SSPS: A Semi-Supervised Pattern Shift for Classification

Enliang Hu[1,2*] Xuesong Yin[1] Yongming Wang[3] SongCan Chen[1]

1. Dept. of Computer Science & Engineering,

Nanjing University of Aeronautics & Astronautics, Nanjing, 210016, China

2. Dept. of Mathematics, Yunnan Normal University, Kunming 650092, China

3. Dept. of Management and Economics, Kunming University of Science & Technology, Kunming 650093, China

Abstract: Recently, a great amount of efforts have been spent in the research of unsupervised and (semi-)supervised *dimensionality reduction* (DR) techniques, and DR as a preprocessor is widely applied into classification learning in practice. However, on the one hand, many DR cannot necessarily lead to a better classification performance. On the other hand, DR is often plagued in the problem of estimation of retained dimensionality for real-world data. Alternatively, in this paper, we propose a new semi-supervised data preprocessing technique, named semi-supervised *pattern shift* (SSPS). The advantages of SSPS lie in the fact that not only the estimation of retained dimensionality can be avoided naturally, but a new shifted pattern representation that may be more favorable to classification is obtained as well. As a further extension of SSPS, we develop its fast and out-of-sample versions respectively, both of which are based on a shape-preserved subset selection trick. The final experimental results demonstrate that the proposed SSPS is promising and effective in classification application.

Key words: dimensionality reduction; semi-supervised learning; manifold learning; classification; semi-supervised pattern shift; out-of-sample extension.

1 Introduction

In recent years, dimensionality reduction (DR) has become an extremely popular approach in computer vision and pattern recognition application with attractive theory as well [1, 2]. Depending on how much supervised information is provided and used, DR approaches can be

* Corresponding author: Tel: +86-25-84896481-12106, Fax: +86-25-84498069. Email: helnuaa@nuaa.edu.cn

roughly categorized as unsupervised, semi-supervised and supervised. The most popular unsupervised DR approaches include principal component analysis (PCA) [3], local linear embedding (LLE) [7], isometric mapping (ISOMAP) [8], Laplacian-eigen (LE) mapping [9], kernel PCA (KPCA) [6, 17], locality preserving projection (LPP) [5], etc.. Meanwhile, the most noted full-supervised DR approach should be linear discriminant analysis (LDA) [4] and it is widely applied into classification learning including its variations. Especially, as a current hot topic, the semi-supervised DR approach has been developing and it mainly comprise semi-supervised discriminant analysis (SDA) [22], Laplacian linear discriminant analysis (LapLDA) [30], semi-supervised linear/kernel discriminant analysis (SSLDA/SSKLDA) [26], semi-supervised linear/kernel maximum margin criterion (SSMMC/SSKMMC) [26], etc.. In addition, Zhang et al. also proposed a semi-supervised dimensionality reduction (SSDR) [23] approach, which focused on incorporating both the must-link and cannot-link constraints into PCA. Generally, both unsupervised and (semi-)supervised DR approaches devote to capturing a new pattern representation through mapping original data into a lower-dimensionally latent space. Thereby, as another point of view, DR is a data preprocessor and it has widely given services to applications such as classification [10], clustering [11] and metric learning [12, 13].

However, on the one hand, some DR can not necessarily lead to better performance in classification application. For instance, following the Bengio's and Maaten's experimental results on many benchmark datasets [18, 19], it shows that some classification accuracies on data reduced by DR are even lower than those without DR. The main reason may be that, for these DRs, they are hard to balance global and local structure since they heavily focus on either globally geometric structure such as PCA or locally geometric structure such as LLE. As a consequence, omitting local structure will ignore within-class cohesion and omitting global structure will lose between-class separation partly.

On the other hand, many DR approaches nearly involve a crux — how many leading dimensionalities or components should be kept for reduced data. At the same time, it has been found that the algorithmic performance is extremely sensitive to the retained dimensionality sometimes. Despite many researchers have paid much attention to the intrinsic dimensionality estimation [14~16], such a problem is still open so far. Moreover, a more complicated problem

aroused is whether the intrinsic dimensionality is identical to the best retained dimensionality in classification learning. In fact, DR may bear a danger to yield worse separability than without DR sometimes since the patterns of high-dimensional space may be more linearly separable rather than those of lower-dimensional space by the Cover Theorem [25]. In other word, an improper retained dimensionality is more dangerous to overlap different-class patterns each other such that the between-class separation will be weaken.

To avoid the problem of retained dimensionality, a natural method is to preprocess data directly in original space. This implies to seek a new pattern representation through pattern shift instead of DR. In this way, we propose a semi-supervised data preprocessing technique called SSPS in this paper. The advantage of SSPS is that we can not only sidestep the plague of retained dimensionality problem but also generate a classification-oriented pattern representation in original space. Furthermore, in order to lessen time consumption, we develop a faster SSPS (F-SSPS for short), which bases on a shape-preserved subset selection scheme (SPSub for short). Meanwhile, an out-of-sample extension [20, 21] of SSPS can be naturally derived from F-SSPS. In addition, we can conveniently combine SSPS with DR to form a two-sides preprocessing technique, e.g., "SSPS+PCA" method is tested in our experiments.

To the best of our knowledge, we have not noticed any similar work like SSPS before. The remainder of this paper is organized as: In Section 2, we introduce some basic conception and formulate SSPS. In Section 3, with SPSub scheme, F-SSPS and an out-of-sample extension of SSPS are developed further and combining SSPS with DR such as "SSPS+DR" is also presented. In Section 4, some experimental results on both synthetic and real-world datasets are reported respectively. Discussions and conclusions are offered in the last Section

2 Semi-supervised pattern shift

The main idea of the SSPS is to draw intra-class points together while keep between-class points far away relatively so that the within-class distances become smaller and the between-class distances change further simultaneously. In order to help understand the proposed SSPS in advance, a comparative illustration of SSPS to SSKLDA, SSKMMC, LapLDA and ISOMAP is showed in Fig. 1. More specifically, in the top left of Fig. 1, a toy data consist of 4 Arcs. Where,

the class-distributions of 4 arcs are interlaced each other since 1-th, 3-th arcs belong to one class and 2-th, 4-th arcs to the other class. The semi-supervised information is five labeled points (filled with black color face) for each class. For such 4arcs data, its different 1-dimensional embeddings performed by SSKLDA, SSKMMC, LapLDA and ISOMAP[1] are respectively showed in the top row. Meanwhile, the shifted patterns of 4arcs with different parameters μ (interpreted later) are displayed in the bottom row.

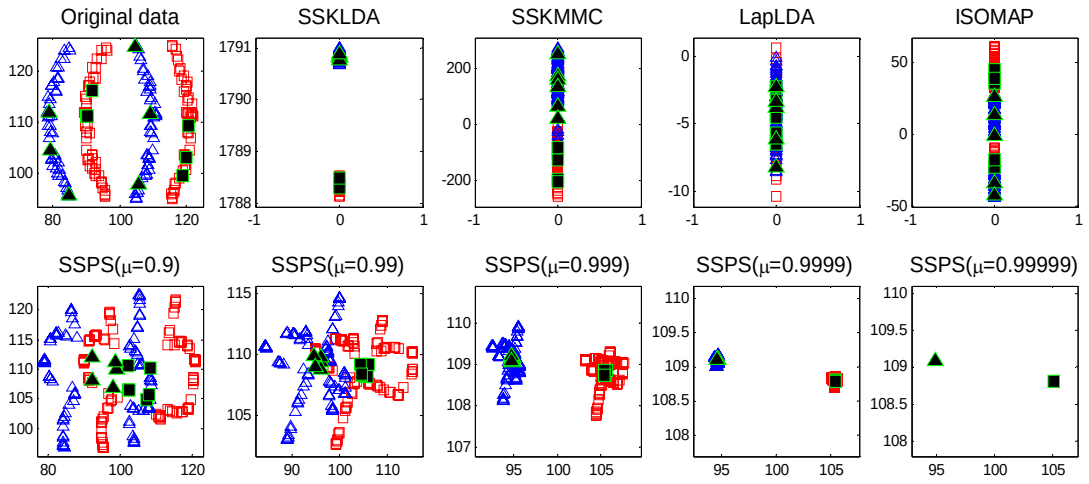

Fig. 1 Top row: original 4Arcs and 1-dimensional embeddings performed by SSKLDA, SSKMMC, LapLDA and SISOMAP. Bottom row: shifted patterns by SSPS with different μs.

From Fig. 1, we can clearly observe: 1) LapLDA and ISOMAP make between-class points overlapping each other seriously; 2) at a proper μ, the intra-class labeled points are drawn into together respectively and all unlabeled points are agglomerated into two more compact clusters; 2) both SSKLDA and SSKMMC also separate two classes away, but their resulting intra-class cohesion is less than that in shifted patterns with μ = 0.9999 and 0.99999. In summary, both the intra-class cohesion and the inter-class separation in Fig. 1 partially imply that SSPS may make benefit to classification learning than some DRs.

Now, we begin to formulate the proposed SSPS. Let $X_l = \{x_1, x_2, \cdots, x_l\}$ and $X_u = \{x_{l+1}, \cdots, x_{l+u}\}$ be respectively the labeled samples and unlabeled samples and $X = [X_l; X_u]$ is the entire samples. Sometimes, $X = [x_1, x_2, \cdots, x_l, x_{l+1}, \cdots, x_{l+u}]^T$ is also denoted as a sample-matrix and a reader can distinguish "sample-set" from "sample-matrix" by

[1] The nearest neighbor number is uniformly set 15 in SSPS, SSKLDA, SSKMMC, LapLDA and SISOMAP.

context. Let $N_k(x_i)$ consists of *k*-nearest neighbors of sample x_i and s_{ij} reflect how close between x_i and x_j, which is defined as

$$s_{ij} = \begin{cases} \exp\{-\|x_i - x_j\|^2 / t\} + 1 & \text{if } L_{ij} = 1 \\ \exp\{-\|x_i - x_j\|^2 / t\} & \text{if } x_i \in N_k(x_j) \text{ or } x_j \in N_k(x_i) \cdot \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Where, L_{ij} indicates whether a sample pair (x_i, x_j) belongs to same class or not, i.e., L_{ij} = 1 if *label* (x_i) = *label* (x_j) and x_i, $x_j \in X_l$, otherwise zero. For each $x_i \in X$, denoting its shifted representation as x_i^* such that the shifted pattern set of $X = [x_1, \cdots x_l, x_{l+1}, \cdots, x_{l+u}]^T$ can be naturally written $X^* = [x_1^*, \cdots, x_l^*, x_{l+1}^* \cdots, x_{l+u}^*]^T$. Thus, we can formulate SSPS by minimizing the follows objective:

$$\Re(X^*) = \frac{1}{2}\left(\mu \sum_{i,j=1}^{l+u} s_{ij} \left\|x_i^* - x_j^*\right\|^2 + (1-\mu) \sum_{i=1}^{l+u} \left\|x_i - x_i^*\right\|^2\right). \quad (2)$$

Within parenthesis of eq. (2), the first and second terms respectively model locally and globally geometric structures of original data *X*, and both of them are integrated by a tradeoff parameter μ restricted to [0,1). More specifically, in the first term, the distance of a pair (x_i, x_j) will become small if s_{ij} is large (i.e. x_i and x_j are close to each other in original space or they have same class label), implying that the first term aims at preserving not only local structure but also class relationship. In the second term, the distance between x_i and x_i^* measure a shift level of x_i. Meanwhile, the second term has two functions: one is to prevent eq. (2) from degenerating into zeros; the other is to preserve global structure partially. Similar to the PCA criterion in minimizing reconstruction error between the original points and its reconstructed point, we can control shift level by the difference between the original point and its shifted point. So, analogous to using PCA criterion in preserving global structure in ref. [23], the second term of eq. (2) also has ability to global structure preservation for shifted patterns.

As mentioned in ref. [30], global and local structures can be complementary to each other even though one of them may be more important than the other one in certain applications. In eq. (2), if taking a big μ value, local geometric structure will dominate the shifted result, otherwise global structure will act a dominative role. In fact, a big μ value will be likely to capture more compact

shifted patterns when classes are clearly-separated. On the contrary, a small μ value should be safer to avoid different-class shifted patterns further overlapping to each other when classes are not clearly-separated.

Let $S = (s_{ij})$ and $D = \mathrm{diag}(d_{ii})$ with $d_{ii} = \sum_j s_{ij}$, after the simple analysis, we can rewrite eq. (2) in matrix form as follows:

$$\Re(X^*) = \frac{\mu}{2} trace\left(X^{*T}(D-S)X^*\right) + \frac{1-\mu}{2} trace\left((X^*-X)(X^*-X)^T\right).$$

Thus, the optimal shifted patterns can be obtained by the following minimization objective:

$$\tilde{X}^* = \arg\min_{X^*} \Re(X^*). \tag{3}$$

For minimizing eq. (3), after computing and zeroing its derivative with respect to X^*, we have

$$\left((1-\mu)I + \mu(D-S)\right)X^* - (1-\mu)X = 0.$$

Where, I is an identity matrix whose size is clear from the context. Consequently, we obtain

$$\tilde{X}^* = \left(I + \frac{\mu}{1-\mu}(D-S)\right)^{-1} X \quad \text{for} \quad \mu \in [0,1).$$

If restricting $\sum_j s_{ij} = 1$ (i.e. $D = I$), then $\tilde{X}^* = (1-\mu)\left(I + \mu S\right)^{-1} X$. After denoting $A(\mu) = (1-\mu)\left(I + \mu S\right)^{-1}$, we have

$$\tilde{X}^* = A(\mu)X. \tag{4}$$

From eq. (4), a shift transformation T can be induced as follows:

$$T: X \to \tilde{X}^* \quad \text{or} \quad \tilde{X}^* = T(X,\mu) = A(\mu)X. \tag{5}$$

According to eqs. (4)~(5), we know $T(X,0) = X$ and $\lim_{\mu \to 1} T(X,\mu) = \mathbf{0}$ (zero matrix). We call $\tilde{X}^*$ the shifted image of X and name $A(\mu)$ shift matrix. Now, let us review the bottom row of Fig. 1, in which the different shifted patterns of 4Arcs with μ = 0.9, 0.99, 0.999, 0.9999 and 0.99999 are shown in order. Below, we will give some extensions of the original SSPS.

3 Extending SSPS to F-SSPS and out-of-sample cases

The main computation complexity of eq. (4) is $O(n^3)$ (n = l+u) from matrix inverse calculation, implying that SSPS cannot scale well. At the same time, SSPS hasn't out-of-sample predicting ability due to its transductive property [29]. Therefore, in this section, we will further discuss two issues: 1) how to reduce computation cost in eq. (4) further. 2) how to extend SSPS to adapt out-of-sample data. In addition, we combine SSPS with DR such as "SSPS+PCA" scheme when a DR preprocessing is needed too.

3.1 SPSub and F-SSPS

In order to speed up SSPS, we use a shape-preserved subset selection scheme named SPSub to reduce work-set in eq. (4). Inspired by the subset selection trick "SmartSub" in ref. [24], points located in high density region should have a higher priority to be sampled, i.e., the geometry shape of data's distribution can be preserved as much as possible after "SmartSub" selection. Considering that a too small-size work-set cannot maintain the distributive shape of original data, we need control the ratio of sizes of work-set size to off-work-set.

Let I_{in} and I_{out} respectively correspond to the indices of work-set X_S (i.e., selected points) and off-work set X_R. Especially, we should ensure that all labeled samples are chosen in I_{in}. Thus, instead of calculating $\tilde{X}_R^*$ by $T(X_R, \mu)$, our F-SSPS aims at approximately obtaining $\tilde{X}_R^*$ through the locally lest square error reconstruction from $\tilde{X}_S^*$. The reconstruction coefficients w_{ij}s can be gotten as follows:

$$w_{ji} = \arg\min \left\| x_j - \sum_{x_i \in X_S \cap N_k(x_j)} w_{ji} x_i \right\|^2, \quad \forall x_j \in X_R . \tag{6}$$

Thus, we will get $\tilde{X}_R^* = \left\{ \tilde{x}_j^* \middle| x_j \in X_R \right\}$, in which each x_j's shifted pattern $\tilde{x}_j^*$ is approximately reconstructed as

$$\tilde{x}_j^* = \sum_{\tilde{x}_i^* \in \tilde{X}_S^*} w_{ji} x_i^* . \tag{7}$$

We outline SPSub and F-SSPS in Tab. 1 and Tab. 2 respectively.

Tab. 1 The procedure of shape-preserved subset selection (SPSub)

$[X_S, X_R]$ = **SPSub**(X, *s_ratio*)
Input : $X = \{x_1,\ldots, x_l, x_{l+1}, \ldots, x_{l+u}\}$; *s_ratio* – selection ratio. **Output**: X_S, X_R .
Initialization : $\upsilon_{ij} = \exp(\frac{\|x_i - x_j\|}{2\sigma})$ $\forall x_i, x_j \in X$; $I_S \leftarrow \{1,\ldots,l\}$; $I_R \leftarrow \{l+1,\ldots,l+u\}$.
While $\frac{\|I_S\|}{\|I_S\|+\|I_R\|} < s\_ratio$ Find $i = \arg\max_{i\in I_{out}}(\sum_j \upsilon_{ij})$; $I_S \leftarrow I_S \cup \{i\}$, $I_R \leftarrow I_R \setminus \{i\}$; end while Return $X_S = \{x_i \mid x_i \in I_S\}$ and $X_R = \{x_i \mid x_i \in I_R\}$.

Tab. 2 The main steps of fast SSPS (F-SSPS)

Inputs:	X– original samples; μ– trade-off factor; *s_ratio*– selection ratio.
Output:	$\tilde{X}^*$ – the shifted pattern of X.
Step 1:	$[X_S, X_R]$ = SPSub(X, *s_ratio*); (where X_S and X_R are the work-set and off-work-set respectively)
Step 2:	Obtain $\tilde{X}_S^* = T(X_S, \mu)$ according to eq. (4);
Step 3:	Get $\tilde{X}_R^*$ by eqs. (6)~(7);
Step 4：	Return $\tilde{X}^* = \tilde{X}_S^* \cup \tilde{X}_R^*$.

In Tab. 2, the main time complexities in Steps 1~3 are approximately $O(|I_S|^2)$, $O(|I_S|^3)$ and $O(|I_R||k|^3)$ respectively. So, the total complexity of F-SSPS is about $O(|I_S|^3+|I_R||k|^3)$, which is far lower than $O((|I_S|+|I_R|)^3)$ in the original SSPS at a lager $|I_R|$.

3.2 An out-of-sample extension of SSPS

An out-of-sample extension of SSPS can be conveniently and naturally induced from F-SSPS. More specifically, Let X_{in} and X_{out} be respectively in-sample and out-of-sample data. After setting $X_S = X_{in}$ and $X_R = X_{out}$, we can directly compute $\tilde{X}_R^*$ as the shifted patterns of X_{out} by F-SSPS.

3.3 SSPS+DR

Although directly working in original space, SSPS can be also naturally and conveniently incorporated into DR approach. In other words, when a DR procedure is needed too, we can combine SSPS with DR. Concretely, if each v_i is a projective vector, then $V = (v_1, v_2, \ldots, v_d)$ becomes a projective matrix severed as DR. Based on both V and A (shift matrix), a new preprocessed pattern representation is obtained as

$$\hat{X} = AXV \,. \tag{8}$$

It is interesting that eq. (8) can be regarded as a two-side preprocessing for X, i.e., left SSPS and right DR. Furthermore, "SSPS+DR" can be interpreted as first SSPS then DR or first DR then SSPS.

As an illustration, the results of SSPS+PCA on 4Arcs are displayed in Fig. 2, from which we can observe that PCA can retain the fruits of good separability obtained by SSPS with μ = 0.999~0.99999).

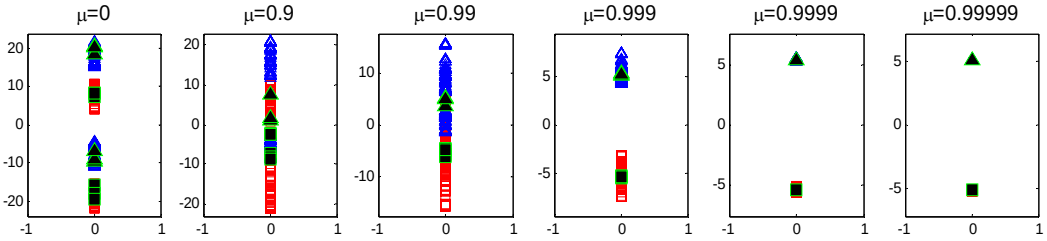

Fig. 2 The resulted patterns of 4Arcs by SSPS+PCA(1D) for different shift levels in SSPS respectively.

In following experiment part, we will evaluate SSPS in semi-supervised classification settings.

4 The experimental part

In this paper, we will use the SSPS as data preprocessor before transductive classification [29], in which many traditional supervised classifiers work empirically unsatisfactory due to only very few labeled training samples available. For testing SSPS in classification, our procedure consists of two steps: 1) capturing the shifted patterns of both labeled and unlabeled samples by SSPS; 2) training and predicting a classification algorithm on the labeled and unlabeled points of shifted patterns respectively.

4.1 Experiment settings

We execute our experiment on 3 synthetic datasets and 4 real-world datasets throughout this paper, some of whose basic description are listed in Tab. 3.

Tab. 3 Basic descriptions of seven datasets

Dataset	# of classes	# of dimensionalities	# of points	comment	citation
4Arcs	2	2	200	synthetic	
3Circles	2	2	150	synthetic	
3Spirals	2	3	378	synthetic	[28]
Digit1	2	241	1500	real-world	[18]
USPS	10	256	1500	real-world	[18]
Control	6	60	600	real-world	[28]
Waveform	2	21	1652	Real-world	[30]

Following in ref. [18], we set the nearest neighbor number k = 5 in $N_k(\cdot)$ uniformly such that many local "clusters" are firstly connected by such local relationship. To avoid "isolated regions", we link all "clusters" in a global connected graph by adding some shortest edges generated by *minimum spanning tree* algorithm.

As a comparison, SSPS, F-SSPS and SSPS+PCA are compared with SSKLDA, SSKMMC and LapLDA. In addition, ISOMAP acts as a baseline. The tuning parameters (i.e., kernel parameter and regularization parameter, etc..) is used in SSKLDA, SSKMMC and LapLDA by 3-folds cross validation under labeled # = 10%, where Gaussian kernel is always used in SSKLDA and SSKMMC. At the same time, the tradeoff parameter μ in eq. (1) is selected by cross validation on set $1-\{1.5^{0},1.5^{-1},1.5^{-2}\cdots,1.5^{-i},\cdots,1.5^{-25}\}$.

The reported classification accuracies respectively come from 5 typical supervised classifiers: nearest neighbor (NN), Fisher linear discriminator (FLD), manifold rank (MRank) [31], naïve Bayes (NB) and radial base function network (RBFNet). To evaluate our approach under the different cases of labeled sample number, we give two settings. Concretely, for datasets Digit1 and USPS, we follow the two settings in ref. [18], i.e., the labeled-sample number = 10 and 100 respectively. For 4Arcs, 3Circles, 3Spirals, Control and Waveform, we provide two settings of the labeled-sample number = 5% and 10% respectively. Each reported accuracy on such two settings is averaged over 12 trials.

4.2 Illustrations and Classification experiments on synthetic datasets

To help understand SPSub and F-SSPS further, for 3Circles and 3Spirals, we display their results of subset selection by SPSub (*s_ratio* = 0.5) within Figs. 3~6. Therein, we respectively mark the work-set points of class +1 and -1 with symbol '□' and '△', while the off-work-set points with symbol '✳' and '＋' respectively. In addition, we also draw a link-line between each work-point pair *i* and *j* with $s_{ij} \neq 0$ and fill each 5 labeled points in one circle or spiral with black face color.

Despite only keeping 50% points in work-set, we can observe that SPSub almost doesn't destroy the original geometric distribution in the middle plots of Figs. 3~4. Meanwhile, in right plots of Figs. 3~4, we observe that SSPS can squeeze intra-class points closer and make inter-class points far away relatively no matter to 3Circles or 3Spirals.

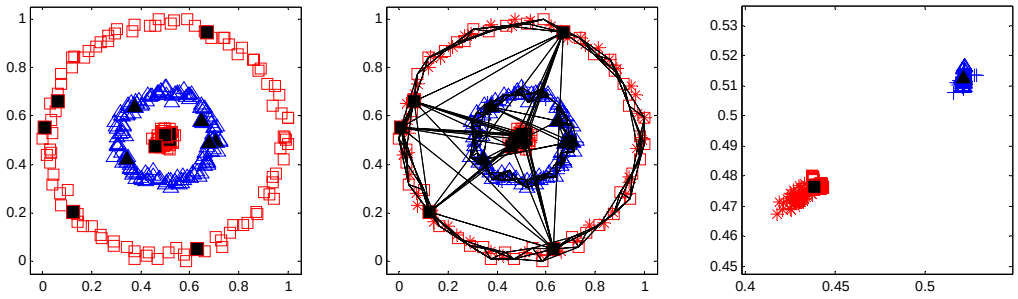

Fig. 3 Left: the original data 3Circles with five labeled points for each circle; Middle: the in-sample and out-of-sample points generated by SPSub with *s_ratio* = 0.5; Right: the shifted patterns of 3Circles by F-SSPS(0.5) with μ = 0.9999 therein.

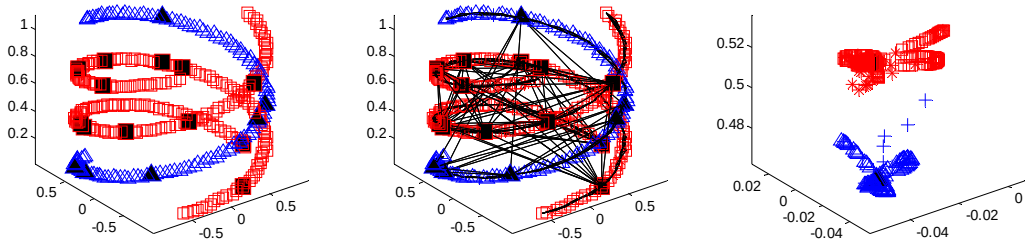

Fig. 4 Left: the original 3Spirals data with five labeled points for each circle; Middle: the in-sample and out-of-sample points generated by SPSub with *s_ratio* = 0.5; Right: the shifted patterns of 3Spirals by F-SSPS(0.5) with μ = 0.9999 therein.

As a comparison in classification performance, we tabulate the classification accuracies on 4Arcs, 3Circles and 3Spirals in Tabs. 4~6, in which F-SSPS(0.5) meaning *s_ratio*=0.5 and PCA(1D) indicating the retained dimensionality # = 1 in PCA.

Tab. 4 Comparative average classification accuracies (%) on 4Arcs

	Labeled # = 5%					Labeled # = 10%				
	NN	FLD	MRank	NB	RBFNet	NN	FLD	MRank	NB	RBFNet
SSPS	**100.00**	**100.00**	**100.00**	94.41	**100.00**	**100.00**	**100.00**	**100.00**	**97.18**	**100.00**
F-SSPS(0.5)	94.37	94.37	**100.00**	88.16	94.37	96.71	94.12	**100.00**	90.79	94.12
SSPS+PCA(1D)	**100.00**	**100.00**	**100.00**	94.02	**100.00**	**100.00**	**100.00**	**100.00**	96.11	**100.00**
Original-data	92.46	55.63	**100.00**	59.35	51.02	96.76	55.88	**100.00**	63.01	50.79
SSKLDA	97.47	98.27	98.23	**97.92**	98.23	93.56	93.06	95.65	94.26	94.21
SSKMMC	90.47	85.86	94.5	93.35	81.29	90.74	90.19	91.48	93.52	87.96
LapLDA	51.2	50.53	62.41	50.8	49.65	52.59	53.66	60.93	52.36	53.06
ISOMAP	76.51	51.33	85.9	69.33	54.43	81.2	50.88	85.51	71.99	52.55

Tab. 5 Comparative average classification accuracies (%) on 3Circles

	Labeled # = 5%					Labeled # = 10%				
	NN	FLD	MRank	NB	RBFNet	NN	FLD	MRank	NB	RBFNet
SSPS	**100.00**	**100.00**	**100.00**	95.2	**100.00**	**100.00**	**100.00**	**100.00**	**98.09**	**100.00**
F-SSPS(0.5)	**100.00**	**100.00**	**100.00**	90.41	**100.00**	**100.00**	**100.00**	**100.00**	96.79	**100.00**
SSPS+PCA(1D)	**100.00**	**100.00**	**100.00**	91.14	**100.00**	**100.00**	**100.00**	**100.00**	92.31	**100.00**
Original-data	81.05	49.44	**100.00**	59.53	48.74	95.06	48.83	**100.00**	68.98	47.47
SSKLDA	99.3	99.53	99.56	**98.57**	99.53	87.1	88.49	91.11	87.78	88.02
SSKMMC	94.68	90.82	95.99	96.52	89.24	96.27	92.9	97.25	97.75	84.23
LapLDA	50.85	50.82	64.01	52.28	50.7	54.88	49.51	63.61	53.77	50.9
ISOMAP	83.8	80	91.43	72.46	66.58	86.76	80.99	91.76	79.04	64.48

Tab. 6 Comparative average classification accuracies (%) on 3Spirals

	Labeled # = 5%					Labeled # = 10%				
	NN	FLD	MRank	NB	RBFNet	NN	FLD	MRank	NB	RBFNet
SSPS	**100.00**	**100.00**	**100.00**	**99.21**	**100.00**	**100.00**	**100.00**	**100.00**	**98.76**	**100.00**
F-SSPS(0.5)	94.03	93.12	95.95	86.46	94.03	94.01	93.91	95.96	89.04	94.03
SSPS+PCA(1D)	**100.00**	**100.00**	**100.00**	97.57	**100.00**	**100.00**	**100.00**	**100.00**	96.69	**100.00**
Original-data	72.92	52.41	95.95	55.95	62.66	83.09	52.68	95.96	52.7	64.3
SSKLDA	91.64	91.83	95.32	93.75	93.87	97.39	97.93	98.12	98.05	98.1
SSKMMC	72.08	74.26	74.49	69.44	71.71	69.37	73.88	73.59	66.84	67.88
LapLDA	57.5	52.04	69.07	52.18	65.35	63.16	49.63	73.85	56.53	66.15
ISOMAP	84.49	75.19	86.67	78.63	80.19	84.16	77.46	86.53	79.85	82.31

From Tabs. 4~6, we can clearly observe that: 1) the best accuracies among SSPS, F-SSPS(0.5) and SSPS+PCA(1D) nearly reach 100% on all 4Arcs, 3Circles and 3Spirals except that in NB; 2) despite different classifier owns different strength, for all 5 classifiers except NB, their best

accuracies among SSPS, F-SSPS(0.5) and SSPS+PCA(1D) uniformly exceeds the corresponding accuracies with respect to SSKLDA, SSKMMC, Original-data and ISOMAP; 3) many accuracies of F-SSPS(0.5) and SSPS+PCA(1D) are comparable to those of SSPS, implying that both schemes of SPSub and SSPS+DR are effective for classification learning here.

4.3 Classification experiments on real-world datasets

We use 4 real-world data Digit1, USPS, Control and Waveform as real-world benchmark data here, whose basic properties can be found in their citations. Before implementing SSKMMC and ISOMAP, we have to firstly estimate the retained dimensionality for data used. Here, based on three estimators: *maximum likelihood estimator*, *eigenvalue-based estimator* and *geodesic minimum spanning tree estimator* [14~16], each retained dimensionality is calculated as the rounding of the average outputs of these estimators. By such way, the retained dimensionality values are respectively estimated as 20 for Digit-1, 11 for USPS, 10 for Control and 11 for Waveform.

We tabulate the classification accuracies in Tabs. 7~10, which correspond to Digit1, USPS and Control in order. Where, F-SSPS(0.8) and F-SSPS(0.6) respectively mean *s_ratio* = 0.8 and 0.6 for SPSub, and PCA(95%) means that 95% variance energy is retained in PCA.

Tab. 7 Comparative average classification accuracies (%) on Digit-1

	Labeled #= 10					Labeled #= 100				
	NN	FLD	MRank	NB	RBFNet	NN	FLD	MRank	NB	RBFNet
SSPS	94.62	92.80	98.39	91.63	94.71	97.75	97.45	**98.40**	97.67	97.75
F-SSPS(0.8)	**95.68**	93.51	98.18	**92.01**	95.91	**97.89**	97.49	98.31	**97.82**	97.90
F-SSPS(0.6)	92.91	88.04	**98.46**	88.89	93.62	97.36	96.89	98.34	97.29	97.38
SSPS+PCA(95%)	94.67	80.88	98.37	84.61	94.68	97.76	97.76	98.38	96.79	97.76
Original-data	76.53	67.68	97.87	68.55	61.26	93.88	86.63	97.92	94.18	75.13
SSKLDA	91.25	**95.01**	93.12	91.55	90.32	93.42	**97.85**	95.03	96.46	95.28
SSKMMC	76.00	53.54	51.12	87.00	**96.32**	77.67	57.02	51.18	91.00	**98.76**
LapLDA	50.51	52.84	60.54	51.25	52.70	51.34	54.47	61.69	52.61	54.20
ISOMAP	76.57	62.61	97.73	67.43	62.34	94.21	90.70	97.79	91.17	73.57

Tab. 8 Comparative average classification accuracies (%) on USPS

	Labeled #= 10					Labeled #= 100				
	NN	FLD	MRank	NB	RBFNet	NN	FLD	MRank	NB	RBFNet
SSPS	81.18	70.31	96.84	83.75	78.80	94.29	**95.55**	97.07	82.82	84.10
F-SSPS(0.8)	81.34	75.50	95.40	**85.45**	78.84	93.72	94.20	95.33	84.98	90.67
F-SSPS(0.6)	81.27	**76.24**	94.33	83.23	78.88	93.70	93.80	95.30	**89.88**	**92.96**
SSPS+PCA(95%)	81.05	72.49	**96.86**	82.84	78.65	94.42	93.93	**97.17**	85.39	83.87
Original-data	80.18	68.76	80.02	80.11	78.80	92.36	84.71	79.99	82.82	78.93
SSKLDA	**82.00**	73.20	90.85	82.13	76.72	**96.04**	89.13	96.22	85.43	91.32
SSKMMC	81.56	61.48	81.56	83.04	78.64	83.30	62.82	83.28	87.00	88.00
LapLDA	53.86	55.94	82.41	80.27	67.48	55.93	59.88	84.87	83.12	68.65
ISOMAP	79.11	69.30	80.02	79.72	**78.99**	92.99	84.43	79.99	88.07	78.40

Tab. 9 Comparative average classification accuracies (%) on Control

	Labeled #= 5%					Labeled #= 10%				
	NN	FLD	MRank	NB	RBFNet	NN	FLD	MRank	NB	RBFNet
SSPS	**97.47**	**97.16**	99.96	96.14	**96.86**	**98.38**	97.08	99.94	97.87	98.33
F-SSPS(0.8)	96.67	95.25	99.37	96.02	96.49	98.18	96.22	99.71	98.02	98.09
F-SSPS(0.6)	96.51	96.01	98.74	95.72	96.39	98.06	96.98	99.27	97.56	98.02
SSPS+PCA(95%)	**97.47**	96.43	**100.00**	**96.40**	96.84	**98.38**	98.36	99.94	**98.72**	**98.36**
Original-data	93.80	69.17	84.99	79.31	67.51	95.74	57.73	84.68	75.51	71.73
SSKLDA	94.00	89.21	**100.00**	95.11	96.02	96.83	**98.54**	**100.00**	96.10	98.02
SSKMMC	95.97	84.59	**100.00**	93.06	91.47	97.31	73.23	99.32	96.74	95.49
LapLDA	55.06	57.78	65.07	56.45	57.57	58.95	63.32	67.55	59.48	62.07
ISOMAP	94.59	71.73	93.36	68.41	65.60	97.05	73.16	93.36	68.84	69.43

Tab. 10 Comparative average classification accuracies (%) on Waveform

	Labeled #= 5%					Labeled #= 10%				
	NN	FLD	MRank	NB	RBFNet	NN	FLD	MRank	NB	RBFNet
SSPS	**90.06**	90.41	91.25	**90.62**	**89.92**	**89.94**	91.22	91.30	**90.69**	**89.90**
F-SSPS(0.8)	89.33	89.42	90.33	89.59	89.21	89.28	91.26	90.40	89.28	89.22
F-SSPS(0.6)	89.29	89.34	90.02	89.57	89.18	89.08	91.25	89.84	89.01	89.07
SSPS+PCA(95%)	90.01	90.74	**91.67**	89.07	89.86	**89.94**	91.43	**91.59**	89.58	**89.90**
Original-data	88.40	88.83	89.79	89.09	75.90	88.23	91.22	89.61	87.80	75.13
SSKLDA	89.02	**90.69**	88.24	86.44	85.06	89.12	**92.08**	89.14	87.20	86.35
SSKMMC	63.63	56.51	53.38	85.72	84.55	62.22	64.07	53.56	89.67	88.52
LapLDA	82.55	79.46	81.76	78.61	79.49	86.91	72.23	74.92	67.62	72.33
ISOMAP	88.73	91.16	90.74	89.61	88.14	89.30	91.46	90.79	89.31	87.05

From Tabs. 7~10, we can get several interesting observations as follows:

- Almost all accuracies of SSPS, SSKLDA, SSKMMC and ISOMAP outperform those of LapLDA on dataset Digit-1, USPS and Control, indicating that the nonlinear preprocessing method seems more appropriate than linear method for these three dataset. At the same time, some accuracies of SSKMMC, LapLDA and ISOMAP are even less than those of Original-data, demonstrating that DR method cannot necessarily lead to a better classification performance on four datasets used here.
- For the most part, on all these datasets, the best one of SSPS, F-SSPS(0.8), F-SSPS(0.6) and SSPS+PCA(95%) get a comparable or even higher accuracy than SSKLDA and SSKMMC by the most classification algorithms used here. This implies that our starting-point or motivation in eq. (2) is reasonable.
- Many accuracies of F-SSPS(0.8) and F-SSPS(0.6) are comparable or even occasionally exceed those of SSPS such as the accuracies on Digit1, demonstrating that the proposed SPSub and F-SSPS can work well in most cases. Meanwhile, the most accuracies of SSPS+PCA(95%) exceed those of SSPS on the dataset USPS, Control and Waveform, which indicates that the proposed "SSPS+DR" scheme is effective and PCA can preserve the main fruit of SSPS.

4.4 Testing on out-of-sample data

In order to evaluate SSPS on out-of-sample data, we randomly split a dataset into in-sample data and out-of-sample data, of which the classification accuracies generated by NN classifier are compared. Under setting of labeling 10% samples, the comparative results as the ratio of out-of-sample data descends are shown in Figs. 5~6.

From Figs. 5~6, we can observe that: 1) the most accuracies of in-sample data are higher than those of corresponding out-of-sample data in general; 2) as the size of out-of-sample data lessen (i.e., the size of in-sample data grows), the difference between two accuracies of out-of-sample and in-sample data becomes smaller; 3) in comparison with Tabs. 4~10, the corresponding accuracies here are less than those of F-SSPS, demonstrating the effectiveness of our subset selection scheme since no SPSub is used for out-of-sample data selection. These above

observations consist with an intuition that a large-size in-sample data is more likely to preserve and recover data's underlying distribution, hence implying that not only the better shifted patterns of in-sample data but also the exacter shifted patterns of out-of-sample data can be captured under a larger-size work-set case.

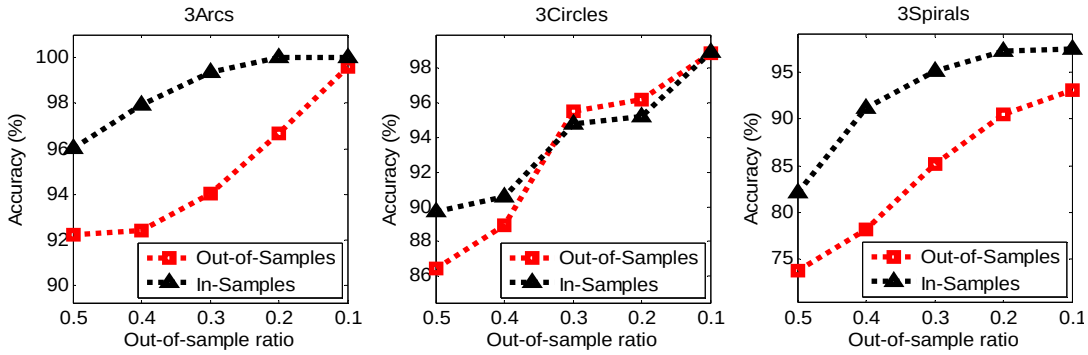

Fig. 5 Comparative accuracies on out-of-sample and in-sample data generated by NN classifier for three synthetic datasets (labeled # = 10% therein).

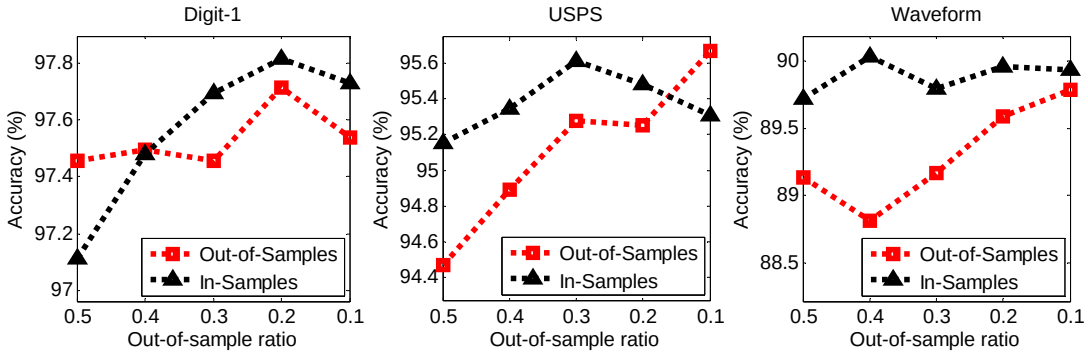

Fig. 6 Comparative accuracies on out-of-sample and in-sample data generated by NN classifier for three real-world datasets (labeled # = 10% therein).

5 Discussions and conclusions

In recent years, the research of semi-supervised DR has received significant attentions [6~9, 22, 23]. Some spectral graph-based DRs such as ISOMAP, LLE and LE are usually plagued in the retained dimensionality estimation problem. Different from DR methods, Our SSPS naturally avoids the problem of retained dimensionality estimation since it shifts pattern directly in original space. Additionally, despite other methods such as semi-supervised metric learning method may get better performance in certain classification applications, a learnt metric cannot always conveniently apply into some classification algorithms such as Fisher linear discriminator (FLD), manifold rank (MRank) [31], naïve Bayes (NB) and radial base function network (RBFNet). In

this sense, a data preprocessor such as SSPS and semi-supervised DR is more extensive than semi-supervised metric learning.

The main advantages of our SPSS can be summarized as: 1) instead of DR, SSPS naturally avoid reducing original data into a lower-dimensional space, which make us to be far away from the problem of estimation of retained-dimensionality; 2) SSPS can be conveniently speeded up by F-SSPS based on the developed SPSub scheme; 3) through its out-of-sample extension, SSPS can work on not only in-sample but also out-of sample data, meaning that a unseen sample can be also shifted easily; 4) SSPS can be naturally and conveniently combined with DR technique, e.g., SSPS+PCA in our experiments.

Nevertheless, we also have to point out that, in analogy to many spectral-graph methods, SSPS also inherits some common problems, e.g., how to determine the best nearest neighbor number k, how to repair an under-sampling density distribution and how to eliminate outliers and noises. If such open problems can be partially resolved by some new techniques in future, they will benefit our SSPS too.

References

[1] J. Ham, D. Lee, S. Mika and B. Scholkopf. A kernel view of the dimensionality reduction of manifolds. Technical Report, Max Planck Institute for Biological Cybernetics,Germany, 2003.

[2] S. Yan, D. Xu, B. Zhang, H. Zhang, Q. Yang and S. Lin. Graph embedding and extension: a general framework for dimensionality reduction, IEEE Transactions on Pattern Analysis and Machine Intelligence, 29: 40-51, 2007.

[3] H.Hotelling. Analysis of a complex of statistical variables into principle components. Journal of Educational Psychology, 24: 417-441, 1933.

[4] R.A. Fisher. The use of multiple measurements in taxonomic problem. Annals of Eugenics, 7, 179-188, 1936.

[5]X. He and P. Niyogi. Locality preserving projections. In: Advances in Neural Information Processing Systems 16 (NIPS'03), MIT Press, Cambridge, MA, 2003.

[6] B. Scholkopf, A.J. Smola and K.-R. M¨uller. Nonlinear component analysis as a kernel eigenvalue problem. Neural Computation, 10(5):1299-1319, 1998.

[7] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. Science, 290: 2323-2326, 2000.

[8] J.B. Tenenbaum, V. de Silva and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. Science, 290: 2319-2323, 2000.

[9] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation, 15(6): 1373-1396, 2003.

[10]L. K. Shi et al. Text Classification Based on Nonlinear Dimensionality Reduction Techniques and Support Vector Machines. In: Proceedings of the third International Conference on Natural Computation (ICNC'07), 674-677, 2007.

[11]C. Ding and X. F. He et al. Adaptive dimension reduction for clustering high dimensional data. In: roceedings

of the second IEEE International Conference on Data Mining (ICDM'02), 147-154, 2002.

[12] F. X. Li, J. Yang and J. Wang. A transductive framework of distance metric learning by spectral dimensionality reduction. In: Proceeding of International Conference on Machine Learning (ICML'07), 513–520, 2007.

[13] L. Torresani and K. Lee. Large margin component analysis. In: Advances in Neural Information Processing Systems 20 (NIPS'07), 1385–1392, 2007.

[14] E. Levina and P.J. Bickel. Maximum likelihood estimation of intrinsic dimension. In: Advances in Neural Information Processing Systems, 17 (NIPS'04), Cambridge, MA, USA, The MIT Press, 2004.

[15] K. Fukunaga and D.R. Olsen. An algorithm for finding intrinsic dimensionality of data. IEEE Transactions on Computers, C-20:176–183, 1971.

[16] L.J.P. van der Maaten. An Introduction to Dimensionality Reduction Using Matlab. Technical Report MICC 07-07, 2007. Maastricht University, Maastricht, The Netherlands.

[17] J. S. Taylor and C. Nello. Kernel methods for pattern analysis, Cambridge University Press, 2004.

[18] Y. Bengio, O. Delalleau and N. L. Roux. Analysis of benchmarks. Semi-Supervised Learning. (Eds.) Chapelle, O., B. Scholkopf, A. Zien, MIT Press, Cambridge, Mass., USA, 2006.

[19] L.J.P. van der Maaten, E.O. Postma and H.J. van den Herik. Dimensionality reduction: a comparative review. Submitted to Neurocognition, 2008.

[20] Y. Bengio, J.-F. Paiement, P. Vincent, O. Delalleau, N. Le Roux and M. Ouimet. Out-of-Sample Extensions for LLE, Isomap, MDS, Eigenmaps, and Spectral Clustering. In: Advances in Neural Information Processing Systems (NIPS'03), 2003.

[21] T. J. Chin, D. Suter. Out-of-Sample Extrapolation of Learned Manifolds. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30(9), 1547-1556, 2008.

[22] D. Cai, X. F. He and J. W. Han. Semi-supervised discriminant analysis. IEEE International Conference on Computer Vision (ICCV'07), Rio de Janeiro, Brazil, Oct., 2007.

[23] D. Q. Zhang, Zh. H. Zhou and S. C. Chen. Semi-supervised dimensionality reduction. Proceedings of the 7th SIAM International Conference on Data Mining (SDM'07), Minneapolis, MN, 629-634, 2007.

[24] O. Delalleau, Y. Bengio and N. Le Roux. Large-Scale Algorithms. (Eds.) Chapelle, O., B. Scholkopf, A. Zien, MIT Press, Cambridge, Mass., USA, 333-341, 2006.

[25] S. Haykin. Neural Networks: A Comprehensive Foundation. Tsinghua University Press, 2001.

[26] Y.Q. Song, F. P Nie, C. S. Zhang and S. M. Xiang. A Unified Framework for Semi-Supervised Dimensionality Reduction. Pattern Recognition, Vol. 41, 2789-2799, 2008.

[27] X. J . Zhu. Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-. Madison, USA, 2005.

[28] M. Breitenbach and G. Grudic. (2005). Clustering through ranking on manifolds, In: proceeding of International Conference on Machine Learning (ICML'05), 2005.

[29] Y. Bengio, O. Delalleau and N. L. Roux. A discussion of Semi-Supervised Learning and transductive. Semi-Supervised Learning. (Eds.) Chapelle, O., B. Scholkopf, A. Zien, MIT Press, Cambridge, Mass., USA, Chapter 1, 2006.

[30] J.H. Chen, J. P. Ye and Q. Li. Integrating Global and Local Structures: A Least Squares Framework for Dimensionality Reduction. In IEEE Conference on Computer Vision and Pattern Recognition, 17-22, 2007.

[31] J. He, M. Li, H. J. Zhang, H. Tong and C. Zhang. Manifold-ranking based image retrieval. In: Proceeding 12th ACM International Conference on Multimedia, 2004.

[32] D. Zhou and B. Schölkopf. Discrete Regularization. Semi-supervised learning. 221-232. (Eds.) Chapelle, O., B. Scholkopf, A. Zien, MIT Press, Cambridge, Mass., USA, 2006.