Jointly Learning Data-Dependent Label and Locality-Preserving Projections

Chang Wang
IBM T. J. Watson Research Lab
19 Skyline Drive
Hawthorne, New York 10532
wangchan@us.ibm.com

Sridhar Mahadevan
Computer Science Department
University of Massachusetts
Amherst, Massachusetts 01003
mahadeva@cs.umass.edu

Abstract

This paper describes a novel framework to jointly learn data-dependent label and locality-preserving projections. Given a set of data instances from multiple classes, the proposed approach can automatically learn which classes are more similar to each other, and construct discriminative features using both labeled and unlabeled data to map similar classes to similar locations in a lower dimensional space. In contrast to linear discriminant analysis (LDA) and its variants, which can only return $c-1$ features for a problem with c classes, the proposed approach can generate d features, where d is bounded only by the number of the input features. We describe and evaluate the new approach both theoretically and experimentally, and compare its performance with other state of the art methods.

1 Introduction

In many machine learning applications involving data from multiple classes, it is highly desirable to map high dimensional data instances to a lower dimensional space, with a constraint that the instances from similar classes will be projected to similar locations in the new space. For example, we might use 4 values to label query-document pairs in a ranking algorithm: 1-"excellent", 2-"good", 3-"fair", 4-"bad". The pairs labeled with "excellent" should be more similar to the instances labeled with "good", compared to the instances labeled with "bad".

There are several challenges to address in this process. Firstly, the relations between different classes are not known in many applications, so we have to learn from the data about which classes are similar to each other and how similar they are. Secondly, the given labeled data is not sufficient in many situations. This could result in overfitting problems, if we learn the mappings solely from the labeled data. Thirdly, although some dimensionality reduction approaches can be applied here to handle the challenges, one limitation of them is that the dimensionality of the resulting data is bounded by the number of classes. One such example is Linear Discriminant Analysis (LDA) [Fukunaga, 1990]. For a data set with c class labels, LDA type approaches can only achieve a $c-1$ dimensional embedding (since the matrix to model the between-class difference only has $c-1$ nontrivial eigenvectors). This means LDA type approaches only yield a 1D embedding for a dataset with two class labels (positive/negative), even when the data is originally defined by several hundreds of features.

In this paper, we propose an approach (called "discriminative projections") that automatically constructs data-dependent projections for both data instances and labels to construct new representations. The novel part of this approach mainly comes from the projection function for the labels. An illustration of this idea is given in Figure 1. Discriminative projections has the goal to automatically learn the relations between different classes, eliminate useless features and improve the speed and performance of classification, clustering, ranking, and multi-task learning algorithms. This approach is designed to handle the situation when the given labeled data is not sufficient. The goal is achieved by exploring the relations between a small amount of labeled data that reflects the class separability and a large amount of unlabeled data that reflects the intrinsic structure of the whole dataset. Our work is related to previous work on regression models, manifold regularization [Belkin, Niyogi, and Sindhwani, 2006], linear discriminant analysis (LDA) [Fukunaga, 1990], Canonical Correlation Analysis (CCA) [Hotelling, 1936] and dimensionality reduction methods such as locality-preserving projections (LPP) [He and Niyogi, 2003]. Discriminative projections has the following advantages: (1) It automatically constructs data-dependent label and locality-preserving projections to help map similar classes to similar locations in the new space. (2) It makes use of both labeled and unlabeled data, and is less prone to overfitting problems. (3) Dimensionality of the resulting data is only bounded by the number of the input features rather than the number of classes. This is particularly useful for applications with a large amount of input features but a small number of classes. (4) The algorithm only requires specifying a single parameter and this paper provides an intuitive way to set its value.

2 Related Work

Our approach learns discriminative projections to map high-dimensional data instances and their corresponding labels to a new space, leveraging the given class label information and the data manifold topology so that instances from similar classes will be mapped to similar locations. Our new approach is related to the ideas of manifold regularization,

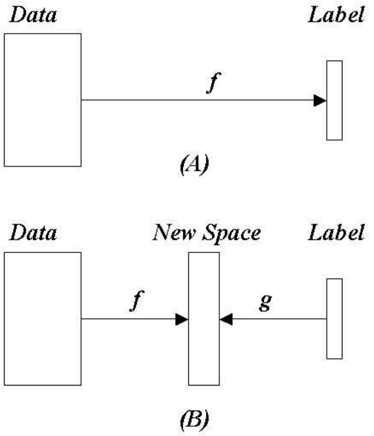

Figure 1: Illustration of regular learning approaches (A), and our approach (B).

LDA, regular dimensionality reduction and Canonical Correlation Analysis.

Linear regression involves estimating a coefficient vector to map the data instances to real-valued outputs (or continuous class labels). For example, given a set of instances $\{x_i\}$ defined in a p dimensional space, a linear regression model computes $\beta_0, \cdots, \beta_p$ such that label y_i can be approximated by

$$\hat{y_i} = \beta_0 + \beta_1 x_i(1) + \cdots + \beta_p x_i(p) \text{ for } i = 1, \ldots, n.$$

The framework of manifold regularization [Belkin, Niyogi, and Sindhwani, 2006] combines the standard loss functions associated with regression or classification with an additional term that preserves the local geometry of the given data manifold (the framework has another term corresponding to an ambient regularizer). One problem solved under this framework can be characterized as follows: given an input dataset $X = (x_1, \cdots, x_m)$ and label information $Y = (y_1, \cdots, y_l)$ $(l \leq m)$, we want to compute a function f that maps x_i to a new space, where $f^T x_i$ matches x_i's label y_i. In addition, we also want f to preserve the neighborhood relationship within dataset X (making use of both labeled and unlabeled data). This problem can be viewed as finding an f that minimizes the cost function:

$$C(f) = \sum_{i \leq l} (f^T x_i - y_i)^2 + \mu \sum_{i,j} (f^T x_i - f^T x_j)^2 W_X(i,j). \quad (1)$$

We can interpret the first mean-squared error term of $C(f)$ as penalizing the difference between a one-dimensional projection of the instance x_i and the label y_i. The second term enforces the preservation of the neighborhood relationship within X (where W_X is a similarity measure). Under this interpretation, manifold regularization constructs embeddings preserving both the topology of the manifold and a 1-dimensional real-valued output structure. The proposed approach generalizes this idea to compute higher order locality-preserving discriminative projections.

Many linear (e.g., PCA, LPP) and nonlinear (e.g., Laplacian eigenmaps [Belkin and Niyogi, 2003]) dimensionality reduction methods convert dimensionality reduction problems to an eigenvalue decomposition. One key limitation of these approaches is that when they learn lower dimensional embeddings, they do not take label information into account. So only the information that is useful to preserve the topology of the whole manifold is guaranteed to be kept, and the discriminative information separating instances from different classes may be lost. For example, when we are required to describe a human being with a couple of words, we may use such characteristics as two eyes, two hands and so on. However, none of these features is useful to separate men from women.

Linear Discriminant Analysis (LDA) and some of its extensions like semi-supervised discriminant analysis [Cai, He, and Han, 2007; Zhao et al., 2007] find a dimensionality-reducing projection that best separates two or more classes of objects or events. The resulting combination may be used as a linear classifier, or for dimensionality reduction before later classification. However, for a dataset with c class labels, LDA type approaches can only achieve a $c-1$ dimensional embedding (since the matrix to model the between-class difference only has $c-1$ nontrivial eigenvectors). The proposed approach can be distinguished from some recent work. LDPP [Villegas and Paredes, 2008] learns the dimensionality reduction and nearest neighbor classifier parameters jointly. LDPP does not preserve the topology of the given dataset. The algorithm in [Pham and Venkatesh, 2008] provides a framework to learn a (local optimal) linear mapping function that maps the given data to a new space in order to enhance a given classifier. Their mapping function is designed for classification only and does not preserve the topology of the dataset. [Zhang, Zhou, and Chen, 2007] incorporates pairwise constraints for finding a better embedding. This approach does not consider the fact that some classes are more similar to each other compared to the other classes.

Similar to our approach, the well-known Canonical Correlation Analysis (CCA) [Hotelling, 1936] and recent work on label embedding [Weinberger and Chapelle, 2008] also simultaneously compute two mapping functions. For example, CCA finds linear functions that map instances from two different sets to one space, where the correlation between the corresponding points is maximized. There is a fundamental difference between our approach and these approaches: Our approach can make use of unlabeled data to handle overfitting problems, while approaches like CCA cannot.

3 Overall Framework

We introduce the overall framework in this section. It is helpful to review the notation described below. In particular, we assume that class labels can be viewed as c-dimensional real-valued vectors if there are c possible labels.

3.1 The Problem

Assume the given dataset $X = (x_1, \cdots, x_m)$ is a $p \times m$ matrix, where instance x_i is defined by p features. c = number of classes in X. Label y_i is a $c \times 1$ vector representing x_i's class label. If x_i is from the j^{th} class, then $y_i(j) = 1$; $y_i(k) = 0$ for any $k \neq j$. We also assume x_i's label is given as y_i for $1 \leq i \leq l$; x_i's label is not available for $l+1 \leq i \leq m$. $Y = (y_1, \cdots, y_l)$ is a $c \times l$ matrix.

The problem is to compute mapping functions f (for data instances) and g (for labels) to map data instance $x_i \in R^p$ and label $y_i \in R^c$ to the same d-dimensional space, where the topology of the data manifold is preserved, the instances from different classes are separated and $d \ll p$. Here, f is a $p \times d$ matrix and g is a $c \times d$ matrix.

3.2 The Cost Function

The solution to the overall problem of learning discriminative projections can be formulated as constructing mapping functions f and g that minimize the cost function $C(f, g) =$

$$\frac{\sum_{i \le l} \|f^T x_i - g^T y_i\|^2 + \mu \sum_{i,j} \|f^T x_i - f^T x_j\|^2 W_X(i,j)}{\sum_{i \le l} \sum_{k=1, s_k \ne y_i}^{c} \|f^T x_i - g^T s_k\|^2},$$

where s_k and W_X are defined as follows: s_k is a $c \times 1$ matrix. $s_k(k) = 1$, and $s_k(j) = 0$ for any $j \ne k$. S_k is a $c \times l$ matrix= $(s_k, \cdots, s_k)$. W_X is a matrix, where $W_X(i,j)$ is the similarity (could be defined by k-nearest neighbor approach) between x_i and x_j.

Here, $f^T x_i$ is the mapping result of x_i. $g^T y_i$ (or $g^T s_k$) is the mapping result of label y_i (or s_k). The first term in the numerator represents the difference between the projection result of any instance x_i and its corresponding label y_i. We want this value to be small, since this makes x_i be close to its true label. The second term in the numerator models the topology of dataset X using both labeled and unlabeled data. When it is small, it encourages the neighborhood relationship within X to be preserved. μ is a weight to balance the first and second terms. It is obvious that we want the numerator of $C(f, g)$ to be as small as possible. The denominator models the distance between the projection result of each instance x_i and all the labels other than the correct label. We want this value to be as large as possible, since this makes x_i be far away from its wrong labels. Thus, minimizing $C(f, g)$ is equal to learning f and g jointly to preserve the topology of dataset X, and project instances to a new lower dimensional space, where the instances from the same class are close to each other and the instances from different classes are well separated from each other.

3.3 High Level Explanation

Manifold regularization addresses the problem of learning projections to map the data instances (with known labels) to their class labels, preserving the manifold topology of the whole dataset (considering both labeled and unlabeled data). The ability to make use of unlabeled data reduces the possibility of overfitting. A loss function example under this framework is described in Equation (1), and can be generalized for our problem. Our goal is to jointly separate the instances from different classes and learn data-dependent labels, so whether the data will be projected to its original label is not important as long as the instances from the same class will stay together and the instances from different classes will be separated. In our algorithm, we have a mapping function f for data instances, and g for labels such that f and g can work together to map the data instances and labels to a new space, where the instances and their new labels are matched. The mapping g is decided by both the given class labels and the data manifold topology, and allows us to scale the entries of the label vector by different amounts, which then allows better projections of points. With the flexibility offered by g, we can project similar classes to similar locations (this is learned from the data manifold topology), and achieve the embedding results of a dimensionality bounded by the number of input features rather than the number of classes (without using g, our result is also bounded by c).

In summary, the numerator of our loss function encourages the instances with the same label to stay together, preserving the data manifold topology. The denominator of the loss function encourages the instances with different labels to be away from each other. The mapping g provides the best label projection with regard to the given loss function, and the mapping f projects the data to match the resulting new labels. Manifold topology is respected to handle the possible overfitting problems.

3.4 Discriminative Projections: The Main Algorithm

Some notation used in the algorithm is as follows:
$\gamma = (f^T, g^T)^T$ is a $(p + c) \times d$ matrix. $Tr()$ means trace. I is an $l \times l$ identity matrix. L_X is a graph Laplacian matrix [Belkin and Niyogi, 2003] corresponding to W_X.

$$U_1 = \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix}_{m \times m}, U_2 = U_3^T = \begin{pmatrix} I \\ 0 \end{pmatrix}_{m \times l}, U_4 = I.$$

The algorithmic procedure is as follows:

1. **Construct matrices A, B and C:**

$$\begin{aligned} A &= \begin{pmatrix} X & 0 \\ 0 & Y \end{pmatrix} \begin{pmatrix} U_1 & -U_2 \\ -U_3 & U_4 \end{pmatrix} \begin{pmatrix} X^T & 0 \\ 0 & Y^T \end{pmatrix} \\ B &= \sum_{k=1}^{c} \begin{pmatrix} X & 0 \\ 0 & S_k \end{pmatrix} \begin{pmatrix} U_1 & -U_2 \\ -U_3 & U_4 \end{pmatrix} \begin{pmatrix} X^T & 0 \\ 0 & S_k^T \end{pmatrix} \\ C &= \begin{pmatrix} X & 0 \\ 0 & Y \end{pmatrix} \begin{pmatrix} \mu L_X & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} X^T & 0 \\ 0 & Y^T \end{pmatrix} \end{aligned}$$

2. **Compute $\gamma = (\gamma_1, \cdots, \gamma_d)$: the d minimum eigenvectors of the generalized eigenvalue decomposition equation:**

$$(A + C)x = \lambda(B + C)x.$$

3. **Compute discriminative projections f and g:**
$\gamma = (\gamma_1, \cdots, \gamma_d)$ is a $(p + c) \times d$ matrix, whose top p rows= mapping function f, the next c rows= mapping function g. i.e.

$$\begin{pmatrix} f \\ g \end{pmatrix} = (\gamma_1, \cdots, \gamma_d).$$

4. **Compute the d-dimensional embedding of dataset X:**
The d-dimensional embedding of X is $f^T X$, whose i^{th} column represents the embedding of x_i.

3.5 Justification

Theorem 1. *The d minimum eigenvector solutions of the equation $(A + C)x = \lambda(B + C)x$ provide the optimal d-dimensional discriminative projections to minimize the cost function $C(f, g)$.*

Proof: Given the input and the cost function, the problem is formalized as:

$$\{f, g\} = \arg_{f,g} \min(C(f, g)).$$

When $d = 1$, we define M, N and L as follows:

$$M = \sum_{i\leq l}(f^T x_i - g^T y_i)^2, N = \sum_{i\leq l}\sum_{k=1}^{c}(f^T x_i - g^T s_k)^2,$$

$$L = \mu \sum_{i,j}(f^T x_i - f^T x_j)^2 W_X(i, j).$$

$$\begin{aligned} &\arg_{f,g} \min(C(f, g)) \\ &= \arg_{f,g} \min \frac{M + L}{N - M} = \arg_{f,g} \max \frac{N - M}{M + L} \\ &= \arg_{f,g} \max \frac{N + L}{M + L} = \arg_{f,g} \min \frac{M + L}{N + L}. \end{aligned}$$

$$\begin{aligned} M &= (f^T X, g^T Y) \begin{pmatrix} U_1 & -U_2 \\ -U_3 & U_4 \end{pmatrix} \begin{pmatrix} X^T f \\ Y^T g \end{pmatrix} = \gamma^T A\gamma. \\ N &= (f^T, g^T)B \begin{pmatrix} f \\ g \end{pmatrix} = \gamma^T B\gamma. \\ L &= \mu f^T X L_X X^T f = \gamma^T C\gamma. \end{aligned}$$

$$\arg_{f,g} \min C(f, g) = \arg_{f,g} \min \frac{\gamma^T (A + C)\gamma}{\gamma^T (B + C)\gamma}.$$

It follows directly from the Lagrange multiplier method that the optimal solution that minimizes the loss function $C(f, g)$ is given by the minimum eigenvector solution to the generalized eigenvalue problem: $(A + C)x = \lambda(B + C)x$.

When $d > 1$,

$$\begin{aligned} M &= \sum_{i\leq l} \|f^T x_i - g^T y_i\|^2 = Tr((\gamma_1 \cdots \gamma_d)^T A(\gamma_1 \cdots \gamma_d)). \\ N &= \sum_{i\leq l}\sum_{k=1}^{c} \|f^T x_i - g^T s_k\|^2 \\ &= Tr((\gamma_1 \cdots \gamma_d)^T B(\gamma_1 \cdots \gamma_d)). \\ L &= \mu \sum_{i,j} \|f^T x_i - f^T x_j\|^2 W_X(i, j) \\ &= Tr((\gamma_1 \cdots \gamma_d)^T C(\gamma_1 \cdots \gamma_d)). \end{aligned}$$

$$\begin{aligned} &\arg_{f,g} \min C(f, g) \\ &= \arg_{f,g} \min \frac{Tr((\gamma_1 \cdots \gamma_d)^T (A + C)(\gamma_1 \cdots \gamma_d))}{Tr((\gamma_1 \cdots \gamma_d)^T (B + C)(\gamma_1 \cdots \gamma_d))}. \end{aligned}$$

Standard approaches [Wilks, 1963] show that the solution to $\gamma_1 \cdots \gamma_d$ that minimizes $C(f, g)$ is provided by the eigenvectors corresponding to the d lowest eigenvalues of the equation: $(A + C)x = \lambda(B + C)x$. □

The mapping functions f and g are linear. For them to be nonlinear, we can directly compute the embedding result of each given instance and label (use u_i to replace $f^T x_i$ and v_j to replace $g^T y_j$). The corresponding cost function and algorithm can be given in a similar manner as the linear case discussed in this paper. This new problem is in fact technically less challenging. The same problem can also be solved using the framework of kernel CCA [Kuss and Graepel, 2003].

4 Experimental Results

In this section, we test discriminative projections, LDA, CCA and LPP using two datasets: recognition of handwritten digits using the USPS dataset (an image dataset with multiple classes), and TDT2 data (a text dataset with multiple classes). We use the following simple strategy to decide the value of μ in the loss function $C(f, g)$. Let s be the sum of all entries of W_X and $l =$ the number of training examples with labels, then l/s balances the scales of the first term and second term in the numerator of $C(f, g)$. We let $\mu = l/s$, if we treat accuracy and topology preservation as equally important. We let $\mu > l/s$, when we focus more on topology preservation; $\mu < l/s$, when accuracy is more important. In this paper, we use $\mu = l/s$ for discriminative projections.

4.1 USPS Digit Data (Image Data)

The USPS digit dataset (www.gaussianprocess.org/gpml/data) has 9298 images. We randomly divided it into a training set (2000 cases) and a test set (7298 cases). Each image contains a raster scan of the 16 × 16 grey level pixel intensities. The intensities have been scaled to the range [-1, 1].

We first computed lower dimensional embeddings of the data using discriminative projections, LDA, CCA and Locality Preserving Projections (LPP). This dataset has 10 labels, so LDA can only return an embedding of 9 or less dimensions. LPP, CCA and discriminative projections can return an embedding of any dimensionality. The 3D and 2D embedding results are shown in Figure 2.

To quantitatively study how the discriminative information is preserved by different approaches, we ran a leave-one-out test. We first computed 9D and 100D embeddings using discriminative projections, LPP and CCA. We also computed 9D embedding using LDA. Then we checked for each point x_i whether at least one point from the same class were among its K nearest neighbors in the new space. We tried $K = 1, \cdots, 10$. The results are summarized in Figure 3. From the figure, we can see that discriminative projections (100 dimensional), and LPP (100 dimensional) achieve similar performance (LPP is slightly worse), and outperform the other approaches. This shows that discriminative projections and LPP with more features are able to better preserve the discriminative information of the given dataset compared to using less features. Recall that the mapping results from LDA are bounded by the number of classes, and can not generate embedding results of high dimensionality in this test. Comparing all 4 approaches resulting in 9D embedding results, CCA and LDA are still worse than Discriminative projections and LPP. We checked the projection results of the training data, and found that CCA and LDA worked perfectly for the training data. This implies that LDA and CCA results overfitted for the training data, and did not generalize well to the test data. On the contrary, discriminative projections and LPP are unlikely to run into overfitting, since they take the unlabeled data into consideration when the projections are constructed.

We also used this example to visualize the new "prototype" of each label in a 2D space (Figure 4). The original labels are in a 10D space. The new labels are constructed by projecting the old labels onto the space spanned by the first two columns of mapping function g. When $\mu = 10^3$, we can see

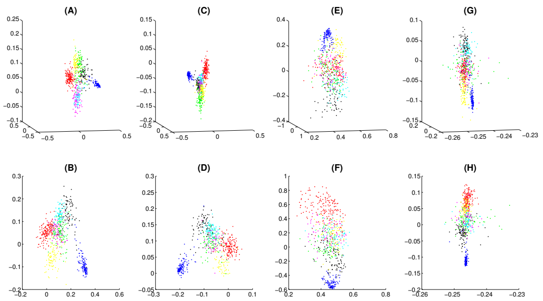

Figure 2: USPS digit test: (the color represents class label): (A) discriminative projections using 3D embedding; (B) discriminative projections using 2D embedding; (C) LDA using 3D embedding; (D) LDA using 2D embedding; (E) LPP using 3D embedding; (F) LPP using 2D embedding; (G) CCA using 3D embedding; (H) CCA using 2D embedding.

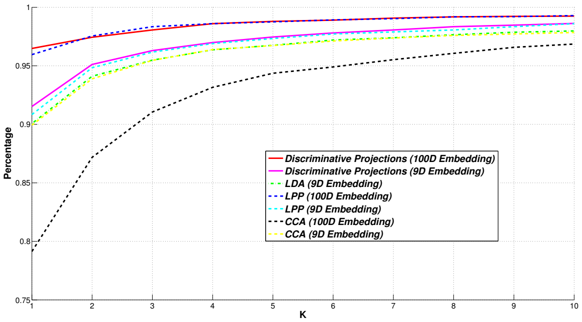

Figure 3: USPS test: This experiment measures how well the discriminative information is preserved.

from the left figure that new labels of similar digits are close to each other in the new space. For example, '0' and '8' are together; '3', '6' and '9' are also close to each other. When μ is large, we focus more on topology preservation. The result makes sense, since to preserve local topology of the given dataset, similar digits have a large chance of being projected to similar locations. We ran another test with less respect to manifold topology (by setting $\mu = 10^{-3}$). In the new scenario, the new labels were much better separated in the new space (right figure). This experiment shows that the mapping g allows us to scale the entries of the label vector by different amounts for different applications, which then allows more flexible projections of instances.

4.2 TDT2 Data (Text Data)

The TDT2 corpus consists of data collected during the first half of 1998 and taken from 6 sources, including 2 newswires (APW, NYT), 2 radio programs (VOA, PRI) and 2 television programs (CNN, ABC). It consists of more than 10,000 documents which are classified into 96 semantic categories. In the dataset we are using, the documents that appear in more than one category were removed, and only the largest 4 categories were kept, thus leaving us with 5,705 documents in total.

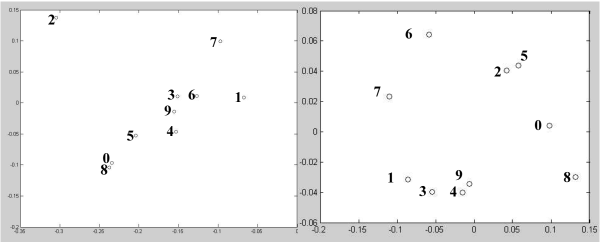

Figure 4: (left) Projection results of 10 USPS digit labels (μ=1000). (right) Projection results of 10 USPS digit labels (μ=0.001).

We applied our approach, LDA, CCA and LPP to the TDT2 data assuming label information of 1/3 documents from each class was given, i.e. $l = 5,705/3$. To see how the discriminative information is preserved by different approaches, we ran a similar leave-one-out test. Again, we first computed 3D and 100D embeddings using discriminative projections CCA and LPP. We also computed the 3D embedding using LDA. Then we checked for each document x_i whether at least one document from the same class was among its K nearest neighbors in the new space (we use this as correctness). We tried $K = 1, \cdots, 10$. The results are summarized in Figure 5. From this figure, we can see that discriminative projections perform much better than the other approaches in all 10 tests followed by LPP (100D), LPP(3D), LDA(3D) CCA(100D) and CCA(3D).

Generally speaking, LDA does a good job at preserving discriminative information, but it does not preserve the topology of the given manifold and not suitable for many dimensionality reduction applications, which need an embedding defined by more than $c - 1$ features. Further, when the labeled data is limited, LDA could run into overfitting problems. Similar to LDA, CCA is also likely to have overfitting problems when the labeled data is not sufficient. To see how overfitting problems affect the performance, we applied all four approaches to the data, and visualized the 2D embedding results of the training data and test data in Figure 6. The figure shows that the training and test embedding results of discriminative projections and LPP are similar, but quite different for CCA and LDA. For CCA and LDA, the embedding results of the training data from each individual class converge to a single point in the figure. However, the embedding results are scattered across the figures for the test data. This strongly indicates that their projection functions are over-tuned to the training data and do not generalize well to the test data. As a representative approach of regular dimensionality reduction approaches, LPP can preserve the manifold topology, return the embedding result of a dimensionality only bounded by the number of input features. However, LPP totally disregards the label information, and behaved much worse than discriminative projections in this test: 10% worse on 100D embedding and 25% worse on 3D embedding results. Discriminative projections combines the ideas of LDA, CCA and LPP, such that both manifold topology and the class separability will be preserved. In addition, depending on the applications, users may decide how to choose μ to balance the two goals. If we focus more on the manifold topology, we choose a larger

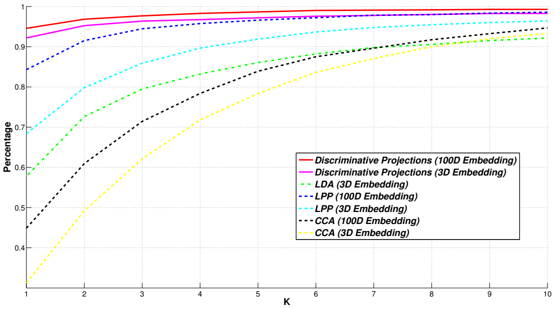

Figure 5: TDT2 test: This experiment measures how well the discriminative information is preserved.

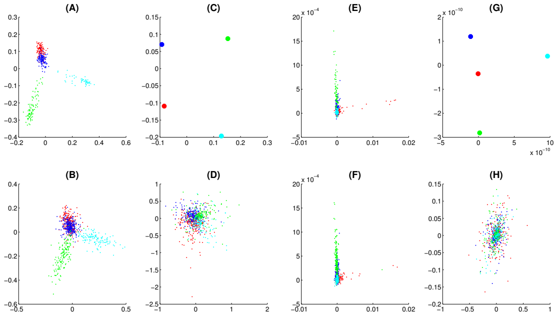

Figure 6: USPS digit test: (the color represents class label): (A) discriminative projections using 2D embedding (training); (B) LDA using 2D embedding (training); (C) LPP using 2D embedding (training); (D) CCA using 2D embedding (training); (E) discriminative projections using 2D embedding (testing); (F) LDA using 2D embedding (testing); (G) LPP using 2D embedding (testing); (H) CCA using 2D embedding (testing).

value for μ; otherwise, we choose a smaller value for μ.

5 Conclusions

In this paper, we introduced a novel approach (“discriminative projections”) to jointly learn data-dependent label and locality-preserving projections. The new approach is able to construct discriminative features to map high-dimensional data instances to a new lower dimensional space, preserving both manifold topology and class separability. Leveraging the flexibility of labels, discriminative projections goes beyond LDA and LPP in that it can project similar classes to similar locations, and provide an embedding of an arbitrary dimensionality rather than $c - 1$ for LDA in a problem with c class labels. It also differs from the other regular dimensionality reduction since the discriminative information to separate instances from different classes will be preserved. Our approach is a semi-supervised approach making use of both labeled and unlabeled data. It is general, since it can handle both two class and multiple class problems. In addition to the theoretical validations, we also presented real-world applications of our approach to information retrieval and a digit recognition task in image analysis.

Acknowledgments

We thank the reviewers for their helpful comments. This material is based upon work supported by the National Science Foundation under Grant Nos. NSF CCF-1025120, IIS-0534999, and IIS-0803288.

References

[Belkin and Niyogi, 2003] Belkin, M., and Niyogi, P. 2003. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation* 15:1373–1396.

[Belkin, Niyogi, and Sindhwani, 2006] Belkin, M.; Niyogi, P.; and Sindhwani, V. 2006. Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research* 2399–2434.

[Cai, He, and Han, 2007] Cai, D.; He, X.; and Han, J. 2007. Semi-supervised discriminant analysis. In *Proceedings of the International Conference on Computer Vision (ICCV)*.

[Fukunaga, 1990] Fukunaga, K. 1990. *Introduction to statistical pattern classification*. Academic Press.

[He and Niyogi, 2003] He, X., and Niyogi, P. 2003. Locality preserving projections. In *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*.

[Hotelling, 1936] Hotelling, H. 1936. Relations between two sets of variates. *Biometrika* 10:321–377.

[Kuss and Graepel, 2003] Kuss, M., and Graepel, T. 2003. The geometry of kernel canonical correlation analysis. Technical report, Max Planck Institute for Biological Cybernetics.

[Pham and Venkatesh, 2008] Pham, D. S., and Venkatesh, S. 2008. Robust learning of discriminative projection for multicategory classification on the stiefel manifold. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Villegas and Paredes, 2008] Villegas, M., and Paredes, R. 2008. Simultaneous learning of a discriminative projection and prototypes for nearest-neighbor classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Weinberger and Chapelle, 2008] Weinberger, K., and Chapelle, O. 2008. Large margin taxonomy embedding for document categorization. In *Proceedings of the Advances in Neural Information Processing Systems*.

[Wilks, 1963] Wilks, S. S. 1963. *Mathematical statistics*. Wiley.

[Zhang, Zhou, and Chen, 2007] Zhang, D.; Zhou, Z.; and Chen, S. 2007. Semi-supervised dimensionality reduction. In *In Proceedings of the 7th SIAM International Conference on Data Mining*, 11–393.

[Zhao et al., 2007] Zhao, D.; Lin, Z.; Xiao, R.; and Tang, X. 2007. Linear laplacian discrimination for feature extraction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.