Pattern Recognition 45 (2012) 1363–1372

Contents lists available at SciVerse ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Supervised neighborhood graph construction for semi-supervised classification

Mohammad Hossein Rohban, Hamid R. Rabiee *

Digital Media Lab, AICTC Research Center, Department of Computer Engineering, Sharif University of Technology, Azadi Street, Tehran, Iran

ARTICLE INFO

Article history:
Received 16 March 2011
Received in revised form
9 July 2011
Accepted 2 September 2011
Available online 22 September 2011

Keywords:
Graph-based semi-supervised learning
Manifold assumption
Neighborhood graph construction

ABSTRACT

Graph based methods are among the most active and applicable approaches studied in semi-supervised learning. The problem of neighborhood graph construction for these methods is addressed in this paper. Neighborhood graph construction plays a key role in the quality of the classification in graph based methods. Several unsupervised graph construction methods have been proposed that have addressed issues such as data noise, geometrical properties of the underlying manifold and graph hyper-parameters selection. In contrast, in order to adapt the graph construction to the given classification task, many of the recent graph construction methods take advantage of the data labels. However, these methods are not efficient since the hypothesis space of their possible neighborhood graphs is limited. In this paper, we first prove that the optimal neighborhood graph is a subgraph of a k'-NN graph for a large enough k', which is much smaller than the total number of data points. Therefore, we propose to use all the subgraphs of k'-NNs graph as the hypothesis space. In addition, we show that most of the previous supervised graph construction methods are implicitly optimizing the smoothness functional with respect to the neighborhood graph parameters. Finally, we provide an algorithm to optimize the smoothness functional with respect to the neighborhood graph in the proposed hypothesis space. Experimental results on various data sets show that the proposed graph construction algorithm mostly outperforms the popular k-NN based construction and other state-of-the-art methods.

1. Introduction

Different semi-supervised methods use different prior knowledge to take advantage of unlabeled data in the learning process. Two known such assumptions are the cluster and manifold assumptions. In the latter assumption, the label function will change smoothly on the data manifold. In order to model the manifold smoothness functional, neighborhood graphs are constructed from the unlabeled data [1]. Necessary operators for expression of the smoothness functional, such as the Laplace–Beltrami operator, can be estimated using the adjacency matrix of the neighborhood graphs. As a result, the neighborhood graph construction plays an important role in the methods that employ the manifold assumption.

Several neighborhood graph construction schemes have been proposed in recent years [2–4]. In these schemes, each node of the graph corresponds to a labeled or unlabeled data point. Each scheme should guarantee sparse graph construction to ensure that the semi-supervised learner runs efficiently. Graph construction methods can be divided into two groups of supervised and unsupervised schemes. Unsupervised methods do not use the labels of the labeled data in the construction scheme. In contrast with these methods, supervised schemes use data labels to optimize the graph structure to better fit the classification task.

Among the most simple unsupervised construction schemes are the ε thresholding and k-NN methods. In ε thresholding each node is connected to the nodes that are within its ε Euclidean distance. This thresholding is prone to generating disconnected or almost-complete graphs. On the other hand, in the k-NN approach undirected links are made between each node to its k nearest neighbors. Therefore, the k-NN approach has the advantage of being robust to problems occurred in the case of choosing an inappropriate fixed threshold.

Numerous unsupervised methods are proposed to improve the k-NN graph construction in different aspects [2,3,5,6]. Because of making undirected links, the k-NN method may produce nodes with unbalanced number of edges. To alleviate this problem, the b-matching method is proposed in [2], to produce a balanced graph, i.e. each node will have the same number of edges. The noise in data is another issue which was addressed to improve the k-NN method, since the pairwise Euclidean distance is highly sensitive to noise. The method of manifold denoising uses the diffusion process on the data points, rather than the labels, on the

* Corresponding author. Tel.: +98 21 66166683; fax: +98 21 66047662.
E-mail addresses: rahban@ce.sharif.edu (M.H. Rohban), rabiee@sharif.edu (H.R. Rabiee).

0031-3203/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2011.09.001

initial k-NN graph to remove the noise of the data [5]. The k-NN graph construction and diffusion process on the data points are performed alternatively to denoise the manifold. Although manifold denoising makes the classical methods robust against the noise, the problem of non-adaptive selection of the neighborhood parameters remains unsolved. Motivated by the methods of sparse coding, ℓ^1 norm reconstruction was used to produce an adaptive representation of the data points that is both sparse and more robust to noise than the ε thresholding and k-NN methods [3]. In this scheme, each data point is reconstructed as a linear combination of other data points while minimizing the ℓ^1 norm of the reconstruction coefficients. Each node will be linked to the nodes with non-zero coefficients of its reconstruction. In addition, unlike the classical methods, this scheme has the advantage of choosing the number of neighbors for each data, adaptively. In an alternative adaptive scheme, estimation of the linear error of data point reconstruction is used to identify whether a nearest neighbor of a point resides on the manifold tangent space at that point [6]. Each point is linked only to its nearest neighbors that reside on the manifold tangent space at that point. Therefore, k will be large in a point if the manifold curvature is low in that point and vice versa.

Since the mentioned unsupervised methods do not take advantage of the data labels for graph construction, the noise removal and parameter selection quality is quite problem-dependent. For example although the b-matching method provides considerable improvement on the TEXT data set, it will make little improvement on a digit recognition data set (USPS). A number of graph improvement schemes addressed this problem [4,7–9]. In [9], leave-one-out (LOO) error of the classifier is used as the objective function to tune the bandwidths of the Gaussian edge weighting function. The optimization of the LOO objective function may lead to ill-conditioned solutions such as disconnected neighborhood graphs. Therefore, the authors proposed to add a regularization term to the LOO error to exclude degenerate graphs from the possible solutions. This regularization term is the variance of the inverse bandwidths. The minimization of this term will avoid abnormal bandwidths, and hence degenerate solutions are excluded. In fact the regularization term restricts the hypothesis space of the possible neighborhood graphs to make the graph construction well-posed. Although this scheme results in improvement of the accuracy of classification on many data sets, the bandwidth parameter selection is global and hence the optimal neighborhood graph may not be included in the possible set of solutions. That is, the graph hypothesis space is too limited. Similar method is proposed by [10] for supervised graph construction. The Mahalanobis based distance learning is also applied in the Gaussian edge weighting. This method uses a self-training approach in a graph based classification method. The labels of the unlabeled data are estimated using the classification method and then used for distance learning. The graph construction using the learned metric and label estimation are performed alternatively, until the method converges. Since the learned metric is global in the feature space, it has the similar limitations of the LOO minimization method. In addition, the Mahalanobis metric learning algorithms, use the computationally demanding semi-definite programming (SDP) methods to learn the parameters of the metric. This will cause the metric learning methods to be impractical for large data sets.

Sharpening of the graph edges is proposed to maximize the smoothness of the optimal label function with respect to the weights of the graph [4]. It has been shown that similar to the LOO method, the optimization of the weights without restricting the graph hypothesis space may lead to the degenerate solutions. The authors proposed an ad hoc solution by disconnecting the edges connecting any two labeled points and the edges that are from an unlabeled data to a labeled data. Since the authors forced the optimal graph to be very similar to the initial graph, the hypothesis space will be too limited.

Methods of spectral kernel learning are proposed to build a task specific kernel from the Laplacian matrix [8]. A classical construction method may be used to produce an initial graph. The Laplacian operator on the graph is then decomposed into its eigenvectors. The method then takes advantage of the labeled data to change the kernel eigenvalues in order to minimize the Frobenius distance of the kernel induced by the Laplacian matrix with the optimal kernel obtained by the labeled data [8]. It is important to note that the eigenvalue transformation is monotonic in order to make the kernel consistent with the smoothness assumption. Since the eigenvectors of the graph Laplacian matrix will not change, the structure of the graph may not be subject to the necessary change in the initial graph. Therefore, like the previous supervised methods, the hypothesis space is limited in this method too.

Using Gaussian Process (GP) to model the label function, the parameters of the neighborhood graph can be learned using the maximization of the expectation of the marginal likelihood through the EM algorithm [7]. Since the objective function is not concave the EM may lead to suboptimal solutions. This problem will get worse and the solution will be sensitive to the initial solution if the number of optimization variables is increased in the case of edge weights learning.

In most of the mentioned supervised methods, in order to make the problem well-posed, the neighborhood graph hypothesis space is too restricted. This restriction may cause the optimal graph to be excluded from the hypothesis space. The paper is aimed to show that the optimal graph is a subgraph of a k'-NN graph, for a large enough k', and then perform the optimization on this set. Therefore, the hypothesis space will be the subgraphs of the k'-NN graphs.

We will first show that the objective functions of three supervised methods consisting of the graph edge sharpening (GES) [4], spectral kernel learning (SKL) [8] and marginal likelihood (ML) [7] are almost the same. Then we propose to use the expected marginal likelihood as an objective function. The advantage of this objective function is that it can be expressed as a linear form consisting of all related edge weights. This objective function is then optimized with respect to the edge weights. Since the number of edge weights is high, instead of using EM, we propose a method to estimate the expectation of the marginal likelihood. The estimated objective is then optimized using an optimal greedy method. The proposed method is then compared with the popular k-NN, ML and SKL methods of graph construction. To investigate whether the graph improvement methods may further reduce the error rate, the SKL and ML methods are applied on both the k-NN and the proposed method graphs.

The rest of the paper is organized as follows. In Section 2, the basics and notations of the graph based semi-supervised learning is described. In Section 3, the three prominent supervised graph improvement methods, namely GES, SKL and ML are analyzed. In Section 4, the proposed method is described. In Section 5, the proposed method is compared with other methods. Finally, the paper is concluded in Section 6.

2. Basics and notations

Consider $\mathbf{y}=\{y_1,\ldots,y_l\}$ as the data labels, and $X_l=\{\mathbf{x}_1,\ldots,\mathbf{x}_l\}$ and $X_u=\{\mathbf{x}_{l+1},\ldots,\mathbf{x}_{l+u}\}$ as the labeled and unlabeled data sets, respectively. We assume that $X=X_l\cup X_u$ and $\mathbf{y}$ are given. The problem is to find an estimate $\mathbf{f}=\{f_1,\ldots,f_{l+u}\}$ for $\mathbf{y}=\{y_1,\ldots,y_{l+u}\}$, known as the label function. It is shown that without any prior knowledge on the label function, this estimation problem is ill-posed [11,12].

The manifold assumption is a known prior knowledge on the label function in the field of the semi-supervised learning [13].

It states that the desired label function should be smooth over the manifold of the data. In other words, two nearby points should have similar labels. Smoothness of a label function $\mathbf{f}$ will be high if $S(\mathbf{f})$ with the following definition is low:

$$S(\mathbf{f}) = \sum_{i,j=1}^{l+u} \mathbf{W}_{ij}(f_i - f_j)^2 \tag{1}$$

where $\mathbf{W}_{ij} = \exp\{-\|\mathbf{x}_i - \mathbf{x}_j\|^2/2\sigma^2\}$ for all i and j for which either $\mathbf{x}_i$ is among the k-NN of $\mathbf{x}_j$ or vice versa, otherwise $\mathbf{W}_{ij}$ is zero.

$\mathbf{W}$ represents the adjacency matrix of the neighborhood graph. Given the neighborhood graph Laplacian $\mathbf{L}=\mathbf{D}-\mathbf{W}$, where $\mathbf{D}$ is the diagonal matrix of degree of each node with $\mathbf{D}_{ii} = \sum_j \mathbf{W}_{ij}$, it is easy to show that Eq. (1) can be rewritten as [1]

$$S(\mathbf{f}) = \sum_{i,j=1}^{l+u} \mathbf{L}_{ij} f_i f_j = \mathbf{f}^\top \mathbf{L}\mathbf{f} \tag{2}$$

Let $\lambda_1 \le \lambda_2 \le \cdots \le \lambda_{l+u}$ and $\{\mathbf{v}_1, \ldots, \mathbf{v}_{l+u}\}$ be the spectral decomposition of the Laplacian matrix. Then using $\mathbf{L} = \sum_{i=1}^{l+u} \lambda_i \mathbf{v}_i \mathbf{v}_i^\top$, Eq. (2) can be stated as

$$S(\mathbf{f}) = \sum_{i=1}^{l+u} \lambda_i (\mathbf{v}_i^\top \mathbf{f})^2 \tag{3}$$

It is easy to show that the value of λ_i is corresponding to the smoothness of the eigenvector $\mathbf{v}_i$ [1]. Therefore, the last equation shows that $S(\mathbf{f})$ will increase if $\mathbf{f}$ is similar to the non-smooth eigenvectors, i.e. the eigenvectors with high value of the eigenvalue.

The manifold assumption then can be employed in the Tikhonov regularization method to find an estimate of the label function:

$$\min_{\mathbf{f}} \|\mathbf{C}\mathbf{f} - \mathbf{y}\|^2 + \gamma \mathbf{f}^\top \mathbf{L}\mathbf{f} \tag{4}$$

where $\mathbf{C}$ is a selection matrix with $\mathbf{C} = (\mathbf{I}_{l\times l}\ \mathbf{0}_{l\times u})$. Equating the derivative of the objective function to zero, we may obtain the optimal label function as

$$\mathbf{f} = (\mathbf{C}^\top \mathbf{C} + \gamma \mathbf{L})^{-1} \mathbf{C}^\top \mathbf{y} \tag{5}$$

3. Supervised graph improvement methods

In the GES method [4], it is shown that the smoothness functional $S(\mathbf{f})$ for the optimal label function can be stated as

$$S(\mathbf{f}) = \mathbf{y}^\top \mathbf{y} - \mathbf{y}_e^\top (\mathbf{I} + \gamma \mathbf{L})^{-1} \mathbf{y}_e \tag{6}$$

in the case that the Tikhonov regularization is done in the following way:

$$\min_{\mathbf{f}} \|\mathbf{f} - \mathbf{y}_e\|^2 + \gamma \mathbf{f}^\top \mathbf{L}\mathbf{f} \tag{7}$$

where $\mathbf{y}_e = (\mathbf{y}^\top\ \mathbf{0}_{1\times u})^\top$. The optimal graph should minimize this functional. That is, according to the manifold assumption it is expected that the neighborhood graph will be a graph over which the label function is smooth. Hence the problem can be stated as

$$\max_{\mathbf{G}} \mathbf{y}_e^\top (\mathbf{I} + \gamma \mathbf{L})^{-1} \mathbf{y}_e \quad \text{s.t. } \mathbf{W}_{ij} \ge 0 \tag{8}$$

where $\mathbf{G}$ is the set of neighborhood graph parameters. Clearly, the eigenvectors of $(\mathbf{I}+\gamma\mathbf{L})^{-1}$ will be the same as the eigenvectors of $\mathbf{L}$. However, the eigenvalues λ_i will be changed to $1/(1+\gamma\lambda_i)$. Therefore, the objective function of Eq. (8) can be rewritten in terms of the eigenvalues and eigenvectors of the Laplacian matrix as follows:

$$\max_{\mathbf{G}} \sum_{i=1}^{l+u} \frac{1}{1+\gamma\lambda_i} (\mathbf{v}_i^\top \mathbf{y}_e)^2 \tag{9}$$

Maximizing the objective function will maximize the smoothness of the label function $\mathbf{y}_e$, because the objective will increase if $\mathbf{y}_e$ is similar to the smooth eigenvectors.

It is easy to show that one of the optimal solutions of (8) is a graph with a diagonal adjacency matrix, which is a degenerate solution. The authors in [4] proposed an ad hoc solution for (8) which is a valid graph:

$$\mathbf{W}_{opt} = \begin{pmatrix} \text{diagonal matrix} & 0 \\ \mathbf{W}_{ul} & \mathbf{W}_{uu} \end{pmatrix} \tag{10}$$

where u and l subscripts are corresponded to the labeled and unlabeled indexes. It is noticeable that the proposed optimal solution does not depend on the labeled data. This means that GES is not really a task specific graph construction.

In the ML method [7], it is assumed that the label function $\mathbf{f}$ is a Gaussian stochastic process with $\mathbf{f} \sim \mathbf{N}(\mathbf{0}, \mathbf{L}^+)$ with $y_i = f_i + \varepsilon_i$ and $\varepsilon \sim \mathbf{N}(0, \sigma^2\mathbf{I})$, where $\mathbf{L}^+$ is the pseudo-inverse of the Laplacian matrix. It is easy to show that $\mathbf{y}$ will be a Gaussian stochastic process with $\mu_{\mathbf{y}} = \mathbf{0}$ and $\Sigma_{\mathbf{y}} = \mathbf{L}_l^+ + \sigma^2\mathbf{I}$ [14]. The maximization of the marginal likelihood $p(\mathbf{y}|X,\Theta)$ with respect to Θ will yield a maximum likelihood estimation of the parameter Θ. In the graph improvement problem, Θ may be the parameters of the weight function of the edges (for example the bandwidth of the Gaussian weighting) and the parameter of the Laplacian eigenvalues transformation. The marginal likelihood will be [7]

$$\log p(\mathbf{y}|X,\Theta) = -\frac{1}{2}\mathbf{y}^\top (\mathbf{L}_l^+ + \sigma^2\mathbf{I})^{-1}\mathbf{y} - \frac{1}{2}\log\det(\mathbf{L}_l^+ + \sigma^2\mathbf{I}) - \frac{l}{2}\log 2\pi \tag{11}$$

Let $\mathbf{y}_t = (y_1, \ldots, y_l, y_{l+1}, \ldots, y_{l+u})$ be the ground truth value of the label function. Then the marginal likelihood of $\mathbf{y}_t$ will be a random variable. As an alternative approach, the expected value of this random variable may be maximized with respect to the unknown parameters:

$$\mathbb{E}[\log p(\mathbf{y}_t|X,\Theta)] = -\frac{1}{2}\mathbb{E}[\mathbf{y}_t]^\top (\mathbf{L}^+ + \sigma^2\mathbf{I})^{-1}\mathbb{E}[\mathbf{y}_t] - \frac{1}{2}\log\det(\mathbf{L}^+ + \sigma^2\mathbf{I}) - \frac{l+u}{2}\log 2\pi \tag{12}$$

where all the expectations are given with respect to $p(y_t|X,\mathbf{y},\Theta)$. The latter optimization problem has the advantage that rather than $\mathbf{L}_l^+$, which is the labeled part of the kernel matrix, $\mathbf{L}^+$ will appear in the objective function. This makes it possible to optimize all the entries of the kernel matrix $\mathbf{L}^+$. The first term in Eq. (11) can be considered as the smoothness of the label function $\mathbf{y}$:

$$\mathbf{y}^\top (\mathbf{L}_l^+ + \sigma^2\mathbf{I})^{-1}\mathbf{y} = \sum_{i=1}^{l} \frac{1}{\mu_i^{(l)} + \sigma^2} (\mathbf{v}_i^{l\top}\mathbf{y})^2 \tag{13}$$

where $\{\mu_i^{(l)}, \mathbf{v}_i^l | 1 \le i \le l\}$ is the spectral decomposition of $\mathbf{L}_l^+$. We may note that $\mathbf{L}^+$ is a kernel matrix consistent with the manifold assumption. Therefore, $\mathbf{L}_l^+$ is the induced kernel of $\mathbf{L}^+$ on the labeled points. As a result, greater $\mu_i^{(l)}$ corresponds to smoother $\mathbf{v}_i^l$. Since $1/(\mu_i^{(l)} + \sigma^2)$ is a decreasing function of smoothness, comparing this equation and Eq. (3) shows that the objective function will penalize more if the label function $\mathbf{y}$ is similar to the non-smooth eigenvectors of $\mathbf{L}_l^+$.

In SKL method [8], the spectral decomposition of the kernel matrix $\mathbf{L}^+$ is obtained as $\{\mu_1, \ldots, \mu_{l+u}\}$ and $\{\mathbf{v}_1, \ldots, \mathbf{v}_{l+u}\}$ [8]. Therefore, we have $\mathbf{K} = \sum_{i=1}^{l+u} \mu_i \mathbf{v}_i \mathbf{v}_i^\top$. The values of μ_i are then optimized such that the Frobenius angle of the kernel matrix for the labeled data and the optimal kernel $\mathbf{T} = \mathbf{y}\mathbf{y}^\top$ is minimized. The value of μ_i is inverse proportional to the smoothness of the basis function $\mathbf{v}_i$. The optimization is constrained to conserve the order of smoothness of the eigenvectors. To avoid over-fitting, the trace of the kernel matrix will be constrained to be one:

$$\max_{\mu_i} \quad \langle \mathbf{K}_l, \mathbf{T} \rangle$$

$$\text{s.t.}\quad \mathbf{K}=\sum_{i=1}^{l+u}\mu_i\mathbf{v}_i^l\mathbf{v}_i^{l\top},\quad \mu_i\geq 0$$
$$\text{tr}(\mathbf{K})=1,\quad \mu_i\geq\mu_{i+1},\ \forall i\in\{1,\ldots,l+u-1\} \tag{14}$$

Rewriting of Eq. (14) will yield

$$\begin{aligned}\langle \mathbf{K}_l,\mathbf{y}\mathbf{y}^\top\rangle_F &= \text{tr}\left(\sum_{i=1}^{l+u}\mu_i\mathbf{v}_i^l\mathbf{v}_i^{l\top}\mathbf{y}_e\mathbf{y}_e^\top\right)\\ &=\sum_{i=1}^{l+u}\mu_i\ \text{tr}[(\mathbf{v}_i^{l\top}\mathbf{y}_e)(\mathbf{y}_e^\top\mathbf{v}_i^l)]\\ &=\sum_{i=1}^{l+u}\mu_i(\mathbf{v}_i^{l\top}\mathbf{y}_e)^2\end{aligned} \tag{15}$$

where we have considered the fact that tr(**ABCD**) = tr(**BCDA**) and $\mathbf{v}_i^l$ is the labeled part of the ith eigenvector. It is interesting to observe that similar to the GES and ML methods, SKL also optimizes the smoothness functional for graph construction, implicitly. The main difference between the previous supervised methods is that they use different optimization variables. In GES and ML, parameters of the weight function of the graph edges are subject to the optimization, whereas in SKL the eigenvalues of the kernel matrix are the optimization variables. In addition, the argument to the smoothness function may be changed in various methods. In SKL and GES methods, the smoothness of the label function $\mathbf{y}$ is the objective function. However, in ML method the objective function can be the expected value of the smoothness of $\mathbf{y}_t$, which is the ground truth labels of all the points. Furthermore, to avoid degenerate solutions all these methods restrict the graph hypothesis space. In GES, the desired graph is forced to be the same as the k-NN graph with some edges changed to be uni-directed and the edges between the labeled points be removed. In SKL, the structure of the graph remains almost the same as the k-NN graph, since the eigenvectors of the graph Laplacian remain unchanged. In ML, the possible set of graphs is the subgraphs of k-NN graphs where the edges with the end point distance greater than a threshold are weakened due to the exponential edge weighting.

4. The proposed method

In this section, we first propose an appropriate linear objective function for graph construction. To make the optimization problem well-posed, a prior knowledge for the structure of the neighborhood graph is proposed and justified. Finally, the algorithm to solve the proposed optimization problem is provided.

4.1. The proposed objective function

We propose to maximize expectation of the smoothness of the ground truth labels with respect to the graph edge weights:

$$\min_{\mathbf{W}}\quad \mathbb{E}_{\mathbf{y}_t|X,\mathbf{y}}\left[\sum_{i,j}\mathbf{W}_{ij}(y_{t,i}-y_{t,j})^2\right]$$
$$\text{s.t.}\quad \mathbf{W}_{ij}\geq 0 \tag{16}$$

This formulation has the advantage that in contrast to the SKL and GES methods, the objective function can be posed as a linear function of all the edge weights, and hence the solution to the optimization of this problem can be easily obtained through closed form solutions. The objective function of the SKL method (Eq. (15)) is inversely proportional to the smoothness of the $\mathbf{y}_e$, which contains only the labels of the labeled data points. Therefore, since the labels of the unlabeled points are considered to be zero in $\mathbf{y}_e$, to state the objective function of SKL in a linear form similar to (16), only the weights of the edges from labeled points to themselves and the edges from labeled points to the unlabeled points will appear in the objective function. The same problem will happen to the GES objective function, which is also inversely proportional to the smoothness of $\mathbf{y}_e$.

If we consider a 2 class problem, where $y_i\in\{-1,1\}$, Eq. (16) can be stated as

$$\min_{\mathbf{W}}\quad \sum_{i,j}\mathbf{W}_{ij}\mathbb{E}_{\mathbf{y}_t|\mathbf{x},\mathbf{y}}[(y_{t,i}-y_{t,j})^2]=\sum_{i,j}4\mathbf{W}_{ij}\ \Pr(y_{t,i}\neq y_{t,j}|\mathbf{x}_i,\mathbf{x}_j,\mathbf{y})$$
$$\text{s.t.}\quad \mathbf{W}_{ij}\in\{0,1\} \tag{17}$$

The solution to this problem may not be a symmetric matrix. We use the new matrix $\mathbf{W}^*=\max(\mathbf{W},\mathbf{W}^\top)$ to remedy this problem.

To solve the proposed optimization problem we need to estimate $p_{i,j}=\Pr(y_{t,i}\neq y_{t,j}|\mathbf{x}_i,\mathbf{x}_j,\mathbf{y})$. As we will show the estimation of $p_{i,j}$ may be harder or as hard as the estimation of the optimal Bayes labels of the unlabeled points.

Lemma 1. *Using the maximum likelihood estimation (MLE) of $\mathcal{P}=\{p_{i,j}|1\leq i,j\leq l+u\}$, it is possible to find the MLE of the optimal (Bayes) labels of the unlabeled data X_u.*

Proof. Consider an arbitrary unlabeled point $\mathbf{x}_k$, $l+1\leq k\leq l+u$, with its label random variable Y_k. Then we have

$$\begin{aligned}p_{k,1}&=\Pr(Y_k\neq Y_1|\mathbf{x}_1,\mathbf{x}_k,Y_1=y_1,\ldots,Y_l=y_l)\\ &=\Pr(Y_k\neq y_1,Y_1=y_1|\mathbf{x}_1,\mathbf{x}_k,Y_1=y_1,\ldots,Y_l=y_l)\\ &\quad+\Pr(Y_k\neq -y_1,Y_1=-y_1|\mathbf{x}_1,\mathbf{x}_k,Y_1=y_1,\ldots,Y_l=y_l)\\ &=\Pr(Y_k\neq y_1|\mathbf{x}_1,\mathbf{x}_k,Y_1=y_1,\ldots,Y_l=y_l)\end{aligned} \tag{18}$$

The optimal Bayes label of $\mathbf{x}_k$ can be stated as below:

$$y_k^*=\begin{cases}-y_1, & \Pr(Y_k\neq y_1|\mathbf{x}_1,\mathbf{x}_k,Y_1=y_1,\ldots,Y_l=y_l)\geq 0.5\\ y_1 & \text{otherwise}\end{cases} \tag{19}$$

Using (18), Eq. (19) can be stated as

$$y_k^*=-y_1(2\mathbb{I}(p_{k,1}-0.5\geq 0)-1)=g(p_{k,1}) \tag{20}$$

where $\mathbb{I}(\cdot)$ is the indicator function. Since y_k^* is a function of $p_{k,1}$, we may use the invariance property of MLE [15] to find the MLE of y_k^*:

$$y_k^{mle}=g(p_{k,1}^{mle}) \tag{21}$$

Therefore, if we have the MLE of $p_{k,1}$ we can find the MLE of y_k^*. However, since g is not a one-to-one function, the converse is not true. □

Maximum likelihood estimation generally shows nice properties only when the number of labeled samples is high. For example under certain regularity conditions on the probability density function, it may be shown that MLE is an efficient estimator [15]. Lemma 1 may be useful in the case that a large number of labeled data is available. However, in real world problems this is not the case. To consider these cases, asymptotic convergence rate of the consistent estimators of $p_{k,1}$ and y_k^* are compared in Lemma 2.

Lemma 2. *If $\hat{\theta}$ is a consistent estimator of $p_{1,k}$ with the convergence rate of $a(l,\varepsilon)$, with l being the number of labeled samples, then there exists a consistent estimator of the optimal Bayes label of $\mathbf{x}_k$ with the convergence rate of $\mathcal{O}(a(l,\varepsilon))$.*

Proof. Since $\hat{\theta}$ is a consistent estimator of $p_{1,k}$ we have

$$\forall\varepsilon>0:\Pr(|\hat{\theta}-p_{1,k}|\leq\varepsilon)\geq 1-a(l,\varepsilon),$$
$$\lim_{l\to\infty}a(l,\varepsilon)=0 \tag{22}$$

for a function $a(\cdot,\cdot)$, and l is the number of labeled data. Now consider the estimator $\hat{\alpha}=-y_1(2\mathbb{I}(\hat{\theta}\geq 0.5)-1)$. We have

$$\forall 0<\varepsilon<|p_{1,k}-0.5|$$

$$\Pr(|\hat{\alpha}-y_k^*|\leq\varepsilon)=\begin{cases}\Pr(\hat{\theta}\in(0.5,+\infty)), & p_{1,k}\geq 0.5\\ \Pr(\hat{\theta}\in(-\infty,0.5)), & p_{1,k}<0.5\end{cases}$$
$$\geq\Pr(|\hat{\theta}-p_{1,k}|\leq\varepsilon)\geq 1-a(l,\varepsilon) \quad (23)$$

Therefore, $\hat{\alpha}$ will be a consistent estimator for y_k^* with the convergence rate of $\mathcal{O}(a(l,\varepsilon))$. □

According to Lemma 2, using an estimation of $p_{k,1}$ for graph construction fails. That is, to solve the graph construction problem for label estimation, we have to solve a problem as hard as (or even harder than) the main problem. In a similar manner, the authors in [16] have used SVM to estimate $p_{i,j}$, implicitly. This method clearly fails when the original problem may not be solved efficiently by SVM. For example, when the number of training samples are low, SVM with linear or RBF kernels fails. However, the main problem may be solved efficiently using a stronger regularization such as manifold based methods.

To remedy this problem, we use a prior knowledge on the graph structure. In this case, estimation of $p_{i,j}$ may be a more specific problem than its estimation in the unconstrained neighborhood graph, since we have $p_{i,j}=\Pr(y_{t,i}\neq y_{t,j}|\mathbf{x}_i,\mathbf{x}_j,\mathbf{y},\mathcal{R}(\mathbf{x}_i,\mathbf{x}_j))$, where $\mathcal{R}$ is the prior knowledge on the neighborhood graph.

We will show in the next subsection that the optimal neighborhood graph will be a subgraph of the k'-NN graph, for a large enough $k'\ll l+u$. Using this prior knowledge, the optimization problem (17) can be stated as follows:

$$\min_{\mathbf{W}_{ij}\in\{0,1\}}\ \sum_{i,j}4\mathbf{W}_{ij}\Pr(y_{t,i}\neq y_{t,j}|\mathbf{x}_i,\mathbf{x}_j,\mathbf{y},\{\mathbf{x}_i\in N_{k'}(\mathbf{x}_j)\vee\mathbf{x}_j\in N_{k'}(\mathbf{x}_i)\})$$
$$\text{s.t. }\forall i,j:\mathbf{x}_j\notin N_{k'}(\mathbf{x}_i)\wedge\mathbf{x}_i\notin N_{k'}(\mathbf{x}_j):\mathbf{W}_{ij}=0,$$
$$\forall i\ \sum_{\mathbf{x}_j\in N_{k'}(\mathbf{x}_i)\vee\mathbf{x}_i\in N_{k'}(\mathbf{x}_j)}\mathbf{W}_{ij}=k \quad (24)$$

where $N_k(\mathbf{x}_i)$ is the set of k-NNs of $\mathbf{x}_i$. It is noticeable that Eq. (18) does not hold in this case. Since there may be no labeled point in the nearest neighbors of an arbitrary unlabeled point $\mathbf{x}_k$, and $\mathbf{x}_k$ may not be included in the nearest neighbors of any labeled point. The optimization problem in (24) can be solved using a greedy method, by connecting the ith node to the nodes j, where $p_{i,j}$ is among the k smallest values of $\{p_{i,m}|\mathbf{x}_m\in N_{k'}(\mathbf{x}_i)\vee\mathbf{x}_i\in N_{k'}(\mathbf{x}_m)\}$. Since the proposed objective function can be solved using a greedy method, we can show that if we set $p_{i,j}\propto\|\mathbf{x}_i-\mathbf{x}_j\|^2$, the k-NN graph will be the optimal graph. In this case, for each node i, $\mathbf{W}_{ij}$ will be set to one for nodes j that the value of $\|\mathbf{x}_i-\mathbf{x}_j\|^2$ is among the k minimum values of $\{\|\mathbf{x}_i-\mathbf{x}_j\|^2;1\leq j\leq l+u\}$. Actually we may better estimate $p_{i,j}$ by considering the data labels. We propose to estimate the values of $p_{i,j}$ using a discriminative classification approach such as SVM [17]. This estimation process can be seen as a link classification. It classifies the links $(\mathbf{x}_i,\mathbf{x}_j)$ between nearest neighbor points into two classes of inter- and intra-class links. For the inter-class links we have $p_{i,j}=\Pr(y_{t,i}\neq y_{t,j}|\mathbf{x}_i$ and $\mathbf{x}_j$ are among the $k'-$ NNs of each other$)\geq 0.5$. Similarly we have $p_{i,j}<0.5$ for intra-class links.

4.2. The prior knowledge on the neighborhood graph

To find an appropriate prior knowledge on the graph, we need to define the optimal neighborhood graph.

Definition 1. Let $N_\varepsilon^{\mathcal{M}}(\mathbf{x}_i)=\{\mathbf{x}_j:d_{\mathcal{M}}(\mathbf{x}_i,\mathbf{x}_j)\leq\varepsilon\}$ be the ε neighborhood of $\mathbf{x}_i$ on the manifold $\mathcal{M}$, where $d_{\mathcal{M}}$ is the geodesic distance on $\mathcal{M}$. In addition, let $N_\varepsilon^{\mathbb{R}^m}(\mathbf{x}_i)=\{\mathbf{x}_j:\|\mathbf{x}_i-\mathbf{x}_j\|_2\leq\varepsilon\}$ be the ε neighborhood of $\mathbf{x}_i$ in the ambient space $\mathbb{R}^m$.

Optimal neighborhood graph may be defined based on $N_\varepsilon^{\mathcal{M}}(\mathbf{x}_i)$, rather than $N_\varepsilon^{\mathbb{R}^m}(\mathbf{x}_i)$. That is in the optimal graph, each node $\mathbf{x}_i$ will be connected to nodes in $N_\varepsilon^{\mathcal{M}}(\mathbf{x}_i)$. We are going to recover the neighborhood graph of the data manifold in this paper. It should be mentioned that the notation of the neighborhood refers to the distances of the points on the manifold rather than the ambient space. Moreover, it is noticeable that the neighborhood graph is used to approximate the smoothness integral on the manifold [1]. As a result, edges of this graph should be connected based on the neighborhood on the manifold. Therefore, we should use $N_\varepsilon^{\mathcal{M}}(\mathbf{x}_i)$, rather than $N_\varepsilon^{\mathbb{R}^m}(\mathbf{x}_i)$.

We will show that under the high sampling rate of the manifold, with high probability, the optimal neighborhood graph is a subgraph of an $\varepsilon+\varepsilon'$-ball graph, based on the Euclidean distances in the ambient space $\mathbb{R}^m$.

Theorem 1. *Given $\varepsilon,\varepsilon',\mu>0$, there exists a large enough sampling rate of the manifold $\mathcal{M}$ such that* $\Pr(\forall\mathbf{x}_i:N_\varepsilon^{\mathcal{M}}(\mathbf{x}_i)\subseteq N_{\varepsilon+\varepsilon'}^{\mathbb{R}^m}(\mathbf{x}_i))\geq 1-\mu$.

Proof. Authors in [18] have shown that for given $\lambda,\mu>0$ there exists a sampling of $\mathcal{M}$ with rate $\alpha(\lambda,\mu)$ such that

$$\Pr\left(\forall\mathbf{x}_i,\mathbf{x}_j:\left|1-\frac{d_G(\mathbf{x}_i,\mathbf{x}_j)}{d_{\mathcal{M}}(\mathbf{x}_i,\mathbf{x}_j)}\right|\leq\lambda\right)\geq 1-\mu \quad (25)$$

where $d_G(\mathbf{x}_i,\mathbf{x}_j)$ is the length of the shortest path between $\mathbf{x}_i$ and $\mathbf{x}_j$ in the neighborhood graph G, with the edges labeled by the Euclidean distance of the edges end points. We have

$$1-\mu\leq\Pr(\forall\mathbf{x}_i,\mathbf{x}_j:|d_G(\mathbf{x}_i,\mathbf{x}_j)-d_{\mathcal{M}}(\mathbf{x}_i,\mathbf{x}_j)|\leq\lambda d_{\mathcal{M}}(\mathbf{x}_i,\mathbf{x}_j))$$
$$\leq\Pr(\forall\mathbf{x}_i,\mathbf{x}_j:|d_G(\mathbf{x}_i,\mathbf{x}_j)-d_{\mathcal{M}}(\mathbf{x}_i,\mathbf{x}_j)|\leq M\lambda) \quad (26)$$

where $M=\sup_{\mathbf{x}_i,\mathbf{x}_j\in\mathcal{M}}d_{\mathcal{M}}(\mathbf{x}_i,\mathbf{x}_j)$ and assuming the manifold is bounded. Let $\lambda'=M\lambda$.

Now consider the related sampling rate $\alpha(\varepsilon',\mu)$ for which Eq. (26) holds for $\lambda'=\varepsilon'$. Let $N_i=\{\mathbf{x}_j:\mathbf{x}_j\in N_\varepsilon^{\mathcal{M}}(\mathbf{x}_i)\}$ be the set of ε neighborhood of $\mathbf{x}_i$ on $\mathcal{M}$. Then, with probability of at least $1-\mu$, for all $\mathbf{x}_i$ and $\mathbf{x}_j\in N_i$, we have $|d_{\mathcal{M}}(\mathbf{x}_i,\mathbf{x}_j)-d_G(\mathbf{x}_i,\mathbf{x}_j)|\leq\varepsilon'$. Therefore, $d_G(\mathbf{x}_i,\mathbf{x}_j)\leq d_{\mathcal{M}}(\mathbf{x}_i,\mathbf{x}_j)+\varepsilon'\leq\varepsilon+\varepsilon'$. Furthermore, $\|\mathbf{x}_i-\mathbf{x}_j\|_2\leq d_G(\mathbf{x}_i,\mathbf{x}_j)$. Hence, $\|\mathbf{x}_i-\mathbf{x}_j\|_2\leq\varepsilon'+\varepsilon$. That is there exists a sampling of $\mathcal{M}$ with rate $\alpha(\varepsilon',\mu)$ for which with probability of at least $1-\mu$, the ε neighborhood of each point $\mathbf{x}_i$ on $\mathcal{M}$ lie in the $\varepsilon'+\varepsilon$ neighborhood of that point in $\mathbb{R}^m$. □

One of the main consequences of Theorem 1 is that for a densely sampled manifold, in order to find the ε neighbors on the manifold (optimal neighborhood), we may search in a larger (but comparable to ε) neighborhood of the ambient space. This fact helps to bound the space of possible neighborhood graph structures. Therefore, it may be employed in an optimization scheme to make the graph construction problem well-posed.

Since ε neighborhood graphs are not suitable for the classification problem, we may generalize this theorem to consider the k-NN graphs. To do this, note that for an ε neighborhood graph $G_\varepsilon=(V_\varepsilon,E_\varepsilon)$, there exist k and k' such that G_ε is a super-graph and subgraph of k-NN and k'-NN graphs, respectively. To do this we may take $k=\min_{v_i\in V_\varepsilon}deg(v_i)$ and $k'=\max_{v_i\in V_\varepsilon}deg(v_i)$.

We find such a k for the ε neighborhood (on $\mathcal{M}$) graph and k' for the $\varepsilon+\varepsilon'$ neighborhood (on ambient space) graph. With probability of at least $1-\mu$, we have

$$E_k^{\mathcal{M}}\subseteq E_\varepsilon^{\mathcal{M}}\subseteq E_{\varepsilon+\varepsilon'}^{\mathbb{R}^m}\subseteq E_{k'}^{\mathbb{R}^m} \quad (27)$$

where $(V,E_k^{\mathcal{M}})$ and $(V,E_k^{\mathbb{R}^m})$ denotes the k-NN graphs on $\mathcal{M}$ and $\mathbb{R}^m$, respectively, and $(V,E_\varepsilon^{\mathcal{M}})$ and $(V,E_{\varepsilon+\varepsilon'}^{\mathbb{R}^m})$ are the ε and $\varepsilon+\varepsilon'$ neighborhood graphs on $\mathcal{M}$ and $\mathbb{R}^m$, respectively. Since, we may take

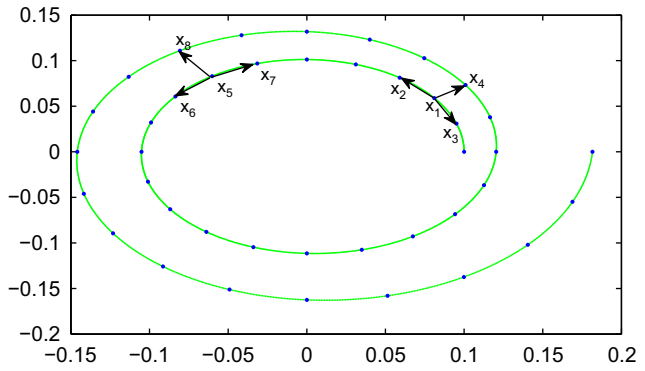

Fig. 1. The optimal graph is a subgraph of the 3-NN graph. Dark points are the sample points of the manifold. The three nearest neighbor points of $\mathbf{x}_1$ and $\mathbf{x}_5$ are indicated with arrows.

$\varepsilon+\varepsilon'$ small enough (by increasing the sampling rate), k' will be a small fraction of all the data point number $l+u$.

As an example, consider the manifold shown in 2D in Fig. 1. It can be easily seen that the optimal 2-NN graph is the subgraph of the 3-NN graph in the ambient space. It is obvious that the edges from $\mathbf{x}_1$ to $\mathbf{x}_4$ and $\mathbf{x}_5$ to $\mathbf{x}_8$ should be removed in the optimal neighborhood graph.

4.3. Link classification

Link classification is aimed to estimate the values of $p_{i,j}$. The input to the classifier is a nearest neighbor link $(\mathbf{x}_i,\mathbf{x}_j)$ and the output is 0 or 1 if the probability that $y_{t,i}\neq y_{t,j}$ is less or greater than 0.5, respectively. That is the classification is aimed to discriminate the intra- and inter-class nearest neighbor links. Training data generation and the appropriate classifier selection are the main issues that will be described next.

Consider the labeled data $X_l=\{\mathbf{x}_1,\ldots,\mathbf{x}_l\}$ with the corresponding labels $\mathbf{y}=\{y_1,\ldots,y_l\}$, and the labeled sets $D_l^j=\{\mathbf{x}_i\in X_l; y_i=j\}$ for $1\leq j\leq n_C$, where n_C is the number of classes. Then define intra- and inter-class nearest neighbors of $\mathbf{x}_i$ as $N_{k_0}^W(\mathbf{x}_i)=\{\mathbf{x}_j\in D_l^{y_i};$ $\mathbf{x}_j$ is within the k_0-NNs of $\mathbf{x}_i$ in $D_l^{y_i}\}$ and $N_{k_0}^B(\mathbf{x}_i)=\{\mathbf{x}_j\in X_l\backslash D_l^{y_i};\mathbf{x}_j$ is within the k_0-NNs of $\mathbf{x}_i$ in $X_l\backslash D_l^{y_i}\}$. It is clear that $\forall i: |N_{k_0}^B(\mathbf{x}_i)|=|N_{k_0}^W(\mathbf{x}_i)|=k_0$. Then the positive and negative training data sets will be $D_{link}^P=\{(\mathbf{x}_i,\mathbf{x}_j);\mathbf{x}_i\in X_l,\mathbf{x}_j\in N_{k_0}^B(\mathbf{x}_i)\}$ and $D_{link}^N=\{(\mathbf{x}_i,\mathbf{x}_j);\mathbf{x}_i\in X_l,$ $\mathbf{x}_j\in N_{k_0}^W(\mathbf{x}_i)\}$, respectively.

Let $T_{\mathcal{M}}(\mathbf{x}_i)$ be the tangent space of $\mathcal{M}$ at point $\mathbf{x}_i$. It is clear that because of the manifold assumption, the nearest neighbors of $\mathbf{x}_i$ in $\mathbb{R}^m$ that lie on $T_{\mathcal{M}}(\mathbf{x}_i)$ will have almost the same label y_i. That is for a large number of negative links $(\mathbf{x}_i,\mathbf{x}_j)$, $\mathbf{x}_j-\mathbf{x}_i$ resides on a local neighborhood of $\mathbf{x}_i$ on $T_{\mathcal{M}}(\mathbf{x}_i)$. In addition, for any member of D_{link}^P, $(\mathbf{x}_i,\mathbf{x}_j)$, although $\mathbf{x}_i$ and $\mathbf{x}_j$ are nearest neighbors in the ambient space, they will not be neighbors on the manifold. This comes from the facts that $y_i\neq y_j$ for these links and according to the manifold assumption two neighbor points on the manifold may have the same label with high probability. Therefore, $\mathbf{x}_j-\mathbf{x}_i$ lies outside a local neighborhood of $\mathbf{x}_i$ on $T_{\mathcal{M}}(\mathbf{x}_i)$. We may conclude that for each $\mathbf{x}_i$, $(\mathbf{x}_j-\mathbf{x}_i)$'s lie almost in two different parts of the ambient space for the positive and negative links. This fact is illustrated for a simple example in Fig. 2.

Hence if we consider the link data points of the form $(\mathbf{x}_i,\mathbf{x}_j-\mathbf{x}_i)$, the link data points can be classified locally on the manifold using an SVM classifier with a Gaussian kernel. Since the tangent space of a regular manifold changes smoothly on the manifold, for nearby $\mathbf{x}_i$'s on the manifold, the tangent space and its complement remain almost the same. Therefore, the positive and negative links for nearby $\mathbf{x}_i$'s are almost separable by an SVM classifier with radial basis kernel. This fact is shown for a simple example in Fig. 3.

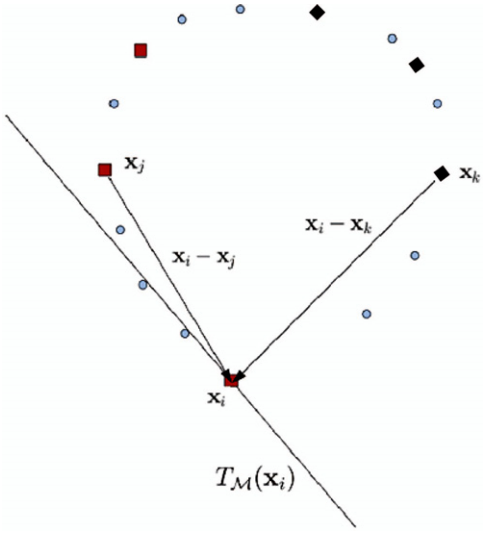

Fig. 2. $\mathbf{x}_j-\mathbf{x}_i$ lies on two complement subspaces for inter- and intra-class links. Squares and diamonds are the labeled data. Circles are the unlabeled data that are sampled from the manifold.

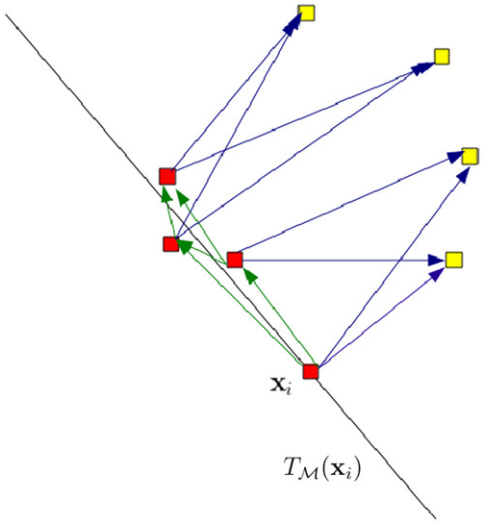

Fig. 3. Nearby inter- and intra-class links are locally separable on the manifold, since they lie approximately on two complement subspaces.

Therefore, we split the links data of two classes into several local sets on the manifold. One simple way to do this is to split the links according to the estimated label of one of its end points using Tikhonov regularization. It is noticeable that the data points classification based on the Tikhonov regularization over the k-NN graph almost implies the notation of the locality on the manifold, because according to the manifold assumption, data points on the manifold with the same labels occur almost close to each other on the manifold. Consider the link data be split as $D_{link}(m)=\{(\mathbf{x}_i,\mathbf{x}_j-\mathbf{x}_i):$ $(\mathbf{x}_i,\mathbf{x}_j)\in D_{link}^N\cup D_{link}^P; f_i=m\}$ for $1\leq m\leq n_C$. We train a new SVM classifier for each $D_{link}(m)$.

4.4. Graph construction

Prior to use of the link classifier in the graph construction phase, the labels of the data in the transduction set X_u should be estimated. Simple k-NN graph and Tikhonov Regularization using the Laplacian matrix in a one-against-all classification scheme is used to obtain an estimation $\hat{y}_i=f(\mathbf{x}_i)$ of the data point labels. For each point $\mathbf{x}_i$, let $N_d^{\mathbb{R}^m}(\mathbf{x}_i)=\{\mathbf{z}_1^{(i)},\ldots\mathbf{z}_d^{(i)}\}$. Pairs of $(\mathbf{x}_i,\mathbf{z}_j^{(i)})$ are classified using the SVM

related to $\hat{y}_i$ (in the ascending order of j), where f is the estimation of labels using a graph based method. Using the greedy solution to the optimization of Eq. (24), the k first pairs that are classified as negative are established. If there are less than k negative pairs in $N_{k'}(\mathbf{x}_i)$, pairs of $\mathbf{x}_i$ with its $(k'+u)$-NNs are established without any classification, where $u \geq 1$, until k established links for $\mathbf{x}_i$ is achieved. This step is necessary, since we require each node to have at least k neighbors on the graph. Otherwise, the graph may be disconnected, which leads to poor generalization performance [4]. Algorithm 1 describes the proposed graph construction method.

Algorithm 1. The proposed supervised graph construction algorithm.

```
L ← Laplacian matrix of the k-NN graph of the whole data set
g ← one-against-all classification of all data using Tikhonov Regularization
for all data point x_i do
  {z_1, ..., z_{l+u-1}} ← N_{l+u-1}(x_i)
  selected ← 0
  j ← 0
  while selected < k and j ≤ k' do
    j ← j+1
    class ← class of (z_i, z_j) classified by the g_i th SVM
    if class = 0 then
      connect x_i to z_j
      selected ← selected+1
    end if
  end while
  if selected < k then
    repeat
      j ← j+1
      connect x_i to z_j
      selected ← selected+1
    until selected = k
  end if
end for
```

5. Experimental results

A number of popular data sets, namely MNIST [19] and USPS [20] (digit recognition), ISOLET from the UCI repository (spoken letter recognition), ForestCover from the UCI repository (Forest-Cover type classification), Corel1 (an image categorization data set from Corel data set) [21] and HyperSpectral (a 220 band remote sensing data set downloaded from Purdue University remote sensing project page (June 12, 1992 AVIRIS image North-South flightline)) are selected. The characteristics of the data sets are summarized in Table 1. 1000 data points of each data set were chosen as the training and test sets. Selection is done in a way that almost the same number of data points corresponding to each label is chosen. To speed up the process of graph construction and to remove the noise in the data, the dimensionality of the data was reduced by applying the PCA algorithm. The regularization trade off parameter γ is set to the fixed value of 0.01, and the number of nearest neighbors k is set to 7, which can also be tuned by the standard cross validation method. The number of training links per training data, k_0 ($2 \leq k_0 \leq 10$), can be chosen by cross validation and k' is set to a large value (for example $k' = 40$). The method of one-against-all is used to solve the multi-class problem. That is, for each data $\mathbf{x}_i$ and class j, $f_{i,j}$ will be the estimated label that $\mathbf{x}_i$ belongs to class j. The assigned class of $\mathbf{x}_i$ will be $\text{argmax}_j f_{i,j}$. The number of learned eigenvalues in SKL is set to 500. All problem are multi-class, except HyperSpectral data set which we considered the problem of separation of the first class against the other classes.

Table 1
Data sets characteristics.

Data set	Number of data points	Number of features	Type of features
MNIST	60,000	784	Integer in [0, 255]
USPS	11,000	256	Integer in [0, 255]
ISOLET	7797	617	Real in [−1, 1]
ForestCover	119,104	54	Integer in [−146, 7173]
COREL1	1000	144	Integer in [0, 7]
HyperSpectral	21,025	220	Integer in [955, 9604]

5.1. Link classification performance

We compared the performance of the link classification algorithm in three cases. These cases include:

- One SVM: Use a single SVM to classify all the links.
- Multiple SVMs: Split the link data set according to the estimated label of one of their end points.
- All Links: The same as multiple SVMs, except that we used all links instead of links with nearest neighbors end points. In fact this method is similar to the idea of [16], which performs the classification on all points.

Two hundred samples are selected as the labeled set. k_0 and k' are set to 5 and 40, respectively. The dimensionality of the data is reduced to 30 using PCA. The accuracies of the link classification in these cases are shown in Table 2.

It is noticeable that the proposed Multiple SVMs method almost outperforms the other methods. In addition, the higher accuracies of the Multiple SVMs method compared with the All Links method in most cases show that the estimation of $p_{i,j}$ is simpler when we use the proposed prior knowledge on the neighborhood graph. Therefore, we expect that the graph construction using the proposed method outperforms the simple k-NN method.

5.2. Graph construction performance

To evaluate the proposed method quantitatively, a number of classification experiments with different labeled sets is designed. Each experiment is repeated 20 times to find the confidence interval of the error. The proposed method is compared with the SKL [8], GES [4] and ML [7] methods. In order to further investigate the quality of our graph construction, SKL and ML are applied on top of the graph constructed by the proposed method. In all RBF SVMs, the data is whitened in the pre-processing step and the bandwidth is taken proportional to squared root of the number of features. It is noticeable that data whitening scales each feature almost in $[-1,1]$ along the covariance principal directions. Average classification error rate of different methods when the number of PCA dimensions is 30 and the labeled set size is 200 are compared in Table 3.

Table 2
Accuracies of link classification for different link classification algorithms.

Data set	One SVM (%)	Multiple SVMs (%)	All links (similar to [16]) (%)
MNIST	74.2 ± 0.5	$\mathbf{79.0 \pm 0.5}$	76.0 ± 0.3
COREL1	69.1 ± 0.3	$\mathbf{74.2 \pm 0.3}$	72.2 ± 0.2
ForestCover	$\mathbf{74.8 \pm 0.5}$	74.2 ± 0.4	72.9 ± 0.5
ISOLET	74.7 ± 0.3	$\mathbf{77.8 \pm 0.3}$	73.6 ± 0.2
USPS	76.4 ± 0.4	$\mathbf{81.6 \pm 0.4}$	75.7 ± 0.4
HyperSpectral	56.7 ± 0.5	61.7 ± 0.7	$\mathbf{66.4 \pm 0.4}$

Table 3
Classification error rate of different methods.

Method	MNIST (%)	COREL1 (%)	ForestCover (%)	ISOLET (%)	USPS (%)	HyperSpectral (%)
Linear SVM	24.2 ± 0.4	20.0 ± 0.3	22.2 ± 0.4	25.3 ± 0.5	21.7 ± 0.4	29.1 ± 0.4
RBF SVM	17.4 ± 0.4	20.5 ± 0.3	18.0 ± 0.4	22.7 ± 0.4	15.5 ± 0.3	28.3 ± 0.4
k-NN	14.0 ± 0.3	22.0 ± 0.3	16.6 ± 0.4	25.9 ± 0.4	13.2 ± 0.3	26.7 ± 0.5
GES	14.0 ± 0.4	21.7 ± 0.3	17.2 ± 0.3	26.7 ± 0.5	13.5 ± 0.4	26.8 ± 0.4
SKL	14.6 ± 0.4	24.3 ± 0.4	16.5 ± 0.4	28.3 ± 0.5	13.5 ± 0.3	29.0 ± 0.5
ML	12.8 ± 0.4	21.1 ± 0.3	16.5 ± 0.4	25.5 ± 0.5	11.8 ± 0.3	26.9 ± 0.6
Sk-NN	13.2 ± 0.3	20.1 ± 0.3	15.5 ± 0.3	22.0 ± 0.4	12.3 ± 0.3	$\mathbf{25.7 \pm 0.5}$
Sk-NN+ML	$\mathbf{12.1 \pm 0.3}$	20.0 ± 0.3	13.9 ± 0.4	23.0 ± 0.5	$\mathbf{11.2 \pm 0.3}$	26.1 ± 0.5
Sk-NN+SKL	12.5 ± 0.3	$\mathbf{19.6 \pm 0.3}$	$\mathbf{13.2 \pm 0.3}$	$\mathbf{20.8 \pm 0.3}$	11.7 ± 0.3	27.4 ± 0.5

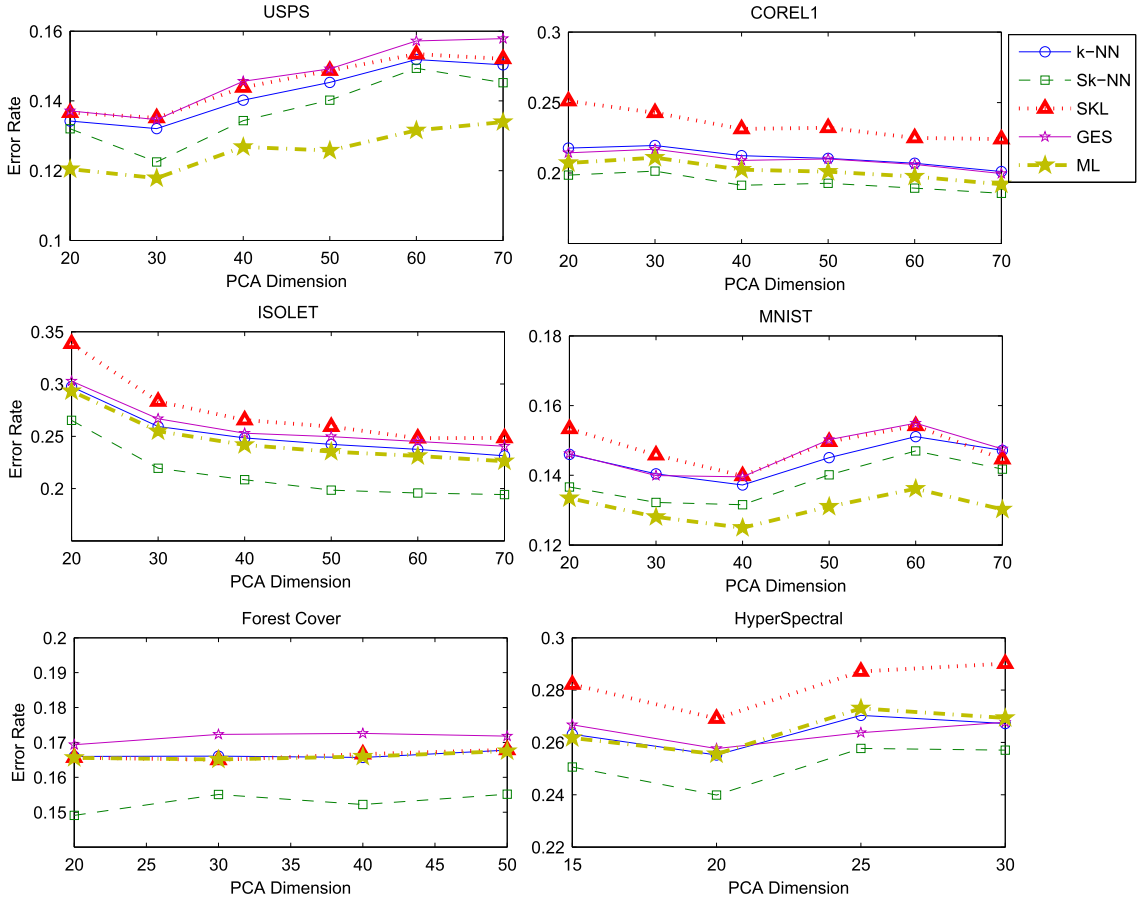

Fig. 4. Comparison of the average error rates of different methods in USPS, COREL1, ISOLET, MNIST, ForestCover and HyperSpectral data sets.

Error rates of the k-NN, SKL, ML, GES and Sk-NN methods are plotted against the number of PCA dimensions in Fig. 4. Sk-NN represents the proposed supervised k-NN method.

It is noted that the proposed method performance is competitive to the ML method. That is, the Sk-NN method outperforms ML on COREL1, ISOLET, ForestCover and HyperSpectral data sets. On the other hand, ML performs a little better on USPS and MNIST data sets. In contrast to Sk-NN, the ML method fails to improve the performance of classification on some data sets such as ISOLET, ForestCover and HyperSpectral.

The combinations of Sk-NN and other methods are compared in Fig. 5. When other methods are applied on top of Sk-NN, the performance is improved. Although SKL has almost the highest error rate, the Sk-NN+SKL method has the minimum error rate in data sets such as ISOLET, MNIST and ForestCover. This means that SKL needs a suitable neighborhood graph as a base to improve the classification accuracy. Therefore, the proposed method may also be used as a suitable basis for the other methods.

To evaluate robustness of the proposed method, the difference between error rates of the Sk-NN and k-NN methods are plotted against k' in Fig. 6 using 200 labeled samples. As can be seen, the differences do not exhibit large deviation from the mean values in all data sets, which shows that the proposed method is almost robust to this parameter.

6. Conclusion

In this paper, a novel supervised graph construction scheme is proposed. We have shown that under the using of large enough manifold sampling rate, the optimal neighborhood graph is subgraph of a k'-NN graph with high probability. Therefore, we used this assumption as the prior knowledge. In contrast with other methods, this assumption also helps to take a suitable hypothesis space for the neighborhood graph. In addition, the proposed method makes it possible to learn the structure of the

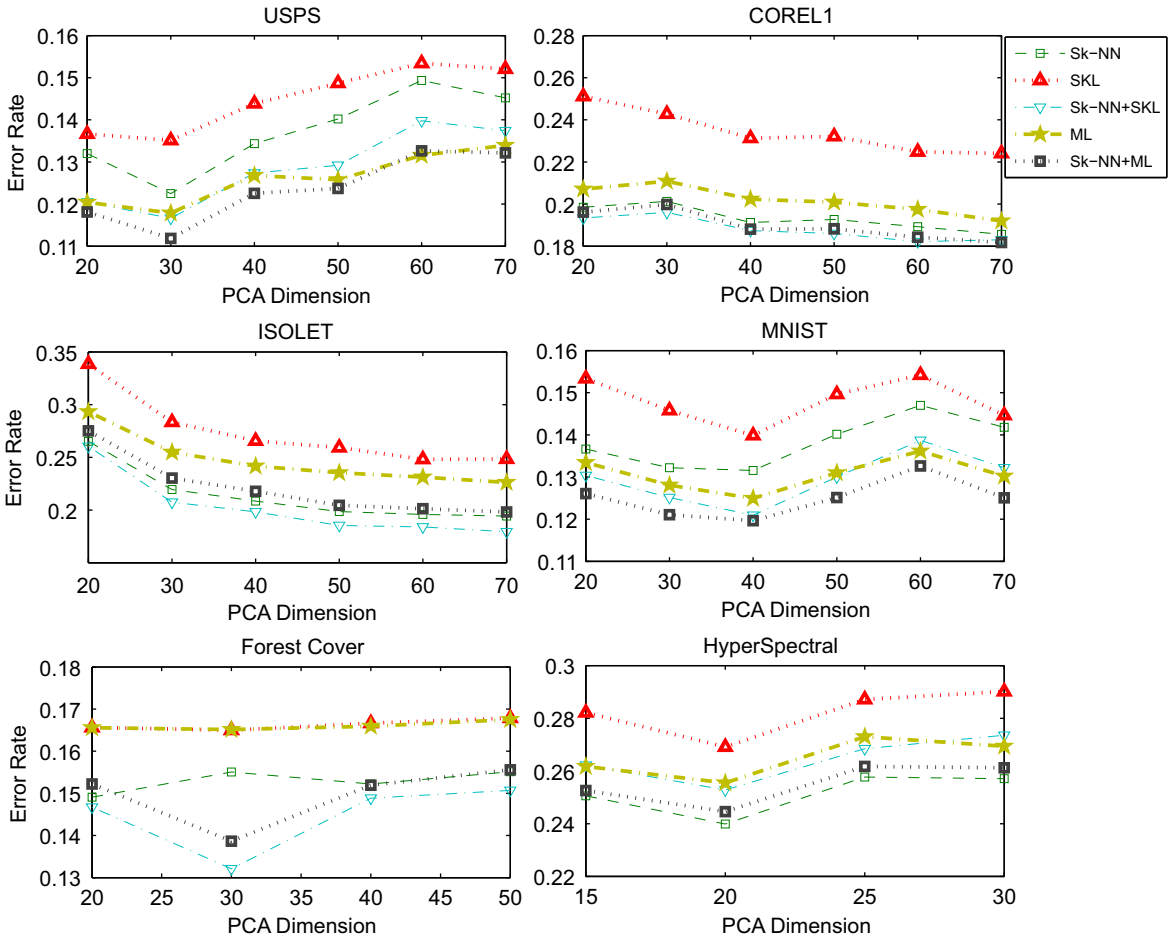

Fig. 5. Comparison of the average error rates of combined methods in USPS, COREL1, ISOLET, MNIST, ForestCover and HyperSpectral data sets.

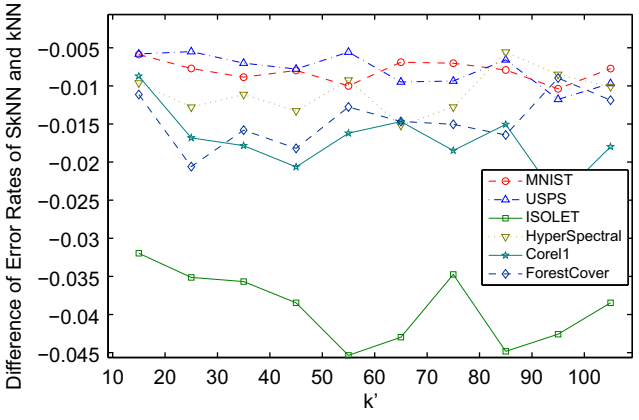

Fig. 6. Difference between error rates of the S*k*-NN and *k*-NN methods as a function of *k*′.

graph by maximizing the expectation of the smoothness. To avoid the local minima problem, a method to directly calculate the expectation is proposed. Experimental results show that the proposed method is superior or competitive with other state-of-the-art supervised graph construction methods.

References

[1] M. Belkin, Problems of Learning on Manifolds, Ph.D. Thesis, Department of Mathematics, University of Chicago, 2003.

[2] T. Jebara, J. Wang, S.F. Chang, Graph construction and *b*-matching for semi-supervised learning, in: International Conference on Machine Learning, 2009.

[3] B. Cheng, J. Yang, S. Yan, Y. Fu, T.S. Huang, Learning with l1-graph for image analysis, IEEE Transactions on Image Processing 19 (2010) 858–866.

[4] H. Shin, N.J. Hill, G. Rätsch, Graph based semi-supervised learning with sharper edges, in: European Conference on Machine Learning, 2006.

[5] M. Hein, M. Maier, Manifold denoising, in: Advances in Neural Information Processing Systems, 2006.

[6] J. Wang, Z. Zhang, H. Zha, Adaptive manifold learning, in: Advances in Neural Information Processing Systems, 2005.

[7] A. Kapoor, Y. Qi, H. Ahn, R. Picard, Hyperparameter and kernel learning for graph based semi-supervised classification, in: Advances in Neural Information Processing Systems, 2006.

[8] X. Zhu, J. Kandola, Z. Ghahramani, J. Lafferty, Nonparametric transforms of graph kernels for semi-supervised learning, in: Advances in Neural Information Processing Systems, 2005.

[9] X. Zhang, W.S. Lee, Hyperparameter learning for graph based semi-supervised learning algorithms, in: Advances in Neural Information Processing Systems, 2006.

[10] P.S. Dhillon, P.P. Talukdar, K. Crammer, Inference driven metric learning for graph construction, in: 4th North East Student Colloquium on Artificial Intelligence, 2010.

[11] C. Schaffer, A conservation law for generalization performance, in: International Conference on Machine Learning, 1994.

[12] D. Wolpert, The lack of a priori distinctions between learning algorithms, Neural Computation 8 (7) (1996) 1341–1390.

[13] X. Zhu, Semi-supervised Learning Literature Survey, Technical Report 1530, Department of Computer Sciences, University of Wisconsin, Madison, 2005.

[14] C.E. Rasmussen, C.K.I. Williams, Gaussian Processes for Machine Learning, The MIT Press, 2006.

[15] G. Casella, R.L. Berger, Statistical Inference, Duxbury Press, 2001.

[16] A. Alexandrescu, K. Kirchhoff, Data-driven graph construction for semi-supervised graph-based learning in NLP, in: HLP NAACL, 2007.

[17] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines and Other Kernel-based Learning Methods, Cambridge University Press, 2000.

[18] M. Bernstein, V. de Silva, J. Langford, J. Tenenbaum, Graph Approximations to Geodesics on Embedded Manifolds, Technical Report, 2000.
[19] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, in: Proceedings of the IEEE, 1998, pp. 2278–2324.
[20] J.J. Hull, A database for handwritten text recognition research, IEEE Transactions on Pattern Analysis and Machine Intelligence (1994) 550–554.
[21] Y. Chen, Image categorization by learning and reasoning with regions, Journal of Machine Learning Research 5 (2004) 913–939.

Mohammad Hossein Rohban received his M.Sc. in Artificial Intelligence from Sharif University of Technology, Tehran, Iran, in 2008 and a B.Sc. in Software Engineering from the same university. He is currently a Ph.D. student in the Department of Computer Engineering, Sharif University of Technology. His current research interests include semi-supervised learning, geometry-based pattern recognition, and statistical learning theory.

Hamid R. Rabiee (SM, IEEE) received his B.S. and M.S. degrees (with great distinction) in Electrical Engineering from CSULB, USA, his EEE in Electrical and Computer Engineering from USC, USA, and his Ph.D. in Electrical and Computer Engineering from Purdue University, West Lafayette, USA, in 1996. From 1993 to 1996 he was a Member of Technical Staff at AT&T Bell Laboratories. From 1996 to 1999 he worked as a Senior Software Engineer at Intel Corporation. He was also with PSU, OGI and OSU Universities as an adjunct professor of Electrical and Computer Engineering from 1996 to 2000. Since September 2000, he has joined Sharif University of Technology (SUT), Tehran, Iran. He is the founder of Sharif University Advanced Information and Communication Technology Research Center (AICTC), Sharif University Advanced Technologies Incubator (SATI), Sharif Digital Media Laboratory (DML) and Mobile Value Added Services (MVAS) laboratories. He is currently an Associate Professor of the Computer Engineering at Sharif University of Technology, an Adjunct Professor of Computer Science at UNB, Canada, and the Director of AICTC, DML and MVAS. He has been the initiator and director of national and international level projects in the context of UNDP International Open Source Network (IOSN) and Iran National ICT Development Plan. Dr. Rabiee has received numerous awards and honors for his Industrial, scientific and academic contributions. He has acted as chairman in a number of national and international conferences, and holds three patents. He is also a Senior Member of IEEE.