Spectral Kernel Learning for Semi-Supervised Classification

Wei Liu
The Chinese University of Hong Kong
wliu5@ie.cuhk.edu.hk

Buyue Qian
University of California Davis
qianbuyue@gmail.com

Jingyu Cui
Stanford University
jycui@stanford.edu

Jianzhuang Liu
The Chinese University of Hong Kong
jzliu@ie.cuhk.edu.hk

Abstract

Typical graph-theoretic approaches for semi-supervised classification infer labels of unlabeled instances with the help of graph Laplacians. Founded on the spectral decomposition of the graph Laplacian, this paper learns a kernel matrix via minimizing the leave-one-out classification error on the labeled instances. To this end, an efficient algorithm is presented based on linear programming, resulting in a transductive spectral kernel. The idea of our algorithm stems from regularization methodology and also has a nice interpretation in terms of spectral clustering. A simple classifier can be readily built upon the learned kernel, which suffices to give prediction for any data point aside from those in the available dataset. Besides this usage, the spectral kernel can be effectively used in tandem with conventional kernel machines such as SVMs. We demonstrate the efficacy of the proposed algorithm through experiments carried out on challenging classification tasks.

1 Introduction

A hot stream of broad research interests in recent years is *semi-supervised learning* (SSL) [Chapelle *et al.*, 2006] which deals with the situations of sparse labeled data together with abundant unlabeled data and utilizes both labeled and unlabeled data in algorithms. SSL has been proved effective in a lot of practical cases, since it is often cheap to obtain unlabeled data by an automatic procedure or scripts but quite expensive to identify the labels of data. Incorporating unlabeled data into learning tasks, SSL triggers a wide range of real-world machine learning applications where labeled samples are very scarce.

Current research on semi-supervised learning is mostly attracted by semi-supervised classification. A family of graph-based approaches [Belkin *et al.*, 2006][Sindhwani *et al.*, 2005][Zhou *et al.*, 2004][Zhu *et al.*, 2003][Zhu *et al.*, 2005], founded on spectral graph theory [Chung, 1997], either put forward new semi-supervised kernels or propose new graph-based regularization frameworks with labeled and unlabeled data. Although graph-based semi-supervised learning has been studied extensively, so far there are few comprehensive techniques to integrate graph-theoretic regularization and nonparametric kernel learning effectively together for classification tasks. To this end, we present a flexible as well as scalable algorithm for learning spectral kernels through taking advantage of regularization methodology and nonparametric kernel construction piloted by graph Laplacians.

2 Related Work

It is pointed out in [Sindhwani *et al.*, 2005] that establishing data-dependent kernels will be more and more important for supervised or semi-supervised learning tasks. As a paradigm, building kernels with graph Laplacians receives increasing attention in the contexts of clustering [Saerens *et al.*, 2004] and classification [Zhang and Ando, 2006]. [Saerens *et al.*, 2004] shows that the average commute time between any two data points can be computed using the pseudoinverse L^+ of the graph Laplacian matrix L. It turns out that L^+ is a valid kernel and can be utilized for kernel k-means clustering and SVM classification. We call L^+ the *graph Laplacian kernel* or *Laplacian kernel* in what follows.

Since the nature of L^+ is parametric with respect to the eigenspectrum of L, we herewith refer to it as the parametric spectral kernel. Another parametric spectral kernel proposed in the early work [Kondor and Lafferty, 2002] is the Laplacian diffusion kernel $\exp(-\alpha L)$. As opposed to the parametric kernels, we would desire to obtain more flexible kernels in order to avoid tedious selection among different parametric families. This objective motivates a novel topic of kernel learning, namely *nonparametric kernel learning*. For example, [Lanckriet *et al.*, 2004] uses kernel-target alignment to learn nonparametric kernel matrices in a supervised mode. The seminal works [Zhu *et al.*, 2005][Hoi *et al.*, 2006] conduct semi-supervised kernel alignment.

These optimization problems involved in kernel alignment are generally solved using *semidefinite programming* (SDP) [Boyd and Vandenberge, 2004]. However, the computational complexity of SDPs has restricted their applications to small scale problems, and thus prevented nonparametric kernel learning from being applicable to large scale problems, e.g., large scale semi-supervised learning. Alternatively, [Zhu *et al.*, 2005] and [Hoi *et al.*, 2006] show that these optimization problems can be more efficiently solved by *quadratically constrained quadratic programming* (QCQP) or *quadratic*

programming (QP). Nonetheless, a significant limitation for all these nonparametric kernel approaches is that they only support transductive settings and cannot be extended to unseen data points.

3 Regularized Spectral Kernels

[Zhou *et al.*, 2004] proposed a transductive inference algorithm to impose label smoothness along graphs, which coordinates the local and global consistency within the label propagation process. Suppose we have a data set $\mathcal{X} = \{\mathbf{x}_1, \cdots, \mathbf{x}_l, \mathbf{x}_{l+1} \cdots, \mathbf{x}_n\} \subset \mathbb{R}^d$ of which the first l points belonging to c classes are labeled as $y_i \in \mathcal{Y} = \{1, \cdots, c\}$ and the rest points are unlabeled. Intuitively, we define a labeling function $f : \mathcal{X} \to \mathbb{R}$ to assign a discrete label to each data point according to particular class, i.e., $f(\mathbf{x}_i) = k$ if and only if $\mathbf{x}_i$ belongs to the kth class.

Given l labeled examples, we define a class indicator matrix as $C \in \mathbb{R}^{n \times c}$ in which $C_{ik} = 1$ if $y_i = k \in \mathcal{Y}$ and $C_{ik} = 0$ otherwise. We intend to seek a real-valued matrix $F \in \mathbb{R}^{n \times c}$ to express a full classification on the entire dataset $\mathcal{X}$ so that the exact labeling is launched by

$$f(\mathbf{x}_i) = \arg \max_{1 \le k \le c} F_{ik}. \tag{1}$$

Let us construct an undirected, weighted graph $G(V, W)$ that exposes the underlying manifold structure of the data. Each data point in $\mathcal{X}$ corresponds to a node in V. An edge is established between two nodes v_i and v_j if the corresponding two points $\mathbf{x}_i$ and $\mathbf{x}_j$ are among k nearest neighbors of each other. For convenience, we adopt the trick proposed in [Hein and Maier, 2007] to construct G as a k-nearest neighbor (k-NN) graph. Let us define a distance function $h(\mathbf{x}) = \|\mathbf{x} - \mathbf{x}^{(k)}\|$ where $\mathbf{x}^{(k)}$ is the kth nearest neighbor of $\mathbf{x}$ in $\mathcal{X}$. The weight matrix $W \in \mathbb{R}^{n \times n}$ associated with G is formed subsequently as

$$W_{ij} = W(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\max\{h(\mathbf{x}_i)^2, h(\mathbf{x}_j)^2\}}\right), \quad \text{if } \|\mathbf{x}_i - \mathbf{x}_j\| \le \max\{h(\mathbf{x}_i), h(\mathbf{x}_j)\}, \tag{2}$$

and $W_{ij} = 0$ otherwise. Note that we set W_{ii} = 0 to avoid self-loops. We further denote the diagonal degree matrix $D \in \mathbb{R}^{n \times n}$ with $D_{ii} = \sum_{j=1}^{n} W_{ij}$.

The regularization framework proposed by [Zhou *et al.*, 2004] can learn a global classification as

$$F^* = (1 - \alpha)(I - \alpha S)^{-1} C, \tag{3}$$

where $0 < \alpha < 1$ is the regularization parameter and $S = D^{-1/2} W D^{-1/2}$ symmetrically normalizes W. The calculated matrix F^* stacks the final class assignment.

Now we revisit Zhou *et al.*'s approach from the perspective of kernels, still exploiting the expression of eq. (3). The graph Laplacian matrix $L = D - W$ is the central ingredient of spectral graph theory [Chung, 1997]. In this paper, we choose the normalized graph Laplacian $\mathcal{L} = D^{-1/2} L D^{-1/2} = I - S$ and perform eigenvalue decomposition on it, yielding the eigensystem $\{(\lambda_i, \mathbf{v}_i)\}_{i=1}^{n}$ such that $\mathcal{L} = V \Lambda V^T$, where $0 \le \lambda_1 \le \lambda_2 \cdots \le \lambda_n$, $\Lambda = \mathrm{diag}(\lambda_1, \cdots, \lambda_n)$, and $V = [\mathbf{v}_1, \cdots, \mathbf{v}_n]$. Then we can rewrite eq. (3) as follows

$$\begin{aligned} F^* &= (1 - \alpha)\left((1 - \alpha) I + \alpha (I - S)\right)^{-1} C \\ &= \left(I + \frac{\alpha}{1 - \alpha} \mathcal{L}\right)^{-1} C = V \left(I + \frac{\alpha}{1 - \alpha} \Lambda\right)^{-1} V^T C \\ &= V \Theta V^T C = K^r C, \end{aligned} \tag{4}$$

where Θ is a diagonal matrix with entries in the diagonal being $\theta_i = \frac{1 - \alpha}{1 - \alpha + \alpha \lambda_i}$. Because $\lambda_i \in [0, 2]$ [Chung, 1997], we have $\theta_1 \ge \theta_2 \ge \cdots \ge \theta_n \ge 0$. Hence the matrix $K^r = V \Theta V^T \in \mathbb{R}^{n \times n}$ is positive semidefinite and herewith behaves as a kernel matrix.

Specially, we call K^r the *regularized spectral kernel* since it is derived from the regularization framework and also built upon spectral transformation of $\mathcal{L}$, i.e., $\Lambda \to \Theta$. By engaging K^r, it is not difficult to translate eq. (1) into

$$f(\mathbf{x}_i) = \arg \max_{1 \le k \le c} \sum_{y_j = k} K^r_{ij}, \tag{5}$$

which is a multi-class classifier directly derived from the regularized kernel matrix.

4 Transductive Spectral Kernels

In this section, we describe how to learn an effective data-dependent kernel for semi-supervised learning. Prior work on producing data-dependent kernels may be roughly classified into two categories: (i) choosing parametric families of Laplacian kernels, and (ii) learning a nonparametric kernel matrix over the seen data points alone. Here we present a new algorithm for learning a nonparametric spectral kernel based on the spectral decomposition of the graph Laplacian.

Our kernel learning algorithm introduces a universal procedure allowing: (1) to compute similarities between nodes in an undirected, weighted graph in a nonparametric mode, (2) to maintain a similarity gap between nodes with different labels in the graph, and (3) to compute similarities between an unseen node[1] and any node in the graph. Computing similarities between node pairs including unseen ones enables searching the data point which is most relevant (i.e., similar) to a given point, thus making clustering or classification feasible. This paper focuses on the application of such a technique to semi-supervised classification.

Motivated by the Laplacian kernel [Saerens *et al.*, 2004][Zhang and Ando, 2006] and its nonparametric spectral transform [Zhu *et al.*, 2005], we have the following theorem.

Theorem 1. *If a positive semidefinite matrix $\mathcal{L} \in \mathbb{R}^{n \times n}$ has an eigensystem $\{(\lambda_i, \mathbf{v}_i)\}_{i=1}^{n}$ ($0 \le \lambda_1 \le \cdots \le \lambda_n$), then the family of matrices $K = \sum_{i=1}^{n} \mu_i \mathbf{v}_i \mathbf{v}_i^T$ ($\mu_i \ge 0$) produces nonparametric kernels with K as kernel matrices.*

Proof. Because $\mu_i \ge 0$ and

$$\begin{aligned} K &= [\mathbf{v}_1, \cdots, \mathbf{v}_n] \begin{bmatrix} \sqrt{\mu_1} & & \\ & \cdots & \\ & & \sqrt{\mu_n} \end{bmatrix}^2 [\mathbf{v}_1, \cdots, \mathbf{v}_n]^T \\ &= (V U^{\frac{1}{2}})(V U^{\frac{1}{2}})^T \end{aligned} \tag{6}$$

[1] An unseen node or data point means a data point not in the dataset $\mathcal{X}$.

where $U = \text{diag}(\mu_1, \cdots, \mu_n)$, K is certainly positive semidefinite and thus a valid kernel matrix. ■

The elicited kernel matrix K is a linear combination of a set of basic kernels $\{K_i = \mathbf{v}_i \mathbf{v}_i^T\}_{i=1}^n$. So far, we are able to formulate the problem of nonparametric kernel learning into finding a nonparametric spectral transform $\tau : \lambda_i \rightarrow \mu_i$ such that optimizing K can be transferred to optimizing the spectral coefficients $\{\mu_i\}$.

4.1 Spectral Constraints

[Zhu *et al.*, 2005] imposed a decreasing order on the spectral coefficients $\mu_i = \tau(\lambda_i)$ for the sake of encouraging smooth components of the target kernel. The smaller λ_i, the more K_i should be favored in K, which motivates the order constraints

$$\mu_i \geq \mu_{i+1}, \quad i = 1, \cdots, n-1. \tag{7}$$

The regularized spectral kernel also satisfies the decreasing spectral order since $\theta_1 \geq \theta_2 \geq \cdots \geq \theta_n$. Nonetheless, a lot of order constraints may not suit the task at hand, resulting in overly constrained kernels. Hence, we only impose these constraints over the more smooth components of K that correspond to the m smallest eigenvalues $\{\lambda_i\}_{i=1}^m$ $(m < n)$ of the graph Laplacian $\mathcal{L}$ with $\lambda_i \neq 1$. Why the condition $\lambda_i \neq 1$ is imposed will be investigated in Subsection 4.3.

Following [Hoi *et al.*, 2006], we adopt the decaying constraints

$$\mu_i \geq \eta \mu_{i+1}, \quad i = 1, \cdots, m-1. \tag{8}$$

$\eta \geq 1$ is introduced as a decay factor that is an important parameter to control the decaying rate of spectral coefficients and influence the outcome performance of the learned kernel K.

4.2 Learning with Linear Programming

The target spectral kernel is herewith generated by

$$K = \sum_{i=1}^m \mu_i K_i = \sum_{i=1}^m \mu_i \mathbf{v}_i \mathbf{v}_i^T = (\bar{V}\bar{U}^{\frac{1}{2}})(\bar{V}\bar{U}^{\frac{1}{2}})^T, \tag{9}$$

in which $\bar{V} = [\mathbf{v}_1, \cdots, \mathbf{v}_m]$ and $\bar{U} = \text{diag}(\mu_1, \cdots, \mu_m)$. Eq. (9) discloses a spectral embedding $\bar{U}^{\frac{1}{2}}\bar{V}^T \in \mathbb{R}^{m \times n}$ for the raw sample matrix $X = [\mathbf{x}_1, \cdots, \mathbf{x}_n] \in \mathbb{R}^{d \times n}$. We denote the spectral embedding as $\Phi = [\phi_1, \cdots, \phi_n] = \bar{U}^{\frac{1}{2}}\bar{V}^T$ and have $\phi_i^T \phi_j = K_{ij}$. ϕ_i is exactly the m-dimensional spectral representation of $\mathbf{x}_i$ through eigen-decomposing $\mathcal{L}$. This type of spectral embedding originates from the well-know spectral clustering algorithm [Ng *et al.*, 2002]. Now, we hope that the spectral embedding is sufficiently smooth over the graph G, and then form the smoothness measure as

$$\begin{aligned} \mathcal{Q}_1(K) &= \frac{1}{2} \sum_{i,j=1}^n \left\| \frac{\phi_i}{\sqrt{D_{ii}}} - \frac{\phi_j}{\sqrt{D_{jj}}} \right\|^2 W_{ij} = tr(\Phi \mathcal{L} \Phi^T) \\ &= tr(\mathcal{L}\Phi^T\Phi) = tr(\mathcal{L}K), \end{aligned} \tag{10}$$

where $tr(.)$ stands for the matrix trace operator. Importantly, minimizing $\mathcal{Q}_1(K)$ encourages the smoothness of the spectral embedding Φ, which is actually embodied in the desired spectral kernel $K = \Phi^T\Phi$.

Let us set up two sets $\mathcal{S}$ and $\mathcal{D}$ to capture pairwise similarities and dissimilarities. Formally, we set $\mathcal{S} = \{(i,j) | \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ share the same label } (y_i = y_j, i \neq j)\}$ and $\mathcal{D} = \{(i,j) | \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are differently labeled } (y_i \neq y_j)\}$. By engaging eq. (10) and the idea of soft margin introduced by SVMs [Schölkopf and Smola, 2002], we suggest a convex optimization criterion as follows

$$\begin{aligned} \min_K \quad & \mathcal{Q}(K) = tr(\mathcal{L}K) + \\ & \beta \sum_{i=1}^l \left[1 + \sum_{j,(i,j)\in\mathcal{D}} K_{ij} - \sum_{j,(i,j)\in\mathcal{S}} K_{ij} \right]_+ \end{aligned} \tag{11}$$

where $\lceil x \rceil_+ = \max(x, 0)$ denotes the hinge loss as used in SVMs and $\beta > 0$ is the trade-off parameter. The two competing terms in the above equation are both critical to achieving a good spectral kernel under transductive settings. The first one penalizes large variations of the kernel matrix along the graph G, while the second one penalizes a small gap between the total similarity among the same labeled points and the total similarity among the differently labeled points.

Furthermore, minimizing $\mathcal{Q}(K)$ is equivalent to minimizing the leave-one-out classification error on the labeled instances when using eq. (5) as the classifier. Therefore, the kernel we are pursuing truly incorporates semi-supervised information as opposed to the Laplacian kernel, the regularized spectral kernel, and the other composite kernels, all of which are unsupervised in terms of construction mechanism.

It is not difficult to show $\mathcal{Q}_1(K) = tr(\mathcal{L}K) = \sum_{i=1}^m \lambda_i \mu_i$.[2] Let form an indicator matrix as $E = (e_{ij})_{ij} = [\mathbf{e}_1, \cdots, \mathbf{e}_n] \in \mathbb{R}^{n \times n}$ where $e_{ij} = 1$ if $(i,j) \in \mathcal{S}$, $e_{ij} = -1$ if $(i,j) \in \mathcal{D}$, and $e_{ij} = 0$ otherwise. The target kernel matrix with the spectral constraints can hitherto be put forward as the solution to the following linear program

$$\begin{aligned} \min_{\{\mu_t, \xi_i\}} \quad & \sum_{t=1}^m \lambda_t \mu_t + \beta \sum_{i=1}^l \xi_i \\ \text{subject to} \quad & \sum_{j=1}^n e_{ij} K_{ij} \geq 1 - \xi_i, \; i = 1, \cdots, l \\ & \mu_t \geq \eta \mu_{t+1}, \; t = 1, \cdots, m-1 \\ & \mu_t \geq 0, \; t = 1, \cdots, m \\ & \xi_i \geq 0, \; i = 1, \cdots, l \end{aligned} \tag{12}$$

where ξ_i's are introduced as slack variables.

Let $vec(P)$ denote the column vectorization of a matrix P and let $M = [vec(K_1), \cdots, vec(K_m)] \in \mathbb{R}^{n^2 \times m}$. Rewrite $M = [M_1^T, \cdots, M_n^T]^T$ in which $M_i \in \mathbb{R}^{n \times m}$, and then define an $l \times m$ matrix as

$$T = \begin{bmatrix} \mathbf{e}_1^T M_1 \\ \cdots \\ \mathbf{e}_l^T M_l \end{bmatrix}. \tag{13}$$

[2] Let $U = \text{diag}(\mu_1, \cdots, \mu_m, 0, \cdots, 0)$ and then $tr(\mathcal{L}K) = tr(V\Lambda V^T V U V^T) = tr(\Lambda U) = \sum_{i=1}^m \lambda_i \mu_i$.

Table 1: The semi-supervised classification algorithm via learning a transductive spectral kernel.

Algorithm. Semi-Supervised Classification Using TSK
Step 1. Construct a k-NN graph $G(V, W)$ upon samples $\mathcal{X} = \{\mathbf{x}_1, \cdots, \mathbf{x}_n\} \subset \mathbb{R}^d$ from c classes, and compute the normalized graph Laplacian $\mathcal{L}$.
Step 2. Solve the sparse eigenvalue problem $\mathcal{L}\mathbf{v} = \lambda\mathbf{v}$ and retain the $m < n$ smallest eigenvalues $\{\lambda_i\}$ and corresponding eigenvectors $\{\mathbf{v}_i\}$ such that $\lambda_1 \leq \cdots \leq \lambda_m$ and $\lambda_i \neq 1$.
Step 3. Solve the linear program eq. (14) and obtain the TSK $K = \sum_{i=1}^{m} \mu_i \mathbf{v}_i \mathbf{v}_i^T$.
Step 4. For each unlabeled point $\mathbf{x}_i$, infer its label as $f(\mathbf{x}_i) = \arg\max_{1 \leq k \leq c} \sum_{y_j = k} K_{ij}$. For an unseen point $\mathbf{x}$, use eq. (19) to predict its label as $f(\mathbf{x}) = \arg\max_{1 \leq k \leq c} \sum_{y_j = k} K_{x,j}$.

Consequently, the linear program eq. (12) can be succinctly rewritten as follows

$$\begin{aligned} \min_{\boldsymbol{\mu}, \boldsymbol{\xi}} \quad & \boldsymbol{\lambda}^T \boldsymbol{\mu} + \beta \mathbf{1}^T \boldsymbol{\xi} \\ \text{subject to} \quad & T\boldsymbol{\mu} \geq \mathbf{1} - \boldsymbol{\xi} \\ & \Gamma \boldsymbol{\mu} \geq 0 \\ & \boldsymbol{\mu} \geq 0 \\ & \boldsymbol{\xi} \geq 0 \end{aligned} \tag{14}$$

where $\boldsymbol{\lambda} = [\lambda_1, \cdots, \lambda_m]^T$, $\boldsymbol{\mu} = [\mu_1, \cdots, \mu_m]^T$, $\boldsymbol{\xi} = [\xi_1, \cdots, \xi_l]^T$, $\mathbf{1} = [1, \cdots, 1]^T \in \mathbb{R}^l$, and $\Gamma \in \mathbb{R}^{(m-1) \times m}$ with nonzero entries being $\Gamma_{ii} = 1$ and $\Gamma_{i,i+1} = -\eta$ $(1 \leq i \leq m-1)$.

Since all of the objective function and the constraints are linear with respect to $[\boldsymbol{\mu}^T, \boldsymbol{\xi}^T]^T \in \mathbb{R}^{m+l}$, the linear program eq. (14) is substantially efficient. Linear programming is one of the most thoroughly studied algorithmic fields and many excellent software packages are available for it. We opt for MATLAB to solve this linear program.

After the optimal kernel matrix is learned, we are capable of setting up a classifier like eq. (5). The only change is to replace the regularized spectral kernel K^r in eq. (5) with the learned nonparametric kernel K which is referred to as the *transductive spectral kernel* (TSK) as it carries out transductive inference via introducing the hinge loss in eq. (11). We depict the algorithm for learning a TSK along with semi-supervised classification in Table 1. This algorithm can readily be implemented via invoking a sparse eigenvalue solver and a linear program both of which scale well to hundred thousands of data points including the labeled and unlabeled ones.

4.3 Out-of-Sample Extension

Whatever the theme of learning is, it must be clearly stated that the issues surrounding the learning paradigm are universal, and not just customized to available samples. Since previous nonparametric kernels are rather restrictive to seen points, we show that we can overcome the limitation of non-inductive inference associated with previous nonparametric kernel approaches. By investigating the structure of K, we find out for each pair of points $\mathbf{x}_i$ and $\mathbf{x}_j$ in the dataset it holds that

$$K_{ij} = \sum_{t=1}^{m} \mu_t v_{it} v_{jt} = \bar{V}_{i.} \bar{U} \bar{V}_{j.}^T, \; K = \bar{V} \bar{U} \bar{V}^T, \tag{15}$$

where v_{it} denotes the ith entry of the eigenvector $\mathbf{v}_t$ and $\bar{V}_{i.} = [v_{i1}, \cdots, v_{im}]$ is the ith row vector of the matrix $\bar{V} \in \mathbb{R}^{n \times m}$.

In order to classify an unseen point $\mathbf{x}$, we have to know

$$K(\mathbf{x}, \mathbf{x}_j) = K_{x,j} = \bar{V}_{x.} \bar{U} \bar{V}_{j.}^T \tag{16}$$

for all $\mathbf{x}_j$'s. The kernel function $K(,)$ should be induced from the seen point cloud as well as the learned kernel matrix K. Eq. (16) reveals that induction can be achieved once the spectral representation $\bar{V}_{x.} \in \mathbb{R}^{1 \times m}$ of an unseen point $\mathbf{x}$ is known, in other words, semi-supervised induction can be realized by utilizing the out-of-sample extension trick of spectral clustering. Fortunately, the formalized out-of-sample extension for spectral clustering has been presented in [Bengio *et al.*, 2004], which we use to derive

$$\bar{V}_{x,t} = \frac{1}{1 - \lambda_t} \sum_{i=1}^{n} \frac{W_{x,i}}{\sqrt{D_{xx} D_{ii}}} v_{it}, \quad t = 1, \cdots, m \tag{17}$$

where the inductive weights $W_{x,i} = W(\mathbf{x}, \mathbf{x}_i)$ are similarly computed using eq. (2), and D_{xx} is thus computed by $D_{xx} = \sum_{i=1}^{n} W_{x,i}$. Let $\mathbf{w}_x = [W_{x,1}/\sqrt{D_{xx} D_{11}}, \cdots, W_{x,n}/\sqrt{D_{xx} D_{nn}}]^T \in \mathbb{R}^n$ and then simplify eq. (17) as

$$\bar{V}_{x.} = \mathbf{w}_x^T \bar{V} (I - \bar{\Lambda})^{-1}, \tag{18}$$

where $\bar{\Lambda} = \text{diag}(\lambda_1, \cdots, \lambda_m) \in \mathbb{R}^{m \times m}$.

By doing so, we are able to predict the label for any unseen point $\mathbf{x}$ by

$$f(\mathbf{x}) = \arg \max_{1 \leq k \leq c} \sum_{y_j = k} K_{x,j} = \arg \max_{1 \leq k \leq c} \sum_{y_j = k} \bar{V}_{x.} \bar{U} \bar{V}_{j.}^T, \tag{19}$$

where $\bar{V}_{x.}$ is induced from the point cloud residing on the dataset $\mathcal{X}$, while $\bar{V}$ and $\bar{U}$ has been learned in Step 2 and Step 3 of the presented algorithm, respectively. Finally, we augment the inductive inference eq. (19) in Table 1.

To verify the inductive inference formula eq. (19) as well as guarantee the generalization capability of the TSK K, we claim the following theorem which shows that the TSK used in either transductive or inductive setting is definitely positive semidefinite.

Theorem 2. *If $\lambda_i \neq 1$ for $1 \leq i \leq m$, the expanded symmetric matrix $\widetilde{K} = (K(\mathbf{x}_i, \mathbf{x}_j))_{i,j=1}^{n+s}$ based on n seen samples $\{\mathbf{x}_1, \cdots, \mathbf{x}_n\}$ and s new samples $\{\mathbf{x}_{n+1}, \cdots, \mathbf{x}_{n+s}\}$ is still a kernel matrix.*

Proof. After learning the kernel matrix $K = \bar{V} \bar{U} \bar{V}^T$ on the available dataset, we have to infer $\bar{V}_{(n+1).}, \cdots, \bar{V}_{(n+s).}$ for the coming s new samples to compute $\widetilde{K}$. Corresponding to each new point $\mathbf{x}_{n+t}$, we first obtain the induced weight vector $\mathbf{w}_t$ $(t = 1, \cdots, s)$ and afterwards form a matrix $J = [\mathbf{w}_1, \cdots, \mathbf{w}_s] \in \mathbb{R}^{n \times s}$.

When $1 \leq i, j \leq n$, $\widetilde{K}_{ij} = K_{ij}$. When $n < i \leq n+s$ and $1 \leq j \leq n$, we apply eq. (18) to derive

$$\widetilde{K}_{ij} = \bar{V}_{i\cdot}\bar{U}\bar{V}_{j\cdot}^T = \mathbf{w}_{i-n}^T\bar{V}(I-\bar{\Lambda})^{-1}\bar{U}\bar{V}_{j\cdot}^T.$$

When $1 \leq i \leq n$ and $n < j \leq n+s$, we also have

$$\widetilde{K}_{ij} = \bar{V}_{i\cdot}\bar{U}\bar{V}_{j\cdot}^T = \bar{V}_{i\cdot}\bar{U}(I-\bar{\Lambda})^{-1}\bar{V}^T\mathbf{w}_{j-n}.$$

When $n < i, j \leq n+s$, we can arrive in

$$\widetilde{K}_{ij} = \mathbf{w}_{i-n}^T\bar{V}(I-\bar{\Lambda})^{-2}\bar{U}\bar{V}^T\mathbf{w}_{j-n}.$$

Hence, we complete the expanded matrix $\widetilde{K}$ by

$$\begin{aligned}\widetilde{K} &= \begin{bmatrix} \bar{V}\bar{U}\bar{V}^T & \bar{V}\bar{U}(I-\bar{\Lambda})^{-1}\bar{V}^TJ \\ J^T\bar{V}(I-\bar{\Lambda})^{-1}\bar{U}\bar{V}^T & J^T\bar{V}(I-\bar{\Lambda})^{-2}\bar{U}\bar{V}^TJ \end{bmatrix} \\ &= \begin{bmatrix} \bar{V} & 0 \\ 0 & J^T\bar{V} \end{bmatrix}\begin{bmatrix} \bar{U} & (I-\bar{\Lambda})^{-1}\bar{U} \\ (I-\bar{\Lambda})^{-1}\bar{U} & (I-\bar{\Lambda})^{-2}\bar{U} \end{bmatrix}\begin{bmatrix} \bar{V}^T & 0 \\ 0 & \bar{V}^TJ \end{bmatrix} \\ &= PQP^T, \end{aligned} \quad (20)$$

where $Q \in \mathbb{R}^{2m\times 2m}$ is symmetric and has $2m$ eigenvalues of $\{\mu_i + \frac{\mu_i}{(1-\lambda_i)^2}\}_{i=1}^m$ and m 0's. Because $\mu_i \geq 0$ and $\lambda_i \neq 1$, Q is positive semidefinite; consequently, $\widetilde{K}$ is also positive semidefinite and becomes a valid kernel matrix. ■

5 Experimental Results

In this section, we conduct experiments on real-world datasets to testify our algorithm which we notate as TSK for semi-supervised learning (SSL) using transductive spectral kernels. We compare it with the state-of-the-art SSL algorithms: the Gaussian fields and harmonic functions [Zhu *et al.*, 2003], the local and global consistency [Zhou *et al.*, 2004], and LapSVM [Belkin *et al.*, 2006]. By applying SVM as the final classifier, we also contrast TSK with two competitive spectral kernels, the order-constrained spectral kernel, abbreviated as "OSK" [Zhu *et al.*, 2005], and the fast-decay spectral kernel, abbreviated as "DSK" [Hoi *et al.*, 2006].

In detail, we use four UCI datasets [Asuncion and Newman, 2007] and the WebKB dataset [Sindhwani *et al.*, 2005] as our experimental testbeds. Table 2 describes the fundamental information about these benchmark datasets. Empirically, we fix $\eta = 2$ throughout all our experiments.

Table 2: Dataset Information: the numbers of features, samples and classes.

DATASET	# FEATURES	# SAMPLES	# CLASSES
HEART	13	270	2
IONOSPHERE	34	351	2
SONAR	60	208	2
WINE	13	178	3
WEBKB	4840	1051	2

5.1 UCI Datasets

We experiment seven methods on four UCI datasets: HEART, IONOSPHERE, SONAR and WINE. The input features are normalized to range $[0, 1]$. Results are averaged over 20 random selection of 20 labeled data points of which there is one sample at least for each class. We construct k-NN graphs with $k = 6$ for all datasets. For HEART and WINE, we take $m = 10$ eigenvectors for spectral kernel learning. For IONOSPHERE and SONAR, we take $m = 30$ eigenvectors. The width of the RBF kernel for LapSVM and the initial kernel for DSK learning are set by cross validation. For the harmonic function method, the consistency method, and LapSVM, we also construct the same k-NN graph using eq. (2), which facilitates fair comparison with our method.

Table 3: Transductive classification accuracies on UCI datasets.

CRR (%)	HEART	IONOSPHERE	SONAR	WINE
harmonic	59.86±4.07	76.83±6.84	60.82±6.35	67.78±3.20
consistency	60.30±3.68	70.17±9.07	62.18±6.03	68.39±3.03
TSK	64.06±4.42	77.89±6.37	68.38±4.56	72.28±1.24
LapSVM	73.66±1.60	82.95±1.84	68.24±1.28	93.77±1.10
OSK+SVM	65.88±1.69	83.04±2.10	64.68±1.57	92.72±1.32
DSK+SVM	76.30±1.33	88.55±1.32	71.76±1.07	95.63±0.45
TSK+SVM	**78.40±1.30**	**90.25±2.10**	**72.80±0.95**	**96.34±0.33**

The classification results are shown in Table 3. It can be clearly observed that transductive classification accuracy on the unlabeled samples achieved by TSK consistently outperforms those achieved by the harmonic and consistency methods. TSK+SVM consistently outperforms the other three kernel machines. These results validate that our method uses information from labeled and unlabeled data pertinently to improve the performance of nonparametric kernels.

5.2 WebKB Dataset

WebKB dataset is a subset of web documents of the computer science departments at four universities. This dataset extensively used for semi-supervised learning experiments consists of two categories: course and non-course. For each document, there are two representations: the textual content of the webpage (which we will call page representation) and the anchortext on links on other webpages pointing to the webpage (link representation). We generate bag-of-words feature vectors for both representations. For the page representation, 3000 features were selected according to information gain. For the link representation, 1840 features were generated with no feature selection. The columns of the document-word matrix were scaled based on inverse document frequency weights (IDF) for each word and the resulting term frequency (TF)-IDF feature vectors were length normalized.

We use the two categories as two classes, and their two representations, page and link, for two groups of experiments. This time the cosine kernel is adopted for running SVM. The same 100-NN graph for the harmonic method, the consistency method, and our method is constructed. The regularization parameter α in the consistency method and RSK is fixed at 0.99. We set the number of basic kernels m to 800. We uniformly divide this WebKB dataset into two disjoint sets of which 80% data are used for transductive setting and the rest for inductive setting. In the transductive setting, we compare TSK with 1NN, SVM, the harmonic method, and the consistency method. Otherwise in the inductive setting, we compare TSK with 1NN, SVM, and the regularized spec-

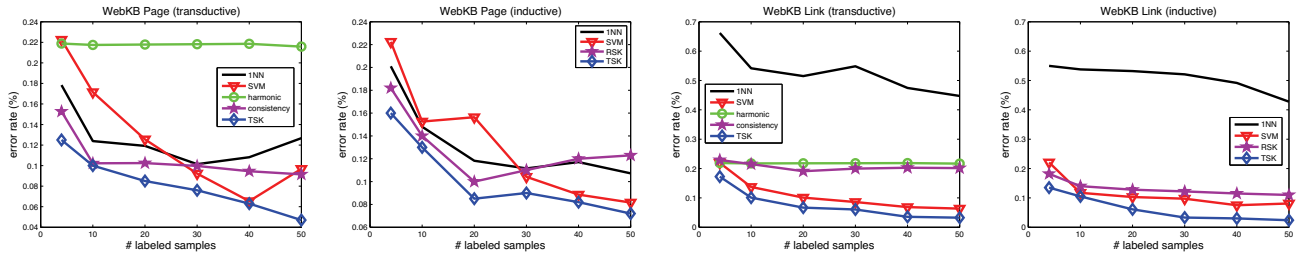

Figure 1: The average transductive and inductive classification error rates on the WebKB dataset composed of two classes.

tral kernel (RSK) stated in Section 3 which is a parametric spectral kernel and may be thought as an inductive version of the consistency method.

Fig. 1 displays comparative results over WebKB. Our TSK-based SSL algorithm is clearly superior to the other algorithms whether in the transductive setting or in the inductive setting.

6 Conclusion

We have proposed an algorithm that efficiently learns a non-parametric kernel called a transductive spectral kernel (TSK) which allows to compute similarities between any pair of seen samples in the available dataset, in the meantime maintain a similarity gap between any pair of seen samples taking different labels, and induce similarities between an unseen sample and any seen sample. A simple yet effective multi-class classifier can be constructed from TSK directly. In addition, TSK works well in conceit with conventional kernel machines such as SVMs. Experiments performed on real-world datasets validate that the learned spectral kernel is quite beneficial to semi-supervised classification.

Acknowledgements

This work was supported by two grants from the Research Grants Council of the Hong Kong SAR, China (Project No. CUHK 414306 and 415408).

References

[Asuncion and Newman, 2007] A. Asuncion and D. J. Newman. {*UCI*} *Machine Learning Repository*. University of California, Irvine, http://www.ics.uci.edu/∼mlearn/MLRepository.html, 2007.

[Belkin *et al.*, 2006] M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: a geometric framework for learning from examples. *Journal of Machine Learning Research*, 7:2399–2434, 2006.

[Bengio *et al.*, 2004] Y. Bengio, J. F. Paiement, P. Vincent, O. Delalleau, N. L. Roux, and M. Ouimet. Out-of-sample extensions for lle, isomap, mds, eigenmaps, and spectral clustering. In *NIPS 16*, 2004.

[Boyd and Vandenberge, 2004] S. Boyd and L. Vandenberge. *Convex Optimization*. Cambridge University Press, Cambridge, UK, 2004.

[Chapelle *et al.*, 2006] O. Chapelle, B. Schölkopf, and A. Zien. *Semi-Supervised Learning*. The MIT Press, Cambridge, MA, 2006.

[Chung, 1997] F. Chung. *Spectral Graph Theory*. CBMS Regional Conference Series in Mathematics, American Mathematical Society, 1997.

[Hein and Maier, 2007] M. Hein and M. Maier. Manifold denoising. In *NIPS 19*, 2007.

[Hoi *et al.*, 2006] S. Hoi, M. Lyu, and E. Chang. Learning the unified kernel machines for classification. In *Proc. KDD*, 2006.

[Kondor and Lafferty, 2002] R. I. Kondor and J. Lafferty. Diffusion kernels on graphs and other discrete input spaces. In *Proc. ICML*, 2002.

[Lanckriet *et al.*, 2004] G. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui, and M. Jordan. Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, 5:27–72, 2004.

[Ng *et al.*, 2002] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *NIPS 14*, 2002.

[Saerens *et al.*, 2004] M. Saerens, F. Fouss, L. Yen, and P. Dupont. The principal components analysis of a graph, and its relationships to spectral clustering. In *Proc. ECML*, 2004.

[Schölkopf and Smola, 2002] B. Schölkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. The MIT Press, Cambridge, MA, 2002.

[Sindhwani *et al.*, 2005] V. Sindhwani, P. Niyogi, and M. Belkin. Beyond the point cloud: from transductive to semi-supervised learning. In *Proc. ICML*, 2005.

[Zhang and Ando, 2006] T. Zhang and R. Ando. Analysis of spectral kernel design based semi-supervised learning. In *NIPS 18*, 2006.

[Zhou *et al.*, 2004] D. Zhou, O. Bousquet, T. Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. In *NIPS 16*, 2004.

[Zhu *et al.*, 2003] X. Zhu, Z. Ghahramani, and J. Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *Proc. ICML*, 2003.

[Zhu *et al.*, 2005] X. Zhu, J. Kandola, Z. Ghahramani, and J. Lafferty. Nonparametric transforms of graph kernels for semi-supervised learning. In *NIPS 17*, 2005.