Semi-Supervised Spectral Mapping for Enhancing Separation between Classes

Weiwei Du
Kyoto Institute of Technology
Matsugasaki, Kyoto 606-8585 Japan
duweiwei@kit.ac.jp

Kiichi Urahama
Kyushu University
Shiobaru 4-9-1, Fukuoka 815-8540 Japan
urahama@design.kyushu-u.ac.jp

Abstract

We present a spectral mapping technique for semi-supervised pattern classification. Importance scores of features are firstly evaluated with a semi-supervised feature selection algorithm by Zhao et al. Training data are then embedded into a low-dimensional space with a spectral mapping derived from the selected and weighted feature vectors with which test data are classified by the nearest neighbor rule. The performance of the proposed pattern classification algorithm is examined with synthetic and real datasets.

1 Introduction

Spectral embedding (SE)[1, 2, 3] is a representative graph spectral method for mapping data nonlinearly into a low-dimensional classification space. While SE is basically unsupervised algorithm similarly to the principal component analysis (PCA), its extension to semi-supervised learning has been proposed[4, 5].

The PCA and SE are a kind of distance metric learning[6]. In the semi-supervised SE, the distance metric is distorted locally on the basis of label information of data. While this local distortion is propagated along graph links to unlabeled data, it is difficult for this method to get globally consistent distance metric, hence its classification performance is insufficient.

Alternative to such local metric modulation, if we incorporate a semi-supervised metric learning technique giving a global distance metric into the SE, we can obtain a new semi-supervised SE algorithm with higher classification rates.

As a such semi-supervised distance metric learning algorithm, a simple semi-supervised feature scoring technique has been proposed and has proven to be effective for image recognition by Zhao et al.[7].

We incorporate this feature scoring technique into the SE algorithm and propose a new semi-supervised pattern classification method. Our contribution in this paper lies in the generalization of the SE algorithm. This generalization enables the SE to map data with enhanced separation between classes. This enhancement in class separation improves the classification rate of the method utilizing the SE mapping. We verify this improvement in the classification rate with some experiments for a synthetic toy dataset and real datasets popularly used for benchmark test of classifiers.

2 Spectral Mapping

Let there be given m training data of feature vectors f_i from which the similarity between data i and j is expressed by

$$s_{ij} = e^{-\alpha\|f_i - f_j\|^2} \quad (i, j = 1, ..., m) \tag{1}$$

2.1 Spectral Embedding

In the spectral embedding method[1, 2], each datum i is mapped to the coordinate x_i given by

$$\begin{array}{ll} \max & \sum_{i=1}^{m}\sum_{j=1}^{m} x_i s_{ij} x_j \\ \text{subj.to} & \sum_{i=1}^{m} d_i x_i^2 = 1 \end{array} \tag{2}$$

where $d_i = \sum_j s_{ij}$. With this mapping, mutually similar data with large s_{ij} are projected close together with near x_i and x_j. This mapping is equivalent to the Laplacian eigenmaps[3]. The most fundamental form of the constraint condition $\sum_i d_i x_i^2 = 1$ in eq.(2) is $\sum_i x_i^2 = 1$ that is the normalization of the norm of $x = [x_1, ..., x_m]^T$[8]. The role of multiplied d_i is to homogenize x_i as is explained below.

Eq.(2) is summarized in the vector form as

$$\begin{array}{ll} \max & \mathbf{x}^T S\mathbf{x} \\ \text{subj.to} & \mathbf{x}^T D\mathbf{x} = 1 \end{array} \tag{3}$$

where $S = [s_{ij}]$ is the similarity matrix and D=diag$(d_1, ..., d_m)$ is the normalization weight matrix. $\mathbf{x}^T S\mathbf{x}$ represents the cohesiveness of data and $\mathbf{x}^T D\mathbf{x}$ denotes their variance. The solution of eq.(3) is the generalized eigenvector of $S\mathbf{x} = \lambda D\mathbf{x}$ of which principal eigenvector with the eigenvalue 1 is constant $[1, ..., 1]/\sqrt{\sum_i d_i}$ due to the homogenization effect of $\mathbf{x}^T D\mathbf{x} = 1$, hence we discard it and use the eigenvectors from the second to the $(p+1)$th when we project the data into a p-dimensional space.

2.2 Generalized Spectral Embedding

This mapping algorithm is unsupervised one where the distance between data is Euclidean as is in eq.(1), i.e. $(f_i - f_j)^T I(f_i - f_j)$ where I is the identity matrix. In the semi-supervised learning, label information is given for some training data, which induces the modulation of distance metric into a generalized quadratic form $(f_i - f_j)^T A(f_i - f_j)$ with a metric matrix A with which the similarity s_{ij} is modified to $\tilde{s}_{ij} = e^{-\alpha(f_i - f_j)^T A(f_i - f_j)}$ which enhances class separability on the basis of label information.

Then a straightforward extension of eq.(2) is modification of both s_{ij} and d_i to $\tilde{s}_{ij}$ and $\tilde{d}_i = \sum_j \tilde{s}_{ij}$. This simple scheme, however, does not work well because the normalization $\sum_i \tilde{d}_i x_i^2 = 1$ uniformizes x_i as was written above that the first eigenvector is constant. This strong equalization effect in eq.(2) cancels the enhanced class separability gained with $\tilde{s}_{ij}$.

Hence we preserve d_i in its unmodified form and propose, in this paper, to modify eq.(3) into a semi-supervised form

$$\begin{array}{ll} \max & \mathbf{x}^T \tilde{S} \mathbf{x} \\ \text{subj.to} & \mathbf{x}^T D \mathbf{x} = 1 \end{array} \tag{4}$$

of which solution is the generalized eigenvector of $\tilde{S}\mathbf{x} = \mu D\mathbf{x}$. Different from eq.(3) of which first eigenvector is constant and is discarded, the first eigenvector of eq.(4) is not constant and contains useful information about data structure. So, we use its eigenvectors from the first to the pth one for embedding data into p-dimensional space. We call this mapping the Generalized Spectral Embedding (GSE). The practical form of $\tilde{S}$ will be shown after the next section.

3 Semi-Supervised Feature Scoring

In this section, we review the semi-supervised feature selection method by Zhao et al.[7]. Assume the training data be partially labeled. We construct the within class similarity $s_{w,ij}$ and between class similarity $s_{b,ij}$ as

$$s_{w,ij} = \begin{cases} \gamma_1 & i,j \in \text{same class} \\ 1 & i \text{ or } j \text{ is unlabeled and} \\ & i \in \text{kNN}(j) \text{ or } j \in \text{kNN}(i) \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

$$s_{b,ij} = \begin{cases} \gamma_2 & i,j \in \text{different classes} \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where $\text{kNN}(i)$ is the set of k nearest neighbors of i. We set $k = 5$ in the following experiments.

We next construct the within class Laplacian matrix L_w and the between class Laplacian matrix L_b as

$$\begin{array}{l} L_w = D_w - S_w \\ L_b = D_b - S_b \end{array} \tag{7}$$

where D_w=diag$(d_{w,1}, ..., d_{w,m})$, $d_{w,i} = \sum_j s_{w,ij}$ and D_b=diag$(d_{b,1}, ..., d_{b,m})$, $d_{b,i} = \sum_j s_{b,ij}$.

We finally compute the importance score of each feature. Let the feature be an n-dimensional vector $f_i = [f_{1i}, ..., f_{ni}]^T$. We define the vector of the r-th feature as $g_r = [f_{r1}, ..., f_{rm}]^T$ from which the score is computed by

$$L_r = \frac{g_r^T L_b g_r}{g_r^T L_w g_r} \quad (r = 1, ..., n) \tag{8}$$

which is large for an important feature similarly to Fisher's discriminant criterion. Zhao et al.[7] select the features with L_r greater than a threshold and called this technique the Locality Sensitive Discriminant Feature (LSDF).

4 Semi-Supervised Pattern Classification

We incorporate this LSDF into the GSE in section 2.2. Since the LSDF gives the score of each feature, we restrict the metric matrix A diagonal and set it as $A = L^2$ where $L = \text{diag}(L_1, ..., L_n)$ with the LSDF score L_r, that is, we modify the similarity in eq.(1) to

$$\tilde{s}_{ij} = e^{-\alpha(f_i - f_j)^T L^2 (f_i - f_j)} \tag{9}$$

where we set L_r below a threshold to 0.

As was written in section 2.2, we construct the similarity matrix $\tilde{S}$ in eq.(4) from these modified $\tilde{s}_{ij}$, while maintaining the matrix D in the original form calculated from the unmodulated similarity in eq.(1). As was explained in section 2.2, the aim of this modification of the SE lies in the relaxation of the too strong homogenization effect of the normalization in the SE. The weighting with L in the constraint condition $\mathbf{x}^T D\mathbf{x} = 1$ makes the mapping of data globally uniform and brings data close even in different classes. Hence we uniformize L in the constraint $\mathbf{x}^T D\mathbf{x} = 1$ into the identity matrix which relaxes this homogenization effect on x and regains the enhancement of separation of classes acquired with modified $\tilde{s}_{ij}$.

We map the training data with this GSE into the $(c-1)$-dimensional classification space where c is the number of classes.

Our proposed technique is summarized as

Step 1: We compute the original similarity $s_{ij} = e^{-\alpha_2 \|f_i - f_j\|^2}$ and construct $D = \text{diag}(d_1, ..., d_m), d_i = \sum_j s_{ij}$.

Step 2: We compute the feature score L_r with the

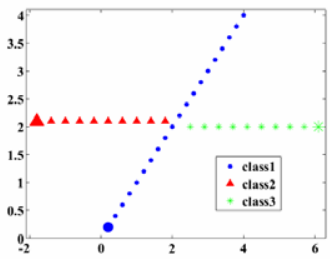
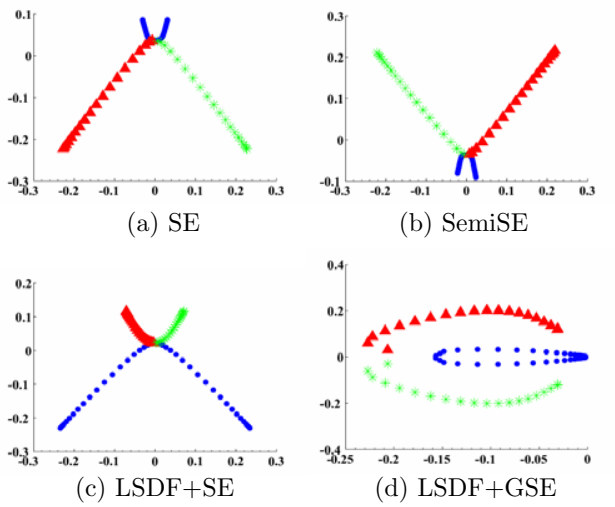


Figure 1: Synthetic data.



(a) SE (b) SemiSE (c) LSDF+SE (d) LSDF+GSE

Figure 2: Mapped data.

Table 1: Error rates for test data.

	test error(%)
SE	18.75
semiSE	17.50
LSDF	33.75
LSDF+SE	33.75
LSDF+GSE	0

LSDF in section 3 and construct $L = \mathrm{diag}(L_1, ..., L_n)$.
Step 3: We compute the modified similarity $\tilde{s}_{ij} = e^{-\alpha_1 (f_i - f_j)^T L^2 (f_i - f_j)}$ and construct $\tilde{S} = [\tilde{s}_{ij}]$.
Step 4: We execute the GSE in section 2.2 and compute the eigenvectors from the first to $(c-1)$th one.
Step 5: We map every training datum into $(c-1)$-dimensional space and we label all the unlabeled data by the nearest neighbor rule with the weighted distance $(f_i - f_j)^T L^2 (f_i - f_j)$ to labeled data.

This finishes the learning phase where all training data are labeled. In a test phase, we classify test data by the nearest neighbor rule with the weighted distance between test data and all the training data.

5 Experiments

We compare the performance of the proposed method LSDF+GSE with SE[1, 2], semi-supervised SE (SemiSE)[5], LSDF[7] and LSDF+SE. In each method, we adjust their parameters to the value with their best performance.

5.1 Synthetic Data

We firstly experiment with the data in Fig.1 which includes 3 classes. Data are arranged on three straight lines. Two horizontal lines are composed of 40 data points and the central inclined line includes 80 points. At each line, data are separated into the training and test data interleaving one by one. Only sampled training data are plotted in Fig.1 where the large marks at the left, right and bottom ends are labeled data which exist only one in each class.

Mapped training data are shown in Fig.2 where ▲ and ∗ are two horizontal line classes in Fig.1 and ● is the center slant line class. The proposed method succeeds to separate three classes while they are glued together in other methods. We verify with this figure that the GSE in section 2.2 is efficient for enhancing the separation of classes and hence improve the classification rate. The error rates for test data are shown in table 1 where our proposed method (LSDF+GSE) can classify test data perfectly. Note that no improvement is gained by the combination of LSDF and SE. Thus the proposed generalization for the SE is essential for the classifier.

Table 2: Data configuration.

dataset	dim.	class	data	labeled	test
iris	4	3	150	9	60
liver	6	2	345	6	173
iono.	34	2	351	6	176
vote	16	2	435	6	218
crx	15	2	690	6	345

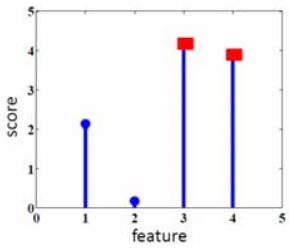


Figure 3: LSDF score for iris dataset.

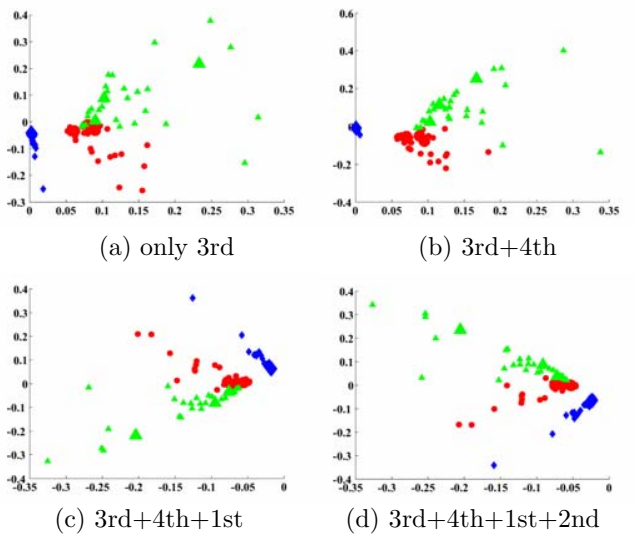
5.2 Real Data

We next experiment with five dataset: iris, liver, ionosphere, vote and crx in the UCI benchmark data[6] popularly used for testing the performance of classifiers. Their data configuration is shown in table 2.

5.2.1 Feature Score

We firstly examine the feature score in the LSDF for the iris dataset which includes four features: sepal length, sepal width, petal length and petal width. Their LSDF scores are 2.13, 0.18, 4.16 and 3.90 as is shown in Fig.3 where ■ marks on the third (petal length) and the fourth feature (petal width) denotes that the classification rate is highest when we select these two features, i.e. set L_1 and L_2 to zero.

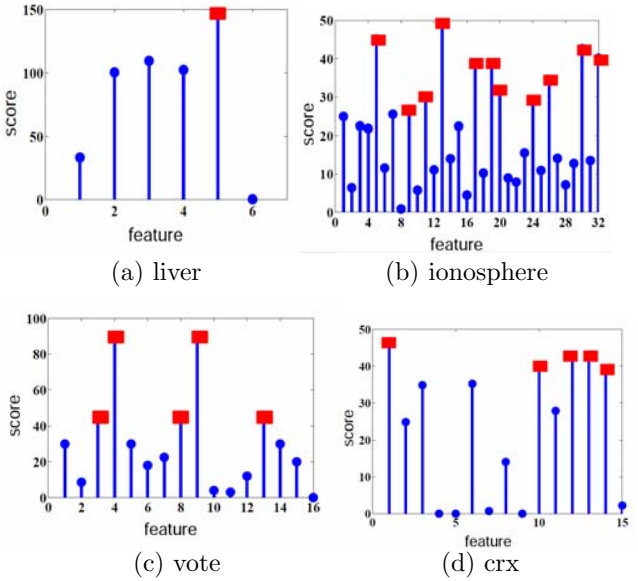
Mapped training data are shown in Fig.4 where four cases of combination of features are examined in the order of value of L_r: (1) only 3rd feature, (2) 3rd and 4th features, (3) 3rd+4th+1st features and (4) all 3rd+4th+1st+2nd features. In Fig.4, labeled data are shown with large marks. Class separation is largest in



(a) only 3rd (b) 3rd+4th

(c) 3rd+4th+1st (d) 3rd+4th+1st+2nd

Figure 4: Mapped iris data.



(a) liver (b) ionosphere

(c) vote (d) crx

Figure 5: LSDF scores.

Table 3: Error rates for each combination of features.

combination	test error(%)
only 3rd	5.00
3rd+4th	1.67
3rd+4th+1st	8.33
3rd+4th+1st+2nd	11.67

Table 4: Error rates for UCI benchmark datasets.

test error	iris	liver	iono.	vote	crx
SE	3.33	49.71	36.36	15.14	32.75
SemiSE	3.33	49.13	32.95	15.14	32.46
LSDF	3.33	39.31	18.75	14.22	33.91
LSDF+SE	3.33	37.57	32.95	14.22	33.62
LSDF+GSE	1.67	35.84	16.48	11.93	27.83

Fig.4(b), i.e. selection of the 3rd and the 4th features is the best.

The error rates of test data are shown in table 3 for these feature combinations. Coincident with the result of Fig.4, the classification rate is highest when the 3rd and the 4th features are selected. This superiority of the 3rd and the 4th features is the well known fact for the iris dataset.

The LSDF scores in other four datasets are shown in Fig.5 where ■ marks denote the best selection of features similarly to Fig.3.

5.2.2 Classification Rates

The error rates of five algorithms for these iris, liver, ionosphere, vote and crx datasets are shown in table 4. We use the selected features marked with ■ in Fig.3 and Fig.5. The classification rate of the proposed method (LSDF+GSE) is highest among these methods.

6 Conclusion

We have presented a semi-supervised spectral mapping method where the semi-supervised feature scoring technique by Zhao et al. is incorporated into the spectral embedding algorithm. We have extended the spectral mapping algorithm to a generalized form and have shown that this generalization is effective for improving the classification rate of the proposed semi-supervised pattern classifier. Theoretical elaboration of the proposed method is a subject of future researches.

References

[1] Y. Koren: "On spectral graph drawing," *Proc. COCOON*, pp.496-508, 2003.

[2] K. Inoue and K. Urahama: "Extraction of arbitrarily shaped clusters by multivariate mapping method," *Trans. IEICE Inf. & Syst.*, vol.J84-D-II, no. 2, pp.229-237 (in Japanese), 2001.

[3] M. Belkin and P. Niyogi: "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Comp.*, vol.15, no. 6, pp.1373-1396, 2003.

[4] S. D. Kamvar, D. Klein and C. D. Manning: "Spectral learning," *Proc. IJCAI*, pp.561-566, 2003.

[5] W. Du and K. Urahama: "Semi-supervised pattern classification utilizing fuzzy clustering and nonlinear mapping of data," *J. Adv. Comput. Intell. Inform.*, vol.11, no.9, pp.1159-1164, 2007.

[6] L. Yang: "Distance metric learning: a comprehensive survey," *Tech. Rep., Michigan State Univ.*, http://www.cs.cmu.edu/~liuy/frame-survey-v2.pdf, 2006.

[7] J. Zhao, K Lu and X. He: "Locality sensitive semi-supervised feature selection," *Neurocomputing*, vol.71, no.10-12, pp.1842-1849, 2008.

[8] W. Du and K. Urahama: "Unsupervised and semi-supervised graph-spectral algorithms for robust extraction of arbitrarily shaped fuzzy clusters," *J. Adv. Comput. Intell. Inform.*, vol.11, no.6, pp.554-560, 2007.