Optimal Dimensionality of Metric Space for Classification

Wei Zhang 031021073@FUDAN.EDU.CN
Xiangyang Xue XYXUE@FUDAN.EDU.CN
Zichen Sun ZCSUN@FUDAN.EDU.CN
Yue-Fei Guo YFGUO@FUDAN.EDU.CN
Hong Lu HONGLU@FUDAN.EDU.CN
Department of Computer Science and Engineering, Fudan University, Shanghai 200433, P. R. China

Abstract

In many real-world applications, Euclidean distance in the original space is not good due to the curse of dimensionality. In this paper, we propose a new method, called Discriminant Neighborhood Embedding (DNE), to learn an appropriate metric space for classification given finite training samples. We define a discriminant adjacent matrix in favor of classification task, i.e., neighboring samples in the same class are squeezed but those in different classes are separated as far as possible. The optimal dimensionality of the metric space can be estimated by spectral analysis in the proposed method, which is of great significance for high-dimensional patterns. Experiments with various datasets demonstrate the effectiveness of our method.

1. Introduction

Learning an appropriate metric space plays an important role in the field of machine learning. Given finite samples, Euclidean distance in the input space is not good in practical applications due to the curse of high-dimensionality. Recently, many techniques have been introduced to learn a more appropriate metric space. DANN (Discriminant Adaptive Nearest Neighbor) (Hastie & Tibshirani, 1996) learns a locally adaptive metric which approximates the weighted Chi-squared distance to alleviate the curse of dimensionality. RCA (Relevant Component Analysis)(Bar-Hillel et al., 2003) divides the data set into several 'chunklets' and finds a metric by using side-information in the form of equivalence relations. NCA (Neighborhood Components Analysis)(Goldberger et al., 2004) learns a distance measure by maximizing a stochastic variant of the leave-one-out *knn* score on the training set using gradient descent. Xing et al. (2002) and Weinberger et al. (2005) learn distance metrics through semidefinite programming. Furthermore, to reduce computation complexity, some of the above algorithms force the learned distance metric to be low rank by selecting a lower target dimensionality of the new space, such as RCA and NCA; however, the target dimensionality is not determined but selected at random for the purpose of dimensionality reduction.

Actually, learning an appropriate distance metric is equivalent to looking for a good transformation which transforms the input data into another better representation. Hence, those traditional methods on feature extraction can be viewed as metric learning algorithms as well, such as PCA (Principal Component Analysis) and LDA (Linear Discriminant Analysis). PCA is mainly for representation and LDA for classification (Fukunaga, 1990). However, the number of features extracted by LDA is at most $c - 1$ (c is the number of classes), which is suboptimal for classification in Bayes sense unless a posteriori probability function is selected (Fukunaga, 1990). Qiu and Wu (2005) provide an approach to improve LDA, but the target dimensionality of the new space for classification is not clear, and should be tuned in an empirical way.

Another important family of algorithms that learn metrics are the ones that assume all data points reside on an intrinsic manifold and find its embedding in a low-dimensional space, such as Isomap (Tenenbaum et al., 2000), LLE (Locally Linear Embedding) (Roweis & Saul, 2000), Laplacian Eigenmap (Belkin & Niyogi, 2001), and LPP (Locality Preserving Projections) (He et al., 2005). In favor of classification task, LDE (Local Discriminant Embedding) (Chen et al., 2005), MFA (Marginal Fisher Analysis) (Yan et al.,

Appearing in *Proceedings of the 24th International Conference on Machine Learning*, Corvallis, OR, 2007. Copyright 2007 by the author(s)/owner(s).

2005), LFDA (Local Fisher Discriminant Analysis) (Sugiyama, 2006), and SNLE (Supervised Nonlinear Local Embedding) (Cheng et al., 2004) incorporate the class information into the underlying manifold structures. Again, the target dimensionality of the new space for classification should be tuned through experiments in these approaches.

In this paper, we propose a new method, called Discriminant Neighborhood Embedding (DNE), to find a low dimensional embedding for classification given finite labeled samples. We first define a discriminant adjacent matrix in favor of classification task, i.e., neighboring samples in the same class are squeezed but those in different classes are separated as far as possible. In the proposed method, the optimal dimensionality of the metric space can be estimated by spectral analysis, which is of great significance for high dimensional patterns.

The organization of this paper is as follows: In Section 2, we propose the DNE algorithm and formulate the optimal dimensionality of the embedding for classification by spectral analysis. Experimental results on UCI repository and UMIST face database are shown in Section 3. Finally, we give the conclusions and suggestions for future work in Section 4.

2. Discriminant Neighborhood Embedding

2.1. Basic Ideas

Suppose N multi-class data points $\{\mathbf{x}_i \in \Re^D\}_{i=1}^N$ are sampled from the underlying manifold embedded in the ambient space. We seek an embedding characterized by intra-class compactness and extra-class separability. One neighbor for a point is referred to as intra-class neighbor provided that they belong to the same class; and extra-class neighbor otherwise. Assume that there is an interaction between each pair of neighboring data points in the ambient space, which can be distinguished as intra-class attraction or extra-class repulsion. With the imaginary local forces of attraction or repulsion exerted on each point, all data points in the original space are supposed to be pulled or pushed by the discriminant neighborhoods, and tend to move until the optimal state for classification is achieved.

Multi-class data points and the interactions between discriminant neighbors can naturally be characterized by one graph G, the nodes of which represent data points. An edge is put between nodes i and j if $\mathbf{x}_i$ is among the set of (intra- or extra-class) neighbors of $\mathbf{x}_j$ or $\mathbf{x}_j$ is among the set of neighbors of $\mathbf{x}_i$. To distinguish the local intra-class attraction and extra-class repulsion between neighboring points, each edge is assigned $+1$ or -1. Let $Neig^I(i)$ and $Neig^E(i)$ denote the set of intra- and extra-class neighbors of $\mathbf{x}_i$ respectively, then the discriminant adjacent matrix F of graph G which models the underlying supervised manifold structure is as follows:

$$F_{ij} = \begin{cases} +1 & (\mathbf{x}_i \in Neig^I(j) \vee \mathbf{x}_j \in Neig^I(i)) \\ -1 & (\mathbf{x}_i \in Neig^E(j) \vee \mathbf{x}_j \in Neig^E(i)) \\ 0 & otherwise \end{cases} \tag{1}$$

For simplicity, $Neig^I(i)$ and $Neig^E(i)$ can be assigned by choosing the k nearest intra- and extra- class neighbors, respectively. We implement the motion of all points by learning a linear transformation of the input space such that in the transformed space, intra-class compactness and extra-class separability are achieved simultaneously. We denote the transformation by a matrix P, measure intra-class compactness by:

$$\begin{gathered}\Delta(P) = \textstyle\sum_{i,j} \left\| P^\top \mathbf{x}_i - P^\top \mathbf{x}_j \right\|^2, \\ (\mathbf{x}_i \in Neig^I(j) \vee \mathbf{x}_j \in Neig^I(i))\end{gathered} \tag{2}$$

and similarly measure extra-class separability by:

$$\begin{gathered}\delta(P) = \textstyle\sum_{i,j} \left\| P^\top \mathbf{x}_i - P^\top \mathbf{x}_j \right\|^2, \\ (\mathbf{x}_i \in Neig^E(j) \vee \mathbf{x}_j \in Neig^E(i))\end{gathered} \tag{3}$$

then one reasonable criterion for 'good' motion is to minimize:

$$\Phi(P) = \Delta(P) - \delta(P) \tag{4}$$

Minimizing criterion (4) is an attempt to minimize the total distance among intra-class neighbors and to maximize that between extra-class neighbors simultaneously. Using (1), the criterion (4) can also be expressed as:

$$\Phi(P) = \sum_{i,j} \left\| P^\top \mathbf{x}_i - P^\top \mathbf{x}_j \right\|^2 F_{ij} \tag{5}$$

2.2. Optimization

Following some simple algebraic steps, we get that

$$\begin{aligned} \Phi(P) &= \textstyle\sum_{i,j} \left\| P^\top \mathbf{x}_i - P^\top \mathbf{x}_j \right\|^2 F_{ij} \\ &= 2\textstyle\sum_{i,j} \left(\mathbf{x}_i^\top P P^\top \mathbf{x}_i - \mathbf{x}_i^\top P P^\top \mathbf{x}_j \right) F_{ij} \\ &= 2\textstyle\sum_{i,j} \mathrm{tr} \left(\left(P^\top \mathbf{x}_i \mathbf{x}_i^\top P - P^\top \mathbf{x}_j \mathbf{x}_i^\top P \right) F_{ij} \right) \\ &= 2\mathrm{tr} \left(\textstyle\sum_{i,j} \left(P^\top \mathbf{x}_i F_{ij} \mathbf{x}_i^\top P - P^\top \mathbf{x}_j F_{ij} \mathbf{x}_i^\top P \right) \right) \\ &= 2\mathrm{tr} \left(P^\top X S X^\top P - P^\top X F X^\top P \right) \\ &= 2\mathrm{tr} \left(P^\top X (S - F) X^\top P \right) \end{aligned} \tag{6}$$

where tr(·) denotes the trace of matrix and $X = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]$. S is a diagonal matrix whose entries are column (or row) sums of the discriminant adjacent matrix F which is symmetric. Since the quadratic form $\Phi(P)$ is not positive semi-definite, the symmetric matrix $(S - F)$ and $X(S - F)X^T$ are not either, which substantially differs from the Laplacian matrix in Laplacian Eigenmaps (Belkin & Niyogi, 2001) and LPP (He et al., 2005).

Then the optimal transformation matrix P can be obtained by:

$$P_{optimal} = \arg\min_{P} \operatorname{tr}\left(P^\top X(S-F)X^\top P\right) \tag{7}$$

Denote $P = [P_1, P_2, \dots, P_m], (m \le D)$. The column vectors of the matrix P form a basis spanning the new space, and we constrain that they are orthonormal. Then,we reformulate the right side of (7) as:

$$\begin{array}{ll} \min & \sum_{i=1}^{m} P_i^\top X(S-F)X^\top P_i \\ \text{s.t.} & P_i^\top P_i = 1, P_i^\top P_j = 0, (i \ne j) \end{array} \tag{8}$$

Lemma 1 *Suppose that P_i is one normalized column vector, i.e. $P_i^\top P_i = 1$, and that λ_1 is the minimal eigenvalue of the symmetric matrix A, then $\lambda_1 = \min_{P_i^\top P_i = 1}\left(P_i^\top A P_i\right)$, and the minimum is achieved when P_i is the normalized eigenvector of A corresponding to λ_1 .*

Lemma 2 *Suppose that $\lambda_1 \le \dots \le \lambda_{i-1} \le \lambda_i \le \dots \le \lambda_n$ are the eigenvalues of the symmetric matrix A, $P_1, \dots, P_{i-1}$ are the orthonormal eigenvectors corresponding to $\lambda_1, \dots, \lambda_{i-1}$, respectively, and that P_i is one normalized column vector, then $\lambda_i = \min_{P_i^\top P_i = 1, P_i^\top P_j = 0, (j=1,\dots,i-1)}\left(P_i^\top A P_i\right)$, and the minimum is achieved when P_i is the normalized eigenvector of A corresponding to λ_i .*

The proofs of **Lemma 1** and **Lemma 2** are given in (Wang & Shi, 1988).

Suppose that $\lambda_1 \le \dots \le \lambda_D$ are all the eigenvalues of the symmetric matrix $X(S - F)X^\top$. If for each i we ensure that $P_i^\top X(S-F)X^\top P_i$ is as small as possible, then the sum in (8) is as small as possible. Without loss of generality, we begin by minimizing the first item of (8). According to **Lemma 1**, the minimum is achieved when P_1 is the normalized eigenvector corresponding to the smallest eigenvalue λ_1. Then by **Lemma 2**, we minimize the next item by assigning P_2 the normalized eigenvector corresponding to the second smallest eigenvalue λ_2, and so on. Finally, we get:

$$\sum_{i=1}^{m} P_i^\top X(S-F)X^\top P_i = \sum_{i=1}^{m} \lambda_i \tag{9}$$

Since the symmetric matrix $X(S - F)X^\top$ is not positive semi-definite, the eigenvalues of $X(S-F)X^\top$ may be positive, negative or zero: $\lambda_1 \le \dots \le \lambda_d < 0 \le \lambda_{d+1} \le \dots$, where d is the number of the negative eigenvalues. To minimize (9), we select these d negative eigenvalues such that:

$$\sum_{i=1}^{m} \lambda_i = \sum_{i=1}^{d} \lambda_i \tag{10}$$

and the transformation matrix P is just constituted by the eigenvectors of $X(S - F)X^\top$ corresponding to its first d negative eigenvalues.

2.3. The Learned Distance Metric

Once the $D \times d$ transformation matrix P has been learned, the representation of any new sample is obtained by:

$$\mathbf{y}_{new} = P^\top \mathbf{x}_{new} \tag{11}$$

where $\mathbf{x}_{new} \in \Re^D$ and $\mathbf{y}_{new} \in \Re^d$.

In the transformed space, the squared distance between any pair of data points is as follows:

$$\begin{aligned} \operatorname{dist}(\mathbf{x}_i, \mathbf{x}_j) &= \left\|P^\top \mathbf{x}_i - P^\top \mathbf{x}_j\right\|^2 \\ &= (\mathbf{x}_i - \mathbf{x}_j)^\top P P^\top (\mathbf{x}_i - \mathbf{x}_j) \\ &= (\mathbf{x}_i - \mathbf{x}_j)^\top M (\mathbf{x}_i - \mathbf{x}_j) \\ &= \|\mathbf{x}_i - \mathbf{x}_j\|_M^2 \end{aligned} \tag{12}$$

where the matrix $M = PP^\top$ just parameterizes the Mahalanobis distance metric, which is of low rank if $d < D$. The optimal dimensionality of the metric space, i.e., the rank of the learned metric, can be estimated by spectral analysis in our method. Intra-class compactness and extra-class separability are achieved simultaneously in the transformed space and therefore such learned distance metric is appropriate for classification.

Actually, when data points are projected to one specific eigenvector P_i corresponding to eigenvalue λ_i, the criterion Φ in (4) becomes as follows:

$$\begin{aligned} \Phi(P_i) &= \Delta(P_i) - \delta(P_i) \\ &= 2\left(P_i^\top X(S-F)X^\top P_i\right) \\ &= 2P_i^\top \lambda_i P_i \\ &= 2\lambda_i \end{aligned} \tag{13}$$

Intuitively, the eigenvalue λ_i measures the total distance between neighboring samples of the same class

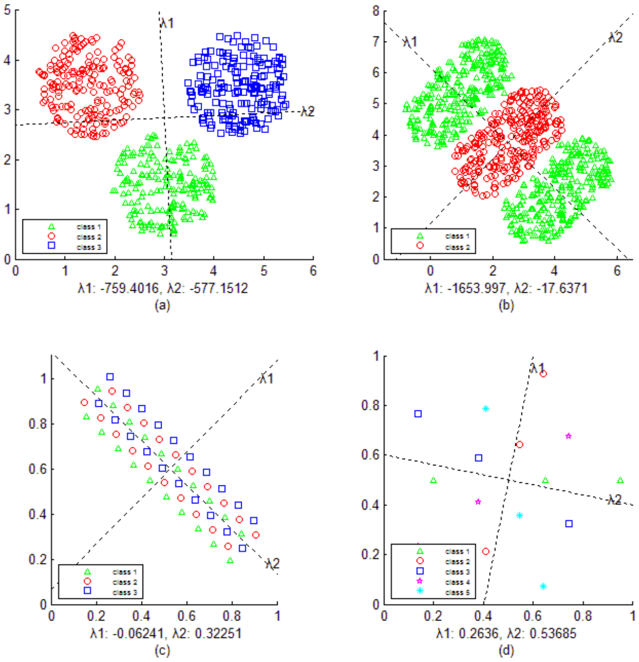

Figure 1. a)Three classes of well clustered data. Both eigenvalues are negative. b)Two classes of data with multimodal distribution. The magnitude of one negative eigenvalue is much larger than the other. c)Three classes of data. One eigenvalue is negative and the other is positive. d)Five classes of data which are not separable. Both eigenvalues are positive. (For all the cases, k=1.)

minus that of different classes when data points are projected to the corresponding eigenvector P_i. If $\lambda_i > 0$, the total distance between intra-class neighbors is larger than that between extra-class neighbors along the direction P_i, and samples tend to be misclassified. So, those eigenvectors corresponding to positive eigenvalues are discarded. Otherwise, if $\lambda_i < 0$, samples tend to be correctly classified. Among all the d negative eigenvalues $\lambda_1 \leq \ldots \leq \lambda_d < 0$, the absolute values of some λ_i's may be much larger than those of others, e.g., $|\lambda_1| \gg |\lambda_d|$. The larger the absolute value of λ_i (< 0) is, the more intra-class compact and more extra-class separable the projected data are. Therefore those of which absolute values are small enough can be ignored. We can then choose t negative eigenvalues with the largest absolute values such that

$$\sum_{i=1}^{t} |\lambda_i| \geq \theta \sum_{i=1}^{d} |\lambda_i| \qquad (14)$$

where θ specifies the proportion that we wish to retain, and for a specified θ, we calculate a minimal t (for instance θ=0.96 in our experiments). By selecting such leading eigenvectors with respect to t ($< d < D$) dominant negative eigenvalues in practice, a subspace of even lower dimensionality is derived.

As an illustration, we show four cases with toy data in Figure 1: a) Three classes of data are well clustered in the input space. Both eigenvalues are negative, and we do not perform dimensionality reduction. b) Two classes of data with multimodal distribution. Although two eigenvalues are both negative, the magnitude of one is much larger than the other; so we can reduce dimensionality by keeping the leading eigenvector with $\lambda_1 = -1653.997$ and discarding the other. c) Three classes of data. We get two directions corresponding to positive and negative eigenvalues respectively. As can be seen, the direction with positive eigenvalue should be removed from the point of view of classification. d) Five classes of data. Both eigenvalues are positive, and we can not perform classification well along each direction. Samples are not separable in the input space or any subspace. To solve this problem, we can first map data points into Hilbert space with *kernel* trick. This is not within the scope of this paper and we would discuss it in our future work. Another extreme case should be pointed out that given infinite samples, total distance between intra-class neighbors is always smaller than that between extra-class neighbors along any direction P_i, so, all the eigenvalues are negative, and dimensionality reduction is not performed accordingly.

3. Experimental Results

In this section, we evaluate the proposed method in comparisons with the state-of-the-art approaches: PCA, LDA, LPP, LDE and NCA. As for LDA, LPP, and LDE, they all include solving a generalized eigenvector problem; to avoid singularities from a small sample size problem encountered in these methods, the techniques suggested by (Belhumeur et al., 1997; He et al., 2005; Chen et al., 2005) are adopted, i.e., PCA is used to pre-process the singular matrix. This problem does not exist in our method since there is no need for matrix inversion. The $k-$nearest neighbors classifier (knn) is employed in our experiments.

We first conduct experiments on three datasets (i.e., *sonar*, *wdbc*, *ionosphere*) from the UCI repository (Murphy & Aha, 1994). For each dataset, we randomly split it into two subsets (30%+70%). First, the 30% subset is chosen for training and the rest 70% for testing. Second, 70% for training and 30% for testing.

All the experiments are repeated 10 times independently and the average results and standard deviations are calculated. Figure 2 shows the training and testing performance. The accuracies on the training set reported are evaluated by leave-one-out. As can be seen, the performance of our method is the same as or better than that of other approaches in most cases.

Experiments are also conducted on the UMIST face database[1](Graham & Allinson, 1998), which contains 575 images of 20 individuals (i.e., 20 classes). Each image is firstly cropped and down-sampled into 56×46 size, and the resulting input vectors are of dimensionality D $= 2576$. For each individual, p $(= 4, 6, 8, 10)$ images are randomly selected for training and the remainder for testing. For each given p, experiments are repeated 10 times independently. Figure 3 shows the average recognition rate with standard deviation over 10 repetitions per p for each method. As can be observed, the distance metric learned by DNE consistently outperforms the others across a range of training sizes.

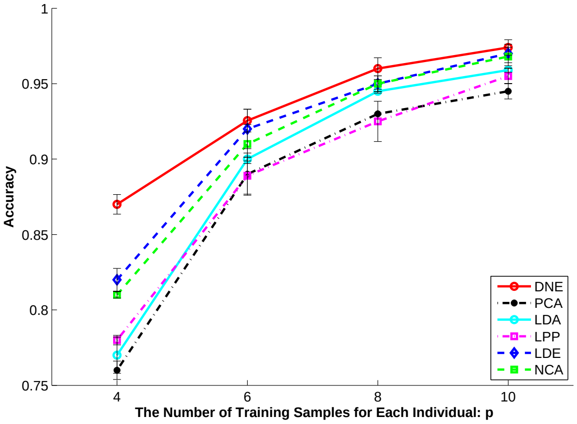

Figure 3. Comparisons of the average accuracies with standard deviations on the UMIST database with $k = 1$.

In the proposed method, the optimal dimensionality is determined by selecting the leading eigenvectors corresponding to negative eigenvalues, which is distinguished from the other approaches. Figure 4, 5, and 6 show the performance of the proposed method across a range of projection dimensions. The upper panels illustrate the classification accuracy (on the training and testing set) as a function of the dimensionality (one run for each dataset). The lower panels show the ascendingly sorted eigenvalue spectra and its corresponding cumulative curve. As can be seen, the classification accuracy increases with projection dimensions added when the corresponding eigenvalues are largely smaller than zero; it achieves its optimum and tends to be stable when eigenvalues are nearly zero; and it even turns to decrease when eigenvalues are positive. In the case that all dimensions are added, the classification accuracy of the learned metric actually yields the same result as that of original Euclidean distance metric. As for the UMIST face database, the dimensionality of the input space is very high ($D = 2576$), and a large number of eigenvalues are zero. During the ranges with zero eigenvalues, the classification accuracy is stable. For the purpose of clarity and saving space, we do not show the range with zero eigenvalues, but just show the ranges with negative and positive eigenvalues in two separate panels per p. For example, the left two columns in Figure 5, i.e., (a) and (b), are corresponding to one run for p=4. Figure 5(a) shows the change of classification accuracy (on the training and testing set) versus the dimensionality at the range with negative eigenvalues, and Figure 5(b) at the range with positive eigenvalues. From Figure 4, 5, and 6, we observe that the trajectory of classification accuracy on the testing set fits with that on the training set in general, and the generalization ability becomes better when the size of training set is large; the shapes of both trajectories are consistent with that of the cumulative eigenspectra curve's *mirror* image. Therefore, it is demonstrated that the optimal subspace for classification is just spanned by such eigenvectors corresponding to the leading negative eigenvalues.

4. Conclusions

In this paper, we propose a new method to find a low-dimensional embedding for classification given finite training samples. Our method is conceptually simple yet effective. The learned metric space achieves intra-class compactness and extra-class separability simultaneously, and the optimal dimensionality can be estimated by spectral analysis. Similar to many manifold learning algorithms, the proposed method considers the local structure which is more important than global structure when *knn* rule is employed. However, it remains unclear how to select the parameter k in a principled manner which defines the locality theoretically. This is the common problem of our method and other related algorithms (LPP, LDE, NCA, etc.), which will be investigated in our future work. Another attempt is to extend DNE with *kernel* trick, which may be useful to the case (See Figure 1(d)) that samples are not separable.

[1] http://images.ee.umist.ac.uk/danny/database.html.

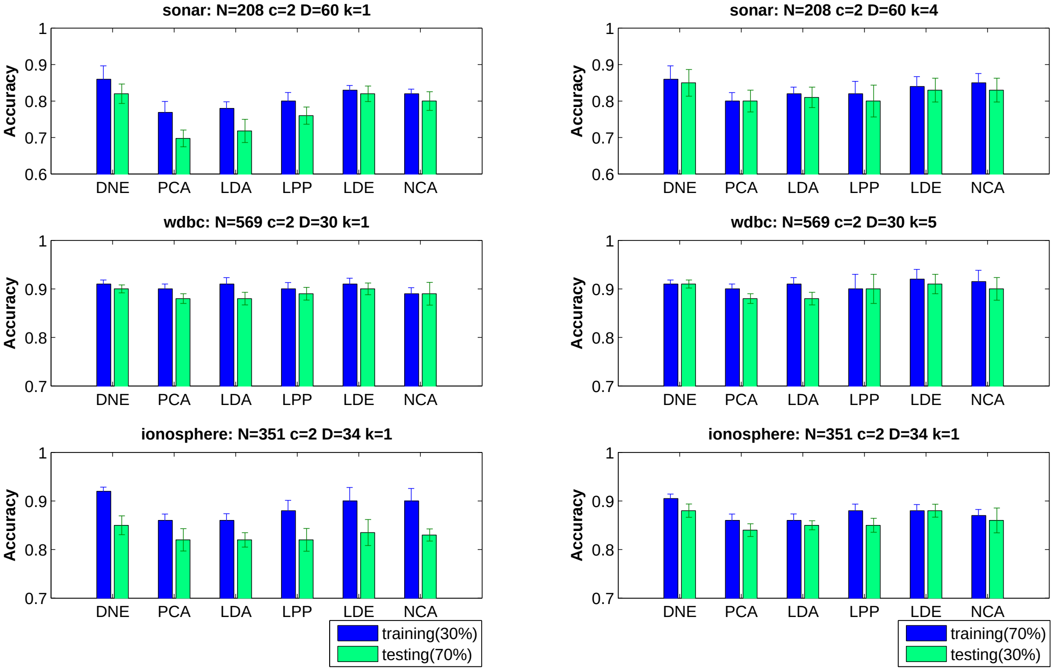

Figure 2. Comparisons of training and testing performances on three UCI datasets. Left: 30% subset is chosen for training and the rest 70% for testing. Right: 70% for training and 30% for testing. Average correct rates and standard deviations are calculated over 10 independent repetitions. In each panel, from left to right six pairs of bars are respectively: 1)DNE, 2)PCA, 3)LDA, 4)LPP, 5)LDE, 6)NCA. Also shown are N-the number of samples, c-the number of classes, D-the dimensionality of input space, k-the parameter selected for *knn*.

Acknowledgments

We would like to thank the anonymous reviewers who gave us useful comments. We would also like to thank Dr. Mingmin Chi for her help. This work was supported in part by Natural Science Foundation of China under contracts 60533100 and 60402007, and Shanghai Municipal R&D Foundation under contracts 05QMH1403, 065115017 and 06DZ15008.

References

Bar-Hillel, A., Hertz, T., Shental, N., & Weinshall, D. (2003). Learning distance functions using equivalence relations. *Proceedings of the 20th International Conference on Machine Learning*.

Belhumeur, P. N., Hespanda, J., & Kiregeman, D. (1997). Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *19(7)*, 711–720.

Belkin, M., & Niyogi, P. (2001). Laplacian eigenmaps and spectral techniques for embedding and clustering. *Advances in Neural Information Processing Systems*. Cambridge, MA.

Chen, H.-T., Chang, H.-W., & Liu, T.-L. (2005). Local discriminant embedding and its variants. *IEEE Conference on Computer Vision and Pattern Recognition*.

Cheng, J., Liu, Q., Lu, H., & Chen, Y.-W. (2004). A supervised nonlinear local embedding for face recognition. *International Conference on Image Processing*.

Fukunaga, K. (1990). *Introduction to statistical pattern recognition*. Boston: Academic Press. 2nd edition, 441-443, 466-467.

Goldberger, J., Roweis, S., Hinton, G., & Salakhutdinov, R. (2004). Neighborhood components analysis. *Advances in Neural Information Processing Systems*. Cambridge, MA.

Graham, D. B., & Allinson, N. M. (1998). Characterizing virtual eigensignatures for general purpose face recognition. *Face Recognition: From Theory to Application*, *163*.

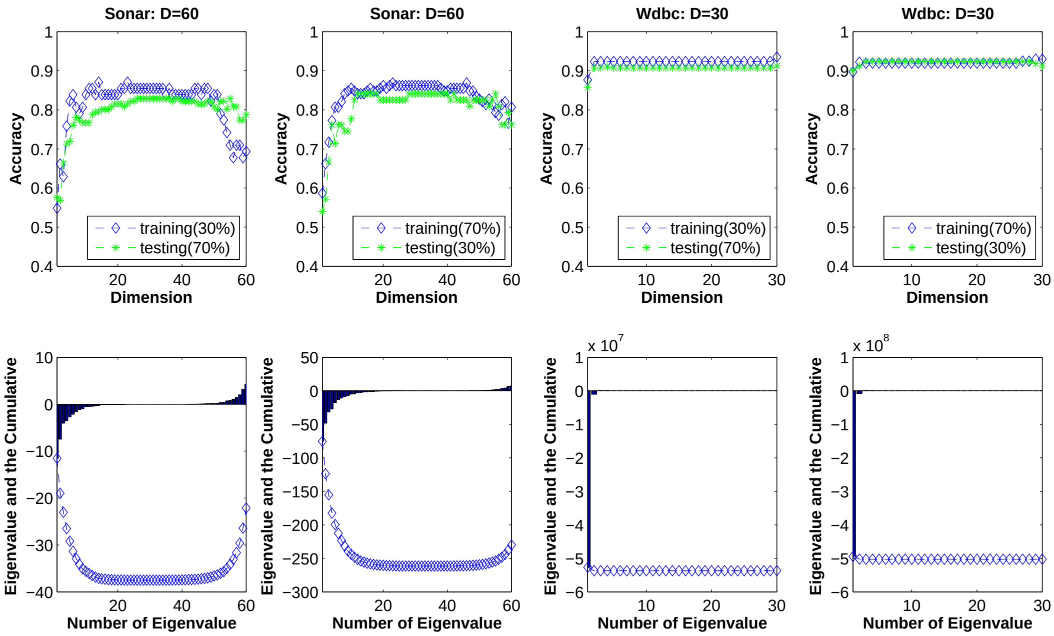

Figure 4. The relation between the performance of DNE across a range of projection dimensions and the corresponding eigenspectra over *sonar* and *wdbc* datasets. Upper panels: The change of classification accuracy versus the dimensionality (each panel involves the performance curve for the training and testing set simultaneously). The dimensionality of the input space, D, is shown as well. Lower panels: The ascendingly sorted eigenvalue spectra and its cumulative curve over each dataset.

Hastie, T., & Tibshirani, R. (1996). Discriminant adaptive nearest neighbor classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *18(6)*, 607–616.

He, X., Yan, S., Hu, Y., Niyogi, P., & Zhang, H. (2005). Face recognition using laplacianfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *27(3)*, 328–340.

Murphy, P. M., & Aha, D. W. (1994). UCI repository of machine learning databases. http://www.ics.uci.edu/∼mlearn/MLRepository.html.

Qiu, X., & Wu, L. (2005). Stepwise nearest neighbor discriminant analysis. *Proceedings of the 19th International Joint Conference on Artificial Intelligence*.

Roweis, S. T., & Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, *290*, 2323–2326.

Sugiyama, M. (2006). Local fisher discriminant analysis for supervised dimensionality reduction. *Proceedings of the 23rd International Conference on Machine Learning*. Pittsburgh, PA.

Tenenbaum, J. B., de Silva, V., & Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science*, *290*, 2319–2323.

Wang, G., & Shi, R. (1988). *Theory of matrix*. Beijing, China: Defense Industry Press, 376-378 (in Chinese).

Weinberger, K. Q., Blitzer, J., & Saul, L. K. (2005). Distance metric learning for large margin nearest neighbor classification. *Advances in Neural Information Processing Systems*. Cambridge, MA.

Xing, E. P., Ng, A. Y., Jordan, M. I., & Russell, S. (2002). Distance metric learning, with application to clustering with side-information. *Advances in Neural Information Processing Systems*. Cambridge, MA.

Yan, S., Xu, D., Zhang, B., & Zhang, H.-J. (2005). Graph embedding: A general framework for dimensionality reduction. *IEEE Conference on Computer Vision and Pattern Recognition*.

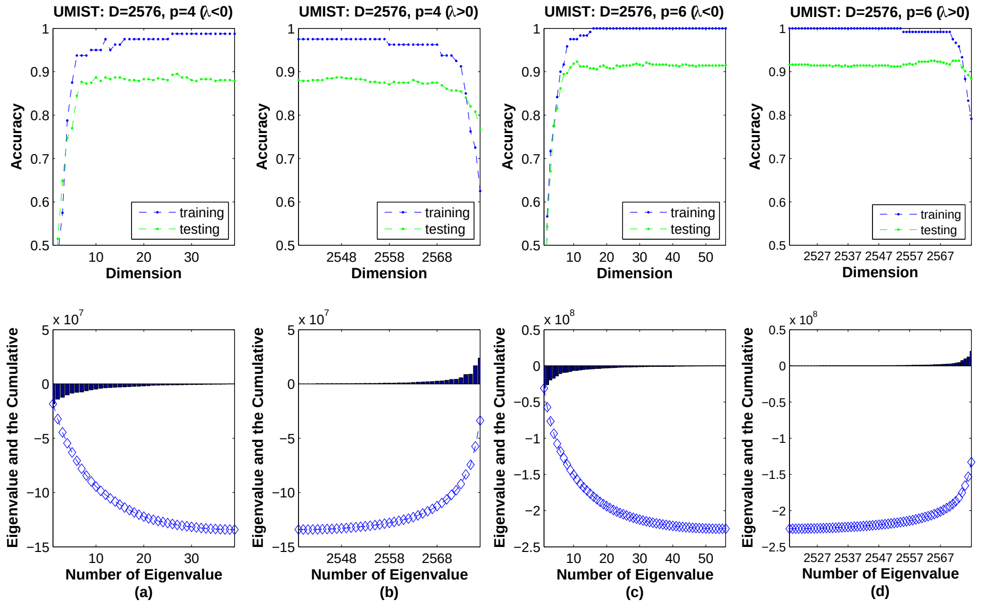

Figure 5. (a)$p = 4$, $\lambda < 0$, (b)$p = 4$, $\lambda > 0$; (c)$p = 6$, $\lambda < 0$, (d)$p = 6$, $\lambda > 0$. Ranges with zero eigenvalues are not shown.

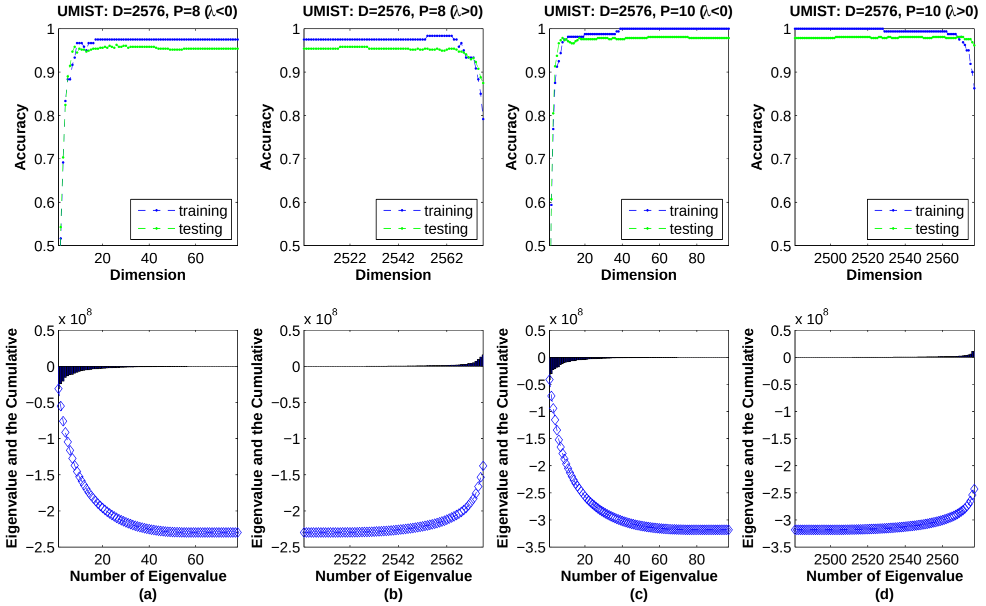

Figure 6. (a)$p = 8$, $\lambda < 0$, (b)$p = 8$, $\lambda > 0$; (c)$p = 10$, $\lambda < 0$, (d)$p = 10$, $\lambda > 0$. Ranges with zero eigenvalues are not shown.

Figure 5 *and* 6. The relation between the performance of DNE across a range of projection dimensions and the corresponding eigenspectra per p(=4,6,8,10) over UMIST dataset. Upper panels of Figure 5 and 6: The change of classification accuracy versus the dimensionality (on the training and testing set). Lower panels: The ascendingly sorted eigenvalue spectra and its cumulative curve. 2576 eigenvalues are derived per run, among which more than 2000 are zero. For the purpose of clarity and saving space, we just show the ranges with negative and positive eigenvalues in two separate panels for each p, and the ranges with zero eigenvalues are not shown.