Non-parallel semi-supervised classification based on kernel spectral clustering

Siamak Mehrkanoon and Johan A. K. Suykens

Abstract— **In this paper, a non-parallel semi-supervised algorithm based on kernel spectral clustering is formulated. The prior knowledge about the labels is incorporated into the kernel spectral clustering formulation via adding regularization terms. In contrast with the existing multi-plane classifiers such as Multisurface Proximal Support Vector Machine (GEPSVM) and Twin Support Vector Machines (TWSVM) and its least squares version (LSTSVM) we will not use a kernel-generated surface. Instead we apply the kernel trick in the dual. Therefore as opposed to conventional non-parallel classifiers one does not need to formulate two different primal problems for the linear and nonlinear case separately. The proposed method will generate two non-parallel hyperplanes which then are used for out-of-sample extension. Experimental results demonstrate the efficiency of the proposed method over existing methods.**

I. Introduction

In the last few years there has been a growing interest in semi-supervised learning in the scientific community. Generally speaking, machine learning can be categorized into two main paradigms, i.e., supervised versus unsupervised learning.

The task in supervised learning is to infer a function from labeled training data. Whereas unsupervised learning refers to the problem where no labeled data are given. In the supervised learning setting the labeling process might be expensive or very difficult. Then the alternative is semi-supervised learning which concerns the problem of learning in the presence of both labeled and unlabeled data [1], [2] and [3]. In most cases one encounters a large amount of unlabeled data while the labeled data are rare.

Most of the developed approaches attempt to improve the performance of the algorithm by incorporating the information from either the unlabeled or labeled part. Among them are graph based methods that assume that neighboring point pairs with a large weight edge are most likely within the same cluster. The Laplacian support vector machine (LapSVM) [4], a state of art method in semi-supervised classification, is one of the graph based methods which provide a natural out-of-sample extension.

Spectral clustering methods, in a completely unsupervised fashion, make use of the eigenspectrum of the Laplacian matrix of the data to divide a dataset into natural groups such that points within the same group are similar and points in different groups are dissimilar to each other. However it has been observed that classical spectral clustering methods suffer from the lack of an underlying model and therefore do not possess an out-of-sample extension naturally.

Kernel spectral clustering (KSC) introduced in [5] aims at overcoming these drawbacks. The primal problem of kernel spectral clustering is formulated as a weighted kernel PCA. Recently the authors in [6] have extended the kernel spectral clustering to semi-supervised learning by incorporating the information of labeled data points in the learning process. Therefore the problem formulation is a combination of unsupervised and binary classification approaches. The concept of having two non-parallel hyperplanes for binary classification was first introduced in [7] where two non-parallel hyperplanes were determined via solving two generalized eigenvalue problem and the method is termed GEPSVM. In this case one obtains two non-parallel hyperplanes where each one is as close as possible to the data points of the one class and as far as possible from the data points of the other class. Some efforts have been made to improve the performance of GEPSVM by providing different formulations such as in [8]. The authors in [8] proposed a non-parallel classifier called TWSVM, that obtains two non-parallel hyperplanes by solving a pair of quadratic programing problems. An improved TWSVM termed as TBSVM is given in [9] where the structural risk is minimized.

Motivated by the ideas given in [10] and [11], recently the least squares twin support vector machines (LSTSVM) is presented in [12] where the primal quadratic problems of TSVM is modified in least square sense and and inequalities are replaced by equalities. In all the above mentioned approaches, for designing a nonlinear classifier, they utilize kernel-generated surfaces. In addition they have to construct different primal problems depending on whether a linear or nonlinear kernel is applied. It is the purpose of this paper to formulate a non-parallel semi-supervised algorithm based on kernel spectral clustering for which we can directly apply the kernel trick and thus its formulation enjoys the primal and dual properties as in a support vector machines classifier [13], [14].

This paper is organized as follows. In Section II, a brief review of binary kernel spectral clustering and its semi-supervised version are given. Section III briefly introduces twin support vector machines. In Section IV a non-parallel semi-supervised classification algorithm based on kernel spectral clustering is formulated. Model selection is explained in Section V. Section VI describes the numerical experiments, discussion, and comparison with other known methods.

The authors are with the Department of Electrical Engineering ESAT-SCD-SISTA, Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, B-3001 Leuven, Belgium (email: {siamak.mehrkanoon, johan.suykens}@esat.kuleuven.be).

978-1-4673-6129-3/13/$31.00 ©2013 IEEE

II. Semi-Supervised Kernel Spectral Clustering

In this section first binary Kernel spectral clustering is briefly reviewed and then the formulation of semi-supervised KSC given in [6] is summarized.

A. Primal and Dual formulation of binary KSC

The method corresponds to a weighted kernel PCA formulation providing a natural extension to out-of-sample data i.e. the possibility to apply the trained clustering model to out-of-sample points. Given training data $\mathcal{D} = \{x_i\}_{i=1}^M$, $x_i \in \mathbb{R}^d$ and adopting the model of the following form

$$e = w^T\varphi(x_i) + b,\ i = 1, \ldots, M,$$

the binary kernel spectral clustering in the primal is formulated as follows:

$$\begin{array}{ll} \min\limits_{w,b,e} & \frac{1}{2}w^Tw - \frac{\gamma}{2}e^TVe \\ \text{subject to} & \Phi w + b1_M = e \end{array} \tag{1}$$

Here $\Phi = [\varphi(x_1), \ldots, \varphi(x_M)]^T$ and a vector of all ones with size M is denoted by 1_M. $\varphi(\cdot) : \mathbb{R}^d \rightarrow \mathbb{R}^h$ is the feature map and h is the dimension of the feature space. $V = \text{diag}(v_1, ..., v_M)$ with $v_i \in \mathbb{R}^+$ is a user defined weighting matrix. Applying the Karush-Kuhn-Tucker (KKT) optimality conditions one can show that the solution in the dual can be obtained by solving an eigenvalue problem of the following form:

$$VM_v\Omega\alpha = \lambda\alpha, \tag{2}$$

where $\lambda = 1/\gamma$, $M_v = I_M - (1/1_M^TV1_M)1_M1_M^TV$, I_M is the $M \times M$ identity matrix and $\Omega_{ij} = K(x_i, x_j) = \varphi(x_i)^T\varphi(x_j)$. It is shown that if

$$V = D^{-1} = \text{diag}(\frac{1}{d_1}, ..., \frac{1}{d_M}),$$

where $d_i = \sum_{j=1}^M K(x_i, x_j)$ is the degree of the $i-$th data point, the dual problem is related to the random walk algorithm for spectral clustering.

For model selection several criteria have been proposed in literature including a Fisher [15] and Balanced Line Fit (BLF) [5] criterion. These criteria utilize the special structure of the projected out-of-sample points or estimate out-of-sample eigenvectors for selecting the model parameters. When the clusters are well separated the out-of-sample eigenvectors show a localized structure in the eigenspace.

B. Primal and Dual formulation of semi-supervised KSC

KSC is an unsupervised algorithm, by nature, but it has shown its ability to also deal with both labeled and unlabeled data at the same time by incorporating the information of the labeled data into the objective function. Consider training data points $\{x_1, ..., x_N, x_{N+1}, .., x_M\}$ where $\{x_i\}_{i=1}^M \in \mathbb{R}^d$. The first N data points do not have labels whereas the last $N_L = M - N$ points have been labeled with $\{y_{N+1}, .., y_M\}$ in a binary fashion. The information of the labeled samples are incorporated to the binary kernel spectral clustering (1) by means of a regularization term which aims at minimizing the squared distance between the projections of the labeled samples and their corresponding labels [5]

$$\begin{array}{ll} \min\limits_{w,b,e} & \frac{1}{2}w^Tw - \frac{\gamma}{2}e^TVe + \frac{\rho}{2}\sum\limits_{m=N+1}^{M}(e_m - y_m)^2 \\ \text{subject to} & \Phi w + b1_M = e, \end{array} \tag{3}$$

where $V = D^{-1}$ is defined in the previous section. Using the Karush-Kuhn-Tucker (KKT) optimality conditions one can show that the solution in the dual can be obtained by solving the following linear system of equations (see [6])

$$\Big(I_M - (\gamma D^{-1} - \rho G)M_S\Omega\Big)\alpha = \rho M_S^Ty \tag{4}$$

where $M_S = I_M - (1/c)1_M1_M^T(\gamma D^{-1} - \rho G)$, I_M is the $M \times M$ identity matrix and $c = 1_M^T(\gamma D^{-1} - \rho G)1_M$. Ω is defined as previously,

$$G = \left[\begin{array}{c|c} 0_{N\times N} & 0_{N\times N_L} \\ \hline 0_{N_L\times N} & I_{N_L} \end{array}\right],$$

and $y = [0, \ldots, 0, y_{N+1}, \ldots, y_M]$. The model selection is done by using an affine combination (where the weight coefficients are positive) of a Fisher criterion and classification accuracy for labeled data.

III. Twin support vector machine based classifier

Consider a given training dataset $\{x_i, y_i\}_{i=1}^N$ with input data $x_i \in \mathbb{R}^d$ and output data $y_i \in \{-1, 1\}$ which consists of two subsets $X^1 = [x_1^1, \ldots, x_{n_1}^1]$ and $X^2 = [x_1^2, \ldots, x_{n_2}^2]$ corresponding to points belonging to class 1 and -1 respectively. The linear twin support vector machine (TSVM) [8], seeks two non-parallel hyperplanes

$$f_1(x) = w_1^Tx + b_1 = 0, \quad f_2(x) = w_2^Tx + b_2 = 0$$

such that each hyperplane is closest to the data points of one class and farthest from the data points of the other class. Here $w_1, w_2 \in \mathbb{R}^d$ and $b_1, b_2 \in \mathbb{R}$. In linear TSVM one has to solve the following optimization problems [8]:

$$\text{TWSVM1} \quad \begin{array}{ll} \min\limits_{w_1,b_1,\xi} & \frac{1}{2}\|X^1w_1 + b_11_{n_1}\|^2 + c_11_{n_2}^T\xi \\ \text{subject to} & -(X^2w_1 + b_11_{n_2}) + \xi \geq 1_{n_2}, \end{array}$$

$$\text{TWSVM2} \quad \begin{array}{ll} \min\limits_{w_2,b_2,\xi} & \frac{1}{2}\|X^2w_2 + b_21_{n_2}\|^2 + c_21_{n_1}^T\xi \\ \text{subject to} & (X^1w_2 + b_21_{n_1}) + \xi \geq 1_{n_1}. \end{array}$$

The vector of all ones with size j is denoted by 1_j. Introducing Lagrange multipliers α and β will lead to the following dual quadratic programming problems:

$$\begin{array}{ll} \min\limits_{\alpha} & 1_{n_2}^T\alpha - \frac{1}{2}\alpha^TG(H^TH)^{-1}G^T\alpha \\ \text{subject to} & 0 \leq \alpha \leq c_11_{n_2}, \end{array}$$

and

$$\min_{\beta} \quad 1_{n_1}^T\beta - \frac{1}{2}\beta^T R(Q^TQ)^{-1}R^T\beta$$
$$\text{subject to} \quad 0 \le \beta \le c_2 1_{n_1},$$

where $G = [X^2, 1_{n_2}]$, $H = [X^1, 1_{n_1}]$, $R = [X^1, 1_{n_1}]$ and $Q = [X^2, 1_{n_2}]$. $\alpha \in \mathbb{R}^{n_2}$ and $\beta \in \mathbb{R}^{n_1}$ are Lagrange multipliers. A Geometric interpretation of TSVM is given in Fig. 1. The class membership of a new data point is based on its proximity to the obtained two non-parallel hyperplanes. After completing the training stage and obtaining two non-parallel hyperplanes, the label of an unseen test point x^* is determined depending on the perpendicular distances of the test point from the hyperplanes:

$$\text{Label}(x^*) = \arg\min_{k=1,2}\{d_k(x^*)\}, \tag{5}$$

where

$$d_k(x^*) = \frac{|w_k^T x^* + b_k 1_n|}{\|w_k\|_2}, k = 1, 2.$$

The least squares version of TSVM which is termed as LSTSVM proposed in [12], for the linear kernel, consists of the following optimization problems:

$$\text{LSTSVM1} \quad \min_{w_1,b_1,\xi} \quad \frac{1}{2}\|X^1w_1 + b_1 1_{n_1}\|^2 + \frac{c_1}{2}\xi^T\xi$$
$$\text{subject to} \quad -(X^2w_1 + b_1 1_{n_2}) + \xi = 1_{n_2}$$

$$\text{LSTSVM2} \quad \min_{w_2,b_2,\xi} \quad \frac{1}{2}\|X^2w_2 + b_2 1_{n_2}\|^2 + \frac{c_2}{2}\xi^T\xi$$
$$\text{subject to} \quad (X^1w_2 + b_2 1_{n_1}) + \xi = 1_{n_1}.$$

Here the inequality constraints are replaced with equality constraints and as a consequence the loss function is also changed to a linear squares loss function. Considering the first optimization problem (LSTSVM1), substituting the equality constraint into the objective function and setting the gradient with respect to w_1 and b_1 will lead to a system of linear equations to be solved. The same approach can be applied for second optimization problem LSTSVM2 for obtaining w_2 and b_2. Finally one has to solve the following linear systems:

$$\begin{bmatrix} w_1 \\ b_1 \end{bmatrix} = -(G^TG + \frac{1}{c_1}H^TH)^{-1}G^T 1_{n_2},$$

and

$$\begin{bmatrix} w_2 \\ b_2 \end{bmatrix} = (H^TH + \frac{1}{c_2}G^TG)^{-1}H^T 1_{n_1},$$

where G and F are defined as previously. As it can been seen, in contrast with classical support vector machines introduced in [16] the TWSVM formulation [8] does not take the structural risk minimization into account. Therefore the authors in [9] proposed an improved version of TWSVM, called TBSVM, by adding a regularization term in the objective function aiming at minimizing the structural risk by maximizing the margin. In their regularization term, the bias term is also penalized, but this will not affect the result significantly and will change the optimization problem slightly. From a geometric point of view it is sufficient to penalize the w vector in order to maximize the margin.

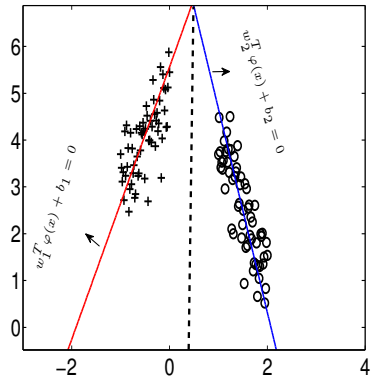


Fig. 1. Geometric interpretation of least squares twin support vector machines. Data points of class I and II are marked by plus and circle signs respectively. The non-parallel hyperplanes $w_1^T\varphi(x) + b_1 = 0$ and $w_2^T\varphi(x) + b_2 = 0$, which are obtained by solving the above optimization problems, are depicted by solid lines. The obtained decision boundary is shown by the dashed line.

For a nonlinear kernel, TSVM, LSTSVM and TBSVM methods utilize the kernel generated surface to construct the optimization problems (see the linked references for more details). As opposed to these approaches, in our formulation for semi-supervised learning the burden of designing another two optimization formulations, when the nonlinear kernel is used, is removed by applying the Mercer's theorem and kernel trick directly. The optimal representations of the models are also shown in this way.

IV. Non-parallel semi-supervised KSC

Suppose the training data set $\mathcal{X}$ consists of M data points and is defined as follows:

$$\mathcal{X} = \{\underbrace{x_1, ..., x_N}_{\substack{Unlabeled \\ (\mathcal{X}_U)}}, \underbrace{x_{N+1}, .., x_{N+\ell_1}}_{\substack{Labeled\,with\,(+1) \\ (\mathcal{X}_{L_1})}}, \underbrace{x_{N+\ell_1+1}, .., x_{N+\ell_1+\ell_2}}_{\substack{Labeled\,with\,(-1) \\ (\mathcal{X}_{L_2})}}\}$$

where $\{x_i\}_{i=1}^M \in \mathbb{R}^d$. Let us decompose the training data into unlabeled and labeled parts as $\mathcal{X} = \mathcal{X}_U \cup \mathcal{X}_{L_1} \cup \mathcal{X}_{L_2}$ where subsets $\mathcal{X}_U$, $\mathcal{X}_{L_1}$ and $\mathcal{X}_{L_2}$ consisting of N_U unlabeled samples, N_{L1} samples of class I and N_{L2} samples of II respectively. Note that the total number of samples is $M = N_U + N_{L1} + N_{L2}$. The target values are denoted by set $\mathcal{Y}$ which consists of binary labels:

$$\mathcal{Y} = \{\underbrace{+1, \ldots, +1}_{y^1}, \underbrace{-1, \ldots, -1}_{y^2}\}.$$

The same decomposition procedure is applied for the available target values i.e. $\mathcal{Y} = y^1 \cup y^2$ where y^1 and y^2 consist of labels of the samples from class I and II respectively.

We seek two non-parallel hyperplanes

$$f_1(x) = w_1^T\varphi(x) + b_1 = 0, \quad f_2(x) = w_2^T\varphi(x) + b_2 = 0$$

where each one is as close as possible to the points of its own class and as far as possible from the data of the other class.

Remark 4.1: In general if one uses a nonlinear feature map $\varphi(\cdot)$, obviously two non-parallel hyper-surfaces will be obtained. But in the rest of this paper, the term "hyperplane" is used.

A. Primal-Dual Formulation of the Method

We formulate a non-parallel semi-supervised KSC, in the primal, as the following two optimization problems:

$$\begin{aligned}\min_{w_1,b_1,e,\eta,\xi} \quad & \frac{1}{2}w_1^T w_1 + \frac{\gamma_1}{2}\eta^T\eta + \frac{\gamma_2}{2}\xi^T\xi - \frac{\gamma_3}{2}e^T D^{-1} e \\ \text{subject to} \quad & w_1^T\varphi(x_i) + b_1 = \eta_i,\ \forall x_i \in I \\ & y_i^2\Big[w_1^T\varphi(x_i) + b_1\Big] + \xi_i = 1,\ \forall x_i \in II \\ & w_1^T\varphi(x_i) + b_1 = e_i,\ \forall x_i \in \mathcal{X}\end{aligned} \tag{6}$$

where γ_1, γ_2 and $\gamma_3 \in \mathbb{R}^+, b_1 \in \mathbb{R}$, $\eta \in \mathbb{R}^{N_{L1}}$, $\xi \in \mathbb{R}^{N_{L2}}$, $e \in \mathbb{R}^M$, $w_1 \in \mathbb{R}^h$. $\varphi(\cdot) : \mathbb{R}^d \to \mathbb{R}^h$ is the feature map and h is the dimension of the feature space.

$$\begin{aligned}\min_{w_2,b_2,e,\rho,\nu} \quad & \frac{1}{2}w_2^T w_2 + \frac{\gamma_4}{2}\rho^T\rho + \frac{\gamma_5}{2}\nu^T\nu - \frac{\gamma_6}{2}e^T D^{-1} e \\ \text{subject to} \quad & w_2^T\varphi(x_i) + b_2 = \rho_i,\ \forall x_i \in II \\ & y_i^1\Big[w_2^T\varphi(x_i) + b_2\Big] + \nu_i = 1,\ \forall x_i \in I \\ & w_2^T\varphi(x_i) + b_2 = e_i,\ \forall x_i \in \mathcal{X}\end{aligned} \tag{7}$$

where γ_4, γ_5 and $\gamma_6 \in \mathbb{R}^+$. $b_2 \in \mathbb{R}$, $\rho \in \mathbb{R}^{N_{L2}}$, $\nu \in \mathbb{R}^{N_{L1}}$, $e \in \mathbb{R}^M$, $w_2 \in \mathbb{R}^h$. $\varphi(\cdot)$ is defined as previously.

The intuition for the above formulation can be expressed as follows. Consider optimization problem (6), the first constraint is the sum of squared distances of the points in class I from the first hyperplane i.e. $f_1(x)$ and minimizing this distance will make $f_1(x)$ to be located close to points of class I. The second constraint will push $f_1(x)$ away from data points of class II (the distance of $f_1(x)$ from the points of class II should be at least 1). The third constraint is part of the core model (KSC). A similar argument can be made for the second optimization problem (7). By solving optimization problems (6) and (7) one can obtain two non-parallel hyperplanes where each one is surrounded by the data points of the corresponding cluster (class). Let us assume that class I and II consist of samples with targets (+1) and (-1) respectively. Then one can manipulate the objective function of the above optimization problems and rewrite them in primal as follows:

$$\begin{aligned}\min_{w_1,b_1,e} \quad & \frac{1}{2}w_1^T w_1 + \frac{\gamma_1}{2}e^T A e + \frac{\gamma_2}{2}(S_1 + Be)^T(S_1 + Be) \\ & - \frac{\gamma_3}{2}e^T D^{-1} e \\ \text{subject to} \quad & w_1^T\varphi(x_i) + b_1 = e_i,\ \forall x_i \in \mathcal{X}\end{aligned} \tag{8}$$

$$\begin{aligned}\min_{w_2,b_2,e} \quad & \frac{1}{2}w_2^T w_2 + \frac{\gamma_4}{2}e^T B e + \frac{\gamma_5}{2}(S_2 - Ae)^T(S_2 - Ae) \\ & - \frac{\gamma_6}{2}e^T D^{-1} e \\ \text{subject to} \quad & w_2^T\varphi(x_i) + b_2 = e_i,\ \forall x_i \in \mathcal{X},\end{aligned} \tag{9}$$

where

$$A = \left[\begin{array}{c|c|c} 0_{N_U\times N_U} & 0_{N_U\times N_{L1}} & 0_{N_U\times N_{L2}} \\ \hline 0_{N_{L1}\times N_U} & I_{N_{L1}} & 0_{N_{L1}\times N_{L2}} \\ \hline 0_{N_{L2}\times N_U} & 0_{N_{L2}\times N_{L1}} & 0_{N_{L2}\times N_{L2}} \end{array}\right], \tag{10}$$

$$B = \left[\begin{array}{c|c|c} 0_{N_U\times N_U} & 0_{N_U\times N_{L1}} & 0_{N_U\times N_{L2}} \\ \hline 0_{N_{L1}\times N_U} & 0_{N_{L1}\times N_{L1}} & 0_{N_{L1}\times N_{L2}} \\ \hline 0_{N_{L2}\times N_U} & 0_{N_{L2}\times N_{L1}} & I_{N_{L2}} \end{array}\right] \tag{11}$$

$$S_1 = B1_M, \quad S_2 = A1_M. \tag{12}$$

Here 1_M is vector of all ones with size M. $I_{N_{L1}}$ and $I_{N_{L2}}$ are identity matrix of size $N_{L1} \times N_{L1}$ and $N_{L2} \times N_{L2}$ respectively. One can further simplify the objective of the (8) and (9) and rewrite them as follows:

$$\begin{aligned}\min_{w_1,b_1,e} \quad & \frac{1}{2}w_1^T w_1 - \frac{1}{2}e^T(\gamma_3 D^{-1} - \gamma_1 A - \gamma_2 B)e + \\ & \frac{\gamma_2}{2}(S_1^T S_1 + 2S_1^T e) \\ \text{subject to} \quad & \Phi w_1 + b_1 1_M = e,\end{aligned} \tag{13}$$

$$\begin{aligned}\min_{w_2,b_2,e} \quad & \frac{1}{2}w_2^T w_2 - \frac{1}{2}e^T(\gamma_6 D^{-1} - \gamma_4 B - \gamma_5 A)e + \\ & \frac{\gamma_5}{2}(S_2^T S_2 - 2S_2^T e) \\ \text{subject to} \quad & \Phi w_2 + b_2 1_M = e,\end{aligned} \tag{14}$$

Here $\Phi = [\varphi(x_1), \ldots, \varphi(x_M)]^T$.

Lemma 4.1: Given a positive definite kernel function $K : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ with $K(x, z) = \varphi(x)^T\varphi(z)$ and regularization constants $\gamma_1, \gamma_2, \gamma_3 \in \mathbb{R}^+$, the solution to (13) is obtained by solving the following dual problem:

$$(V_1 C_1 \Omega - I_M)\alpha = \gamma_2 C_1^T S_1 \tag{15}$$

where $V_1 = \gamma_3 D^{-1} - \gamma_1 A - \gamma_2 B$ and $C_1 = I_M - (1/1_M^T V_1 1_M)1_M 1_M^T V_1$. $\alpha = [\alpha_1, \ldots, \alpha_M]^T$ and $\Omega = \Phi\Phi^T$ is the kernel matrix.

Proof: The Lagrangian of the constrained optimization problem (13) becomes

$$\mathcal{L}(w_1, b_1, e, \alpha) = \frac{1}{2}w_1^T w_1 - \frac{1}{2}e^T(\gamma_3 D^{-1} - \gamma_1 A - \gamma_2 B)e + \frac{\gamma_2}{2}(S_1^T S_1 + 2S_1^T e) + \alpha^T\Big(e - \Phi w_1 - b_1 1_M\Big)$$

where α is the vector of Lagrange multipliers. Then the Karush-Kuhn-Tucker (KKT) optimality conditions are as

follows,

$$\begin{cases} \frac{\partial \mathcal{L}}{\partial w_1} = 0 \rightarrow \ w_1 = \Phi^T \alpha, \\ \frac{\partial \mathcal{L}}{\partial b_1} = 0 \rightarrow \ 1_M^T \alpha = 0, \\ \frac{\partial \mathcal{L}}{\partial e} = 0 \rightarrow \ \alpha = (\gamma_3 D^{-1} - \gamma_1 A - \gamma_2 B)e - \gamma_2 S_1, \\ \frac{\partial \mathcal{L}}{\partial \alpha} = 0 \rightarrow \ e = \Phi w_1 + b_1 1_M. \end{cases} \tag{16}$$

Elimination of the primal variables w_1, e and making use of Mercer's Theorem, will result in the following equation

$$V_1 \Omega \alpha + b_1 V_1 1_M = \alpha + \gamma_2 S_1 \tag{17}$$

where $V_1 = \gamma_3 D^{-1} - \gamma_1 A - \gamma_2 B$. From the second KKT optimality condition and (17), the bias term becomes:

$$b_1 = (1/1_M^T V_1 1_M)(1_M^T \gamma_2 S_1 - 1_M^T V_1 \Omega \alpha). \tag{18}$$

Substituting the obtained expression for the bias term b_1 into (17) along with some algebraic manipulation one can obtain the solution in dual as the following linear system:

$$\gamma_2 \left(I_M - \frac{V_1 1_M 1_M^T}{1_M^T V_1 1_M} \right) S_1 = V_1 \left(I_M - \frac{1_M 1_M^T V_1}{1_M^T V_1 1_M} \right) \Omega \alpha - \alpha.$$

■

Lemma 4.2: Given a positive definite kernel function $K : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ with $K(x, z) = \varphi(z)^T \varphi(z)$ and regularization constants $\gamma_4, \gamma_5, \gamma_6 \in \mathbb{R}^+$, the solution to (14) is obtained by solving the following dual problem:

$$(I_M - V_2 C_2 \Omega)\beta = \gamma_5 C_2^T S_2, \tag{19}$$

where $V_2 = \gamma_6 D^{-1} - \gamma_4 B - \gamma_5 A$, $\beta = [\beta_1, \ldots, \beta_M]^T$ are the Lagrange multipliers and $C_2 = I_M - (1/1_M^T V_2 1_M) 1_M 1_M^T V_2$. Ω and I_M are defined as previously.

Proof: The Lagrangian of the constrained optimization problem (14) becomes

$$\mathcal{L}(w_2, b_2, e, \beta) = \frac{1}{2} w_2^T w_2 - \frac{1}{2} e^T (\gamma_6 D^{-1} - \gamma_4 B - \gamma_5 A) e + \frac{\gamma_5}{2} (S_2^T S_2 - 2 S_2^T e) + \beta^T \left(e - \Phi w_2 - b_2 1_M \right)$$

where β are the Lagrange multipliers. Then the Karush-Kuhn-Tucker (KKT) optimality conditions are as follows,

$$\begin{cases} \frac{\partial \mathcal{L}}{\partial w_2} = 0 \rightarrow \ w_2 = \Phi^T \beta, \\ \frac{\partial \mathcal{L}}{\partial b_2} = 0 \rightarrow \ 1_M^T \beta = 0, \\ \frac{\partial \mathcal{L}}{\partial e} = 0 \rightarrow \ \beta = (\gamma_6 D^{-1} - \gamma_4 B - \gamma_5 A) e + \gamma_5 S_2, \\ \frac{\partial \mathcal{L}}{\partial \beta} = 0 \rightarrow \ e = \Phi w_2 + b_2 1_M. \end{cases} \tag{20}$$

After elimination of the primal variables w_2, e and making use of Mercer's Theorem, one can obtain the following equation

$$V_2 \Omega \beta + b_2 V_2 1_M = \beta - \gamma_5 S_2 \tag{21}$$

where $V_2 = \gamma_6 D^{-1} - \gamma_4 B - \gamma_5 A$. From (21) and the second KKT optimality condition, the bias term becomes:

$$b_2 = (1/1_M^T V_2 1_M)(-1_M^T \gamma_2 S_2 - 1_M^T V_2 \Omega \beta). \tag{22}$$

Substituting the obtained expression for the bias term b_2 into (21) along with some algebraic manipulation will result in the following linear system:

$$\gamma_5 \left(I_M - \frac{V_2 1_M 1_M^T}{1_M^T V_2 1_M} \right) S_2 = \beta - V_2 \left(I_M - \frac{1_M 1_M^T V_2}{1_M^T V_2 1_M} \right) \Omega \beta.$$

■

B. Different approach to reach the same formulation

In this section we provide an alternative approach to produce a non-parallel semi-supervised formulation based on KSC. In addition we show that this approach will lead to the same optimization problems (13) and (14) obtained in previous subsection.

Thus in order to produce non-parallel hyperplanes, let us directly start from (3) and formulate the following optimization problems in the primal:

$$\begin{aligned} \min_{w_1, b_1, e} \quad & \frac{1}{2} w_1^T w_1 - \frac{\gamma_3}{2} e^T V e + \frac{\gamma_2}{2} \sum_{t \in II} (e_t - y_t)^2 + \\ & \frac{\gamma_1}{2} \sum_{t \in I} e_t^2 \\ \text{subject to} \quad & \Phi w_1 + b_1 1_M = e \end{aligned} \tag{23}$$

$$\begin{aligned} \min_{w_2, b_2, e} \quad & \frac{1}{2} w_2^T w_2 - \frac{\gamma_6}{2} e^T V e + \frac{\gamma_5}{2} \sum_{t \in I} (e_t - y_t)^2 + \\ & \frac{\gamma_4}{2} \sum_{t \in II} e_t^2 \\ \text{subject to} \quad & \Phi w_2 + b_2 1_M = e, \end{aligned} \tag{24}$$

where I and II indicate the two classes. Let us rewrite the primal optimization problems (23) and (24) in matrix form as follows:

$$\begin{aligned} \min_{w_1, b_1, e} \quad & \frac{1}{2} w_1^T w_1 - \frac{\gamma_3}{2} e^T V e + \frac{\gamma_2}{2} (e^T B e - 2 y_1 e + y_1^T y_1) \\ & + \frac{\gamma_1}{2} e^T A e \\ \text{subject to} \quad & \Phi w_1 + b_1 1_M = e \end{aligned} \tag{25}$$

$$\begin{aligned} \min_{w_2, b_2, e} \quad & \frac{1}{2} w_2^T w_2 - \frac{\gamma_6}{2} e^T V e + \frac{\gamma_5}{2} (e^T A e - 2 y_2^T e + y_2^T y_2) \\ & + \frac{\gamma_4}{2} e^T B e \\ \text{subject to} \quad & \Phi w_2 + b_2 1_M = e, \end{aligned} \tag{26}$$

where A and B are defined in (10) and (11), $y_1 = [0_{N_U}, 0_{N_{L1}}, -1_{N_{L2}}]$ and $y_2 = [0_{N_U}, 1_{N_{L1}}, 0_{N_{L2}}]$.

Therefore optimization problems (25) and (26) are equivalent to (13) and (14) respectively.

V. Model selection

The performance of the non-parallel semi-KSC model depends on the choice of the tuning parameters. In this paper for all real experiments the Gaussian RBF kernel is used. The optimal values of $\{\gamma_i\}_{i=1}^{6}$ and the kernel parameter σ are obtained by evaluating the performance of the model (classification accuracy) on the validation set using a meaningful grid search over the parameters. In our proposed algorithm, we set $\gamma_1 = \gamma_4$, $\gamma_2 = \gamma_5$ and $\gamma_3 = \gamma_6$ to reduce the computational complexity of parameter selection.

Noting that both labeled and unlabeled data points are involved in the learning process, it is natural to have a model selection criterion that makes use of both labeled and unlabeled data points. Therefore one may combine two criterion which one of them is designed for evaluating the performance of the model on the unlabeled data points (evaluation of clustering results) and the other one will maximize the classification accuracy.

A common approach for evaluation of clustering results is to use cluster validity indices [17]. Any internal clustering validity approach such as Silhouette index, Fisher index or Davies-Bouldin index (DB) can be utilized. In this paper Fisher criterion [15], [6] is utilized to assess the clustering results. This criterion measures how localized the cluster appear in the out-of-sample solution by minimizing the within-cluster scatter while maximizing between-cluster scatter.

The proposed model selection criterion can be expressed as follows:

$$\underset{\gamma_1,\gamma_2,\gamma_3,\sigma}{\operatorname{argmax}} = \kappa\,\mathrm{F}(\gamma_1,\gamma_2,\gamma_3,\sigma) + (1-\kappa)\mathrm{Acc}(\gamma_1,\gamma_2,\gamma_3,\sigma)$$

which is an affine combination of Fisher criterion (F) and classification accuracy (Acc). $\kappa \in [0,1]$ is a user-defined parameter that controls the trade off between the importance given to unlabeled and labeled samples. In case few labeled samples are available one may give more weight to Fisher criterion and vice versa. After completing the training stage the labels of unseen test points $\mathcal{X}_{test} = \{x_{test}^1, \ldots, x_{test}^n\}$ are determined by using (5) where

$$d_k(\mathcal{X}_{test}) = \frac{|\Phi_{test} w_k + b_k 1_n|}{\|w_k\|_2}, k = 1, 2. \qquad (27)$$

Here $\Phi_{test} = [\varphi(x_{test}^1), \ldots, \varphi(x_{test}^n)]^T$. The procedure of the proposed non-parallel semi-supervised classification is outlined in Algorithm 1.

Algorithm 1 Non-parallel semi-supervised KSC

Input: Training data set $\mathcal{X}$, labels $\mathcal{Y}$, the tuning parameters $\{\gamma_i\}_{i=1}^{3}$, the kernel bandwidth σ and the test set $\mathcal{X}_{test}$.
Output: Class membership of test data.

1: Solve the dual linear system (15) to obtain α and compute the bias term b_1 by (18). Therefore the first hyperplane (hypersurface) can be constructed.
2: Solve the dual linear system (19) to obtain β and compute the bias term b_2 by (22). Therefore the second hyperplane (hypersurface) can be constructed.
3: Compute the class membership of test data points using (5) and (27).

VI. Numerical Experiments

In this section some experimental results on synthetic and real datasets are given. The synthetic problem consist of four Gaussians with some overlap. The full dataset includes 200 data points. Each one of the training and validation sets consist of 100 points randomly selected form the entire dataset. Artificially binary labels are introduced for eight points.

The performance of the Semi-KSC [6] and the proposed method in this paper when a linear kernel is used are shown in Fig. 1. Due to the ability of the method to produce two non-parallel hyperplanes the data points are almost correctly classified whereas Semi-KSC is not able to do the task well in this case. This example can motivate the use of non-parallel Semi-KSC over Semi-KSC.

The performance of the method is also tested on some of the benchmark datasets for semi-supervised learning described in [18]. The benchmark consists of four data sets as shown in Table 1. The first two i.e. `g241c` and `g241d`, which consist of 1500 data points with dimensionality of 241, were artificially created. The other two datasets `BCI` and `Text` were derived from real data. All datasets have binary labels. `BCI` has 400 data points and dimension 117. `Text` includes 1500 data points with dimensionality 11960.

For each data set, 12 splits into labeled points and remaining unlabeled points is already provided (each split contains at least one point of each class). The tabulated results indicate the variability with respect to these 12 splits. In order to have a fair comparison with the recorded results in [6], for each split a training set of $N_{tr} = 150$ unlabeled samples are randomly selected for `BCI` data set and for all the other data sets $N_{tr} = 600$. The same number of unlabeled samples used in training sets are drawn at random to form the unlabeled samples of the validation sets. Among the labeled data points, 70% is used for training and 30% for the validation sets.

The result of the proposed method (NP-Semi-KSC) is compared with that of Semi-KSC, Laplacian SVM (LapSVM) [4] and its recent version LapSVMp [19] recorded in [6] over the datasets mentioned. When few labeled data points are available the proposed method shows a comparable result with respect to other methods. But as the number of labeled data points increases NP-Semi-KSC outperforms in most cases the other methods.

In Fig. 2, the comparison between NP-Semi-KSC and Semi-KSC is shown for three real data sets taken from UCI machine learning repository [20]. We performed two cases corresponding to different percentages of the labeled data points used in the learning process. The results shown in Fig. 2, are obtained by averaging over 10 simulation runs for different κ values when RBF kernel is used. In both cases, NP-Semi-KSc obtains the maximum accuracy over the whole range of κ values for these three datasets. The obtained results when linear kernel is used, are depicted in Fig. 4.

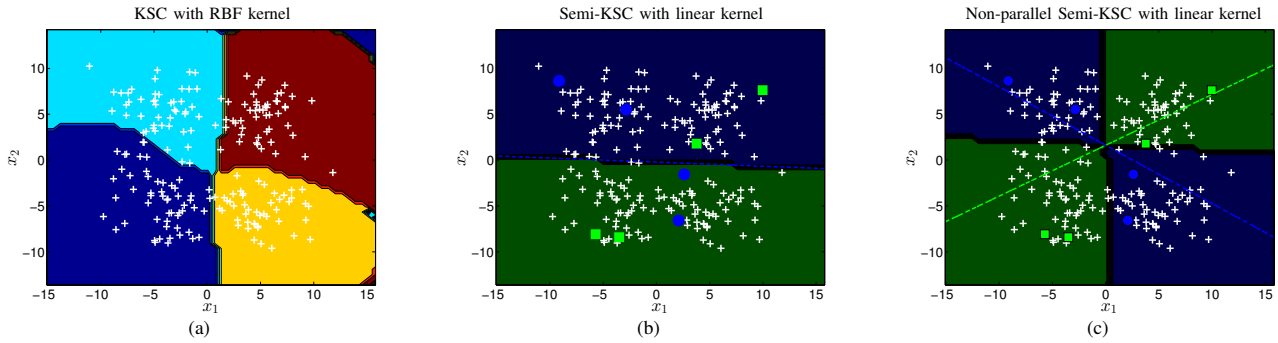


Fig. 2. Toy Problem-Four Gaussians with some overlap. The training and validation parts consist of $N_{tr} = 100$ and $N_{val} = 100$ unlabeled data points respectively. The labeled data points of two classes are depicted by the blue squares and green circles. (a) Result of kernel spectral clustering (completely unsupervised). (b): Result of semi-supervised kernel spectral clustering when linear kernel is used. The separating hyperplane is shown by blue dashed line. (c): Result of the proposed non-parallel semi-supervised KSC when linear kernel is used. Two non- parallel hyperplanes are depicted by blue and green dashed lines.

TABLE I

AVERAGE MISCLASSIFICATION TEST ERROR $\times 100\%$. THE CALCULATION OF THE TEST ERROR IS DONE BY EVALUATING THE METHODS ON THE FULL DATA SETS. TWO CASES FOR THE LABELED DATA SIZE ARE CONSIDERED (I.E. # LABELED DATA POINTS=10 AND 100). IN THE CASE OF 10 LABELED DATA POINTS, THE PERFORMANCE OF THE PROPOSED METHOD IS COMPARABLE TO THAT OF THE OTHER METHODS. WHEN 100 LABLED DATA POINTS ARE USED, THE PROPOSED METHOD SHOWS A BETTER PERFROMANCE COMPARED TO LAPSVM [4], LAPSVMP [19] AND SEMI-KSC [6].

# of Labeled data	**Method**	g241c	g241d	BCI	Text
10	LapSVM	0.48 ± 0.02	0.42 ± 0.03	0.48 ± 0.03	0.37 ± 0.04
	LapSVMp	0.49 ± 0.01	0.43 ± 0.03	0.48 ± 0.02	0.40 ± 0.05
	Semi-KSC	$\mathbf{0.42 \pm 0.03}$	0.43 ± 0.04	$\mathbf{0.46 \pm 0.03}$	$\mathbf{0.29 \pm 0.06}$
	NP-Semi-KSC	0.44 ± 0.03	$\mathbf{0.41 \pm 0.02}$	0.47 ± 0.03	0.40 ± 0.05
100	LapSVM	0.40 ± 0.06	0.31 ± 0.03	0.37 ± 0.04	0.27 ± 0.02
	LapSVMp	0.36 ± 0.07	0.31 ± 0.02	0.32 ± 0.02	0.32 ± 0.02
	Semi-KSC	0.29 ± 0.05	0.28 ± 0.05	0.28 ± 0.02	$\mathbf{0.22 \pm 0.02}$
	NP-Semi-KSC	$\mathbf{0.23 \pm 0.01}$	$\mathbf{0.26 \pm 0.02}$	$\mathbf{0.26 \pm 0.01}$	0.23 ± 0.02

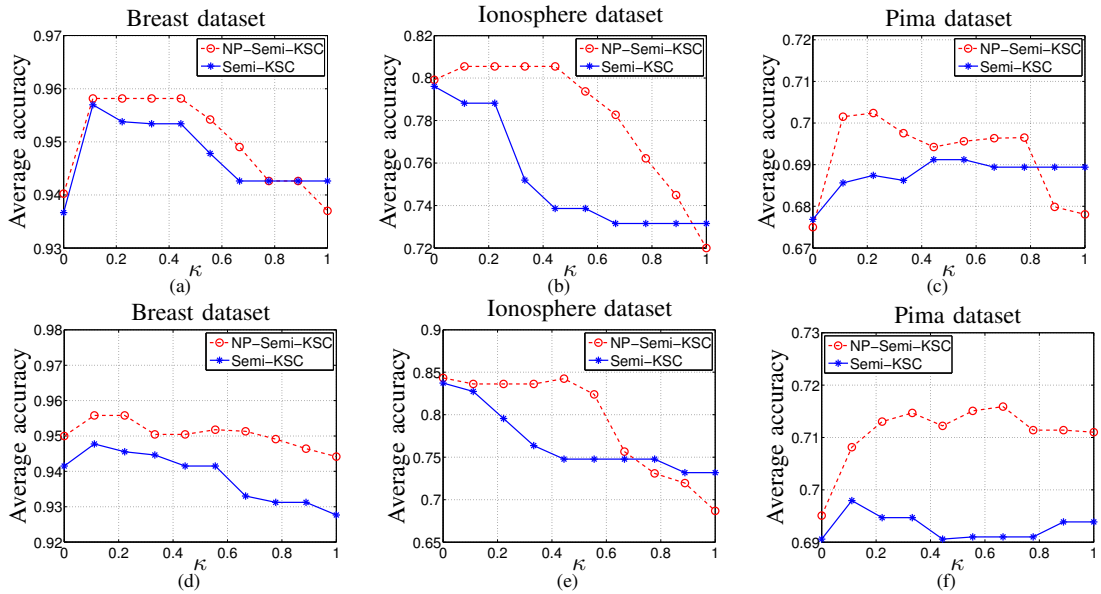


Fig. 3. Obtained average accuracy over 10 simulation runs, on the unseen test set, using semi-supervised KSC and Non-parallel semi-KSC approaches for three real datasets (Ionosphere, Pima and Breast) taken from UCI benchmark repository when RBF kernel is used. Two scenarios are studied. In the first scenario, the training points consist of 5% labeled data points as well as 10% unlabeled data points of each class. The same percentages are used to construct the validation set. The remanings are used as test points. In the second scenario, the training points consist of 10% labeled data points as well as 10% unlabeled data points of each class. The same percentages are used to construct the validation set. The remanings are used as test points. (a),(b),(c): The obtained results for the first scenario. (d),(e),(f): The obtained results for the second scenario.

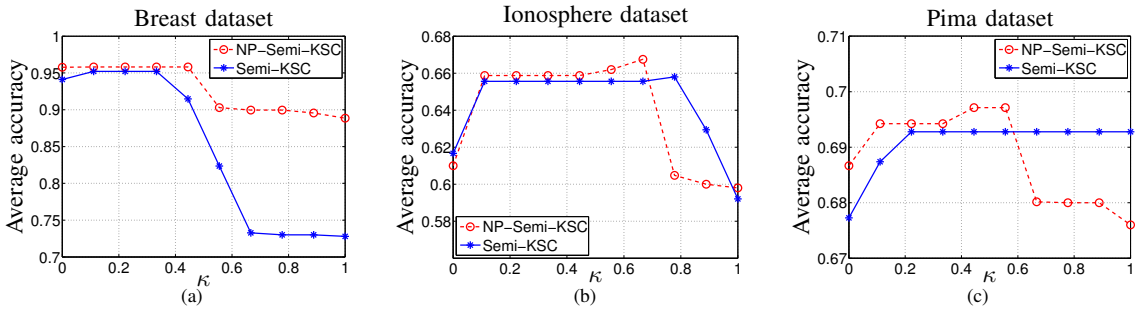


Fig. 4. Obtained average accuracy over 10 simulation runs, on the unseen test set, using semi-supervised KSC and Non-parallel semi-KSC approaches when *linear kernel* is used. The training points consist of 5% labeled data points as well as 10% unlabeled data points of each class. The same percentages are used to construct the validation set. The remanings are used as test points.

VII. Conclusions

In this paper, a non-parallel semi-supervised formulation based on kernel spectral clustering is developed. In theory Semi-KSC formulation is a special case of the proposed method when parameters of the new model are chosen appropriately. The validity and applicability of the proposed method is shown on synthetic examples as well as on real benchmark datasets. Due to the possibility of using different types of loss functions, alternative non-parallel semi-supervised classifiers can be considered in future work.

Acknowledgments

This work was supported by: • Research Council KUL: GOA/10/09 MaNet, PFV/10/002 (OPTEC), several PhD/postdoc & fellow grants • Flemish Government: ○ IOF: IOF/KP/SCORES4CHEM; ○ FWO: PhD/postdoc grants, projects: G.0320.08 (convex MPC), G.0558.08 (Robust MHE), G.0557.08 (Glycemia2), G.0588.09 (Brain-machine), G.0377.09 (Mechatronics MPC); G.0377.12 (Structured systems) research community (WOG: MLDM); ○ IWT: PhD Grants, projects: Eureka-Flite+, SBO LeCoPro, SBO Climaqs, SBO POM, O&O-Dsquare • Belgian Federal Science Policy Office: IUAP P7/ (DYSCO, Dynamical systems, control and optimization, 2012-2017) • IBBT • EU: ERNSI, FP7-EMBOCON (ICT-248940), FP7-SADCO (MC ITN-264735), ERC ST HIGHWIND (259 166), ERC AdG A-DATADRIVE-B • COST: Action ICO806: IntelliCIS • Contract Research: AMINAL • Other: ACCM. Johan Suykens is a professor at the KU Leuven, Belgium.

References

[1] S. Matthias, Learning with labeled and unlabeled data, *Inst. Adapt. Neural Comput.*, Univ. Edinburgh, Edinburgh, U.K., Tech. Rep., 2001.

[2] Z. Xiaojin, Semi-supervised learning literature survey, Comput. Sci., Univ. Wisconsin-Madison, Madison, WI, Tech. Rep. 1530, 2007 [Online]. Available: http://www.cs.wisc.edu/ jerryzhu/pub/ssl_survey.pdf

[3] M.M. Adankon, M. Cheriet and A. Biem, "Semisupervised Least Squares Support Vector Machine," *IEEE Transactions on Neural Networks,* vol. 20, no. 12, pp. 1858-70, 2009.

[4] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *J. Mach. Learn. Res.*, vol. 7, pp. 2399-2434, 2006.

[5] C. Alzate and J. A. K. Suykens, "Multiway spectral clustering with out-of-sample extensions through weighted kernel PCA," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 2, pp. 335-347, 2010.

[6] C. Alzate and J. A. K. Suykens, "A Semi-Supervised Formulation to Binary Kernel Spectral Clustering," *In Proc. of the 2012 IEEE World Congress on Computational Intelligence (IEEE WCCI/IJCNN 2012)*, Brisbane, Australia, pp. 1992-1999, 2012.

[7] O.L. Mangasarian and E.W. Wild, "Multisurface Proximal Support Vector Machine Classification via Generalized Eigenvalues," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 1, pp. 69-74, 2006.

[8] Jayadeva, R. Khemchandani and S. Chandra, "Twin Support Vector Machines for Pattern Classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 5, pp.905-910, 2007.

[9] Y.H. Shao, C.H. Zhang, X.B Wang, and N.Y. Deng, "Improvements on Twin Support Vector Machines," *IEEE Transactions on Neural Networks*, vol. 22, no. 6, pp. 962-968, 2011.

[10] J. A. K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, J. Vandewalle, *Least Squares Support Vector Machines*, World Scientific Pub. Co., Singapore, 2002.

[11] G. Fung and O.L. Mangasarian, "Proximal Support Vector Machine Classifiers," *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, Pages 77-86, 2001.

[12] M.A. Kumar, M. Gopal, "Least squares twin support vector machines for pattern classification," *Expert Systems with Applications,* 36, pp. 7535-7543, (2009).

[13] J.A.K. Suykens, & J. Vandewalle, Least squares support vector machine classifiers. *Neural Processing Letters*, 9 (3),pp. 293-300, (1999).

[14] J.A.K. Suykens, C. Alzate, K. Pelckmans, "Primal and dual model representations in kernel-based learning", *Statistics Surveys*, vol. 4, pp. 148-183, 2010.

[15] C. Bishop, *Pattern Recognition and Machine Learning.* Springer, 2006.

[16] V. Vapnik, *Statistical learning theory*, New York: Wiley, 1998.

[17] J.C. Bezdek and R.P. Nikhil, "Some new indexes of cluster validity", *IEEE Transactions on Systems, Man and Cybernetics*, 28(3), pp. 301-315, 1998.

[18] O. Chapelle, B. Schölkopf, and A. Zien, Eds., Semi-Supervised Learning. Cambridge, MA: MIT Press, 2006.

[19] S. Melacci and M. Belkin, "Laplacian support vector machines trained in the primal," *Journal of Machine Learning Research*, vol. 12, pp. 1149-1184, 2011.

[20] A. Frank and A. Asuncion. UCI machine learning repository. [http://archive.ics.uci.edu/ml]. Irvine, University of California, School of Information and Computer Science, 2010.