Adaptively Discriminant Locality Preserving Projection

Zipei Chen1,\*

**1College of Art and Science**

**University of Miami, Florida 33146, United States**

\*zxc329@miami.edu

***Abstract*—Dimensionality reduction has been playing a significant role in many fields such as recognition, classification, clustering, high-dimensionality data compression. However, due to the existence of noises in the feature space of the original data, manifold learning methods take risks of finding the k nearest neighbors. LAPP designed a "coarse to fine" strategy to iteratively obtain the optimal subspace to solve this problem and obtain the optimal subspace. However, Since the discriminant information is also essential for the recognition and classification, ADLPP combined this “coarse to fine” idea with the idea of Supervised learning, which could not only preserve the local information after projection, solve the problem of noises and obtain the optimal subspaces, but also gain better performance on classification.**

Keywords: dimesionality reduction; supervised learning; manifold learning; LPP; DLPP; LAPP; ADLPP

##### 1.INTRODUCTION

In the era of big data, researchers are confronting the problem of the high dimensionality of data in a vast amount of research fields ranging from face recognition to business data mining [1-3]. Imagine we have a dataset with the size of nm, where n refers to the number of the total samples and m refers to the dimension of each sample. If m is huge, directly using these data for calculations will result in many kinds of problems such as insufficient ram capacity. For example, a data set of housing information, each sample contains 50 features ranging from the price of the houses and the house's areas to some small details such as the wallpaper texture. It is very straightforward that many features are not very important and relevant to the sample and fail to reflect the intrinsic structure of the sample directly. Keeping containing such irrelevant features could result in degraded performance of a decision system [4]. Therefore, eliminating these redundant data and reducing dimensionality has become an indispensable technique in many fields. The main idea of dimension reduction is to find a new axis and project our original data onto new axes to realize the dimension reduction.

There are three mainstream dimensionality reduction research lines: unsupervised learning method, supervised learning method, and manifold learning method. PCA (Principal Component Analysis) is currently the most universally known and widely used unsupervised learning dimensionality reduction algorithm. Its core idea is to construct the covariance matrix and then calculate the optimal eigenvalue and eigenvector based on the covariance matrix we constructed before, in other words, to ensure that the variance of the data is as large as possible after projection. Finally, take the top k best eigenvectors and project the original data onto these eigenvectors to realize dimensionality reduction. However, Since PCA is unsupervised learning, it does not use label information, so when PCA is dealing with data with category labels, their label information may not be preserved after dimensionality reduction. At this time, LDA appeared. LDA is a well-known algorithm in the field of supervised learning dimensionality reduction. Its core idea is to construct the within-class scatter matrix to make sure the data with the same label stay closer to each other and then build the inter-class scatter matrix to ensure the distance between data with different labels as far as possible. Therefore, the data could be successfully classified, and thus their label information would be preserved. However, Both PCA and LDA could only see the global structure of the space; the data could not remain their local information after projection [1, 2]. Imagine that we now have a globe and want to measure the distance between Australia and United States. It is straightforward that we can use a soft tape measure to measure the distance. Suppose we now put the globe in a three-dimensional rectangular coordinate system, and the coordinates of each country can be expressed as (x, y, z). Can we use the Euclidean distance based on the coordinates information to calculate the distances at this time? No, because the local information that the globe is a sphere would be lost based on Euclidean distance measurement. Hence, the manifold learning dimensionality reduction method appeared. Unlike LDA and PCA, which could only see the Euclidean structure of the space, the manifold learning method effectively finds the embedding that successfully preserves the local information and obtains the manifold structure [1]. Isometric Mapping (ISOMAP) is to find a low-dimensional space so that the distance between samples is basically the same in high-dimensional space and low-dimensional space, locally linear embedding (LLE) aimed to preserve the linear relationship between the samples in each field After the samples are mapped from the high-dimensional space to the low-dimensional space [5, 6]. Although Both of them achieve nice performance in different research fields, they are not capable of providing an explicit mapping from the original high dimension to target dimension space [7, 9, 8]. LPP (Locality Preserving Projection) successfully solved this problem by preserving the local neighborhood information and therefore obtaining the linear transformation matrix. DLPP (Discriminant Locality Preserving Projection) applied supervised learning and used labels to better perform classification tasks by minimizing the distance between the data within the class and the distances between the classes [10].

Although LPP and DLPP have achieved satisfactory results in practical applications, however, due to the existence of noise, when LPP and LDPP are looking for neighboring points, they will inevitably find some wrong nearest neighbors, which will result in the drop-down of the accuracy and the deterioration performance of the model. Furthermore, due to the high dimensionality of the original data, such as the face dataset, it is very tough for us to remove noises directly from the original data. Also, we have no idea of the optimal subspace. Therefore, LAPP solved this problem by determining the neighboring information in the optimal subspace rather than in the original feature space. Since the noises in the new feature space after the transformation is lower than the noises in the original feature spaces, LAPP iteratively obtains the subspace, determines the neighboring information in the subspace, obtains the new subspace, and ends this loop whenever it finds the optimal subspace [8].

From the perspective of this coarse-to-fine strategy of LAPP and the supervised learning idea aimed for better performance on the classification task, we proposed ADLPP (adaptively Discriminant Locality Preserving Projection）which not only applied the label information for maximizing the distances between different classes and minimizing the distances between each sample within each class but also applied this coarse-to-fine idea by adaptively find the optimal subspace. With the addition of labeling information, ADLPP performed better on classification tasks than LAPP. Furthermore, ADLPP also performed better than DLPP because of the more precise determination of nearest neighbors on optimal subspace than the original feature space.

##### 2. RELATED WORKS

This section reviews the related three algorithms (LPP, DLPP, and LAPP).

*2.1* *LPP*

Different from other manifold learning methods (ISOMAP, LLE), which is nonlinear and could only work on training data because it is hard for these methods to find a satisfying nonlinear transformation on new testing data, LPP is linear, which means it could work on data defined anywhere in the space, not just in the training set. Imagine our dataset is X = {, , } with the size of n \* m. where n is the number of samples and m is the number of features. The goal for the LPP method is to find a transformation matrix A that transform our original data X into Y = {, , } =X with the size of n k where n is the number of samples and k is the target number of features. Also, the transformation matrix A we found need to satisfy that the similarity between each sample in the original dataset could be preserved after the projection. Therefore, the objective function could be like this:

Where Y is the transformed data and S is the similarity matrix (weight matrix) which represents the similarity between each sample in the original feature space.

The main procedure of this algorithm is as follows:

**(1) constructing the similarity matrix (weight matrix) W:**

To construct the similarity matrix, we need to find the k nearest neighbors of each node. This is realized by calculating the Euclidean distance between and nodes other than , and then sorting to find the k nearest nodes. Two Nodes will be connected if both are each other’s k nearest neighbors. Then we need to calculate the weight between each node with their k nearest neighbors. There are two methods for computing the weight.

1. *; Otherwise, 0.* (2)

*;*

Usually, we take the Heat kernel to calculate the weight matrix because LPP would lose the distance information if we take the second method.

**(2) reduce the objection function based on the Laplacian matrix L = D-W:**

Let a be the transformation vector, . By simple algebra formulation, the object function could be reduced to:

*=*

*=* (3)

*=*

*=*

*= ) =*

Where W is the similarity matrix, D is a diagonal matrix, the entry of D is the Column sum of W, L is the Laplacian matrix which equals to D - W.

**(3) put constrain and solve the generalized eigen vector problem:**

In order to reduce the scaling factor, impose constrain , Therefore the objection function reduced to:

The problem reduced to solve a generalized eigen vector problem to find a:

*2.2 DLPP*

LPP ignored label information which is important for the recognition task. DLPP **[10]** combined the idea of Supervised learning and Manifold learning. therefore, the data after transformation could not only preserve the local information but also performed better on classification task than other manifold learning method. After adding the discriminant information into the original LPP, the object function become this:

(6)

Where , M is the weight matrix of the within-class data. represents the weights between sample i and sample j. M contains the label information, if sample i and sample j are not in the same class, the weight would be set to 0. , represents the average point of each class, represents the weights between and .

The main procedure of this algorithm is as follows:

**(1) constructing the similarity matrix (weight matrix) M and W:**

The only difference between LPP and DLPP in this step is the weight calculation of M. Since M contains the label information, the rule becomes:

1. *belong to the same* (7)

**(2) reduce the objection function based on the Laplacian matrix L = D-W:**

The algebra formulation is similar with LPP, which is to construct Laplacian matrix:

=

= (8)

=

=

=

Where M is the similarity matrix for within-class matrix, W is the similarity matrix for between-class matrix. D is the diagonal matrix which has each entry equal to the column sum of M, K is the diagonal matrix which has each entry equal to the column sum of W. M, and are the Laplacian matrix. Suppose our original data is X, and the average point for each class is B, transformation vector is a, Then z = , = .

**(3) solve the generalized eigen vector problem to find the transformation vector a:**

*2.3 LAPP*

LPP is taking risks when choosing the k nearest neighbors because of the existence of noises in the original data set. The wrong determination of k nearest neighbors would therefore lower the performance of the whole algorithm. LAPP takes a coarse-to-fine strategy to iterate the LPP until finding the optimal subspace which has satisfied the requirement of a small number of noises.

Algorithm1 shows the pseudo-code of the main procedure of LAPP [8].

**Algorithm1: pseudo-code of Locality Adaptive Preserving Projections.**

**Input**: X = [,final dimension d, threshold , maximum iteration number T.

**Output:** transformation matrix **A**

1. calculate the similarity matrix () between
2. calculate **S, D** and **L** according to ()
3. solve the generalized Eigenvalue problem to obtain **A**
4. iteration = 0
5. **while** iteration < T
6. obtain the transformed data  **=**
8. Measure the similarity matrix ()
9. Calculate **S, D** and **L**
10. Solve a generalized eigenvalue problem to obtain A:
11. **if** diff (**A**, ) <
12. **A** = **, break;**
13. **end if**
14. iteration = iteration +1
15. **end while**
16. **return A**

##### 3. ADLPP

With the help of adaptively obtaining the optimal subspace, LAPP achieved satisfactory results on both local information preservation and recognition task. However, Since LPP didn’t adopt discriminant information which is one of the important information for the recognition task, the precision on classification and recognition task of LAPP would also be limited, although it is good. DLPP takes the benefit of discriminant information which enable this algorithm to maximize its capability on classification and recognition task. However, DLPP contains the same problem as LPP does. In order to preserve the locality information, the process to find the k nearest neighbors is indispensable, and DLPP would also take risks at choosing the k nearest neighbors because of the existence of noises in the original feature space. Therefore, it is straightforward that we can use the idea of adaptively obtaining the optimal low noises containing subspace to complement DLPP's problem. Also, the idea of Supervised learning from DLPP to harness the benefit of labels would also complement LAPP’s problem. ADLPP is the combination of these two ideas which has better performance on recognition and locality information preserving.

Specifically, we compute the within-class similarity matrix of the whole dataset based on the rules defined at (7). Then we compute the diagonal matrix based on the column sum of and compute depend on . After we finished the within-class matrix, we then repeat the same process to construct the between-class matrix and therefore the diagonal matrix and the Laplacian matrix . After that, we solve the generalized eigenvector problem to find the transformation matrix a and obtain the new subspace = .

Finally, we repeat the whole procedure until we find the optimal subspace. There are two ways to determine the optimal subspace:

1. if the iteration times reach the maximum iteration number T we set.
2. if the difference between the transformation matrixes a in two continuous time is lower than the Threshold we set.

Algorithm2 shows the pseudo-code.

**Algorithm2: pseudo-code of Adaptively Discriminant Locality Preserving Projections.**

**Input**: X = [, Y = [final dimension d, threshold , maximum iteration number T.

**Output:** optimal subspace X and transformation matrix a.

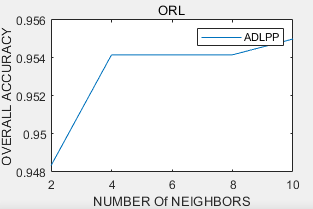
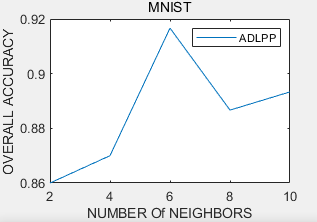
1. compute the within-class similarity matrix () between based on (7)
2. compute the diagonal matrix and Laplacian matrix based on .
3. compute the within-class similarity matrix () between based on (2)
4. compute the diagonal matrix and Laplacian matrix based on .
5. solve the generalized eigen vector problem (9) to find a.
6. iteration = 0
7. **while** iteration < T
8. obtain the transformed data  **=**
10. compute
11. compute
12. solve the generalized eigen vector problem to obtain **a**
13. **if** <
15. ;
16. **end** **if**
17. iteration = iteration +1
18. **end while**
19. **return**

##### 4.EXPERIMENT RESULT

To evaluate the performance of our model, we conduct experiments on the MNIST handwritten dataset (MNIST), Olivetti research laboratory database (ORL), Yale face database (Yale). we made a comparison of ADLPP with other four dimensionality reduction models including two supervised learning local models (LDA, LFDA) and two manifold learning local models (LPP, LAPP).

*4.0 Evaluation of k nearest neighbors*

Since for each local models including ADLPP need to find the k-nearest neighbors while constructing the Similarity matrix. Fig. 1. shows the overall accuracy of ADLPP based on different value of k after the training on each dataset. We can see the choice of number of neighbors indeed effect the performance of the model. Therefore, we applied the highest performance value of k for each model in the following experiments result comparisons.



图表, 折线图

描述已自动生成

Figure 1. the overall accuracy based on different value of k

*4.1 Results on MNIST dataset*

形状, 箭头

描述已自动生成

Figure 2. the exemplar of MNIST dataset

First, we conducted the experiment on the handwritten digit recognition dataset (MNIST). MNIST is the dataset containing the handwritten digits from 0 to 9. Fig. 2. shows the exemplar of the MNIST dataset. The size of each image is 28 28, and each pixel is a 784-dimensional feature vector. MNIST has separate training set and testing set. In this experiment, we select 2000 samples from the training set for training and select 2000 samples from the testing set for evaluation. Fig. 3. shows the overall accuracy of the five models on different values of dimension. Table 1 shows the Error rate with 6 training images per individual on the MNIST dataset. The dimension that results in the best error rate for each model is shown in the parenthesis. We observed that the overall performance of LAPP is better than the other four models. For this MNIST dataset, there is a problem for manifold learning. Since the hand-written digits contain no manifold information, it is very difficult for LAPP and LPP to capture the latest information. This result is consistent with the previous studies of [7], which also indicated that the performance of dimensionality reduction methods depends on the nature of the dataset [13]. With the help of the discriminant information, ADLPP happily solved this problem.

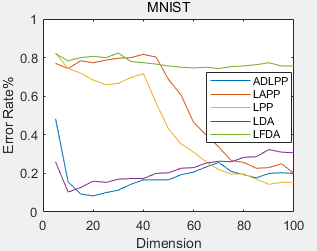
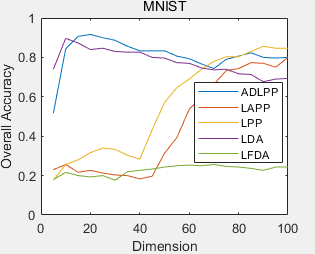


Figure3. the overall accuracy and error rate on dimensions from 1-100 training on MNIST dataset

Table 1. The error rate of each model on MNIST dataset

|  |  |
| --- | --- |
| Models | Error rate (%) |
| LFDA | 0.7433 (14) |
| LAPP | 0.2000 (100) |
| LPP | 0.1433 (90) |
| LDA | 0.1043 (10) |
| ADLPP | **0.0833 (30)** |

*4.2 Results on ORL dataset*

一群人的照片

描述已自动生成

Figure 4. the exemplar of ORL dataset

The Olivetti Research Laboratory database (ORL) contains 400 face images of 40 individuals and there are 10 samples for each individual [12]. These images were taken at various conditions: different times, facial expressions (open or closed eyes, smiling or not smiling), facial details facial details (wearing glasses or not wearing glasses) and lighting effects. Fig. 4. shows the exemplar of the ORL dataset. those images are 32 x 32 grayscale images, and each pixel is a 1024-dimensional feature vector. We randomly select 6 images for each individual as our training set and use the remaining images as our testing test. We repeat this selection for 5 times and report the average accuracy. Fig. 5. shows the overall accuracy of the 5 models on different dimensions. Table 2 shows the Error rate with 6 training images per individual on the ORL dataset. The dimension that results in the best error rate for each model is shown in the parenthesis. We could observe that the overall performance of ADLPP is better than the other 4 models.

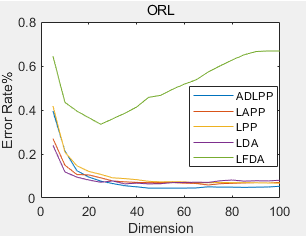
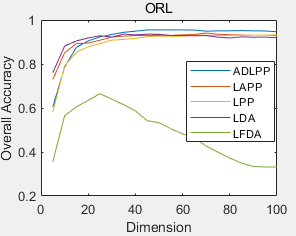


Figure 5. the overall accuracy and error rate on dimensions from 1-100 training on ORL dataset

Table 2. The error rate of each model on ORL dataset

|  |  |
| --- | --- |
| Models | Error rate (%) |
| LFDA | 0.3350 (5) |
| LAPP | 0.0592 (70) |
| LPP | 0.0675 (100) |
| LDA | 0.0642 (35) |
| ADLPP | **0.0450 (30)** |

*4.3 Results on YALE dataset*

许多狗的照片

描述已自动生成

Figure 6. the exemplar of YALE dataset

The YALE database contains 165 images of 15 different individuals and there are 11 images for each individual with different facial expressions [2]. The size of those images is 32 32 and they are both grayscale images. Figure 6 represents the exemplar of YALE dataset.

In this experiment, we randomly select 6 images per individual as our training set and use the remaining images for testing. We repeat this procedure for 5 times and report the average result. Fig. 7. represents the overall accuracy of the 5 models on different dimensions. We could observe that the overall performance of ADLPP is better than the other two manifold learning models (LPP and LAPP) and performed better than LDA when dimension is larger than 47.

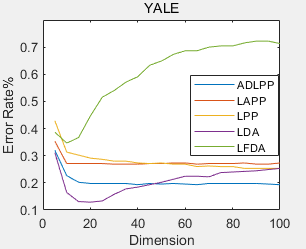
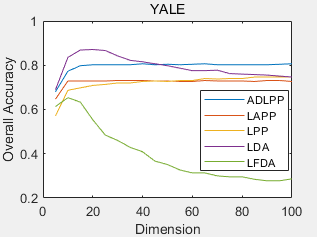


Figure 7. the overall accuracy and error rate on dimensions from 1-100 training on YALE dataset

##### 5.Conclusion

Dimensionality reduction has become an indispensable technique in many research fields. It has been playing a significant role in the process of recognition, classification, clustering, high-dimensionality data compression and so on. Due to the existence of noises in the feature space of the original data, manifold learning method would take risks of finding the k nearest neighbors. Therefore, LAPP designed an “coarse to fine” strategy to iteratively obtain the optimal subspace. However, Since the discriminant snformation is also very important for the recognition and classification, ADLPP combined the idea of LAPP and Supervised learning which could not only preserve the local information after projection and obtain the optimal subspaces, but also gain better classification result.

##### References

1. He, Xiaofei, et al. "Face recognition using laplacianfaces." *IEEE transactions on pattern analysis and machine intelligence* 27.3 (2005): 328-340.
2. Belhumeur, Peter N., Joao P. Hespanha, and David J. Kriegman. "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection." *IEEE Transactions on pattern analysis and machine intelligence* 19.7 (1997): 711-720.
3. Zhong, Xiao, and David Enke. "Forecasting daily stock market return using dimensionality reduction." *Expert Systems with Applications* 67 (2017): 126-139.
4. Qiao, Lishan, Song can Chen, and Xiaoyang Tan. "Sparsity preserving projections with applications to face recognition." *Pattern Recognition* 43.1 (2010): 331-341.
5. Tenenbaum, Joshua B., Vin De Silva, and John C. Langford. "A global geometric framework for nonlinear dimensionality reduction." *science* 290.5500 (2000): 2319-2323.
6. Roweis, Sam T., and Lawrence K. Saul. "Nonlinear dimensionality reduction by locally linear embedding." *science* 290.5500 (2000): 2323-2326.
7. Van Der Maaten, Laurens, Eric Postma, and Jaap Van den Herik. "Dimensionality reduction: a comparative." *J Mach Learn Res* 10.66-71 (2009): 13.
8. Aiguo Wang, Shenghui Zhao, Jinjun Liu, Jing Yang, Li Liu, Guilin Chen, Locality adaptive preserving projections for linear dimensionality reduction, Expert Systems with Applications, Volume 151, (2020), 113352, ISSN 0957-4174, <https://doi.org/10.1016/j.eswa.2020.113352>.
9. Krstanovic, Lidija , et al. "GMMs similarity measure based on LPP-like projection of the parameter space." Expert Systems with Applications An International Journal 66.Dec.(2016):136-148.
10. Yu, Weiwei, Xiaolong Teng, and Chongqing Liu. "Face recognition using discriminant locality preserving projections." *Image and Vision Computing* 24.3 (2006): 239-248.
11. Samaria, Ferdinando S., and Andy C. Harter. "Parameterisation of a stochastic model for human face identification." *Proceedings of 1994 IEEE workshop on applications of computer vision*. IEEE, 1994.
12. Passalis, Nikolaos, and Anastasios Tefas. "Dimensionality reduction using similarity-induced embeddings." *IEEE transactions on neural networks and learning systems* 29.8 (2017): 3429-3441.