GRAPH EMBEDDING DISCRIMINATIVE UNSUPERVISED DIMENSIONALITY REDUCTION

by

YUN LIU

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

THE UNIVERSITY OF TEXAS AT ARLINGTON

December 2014

Acknowledgements

I would like to express my gratitude towards my supervisor, Associate Professor Heng Huang, for his invaluable directions and support throughout my research efforts towards this dissertation. His insights and suggestions to the research area enlightened me in various detailed aspects throughout the work. His directions with regards to the development of my academic thinking and writing were inspiring and helpful for my current and future work.

Furthermore, I am thankful to Computational Science Laboratory for providing me the necessary infrastructure without which it would have been difficult for me to complete this dissertation. I would also like to thank all the other faculty members, members of the jury who evaluated my dissertation and the other students in Computer Science Laboratory for their extra ordinary support and encouragement.

October 2, 2014

Abstract

GRAPH EMBEDDING DISCRIMINATIVE UNSUPERVISED DIMENSIONALITY REDUCTION

Yun Liu, MS

The University of Texas at Arlington, 2014

Supervising Professor: Heng Huang

In this thesis, a novel graph embedding unsupervised dimensionality reduction method was proposed. Simultaneously, we assigned the adaptive and optimal neighbors on the basis of the projected local distances, thus we developed the dimensionality reduction along with the graph construction. The clustering results could be directly exhibited from the learnt graph which has the explicit block diagonal structure.

The analysis of experimental result on different databases also determines that the proposed dimensionality reduction method is superior to other related dimensionality reduction methods, like PCA and LPP. In this study, we use synthetic data and real-world benchmark data sets. Also experimental results from the clustering experiments revealed the proposed dimensionality reduction method outperformed other clustering methods, such as K-means, Ratio Cut, Normalized Cut and NMF.

Table of Contents

List of Illustrations

List of Tables

Chapter 1

Introduction

Dimensionality reduction explores methods that efficiently reduce high dimensionality data to low dimensionality data for data processing goals such as pattern recognition, information retrieval, machine learning, microarray data analysis, and data mining. Also, feature dimensionality reduction is an active research topic because of its importance and usefulness [1, 2, 3, 4, 5]. We assessed the area of dimensionality reduction by two important methods: feature extraction and feature selection. Feature extraction creates new features resulting from the functions of the original features; and feature selection chooses a subset of the original features. Data processing tasks will become efficient after applying the methods since feature extraction and feature selection both try to reduce the dataset dimensionality. We will review the main concepts of feature extraction and feature selection in this chapter, and we will introduce some basic applications.

Intensive research in dimensionality reduction area is being paid attention in the past decades. Still nowadays its necessitated demand is further increasing and there has been a notable shift in this area since important high-dimensional applications such as gene expression data, text categorization, and document indexing become widely used.

Dimension reduction plays an increasingly important role in the field of machine learning:

- Feature reduction involves removing features that provide non-significant information, thus the process of feature reduction can avoid the redundant information in gene expression data to confuse the analysis.
- Feature reduction makes the classification algorithms to reduce the computational burden and run much faster.
- Feature reduction can help the results of analysis easier and simple to understand.

- It is important to emphasize that feature reduction makes the classification algorithm to concentrate on the most correlated and useful features, thus boost the classification accuracy.
- Feature reduction helps to save the cost of data acquisition.

1.1 Objectives

Dimension reduction has been widely used in machine learning field. Typically, dimension reduction methods includes supervised dimension reduction method and unsupervised dimension reduction method. In the unsupervised dimension reduction method, the graph embedding method is more popular and reliable due to the using of data graph and manifold information.

However, most of the currently available dimensionality reduction methods require an affinity graph constructed beforehand. This affinity graph makes the projection process based on the input of the graph to a large extent. In this study, we proposed a novel graph embedding method for unsupervised dimension reduction. In this method, it is independent on the input of the graph.

1.2 Major contribution of the thesis

The major contributions of the thesis are highlighted in the following:

- A novel graph embedding method for unsupervised dimensionality reduction is proposed and analysis. Simultaneously, we conduct the graph construction in our model with dimensionality reduction. The adaptive and optimal neighbor on the basis of projected local distance is assigned.
- We compare our proposed method with other dimensionality reduction method, such as PCA and LPP. The algorithm is implemented in Matlab and tested on Synthetic data and real benchmark datasets. The experimental results show that

our proposed method is superior to other methods. This proves that our proposed method is very convenient to be applied in the dimensionality reduction.

1.3 Organization of the thesis

To give a distinct view about the thesis report, the chapters are arranged as below:

In Chapter 2, we review the background of pattern recognition system. In pattern recognition system, we introduced the four processes: pre-processing, feature extraction, classification and post-processing.

In Chapter 3, we review the linear feature extraction methods in high-dimensional spaces. Well-known methods used in pattern recognition are reviewed in great details, such as Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA).

In Chapter 4, we introduced the clustering technique review. Since the goal of clustering is to identify the intrinsic group within the unlabeled data, we classify the clustering methods into four categories: partitioning methods, hierarchical methods, density-based methods and grid-based methods.

In Chapter 5, based on the classical unsupervised feature extraction algorithm, we proposed a new novel graph embedding method for unsupervised dimensionality reduction. We try to find an optimization method to solve it. The detail algorithm is explained and summarized.

In Chapter 6, the experimental results are implemented on both synthetic and benchmark data sets. Also, the results are compared with several other state-of-the-art feature extraction algorithms, such as PCA and LPP. Moreover, we evaluate the clustering ability of proposed method on 15 benchmark datasets. It shows that our proposed method superior than other methods, such as K-means, Ratio Cut, Normalized Cut and NMF methods.

Chapter 2

Pattern Recognition System

There are two components in conventional pattern recognition systems: one is feature analysis and another one is pattern classification. Parameter extraction and feature extraction are two steps to achieve feature analysis. We can extract the information relevant from the input data in terms of parameter vector to do pattern classification. This process is called parameter extraction; On the other hand, the parameter vector is projected to a feature vector, this process is called feature extraction. We can apply separately or combinative with parameter extraction even with classification to do feature extraction. Two classical independent feature extraction algorithms are used in dimensional reduction. One is Linear Discriminant Analysis (LDA) and another one is Principal Component Analysis (PCA) [6, 7, 8]. The main principal of PCA and LDA is that we transform the parameter vectors into a feature subspace to extract features. To complete this process we use a linear transformation matrix. However, they have different goals to optimize the transformation matrix. PCA try to get the largest variations for the original feature vector to optimize the transformation matrix. LDA aims the largest ratio of between-class variation and within-class variation when transforming the original feature space to a new feature space.

2.1 Introduction of pattern recognition system

Categorizing of input data into a number of categories or identifiable classes from a background of irrelevant detail by the means of extracting of the significant features is called pattern recognition. The above Figure 2-1 showed the four stages about a typical pattern recognition system [9].

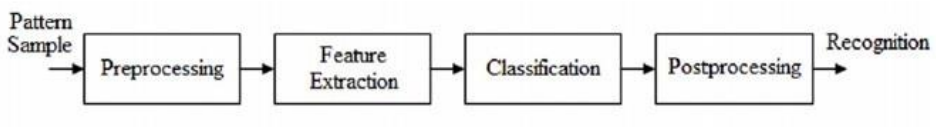

Figure 2-1 A typical pattern recognition system.

At the input of this pattern recognition system, a set of features is presented by an unknown pattern sample. On the other hand, at the output of this pattern recognition system, it is a set of predefined classes. The task of the system is to assign the unknown pattern sample to one of the classes. We will introduce and describe each stage of the pattern recognition system in the following sections. Some basic concepts of pattern recognition also are explained.

2.1.1 Pre-processing

The operations included in this stage improve the representation of the patterns. It may include data registration, noise removal, segmentation, and data normalization, depending on the nature of pattern recognition task. In face recognition system, the face images are registered so as to make sure that the eyes appear in the same coordinates of the images. Some noise which from pattern samples should be reduced in order to increase the correctness of the classification. In pattern recognition systems, the pattern which contained noise may hinder the task of the system, also it may result to the wrong underlying model. For example, filters are usually used to remove noise and enhance higher frequencies in speech recognition problems. Individual patterns should be segment in some recognition tasks. For example, the segmentation faces in an image to create meaningful patterns may require for the feature extraction process.

2.1.2 Feature Extraction

One of the most important issues of pattern recognition is how to selection of the best set of features for dimension reduction. The main goal of feature extraction is to keep as much as

possible of their discriminatory information and reduce the number of features of dataset. Therefore, the aim of a feature extractor with good performance is chooses features which are similar for data in the same class and at the same time, differentiate data in different classes.

2.1.3 Classification

The main goal of the classification is to classify the feature vector provided by the feature extractor to a class. The output of the classifier is typically a discrete selection of one of the predefined classes. All the principal components of a pattern recognition system are applied and used for improving the performance of the classifier. The degree of difficulty of the classification depends on the similarity relations between the patterns of different classes. Therefore, its success is significantly affected by the feature extraction stage.

2.1.4 Post-processing

Based on the classification outputs, we can lower the classification error rate. In order to reduce the classification error rate, this process applies previous information about the problem to achieve the goals. Therefore, the post-processing stage can increase the overall classification accuracy.

2.2 Some Concepts of Pattern Recognition

In the following section, we will introduce two important concepts. One is learning, and another one is generalization.

2.2.1 Learning Types

Many mathematical models have been applied to pattern recognition system. Typically, the mathematical model projects or classifies the patterns to the corresponding classes. Data samples are used to determine a reliable mapping. It is important part for the mapping system.

Learning or training is the procedure of getting the model, and within this procedure, sample patterns be used, which is defined as training set samples. There are three basic types of learning methods depending upon the nature of the pattern recognition task.

- Supervised Learning: For supervised learning, the training stage begins with the given class labels. Based on the given class labels information, the training stage can reduce the total cost for the training set patterns.
- Unsupervised Learning: For unsupervised learning, the training stage begins without the class label information. The main task is to find the similarities of groups which potentially have great probability belonging to same class. For supervised learning, it requires human labor for labeling, however, for unsupervised learning, it is not required human labor for labeling work. Moreover, it is widely used in many applications.
- Reinforcement Learning: For reinforcement learning, when computing the model that maps the patterns to the classes, a feedback is provided by reinforcement learning. This learning mode, the class label information is not required and instead of it, it has feedback information. The feedback information provides the fact that the tentative class is right or wrong.

2.2.2 Generalization

Training set samples is applied to learn the pattern recognition system in order to find the model that projects the pattern samples to their corresponding classifiers.

However, although the pattern recognition system is trained to maximize the performance by using the recognizing training set samples. It still may not recognize the new test samples. This is called generalization. Thus, it is easily to figure it out that generalization ability of a system is related to its performance of recognizing new samples, but not used in the learning stage. Basically, there have two reasons to cause the poor generalization ability. Firstly, pattern recognition system is intensively over-trained on the training set. Typically, we called it is an

over fitting problem. Secondly, the number of feature is really large. However, the number of training samples is small. This is also called the curse of dimensionality.

Chapter 3

Linear feature extraction methods in high-dimensional spaces

In Chapter 2, we introduced the pattern recognition system. In this chapter, we will explain the characteristic properties of high-dimensional spaces. It motivates the use of feature extraction techniques in pattern recognition tasks with high-dimensional sample spaces. Also, we will review the concept of linear feature extraction methods in detail.

3.1 Characteristic Properties of High-Dimensional Data Spaces

In many real-world applications in pattern recognition, information retrieval, the data which contained labeled information are not sufficient. And it is really heavy human labor to label a huge number of data points, also it is time-consuming work. It is really hard to get enough label information. However, this motivates a hot research direction of dimensional reduction. It is sensible to expect the data sample which contains more information in order to improve the accuracy of detecting the classes. As we discussed above, the number of feature is really large and the number of training samples is small. This concept is known as the curse of dimensionality. And a penalty in classification accuracy happened as the number of features increases beyond some point.

Experimental results have explained that high-dimensional sample spaces are usually empty because data typically focus on a certain subset of the sample space but not from the origin as the dimensional feature increases [10]. This explained that the data samples are mostly in a lower dimensionality. Thus, high-dimensional dataset can be extracted to a lower dimensionality subspace without losing important detail by separating among the classes by means of employing the knowledge of feature extraction. It has been also proved that as the dimensionality of the sample space become or close to infinity, lower-dimensional linear projections archive a normality model that explained a normality distribution with a probability approaching one. It shows that the normally distributed high-dimensional data focus on the tails

and uniformly distributed high-dimensional data pay more attention on the corners. So, density estimation work for high-dimensional sample is really a difficult work. And local neighborhoods become empty. This result the detailed density estimation is lost.

To summarize, the dimensionality of the dataset space is necessary to be reduced before applying the classifier to data samples in sample spaces which is high dimensionality data. However, an efficient dimension reduction algorithm should be explored to keep the significant information of the high-dimensional datasets. In this thesis, the dimension reduction techniques for high-dimensional sample spaces are proposed.

3.2 Introduction of Dimensionality reduction

Dimensionality reduction is the study of methods for reducing high-dimensional data to lower dimensional data. Generally, the main aim is to remove the irrelevant and redundant data to lower the cost of computation. Also it aims to avoid data over-fitting problem [11]. It can keep the data which contains important information, thus, it improve the correctness of tasks such as machine learning, pattern recognition. To solve the problem of “curse of dimensionality”, dimensionality reduction is a productive solution. The number of examples for machine learning increases exponentially as the dimension increases linearly [12]. An example of the curse of dimensionality is shown in Figure 3-1.

Consider an application in which a system dataset for processing by the term of a collection of variables. In this situation, the number of features is large and they are irrelevant. There are some factors which will cause irrelevant problem. Firstly, many dimensions will be irrelevant if the noise is larger than the variation which dimensions contained. Secondly, some dimensions will be correlated with each other, this will cause that it will be redundant. Thus, it is necessary to remove the redundant dimensions and irrelevant to reduce the nonsense part. In this way, it will become more economic.

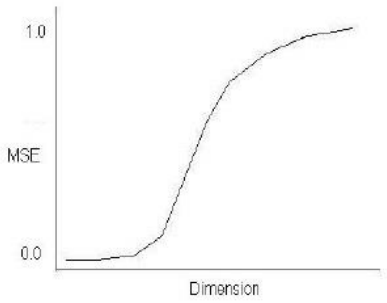

Figure 3-1 Example of curse of dimensionality

From the Figure 3-1, we know MSE is the mean squared error of an 1-nearest neighbor rule [13, 14]. Each dimension is generated uniformly from -1 to 1. From the Figure 3-1, we can get that the dimensionality increasing will lead the MSE increasing sharply.

Dimensionality reduction is a hot research topic at many areas, such as pattern recognition, artificial intelligence, statistics, databases, data mining, text mining, machine learning, visualization and optimization. Dimensionality reduction takes different effect on different areas. For example, dimensionality reduction method applied to extract a small set of features that contained most of the variability of the data in pattern recognition. However, the process of dimensionality reduction is considered as selecting a small subset of features in text mining area. Also, different dimensional reduction methods are applied in different field.

Dimensionality reduction has been a hot and popular research topic currently and there is a lot of work that has been published since decades [15, 16, 17, 18].

Based on the different ways, we can classify dimensionality reduction method in the following: (1) feature selection or feature extraction, (2) linear or nonlinear, (3) supervised or unsupervised, and (4) local or global. Typically, dimensionality reduction methods are often classified into feature selection and feature extraction. In feature selection, a subset of original features is selected to process. However, in feature extraction, it is extracted new features from the original set of features by mapping or projection.

Principal Components Analysis (PCA) is a linear algorithm. The new features are extracted from original features [19] by using a linear projecting. Likewise, nonlinear methods such as Sammon's mapping [20], locally linear embedding [21], and ISOMAP [22] apply a non-linear mapping method to extract new features from originally set of features. Moreover, supervised method use the class label information of the data. For unsupervised methods do the process with data which doesn't contain any label information.

Usually, it is useful to divided supervised dimensionality reduction algorithms into local supervised dimensionality reduction algorithms and global supervised dimensionality reduction algorithms. Features are selected for each category of the class feature in a local method whereas features are chosen for all categories in a global method. We will review the basic concepts and key techniques of feature extraction and feature selection, respectively.

3.2.1 Feature Selection

In feature selection, we can describe the process in the following:

Given a set of features $S = \{v_1, v_2, \dots, v_D\}$, the goal of feature selection is to find a subset S' of S with $|S'| = d$ such that $J(S') \geq J(T)$ for all $T \subset S$, $|T| = d$ where J is the evaluation function. Here d is usually illustrated by the user.

A feature selection algorithm requires the following ingredients: a generation strategy or search strategy, an evaluation function, a validation method [23, 24, 25, 26]. See Figure 3-2 show the feature selection process. This process explains the relationships between these components.

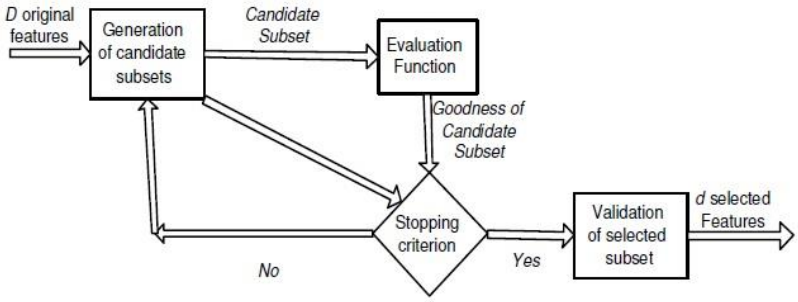

Figure 3-2 Feature selection process

The search strategy tries to find the way to select the set of features for keeping the important information. Typically, exhaustive search is prohibitive, so it is necessary to apply other strategies. The evaluation function assesses a set of features and calculates a ranking possibility for the feature selection process. The stopping criterion has less importance. Typically a certain number of features is selected to choose the stopping criterion for search procedures. Validation is for checking the validity of the selected features.

3.2.2 Feature Extraction

Feature extraction is a special form of dimensional reduction. It is very different from feature selection. Now we define the feature extraction in the following:

Given a set of feature $S = \{v_1, v_2, \ldots, v_D\}$, find a new set of feature S' which originally from a linear or non-linear mapping of S. Here, $|S'| = d$ and $J(S') \geq J(T)$ for all derived set of feature T with $|T| = d$, where J is the evaluation function.

Feature extraction is applied while the current feature has to yield new features. We can explore feature extraction by transforming any original D dimensional feature vector to a new D dimensional feature vector through mapping method. Here we can use linear mapping method or nonlinear mapping method. Generally speaking, mapping process can maintain the important information whereas reducing the feature dimensionality vector.

Figure 3-3 Feature Extraction Process

Figure 3-3 explains the feature extraction process. We will describe two methods to explain the difference between the feature extraction and feature selection.

3.3 Linear Feature Extraction Methods

Feature extraction has been one of the most significant and popular topic of pattern recognition, especially in machine learning area. Most of the feature extraction methods pay attention on getting the linear transformations. Based on the linear transformations, dimensionality reduction maps the original high-dimensional space into a lower-dimensional space which keeps most significant information. As mentioned above, the main goal of dimensionality reduction by feature extraction is that it may reduce the negative impact of the curse of dimensionality [27]. Also linear feature extractions methods are often used to process data before classification. This process can be considered as pre-processors. In the following sections we introduce these linear methods. There has some classic method such as Principal Component Analysis, Linear Discriminant Analysis and Locality Preserving Projections.

3.4 Principal Component Analysis for Feature Extraction

In this section, we will introduce a classic method for feature extraction.

3.4.1 Definition and Derivation of PCA

The main aim of PCA is to reduce the high-dimensionality data set which contains plenty of interrelated variables, at the same time, maintaining the important variation contained in the data set. In other words, the goal of PCA is to find a subspace whose basis vectors correspond

to the max-variance directions in the original space and the global scatter is maximized after the projection of datasets. The mean square of the Euclidean distance between any pair of the projected dataset characterized the global scatter. Here, let $X_{1,}X_2, \dots, X_M$ denotes a set of training dataset which are R^n dimensions. The original dataset projected training samples into subspace which is denoted by $y_1, y_2, \dots, y_M$.

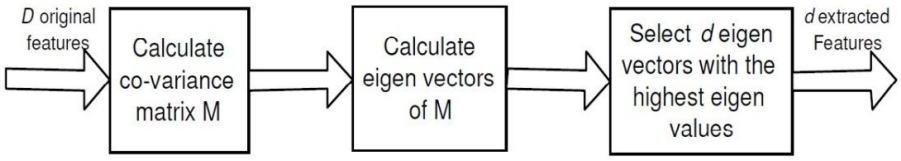

Figure 3-4 Procedures for PCA

The object function of PCA is defined by the below equation,

$$J_T(w) = \max \frac{1}{2}\frac{1}{M}\frac{1}{M}\sum_{i=1}^{M}\sum_{j=1}^{M} \| y_i - y_j \|^2 \quad (Eq. 3.1)$$

Where $y_i = w^T x_i, i = 1, \dots, M$. After deduction, the above equation can be rewritten as

$$J_T(w) = \max tr(w^T S_T w) \quad (Eq. 3.2)$$

Here, we define S_T as below:

$$S_T = \frac{1}{2}\frac{1}{M}\frac{1}{M}\sum_{i=1}^{M}\sum_{j=1}^{M}(x_i - x_j)(x_i - x_j)^T \quad (Eq. 3.3)$$

The above equation explains that S_T is intrinsically the covariance matrix of dataset. The Lagrange multiplier is applied to $J_T(w)$. We can easily get the optimized solution $S_T w_i = \lambda w_i$. Therefore, the projection matrix w which maximizing $J_T(w)$ can be choosed as the eigenvectors of the largest eigenvalues of S_T. At the same time, we can choose a set of axes of PCA by using the d eigenvectors of S_T corresponding to the d largest eigenvalues.

3.5 Linear Discriminant Analysis for Feature Extraction

Unlike PCA, the main aim of Linear Discriminant Analysis(LDA) is to find the optimal subspace that can maximize between class scatter and minimize the within class scatter simultaneously. The fisher criterion is defined as follows:

$$J_F(w) = \max \frac{|w^T S_B w|}{|w^T S_W w|} \quad (Eq. 3.4)$$

Where S_W is the within-class scatter matrices and S_B is the between-class scatter matrices. Here $S_B + S_W = S_T$ and both are semi-positive definite matrix. The optimized solution of $J_F(w)$ are the generalized eigenvectors $w_1, w_2, \dots, w_d$ of $S_B w = \lambda S_W w$ corresponding to the d largest eigenvalues.

3.5.1 Linear Discriminant Analysis Review

Denote $x_i \in \mathcal{R}^d, (i = 1,2, \dots, n)$ be d-dimensional data and $l_i \in \{1,2, \dots, c\}$ be associated class labels, where n is the number of data and c is the number of classes. Let n_i be the number of data in the class i. LDA is to learn a linear transformation $W: \mathcal{R}^d \to \mathcal{R}^m$, and $W \in \mathcal{R}^{d \times m}$. After the linear transformation, the original high-dimensional data x is transformed into a low-dimensional vector:

$$y = W^T x \quad (Eq. 3.5)$$

In the following, we define two scatter matrix, the within-class scatter matrix is S_w:

$$S_w = \sum_{i=1}^{c} \sum_{j:l_j=1} (x_j - m_i)(x_j - m_i)^T \quad (Eq. 3.6)$$

And the between-class scatter matrix S_b is defined as:

$$S_b = \sum_{i=1}^{c} n_i (m_i - m)(m_i - m)^T \quad (Eq. 3.7)$$

Where $m_i, (i = 1, 2, \dots, c)$ is the mean of the samples in class i and m is the mean of all the samples:

$$m_i = \frac{1}{n_i} \sum_{j:l_j=1} x_j \quad (Eq. 3.8)$$

$$m = \frac{1}{n} \sum_{j=1}^{n} x_j \quad (Eq. 3.9)$$

The LDA is to maximize the between-class scatter and minimizing the within-class scatter. We get the projection matrix W^* in LDA through solving the following optimization problem:

$$W^* = arg \max_{W \in \mathcal{R}^{d \times m}} tr((W^T S_W W)^{-1} W^T S_b W) \quad (Eq. 3.10)$$

Where $tr(.)$ denotes the trace operator. The solution of the optimization problem is the m largest eigenvectors of $S_w^{-1} S_b$, and the optimal value is $\sum_{i=1}^{m} \lambda_i$, where $\lambda_i, (i = 1,2, \dots, m)$ are the first m largest eigenvalues of $S_w^{-1} S_b$, and m is the projection dimensionality. Figure 3-5 shows the procedures for LDA.

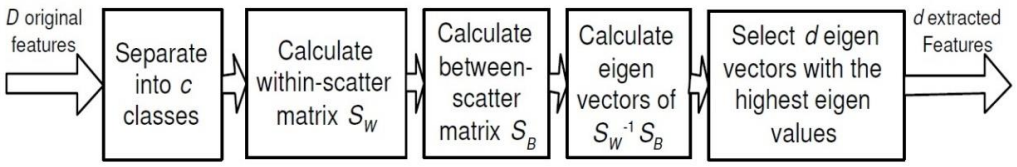

Figure 3-5 Procedures for LDA

From the above, we can notice that the two drawback in LDA. The first one is that it cannot be solved numerically when S_w is singular. The second one is the optimal value monotonously increase according to the increasing of the projection dimensionality. Hence, the optimal dimensionality for discriminant analysis cannot be determined in LDA. Another limitation of LDA is that the projection directions in LDA are smaller than the number of class, which is not sufficient for complicate problem.

3.6 Locality Preserving Projections Review

In this section, we will review the classic method, Locality Preserving Projections (LPP), which is used for feature analysis on many applications. Typically, LPP would get good

performance when apply on the data which is embedded in a nonlinear manifold space. Since LPP is considered as a method that is good at recognize lower dimensional spaces that maintain local relationships on data vectors in the transformed space.

3.6.1 Locality Preserving Projections

LPP extend the local mutual relationships between the input data vectors to the vectors of the projected subspace. This is the optimality criterion of LPP. The following equation will define as:

$$D_N = \min \sum_{i,j} (\vec{y}_i - \vec{y}_j)'(\vec{y}_i - \vec{y}_j) \quad s_{i,j} \qquad (Eq. 3.11)$$

In Equation 3.11, we define the similarity matrix, $= \{s_{i,j}\}_{N \times N}$, which denote the local relationships of the input data vectors as follows:

$$s_{i,j} = \begin{cases} \exp\left(-\dfrac{\| \vec{x}_i - \vec{x}_j \|^2}{\rho}\right), e(\vec{x}_i, \vec{x}_j) = 1 \\ 0, \qquad e(\vec{x}_i, \vec{x}_j) = 1 \end{cases} \qquad (Eq. 3.12)$$

In Equation $Eq. 3.12$, $e(\vec{x}_i, \vec{x}_j)$ is the indicator function, where $\vec{x}_i$ and $\vec{x}_j$ are neighbors and ρ is the heat kernel factor. Here $\vec{x}_i$ is the neighborhood of a given input vector. It can be considered as the K-nearest vectors to $\vec{x}_i$. Furthermore, it also can considered as the set of vectors that fall within a maximum distance defined by threshold ε from $\vec{x}_i$.

Here $y_i = \vec{w}'\vec{x}_i$ is one-dimensional representation of original feature $\vec{x}_i$, and we use y_i to substitute $\vec{y}_i$. So the optimization criterion in Equation 3.11 can be rewritten as follow:

$$\frac{1}{2} D_N = \frac{1}{2} \sum_{i,j} \left(y_i - y_j\right)^2 s_{i,j}$$

$$= \frac{1}{2} \sum_{i,j} \left(\vec{w}'\vec{x}_i - \vec{w}'\vec{x}_j\right)^2 s_{i,j}$$

$$= \sum_i \vec{w}'\vec{x}_i (\sum_j s_{i.j})\, \vec{x}_i'\vec{w} - \sum_{i,j} \vec{w}'\vec{x}_i\, (s_{i,j}) \vec{x}_i'\vec{w}$$

$$= \vec{w}' XLX' \vec{w} \qquad (Eq. 3.13)$$

Where $L = C - S$ is the Laplacian matrix. The matrix C is a diagonal matrix whose entries are the column sums of S, $c_{i,i} = \sum_j s_{i,j}$. Here we apply a constraint on the magnitude of the transformed vectors to achieve an unique solution.

$$\vec{w}' XCX' \vec{w} = 1 \quad (Eq. 3.14)$$

In order to minimize the objective function given in $Eq. 3.13$, a constraint is imposed as follows:

$$XLX' \vec{w} = \lambda XCX' \vec{w} \quad (Eq. 3.15)$$

From $Eq. 3.15$, we can get the linear project matrix W from the eigenvectors associate with d_l smallest non-zero eigenvalues.

Chapter 4

Clustering Technique Review

Clustering is one of the important topics in machine learning and data mining. The main goal of clustering is to get the similar patterns into the same cluster and reveal the significant and meaningful structure of the data [28].

Clustering is one of the most significant unsupervised learning problems in machine learning area. It aims to find a structure in a set of unlabeled data. Clustering can be considered as the process of organizing objects into similar group or class. Thus, a cluster is a collection of objects which contain similar information with each other. That is to say, the goal of clustering is to identify the intrinsic group within the unlabeled data.

4.1 Major clustering methods

Generally speaking, we classify the major clustering methods into four categories. There are partitioning methods, hierarchical methods, density-based methods, and grid-based methods. We will introduce these methods in the following sections.

4.1.1 Partitioning methods

Given a set of n objects, a partitioning method divides the data into k groups. Typically, each group must belong to exactly one group or class. This idea called exclusive cluster separation. Generally speaking, the basic criterion of a good partitioning is that the objects in the same cluster are related between each other. Oppositely, objects in different clusters are far away. In order to achieve global optimality in partitioning-based clustering, it computes prohibitively and requires an exhaustive of all the possible partitions. Thus, it is much better to use popular heuristic methods, such as greedy methods like k-means and the k-medoids method. In this way, it improves the clustering quality and reaches a local optimum.

4.1.2 Hierarchical methods

A hierarchical method creates a hierarchical decomposition of the given set of data objects. Based on the way how hierarchical decomposition formed, it can be classified into agglomerative or divisive. The former method, also called bottom-up method, starts with each object forming a separate group. It merge two group into one group based on the close groups. The divisive approach, also called the top-down approach, starts with all the objects which is in the same cluster. Then the cluster is separated into smaller ones in iteration until a termination condition reached.

However, hierarchical clustering methods has limited that it cannot be undone once merging or splitting is done. This limited fact is helpful since it save computation costs and there is no combinatorial number of different choices problem.

4.1.3 Density-based methods

Typically, the basic idea of partitioning methods is to cluster data point based on the distance between data point. It has good performance on getting spherical-shaped clusters, and this will suffer difficulty in searching clusters of arbitrary shapes. There has other clustering methods which is based on the notion of density in the nearby objects exceeds some threshold. However, density-based methods usually divide a set of objects into multiple exclusive clusters. Generally, density-based methods concentrate exclusive clusters only and fuzzy cluster cannot be considered.

4.1.4 Grid-based methods

Grid-based methods split the object space from a grid structure into a finite small cell. All the clustering operations are performed on the grid structure. The processing time is fast based on grid-based method since it is independent on the number of data objects whereas it only depends on the number of cells in each dimension in the quantized space. It is an efficient

method for many spatial machines learning problem, especially on clustering. Therefore, grid-based methods can be combined with other clustering methods.

Chapter 5

Discriminative Unsupervised Dimensionality Reduction

In this section, we propose a new novel graph embedding method for unsupervised dimensionality reduction. Typically, there has a lot of work for combination of dimensionality reduction and classification. In our study, we explored dimensionality reduction with clustering.

In the previous chapter, we already review the background of dimensional reduction on different area. Such as data mining, statistic, image processing, pattern recognition, etc. In this chapter, we focus on explain dimensional reduction applied on machine learning.

5.1 Introduction of dimensional reduction on machine learning

With the rapid growth of sciences recently, high-dimensional data becomes crowed everywhere and common nowadays. Moreover, these data are literally characterized by an underlying low-dimensional space mostly. This interesting phenomenon draws high attention to the dimensionality reduction technique which become really hot and popular research topic recently. In this situation, high dimensional data needs dimensional reduction techniques to discover important information and knowledge from it by removing the useless feature whereas maintain the significant feature which contain more information.

Dimensionality reduction is a popular technique to discover the intrinsic manifold structure from the high dimensional data. As we mentioned above, dimensionality reduction is one of the most popular topics in machine learning and is utilized in numerous areas. Moreover, it contributes to some specific area, such as computer vision, biology and geosciences. In computer vision, it projects the images from high dimensionality space to low dimensional space, which is considered as a basic pre-processing. It can be applied to image recognition [29, 30], image segmentation [31] and image compression [32]. In biological researches, dimensionality reduction is typically adopted to study high dimensional gene data, such as gene classification

[33] and disease causing genes interaction analysis [34]. Moreover, dimensionality reduction methods can be employed in geosciences so as to deal with the global climate data [35].

We already introduced the supervised dimensionality reduction methods and unsupervised dimensionality reduction methods in the above chapter. We explained some classic method like PCA (Principal Component Analysis) and LDA (Linear Discriminant Analysis). We can figure it out that the unsupervised dimensionality reduction methods are more favorable for some data. Moreover, the graph embedding method is emphasis on in many unsupervised dimensionality reduction methods.

In this paragraph, we will introduce graph embedding method. We all know that the most of state-of-the-art graph embedding dimensionality reduction methods use an affinity graph constructed. This process makes the projecting process based on the input of the graph to a large extent.

In our study, we explore a new unsupervised dimensionality reduction method of a novel graph embedding. In this method, it is not required of input of the graph. Moreover, the graph construction in our model is constructed with the dimensionality reduction at the same time. The adaptive and optimal neighbors are assigned to improve the method based on the projected local distances. In this method, we assume that a larger probability to be connected happens on the data with lower distance apart. Moreover, the learnt graph to an ideal structure is constrained that the graph is block diagonal with the number f connected components to be identity with the number of clusters in the data.

In the following section, we will explain the proposed method and implement it. Also, we do some analysis and conclusion.

5.2 Introduction of Proposed Method

In this section, we will introduce our proposed algorithm.

5.2.1 The Algorithm

First of all, here we summarize the notations used in the following section. Matrices are all written as uppercase letters whereas vectors are written as bold lower case letters. We define a matrix M, $M \in \mathbb{R}^{d\times n}$, this matrix is i-th row, j-th column and ij-th element are denoted by m^i, m_j and m_{ij} respectively. The trace of M is denoted by $Tr(M)$. The Frobenius norm of $M$is defined as $\| M \|_F = \sqrt{\sum_{i=1}^{d}\sum_{j=1}^{n} m_{ij}^2} = \sqrt{\sum_{i=1}^{d} \| m^i \|_2^2}$. For an vector $V \in \mathbb{R}^n$, when $p \neq 0$, l_p-norm $\| V \|_p$is defined as $\| V \|_p = (\sum_{i=1}^{n} |v_i|^p)^{\frac{1}{p}}$.

5.2.2 Graph Embedding Discriminative Unsupervised Dimension Reduction

In this section, we will explain our proposed method. In this method, we apply graph construction embedded in the unsupervised dimensionality reduction. Given data set $X \in \mathbb{R}^{d\times n}$, suppose that we are trying to learn the projection matrix $W \in \mathbb{R}^{d\times m}$. Usually, here an orthogonal constraint should be applied on W, it is $W^TW = I$. Based on the distance formula, we can get the distance between the i-th and j-th data points would be $\| W^T\mathrm{X}_i - W^TX_j \|_2^2$ and we summary all the distance between all pairs of data points. It would be

$$\sum_{i}^{n}\sum_{j}^{n} \| W^TX_i - W^TX_j \|_2^2 = Tr(W^TXHX^TW) \quad (1)$$

Notice that here is H, H is the centering matrix, we define H as follow:

$$H = I - 11^T \quad (2)$$

Traditional dimensionality reduction methods usually solve the problem that requires a graph constructed before hand:

$$\min_{W^TW=I} \frac{\sum_{i,j=1}^{n} \| W^TX_i - W^TX_j \|_2^2 S_{ij}}{Tr(W^TXHX^TW)} \quad (3)$$

We can see from the above formula, these algorithm separate the dimensionality reduction and graph construction, therefore, it dependent on the input of the graph. In this thesis, we develop a novel graph embedding dimensionality reduction method. Now we describe this process using two steps.

Now we consider an affinity matrix S to denote the probability of each data point in X to connect with its neighbors. We define$S \in \mathbb{R}^{n\times n}$. Typically, we assign a pair which has smaller distance with a larger probability. Therefore, we can solve the following problem to construct the graph.

$$\min_{S,W} \frac{\sum_{i,j=1}^{n} \| W^T X_i - W^T X_j \|_2^2 s_{ij}}{Tr(W^T X H X^T W)} \quad (4)$$

$$s.t. \forall_i, s_i^T 1 = 1, 0 \le s_{ij} \le 1, W^T W = I,$$

However, Problem (1) has a trivial solution that only the nearest data point is assigned to a probability whereas all others are assigned 0. In this problem, they don't have any neighbors for data point x. In this case, we apply some constraints on the graph in order to make the graph structure more clearly. Suppose that the number of clusters is k, and the graph S has n data points. Our main aim is that the block diagonal has exactly k connected components. That is to say, it is ideal if the probability within cluster should be nonzero whereas the probability between clusters should be zero. Also, for the probability within a same cluster, it should be equally distributed. However, it is really difficult to achieve it ideally. In this chapter, we develop a novel and simple method to achieve this challenge.

We assign a function value $f_i \in \mathbb{R}^{1\times k}$ to each node i, then we can know, $F \in \mathbb{R}^{n\times k}$, so, we can get the following formula:

$$\sum_{i,j} \| f_i - f_j \|_2^2 s_{ij} = 2Tr(F^T L_s F) \quad (5)$$

Where $L_S = D_S - \frac{S^T+S}{2}$ is the Laplacian matrix in graph theory. Here we define a diagonal matrix as the degree matrix $D_s \epsilon \mathbb{R}^{n\times n}$. For the diagonal matrix, the i-th diagonal element is $\sum_j (s_{ij} +$

$s_{ji})/2$. The Laplacian matrix has an important property when the probability matrix S is nonnegative. The property is described as follows [36, 37].

Theorem 1 The multiplicity k of the eigenvalue 0 of the Laplacian matrix $\mathrm{L_s}$ is equal to the number of connected components in the graph associated with S.

This theorem indicates that the graph possesses an ideal structure if $r(L_S) = n - k$. This ideal structure that we described above could exactly partition the data points into k clusters since this is a block diagonal structure. Therefore, this problem becomes:

$$\min_{S,W} \frac{\sum_{i,j=1}^{n} \| W^T X_i - W^T X_j \|_2^2 s_{ij}}{Tr(W^T X H X^T W)} + \gamma s_{ij}^2 \qquad (6)$$

$$s.t. \forall_i, s_i^T 1 = 1, 0 \le s_{ij} \le 1, W^T W = I, rank(L_S) = n - k$$

From problem (6) we can see that it is hard to solve it if there is a strict constraint on the rank. Based on this problem, we develop a novel algorithm to solve it.

5.2.3 Optimization Algorithm for Problem (6)

For the above problem, $\sigma_i(\mathrm{L_S})$ is the i-th smallest eigenvalue of $\mathrm{L_S}$. We can figure it out that $\sigma_i(\mathrm{L_S}) \ge 0$ since $\mathrm{L_S}$ is positive semi-definite. Thus, the above problem would be equivalent to the following problem if the λ is large enough.

$$\min_{S,W,F} \frac{\sum_{i,j=1}^{n} \| \mathrm{W^T X_i - W^T X_j} \|_2^2 \mathrm{s_{ij}}}{\mathrm{Tr(W^T X H X^T W)}} + \gamma \mathrm{s_{ij}^2} + 2\lambda \sum_{i=1}^{k} \sigma_i(\mathrm{L_S}) \qquad (7)$$

$$\mathrm{s.t.} \forall_i, \mathrm{s_i^T 1}, 0 \le \mathrm{s_{ij}} \le 1, \mathrm{W^T W = I}, \mathrm{F} \in \mathcal{R}^{n \times k}, \mathrm{F^T F = I}$$

In order to make sure that the k smallest eigenvalues of $\mathrm{L_S}$ are zero and the rank of $\mathrm{L_S}$ is $\mathrm{n - k}$. According to the Ky Fan's Theorem [38], we have

$$\sum_{i=1}^{k} \sigma_i(\mathrm{L_S}) = \min_{F \in R^{n \times k}, F^T F = I} \mathrm{Tr(F^T L_S F)} \qquad (8)$$

From Problem (8), we can easily get the following equation:

$$\min_{S,W,F} \frac{\sum_{i,j=1}^{n} \| \mathrm{W^T X_i - W^T X_j} \|_2^2 \mathrm{s_{ij}}}{\mathrm{Tr(W^T X H X^T W)}} + \gamma \mathrm{s_{ij}^2} + 2\lambda \mathrm{Tr(F^T L_S F)} \qquad (9)$$

$$s.t.\ \ \forall_i, s_i^T 1, 0 \le s_{ij} \le 1, W^T W = I, F \in \mathcal{R}^{n\times k}, F^T F = I$$

The Problem (9) can be optimization method to solve.

The first step: fix , S , try to solve F. Then Problem (9) becomes:

$$\min_{F\in\mathcal{R}^{n\times k}, F^T F=I} \mathrm{Tr}(F^T L_S F) \qquad (10)$$

The optimal solution of F in above Problem can be solved by the k eigenvectors corresponding to the k smallest eigenvalues of L_S.

Instead of fixing W, S, the second step is to fix S, F and try to solve W. Then Problem (9) is written in the following form:

$$\min_{W^T W=I} \frac{\sum_{i,j=1}^{n} \| W^T X_i - W^T X_i \|_2^2 s_{ij}}{\mathrm{Tr}(W^T X H X^T W)} \qquad (11)$$

Then, we rewrite as follow:

$$\min_{W^T W=I} \frac{\mathrm{Tr}(W^T L_S X^T W)}{\mathrm{Tr}(W^T X H X^T W)} \qquad (12)$$

We can solve W by a iterative re-weighted method. Then, the Lagrangian function becomes:

$$\mathcal{L}(W, \Lambda) = \frac{\mathrm{Tr}(W^T X L_S X^T W)}{\mathrm{Tr}(W^T X H X^T W)} - \mathrm{Tr}\big(\Lambda(W^T W - I)\big) \quad (13)$$

Then take derivative $w.r.t.W$ and let it be zero, we can get:

$$\left(X L_S X^T - \frac{\mathrm{Tr}(W^T X L_S X^T W)}{\mathrm{Tr}(W^T X H X^T W)} X H X^T\right) W = \Lambda W \qquad (14)$$

Next, we get the following formula for solving the Problem (14):

$$\left(X L_S X^T - \frac{\mathrm{Tr}(W^T X L_S X^T W)}{\mathrm{Tr}(W^T X H X^T W)} X H X^T\right) \qquad (15)$$

So, we can update W iteratively.

One more interesting step is to fix W, F and solving S.Then problem (9) become:

$$\min_{\forall_i, s_i^T 1=1, 0\le s_{ij}\le 1} \frac{\sum_{i,j=1}^{n} \| W^T X_i - W^T X_j \|_2^2 s_{ij}}{\mathrm{Tr}(W^T X H X^T W)} + \gamma s_{ij}^2 + \lambda \sum_{i,j=1}^{n} \| f_i - f_j \|_2^2 s_{ij} \quad (16)$$

The above problem can be solved separately for each s_i as below:

$$\min_{s_i} \sum_{j=1}^{n} (d_{ij}^{wx} s_{ij} + \gamma s_{ij}^2 + \lambda d_{ij}^{f} s_{ij}) \quad (17)$$

$$s.t. s_i^T 1 = 1, 0 \le s_{ij} \le 1$$

Here, $d_{ij}^{wx} = \frac{\|W^T x_i - W^T x_j\|_2^2}{Tr(W^T X H X^T W)}$ and $d_{ij}^{f} = \| f_i - f_j \|_2^2$.

The problem above can be rewritten as below:

$$\min_{s_i^T 1 = 1, 0 \le s_i \le 1} \| s_i + \frac{1}{2\gamma} d_i \|_2^2 \quad (18)$$

Here $d_{ij} = d_{ij}^{wx} + \lambda d_{ij}^{f}$. Based on d_{ij}, we can update s_i. By following the three steps which we described above, it is easily to update F, W and S accordingly.

For the problem (9), we can summary it as below:

Input:

Data matrix $X \in \mathbb{R}^{d \times n}$, number of cluster k, reduced dimension m, parameter γ, a large enough λ.

Output:

Projection $W \in \mathbb{R}^{d \times m}$ and probability matrix $S \in \mathbb{R}^{n \times n}$ with exactly k connected components.

Initialize S by the optimal solution to the problem (9) without the constraint on $\mathrm{rank}(L_S)$, while not converge do

1. Update $L_S = D_S - \frac{S^T + S}{2}$, where $D_S \in \mathbb{R}^{n \times n}$ is a diagonal matrix with the i-th diagonal element as $\sum_j (s_{ij} + s_{ji})/2$.
2. Update F, whose columns are the k eigenvectors of L_S corresponding to the k smallest eigenvalues. Update W, whose columns are the m eigenvectors of matrix in Eq. (15) corresponding to the m smallest eigenvalues. Update W iteratively until converges.
3. For each i, update the i-th row of S by solving the problem (18).

   End while

Chapter 6

Experiments on synthetic data and real-world benchmark data sets.

In this chapter, we will apply our proposed dimensionality reduction method on both synthetic data and benchmark data sets. We denote our proposed Discriminative Unsupervised Dimensionality Reduction as DUDR in this chapter.

6.1 Experiments on Synthetic Datasets

The synthetic data in this experiment is the data set that is stochastically generated two-Gaussian matrix. Based on the Gaussian distribution, we randomly generate two clusters of data. Our main aim is to find an effective projection matrix which makes the two clusters could be separated sharply. In this experiment, we also apply some other popular methods, PCA and LPP, on the dataset to compare the result with our proposed method DUDR. Figure 6-1, Figure 6-2, and Figure 6-3 show the comparison results about it. We can see from the Figure 6-1 that it is easily to get a good project direction if these two clusters are far from each other.

However, the experiment result changed as the distance between the two clusters become close. As we seen, PCA becomes incompetent whereas LPP lose the way to achieve the projection aim. However, our proposed method DUDR method consistently works well even as the distance between the two clusters gets closer. In the previous chapter, we already reviewed the PCA and LPP. PCA is a method is good at performance on the global structure. PCA is not capable of distinguishing one cluster from the other, that's why it becomes incompetent immediately. For LPP, it works worse also when two clusters get too close from each other since it is a method that focused on the local structure. However, our proposed method DUDR pays more attention on the discriminative structure. This improves its projection ability. Figure 6-1, Figure 6-2 and Figure 6-3, these three figures are projection results on the two-Gaussian synthetic data.

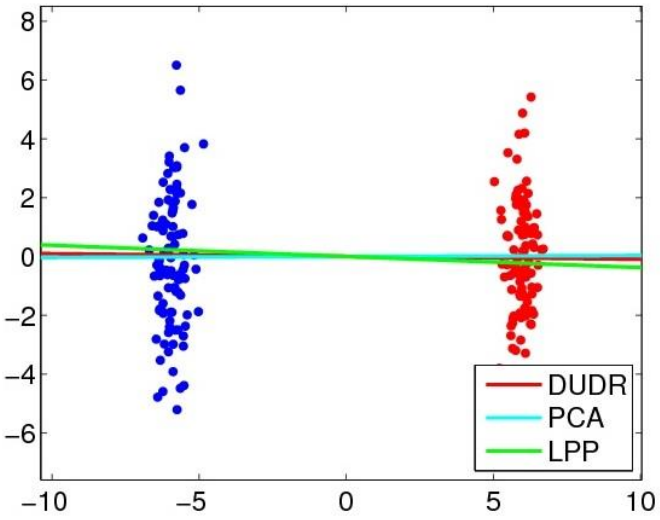

Figure 6-1 Cluster Far Away

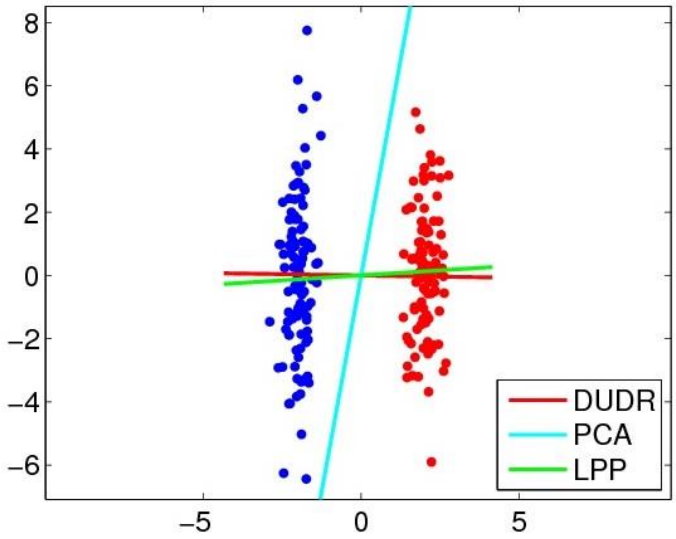

Figure 6-2 Clusters Relatively Close

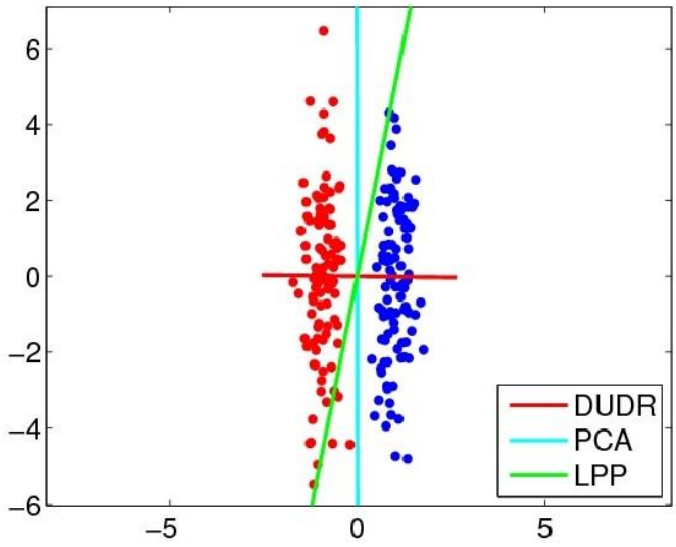

Figure 6-3 Clusters Fairly Close

6.2 Experiments on Real Benchmark Datasets

In this Chapter, we will show the experiment results on real benchmark datasets. The benchmark datasets we used contains: Ecoli, Pathbased, Aggregation, BreastCancer, Compound, Yeast, R15, Glass, Spiral, Abalone, Movements, Jaffe, AR_ImData, XM2VTS, and Coil20. Five of them are shape set data, six of them are data sets from UCI Machine Learning Repository and the other four are image data sets. We summarize these 15 datasets in Table 1.

Table 6-1 Description of data sets

Data sets	# of Instances	Dimensions	Classes
Ecoli	336	7	8
Pathbased	300	2	3
Aggregation	788	2	7
BreastCancer	683	9	2
Compound	399	2	6
Yeast	1484	8	10
R15	600	2	15
Glass	214	9	6
Spiral	312	2	3
Abalone	4177	8	28
Movements	360	90	15
Jaffe	213	1024	10
AR_ImData	840	768	120
XM2VTS50	1180	1024	295
Coil20	1440	1024	20

In this section, we will apply the 15 datasets to test both the projection and clustering ability of DUDR.

6.2.1 Experiments on Projection

We test our proposed method DUDR on the 5 benchmark data sets such as AR_ImData, Movements, Coil20, Jaffe and XM2VT. Similarly, we tested our proposed method DUDR and compared DUDR with PCA method and LPP method in this experiment.

In our experiment, we firstly test these three methods, DUDR, PCA and LPP, and then we learned the projection matrix separately. Based on the results we get, we will apply clustering method, K-means for 100 times with the same initialization. From the 100 runs, we choose the best clustering result.

For LPP method, it requires an affinity matrix constructed beforehand. Based on this, we construct the graph with the self-tune Gaussian method [39]. In this experiment, we set the parameter σ to be self-tuned and we set 5 neighbors for testing.

For our proposed method, DUDR, we also set the parameter λ to be self-tuned. Firstly, we compute the number of zero eigenvalues in each iteration. Here when the number of zero

eigenvalues is larger than k, we divide λ by 2. Whereas the number of zero eigenvalues is smaller than k then we do multiply λ by 2. Otherwise we stop the iteration process. We set the number of projected dimensions from 1 to 100. However, the dimension of the data set Movement is too large and we test the performance with dimensions from 1 to 16.

In this experiment, we get the report of the projection results which shown in Figure 6-4~ Figure 6-8. From those figures, we can verify the projection ability of our proposed method. Obviously, for different circumstances, our proposed DUDR method, it has good performance than PCA and LPP. We can get from the result that DUDR outperforms than PCA and LPP especially in the situation that the number of projected dimension is small. The proposed method is capability of projecting the original data set to a subspace which has small dimensions $k-1$, where k is the number of clusters in the data set. Therefore, the DUDR is able to project the data to a lower dimensional space after clustering. It is apparently to get that the clustering process is important and it makes the dimensionality reduction process more efficient and effective.

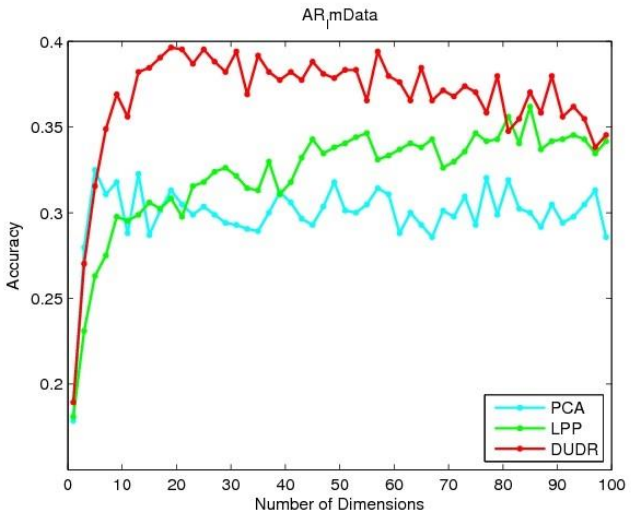

Figure 6-4 Projection results on AR-mData data sets

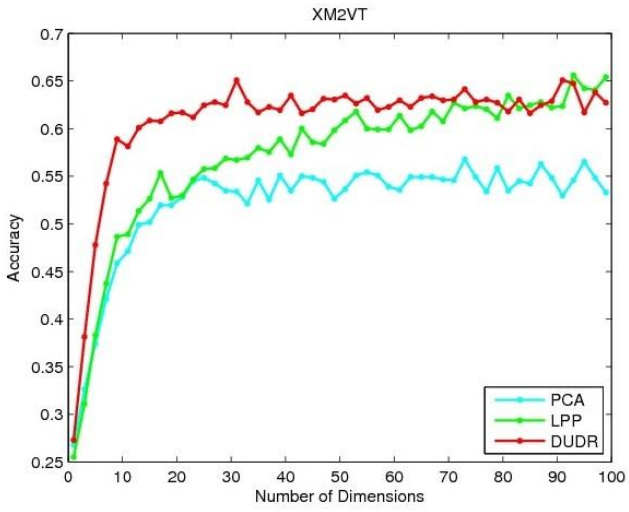

Figure 6-5 Projection results on XM2VT

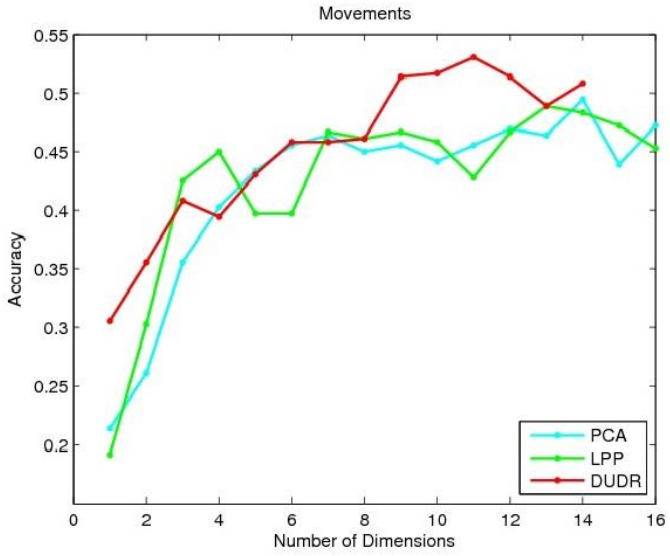

Figure 6-6 Projection results on Movements

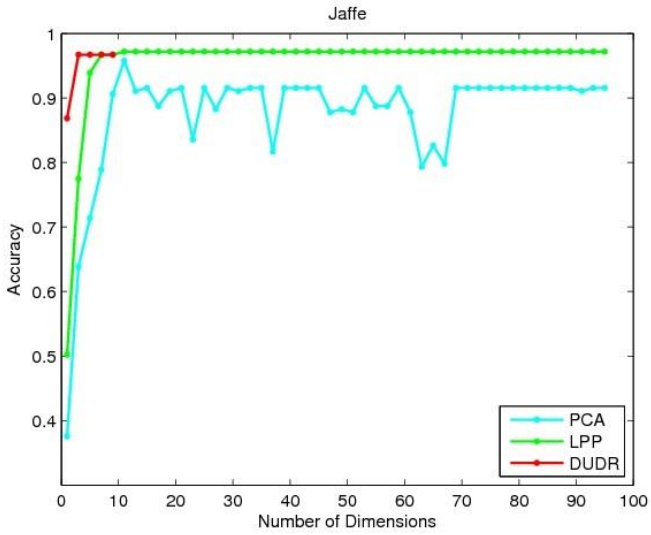

Figure 6-7 Projection results on Jaffe

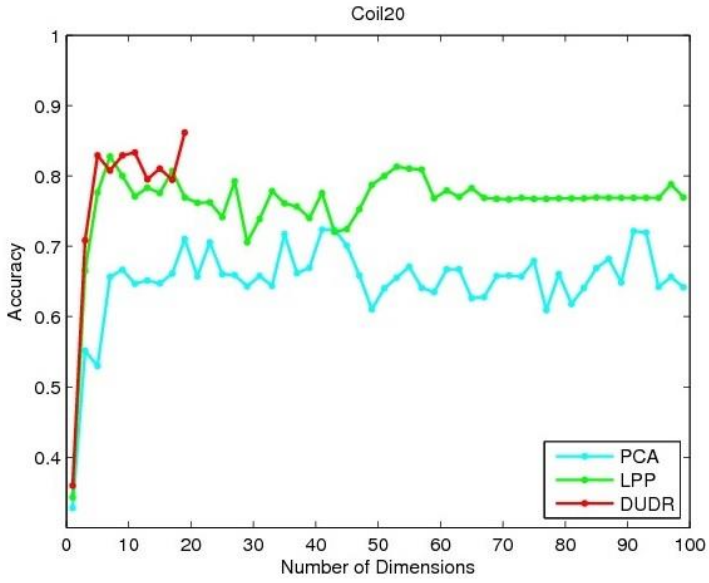

Figure 6-8 Projection results on Coil20

6.2.2 Experiments on Clustering

In this section, we will evaluate the clustering ability of proposed method, DUDR, on all the benchmark datasets. Also, we will compare DUDR with other methods, such as K-means, Ratio Cut, Normalized Cut and NMF methods.

Firstly, we set the number of clusters is k in each data set and the projected dimension in DUDR is $k-1$. Similarly, these methods such as Ratio Cut, Normalized Cut and NMF, all require an affinity matrix as an input. The graph is constructed with the self-tune Gaussian method. And then we run k-means for 100 times with the same initialization for all the methods such as K-means, Ratio Cut and Normalized Cut. After applying the method, we get their average performance, standard deviation and the performance corresponding to the best K-means objection function value whereas for NMF and DUDR, we run once and get the results. We use two clustering metrics to evaluate the experiment result. One is accuracy, another one is NMI (Normalized Mutual Information). We summarize the result in Table 2 and Table 3. From both table, we can easily get that DUDR has better performance than other related methods on the benchmark data sets. Because DUDR run less iteration than other methods (K-means, Ratio Cut and Normalized Cut), it is less time consumed but has a better accuracy. For NMF, it requires a graph constructed beforehand. Moreover, in most cases DUDR is more steady in a certain setting and independent on the initialization than other methods.

Table 6-2 Clustering accuracy on 15 benchmark data sets

	K-Means		RatioCut		NormalizedCut		NMF	DUDR
	%(min_obj)	Average%	%(min_obj)	Average%	%(min_obj)	Average%		
Ecoli	62.80	57.62±5.09	59.23	54.21±4.44	57.44	53.10±4.52	62.80	**83.04**
Pathbased	74.33	74.24±0.97	77.67	77.67±0.00	77.67	77.67±0.00	78.00	**87.00**
Aggregation	85.79	76.90±6.13	99.49	90.04±8.02	99.49	88.18±9.78	98.86	**99.75**
BreastCancer	96.05	96.18±0.05	63.84	63.84±0.00	63.98	63.88±1.01	51.83	**97.36**
Compound	69.42	63.93±10.66	53.63	53.12±4.48	53.13	52.64±3.56	52.38	**80.20**
Yeast	40.70	38.34±2.42	41.44	38.26±2.07	40.03	36.88±2.70	38.01	**50.27**
R15	99.67	82.59±7.54	91.83	83.20±6.68	**99.50**	81.58±7.02	89.83	99.67
Glass	43.46	44.65±3.37	38.79	38.19±1.99	37.38	38.39±2.18	37.85	**48.60**
Spiral	33.97	34.54±0.29	99.68	98.10±7.80	99.68	97.36±9.60	91.03	**100.00**
Abalone	15.44	16.76±0.79	14.32	14.05±0.55	14.53	14.11±0.53	16.61	**22.34**
Movements	45.28	44.24±2.19	45.83	45.79±2.33	45.56	45.10±2.08	46.11	**51.11**
Jaffe	91.08	74.83±8.36	**96.71**	85.17±7.38	**96.71**	80.76±8.29	**96.71**	**96.71**
AR_ImData	28.57	27.43±1.03	34.88	35.32±0.75	36.19	36.54±0.77	37.14	**39.05**
XM2VTS50	51.78	48.47±1.21	57.80	57.44±0.90	65.51	64.77±1.12	67.80	**68.64**
Coil20	65.83	56.16±4.85	78.75	70.73±4.49	79.38	71.43±4.81	70.42	**82.99**

Table 6-3 Clustering NMI on 15 benchmark data sets

	K-Means		RatioCut		NormalizedCut		NMF	DUDR
	%(min_obj)	Average%	%(min_obj)	Average%	%(min_obj)	Average%		
Ecoli	53.44	53.44±2.75	51.26	49.13±2.63	51.56	49.86±2.96	55.87	**72.20**
Pathbased	51.28	51.17±1.29	55.16	55.16±0.00	55.16	55.16±0.00	52.51	**75.63**
Aggregation	79.26	79.10±3.42	98.35	91.39±5.73	98.35	90.93±6.09	96.87	**99.07**
BreastCancer	74.29	74.93±0.24	0.78	0.78±0.00	0.69	0.72±0.37	12.58	**82.58**
Compound	69.68	69.60±6.18	73.37	70.67±4.92	73.34	70.49±4.46	73.26	**79.27**
Yeast	26.17	25.31±0.99	27.95	25.18±1.29	25.36	23.79±1.43	25.13	**30.30**
R15	**99.42**	92.08±3.64	95.68	92.09±2.96	99.13	92.03±2.98	95.28	**99.42**
Glass	**33.25**	32.90±3.23	33.15	29.30±2.96	28.19	29.46±2.50	29.05	28.05
Spiral	0.04	0.05±0.02	98.35	96.43±9.46	98.35	95.88±11.69	75.95	**100.00**
Abalone	15.45	**15.66±0.27**	15.12	14.67±0.44	15.58	14.78±0.40	14.97	15.64
Movements	58.50	57.35±1.84	60.64	61.45±2.17	60.89	59.89±1.95	64.08	**64.53**
Jaffe	91.75	82.71±5.17	96.23	90.51±3.85	96.23	88.67±4.22	96.23	**96.25**
AR_ImData	62.63	61.08±0.91	67.99	68.22±0.53	70.53	69.91±0.36	**70.56**	64.68
XM2VTS50	82.41	81.26±0.52	81.24	81.29±0.95	88.63	88.46±0.27	**89.03**	82.93
Coil20	79.08	73.16±2.43	88.43	84.34±2.00	**89.17**	84.99±2.10	81.22	88.95

Chapter 7

Conclusions

In this thesis, we proposed a novel graph embedding dimensionality reduction model. Simultaneously, we conducted two processes instead of learning a probabilistic affinity matrix before dimensionality reduction. Based on the projected local connectivity, we assigned the adaptive and optimal neighbors to the model.

In the proposed method, the learnt graph has a block diagonal structure with exactly k connected components. Here k denotes the number of clusters. We impose rank constraint on the Laplacian matrix of graph to achieve the goal. In this thesis, we developed an efficient dimensionality reduction method to optimize the proposed objective. Also, we implement the experiments on both synthetic data and 15 real-world benchmark data sets to reveal the superiority of our proposed dimensionality reduction method.

References

[1] N.Kambhatla, “Local Models and Gaussian Mixture Models for Statistal Data Processing”, PhD Thesis, Oregon Graduate Institute of Science and Technology, 1996.

[2] N.Kumar and A.G.Andreou, “A generalization of Linear Discriminant Analysis in Maximum Likelihood Framework”, Proceedings of the Joint Statistical Meeting, Statistical Computing section, Chicago, Aug 4-8, 1996.

[3] K.K.Paliwal, “Dimensionality Reduction of the Enhanced Feature Set for the HMM-Based Speech Recognizer”, Digital Signal Processing, No. 2, pp. 157-173, 1992.

[4] W.L.Poston and D.J.Marchette, “Recursive Dimensionality Reduction Using Fisher’s Linear Discriminant”, Pattern Recognition, Vol. 31, No. 7, pp. 881-888, 1998.

[5] D.X.Sun, “Feature Dimension Reduction Using Reduced-Rank Maximom Likelihood Estimation For Hidden Markov Model”, Proceedings of Internation Conference on Spoken Language Processing, Philadelphia, USA, pp.244-247, 1996.

[6] Bishop, C. M. (1995) Neural Networks for Pattern Recognition. Oxford University Press.

[7] Duda, R. O., Hart, P. E. and Stork, D. G. (2001) Pattern Classification. 2nd edition, John Wiley & Sons, Inc.

[8] Turk, M. and Pentland, A. P. (1991) Eigenfaces for recognition. Journal of Cognitive Neuroscience, 3, 71-86.

[9] H.Cevikalp, "Feature Extraction Techniques in High-Dimensional Space: Linear and Nonliear Approaches", PhD Thesis, Vanderbilt University, 2005.

[10] Jimenez, L. O. and Landgrebe, D. A. (1998) Supervised classification in high dimensional space: geometrical, statistical, and asymptotical properties of multivariate data. IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews, 28(1), 39-54.

[11] Ng. A. Y. Preventing overfitting of crossvalidation data. In Proceedings of Fourteenth International Conference on Machine Learning, pages 245-253, 1997.

[12] R. Bellman. Adaptive Control Processes: A Guided Tour. Princeton University Press, Princeton, 1961.

[13] J. H. Friedman. On bias, variance, 0/1 – loss, and the curse-of-dimensionality. Technical report, Stanford University, 1996.

[14] G. J. McLachlan. Discriminant Analysis and Statistical Pattern Recognition. John Wiley and Sons, New York, 1992.

[15] J. W. Sammon. A non-linear mapping for data structure analysis. IEEE Transactions on Computers, C-18 (5):401-409, 1969.

[16] J.Kittler. Feature selection and extraction, pages 59-83. Academic Press, Orlando, 1986.

[17] G. H. Dunteman. Principal Component Analysis. Sage Publications, 1989.

[18] G.H. John, R. Kohavi, and K. Pfleger. Irrelevant feature and the subset selection problem. In W.W. Cohen and Hirsh H., editors, Machine Learning: Proceedings of the Eleventh International Conference, pages 121-129, New Brunswick, N.J., 1994. Rutgers University.

[19] G. H. Dunteman. Principal Components Analysis. Sage Publications, 1989.

[20] J. W. Sammon. A non-linear mapping for data structure analysis. IEEE Transactions on Computers, C-18 (5):401-409, 1969.

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

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

[23] M. Dash and H. Liu. Feature selection for classification. International Journal of Intelligent Data Analysis, 1(3), 1997.

[24] H. Liu and H. Motoda, editors. Feature Extraction, Construction and Selection: A Data Mining Perspective. Boston: Kluwer Academic Publishers, 1998.

[25] H. Liu and H. Motoda. Feature Selction for Knowledge Discovery and Data Mining. Boston: Kluwer Academic Publishers, 1998.

[26] L. Yu and H. Liu. Efficient feature selection via analysis of relevance and redundancy. Journal of Machine Learning Research, 5:1205-1224, Oct 2004.

[27] Bishop, C. M. (1995) Neural Networks for Pattern Recognition. Oxford University Press.

[28] A. Jain and R. Dubes, Algorithms for Clustering Data. Englewood Cliffs, NJ: Prentice-Hall, 1988

[29] Lee, K.-C., Ho, J., Yang, M.-H., and Kriegman, D. (2003). Video-based face recognition using probabilistic appearance manifolds. In Computer Vision and Pattern Recognition, 2003. Proceedings. IEEE Computer Society Conference on, volume 1, pages I-313. IEEE.

[30] Nie, F., Xiang, S., Song, Y., and Zhang, C. (2007). Optimal dimensionality discriminant analysis and its application to image recognition. 2013 IEEE Conference on Computer Vision and Pattern Recognition, 0:1–8.

[31] Xiang, S., Nie, F., Zhang, C., and Zhang, C. (2009). Interactive natural image segmentation via spline regression. Image Processing, IEEE Transactions on, 18(7):1623–1632.

[32] Ye, J., Janardan, R., and Li, Q. (2004). Gpca: An efficient dimension reduction scheme for image compression and retrieval. In Proceedings of the Tenth ACM

SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD ’04, pages 354–363, New York, NY, USA. ACM.

[33] Dudoit, S., Fridlyand, J., and Speed, T. P. (2002). Comparison of discrimination methods for the classification of tumors using gene expression data. Journal of the American statistical association, 97(457):77–87.

[34] Ritchie, M. D., Hahn, L. W., Roodi, N., Bailey, L. R., Dupont, W. D., Parl, F. F., and Moore, J. H. (2001). Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer. The American Journal of Human Genetics, 69(1):138–147.

[35] G´amez, A., Zhou, C., Timmermann, A., and Kurths, J. (2004). Nonlinear dimensionality reduction in climate data. Nonlinear Processes in Geophysics, 11(3):393–398.

[36] Mohar, B. (1991). The laplacian spectrum of graphs. In Graph Theory, Combinatorics, and Applications, pages 871–898. Wiley.

[37] Chung, F. R. K. (1997). Spectral Graph Theory. CBMS Regional Conference Series in Mathematics, No. 92, American Mathematical Society.

[38] Fan, K. (1949). On a theorem of weyl concerning eigenvalues of linear transformations. i. 35(11):652–655.

[39] Chen, W.-Y., Song, Y., Bai, H., Lin, C.-J., and Chang, E. Y. (2011). Parallel spectral clustering in distributed systems. IEEE Transactions on Pattern Analysis and Machine Intelligence, 33(3):568–586.

Biographical Information

Yun Liu received her first M.S in Electrical and Electronic Engineering from Nanyang Technological University, and her B.S. in Electrical Engineering from Tianjin University of Technology. Her Research interests are Data Mining, Machine Learning, Computer Vision and Pattern Recognition. She published more than six papers in her related area.