# On Discriminative Semi-Supervised Classification

**Fei Wang, Changshui Zhang**
[1]State Key Laboratory of Intelligent Technologies and Systems
Department of Automation, Tsinghua University, Beijing, China. 100084.
feiwang03@mails.thu.edu.cn, zcs@mail.tsinghua.edu.cn

### Abstract

The recent years have witnessed a surge of interests in semi-supervised learning methods. A common strategy for these algorithms is to require that the predicted data labels should be sufficiently smooth with respect to the intrinsic data manifold. In this paper, we argue that rather than penalizing the label smoothness, we can directly punish the discriminality of the classification function to achieve a more powerful predictor, and we derive two specific algorithms: *Semi-Supervised Discriminative Regularization* (*SSDR*) and *Semi-parametric Discriminative Semi-supervised Classification* (*SDSC*). Finally many experimental results are presented to show the effectiveness of our method.

## Introduction

In many practical pattern classification and data mining problems, the aquisition of sufficient labeled data is often expensive and/or time consuming. However, in many cases, large numbers of unlabeled data are far easier to obtain. For example, in text classification, one may have an easy access to a large database of documents (*e.g.* by crawling the web), but only a small part of them are classified by hand.

Consequently, *semi-supervised learning methods*, which aim to learn from both labeled and unlabeled data points, are proposed (Chapelle *et al*., 2006). Many semi-supervised learning algorithms have been proposed in the last decades, for example, the *generative model* based methods (Shahshahani & Landgrebe, 1994; Miller & Uyar, 1997; Nigam *et al*, 2000), *co-training* (Blum & Mitchell, 1998), *transductive SVM* (*TSVM*) (Joachims, 1999), and the approaches based on *statistical physics* theories (Getz, *et al*, 2005; Wang *et al*, 2007).

One basic assumption behind semi-supervised learning is the *cluster assumption* (Chapelle *et al*., 2006), which states that two points are likely to have the same class label if there is a path connecting them passing through the regions of high density only. *Zhou et al*. (Zhou *et al*, 2004) further explored the geometric intuition behind this assumption: (1) nearby points are likely to have the same label; (2) points on the same structure (such as a cluster or a submanifold) are likely to have the same label. Therefore, the learned classification function should be sufficiently smooth with to respect the intrinsic data manifold. Belkin *et al*. (Belkin *et al*., 2006) further proposed an elegant framework for semi-supervised learning, called *manifold regularization*, which learns a specific classification function by minimizing a cost composed of two terms, one is the structure loss and the other is a smoothness measure estimated from both labeled and unlabeled data. In such a way, the learned function would be sufficiently smooth while simultaneously hold a good generalization ability.

However, as the final goal of classification is to discriminate the data points from different classes, it is reasonable that the predicted labels vary smoothly with respect to the intrinsic geometric structure within the same class, but there should definitely be some discontinuities on the decision boundaries, *i.e.*, on the boundaries where the different classes are connected. Moreover, it would be more encouraged if the predicted labels of the data in different classes are more distinct.

Based on the above considerations, in this paper, we propose a novel strategy for semi-supervised classification, which does not require the classification function to be smooth, but to be discriminative. We first construct an unsupervised discriminative kernel based on *discriminant analysis* (Fukunaga, 1990), and then use it to derive two specific algorithms, *Semi-Supervised Discriminative Regularization* (*SSDR*) and *Semi-parametric Discriminative Semi-supervised Classification* (*SDSC*) to realize our strategy. Finally the experimental results on several benchmark data sets are presented to show the effectiveness of our methods.

The rest of this paper is organized as follows. In section 2 we will derive an unsupervised discriminative kernel. The main procedure of the *SSDR* and *SDSC* algorithms will be introduced in detail in section 3 and section 4. The experimental results will be presented in section 5, followed by the conclusions in section 6.

## A Discriminative Kernel

Given a set of data objects $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$ with $\mathbf{x}_i \in \mathbb{R}^D$, the goal of *discriminant analysis* (Fukunaga, 1990) is to learn a projection matrix $\mathbf{P} \in \mathbb{R}^{D\times d}$ with $d \ll D$, such that after projection the data set have a high within class similarity and between-class dissimilarity. Without the loss of

 The work is supported by NSFC (Grant No. 60721003, 60675009).

generality, we assume the data set has been centralized, *i.e.*, $\sum_i \mathbf{x}_i = 0$, then we can define the within-class, between-class and total scatter matrix as

$$\mathbf{S}_w = \sum_c \sum_{\mathbf{x}_i \in \pi_c} (\mathbf{x}_i - \mathbf{m}_c)(\mathbf{x}_i - \mathbf{m}_c)^T = \mathbf{X}\mathbf{L}_w\mathbf{X}^T \quad (1)$$

$$\mathbf{S}_b = \sum_c n_c \mathbf{m}_c \mathbf{m}_c^T = \mathbf{X}\mathbf{L}_b\mathbf{X}^T \quad (2)$$

$$\mathbf{S}_t = \sum_{i=1}^{n} \mathbf{x}_i\mathbf{x}_i^T + \lambda\mathbf{I}_n = \mathbf{X}\mathbf{X}^T + \lambda\mathbf{I}_n \quad (3)$$

where $\mathbf{m}_c = \frac{1}{n_c}\sum_{\mathbf{x}_i \in \pi_c} \mathbf{x}_i$ is the mean vector of the $c$-th class $\pi_c$, and the additional regularization term in $\mathbf{S}_t$ is used to improve the reliability of the estimation of $\mathbf{S}_t$ (Friedman, 1989). We can derive that the matrices

$$\mathbf{L}_w = \mathbf{I} - \mathbf{G}(\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T = \mathbf{I} - \mathbf{F}\mathbf{F}^T \quad (4)$$

$$\mathbf{L}_b = \mathbf{G}(\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T = \mathbf{F}\mathbf{F}^T \quad (5)$$

where $\mathbf{G} \in \mathbb{R}^{n\times C}$ is a dummy class indicator matrix with its $(i,j)$-th entry

$$G_{ij} = \begin{cases} 1, & if\ \mathbf{x}_i \in \pi_j \\ 0, & otherwise \end{cases} \quad (6)$$

and $\mathbf{F} = \mathbf{G}(\mathbf{G}^T\mathbf{G})^{-1/2}$ is the scaled partition matrix. Then a common criterion for learning an optimal subspace for discriminating the data from multiple classes is (Fukunaga, 1990) maximizing

$$\mathcal{J} = tr((\mathbf{P}^T\mathbf{S}_w\mathbf{P})^{-1}(\mathbf{P}^T\mathbf{S}_b\mathbf{P}))$$

which aims to maximize the between-class dissimilarity and within-class similarity simultaneously in the learned subspace. It can be easily verified that $\mathbf{S}_t = \mathbf{S}_w + \mathbf{S}_b$, then $\mathcal{J}$ can be rewritten in an equivalent from of(Fukunaga, 1990)

$$\mathcal{J} = tr((\mathbf{P}^T\mathbf{S}_t\mathbf{P})^{-1}(\mathbf{P}^T\mathbf{S}_b\mathbf{P})) \quad (7)$$

From the *representor theorem* (Schölkopf & Smola, 2002), the transformation matrix can be expressed as $\mathbf{P} = \mathbf{X}\mathbf{H}$ with some coefficient matrix $\mathbf{H} \in \mathbb{R}^{n\times d}$, then

$$\begin{aligned}
&\mathcal{J} = tr((\mathbf{P}^T\mathbf{S}_t\mathbf{P})^{-1}(\mathbf{P}^T\mathbf{S}_b\mathbf{P})) \\
&= tr((\mathbf{H}^T\mathbf{X}^T(\mathbf{X}\mathbf{X}^T + \lambda\mathbf{I}_n)\mathbf{X}\mathbf{H})^{-1}(\mathbf{H}^T\mathbf{X}^T\mathbf{X}\mathbf{L}_b\mathbf{X}^T\mathbf{X}\mathbf{H})) \\
&= tr((\mathbf{H}^T(\mathbf{K}^2 + \lambda\mathbf{K})\mathbf{H})^{-1}(\mathbf{H}^T\mathbf{K}\mathbf{L}_b\mathbf{K}\mathbf{H})) \\
&= tr((\mathbf{H}^T(\mathbf{K}^2 + \lambda\mathbf{K})\mathbf{H})^{-1}(\mathbf{H}^T\mathbf{K}\mathbf{F}\mathbf{F}^T\mathbf{K}\mathbf{H}))
\end{aligned}$$

Viewing $\mathcal{J}$ as a function of $\mathbf{H}$, we can easily derive that the maximization of $\mathcal{J}(\mathbf{H})$ is just a quotient trace optimization problem, whose optimal solution is obtained as $\mathbf{H}^* = [\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_k]$, where $\mathbf{v}_i$ is the eigenvector of the matrix $(\mathbf{K}^2 + \lambda\mathbf{K})^+\mathbf{K}\mathbf{F}\mathbf{F}^T\mathbf{K}$ corresponding to its $i$-th largest eigenvalue (Fukunaga, 1990), where $(\cdot)^+$ denotes the pseudo inverse of a matrix. Bringing $\mathbf{H}^*$ back to $\mathcal{J}$, we can obtain that (Fukunaga, 1990)

$$\begin{aligned}
\mathcal{J}(\mathbf{H}^*) &= tr((\mathbf{K}^2 + \lambda\mathbf{K})^+\mathbf{K}\mathbf{F}\mathbf{F}^T\mathbf{K}) \\
&= tr(\mathbf{F}^T\mathbf{K}(\mathbf{K}^2 + \lambda\mathbf{K})^+\mathbf{K}\mathbf{F}) \\
&= tr\left(\mathbf{F}^T\left(\mathbf{I}_n - \left(\mathbf{I}_n + \frac{1}{\lambda}\mathbf{K}\right)^{-1}\right)\mathbf{F}\right) \quad (8)
\end{aligned}$$

Since $tr(\mathbf{F}^T\mathbf{F})$ is a constant according to its definition, the maximization of $\mathcal{J}$ with respect to $\mathbf{F}$ is equivalent to minimize

$$\tilde{\mathcal{J}} = tr\left(\mathbf{F}^T\left(\mathbf{I}_n + \frac{1}{\lambda}\mathbf{K}\right)^{-1}\mathbf{F}\right) \quad (9)$$

Therefore, the value of $\tilde{\mathcal{J}}$ encodes the discriminality of the labeling $\mathbf{F}$, the smaller $\tilde{\mathcal{J}}$ is, the more discriminative $\mathbf{F}$ would be. Clearly, since $\mathbf{K}$ is a kernel, then

$$\mathbf{D} = \left(\mathbf{I}_n + \frac{1}{\lambda}\mathbf{K}\right)^{-1} \quad (10)$$

is also a kernel, and we call it *discriminative kernel* throughout the paper. We can see that in fact $\mathbf{D}$ is just the inverse of a regularized kernel.

## Discriminative Regularization for Semi-supervised Learning

Belkin *et al*. (Belkin *et al*., 2006) proposed an elegant framework for semi-supervised learning called *Manifold Regularization* (*MR*), which aims to solve for a classification function $f$ by minimizing

$$\hat{\mathcal{J}} = \sum_{i=1}^{l} c(y_i, f(\mathbf{x}_i)) + \gamma_1\|f\|_{\mathcal{H}}^2 + \gamma_2\|f\|_{\mathcal{I}}^2 \quad (11)$$

where $c(\cdot,\cdot)$ is some loss function (*e.g.* hinge loss or square loss), $\|f\|_{\mathcal{H}}$ measures the complexity of $f$ in the *Reproducing Kernel Hilbert Space* $\mathcal{H}$, and $\|f\|_{\mathcal{I}}$ measures the smoothness of $f$ with respect to the intrinsic data manifold.

The penalization of the smoothness of $f$ with respect to the data manifold is common in graph based semi-supervised learning methods. It is motivated by some intuitive assumptions of the data labels (Zhou *et al*, 2004): (1) nearby points tend to have the same label; (2) points on the same structure tend to have the same label. It might be reasonable for the data labels varying smoothly within the same class, however, there might be some label discontinuities between the data from different classes. Moreover, as the final goal of classification is just to discriminate the data from different classes, it would be more encouraged if the predicted labels of the data in different classes are more distinct.

Based on the above considerations, we propose to directly penalize the discriminality of the classification function instead of its smoothness in semi-supervised learning. More concretely, we can define some discriminative regularizer $\|f\|_{\mathcal{D}}$ and then solve for the optimal $f$ by minimizing

$$\eta = \sum_{i=1}^{l} c(y_i, f(\mathbf{x}_i)) + \gamma_1\|f\|_{\mathcal{H}}^2 + \gamma_2\|f\|_{\mathcal{D}}^2 \quad (12)$$

Let's first consider the two-class problem, then $y_i \in \{+1, -1\}$ for $1 \leqslant i \leqslant l$. We use $\mathbf{f} = [f_1, f_2, \cdots, f_n]^T$ to denote the predicted label vector of $\mathcal{X}$ using $f$, then according to the content in section 2, $\|f\|_{\mathcal{D}}$ can be approximated by

$$\|f\|_{\mathcal{D}}^2 = \mathbf{f}^T\mathbf{D}\mathbf{f} \quad (13)$$

By applying the *point cloud norm* introduced in (Sindhwani *et al*., 2005), we can define a new *Reproducing Kernel Hilbert Space* (*RKHS*) $\tilde{\mathcal{H}}$ with its induced inner product

$$\tilde{k}(\mathbf{x}_i, \mathbf{x}_j) = k(\mathbf{x}_i, \mathbf{x}_j) + \mathbf{k}_{\mathbf{x}_i}^T \left(\mathbf{I} + \frac{\gamma_2}{\gamma_1}\mathbf{D}\mathbf{K}\right)^{-1} \mathbf{D}\mathbf{k}_{\mathbf{x}_j} \quad (14)$$

where $k(\cdot,\cdot)$ is some conventional kernel function defined in an *RKHS* with its corresponding kernel matrix $\mathbf{K}$, which is positive semi-definite. $\mathbf{k}_{\mathbf{x}_i} = [k(\mathbf{x}_i, \mathbf{x}_n), \cdots, k(\mathbf{x}_i, \mathbf{x}_n)]^T$. Then Eq.(12) can be rewritten as

$$\eta = \sum_{i=1}^{l} c(y_i, f(\mathbf{x}_i)) + \gamma_1 \|f\|_{\tilde{\mathcal{H}}}^2 \quad (15)$$

Then from the *representation theorem* (Schölkopf & Smola, 2002), $\mathbf{f}$ can be expanded by

$$\mathbf{f}_l = \tilde{\mathbf{K}}_l \boldsymbol{\alpha}_l \quad (16)$$

where $\tilde{\mathbf{K}}_l \in \mathbb{R}^{l\times l}$ is defined on the labeled set with its $(i,j)$-th entry computed as in Eq.(14), $\boldsymbol{\alpha}_l \in \mathbb{R}^{l\times 1}$ is the expansion coefficient vector. Bringing Eq.(16) back into Eq.(15), and adopting the square loss, we can derive the following loss function

$$\eta = \frac{1}{l}\|\mathbf{y}_l - \tilde{\mathbf{K}}_l \boldsymbol{\alpha}_l\|^2 + \gamma_1 \boldsymbol{\alpha}_l^T \tilde{\mathbf{K}}_l \boldsymbol{\alpha}_l \quad (17)$$

where $\mathbf{y}_l = [y_1, y_2, \cdots, y_l]^T$. By setting $\frac{\partial \eta}{\partial \boldsymbol{\alpha}} = 0$, we can get that

$$\boldsymbol{\alpha}_l^* = (\tilde{\mathbf{K}}_l + \gamma_1 l \mathbf{I})^{-1} \mathbf{y}_l \quad (18)$$

For a new testing point $\mathbf{x}$, we can predict its label by

$$f(\mathbf{x}) = \sum_{i=1}^{l} \tilde{k}(\mathbf{x}, \mathbf{x}_i)\alpha_l(i) \quad (19)$$

where $\alpha_l(i)$ represents the $i$-th element of $\boldsymbol{\alpha}$.

For multi-class problems, we can apply the same trick in (Zhou *et al*, 2004), *i.e.*, define $\mathbf{F} \in \mathbb{R}^{n\times C}$ to be a class indicator matrix such that the label of $\mathbf{x}_i$ can be determined as $y_i = argmax_{j\leqslant C} F_{ij}$. That is, the label of $\mathbf{x}_i$ is determined a row vector $\mathbf{r}_i$, which corresponds to the $i$-th row of $\mathbf{F}$, and its final label is equal to the column index of the largest element of $\mathbf{r}_i$. Then the predicted label vector of a new testing point $\mathbf{x}$ can be determined by

$$\mathbf{r}_{\mathbf{x}} = \tilde{\mathbf{k}}_{\mathbf{x}}^T \boldsymbol{\Gamma} \quad (20)$$

where $\tilde{\mathbf{k}}_{\mathbf{x}} = [\tilde{k}(\mathbf{x}_1, \mathbf{x}), \cdots, \tilde{k}(\mathbf{x}_l, \mathbf{x})]^T$, $\boldsymbol{\Gamma} \in \mathbb{R}^{l\times C}$ is the expansion coefficient matrix calculated by

$$\boldsymbol{\Gamma} = (\tilde{\mathbf{K}}_l + \gamma_1 l \mathbf{I})^{-1} \mathbf{Y}_l \quad (21)$$

where $\mathbf{Y}_l \in \mathbb{R}^{l\times C}$ is the initial label indicator matrix with its $(i,j)$-th entry $Y_l(i,j) = 1$ if $\mathbf{x}_i$ belongs to class $j$, otherwise $Y_l(i,j) = 0$. Finally the label of $\mathbf{x}$ can be determined by $arg\max_j \mathbf{r}_{\mathbf{x}}(j)$.

# Semi-supervised Classification without Discriminative Regularization: A Semi-parametric Approach

In the last section, we just followed the standard procedure of the *manifold regularization* approach and derived a discriminative regularization approach for semi-supervised learning. In this section, we will derive a novel discriminative semi-supervised classification approach using the semi-parametric regression technique, which include the discriminative information into the expression of the classification function, rather than penalize it as a regularization term. First let's briefly review some basic concepts in semi-parametric regression.

## Semi-parametric Regression

The standard definition of *semi-parametric regression* is

**Definition 1 (Semi-parametric Regression)**. *Semi-parametric regression refers to the regression models in which the predictor contains both parametric and non-parametric components*.

For example, suppose we want to construct a *predictor* $\tilde{f}$ from $n$ input-output pairs $\{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$ by

$$\tilde{f}^* = \arg\min_{\tilde{f}} \sum_{i=1}^{n} \mathcal{L}(y_i, \tilde{f}(\mathbf{x}_i)) \quad (22)$$

where $\mathcal{L}(\cdot,\cdot)$ is some loss function. Then for *parametric regression*, $\tilde{f}(\mathbf{x})$ can be written as an explicit function of $\mathbf{x}$ which is dependent on some parameters $\mathbf{w}$ (*e.g.*, linear regression); for *non-parametric regression*, $\tilde{f}(\mathbf{x})$ cannot be estimated via an explicit parametric function, *i.e.*, it can only be estimated from the data (*e.g.*, $k$-nearest neighbor classifier); for *semi-parametric regression*, $\tilde{f}$ can be decomposed into two parts as

$$\tilde{f} = f + h$$

where $f$ is a non-parametric predictor which can be estimated from the data set, $h \in span\{\psi_p\}$ is a parametric estimator, with $\{\psi_p\}$ a family of parametric functions.

The semi-parametric model can be useful in many cases (Schölkopf & Smola, 2002), for example, if one has some additional knowledge that the major properties of the data set are described by a small set of independent basis functions $\{\psi_1(\cdot), \psi_2(\cdot), \cdots, \psi_m(\cdot)\}$, or one may want to correct the data from some (*e.g.* linear) trends. From another point of view, the semi-parametric way can also make the model more *understandable* without sacrificing the accuracy (the non-parametric component usually makes the model accurate since it is estimated from the data, while the parametric component can make the model more easily understood since it can be written in an explicit form).

Since label prediction is in fact a special regression problem with discrete targets, we can also construct a semi-parametric model for semi-supervised classification. More concretely, the semi-parametric model is composed of two parts: one part (the non-parametric part) is learned from the labeled data by standard kernel techniques, the other part

(the parametric part) is learned from both labeled and unlabeled data to incorporate the geometrical discriminative information contained in the data set. In other words, in the case when there is no sufficient labeled data, then the non-parametric regression based on purely the labeled data points may not be accurate, therefore we apply a parametric estimator to "correct" the original predictor to make it more accurate.

Before we go into the details of our *Semi-parametric Discriminative Semi-supervised Classification* (*SDSC*) algorithm, first let's see a preliminary theorem (Schölkopf & Smola, 2002).

**Theorem 1 (Semi-parametric Representor Theorem).** *Suppose we are given a nonempty set $\mathcal{X}$, a positive definite real-valued kernel $k$ on $\mathcal{X} \times \mathcal{X}$, a training set $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_l, y_l) \in \mathcal{X} \times \mathbb{R}$, a strictly monotonically increasing real-valued function $\Omega$ on $[0, \infty]$, an arbitrary cost function $c : (\mathcal{X} \times \mathbb{R}^2)^l \to \mathbb{R} \bigcup \{\infty\}$, and a class of functions*

$$\mathcal{F} = \left\{ f \in \mathbb{R}^{\mathcal{X}} \middle| f(\cdot) = \sum_{i=1}^{\infty} \gamma_i k(\cdot, \mathbf{z}_i), \|f\|_k < \infty \right\}$$

*where $\gamma_i \in \mathbb{R}, \mathbf{z}_i \in \mathcal{X}$, and $\|\cdot\|$ is the norm in the Reproducing Kernel Hilbert Space (RKHS) $\mathcal{H}_k$ associated with $k$, i.e., for any $\mathbf{z}_i \in \mathcal{X}, \gamma_i \in \mathbb{R}$,*

$$\left\| \sum\nolimits_{i=1}^{\infty} \gamma_i k(\cdot, \mathbf{z}_i) \right\|^2 = \sum\nolimits_{i,j=1}^{\infty} \gamma_i \gamma_j k(\mathbf{z}_i, \mathbf{z}_j),$$

*a set of $m$ real-valued functions $\{\psi_p\}_{p=1}^m$ on $\mathcal{X}$, with the property that the $l \times m$ matrix $(\psi_p(\mathbf{x}_i))_{ip}$ has rank $m$. Then any $\tilde{f} = f + h$, with $f \in \mathcal{F}$ and $h \in span\{\psi_p\}$, minimizing the regularized risk*

$$c((\mathbf{x}_1, y_1, \tilde{f}(\mathbf{x}_1)), \cdots, (\mathbf{x}_l, y_l, \tilde{f}(\mathbf{x}_l))) + \Omega(\|f\|_k)$$

*admits a representation of the form*

$$\tilde{f}(\cdot) = \sum_{i=1}^{l} \alpha_i k(\mathbf{x}_i, \cdot) + \sum_{p=1}^{m} \gamma_p \psi_p(\cdot) \tag{23}$$

*with $\alpha_i, \gamma_p \in \mathbb{R}$ for all $p = 1, 2, \cdots, m$.*

In theorem 1, the parametric functions $\{\psi_p\}_{p=1}^m$ can be any functions, *e.g.*, in standard *SVM*, $m = 1$ and $\psi_1(\mathbf{x}) = 1$ (Schölkopf & Smola, 2002). Therefore, in the semi-parametric setting, the function $\tilde{f}^*$ minimizing the following structural loss

$$\mathcal{J} = \sum_{i=1}^{l} \mathcal{L}(\mathbf{x}_i, y_i, \tilde{f}(\mathbf{x}_i)) + \delta \|f\|_k^2 \tag{24}$$

would have the form of Eq.(23). Note that in the above loss, the parametric functions $\{\psi_p\}_{p=1}^m$ do not contribute to the regularization term $\|f\|_k^2$. (Schölkopf & Smola, 2002) pointed out that this needs not to be a major concern when $m$ is sufficiently small than $l$.

## Incorporating the Discriminative Information

Now the only problem remained is how to fit semi-parametric regression into the semi-supervised setting, *i.e.*, how to incorporate the discriminative information contained in the data set into the learning process. A natural choice would be to construct some proper parametric functions which carry those geometrical information.

More concretely, in our *SDSC* algorithm, we use only one parametric function, *i.e.*, $m = 1$, which is learned from both labeled and unlabeled data based on the discriminative kernel $\mathbf{D}$. Mathematically, we just adopt $C$ eigenvectors of $\mathbf{D}$ corresponding to its smallest $C$ eigenvalues (since according to Eq.(9), these eigenvectors carry the discriminative information contained in the data set).

Returning to the expression of $\mathbf{D}$ in Eq.(10), we may easily find that it is just the inverse of a regular kernel $\mathbf{K}$. Therefore there is a direct relationship between the eigenvalues of $\mathbf{K}$ and $\mathbf{D}$, *i.e.*, if $\mu$ is the eigenvalue of $\mathbf{K}$, then $1/(1 + \mu/\lambda)$ would be the eigenvalue of $\mathbf{D}$, and their corresponding eigenvectors are exactly the same. Hence the eigenvectors corresponding to the smallest $C$ eigenvalues of $\mathbf{D}$ are just the eigenvectors corresponding to the largest $C$ eigenvalues of $\mathbf{K}$. In this sense, the extraction of the discriminative information using $\mathbf{D}$ is equivalent to performing *kernel PCA* (Schölkopf & Smola, 2002) using the kernel $\mathbf{K}$, *i.e.*, for an arbitrary point $\mathbf{x}$, we can compute its corresponding discriminative information by

$$\psi(\mathbf{x}) = \mathbf{k}_{\mathbf{x}}^T \mathbf{V} \mathbf{\Lambda}^{-1} \tag{25}$$

where $\mathbf{k}_{\mathbf{x}} = [k(\mathbf{x}, \mathbf{x}_1), \cdots, k(\mathbf{x}, \mathbf{x}_n)]^T \in \mathbb{R}^{n \times 1}$, $\mathbf{V} = [\mathbf{v}_1, \cdots, \mathbf{v}_C]$ are the eigenvectors corresponding to the largest $C$ eigenvalues of $\mathbf{K}$, and $\mathbf{\Lambda}$ is a diagonal matrix with the smallest $C$ eigenvalues of $\mathbf{D}$ on its diagonal line. Clearly, $\psi(\mathbf{x})$ is an $1 \times C$ vector, which can be viewed as the soft predicted label vector by purely using the discriminative kernel.

Next, we just set $\psi(\mathbf{x})$ as the parametric part of $f(\mathbf{x})$, then the complete form of $f(\mathbf{x})$ can be written as

$$f(\mathbf{x}) = \mathbf{k}_{l\mathbf{x}}^T \mathbf{\Gamma} + \gamma \psi(\mathbf{x}) \tag{26}$$

where $\mathbf{k}_{l\mathbf{x}} = [k(\mathbf{x}, \mathbf{x}_1), \cdots, k(\mathbf{x}, \mathbf{x}_l)]^T \in \mathbb{R}^{l \times 1}$, and $\mathbf{\Gamma} \in \mathbb{R}^{l \times C}$ are the expansion coefficients. We can see that $f(\mathbf{x})$ is in fact a vector function, its output is a $C \times 1$ predicted vector.

## A Concrete Algorithm

Therefore the only thing left is how to determine the optimal $\mathbf{\Gamma}$ and $\beta$ in Eq.(26). Combining Eq.(26) and Eq.(24) together, and adopting the square loss, we can rewrite the structural loss as

$$\begin{aligned} \mathcal{J} &= \sum_{i=1}^{l} \|\mathbf{y}_i^T - (\mathbf{k}_{l\mathbf{x}_i}^T \mathbf{\Gamma} + \gamma \psi(\mathbf{x}_i))\|^2 + \delta tr(\mathbf{\Gamma}^T \mathbf{K}_l \mathbf{\Gamma}) \\ &= \left\| \mathbf{Y}_l - \mathbf{K}_l^T \mathbf{\Gamma} - \gamma \mathbf{\Psi} \right\|^2 + \delta tr(\mathbf{\Gamma}^T \mathbf{K}_l \mathbf{\Gamma}) \end{aligned} \tag{27}$$

where $\mathbf{y}_i \in \mathbb{R}^{C \times 1}$ is the label vector of $\mathbf{x}_i$, such that $y_i(j) = 1$ is $\mathbf{x}_i$ is labeled as class $j$, and $y_i(j) = 0$ otherwise, $\mathbf{Y}_l = [\mathbf{y}_1, \cdots, \mathbf{y}_l]^T \in \mathbb{R}^{l \times C}$, $\mathbf{\Psi} = [\psi^T(\mathbf{x}_1), \cdots, \psi^T(\mathbf{x}_l)]^T \in$

$\mathbb{R}^{l\times C}$, $\mathbf{K}_l \in \mathbb{R}^{l\times l}$ is the kernel matrix constructed on the labeled set. Then

$$\frac{\partial \mathcal{J}}{\partial \mathbf{\Gamma}} = 2\left(\mathbf{K}_l\mathbf{K}_l\mathbf{\Gamma} + \gamma\mathbf{K}_l\mathbf{\Psi} - \mathbf{K}^l\mathbf{Y}_l + \delta\mathbf{K}^l\mathbf{\Gamma}\right) \quad (28)$$

$$\frac{\partial \mathcal{J}}{\partial \gamma} = 2\left(\mathbf{\Psi}^T\mathbf{K}^l\mathbf{\Gamma} + \gamma\mathbf{\Psi}^T\mathbf{\Psi} - \mathbf{\Psi}^T\mathbf{Y}_l\right) \quad (29)$$

By setting $\frac{\partial \mathcal{J}}{\partial \mathbf{\Gamma}} = 0$ and $\frac{\partial \mathcal{J}}{\partial \gamma} = 0$, we can get the optimal $\mathbf{\Gamma}$ and $\gamma$. However, one can easily discover that the resultant linear equation system is over-determined, and the final solution may not exist. In this case, we can apply some approximate algorithms, such as the least squares, to solve it (Golub & Loan, 1983). Another approach to avoid this problem is that we adopt different scaling factors for different classes, *i.e.*,

$$f(\mathbf{x}) = \mathbf{k}_{l\mathbf{x}}^T\mathbf{\Gamma} + \psi(\mathbf{x})\mathbf{\Upsilon} \quad (30)$$

where $\mathbf{\Upsilon} = diag(\gamma_1, \gamma_2, \cdots, \gamma_C) \in \mathbb{R}^{C\times C}$ is a diagonal matrix with the scaling factors on its diagonal line. In this way, the number of variables we will solve is increased so that the resultant linear equation system could be efficiently solved by

$$\mathbf{\Gamma}_c = \left(\delta\mathbf{I} - \frac{\boldsymbol{\psi}_c\boldsymbol{\psi}_c^T\mathbf{K}_l}{\boldsymbol{\psi}_c\boldsymbol{\psi}_c^T} + \mathbf{K}_l\right)^{-1}\left(\mathbf{I} - \frac{\boldsymbol{\psi}_c\boldsymbol{\psi}_c^T}{(\boldsymbol{\psi}^c)^T\boldsymbol{\psi}^c}\right)\mathbf{y}_c$$

$$\gamma_c = \frac{\boldsymbol{\psi}_c^T\mathbf{y}_c - \boldsymbol{\psi}_c^T\mathbf{K}_l\mathbf{\Gamma}_c}{\boldsymbol{\psi}_c^T\boldsymbol{\psi}_c}$$

where $\mathbf{\Gamma}_c$ represents the $c$-th column of $\mathbf{\Gamma}$, $\boldsymbol{\psi}_c$ denotes the $c$-th column of $\mathbf{\Psi}$, $\mathbf{y}_c$ is the $c$-th column of $\mathbf{Y}_l$.

## Experiments

In this section, we present a set of experiments to show the effectiveness of our method. First let's describe the basic information of the data sets.

### The Data Sets

12 datasets, including 2 artificial datasets and 10 real datasets, are used in our experiments to evaluate the performances of the methods. Table 1 summarizes the characteristics of the datasets.

- **Toy Data Sets: g241c & g241n.** Each data set contains two classes with 350 points in each class, and the data sets are generated in a way of violating the cluster assumptions or misleading class structures. For details one can refer to (Chapelle *et al*., 2006).
- **Image Data Sets: USPS, COIL& Digit1.** The first two data sets are generated from the famous USPS[1] and COIL[2] databases, such that the resultant image data did not appear to be manifolds explicitly. The *digit 1* data set was generated by transforming the image of digit 1, and the image data appears a manifold structure strongly. For the detailed generation information of the three data sets one can refer to (Chapelle *et al*., 2006).

[1]http://www.kernel-machines.org/data.html.

[2]http://www1.cs.columbia.edu/CAVE/software/softlib/coil-20.php

Table 1: Descriptions of the datasets

| Datasets | Sizes | Classes | Dimensions |
|---|---|---|---|
| **g241c** | 1500 | 2 | 241 |
| **g241n** | 1500 | 2 | 241 |
| **USPS** | 1500 | 2 | 241 |
| **COIL** | 1500 | 6 | 241 |
| **digit1** | 1500 | 2 | 241 |
| **cornell** | 827 | 7 | 4134 |
| **texas** | 814 | 7 | 4029 |
| **wisconsin** | 1166 | 7 | 4189 |
| **washington** | 1210 | 7 | 4165 |
| **BCI** | 400 | 2 | 117 |
| **diabetes** | 768 | 2 | 8 |
| **ionosphere** | 351 | 2 | 34 |

- **Text Data Sets: Cornell, Texas, Wisconsin & Washington.** All these four data sets are selected from the famous WebKB database [3], and the web pages are classified into seven categories.
- **Other Data Sets: BCI, Diabetes & Ionosphere.** The BCI data set is provided in (Lal *et al*, 2004), which originates from research toward the development of a brain computer interface (BCI). The Diabetes and Ionosphere data sets are downloaded from the UCI repository[4], which are usually used as the benchmark data sets for semi-supervised learning algorithms.

### Methods & Parameter Settings

Besides our methods, we also implement some other competing methods for experimental comparison. For all the methods, their hyperparameters were set by 5-fold cross validation from some grids introduced in the following.

- ***Local Learning Regularization*** (*LLReg*). The implementation of this algorithm is the same as in (Wu & Schölkopf, 2007), in which we also adopt the mutual neighborhood with its size search from $\{5, 10, 50\}$. The regularization parameter of the local classifier is searched from $\{4^{-3}, 4^{-2}, 4^{-1}, 1, 4^1, 4^2, 4^3\}$, and the tradeoff parameter between the loss and local regularization term is also searched from $\{4^{-3}, 4^{-2}, 4^{-1}, 1, 4^1, 4^2, 4^3\}$.
- ***Laplacian Regularized Least Squares*** (*LapRLS*). The implementation of the algorithm is the same as in (Belkin *et al*., 2006), in which the variance of the Gaussian similarity is also set by the method in (Zhu *et al*, 2003), and the extrinsic and intrinsic regularization parameters are searched from $\{4^{-3}, 4^{-2}, 4^{-1}, 1, 4^1, 4^2, 4^3\}$. For the RKHS, the linear kernel is adopted.
- ***Learning with Local and Global Consistency*** (*LLGC*). The implementation of the algorithm is the same as in (Zhou *et al*, 2004), in which the variance of the Gaussian kernel is also determined by the method in (Zhu *et al*, 2003), and the regularization parameter is searched from $\{4^{-3}, 4^{-2}, 4^{-1}, 1, 4^1, 4^2, 4^3\}$.

[3]http://www.cs.cmu.edu/W̃ebKB/.

[4]http://www.ics.uci.edu/mlearn/MLRepository.html.

Table 2: Experimental results with 10% of the data points randomly labeled

| | *GRF* | *LLGC* | *LLReg* | *LapRLS* | *SSDR* | *SDSC* |
|---|---|---|---|---|---|---|
| *g241c* | $56.34 \pm 2.1665$ | $77.13 \pm 2.5871$ | $65.31 \pm 2.1220$ | $80.44 \pm 1.0746$ | $80.37 \pm 0.2103$ | $\mathbf{83.41 \pm 0.4378}$ |
| *g241n* | $55.06 \pm 1.9519$ | $49.75 \pm 0.2570$ | $73.25 \pm 0.2466$ | $76.89 \pm 1.1350$ | $77.42 \pm 1.2114$ | $\mathbf{79.36 \pm 0.7433}$ |
| *USPS* | $94.87 \pm 1.7490$ | $96.19 \pm 0.7588$ | $95.79 \pm 0.6804$ | $88.80 \pm 1.0087$ | $97.36 \pm 1.3317$ | $\mathbf{98.70 \pm 0.9436}$ |
| *COIL* | $91.23 \pm 1.8321$ | $92.04 \pm 1.9170$ | $86.86 \pm 2.2190$ | $73.35 \pm 1.8921$ | $\mathbf{95.42 \pm 1.2188}$ | $89.20 \pm 1.7734$ |
| *digit1* | $96.95 \pm 0.9601$ | $95.49 \pm 0.5638$ | $97.64 \pm 0.6636$ | $92.79 \pm 1.0960$ | $98.49 \pm 1.5673$ | $\mathbf{99.13 \pm 1.3210}$ |
| *cornell* | $71.43 \pm 0.8564$ | $76.30 \pm 2.5865$ | $79.46 \pm 1.6336$ | $80.59 \pm 1.6665$ | $\mathbf{81.32 \pm 0.7568}$ | $78.23 \pm 1.4576$ |
| *texas* | $70.03 \pm 0.8371$ | $75.93 \pm 3.6708$ | $79.44 \pm 1.7638$ | $78.15 \pm 1.5667$ | $\mathbf{80.87 \pm 1.0219}$ | $77.46 \pm 1.2347$ |
| *wisconsin* | $74.65 \pm 0.4979$ | $80.57 \pm 1.9062$ | $83.62 \pm 1.5191$ | $84.21 \pm 0.9656$ | $\mathbf{84.58 \pm 0.7994}$ | $77.84 \pm 0.5386$ |
| *washington* | $78.26 \pm 0.4053$ | $80.23 \pm 1.3997$ | $86.37 \pm 1.5516$ | $86.58 \pm 1.4985$ | $\mathbf{88.24 \pm 0.7458}$ | $76.97 \pm 1.1932$ |
| *BCI* | $50.49 \pm 1.9392$ | $53.07 \pm 2.9037$ | $51.56 \pm 2.8277$ | $61.84 \pm 2.8177$ | $60.44 \pm 2.1089$ | $\mathbf{62.01 \pm 2.2847}$ |
| *diabetes* | $70.69 \pm 2.6321$ | $67.15 \pm 1.9766$ | $68.38 \pm 2.1772$ | $64.95 \pm 1.1024$ | $\mathbf{72.37 \pm 1.4545}$ | $72.20 \pm 1.5321$ |
| *ionosphere* | $70.21 \pm 2.2778$ | $67.31 \pm 2.6155$ | $68.15 \pm 2.3018$ | $65.17 \pm 0.6628$ | $82.05 \pm 0.7996$ | $\mathbf{86.73 \pm 1.8210}$ |

- ***Gaussian Random Fields*** (*GRF*). The implementation of the algorithm is the same as in (Zhu *et al*, 2003).

For our *Semi-Supervised Discriminative Regularization* (***SSDR***) and *Semi-parametric Discriminative Semi-supervised Classification* (***SDSC***) algorithms, the Gaussian kernel is adopted as the original kernel $\mathbf{K}$, whose variance is set by the method in (Zhu *et al*, 2003). The regularization parameter $\lambda$ for constructing the discriminative kernel $\mathbf{K}$, and the regularization parameters $\gamma_1$, $\gamma_2$ in *SSDR*, $\delta$ in *SDSC*, are all searched from the grid $\{4^{-3}, 4^{-2}, 4^{-1}, 1, 4^1, 4^2, 4^3\}$.

### Experimental Results

In our experiments, we randomly label 10% of the points for each data set, and report the mean classification accuracies and standard deviations of 50 independent runs in table 2, from which we can also see the superiority of our discriminative algorithms.

## Conclusion

In this paper, we propose a novel strategy for semi-supervised classification, in which we directly penalize the discriminality of the classification function instead of punishing its smoothness as in traditional methods. Guided by such idea, we propose two specific algorithms, semi-supervised discriminative regularization and semi-parametric discriminative semi-supervised classification, and finally the experimental results on several benchmark data sets are presented to show their effectiveness.

## References


Belkin, M., Niyogi, P., and Sindhwani, V. (2006). Manifold Regularization: a Geometric Framework for Learning from Examples. *Journal of Machine Learning Research*.

Belkin, M., Matveeva, I., and Niyogi, P. (2004). Regularization and Semi-supervised Learning on Large Graphs. In *COLT* 2004.

Blum, A., and Chawla, S. (2001). Learning from Labeled and Unlabeled Data Using Graph Mincuts. *ICML* 18.

Blum, A., and Mitchell, T. (1998). Combining Labeled and Unlabeled Data with Co-Training. In *COLT* 1998.

Chapelle, O., Schölkopf and Zien, A. (eds.). (2006) *Semi-Supervised Learning*. MIT Press: Cambridge, MA.

Chung, F. R. K. (1997). *Spectral Graph Theory*. American Mathematical Society.

Friedman, J. H. (1989). Regularized Discriminant Analysis. *Journal of American Statistical Association*: 84(405), 165-175.

Fukunaga, F. (1990). *Introduction to Statistical Pattern Recognition*. Academic Press, New York, 2nd edtion.

Golub, G. H. and Loan, C. V. (1983). *Matrix Computations*. Johns Hopkins University Press.

Getz, G., Shental, N., and Domany, E. (2005). Semi-Supervised Learning – A Statistical Physics Approach. In *"Learning with Partially Classified Training Data" – ICML05 workshop*.

Joachims, T. (1999). Transductive Inference for Text Classification using Support Vector Machines. *ICML* 16.

Lal, T. N., Schröder, M., Hinterberger, T., Weston, J., Bogdan, M., Birbaumer, N., and Schölkopf, B. (2004). Support Vector Channel Selection in BCI. *IEEE TBE*, 51(6):1003-1010, 2004.

Miller, D. J., and Uyar, U. S. (1997). A mixture of experts classifier with learning based on both labelled and unlabelled data. In *NIPS 9*, 571-577.

Nigam, K., McCallum, A. K., Thrun, S., and Mitchell, T. (2000). Text classification from labeled and unlabeled documents using EM. *Machine Learning*, vol. 39, no. 2-3, pp.103-134.

Shahshahani, B., and Landgrebe, D. (1994). The effect of unlabeled samples in reducing the small sample size problem and mitigating the hughes phenomenon. *IEEE TGRS* 32:5, 1087-1095.

Schölkopf, B., and Smola, A. J. (2002). *Learning with Kernels*. MIT Press, Cambridge, MA.

Sindhwani, V., Niyogi, P. and Belkin, M. (2005). Beyond the Point Cloud: from Transductive to Semi-Supervised Learning. *Proceedings of the 22nd International Conference on Machine Learning*, 824-831.

Szummer, M., & Jaakkola, T. (2002). Partially Labeled Classification with Markov Random Walks. *NIPS 14*.

Wang, F., Wang, S., Zhang, C., and Winther, O. (2007). Semi-Supervised Mean Fields. In *AISTATS* 11.

Wu, M. and Schölkopf, B. (2007). Transductive Classification via Local Learning Regularization. *AISTATS 11*.

Zhou, D., Bousquet, O., Lal, T. N., Weston, J. and Schölkopf, B. (2004). Learning with Local and Global Consistency. *NIPS 16*.

Zhou, D., Schölkopf, B., and Hofmann, T. (2005). Semi-supervised Learning on Directed Graphs. *NIPS 17*.

Zhu, X., Ghahramani, Z., & Lafferty, J.(2003). Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions. *ICML*.