence expression (40) simplifies to $\|\tau S_y - \hat{K}_y|_{\|_F}$. To give a bound to this error first note that $\tau S_y = t_1 u_1 u_1^T$ where
\[
t_1 = \tau \|y\|_2^2 \quad \text{and} \quad u_1 = \frac{y}{\|y\|_2},
\]
are the unique eigenvalue and eigenvector of $\tau S_y$. Denote $\hat{u}_1 = \sum_{j=1}^{d_{av}} (w^T_j u_1) w_j$. Then,
\[
\|\tau S_y - \hat{K}_y|_{\|_F} = \left\| t_1 u_1 u_1^T - \frac{t_1}{n} \left( \sum_{j=1}^{d_{av}} (w^T_j u_1) \sqrt{n} w_j \right) \left( \sum_{j=1}^{d_{av}} (w^T_j u_1) \sqrt{n} w_j \right)^T \right\|_F
\]
\[
= t_1 \left\| u_1 u_1^T - \hat{u}_1 \hat{u}_1^T \right\|_F^2
\]
\[
= t_1^2 \left( \|u_1 u_1^T\|_F^2 + \|\hat{u}_1 \hat{u}_1^T\|_F^2 - 2 \langle u_1 u_1^T, \hat{u}_1 \hat{u}_1^T \rangle_F \right)
\]
\[
= t_1^2 \left( \|u_1\|_2^2 + \|\hat{u}_1\|_2^2 - 2 \langle u_1, \hat{u}_1 \rangle^2 \right)
\]
\[
\leq 2t_1 (1 - \langle u_1, \hat{u}_1 \rangle^2),
\]
applying that $\|u_1\|_2^2 = 1$ and that $\|\hat{u}_1\|_2^2 \leq \|u_1\|_2^2 = 1$ ($\hat{u}_1$ is the projection of the vector $u_1$ onto span{w_1, \ldots, w_{d_{av}}}). Then replacing in $2t_1 (1 - \langle u_1, \hat{u}_1 \rangle^2)$ the values of $t_1$ and $\hat{u}_1$ we conclude the proof:
\[
2t_1 (1 - \langle u_1, \hat{u}_1 \rangle^2) = 2 \tau \|y\|_2^2 \left( 1 - \frac{1}{\|y\|_2} y^T \sum_{j=1}^{d_{av}} (w^T_j y) w_j \right)
\]
\[
= 2 \tau \|y\|_2^2 \left( 1 - \frac{1}{\|y\|_2} y^T y \right)
\begin{align*}
\left\| y \right\|^2_2 &= n \quad \text{and} \quad \hat{y} = \sum_{j=1}^{d_{av}} (w^T_j y) w_j, \quad \text{that is, the orthogonal projection of the vector $y$ onto the set of eigenvectors of $S_{av}^{[n]}$.} \quad \square
\end{align*}

References