Neural Comput & Applic (2011) 20:1043–1053
DOI 10.1007/s00521-010-0448-7

ORIGINAL ARTICLE

Glocalization pursuit support vector machine

Hui Xue · Songcan Chen

Received: 18 October 2009 / Accepted: 6 September 2010 / Published online: 23 September 2010

Abstract Graph-based methods have aroused wide interest in pattern recognition and machine learning, which capture the structural information in data into classifier design through defining a graph over the data and assuming label smoothness over the graph. Laplacian Support Vector Machine (LapSVM) is a representative of these methods and an extension of the traditional SVM by optimizing a new objective additionally appended Laplacian regularizer. The regularizer utilizes the local linear patches to approximate the data manifold structure and assumes the same label of the data on each patch. Though LapSVM has shown more effective classification performance than SVM experimentally, it in fact concerns more the locality than the globality of data manifold due to the Laplacian regularizer itself. As a result, LapSVM is relatively sensitive to the local change of the data and cannot characterize the manifold quite faithfully. In this paper, we design an alternative regularizer, termed as Glocalization Pursuit Regularizer. The new regularizer introduces a natural global structure measure to grasp the global and local manifold information as simultaneously as possible, which can be proved to make the representation of the manifold more compact than the Laplacian regularizer. We further introduce the new regularizer into SVM to develop an alternative graph-based SVM, called as Glocalization Pursuit Support Vector Machine (GPSVM). GPSVM not only inherits the advantages of both SVM and LapSVM but also uses the structural information more reasonably to guide the classifier design. The experiments both on the toy and real-world datasets demonstrate the better classification performance of our proposed GPSVM compared with SVM and LapSVM.

Keywords Support vector machine · Graph-based method · Structural information · Pattern recognition

1 Introduction

Graph-based methods are currently hot issues in pattern recognition and machine learning [1–4], which aim to further grasp and fuse the structural information in data into the recognition process in order to utilize the latent data knowledge more fully. Generally, these methods first construct a graph over the dataset where the nodes are the given data and the weighted edges reflect the similarity between the data [1]. Then they design the classifier by estimating a function $\boldsymbol{f}$ over the graph, where $\boldsymbol{f}$ should be close to the given labels on the labeled nodes and smooth on the whole graph [1]. This process can be boiled down to a regularization framework in many graph-based methods [1, 5]:

$$\min_{\boldsymbol{f}\in\boldsymbol{R}^n}\left\{\sum_{i=1}^{n} V(y_i,f(\boldsymbol{x}_i)) + \lambda R_{\mathrm{reg}}(\boldsymbol{f})\right\} \tag{1}$$

where the loss function $V(y_i,f(\boldsymbol{x}_i))$ measures the discrepancy between the true labels and the estimated labels produced by $\boldsymbol{f}$ for the data [6]. And the regularizer $R_{\mathrm{reg}}(\boldsymbol{f})$

H. Xue
School of Computer Science and Engineering,
Southeast University, Nanjing, People's Republic of China
e-mail: hxue@seu.edu.cn

H. Xue · S. Chen (✉)
College of Information Science and Technology,
Nanjing University of Aeronautics and Astronautics,
Nanjing, People's Republic of China
e-mail: s.chen@nuaa.edu.cn

embeds the desirable properties of $\boldsymbol{f}$ over the graph, such as smooth, discriminative, consistent.

Usually, the loss function $V(y_i,f(\boldsymbol{x}_i))$ can be selected as the squares function, absolute value function, hinge function, exponential function, logarithm function, ε-insensitive function, and so on, which is relied on the different but favorable statistical properties of these functions themselves related to the Bayes consistency [7–10], and the requirement of real problems such as regression and classification.

The regularizer $R_{\text{reg}}(\boldsymbol{f})$ is a key point in the graph-based methods, which directly specifies the characteristics of $\boldsymbol{f}$ [5]. Tikhonov regularizer [6] emphasizes the global smoothness of $\boldsymbol{f}$, that is, the similar inputs should correspond to the similar outputs produced by $\boldsymbol{f}$ in the whole data space:

$$R_{\text{reg}}(\boldsymbol{f}) = \|D\boldsymbol{f}\|^2 \tag{2}$$

where D is a linear differential operator applied to $\boldsymbol{f}$, which is also referred to as a stabilizer because the smoothness prior involved in it makes $\boldsymbol{f}$ stable [6, 11].

Different from Tikhonov regularizer, Laplacian regularizer [12, 13] concerns the local smoothness of $\boldsymbol{f}$ over the graph. It sets the weights of the edges as:

$$w_{ij} = \begin{cases} \exp(-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2/\sigma^2) & \text{if } \boldsymbol{x}_i \in ne(\boldsymbol{x}_j) \quad \text{or} \quad \boldsymbol{x}_j \in ne(\boldsymbol{x}_i) \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where $ne(\boldsymbol{x}_i)$ denotes the nearest neighborhood of $\boldsymbol{x}_i$, in which the nodes are connected in the graph. The corresponding Laplacian matrix can be computed as [5]:

$$\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W} \tag{4}$$

where $\boldsymbol{W} = [w_{ij}] \in \boldsymbol{R}^{n\times n}$ is the adjacency matrix of the graph. $\boldsymbol{D} \in \boldsymbol{R}^{n\times n}$ is a diagonal matrix and its entries are $\boldsymbol{D}_{ii} = \sum_{j=1}^{n} w_{ij}$. Laplacian regularizer utilizes the local linear patches to approximate the data manifold structure and requires $\boldsymbol{f}$ smooth over the patches:

$$R_{\text{reg}}(\boldsymbol{f}) = \|\boldsymbol{f}\|_I^2 = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}(f_i - f_j)^2 = \boldsymbol{f}^T\boldsymbol{L}\boldsymbol{f} \tag{5}$$

Normalized Laplacian regularizer [14] is also widely used in the graph-based methods. Its Laplacian matrix is normalized symmetrically as:

$$\boldsymbol{L}_n = \boldsymbol{I} - \boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{W}\boldsymbol{D}^{-\frac{1}{2}} \tag{6}$$

where $\boldsymbol{I}$ is the identity matrix, and $\boldsymbol{W}$ and $\boldsymbol{D}$ are the same as in Laplacian regularizer. So the regularizer can be described as:

$$R_{\text{reg}}(\boldsymbol{f}) = \|\boldsymbol{f}\|_{I_n}^2 = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}\left(\frac{f_i}{\sqrt{\boldsymbol{D}_{ii}}} - \frac{f_j}{\sqrt{\boldsymbol{D}_{jj}}}\right)^2 = \boldsymbol{f}^T\boldsymbol{L}_n\boldsymbol{f} \tag{7}$$

Local learning regularizer [5] further builds a linear model in each neighborhood $ne(\boldsymbol{x}_i)$ and then trains the output function $o_i(\cdot)$ of the local model by some supervised learning algorithms. The regularizer requires that f_i should be similar to the output of the model $o_i(\boldsymbol{x}_i)$ in order to well estimate the value of f_i based on the neighborhood of $\boldsymbol{x}_i$ [5]:

$$R_{\text{reg}}(\boldsymbol{f}) = \|\boldsymbol{f} - \boldsymbol{o}\|^2 = \sum_{i=1}^{n}(f_i - o_i(\boldsymbol{x}_i))^2 \tag{8}$$

Discriminative regularizer [4] concentrates on the discriminative property of $\boldsymbol{f}$, which is usually integrated with the squares loss function. The regularizer constructs two graphs to characterize the intra-class compactness and inter-class separability respectively and thus aims to further maximize the margins between the data of the different classes in each local neighborhood:

$$R_{\text{disreg}}(\boldsymbol{f},\eta) = \eta\tilde{S}_w - (1-\eta)\tilde{S}_b \tag{9}$$

where $\tilde{S}_w = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(f_i - f_j)^2 w_{w,ij}$ and $\tilde{S}_b = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(f_i - f_j)^2 w_{b,ij}$ are the metrics defined over the intra-class graph G_w and the inter-class graph G_b, which measure the intra-class compactness and inter-class separability of the outputs respectively. η is the regularizer parameter, $0 \le \eta \le 1$.

In the past decade, different combinations of the loss functions and regularizers or equivalent, prior penalty, have derived a large family of graph-based methods [1, 15–18]. Classical Support Vector Machine (SVM) [19, 20] combines the hinge function with Tikhonov regularizer (or smoothness penalty) in a Reproducing Kernel Hilbert Space (RKHS) [6] to emphasize the discriminability and global smoothness of the solution function $\boldsymbol{f}$. Zhu et al. [17, 21] utilized a squares function with infinity weight and the Laplacian regularizer (or penalty over data manifold) to develop the Gaussian random fields and harmonic function methods which is a continuous relaxation to the difficulty discrete Markov random fields [22] (or Boltzmann machines) [1]. Zhou et al. [14] used the squares function and normalized Laplacian regularizer to improve the consistency of $\boldsymbol{f}$ in the semi-supervised classifier designs in order to make $\boldsymbol{f}$ smooth with respect to the intrinsic structure collectively revealed by labeled and unlabeled samples. Wu and Schölkopf [5] proposed a local learning regularization method for the transductive classification problems by integrating the squares function with local learning regularizer, which leads to the solution with the property that the label of each sample can be well predicted based on its neighbors and their labels. Xue et al. [5] associated the squares function with discriminative regularizer and presented a discriminatively regularized least-squares classification method in supervised learning that focuses on not only the discriminative information but also

on the local geometry of the data and intends to maximize the margins between the data of different classes in each local area.

Different from the above approaches which combine the loss functions with a single regularizer, Laplacian Support Vector Machine (LapSVM) [23] further selects the hinge function as the loss measure and combines the Tikhonov regularizer and Laplacian regularizer as the regularizers. That is, LapSVM is an extension of the traditional SVM by optimizing the objective additionally appended Laplacian regularizer to the SVM objective, which thus can fuse the properties of the two type methods. On the one hand, LapSVM embeds the local structural information of the data manifold into SVM by the Laplacian regularizer with the aim to utilize the geometric distribution to guide the more effective classification. On the other hand, LapSVM still maintains the similar characteristics to SVM, which can maximize the margins between the classes. Moreover, its optimization problem can also be formulated as Quadratic Programming (QP) by some transformations and solved by the same optimization techniques as SVM to obtain the final sparse solutions. Though LapSVM has been showed to be superior to SVM in classification performance experimentally, it is relatively sensitive to the local change of the data manifold due to that it concerns more the locality than the globality of manifold structure. The Laplacian regularizer emphasizes the approximation of local linear patches to the manifold and assumes that all the data on each patch share the same labels. However, its relatively less concern on the global structural information makes it unable to characterize the manifold faithfully.

In this paper, we present an alternative regularizer, termed as Glocalization Pursuit Regularizer, which respects both the globality and the locality of data manifold so that the shortcomings of the Laplacian regularizer can be avoided to some extent. The new regularizer can be proved to characterize the manifold more compactly than the Laplacian regularizer. We further introduce the new regularizer into SVM and present an alternative graph-based SVM, called as Glocalization Pursuit Support Vector Machine (GPSVM). GPSVM not only possesses the merits of both SVM and LapSVM but also captures the local and global structural information more reasonably. Experiments are conducted to demonstrate the superiority of our proposed GPSVM algorithm compared well with SVM and LapSVM.

The rest of the paper is organized as follows. Section 2 introduces the related works. Glocalization Pursuit Support Vector Machine is presented in Sect. 3. Section 4 gives the experimental results. Some conclusions are drawn in Sect. 5.

2 Related work

2.1 Support vector machine (SVM)

Here, we outline SVM to binary classification problems. Given a training set $\{\boldsymbol{x}_i, y_i\}_{i=1}^n \in \boldsymbol{R}^m \times \{\pm 1\}$, the objective of SVM is to learn a classifier $\boldsymbol{f}$ that can maximize the margin between classes [23]:

$$\min_{f \in H_K} \frac{1}{n} \sum_{i=1}^{n} (1 - y_i f(\boldsymbol{x}_i))_+ + \gamma \|\boldsymbol{f}\|_K^2 \tag{10}$$

where $1 - yf(\boldsymbol{x})_+ = \max(0, 1 - yf(\boldsymbol{x}))$ is the hinge loss function. $\|\boldsymbol{f}\|_K^2$ is the standard Tikhonov regularizer in an appropriately chosen RKHS that imposes smoothness conditions on possible solutions. γ is the corresponding regularizer parameter.

By the Representer Theorem [20], the solution is given by:

$$f^*(\boldsymbol{x}) = \sum_{i=1}^{n} \alpha_i^* K(\boldsymbol{x}, \boldsymbol{x}_i) \tag{11}$$

where $K(\boldsymbol{x}, \cdot)$ is the kernel function defined in the RKHS H_K.

Following SVM expositions, the above optimization problem can be equivalently written as [23]:

$$\begin{aligned} \min_{f \in \boldsymbol{H}_K, \xi_i \in \boldsymbol{R}} \quad & \frac{1}{n} \sum_{i=1}^{n} \xi_i + \gamma \boldsymbol{\alpha}^T \boldsymbol{K} \alpha \\ \text{s.t.} \quad & y_i f(\boldsymbol{x}_i) \geq 1 - \xi_i, \quad i = 1, \ldots, n \\ & \xi_i \geq 0, \quad i = 1, \ldots, n \end{aligned} \tag{12}$$

where ξ_i is the penalty for violating the constraints. $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_n]^T$, and $\boldsymbol{K}$ is the corresponding Gram matrix.

Using the Lagrange multipliers techniques, the dual problem of (12) can be represented as [23]:

$$\begin{aligned} \boldsymbol{\beta}^* = \max_{\boldsymbol{\beta} \in \boldsymbol{R}^n} \quad & \sum_{i=1}^{n} \beta_i - \frac{1}{2} \boldsymbol{\beta}^T \boldsymbol{Q} \boldsymbol{\beta} \\ \text{s.t.} \quad & \sum_{i=1}^{n} y_i \beta_i = 0 \\ & 0 \leq \beta_i \leq \frac{1}{n}, \quad i = 1, \ldots, n \end{aligned} \tag{13}$$

where $\boldsymbol{Q} = \boldsymbol{Y}\left(\frac{\boldsymbol{K}}{2\gamma}\right)\boldsymbol{Y}$ and $\boldsymbol{Y} = \mathrm{diag}[y_1, y_2, \ldots, y_n]$. This is a QP problem that can be solved by Sequential Minimal Optimization (SMO) algorithm and so on [20]. The final solution $\boldsymbol{\alpha}^*$ is

$$\boldsymbol{\alpha}^* = \boldsymbol{Y}\boldsymbol{\beta}^*/2\gamma. \tag{14}$$

2.2 Laplacian support vector machine (LapSVM)

For further controlling the complexity as measured by the geometry of the data distribution, LapSVM adds an additional Laplacian regularizer into the SVM objective [23]:

$$\begin{aligned}\min_{f\in \boldsymbol{H}_K,\xi_i\in \boldsymbol{R}} \ & \frac{1}{n}\sum_{i=1}^{n}\xi_i+\gamma_A\|\boldsymbol{f}\|_K^2+\gamma_I\|\boldsymbol{f}\|_I^2 \\ \text{s.t.} \ & y_i f(\boldsymbol{x}_i)\geq 1-\xi_i, \quad i=1,\ldots,n \\ & \xi_i\geq 0, \quad i=1,\ldots,n \end{aligned} \tag{15}$$

The Laplacian regularizer, which is derived by a manifold dimensional reduction method—Laplacian eigenmaps [12], can be constructed as follows:

1. Constructing the adjacency graph: find the set $ne(\boldsymbol{x}_i)$ of the k nearest neighbors of each data point $\boldsymbol{x_i}$ in the cth class ($c = 1, 2$) and put the edges among the $ne(\boldsymbol{x}_i)$.
2. Computing the edge weight:

$$w_{ij}^c=\begin{cases}\exp(-\|\boldsymbol{x}_i-\boldsymbol{x}_j\|^2/\sigma^2) & \text{if } \boldsymbol{x}_i\in ne(\boldsymbol{x}_j) \ \text{ or } \ \boldsymbol{x}_j\in ne(\boldsymbol{x}_i)\\ 0 & \text{otherwise}\end{cases} \tag{16}$$

3. Constructing Laplacian matrix:

$$\boldsymbol{L}_c=\boldsymbol{D}_c-\boldsymbol{W}_c, \quad c=1,2 \tag{17}$$

where $\boldsymbol{D}_c$ is a diagonal matrix, $\boldsymbol{D}_c^{ii}=\sum_{j=1}^{n} w_{ij}^c$. $\boldsymbol{W}_c=\left[w_{ij}^c\right]\in \boldsymbol{R}^{n\times n}$

4. Constructing the Laplacian regularizer $\|\boldsymbol{f}\|_I^2$: let $\boldsymbol{f}=[\boldsymbol{f}_1^T,\boldsymbol{f}_2^T]^T$, where $\boldsymbol{f}_c$ denotes the classification vector in the cth class ($c = 1, 2$). Then,

$$\begin{aligned}\|\boldsymbol{f}\|_I^2 &= \sum_{c=1}^{2}\sum_{i,j=1}^{n_c} w_{ij}^c\left(f_c(\boldsymbol{x}_i)-f_c(\boldsymbol{x}_j)\right)^2=\sum_{c=1}^{2}\boldsymbol{f}_c^T\boldsymbol{L}_c\boldsymbol{f}_c \\ &=\boldsymbol{f}^T\boldsymbol{L}\boldsymbol{f}\end{aligned} \tag{18}$$

where $\boldsymbol{L}=\begin{bmatrix}\boldsymbol{L}_1 & \\ & \boldsymbol{L}_2\end{bmatrix}$.

Consequently, the optimization problem (15) can be rewritten as:

$$\begin{aligned}\min_{f\in \boldsymbol{H}_K,\xi_i\in \boldsymbol{R}} \ & \frac{1}{n}\sum_{i=1}^{n}\xi_i+\gamma_A\boldsymbol{\alpha}^T\boldsymbol{K}\boldsymbol{\alpha}+\gamma_I\boldsymbol{f}^T\boldsymbol{L}\boldsymbol{f} \\ \text{s.t.} \ & y_i f(\boldsymbol{x}_i)\geq 1-\xi_i, \quad i=1,\ldots,n \\ & \xi_i\geq 0, \quad i=1,\ldots,n \end{aligned} \tag{19}$$

Belkin et al. [23] validated that the solution of LapSVM also satisfies the Representer Theorem. Following some transformations, the corresponding dual problem of (19) can still be represented as a QP problem and solved by the same optimization techniques as SVM. The final solution is:

$$\alpha^*=\frac{1}{2}(\gamma_A\boldsymbol{I}+\gamma_I\boldsymbol{L}\boldsymbol{K})^{-1}\boldsymbol{Y}\beta^*. \tag{20}$$

where $\boldsymbol{I}$ is an $n \times n$ identity matrix.

LapSVM relies on the Laplacian regularizer to embed the data distribution, which utilizes the local linear patches defined by the neighborhood sets $ne(\boldsymbol{x}_i)$ to approximate the data manifold. However, the regularizer emphasizes more the local than the global manifold structures. In fact, the data manifold in real-world problems always distributes non-uniformly in the high-dimensional space. When the data distribute compactly, the patch can approximate the linear locality of the manifold well. But when the data distribute sparsely, such an approximation is more likely distorted. Hence, the global geometry of the manifold should also be considered necessarily to differentiate different linear local patches under the varying data distribution conditions and characterize the manifold more faithfully.

3 Glocalization pursuit support vector machine (GPSVM)

In this section, we propose an alternative regularizer, called as Glocalization Pursuit Regularizer. The regularizer introduces a natural measure to characterize the global data distribution inspired by our previous manifold dimensional reduction method—Alternative Robust Local Embedding (ARLE) [24]—and thus can relatively faithfully capture the local and global geometry information simultaneously. We validate that the regularizer can describe the manifold more compactly than the Laplacian regularizer. Therefore, it more likely represents the data distribution factually and facilitates the subsequent design for classifier. We further embed the new regularizer into SVM and propose the alternative GPSVM algorithm. The major properties of GPSVM are discussed below.

3.1 Alternative robust local embedding (ARLE)

ARLE is originally proposed to mitigate the outlier sensitivity problem in Locally Linear Embedding (LLE) [25]. For distinguishing the different linear local patches on the manifold that are constructed by the normal data or the outliers, ARLE defines the local and global weights respectively for each data point to characterize the data distribution.

The local weight measures the relative similarity between each data point $\boldsymbol{x_i}$ and its neighbors, which implies that among all the neighbors of $\boldsymbol{x_i}$, the bigger the local weight between the point and $\boldsymbol{x}_i$ is, the more similar is it to $\boldsymbol{x}_i$ [24]. Concretely, we first use

$$s_{ij}=\exp(-\|\boldsymbol{x}_i-\boldsymbol{x}_j\|/\sigma^2) \tag{21}$$

to measure the similarity between $\boldsymbol{x}_i$ and its neighbor $\boldsymbol{x}_j$ just like the weights in the Laplacian regularizer. Then, we

compute a relatively robust reconstruction $\hat{\boldsymbol{x}}_i$ to $\boldsymbol{x}_i$ by its k neighbors through minimizing:

$$\varepsilon_i = \sum_{j=1}^{k} s_{ij} \left\| \hat{\boldsymbol{x}}_i - \boldsymbol{x}_j \right\|^2 \tag{22}$$

Let $\partial \varepsilon_i / \partial \hat{\boldsymbol{x}}_i = 0$, the optimal reconstruction is given by:

$$\hat{\boldsymbol{x}}_i = \sum_{j=1}^{k} s_{ij} \boldsymbol{x}_j / \sum_{j=1}^{k} s_{ij} \tag{23}$$

Let $d_i = \sum_{j=1}^{k} s_{ij}$, we get the local weight:

$$w_{ij}^{\text{local}} = s_{ij} / d_i \tag{24}$$

The global weight is a natural measure based on the local weights:

$$w_i^{\text{global}} = d_i / d \tag{25}$$

where $d = \sum_{i=1}^{\text{n}} d_i$.

In the probability sense, the local weight reflects the confidence degree of the neighbor $\boldsymbol{x}_j$ relative to $\boldsymbol{x}_i$. The bigger local weight means that $\boldsymbol{x}_j$ is more likely a normal data near $\boldsymbol{x}_i$, otherwise $\boldsymbol{x}_j$ might be an outlier. Similarly, the global weight reflects the confidence degree of the local neighborhood relative to the whole manifold, which can partially tell whether the linear patch defined by the neighborhood can characterize the local geometry faithfully.

Finally, ARLE computes the low-dimensional embedding coordinate $\boldsymbol{z_i}$ for $\boldsymbol{x}_i$ by minimizing the following weighted cost function:

$$\begin{aligned} \boldsymbol{\Phi} &= \sum_{i=1}^{n} w_i^{\text{global}} \left\| z_i - \sum_{x_j \in ne(x_i)} w_{ij}^{\text{local}} z_j \right\|^2 \\ &\text{s.t.} \quad \boldsymbol{Z} \boldsymbol{I}_n = 0 \\ &\qquad \frac{1}{n} \boldsymbol{Z} \boldsymbol{W}^{\text{global}} \boldsymbol{Z}^T = \boldsymbol{I} \end{aligned} \tag{26}$$

where $\boldsymbol{I}_n = [1, \ldots, 1]^T \in \boldsymbol{R}^n$, $\boldsymbol{W}^{\text{global}} = \text{diag}[w_1^{\text{global}}, \ldots, w_n^{\text{global}}]$ and $\boldsymbol{I}$ is the identity matrix.

3.2 Glocalization pursuit regularizer

After the robust reconstruction in ARLE, each $\boldsymbol{x}_i$ has two weights: the local weight reflects the local compactness in the local linear patch, and the global weight reflects the global compactness of the patch on the whole manifold that can serve as a natural global structure measure. So here we introduce the two weights into the construction of the new regularizer $\|\boldsymbol{f}\|^2_{\text{Glocal}}$ to grasp the global and local manifold information simultaneously.

The new regularizer can be constructed as follows:

1. Constructing the adjacency graph: find the set $ne(\boldsymbol{x}_i)$ of the k nearest neighbors of each data point $\boldsymbol{x_i}$ in the cth class ($c = 1, 2$) and put the edges among the $ne(\boldsymbol{x}_i)$.
2. Computing the local weight:

$$w_{ij}^{\text{local}} = \begin{cases} s_{ij}/d_i & \text{if } \boldsymbol{x}_i \in ne(\boldsymbol{x}_j) \quad \text{or} \quad \boldsymbol{x}_j \in ne(\boldsymbol{x}_i) \\ 0 & \text{otherwise} \end{cases}$$

3. Computing the global weight:

$$w_i^{\text{global}} = d_i / \sum_{i=1}^{n} d_i$$

4. Constructing the weighted matrix:

$$\boldsymbol{L}_c^{\text{Glocal}} = (\boldsymbol{I} - \boldsymbol{W}_c^{\text{local}}) \boldsymbol{W}_c^{\text{global}} (\boldsymbol{I} - \boldsymbol{W}_c^{\text{local}})^T, \quad c = 1, 2 \tag{27}$$

5. Constructing the Glocalization pursuit regularizer $\|\boldsymbol{f}\|^2_{\text{Glocal}}$:

$$\begin{aligned} \|\boldsymbol{f}\|^2_{\text{Glocal}} &= \sum_{c=1}^{2} \sum_{i=1}^{n_c} w_i^{\text{global}} \left(f_c(\boldsymbol{x}_i) - \sum_{x_j \in ne(x_i)} w_{ij}^{\text{local}} f_c(\boldsymbol{x}_j) \right)^2 \\ &= \sum_{c=1}^{2} \boldsymbol{f}_c^T \boldsymbol{L}_c^{\text{Glocal}} \boldsymbol{f}_c = \boldsymbol{f}^T \boldsymbol{L}^{\text{Glocal}} \boldsymbol{f} \end{aligned} \tag{28}$$

where $\boldsymbol{L}^{\text{Glocal}} = \begin{bmatrix} \boldsymbol{L}_1^{\text{Glocal}} & \\ & \boldsymbol{L}_2^{\text{Glocal}} \end{bmatrix}$.

We can further prove that the new regularizer has the more compact manifold description than the Laplacain regularizer.

Proposition 1 *Following the definition of* (28), *we have*

$$\|\boldsymbol{f}\|^2_{\text{Glocal}} \leq \|\boldsymbol{f}\|^2_I \tag{29}$$

Proof Without loss of generalization, we first consider the class one ($c = 1$). Let the numbers of the two classes be n_1 and n_2, respectively. Then,

$$\begin{aligned} \boldsymbol{f}_1^T \boldsymbol{L}_1^{\text{Glocal}} \boldsymbol{f}_1 &= \boldsymbol{f}_1^T (\boldsymbol{I} - \boldsymbol{W}_1^{\text{local}}) \boldsymbol{W}_1^{\text{global}} (\boldsymbol{I} - \boldsymbol{W}_1^{\text{local}})^T \boldsymbol{f}_1 \\ &= \sum_{i=1}^{n_1} w_i^{\text{global}} \left(f(\boldsymbol{x}_i) - \sum_{x_j \in ne(x_i)} w_{ij}^{\text{local}} f(\boldsymbol{x}_j) \right)^2 \\ &= \frac{1}{d} \sum_{i=1}^{n_1} \sum_{x_j \in ne(x_i)} s_{ij} \left(f(\boldsymbol{x}_i) - \frac{\sum_{x_j \in ne(x_i)} s_{ij} f(\boldsymbol{x}_j)}{\sum_{x_j \in ne(x_i)} s_{ij}} \right)^2 \\ &\leq \frac{1}{d} \sum_{i=1}^{n_1} \sum_{j=1}^{n_1} s_{ij} \big(f(\boldsymbol{x}_i) - f(\boldsymbol{x}_j) \big)^2 \\ &= \frac{1}{d} \boldsymbol{f}_1^T \boldsymbol{L}_1 \boldsymbol{f}_1 \end{aligned} \tag{30}$$

Following the same deduction, for the class two ($c = 2$), we obtain

$$\boldsymbol{f}_2^T \boldsymbol{L}_2^{\text{Glocal}} \boldsymbol{f}_2 \leq \boldsymbol{f}_2^T \boldsymbol{L}_2 \boldsymbol{f}_2$$

Consequently, we have

$$\|\boldsymbol{f}\|^2_{\text{Glocal}} = \sum_{c=1}^{2} \boldsymbol{f}_c^T \boldsymbol{L}_c^{\text{Glocal}} \boldsymbol{f}_c \leq \sum_{c=1}^{2} \boldsymbol{f}_c^T \boldsymbol{L}_c \boldsymbol{f}_c = \|\boldsymbol{f}\|_I^2$$

□

The above proposition validates that for the same dataset and the same classifier function $\boldsymbol{f}$, the new regularizer can get smaller value than the Laplacian regularizer, implying that it can make the data distribution more compact in the space projected by $\boldsymbol{f}$. The new regularizer introduces the global weight to reflect the different local linear patch distributions on the manifold to some extent. The bigger the value of the global weight, the more compact the data distribution on the patch is, which denotes that the patch is more likely reliable for description of the manifold geometry. Otherwise, if the patch is assigned a smaller weight, its influence will be suppressed in the description. Consequently, the new regularizer can more likely reach the target that the similar data on the manifold within a compact patch in the original space may organize more compactly in the projection space and the dissimilar data within a sparse patch may project more separably, to represent the manifold geometry more faithfully.

3.3 GPSVM algorithm

We embed the new regularizer into SVM objective and present an alternative graph-based SVM algorithm—GPSVM. The corresponding objective optimization problem is as follows:

$$\begin{aligned} \min_{f \in \boldsymbol{H}_K, \xi_i \in \boldsymbol{R}} \; & \frac{1}{n} \sum_{i=1}^{n} \xi_i + \gamma_A \|\boldsymbol{f}\|_K^2 + \gamma_{\text{Glocal}} \boldsymbol{f}^T \boldsymbol{L}^{\text{Glocal}} \boldsymbol{f} \\ \text{s.t.} \quad & y_i f(\boldsymbol{x}_i) \geq 1 - \xi_i, \quad i = 1, \ldots, n \\ & \xi_i \geq 0, \quad i = 1, \ldots, n \end{aligned} \tag{31}$$

Similarly to LapSVM [23], we can also easily prove that the solution to this problem admits a representation in terms of an expansion over the training samples. The proof is based on a simple orthogonality argument as in LapSVM [23, 26, 27]:

Proposition 2 *The solution of the optimization problem* (31) *satisfies the Representer Theorem. That is,*

$$f^*(\boldsymbol{x}) = \sum_{i=1}^{n} \alpha_i^* K(\boldsymbol{x}, \boldsymbol{x}_i) \tag{32}$$

where $K(\boldsymbol{x}, \cdot)$ *is the kernel function defined in the RKHS* H_k.

Proof Any function $\boldsymbol{f} \in H_K$ can be uniquely decomposed into a component $\boldsymbol{f}_{\|}$ in the linear subspace spanned by the kernel functions $\{K(\boldsymbol{x}_i, \cdot)\}_{i=1}^n$, and a component $\boldsymbol{f}_{\perp}$ orthogonal to it. Thus,

$$\boldsymbol{f} = \boldsymbol{f}_{\|} + \boldsymbol{f}_{\perp} = \sum_{i=1}^{n} \alpha_i K(\boldsymbol{x}_i, \cdot) + \boldsymbol{f}_{\perp}$$

By the reproducing property, as the following arguments show, the evaluation of $\boldsymbol{f}$ on any data point x_j $(1 \leq j \leq n)$ is independent of the orthogonal component $\boldsymbol{f}_{\perp}$:

$$\begin{aligned} \boldsymbol{f}(\boldsymbol{x}_j) &= \langle \boldsymbol{f}, K(\boldsymbol{x}_j, \cdot) \rangle \\ &= \left\langle \sum_{i=1}^{n} \alpha_i K(\boldsymbol{x}_i, \cdot), K(\boldsymbol{x}_j, \cdot) \right\rangle + \langle \boldsymbol{f}_{\perp}, K(\boldsymbol{x}_j, \cdot) \rangle \end{aligned}$$

Since the second term vanishes and $\langle K(\boldsymbol{x}_i, \cdot), K(\boldsymbol{x}_j, \cdot) \rangle = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$, it follows that $f(\boldsymbol{x}_j) = \sum_{i=1}^n \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Thus, the empirical terms involving the loss function and the intrinsic norm in the optimization problem (31) depend only on the value of the coefficients $\{\alpha_i\}_{i=1}^n$ and the Gram matrix of the kernel function.

Indeed, since the orthogonal component only increases the norm of $\boldsymbol{f}$ in H_k:

$$\|\boldsymbol{f}\|_K^2 = \left\| \sum_{i=1}^{n} \alpha_i K(\boldsymbol{x}_i, \cdot) \right\|_K^2 + \|\boldsymbol{f}_{\perp}\|_H^2 \geq \left\| \sum_{i=1}^{n} \alpha_i K(\boldsymbol{x}_i, \cdot) \right\|_K^2$$

It follows that the minimizer of (31) must have $\boldsymbol{f}_{\perp} = 0$ and therefore admits a representation $f^*(\cdot) = \sum_{i=1}^n \alpha_i K(\boldsymbol{x}_i, \cdot)$. □

In the practical applications, we often add a scalar b in the (32), which is an unregularized bias term. Hence, we can redescribe the optimization problem (31) as:

$$\begin{aligned} \min \; & \frac{1}{n} \sum_{i=1}^{n} \xi_i + \gamma_A \boldsymbol{\alpha}^T \boldsymbol{K} \alpha + \gamma_{\text{Glocal}} \boldsymbol{\alpha}^T \boldsymbol{K} \boldsymbol{L}^{\text{Glocal}} \boldsymbol{K} \alpha \\ \text{s.t.} \quad & y_i \left(\sum_{i=1}^{n} \alpha_i K(\boldsymbol{x}, \boldsymbol{x}_i) + b \right) \geq 1 - \xi_i, \quad i = 1, \ldots, n \\ & \xi_i \geq 0, \quad i = 1, \ldots, n \end{aligned} \tag{33}$$

where $\boldsymbol{K}$ is the Gram matrix and $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_n]^T$.

Introducing the Lagrange multipliers, we obtain the primal problem as:

$$\begin{aligned} L(\boldsymbol{\alpha}, \xi, b, \boldsymbol{\eta}, \boldsymbol{\gamma}) = & \frac{1}{n} \sum_{i=1}^{n} \xi_i + \boldsymbol{\alpha}^T \left(\gamma_A \boldsymbol{K} + \gamma_{\text{Glocal}} \boldsymbol{K} \boldsymbol{L}^{\text{Glocal}} \boldsymbol{K} \right) \boldsymbol{\alpha} \\ & - \sum_{i=1}^{n} \eta_i \left[y_i \left(\sum_{i=1}^{n} \alpha_i K(\boldsymbol{x}, \boldsymbol{x}_i) + b \right) - 1 + \xi_i \right] \\ & - \sum_{i=1}^{n} \gamma_i \xi_i \end{aligned} \tag{34}$$

Differentiating $L(\boldsymbol{\alpha}, \xi, b, \boldsymbol{\eta}, \boldsymbol{\gamma})$ with respect to $\boldsymbol{\alpha}$, ξ_i, and b, and setting the results equal to zero, we get the following conditions of optimality:

$$\frac{\partial L}{\partial \boldsymbol{\alpha}} = 2\left(\gamma_A \boldsymbol{K} + \gamma_{\text{Glocal}} \boldsymbol{K}\boldsymbol{L}^{\text{Glocal}}\boldsymbol{K}\right)\boldsymbol{\alpha} - \boldsymbol{K}\boldsymbol{Y}\boldsymbol{\eta} = 0$$
$$\frac{\partial L}{\partial \xi_i} = \frac{1}{n} - \eta_i - \gamma_i = 0$$
$$\frac{\partial L}{\partial b} = \sum_{i=1}^{n} \eta_i y_i = 0 \tag{35}$$

where $\boldsymbol{Y} = \text{diag}[y_1, y_2, \ldots, y_n]$.

Substituting (35) into the Lagrange function (34), we can get the dual problem:

$$\boldsymbol{\eta}^* = \max_{\eta \in R^n} \sum_{i=1}^{n} \eta_i - \frac{1}{2}\boldsymbol{\eta}^T \boldsymbol{G}\boldsymbol{\eta}$$
$$\text{s.t.} \quad \sum_{i=1}^{n} \eta_i y_i = 0$$
$$0 \le \eta_i \le \frac{1}{n}, \quad i = 1, \ldots, n \tag{36}$$

where $\boldsymbol{G} = \frac{1}{2}\boldsymbol{Y}\boldsymbol{K}(\gamma_A \boldsymbol{I} + \gamma_{\text{Glocal}}\boldsymbol{L}^{\text{Glocal}}\boldsymbol{K})^{-1}\boldsymbol{Y}$.

The optimization problem (36) is a typical convex optimization similar to SVM, which can be solved by the same QP technique. Let the optimal solution be $\boldsymbol{\eta}^*$, the corresponding expansion coefficient in (32) is

$$\boldsymbol{\alpha}^* = \frac{1}{2}(\gamma_A \boldsymbol{I} + \gamma_{\text{Glocal}}\boldsymbol{L}^{\text{Glocal}}\boldsymbol{K})^{-1}\boldsymbol{Y}\boldsymbol{\eta}^* \tag{37}$$

4 Experiments

To evaluate the proposed GPSVM algorithm, we have performed sets of experiments in both toy and real datasets. In the toy problem, we compared GPSVM with LapSVM and SVM in a two-moon dataset classification case. Furthermore, several real-world datasets in the UCI database (the UCI Machine Learning Repository) have been used to evaluate the classification accuracies derived from the three algorithms.

Due to the relatively better performance of the kernel version, here we uniformly compare the algorithms in the Radial Basis Function (RBF) kernel and soft margin cases. The width parameter σ in the RBF kernel and the regularizer parameters are selected from the set $\{2^{-8}, 2^{-7}, \ldots, 2^{7}, 2^{8}\}$ by the cross-validation. We apply the SMO algorithm [28] to solve the QP problems in the three algorithms.

4.1 Toy problem

Two-moon dataset is a common used toy problem in the comparisons of the classification algorithms. The dataset is first randomly generated under two uniform distributions for the two classes respectively and then made by the sine and cosine transformations. Finally, the randomly normal distributed data are added into the dataset as the noises whose variance can be appointed in advance. Here, we choose the dataset that contains one hundred samples in each class and the variance of the noise 1.4. As shown in Fig. 1a, '·' and '+' denote the training data in the two classes, as well as '*' and '×' denote the testing data.

For characterizing the global manifold trend of the dataset optimally, we first classify the data according to the generating uniform functions without the normal noises. The corresponding discriminant plane is illustrated by the dash dot line as the baseline in Fig. 1. Then, we compare the classification performance of SVM (Fig. 1b), LapSVM (Fig. 1c) and GPSVM (Fig. 1d) in the noise environment. In LapSVM and GPSVM, the number of the k nearest neighbors is fixed to 10. The three subfigures show the discriminant planes of the three algorithms in the dataset. Furthermore, the respective training and testing accuracies are listed in Table 1.

From the results, it can be seen that

- SVM only concerns the separability between the two classes, rather than the data geometry. As a result, the derived discriminant plane always approximately lies in the middle of the boundary points in the training set [29] and cannot reflect the trend of the data manifold completely (Fig. 1b). The difference between the discriminant plane and the baseline is quite obvious. Though SVM can achieve the best training accuracy in the training set, it has poor performance in the testing set.
- LapSVM introduces the local structure of the data manifold into SVM by the Laplacian regularizer and thus can describe the data distribution to some extent. However, as shown in Fig. 1c, due to less emphasizing the global structure information, LapSVM is relatively sensitive to the local variations of the data, and the corresponding discriminant plane is heavily affected by the specific points near the boundary that are more likely noises. Though the plane of LapSVM fits the baseline better than SVM, there also has a big disparity between the two planes, implying that LapSVM cannot characterize the global manifold well. Consequently, LapSVM still has worse performance than GPSVM in the testing set.
- GPSVM captures the local and global structure information of the data manifold simultaneously and gets more reasonable discriminant plane than both SVM and LapSVM which basically accords with the baseline. Therefore, GPSVM has the best classification performance in the testing set, which means that GPSVM has better generalization ability owing to the more reasonable description of the data global distribution.

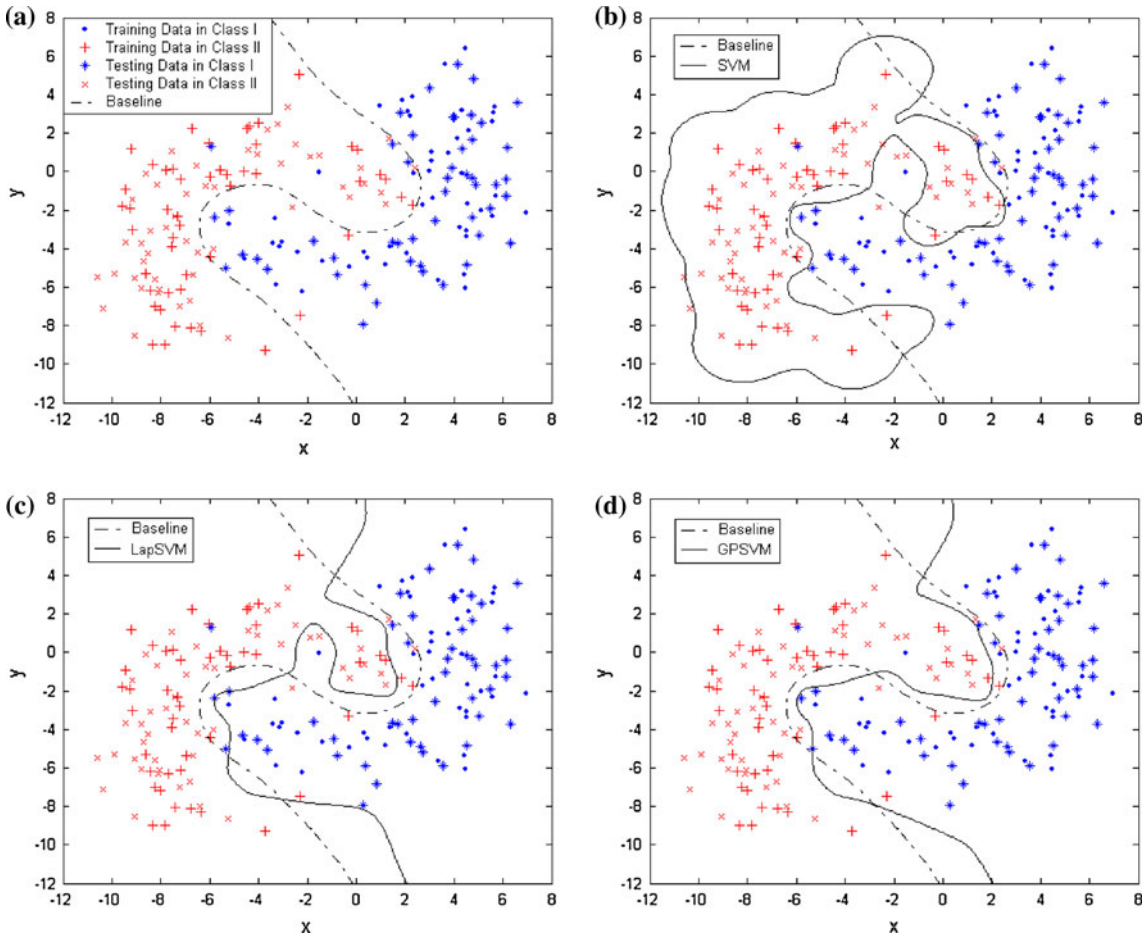

Fig. 1 The discriminant planes in the two-moon dataset: Baseline (**a**), SVM (**b**), LapSVM (**c**), and GPSVM (**d**)

Table 1 The training and testing accuracies (%) of Baseline, SVM, LapSVM, and GPSVM in the two-moon dataset

	Baseline	SVM	LapSVM	GPSVM
Training set	96.00	100.00	96.00	96.00
Testing set	95.00	92.00	92.00	***96.00***

The emphasis values are the best testing accuracies among the compared algorithms in the datasets

4.2 UCI dataset

To further investigate the effectiveness of our proposed GPSVM, we also evaluate its performance in several real-world datasets in the UCI database. For each dataset, we divide the samples into two non-overlapping training and testing sets, and each set contains almost half of samples in each class respectively. This process is repeated ten times to generate ten independent runs for each dataset and then the average results are reported. Throughout the experiments, we choose the best k between two and $(\min_c\{\text{number}(n_c)\} - 1)$ by the cross-validation in LapSVM and GPSVM.

4.2.1 Accuracy comparison

We list the experimental results in Table 2. In each block in the table, the first row is the training accuracy and variance, and the second row is the testing accuracy and variance. We can make several observations from the results:

- GPSVM is consistently superior to SVM in the overall datasets both in the training and testing accuracies, owing to the consideration of the data distribution geometry. Moreover, GPSVM also outperforms LapSVM in almost all the datasets except in Bupa and Ionosphere, because GPSVM further incorporates with the global manifold structure information.
- The training and testing accuracies of GPSVM are basically comparable in the datasets, implying that GPSVM has good generalization performance. The variances further show the good stability of GPSVM.
- In order to find out whether GPSVM is significantly better than SVM and LapSVM in the statistical sense, we perform the t-test on the classification results of the ten

Table 2 The training and testing accuracies (%), and variances compared between SVM, LapSVM, and GPSVM in the UCI datasets

Dataset	Classification accuracy		
	SVM	LapSVM	GPSVM
Automobile	95.63* ± 0.01	95.75 ± 0.01	96.25 ± 0.03
	88.48* ± 0.01	87.85* ± 0.01	***88.99*** ± 0.01
Bupa	75.68* ± 0.08	79.03 ± 0.07	78.59 ± 0.03
	73.06* ± 0.06	***78.72**** ± 0.03	77.25 ± 0.02
Pima	76.04* ± 0.01	78.80 ± 0.05	78.94 ± 0.05
	77.08* ± 0.02	77.88* ± 0.03	***78.26*** ± 0.06
Ionosphere	96.80* ± 0.02	98.51 ± 0.02	98.29 ± 0.03
	95.11* ± 0.02	***98.30*** ± 0.03	98.12 ± 0.05
Sonar	86.54* ± 0.15	95.24 ± 0.09	95.37 ± 0.05
	85.00* ± 0.13	91.26* ± 0.07	***92.58*** ± 0.05
Water	98.47 ± 0.02	98.57 ± 0.04	98.61 ± 0.02
	90.51* ± 0.09	98.21 ± 0.04	***98.53*** ± 0.03
Wdbc	92.54* ± 0.01	98.94 ± 0.04	97.69 ± 0.02
	94.25* ± 0.01	94.84* ± 0.05	***95.21*** ± 0.04

The emphasis values are the best testing accuracies among the compared algorithms in the datasets

'*' Denotes that the difference between GPSVM and the other two methods is significant at 5% significance level, i.e., t-value > 1.7341

runs to calculate the statistical significance of GPSVM. The null hypothesis H_0 demonstrates that there is no significant difference between the mean number of patterns correctly classified by GPSVM and the other two methods. If the hypothesis H_0 of each dataset is rejected at the 5% significance level, i.e., the t-test value is more than 1.7341, the corresponding results in Table 2 will be denoted '*'. Consequently, as shown in Table 2, it can be clearly found that GPSVM possesses significantly superior classification performance compared with the other two methods in almost all datasets, especially according to the testing accuracies.

Remark It is worth to point out that in the experiments, the discrepancy of the accuracies between SVM and the other two algorithms is distinct, but that between LapSVM and GPSVM is relatively slight. The reason more likely lies in the looseness of the global structure measure used in GPSVM. In fact, the goal of the paper is to attempt a beneficial integration of the global manifold information into the locality graph-based classifier design, because the only consideration for the locality seems not sufficient to characterize the data geometry faithfully and globally. Actually, the present work manifests exactly that the whole or global distribution information plays a favorable role in both dimensionality reduction [24] and classifier design. However, for the being time, the related study is relatively less in pattern recognition and machine learning. In future, we will devote to research of how to develop a tighter global structure measure to further characterize the manifold accurately and improve the classifier performance.

4.2.2 Parameter selection

The option of the regularizer parameter is vital for the performance of the graph-based methods, which is still an open problem in machine learning. Usually, the parameters are selected through the cross-validation. Here we choose the dataset Pima to illustrate how the different parameter selections arouse the variations in the average testing accuracies, in order to further compare the classification performance of the three algorithms in the different parameter conditions.

We fix the number of the k nearest neighbors and the width parameter σ in the RBF kernel to 10 and 1

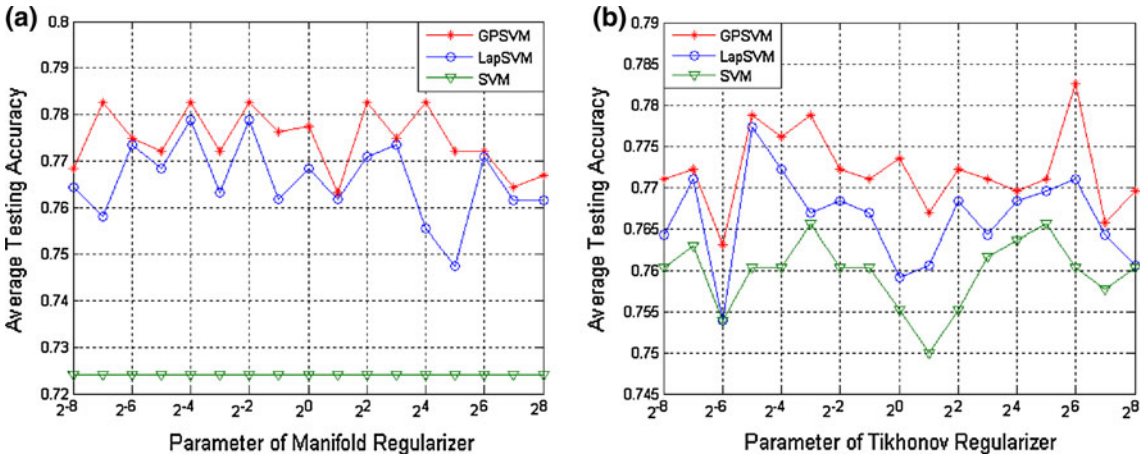

Fig. 2 The average testing accuracy variation corresponding to the different regularizer parameters in the three algorithms in Pima: parameter of manifold regularizer (**a**) and parameter of Tikhonov regularizer (**b**)

respectively. The Laplacian regularizer and Glocalization Pursuit regularizer are uniformly called as the manifold regularizer, for distinguishing from the Tikhonov regularizer. Referring to [23], we first fix the parameter of the Tikhonov regularizer to 2^{-8} and change the parameter of the manifold regularzer from 2^{-8} to 2^{8}. The variations of the average testing accuracies in the three algorithms are showed in Fig. 2a. Similarly, we then fix the parameter of the manifold regularizer to 1 and compare the accuracies corresponding to the transformation of the parameter of the Tikhonov regularizer from 2^{-8} to 2^{8}, as illustrated in Fig. 2b.

On the one hand, the accuracy curves vibrate heavily with the varying parameters in the figures, implying that the different selections of the parameters indeed severely affect the performance of the classifier. Furthermore, the values of the optimal parameter in the various algorithms are quite different as well, which validates the conclusion that without any prior knowledge, we hardly appoint the optimal parameters.

On the other hand, the figures also present that whatever the parameters is in the range under consideration, the accuracies of GPSVM all along excel those of the other two algorithms. Moreover, for the same parameter, GPSVM always possesses the best average testing accuracy, which further verifies the superior performance of GPSVM.

5 Conclusion

In this paper, we propose an alternative regularizer termed as Glocalization Pursuit regularizer. Inspired by our previous ARLE, we first introduce a natural global measure to indicate the global compactness of data manifold distribution based on the local linear patches on the graph. The global measure is then embedded into the regularizer, which has been validated that such embedment can reach more compact manifold description than the traditional Laplacian regularizer. We further add the new regularizer into SVM to propose an alternative graph-based SVM algorithm called as Glocalization Pursuit Support Vector Machine (GPSVM). GPSVM not only inherits the good properties of SVM but also remedies the relative sensitivity of LapSVM to the local variations in the data manifold due to the less emphasis on the global structure information in the Laplacian regularizer. The experimental results have demonstrated the superiority of GPSVM compared with SVM and LapSVM.

Throughout the paper, we classify the various graph-based methods from the different loss functions and regularizers. Here, we choose the framework based on the hinge loss function and the Tikhonov regularizer, and derive the GPSVM algorithm. In future, we will further incorporate the proposed Glocalization Pursuit regularizer with the other loss functions and regularizers, and systematically compare the different classification performances of these algorithms. Furthermore, here we construct the new regularizer based on the manifold dimensional reduction method ARLE. We can further refer to other new manifold methods and combine them with the regularizer to reflect the data distribution more faithfully and finally improve the classifier design.

Acknowledgments This work was supported by National Natural Science Foundations of China (Grant Nos. 60773061, 60905002, 60973097 and 61035003) and Natural Science Foundations of Jiangsu Province of China (Grant No. BK2008381).

References

1. Zhu X (2008) Semi-supervised learning literature survey. Technical report, 1530, Madison: Department of Computer Sciences, University of Wisconsin
2. Bousquet O, Chapelle O, Hein M (2003) Measure based regularization. NIPS, Canada
3. Xue H, Chen S, Zeng X (2008) Classifier learning with a new locality regularization method. Pattern Recogn 41(5):1496–1507
4. Xue H, Chen S, Yang Q (2009) Discriminatively regularized least-squares classification. Pattern Recogn 42(1):93–104
5. Wu M, Schölkopf B (2007) Transductive classification via local learning regularization. Eleventh international conference on artificial intelligence and statistics (AISTATS)
6. Haykin S (2001) Neural networks: a comprehensive foundation. Tsinghua University Press, Beijing
7. Zhang T (2004) Statistical behavior and consistency of classification methods based on convex risk minimization. Ann Stat 32:56–85
8. Zhang T (2004) Statistical analysis of some multicategory large margin classification methods. J Mach Learn Res 5:1225–1251
9. Rosasco L, Vito ED, Caponnetto A, Piana M, Verri A (2004) Are loss functions all the same? Neural Comput 16(5):1063–1076
10. Rosset S, Zhu J, Hastie T (2003) Margin maximizing loss functions. NIPS, Canada
11. Chen Z, Haykin S (2002) On different facets of regularization theory. Neural Comput 14(12):2791–2846
12. Belkin M, Niyogi P Laplacian (2001) Eigenmaps and spectral technique for embedding and clustering. NIPS, 15: Vancouver, British Columbia, Canada
13. Belkin M, Niyogi P, Sindhwani V (2005) On mainfold regularization. In: Proceedings of the 10th international workshop on artificial intelligence and statistics (AISTATS), Savannah Hotel, Barbados, 17–24
14. Zhou D, Bousquet O, Lal TN, Weston J, Schölkopf B (2004) Learning with local and global consistency. In: Thrun S, Saul L, Scholkopf B (eds) Advances in neural information processing systems 16. MIT Press, Cambridge, pp 321–328
15. Holder LB, Cook DJ (2003) Graph-based relational learning: current and future directions. ACM SIGKDD Explor Newsl 5(1):90–93
16. Cook DJ, Holder LB (2000) Graph-based data mining. IEEE Intell Syst 15:32–41
17. Zhu X (2005) Semi-supervised learning with graphs. PhD Thesis, Carnegie Mellon University. CMU-LTI-05-192
18. Belkin M, Matveeva I, P. Niyogi (2004) Regularization and semi-supervised learning on large graphs. COLT

19. Vapnik V (1998) Statistical learning theory. Wiley, New York
20. Cristianini N, Shawe-Taylor J (2004) An introduction to support vector machines and other kernel-based learning methods. Publishing House of Electronics Industry, Beijing
21. Zhu X, Ghahramani Z, Lafferty J (2003) Semi-supervised learning using Gaussian fields and harmonic functions. In: Proceedings of the 20th international conference on machine learning (ICML'03), Washington, DC, 912–919
22. Zhu X, Ghahramani Z (2002) Towards semi-supervised classification with Markov random fields. Technical report CMU-CALD-02-106. Carnegie Mellon University
23. Belkin M, Niyogi P, Sindhwani V (2004) Manifold regularization: A geometric framework for learning from examples. Department of Computer Science, University of Chicago, Tech. Rep: TR-2004-06
24. Xue H, Chen S (2007) Alternative robust local embedding. In: The international conference on wavelet analysis and pattern recognition (ICWAPR), 591–596
25. Roweis ST, Saul LK (2000) Nonlinear dimensionality reduction by locally linear embedding. Science 290(22):2323–2326
26. Schölkopf B, Smola AJ (2002) Learning with Kernels. 644. MIT Press, Cambridge
27. Belkin M (2003) Problems of learning on manifolds. PhD Thesis, The University of Chicago
28. Cristianini N, Shawe-Taylor J (2000) An introduction to support vector machines and other kernel-based learning methods. Cambridge University Press, Cambridge
29. Xue H, Chen S, Yang Q (2008) Structural support vector machine. In: The 15th international symposium on neural networks (ISNN), Part I, LNCS, vol 5263, pp 501–511