VICTORIA UNIVERSITY

RESEARCH ON JOINT SPARSE REPRESENTATION LEARNING APPROACHES

THESIS SUBMITTED IN FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY TITLE

LUYAO TENG

INSTITUTE FOR SUSTAINABLE INDUSTRIES & LIVEABLE CITIES, VU RESEARCH, VICTORIA UNIVERSITY

Supervisor: Professor Hua Wang – Institute for Sustainable Industries & Livable Cities, Victoria University

Associate Supervisor: Professor Yanchun Zhang – Institute for Sustainable Industries & Livable Cities, Victoria University

VICTORIA UNIVERSITY
MELBOURNE AUSTRALIA

Abstract

Dimensionality reduction techniques such as feature extraction and feature selection are critical tools employed in artificial intelligence, machine learning and pattern recognitions tasks. Previous studies of dimensionality reduction have three common problems: 1) The conventional techniques are disturbed by noise data. In the context of determining useful features, the noises may have adverse effects on the result. Given that noises are inevitable, it is essential for dimensionality reduction techniques to be robust from noises. 2) The conventional techniques separate the graph learning system apart from informative feature determination. These techniques used to construct a data structure graph first, and keep the graph unchanged to process the feature extraction or feature selection. Hence, the result of feature extraction or feature selection is strongly relying on the graph constructed. 3) The conventional techniques determine data intrinsic structure with less systematic and partial analyzation. They maintain either the data global structure or the data local manifold structure. As a result, it becomes difficult for one technique to achieve great performance in different datasets.

We propose three learning models that overcome prementioned problems for various tasks under different learning environment. Specifically, our research outcomes are listing as followings:

1) We propose a novel learning model that joints Sparse Representation (SR) and Locality Preserving Projection (LPP), named Joint Sparse Representation and Locality Preserving Projection for Feature Extraction (JSRLPP), to extract informative features in the context of unsupervised learning environment. JSRLPP processes the feature extraction and data structure learning simultaneously, and is able to capture both the data global and local structure. The sparse matrix in the model operates directly to deal with different types of noises. We conduct comprehensive experiments and confirm that the proposed learning model performs impressive over the state-of-the-art approaches.
2) We propose a novel learning model that joints SR and Data Residual Relationships (DRR), named Unsupervised Feature Selection with Adaptive Residual Preserving (UFSARP), to select informative features in the context of unsupervised learning environment. Such model does not only reduce disturbance of different types of noise, but also effectively enforces similar samples to have similar reconstruction residuals. Besides, the model carries graph construction and feature determination simultaneously. Experimental results show that the proposed framework improves the effect of feature selection.
3) We propose a novel learning model that joints SR and Low-rank Representation (LRR), named Sparse Representation based Classifier with Low-rank Constraint (SRCLC), to extract informative features in the context of supervised learning

environment. When processing the model, the Low-rank Constraint (LRC) regularizes both the within-class structure and between-class structure while the sparse matrix works to handle noises and irrelevant features. With extensive experiments, we confirm that SRLRC achieves impressive improvement over other approaches.

To sum up, with the purpose of obtaining appropriate feature subset, we propose three novel learning models in the context of supervised learning and unsupervised learning to complete the tasks of feature extraction and feature selection respectively. Comprehensive experimental results on public databases demonstrate that our models are performing superior over the state-of-the-art approaches.

Keywords: Subspace Learning, Sparse Representation, Low-Rank Representation, Locality Preserving Projection, Data Residual Relationships, Feature Extraction, Feature Selection

Declaration

I, Luyao Teng, declare that the Ph.D. thesis entitled ‘Research on Joint Sparse Representation Learning Approaches’ is no more than 100,000 words in length including quotes and exclusive of tables, figures, appendices, bibliography, references and footnotes. This thesis contains no material that has been submitted previously, in whole or in part, for the award of any other academic degree or diploma. Except where otherwise indicated, this thesis is my own work.

X [redacted]

Luyao Teng

Acknowledgement

First and foremost, I would like to express my heartfelt gratitude to my distinguished and cordial supervisor, Professor Hua Wang. He is always been a source of knowledge, an inspiration for new ideas, and an exemplary figure for my career development. His patient and attentive mentoring are the primary supports for my accomplishment of this degree.

Thanks are also due to my associate supervisor Professor Yanchun Zhang, the members of "CAI" in the Victoria University, and the members of "Lab 511" in the Guangdong University of Technology, for their generous guidance on my research topic that expedites my journey.

I acknowledge enormous support from Associate Professor Xiaozhao Fang. His sharp criticism and endorsement in his technical skills, his efforts, and assists in developing our studies as an extraordinary supervisor are invaluable to me. Thanks also goes to Doctor Peipei Kang, for her dedicated assistance, initial enlightenment and constant encouragement on chasing the research dream with me. I am also thankful to all other scholars around me in the college and our research group, for fruitful collaborations, thought-provoking discussions, and enduring friendships.

Last but not least, I would like to extend my sincere gratitude to my parents, Professor Shaohua Teng and Associate Professor Wei Zhang, for their continuous supports and concerns, and for their faith in me along the way; my partner, Feiyi (Aaron) Tang, for his unconditional love, support, and encouragement.

Publications

[1] **Teng, L.**, Feng, Z., Fang, X., Teng, S., Wang, H., Kang, P., & Zhang, Y., (2019, May). Unsupervised Feature Selection with Adaptive Residual Preserving. Neurocomputing. DOI: 10.1016/j.neucom.2019.05.097

[2] Zhang, Z., **Teng, L. (corresponding author)**, Zhou, M., Wang, J., & Wang, H., (2019, May). Enhanced Branch-and-Bound Framework for a Class of Sequencing Problems. IEEE Transactions on Systems, Man, and Cybernetics: Systems. DOI: 10.1109/TSMC.2019.2916202.

[3] Peng, Z., Zhang, W., Han, N., Fang, X., Kang, P., & **Teng, L.**, (2019, February). Active Transfer Learning. IEEE Transactions on Circuits and Systems for Video Technology. DOI:10.1109/tcsvt.2019.2900467.

[4] **Teng, L.**, Huo, Y., Song, H., Teng, S., Wang, H., & Zhang, Y., (2018, November). A Novel Incremental Dictionary Learning Method for Low Bit Rate Speech Streaming. Web Information Systems Engineering, WISE, 457-471.

[5] Zhang, W., Kang, P., Fang, X., **Teng, L.**, & Han, N. (2018). Joint Sparse Representation and Locality Preserving Projection for Feature Extraction. International Journal of Machine Learning and Cybernetics, 1-15.

[6] Teng, S., Zhang, Z., **Teng, L.**, Zhang, W., Zhu, H., Fang, X., & Fei, L., (2018, May). A Collaborative Intrusion Detection Model using a novel optimal weight strategy based on Genetic Algorithm for Ensemble Classifier. IEEE 22nd International Conference on Computer Supported Cooperative Work in Design, CSCWD, 761-766.

[7] Teng, S., Wu, N., Zhu, H., **Teng, L.**, & Zhang, W. (2017). SVM-DT-based adaptive and collaborative intrusion detection. IEEE/CAA Journal of Automatica Sinica, 5(1), 108-118.

[8] Zhou, Y., Wang, J., Chen, J., Gao, S., & **Teng, L.** (2017). Ensemble of many-objective evolutionary algorithms for many-objective problems. Soft Computing, 21(9), 2407-2419.

[9] Zhu, H., Liu, D., Zhang, S., Zhu, Y., **Teng, L.**, & Teng, S. (2016). Solving the many to many assignment problems by improving the Kuhn–Munkres algorithm with backtracking. Theoretical Computer Science, 618, 30-41.

[10] **Teng, L.**, Zhang, W., Tang, F., Teng, S., & Fu, X. (2016, January). The Study and Application on Multi-dimension and Multi-layer Credit Scoring. In International Conference on Human Centered Computing, 386-399.

[11] **Teng, L.**, Yang, X., Tang, F., Teng, S., & Zhang, W. (2014, November). 3-dimension evaluation method for stock investment based on 2-tuple linguistic. In International Conference on Human Centered Computing, 326-339.

[12] **Teng, L.**, Teng, S., Tang, F., Zhu, H., Zhang, W., Liu, D., & Liang, L. (2014, December). A collaborative and adaptive intrusion detection based on SVMs and

decision trees. 2014 IEEE International Conference on Data Mining Workshop, 898-905.

[13] Tang, A., He, J., Tang, Y., Peng, Z., & **Teng, L.** (2014). Sparse ranking model adaptation for cross domain learning to rank. 網際網路技術學刊, 15(6), 949-962.

[14] Zhang, W., **Teng, L.**, He, X., Teng, S. H., & Zhu, H. (2013). An Association Analysis Method with Varying Threshold and Updating Data Based on Shared Pattern-Trees. Journal of Advanced Mathematics and Applications, 2(2), 196-203.

[15] Teng, S., Huang, S., Huo, Y., **Teng, L.**, & Zhang, W. (2012, November). A new positioning algorithm in mobile network. Joint International Conference on Pervasive Computing and the Networked World, 461-476.

To my parents and partner, for their unconditional love, support, and encouragement

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

In this chapter, we first overview the dimensionality reduction techniques, and then briefly describe the concepts of feature selection, feature extraction, supervised learning and unsupervised learning. Next, we provide the research purposes, challenges, and contributions. Last, we give the outline of the entire thesis.

1.1 Dimensionality Reduction

In today's real applications, the high-dimensional data is growing in exponential speed. Such data contains hundreds or thousands of features. Despite the fact that many of these features are useful, numerous irrelevant or redundant features may cause certain issues, such as low performance and high computational cost, which deteriorate the output result. In order to find useful information from vast amount of data, dimensionality reduction technologies have been widely adopted.

Dimensionality reduction aims to representing high-dimensional data into a meaningful low-dimensional space. Ideally, the low-dimensional space is corresponding to the intrinsic dimension of the original data. The intrinsic dimension is the minimum number of feature dimensions that requires to present the properties of the data. The dimensionality reduction techniques have essential effects on number of applications in processing sensor arrays, image processing, multivariate data analysis, and data mining. For example, to forecast weather accurately, the data pattern of historical time signals records needs to be pinpointed effectively. To classify face images, the noises and irrelevant features need to be eliminated such that informative features are kept and enable users associate the most important features with the training samples. To detect similarities between text and image, a general dictionary needs to be learnt such that the test data regarding the same class can be grouped together even the data sources are from different domains in different types.

This thesis investigates dimensionality reduction techniques in the context of obtaining a subset of informative features from different datasets. On the one hand, the dimensions of data in real applications may represent similarities and correlations, which demonstrate close relationships between features, such that one of the features is representative enough and others need to be eliminated. On the other hand, the noisy features in dataset may make an adverse effect on retrieve useful information. To process this data requires specifically designed dimensionality reduction techniques.

There are numerous different techniques regarding the dimensionality reductions have been proposed, and these approaches are grouped based on two respects: features extraction (Nie, Xu, Tsang, & Zhang, 2010) (Xu, Zhu, Fan, Wang, & Pan, 2013) or feature selection (Shang, Chang, Jiao, & Xue, 2017) (He, Cai, & Niyogi, 2005) (Wen, et al., 2018) (Hou, Nie, Li, Yi, & Wu, 2017) (Fang, et al., 2014) (Cai & Zhu, 2017). In this thesis, we propose two feature extraction techniques and one feature selection technique.

Figure (1-1) demonstrates the hierarchy of dimensionality reduction techniques, we are going to describe the feature extraction in section 1.1.1, and the feature selection in section 1.1.2.

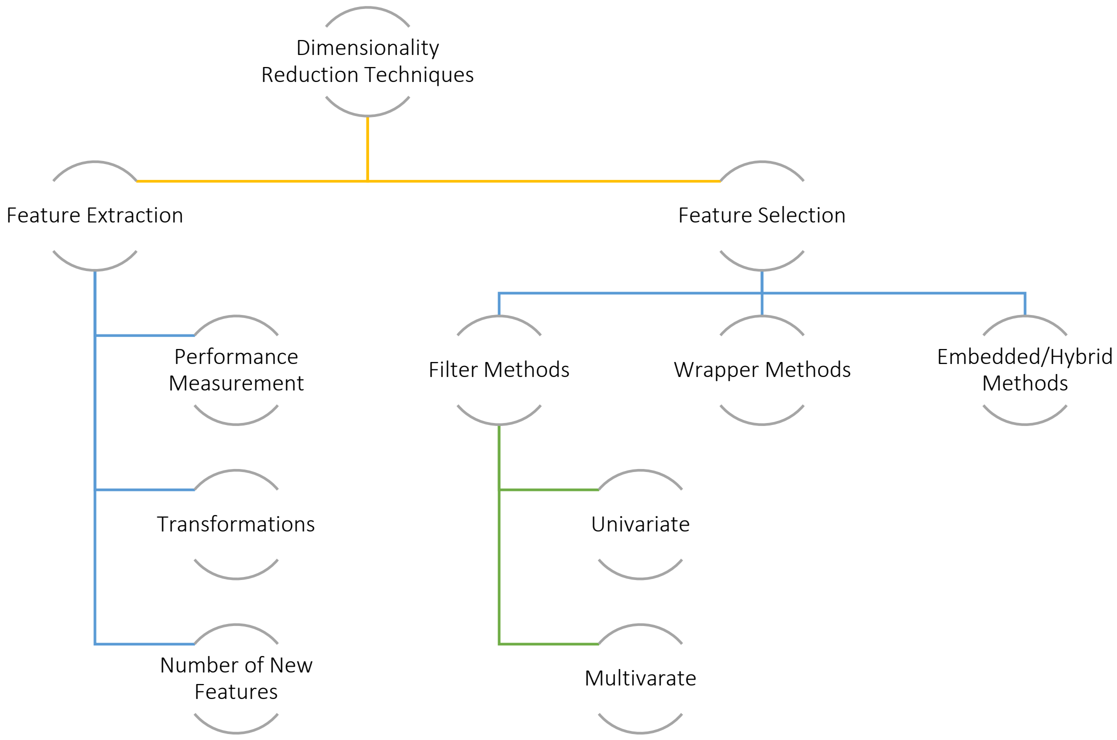


Figure 1-1: The Hierarchy of Dimensionality Reduction Techniques

1.1.1 Feature Extraction

Feature extraction contains general methods that involve transforming original features and creating new feature subset, and the new feature subset is believed to be more representative to the original dataset. The feature extraction issues can be summarized as in Table (1-1).

Table 1-1: Issues of Feature Extraction

Issues	Attributes
Performance Measure	It is going to investigate and select the most suitable features in the dataset.
Transformation	It is going to transform the original attributes into new features.
Number of new features	It helps to determine the minimum number of new features.

Such new subset of features is within a comparatively low-dimensional subspace and aims to preserve the most significance information. Principal component analysis (PCA) is one of the typical and popular feature extraction technique that is going to be demonstrated in detail in Chapter Two.

1.1.2 Feature Selection

Different from feature extraction, feature selection approaches select a subset of features without changing the original features. Generally, these approaches consist of four steps: feature subset generation, subset evaluation, stopping criterion and result validation. During the conversion, various search methods, such as complete search, sequential search and random search, may be applied to generate the subset. In the feature selection phase, the goodness is evaluated using a specific evaluation criterion, and the determined feature subset will continuously be updated and replaced once there comes a newly generated feature subset has higher score unless the stopping criterion has been met.

Feature selection can be realized in three different ways, i.e.: filter methods, wrapper methods, and embedded/hybrid methods.

- Filter methods, which filter out poorly performed features solely based on the statistical properties of variables. Such methods are easily scalable to extensively high dimensional datasets, and have small computation cost. However, they completely ignore the induction algorithm during the feature selection.
- Wrapper methods, which determine the feature subset dataset by utilizing a predetermined learning algorithm. Since these methods consider the interaction between feature subset and model selection, they may achieve higher accuracy rate than filter methods. However, these methods are lack of generality and computational expensive, thus they are always not be the option in the context of large-scale datasets.
- Embedded/Hybrid methods, which combines filters and wrappers together such that the methods are less computationally intensive and have the interaction with learning models.

Table (1-2) illustrates the details of each type of the feature selection methods:

Table 1-2: Attributes of Feature Selection Types

Method Type	Attributes
Filter Methods	• Univariate Methods: Consider the input variables (features, attributes) one by one • Multivariate Methods: Consider whole groups of variables together • Scalability to large-scale datasets • Fast and easy • Generality/not bound to a specific problem
Wrapper Methods	• High accuracy • Lack of generality/bound to classifier • Concern the interaction between feature selection and learning models
Embedded/Hybrid Methods	• Computation cost is less than Wrappers • Considering learning models

1.1.3 Supervised Learning & Unsupervised Learning

Besides to the feature extraction and feature selection, learning models can also be classified into Supervised Learning (Fan, Xu, & Zhang, 2011) and Unsupervised Learning (Yang J. , Zhang, Frangi, & Yang, 2004) depending on the availability of label information. Due to the missing label information, unsupervised learning is more challenged. However, most datasets in real-world applications are unlabeled. In this thesis, we propose two unsupervised study and one supervised study.

1.2 Research Purposes

The principal purpose of this thesis is to build effective dimensionality reduction techniques. Generally, the effectiveness is measured by robustness and accuracy. On the one hand, robustness is crucial to all learning approaches where it enables learning approaches to be spread to different datasets in different scenarios. On the other hand, accuracy provides an intuitive perspective on the performance of techniques. In this thesis, we have conducted extensive experiments to demonstrate the robust and accurate of our models.

1.3 Research Challenges

The existing dimensionality reduction techniques have four major challenges, data structure maintaining, graph-learning system, robust to noises, and computational cost. We elaborate each as follows.

1.3.1 Data Structure Maintaining

In the context of the structure reconstruction, the natural structure of the original high dimensional dataset should be preserved when obtaining feature subset. However, previous studies in data structure maintaining appear to be simplified and partialized. Most of these techniques focus on either data global structure or local manifold structure only. For example, the Sparse Representation (SR) is only able to represent the data global structure, the Locality Preserving Projection (LPP) retains the local neighborhood relationships, and the Low-rank Representation (LRR) focus on the data class structure. None of these techniques consider both global structure and local structure during the data reconstruction. Hence, they are inadequately in representing the data intrinsic structure.

1.3.2 Graph-learning System

Generally, the conventional dimensionality reduction methods process graph-learning procedure and feature extraction or feature selection in two independent steps. As a result, the obtained feature subset is largely based on the previous learned graph. For example, the Isometric Feature Mapping (ISOMAP), Locally Linear Embedding (LLE), and Laplacian Eigenmaps (LE) are popular techniques have been proved in preserving nonlinear relationships, yet they need to plot the data relationships first. Such that the problem become more complicated as it is difficult to define the ineffectiveness result is purely due to a defect of graph construct or the design of algorithm.

1.3.3 Robust to Noises

For the high-dimensional datasets, the noisy and redundant features are inevitable and may adversely affect the performance of algorithms. Conventional techniques focus on obtaining and evaluate features, yet they neglect the effect of noises. These techniques are sensitive to the noises, and with the interference of noises, features obtained may not be the optimal choices.

1.3.4 Computational Cost

As dimensionality reduction techniques are usually confronting large scale datasets with high dimensions, the computational cost can be excessive. The problem lies in the complexity of the datasets. Hence, it can be incredibly expensive to carry out the algorithms.

1.4 Research Contributions

We propose three dimensionality reduction approaches to solve the problems of existing techniques. These approaches are applied under different environment and for different tasks.

- We propose a novel unsupervised feature extraction approach, named Joint Sparse Representation and Locality Preserving Projection (JSRLPP). In this technique, the data structure learning and feature extraction are being processed at the same time within one unified framework. Besides, SR and LPP are working together to preserve the data global and local structure. The sparse matrix in the model helps to eliminate the effects from noises. The comprehensive experimental results have demonstrated the effectiveness of our approach.

- We propose a novel unsupervised feature selection approach, named Unsupervised Feature Selection with Adaptive Residual Preserving (UFSARP). It is worthwhile to notice that we are the first one to propose Data Residual Relationships (DRR) in the context of dimensionality reduction domain. DRR reviews the data intrinsic structure and reconstructs the dataset with the idea of similar samples to have similar reconstruction residuals. This technique is able to better represent the data structure and is robust to noises. Also, it operates the feature selection and data reconstruction simultaneously in a single unified framework. We have conducted extensive experiments and confirmed that our technique is superior over other state-of-the-art techniques.
- We propose a novel supervised feature extraction approach, named Joint Sparse Representation and Low-rank Constraint (SRLRC). In this technique, the LRC does not only work to regularize the between-class structure, but also operates on within-class structure. In addition, the sparse matrix in the algorithm can effectively handle the noises and irrelevant features. The extensive experiments demonstrate that our approach performs impressively better than other similar techniques.

1.4 Thesis's Outline

Three dimensionality learning models have been designed, and the Figure (1-1) demonstrates the whole structure of this thesis.

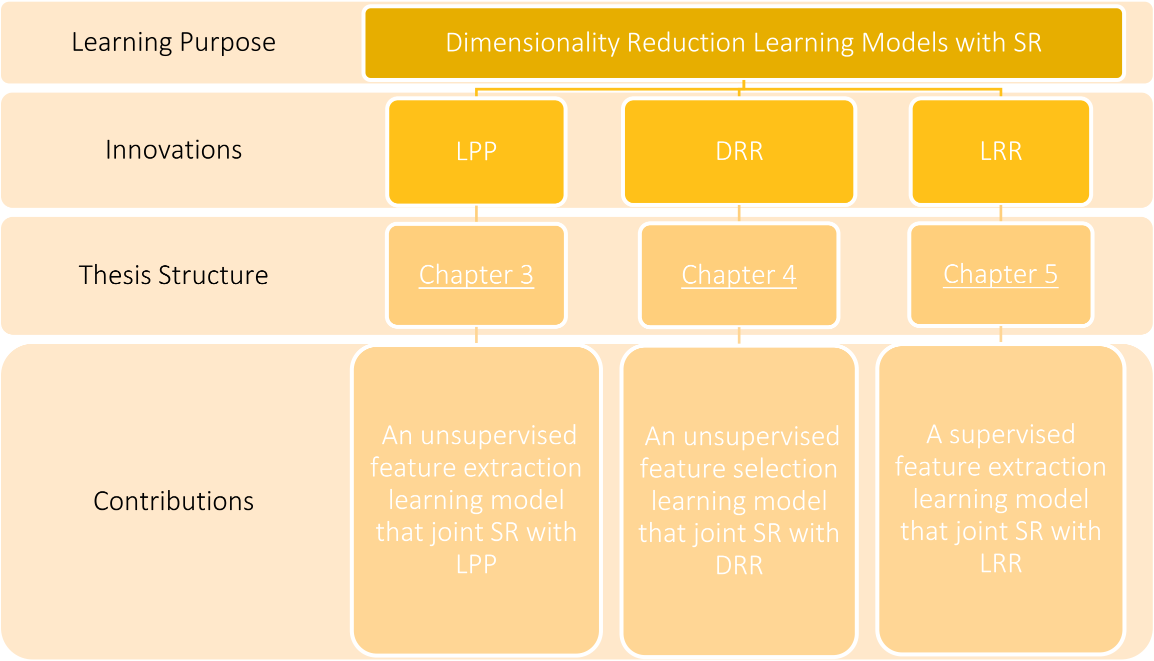

Learning Purpose
Dimensionality Reduction Learning Models with SR
Innovations
LPP
DRR
LRR
Thesis Structure
Chapter 3
Chapter 4
Chapter 5
Contributions
An unsupervised feature extraction learning model that joint SR with LPP
An unsupervised feature selection learning model that joint SR with DRR
A supervised feature extraction learning model that joint SR with LRR

Figure 1-2: Thesis's Organizing Structure

This thesis is organized as follows:

- Chapter One: presents an overview of this thesis. We first introduce the background of dimensionality reduction, including detailed explications and discriminations of feature extraction and feature selection, supervised learning and unsupervised learning, and then we present the research purpose, the challenges and the research contributions. Last, we list the organizing structure.
- Chapter Two: presents a literature review from three relevant aspects for the problems studied in this thesis: the conventional subspace learning approaches, the SR algorithm, and the LRR algorithm. In addition, we provide the related datasets and the sources. Please note that the datasets of the image recognition in dimensionality reduction is excessively rich, we list the datasets have been applied in this thesis only.

- Chapter Three: elaborates the details of processing the SR based unsupervised feature extraction learning model and our extension on LPP, i.e., the JSRLPP. We first point out the problems of the state-of-art algorithms, and identifies the need to be addressed, i.e., the previous studies of LPP learning algorithm separate graph learning away from obtaining informative features. Next, we design the novel learning model JSRLPP, which employs a similarity matrix to adaptively learnt data structure via graphing weighted relationships between samples over the feature subset learning at each iteration. We evaluate our proposed algorithm against the state-of-the-art techniques in dimensionality reduction.
- Chapter Four: elaborates our investigation on the problem of DRR in the context of dimensionality reduction tasks. We first identify the existing drawbacks of conventional approaches on maintaining sample structure. Then we design and operate DRR on directly retaining the residual relationships between samples. Next, we extend DRR to the base on SR that guarantee the quality of data graph reconstructed. Last, we compare USFARP to similar techniques to evaluate its performance.
- Chapter Five: elaborates the details of processing the SR based supervised feature extraction learning model and our extension on LRR, i.e., the SRCLC. We propose two LRC terms in regularizing between-class data structure and within-class data structure, respectively. Then we further enhance our work by employing SR algorithm. We compare SRCLC to the state-of-the-art techniques to prove the effectiveness.

- Chapter Six: focus on the future work and expectations. We provide a brief introduction of currently conducting models on employing kernel trick. In addition, it further discusses the possible future research directions.
- Chapter Seven: concludes the thesis with the results obtained from the three novel learning models. It further discusses the limitations and the future possible extensions of the studies.

Chapter 2 Literature Review

In this Chapter, we review the work on the typical dimensionality reduction approaches from three relevant aspects: the subspace learning approaches, the Sparse Representation (SR) algorithm, and the Low-rank Representation (LRR) algorithm.

This chapter is organized as: we first present a description of notations, then we introduce a serial of those most historic conventional popular subspace learning approaches, including Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Locally linear embedding (LLE), Isometric Feature Mapping (ISOMAP), Laplacian Eigenmaps (LE), and Locality Preserving Projections (LPP). In addition, we explore the SR algorithm, and LRR algorithm. Next, we list the sources and description of databases have involved in research experiments. After all, there is a conclusion of this chapter.

2.1 Notation

In this thesis, we use the bold uppercase letters to denote matrices and use the bold lowercase letters to denote vectors. For an arbitrary matrix $\boldsymbol{F} \in \mathbb{R}^{m \times n}$, $\boldsymbol{f}_i$ stands for the i-th column vector of $\boldsymbol{F}$, and $\boldsymbol{f}_j^T$ stands for the $\boldsymbol{j}$-th row vector of $\boldsymbol{F}$. The $l_{2,1}$-norm is defined as $\|\boldsymbol{F}\|_{2,1} = \sum_{i=1}^{d} \sqrt{\sum_{j=1}^{k} \boldsymbol{F}_{ij}^2}$.

Definitions and Proofs:

Give a vector $\boldsymbol{x}$ *and matrix* $\boldsymbol{X}$*, while* $\boldsymbol{x} \in \mathbb{R}^n$*, and* $\boldsymbol{X} \in \mathbb{R}^{m \times n}$*,*

Definition 1: $\|\boldsymbol{x}\|_0$ *denotes* l_0*-norm, which calculates the number of non-zero elements in the coefficient vector* $\boldsymbol{x}$*.*

Definition 2: $\|\boldsymbol{x}\|_1$ *denotes* l_1*-norm, which calculates the total sum of the absolute value of each element in the coefficient vector* $\boldsymbol{x}$*, that is:* $\|\boldsymbol{x}\|_1 = \sum_{i=1}^{m} |x_i| = |x_1| + |x_2| + \cdots + |x_m|$.

Definition 3: $\|\boldsymbol{x}\|_2$ *denotes* l_2*-norm, which is defined as:* $\|\boldsymbol{x}\|_2 = \sqrt[2]{\sum_{i=1}^{m} |\boldsymbol{x}_i|^2} = (|\boldsymbol{x}_1|^2 + |\boldsymbol{x}_2|^2 + \cdots + |\boldsymbol{x}_m|^2)^{1/2}$*. The* l_2*-norm is also known as Euclidean norm or Frobenius norm.*

Definition 4: $\|\boldsymbol{x}\|_\infty$ *denotes* $l_\infty - norm$ *, which is defined as:* $\|\boldsymbol{x}\|_\infty = max\,(|\boldsymbol{x}_1|, |\boldsymbol{x}_2|, \dots, |\boldsymbol{x}_m|)$*.*

Definition 5: $\|\boldsymbol{X}\|_1$ *denotes* l_1*-norm, which calculates the sum of the absolute value of each element in the coefficient matrix* $\boldsymbol{X}$*, that is:* $\|\boldsymbol{X}\|_1 = \sum_{i=1}^{m} \sum_{j=1}^{n} |x_{ij}|$.

Definition 6: $\|\boldsymbol{X}\|_2$ *denotes* l_2*-norm, which is defined as:* $\|\boldsymbol{X}\|_2 = \left(\sum_{i=1}^{m} \sum_{j=1}^{n} |x_{ij}|\right)^{\frac{1}{2}}$*. The* l_2*-norm is also known as Frobenius norm.*

Definition 7: $\|\boldsymbol{X}\|_\infty$ *denotes* l_∞*-norm, which is defined as:* $\|\boldsymbol{X}\|_\infty = \underset{i=1,\dots,m,j=1,\dots,n}{max}\{|x_{ij}|\}$.

Definition 8: $\|\boldsymbol{X}\|_{2,1}$ *denotes* $l_{2,1}$*-norm, which is defined as:* $\|\boldsymbol{X}\|_{2,1} = \sum_{i=1}^{m}\left(\sum_{j=1}^{n} |x_{ij}|\right)^{\frac{1}{2}}$.

2.2 Subspace Learning

With the advanced enhancement in data collection and storage capabilities, there is an incredibly large number of multidimensional data being collected on a daily base in various research domains, such as in pattern recognition, multichannel EEG, machine learning and gene expression (Abecasis, Cherny, Cookson, & Cardon, 2002) (Bishop, 1995) (Shalkoff, 1989). These kinds of massive multidimensional data are always lying in such a high-dimensional input space with a substantial number of redundancies. Namely, the intrinsic informative features are solely occupying a subset of the original input dimensions. In addition, since processing high-dimensional data is time-consuming, subspace learning sheds light on the dimensionality reduction applications. It projects the original high-dimensional scatters into a low-dimensional space, wherein the true structure of the original dataset can be well preserved.

There are numbers of dimensionality reduction methods being proposed in the preceding decades, and these methods are generally be classified into two groups: linear dimensionality reduction methods and non-linear dimensionality reduction methods. PCA (Yang J. , Zhang, Frangi, & Yang, 2004) (Fan, et al., 2014) and LDA (Fan, Xu, & Zhang, 2011) (Lai, Xu, Yang, Tang, & Zhang, 2013) are typical linear dimensionality reduction methods. LLE, ISOMAP, LE (Belkin & Niyogi, 2003), and LPP (He & Niyogi, 2003) are popular non-linear (manifold) dimensionality reduction methods.

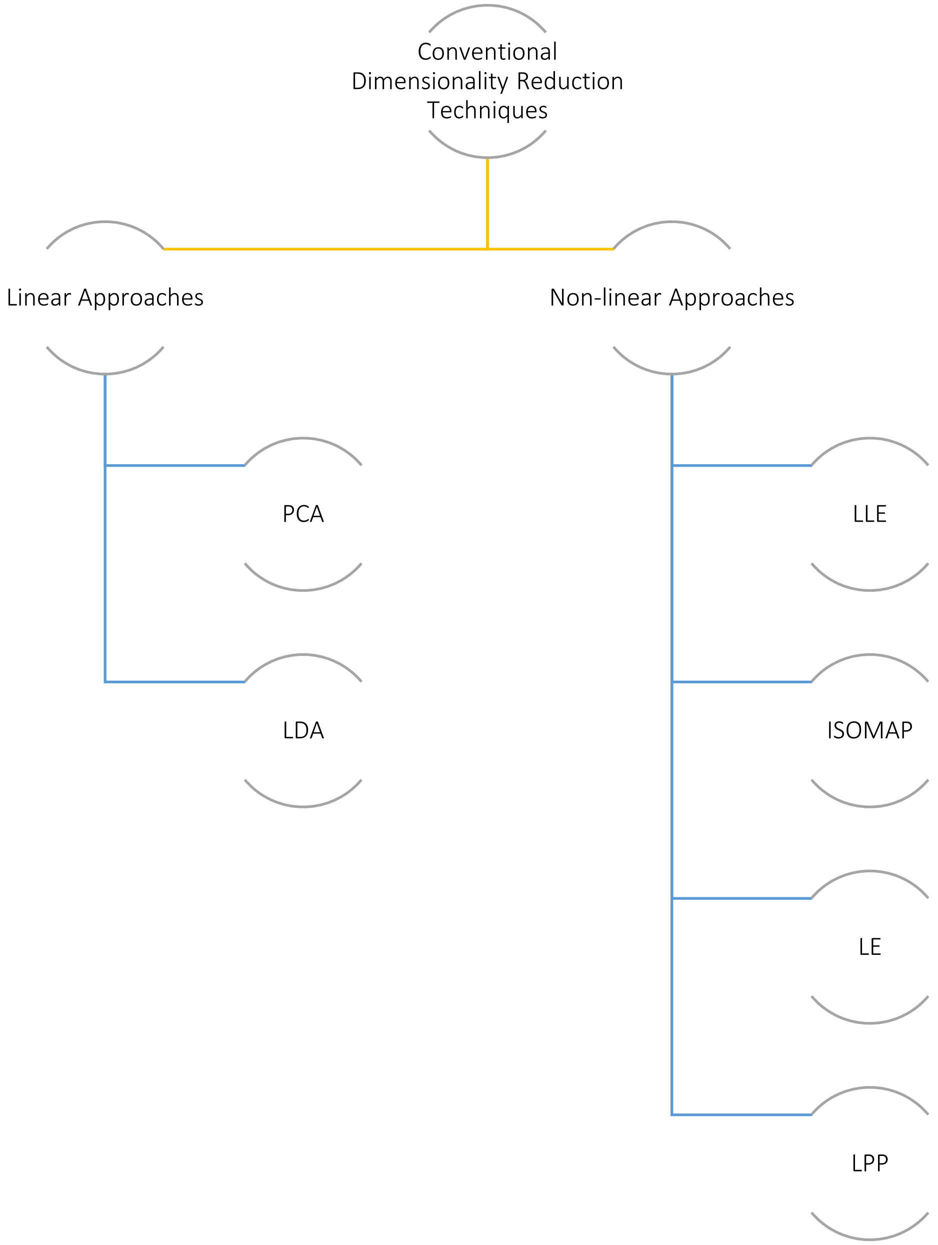


Figure 2-1: Conventional Dimensionality Reduction Approaches

2.3 Linear Subspace Learning Approaches

PCA and LDA are two typical linear subspace learning approaches, and both approaches have been applied in different real applications to extract essential information from a set of redundant or noisy data.

2.3.1 Principle Component Analysis (PCA)

PCA is one of the most popular and widely used feature extraction approach and data representation technique in various domains, such as data compression, image analysis, pattern recognition, time series perdition, etc. The main idea of PCA is to reveal data structure behind the complex dataset, and intends to discover the most significant features, and remove the redundant data points. In processing PCA, the original set of correlated variables is going to be orthogonally transferred into a set of newly formed uncorrelated variables. The principle components (PC) have two properties: 1) each PC is a presentation of a linear combination regarding the raw variables; 2) each PC are uncorrelated to all other PCs. Since PCA is easy to understand and does not have any constraints, it has been applied in various domains.

(Sirovich & Kirby, 1987) are the first one who introduced PCA in characterizing human faces. They discussed that the face image datasets can be approximately reconstructed as a weighted sum of a small collection of images. This obtained collection defines a facial basis, which is also known as Eigen-Images, and a mean image of the database. Then, (Turk & Pentland, 1991) developed a function that projected human face images into a feature space that maximize the data variations. This set of significant features are the eigenvectors, also knowns as the principal components of the face image dataset, and the features can be also called as “Eigenfaces” (Belhumeur, Hespanha, & Kriegman, 1997). In 2006, (Zou, Hastie, & Tibshirani, 2006) combined the PCA with SR for the purpose to improve the robustness of the framework. Additionally, (Skoˇcaj, Leonardis, & Bischof, 2007) proposed the WPCA, which involved a Weight Parameter in the algorithm to reduce the effect of noises.

The PCA algorithm is straightforward to be understood. Considering there is a set of N sample images $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$ are taken from c different humans $\{\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_c\}$. Assuming that the image is in an m-dimensional feature space, and we map the original sample data using a linear transformation to an n-dimensional space, where $m > n$. Assuming $\boldsymbol{y}_k \in \mathbb{R}^m$ are newly formed feature vectors and are represented by the following linear transformation:

$$\boldsymbol{y}_k = \boldsymbol{W}^T \boldsymbol{x}_k$$

$$k = 1, 2, \ldots, N$$

(2-1)

In the above Problem (2-1), $\boldsymbol{W} \in \mathbb{R}^{n \times m}$ is a transformation matrix that is orthonormal columns vectors.

Defined the total sample variation matrix as $\boldsymbol{S}_T$:

$$\boldsymbol{S}_T = \sum_{k=1}^{N} (\boldsymbol{x}_k - \boldsymbol{\mu})(\boldsymbol{x}_k - \boldsymbol{\mu})^T$$

(2-2)

N in the Problem (2-2) denotes to the total number of data images, and $\boldsymbol{\mu} \in \mathbb{R}^N$ denotes to the mean image of the dataset. Then, after implementing the linear transformation, we have a set of feature vectors. These vectors are selected to maximize the total sample variance matrix of the projected sample:

$$\boldsymbol{W}_{opt} = \arg\max_{\boldsymbol{w}} |\boldsymbol{W}^T \boldsymbol{S}_T \boldsymbol{W}|$$

(2-3)

then, we have:

$$[\boldsymbol{w}_1\ \boldsymbol{w}_2\ \dots\ \boldsymbol{w}_m]$$

(2-4)

$\{\boldsymbol{w}_i | i = 1, 2, \dots, m\}$ in Problem (2-4) denotes the set of eigenvectors of $\boldsymbol{S}_T$, which the elements are the m largest eigenvalues.

Despite PCA is widely adopted in many real applications, the technique is suffering from certain limitations:

- PCA assumes that the relationships between variables are linear and it transforms variables entirely based on linearly projecting.

- PCA is only meaningful when all the variables are set to be scaled at the numeric level.
- The projection directions are intending to maximize the total variance of the dataset where this variance does not only include the between-class samples but also involve the within-class samples. As a result, PCA is optimal for data structure reconstruction and in preserving original individual data point information, however, it may not be optimal in discrimination. For instance, the sample faces are under considerable variations, such as under different lighting, facial expression, and pose conditions.

Figure (2-2) and Figure (2-3) are examples that present face images from an identical person under different illumination conditions.

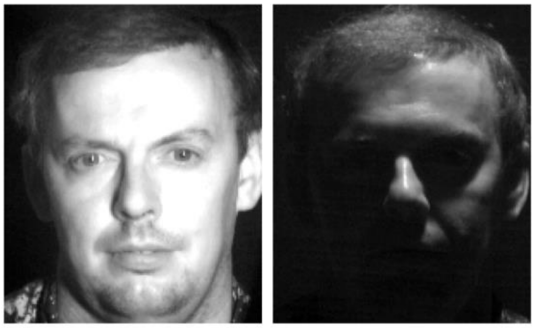


Figure 2-2: The Same Person Under Different Illumination Conditions

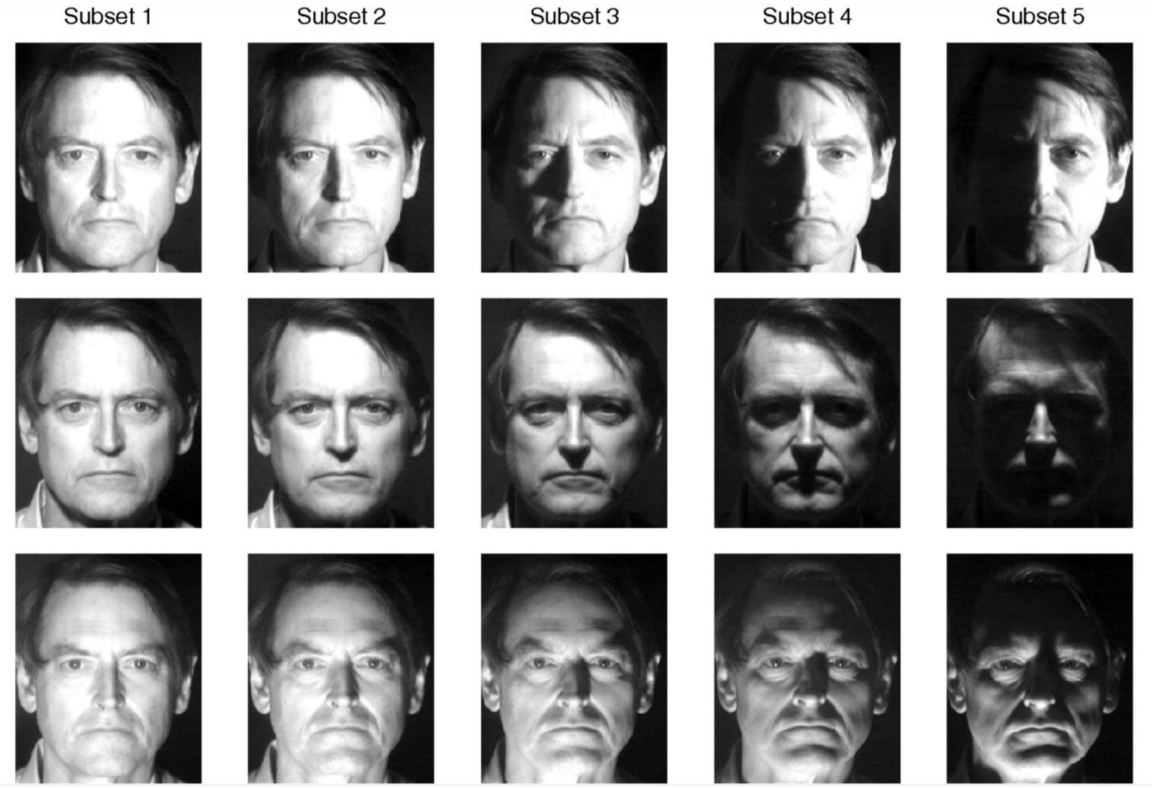


Figure 2-3: Example Images from The Harvard Face Image Database (https://faculty.ucmerced.edu/mhyang/face-detection-survey.html)

2.3.2 Linear Discriminant Analysis (LDA)

LDA is one of the most conventional methods in image recognition. LDA is initially proposed by (Fisher, 1936) for taxonomic classification. Then, the technique has been expended to the domain of image recognition and computer vision. In 1991, (Cheng, Liu, Yang, Zhuang, & Gu, 1991) proposed an algorithm that employed Fisher's discriminator on face image dataset, in which, the features were gathered by a polar quantization of the image. Then in 1996, (Baker & Nayar, 1996) introduced a theory of pattern rejection and built a framework on a Two-Class Linear Discriminant. Also, (Cui, Swets, & Weng, 1995) proposed a method named as Most Discriminating Feature (MDF) to recognize hand gestures.

In the high-dimensional space, LDA is always facing the “small sample size” problem. To end this problem, (Li, Jiang, & Zhang, 2006) has proposed an algorithm, Maximum Margin Criterion (MMC). MMC is working to maximize the average margin between the data classes after dimensionality reduction. Also, to avoid the loss of data space information, (Xiong, Swamy, & Ahmad, 2005) introduced the Two-Dimensional FLD, and (Yang & Dai, 2009) proposed the Two-Dimensional Maximum Margin Feature Extraction approach.

Different from PCA, LDA attempts to distinguish different classes. The transformation matrix in this approach is selected to maximize the between-class variances and minimize the within-class variances. Thus, in LDA, the distance among classes are maximized. However, data information might be lost during the analyzing process.

LDA is an algorithm that maximizes the value of between-class variances divided by within-class variances. Let the between-class variance matrix to be:

$$\boldsymbol{S}_B = \sum_{i=1}^{c} N_i(\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T$$

(2-5)

where $\boldsymbol{\mu}_i$ denotes to the mean image of the i-th person, and N_i expresses to the total number of sample images for the i-th person.

Let the within-class variance matrix to be:

$$\boldsymbol{S}_W = \sum_{i=1}^{c} \sum_{\boldsymbol{x}_k \in \boldsymbol{X}_i} N_i(\boldsymbol{x}_k - \boldsymbol{\mu}_i)(\boldsymbol{x}_k - \boldsymbol{\mu}_i)^T$$

(2-6)

After defining the between-class variances and within-class variances, the value of LDA can be measured.

Suppose $\boldsymbol{W}_{opt}$ is a set of orthonormal vectors that maximize the rate of between-class variance over within-class variance, i.e.:

$$\boldsymbol{W}_{opt} = \arg\max_{\boldsymbol{W}} \frac{|\boldsymbol{W}^T \boldsymbol{S}_B \boldsymbol{W}|}{|\boldsymbol{W}^T \boldsymbol{S}_W \boldsymbol{W}|}$$

(2-7)

In the Problem (2-7), the eigenvectors corresponding to m largest eigenvalues will be chosen.

Figure (2-4) compares PCA and LDA regarding with a two-class dataset. In this example, $N = 20$, $c = 2$, and $m = 1$. So, the 20 data points will be projected onto a line instead of the original 2-dimensional space.

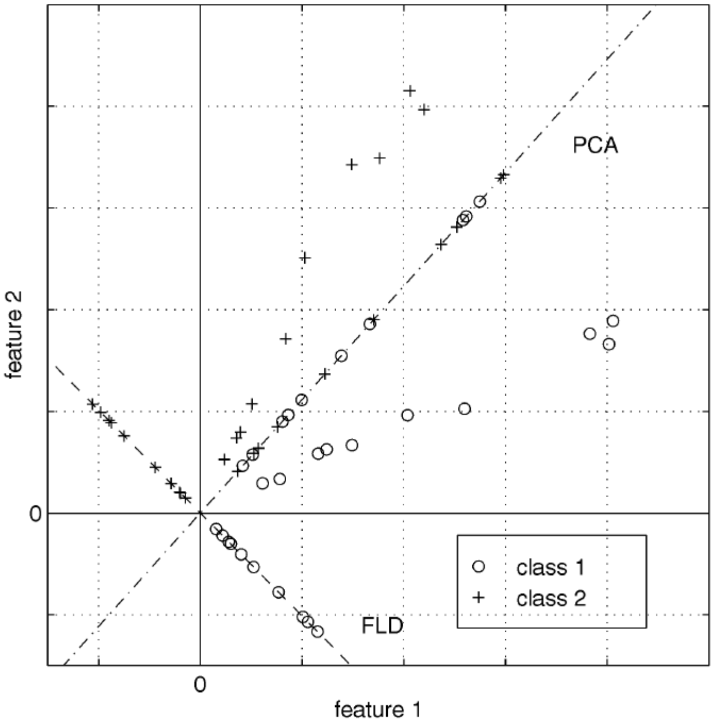


Figure 2-4: PCA vs. LDA

From the above Figure (2-4), it is apparent that data expressed by PCA line has smeared the two classes with each other. PCA encourages to retain the most original data information, it is advanced in representing each individual information, but it fails to distinguish one class from the other. Adversely, data represented by FDA line (LDA line, also called Fisher's Linear Discriminant) has sharp recognizable class information. Namely, the differences between classes are apparent.

Despite the wide usages in many applications, LDA has certain limitations:

- LDA assumes that the relationships between variables are linear and it transforms variables entirely based on linearly projecting.

- LDA is only meaningful when all the variables are set to be scaled at the numeric level.
- When processing LDA, the distance between classes is maximized, yet data within the same class may overlap with each other.

2.4 Non-linear Subspace Learning Approaches

PCA and LDA are both using the global Euclidean structure in graphing the data structure. Both of the approaches guarantee to discover the true structure of datasets that are spreading in or near a linear subspace of the original high-dimensional space. PCA finds out the set of vectors which can maximize the data variance while LDA finds an embedding which maximizes the variance between different classes. However, neither PCA or LDA has little to do with the data manifold structure. In another words, it is difficult for PCA and LDA to discover the data underlying manifold structure behind.

Figure (2-5) demonstrates a three-dimensional dataset with a two-dimensional manifold. PCA and LDA are unable to determine the intrinsic structure as points far apart on the underlying manifold are recognized to be deceptively close while using the classic straight-line Euclidean distance.

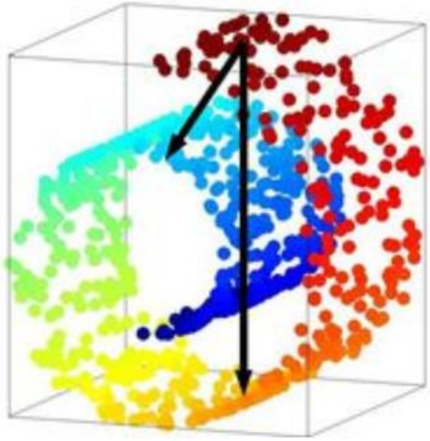


Figure 2-5: Manifold Learning Instance

According to the theory of differential geometry, the data manifold’s intrinsic geometry is able to be fully expressed by the data local metric and the data infinitesimal neighborhoods information. A large number of feature extraction techniques, graph embedding approaches, have been proposed regarding the local metric, such as the LLE (Roweis & Saul, 2000), ISOMAP (Tenenbaum, Silva, & Langford, 2000), etc. All these graph-based approaches are employing an easily measured local metric information to obtain the hidden global geometry of the dataset and are capable to reveal the data nonlinear degrees of freedom that is lying behind the complex natural observations.

Figure (2-6) is an example of a Canonical Problem in dimensionality reduction from different visual perceptions. In the figure, there is a sequence of images regarding a single face that has been observed under various poses and lighting conditions, in no particular order. Each of the images is regarded as a sample data point in a high-dimensional feature space. The dimensions are corresponding to different features, including the illumination of one pixel in the image, and the firing rate of one retinal ganglion cell, etc. The images are (64 pixel × 64 pixel, which is 4096-dimensional input space) lying on an intrinsic three-dimensional manifold that is capable to be parameterized by the two poses variables and one azimuthal illumination angle.

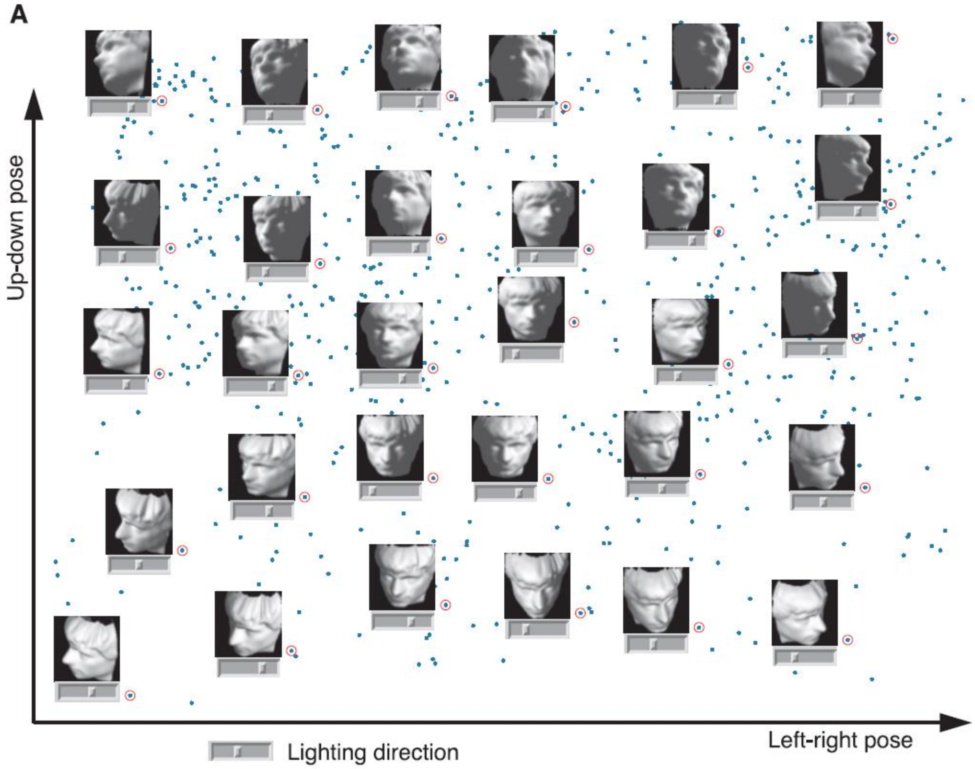


Figure 2-6: A Canonical Dimensionality Reduction Problem from Different Visual Perception

2.4.1 Isometric Feature Mapping (ISOMAP)

The ISOMAP is an approach that was proposed by (Tenenbaum, Silva, & Langford, 2000). Similar to LLE approach, ISOMAP guarantees to converge asymptotically to the data intrinsic structure by preserving local metric information.

ISOMAP is one of the most wildly used approaches in manifold learning. It is built based on the classic MDS, but ISOMAP aims to preserve the true geometry of the dataset. For neighboring points, the geodesic distance can be measured based on the original input-space distance by using Euclidean distance. And for faraway points, the geodesic distance can be determined by the sum of a series of neighbors, which is calculated by summing up the shortest paths in a graph with edges that are connecting the neighboring data points. In Figure (2-7) (B), the Red-Line reveals the geometric distance in the manifold between two particular data points.

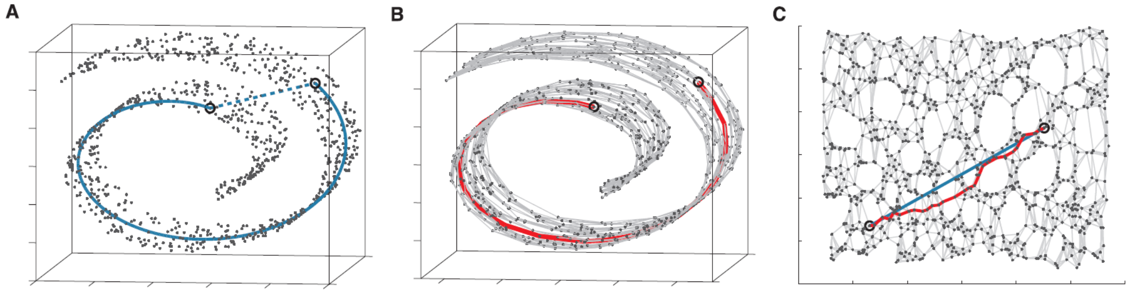


Figure 2-7: The "Swiss roll" problem

There are three-steps for ISOMAP to measure the distance between two sample points:

1. Identifying the neighborhood on the manifold. Two widely applied methods are ε-Balls Method and the k-Nearest Neighbors. These neighborhood relationships are constructed as a weighted graph with edges of weight between neighbors, shown as in Figure (2-7) (B).
2. Computing the geodesic distances between each pair of neighbors on the manifold, and solving the shortest paths problem.
3. Constructing an embedding of the data in a d-dimensional Euclidean space that can preserve the data manifold's intrinsic geometry best, as shown in Figure (2-7) (C).

The objective function of ISOMAP is to minimize the following cost:

$$E = \|\tau(\boldsymbol{D}_G) - \tau(\boldsymbol{D}_Y)\|_{L^2} \tag{2-8}$$

where $\boldsymbol{D}_G$ expresses to the matrix of graph distances, $\boldsymbol{D}_G = \{d_G(i,j)\}$. $\boldsymbol{D}_Y$ represents to the matrix of Euclidean distances that $\boldsymbol{D}_Y = \{d_Y(i,j) = \|\boldsymbol{y}_i - \boldsymbol{y}_j\|\}$ and the $\|\cdot\|_{L^2}$ is the L^2 matrix-norm $\sqrt{\sum_{i,j} A_{ij}^2}$. The operator τ converts the distances to inner products that uniquely characterize the geometry of the dataset.

2.4.2 Locally Linear Embedding (LLE)

The LLE is an approach that was introduced by (Roweis & Saul, 2000). LLE is an unsupervised learning approach that projects the high-dimensional data toward a particular global coordinate system of lower-dimensionality, and the optimizations are not involving local minima. LLE learns the global structure of nonlinear manifolds by exploiting the local metric information.

In LLE, it is expected that each sample image and its neighbors to be close to a locally linear patch of the manifold, and each of the sample points is reconstructed from its neighbors to minimize the reconstruction errors.

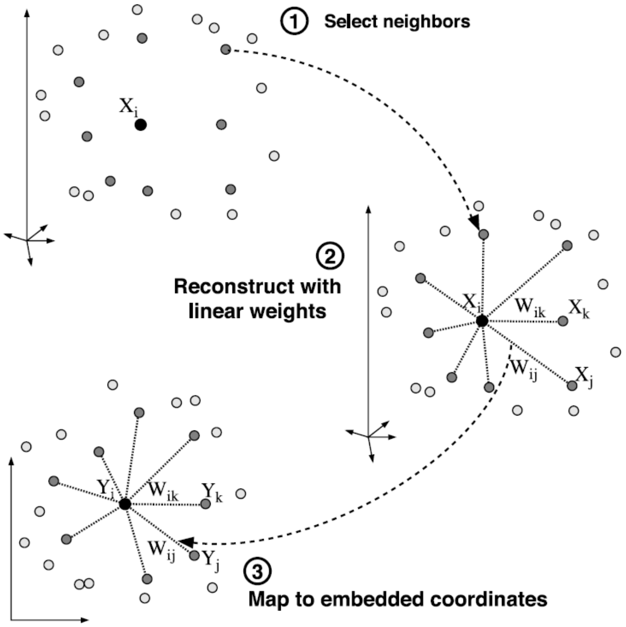


Figure 2-8: Steps of locally linear embedding

From Figure (2-8) we can observe that the LLE reconstruction contains three steps:

1. Assigning neighbors to each sample data point $\boldsymbol{x}_i$. Two widely used methods are ε-balls method and the k-nearest neighbors.
2. Computing the weights $\boldsymbol{W}_{ij}$ between testing point and its neighbors. Then, solving the least-squares problem as shown in Function (2-9)

3. Computing the low-dimensional embedding graph with the fixing weights $\boldsymbol{W}_{ij}$, and minimized the Function (2-10).

The reconstruction errors of LLE are measured by the following cost function:

$$\varepsilon(\boldsymbol{W}) = \sum_i \left| \boldsymbol{x}_i - \sum\nolimits_j \boldsymbol{W}_{ij} \boldsymbol{x}_j \right|^2 \tag{2-9}$$

where $\boldsymbol{x}_i$ is the i-th sample data point, $\boldsymbol{x}_j$ is the j-th sample data point. $\boldsymbol{W}_{ij}$ represent the weight that j-th sample point contributes to the i-th sample point during the reconstruction process.

Function (2-9) is subjected to two constraints:

1. Each sample image point $\boldsymbol{x}_i$ will be represented solely by its neighbors, and LEE enforces weight $\boldsymbol{W}_{ij} = 0$ if $\boldsymbol{x}_j$ is not a neighbor of $\boldsymbol{x}_i$.
2. Each row of the weight matrix has to be summed up to one: $\sum_j \boldsymbol{W}_{ij} = 1$.

Suppose that there is a D-dimensional dataset which is lying on a continuous nonlinear manifold of lower dimensionality d, where $d \ll D$. There is consist of a transformation, rotation, and rescaling that can map the neighborhood coordinates in high-dimensional space into global internal coordinates on the manifold. During the processes, the parameter $\boldsymbol{W}_{ij}$ represents the true geometric properties of the sample points and it works to retain the local geometry structure during the transformations by keeping the weights $\boldsymbol{W}_{ij}$ unchanged.

Thus, the local neighborhood structure will be preserved during the reconstruction of mapping dataset from D dimensions to d dimensions.

As each sample point $\boldsymbol{x}_i$ in the original high-dimensional input space will be mapped into a low-dimensional space, $\boldsymbol{y}_i$, then there is:

$$\epsilon(\boldsymbol{Y}) = \sum_i \left| \boldsymbol{y}_i - \sum\nolimits_j \boldsymbol{W}_{ij} \boldsymbol{y}_j \right|^2 \tag{2-10}$$

In the above Function (2-10), $\boldsymbol{W}_{ij}$ is the weight between points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, and the value of $\boldsymbol{W}_{ij}$ has been calculated in Function (2-9).

2.4.3 Laplacian Eigenmaps (LE)

Besides ISOMAP and LLE, (Belkin & Niyogi, 2003) proposed a different geometrically motivated learning approach, LE. The approach builds a data structure graph based on the dataset neighborhood information.

The Laplacian operator provides an optimal embedding that enables LE to determine the true geometric structure of the underlying manifold. In addition, the combination of Laplacian and Heat Kernel ensures the learning approach to choose the weights of the graph in a principled manner. Furthermore, as ISOMAP and LLE are attempting to preserve all pairwise geodesic distances, they do not show any tendency to cluster. However, the LE learning approach is implicitly emphasizing the natural attribution of clusters that are embedded in the dataset.

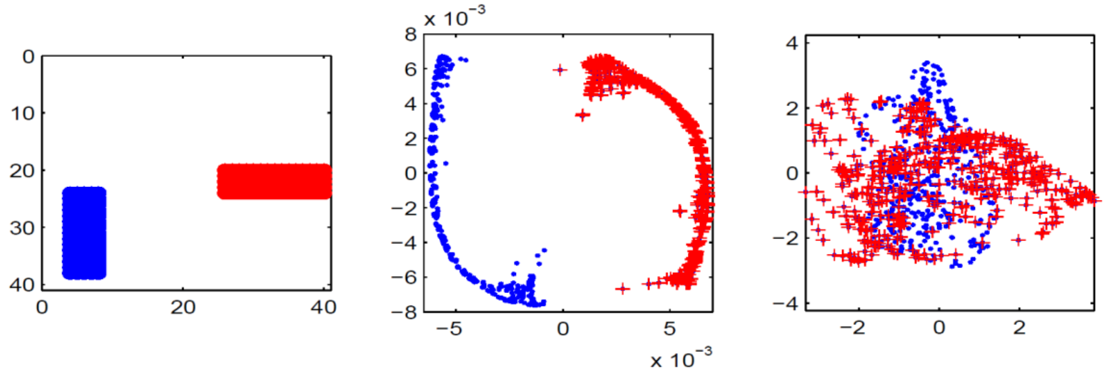


Figure 2-9: LE vs. PCA

Figure (2-9) is a Toy Vision Example that demonstrates the clustering effects by using LE and PCA. The left-hand side image is an instance regarding a picture of a binary image that contains a vertical bar and a horizontal bar located at arbitrary points in a 40×40 visual field. In the experiment, there are 1000 images have been selected. 500 of them carry a vertical bar and 500 hold a horizontal bar at random. The middle panel of Figure (2-9) demonstrated a two-dimensional representation of all samples that is using LE, and the right panel illustrated the result using PCA. It is apparent that LE performs impressive in data clustering.

LE contains three steps:

1. Constructing the adjacency graph. Two widely used methods are ε-balls method and the k-nearest neighbors.

2. Weighting the edges for the adjacency graph. There are two variations for choosing the weights $\boldsymbol{W}_{ij}$, heat kernel and simple-minded.
3. Building the eigenmaps.

The eigenvector problem can be generalized as:

$$L\boldsymbol{y} = \lambda \boldsymbol{D}\boldsymbol{y}$$

(2-11)

In the Problem (2-11), $\boldsymbol{D}$ denotes to a diagonal weighted matrix that the entries are column sum of the weighted matrix $\boldsymbol{W}$, $\boldsymbol{D}_{ii} = \sum_j \boldsymbol{W}_{ji}$. $\boldsymbol{L}$ expresses the Laplacian matrix, $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$. The solution $\{\boldsymbol{y}_0, \boldsymbol{y}_1, \dots, \boldsymbol{y}_{k-1}\}$ are ordered according to the eigenvalues with $\boldsymbol{y}_0$ that is the solution with the smallest eigenvalue. Then the new representation of the original sample image $\boldsymbol{x}_i$ under the embedding towards the lower-dimensional space $\mathbb{R}^m$ is $\{\boldsymbol{y}_1(i), \dots, \boldsymbol{y}_m(i)\}$.

2.4.4 Local Preserving Projection (LPP)

LPP is a linear learning approach that carries the main idea of LE. It is first proposed by (He & Niyogi, 2003) and was designed to preserve the data neighborhood structure. LPP works under the assumption that the nearest neighbor relations in the original high-dimensional input space should be sustained during the data reconstruction. Unlike those aforementioned approaches ISOMAP, LLE, and LE, which are approaches that can only be utilized on the training set and are unable to evaluate the reconstructed map for newly coming testing points, LPP can be employed to any fresh coming data points.

Suppose there is an n-dimensional dataset $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_n\}$ lies on a low-dimensional submanifold. LPP expects that in the target d-dimensional space ($d \ll n$), there exists a set of data points $\{\boldsymbol{y}_1, \boldsymbol{y}_2, \dots, \boldsymbol{y}_n\}$ which are sharing the same local neighborhood relationships as the original dataset. Under this expectation, a weighted graph $G = (V, E, \boldsymbol{W})$ will be constructed, where V indicates to the set of sample points, E expresses to the set of edges, and $\boldsymbol{W} = (w_{ij})$ denotes to a similarity matrix with the entries of weights between two neighbor points. In LPP, the neighborhood structure will be measured by $w_{ij} = \exp\left(-\left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|^2/\beta\right)$ if $\boldsymbol{x}_j$ is among k-nearest neighbors of $\boldsymbol{x}_i$, or $\boldsymbol{x}_i$ is among k-Nearest Neighbors of $\boldsymbol{x}_j$, and otherwise $w_{ij} = 0$.

The objective function of LPP is to minimize:

$$\frac{1}{2}\sum_{ij}\left\|\boldsymbol{y}_i - \boldsymbol{y}_j\right\|^2 w_{ij}$$

(2-12)

where $\boldsymbol{y}_i$ is the transformation result of the original sample point $\boldsymbol{x}_i$, that $\boldsymbol{y}_i = \boldsymbol{A}^T\boldsymbol{x}_i$. $\boldsymbol{A} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \dots, \boldsymbol{a}_d] \in \mathbb{R}^{n\times d}$.

The Function (2-12) can be rewritten as:

$$\frac{1}{2}\sum_{ij}\left\|\boldsymbol{y}_i-\boldsymbol{y}_j\right\|^2 w_{ij}=\frac{1}{2}\sum_{ij}\left(\boldsymbol{A}^T\boldsymbol{x}_i-\boldsymbol{A}^T\boldsymbol{x}_j\right)^T\left(\boldsymbol{A}^T\boldsymbol{x}_i-\boldsymbol{A}^T\boldsymbol{x}_j\right)w_{ij}$$

$$=Tr(\boldsymbol{A}^T\boldsymbol{X}(\boldsymbol{D}-\boldsymbol{W})\boldsymbol{X}^T\boldsymbol{A})=Tr(\boldsymbol{A}^T\boldsymbol{X}\boldsymbol{L}\boldsymbol{X}^T\boldsymbol{A})$$

(2-13)

In Function (2-13), $\boldsymbol{D}$ denotes to a diagonal weighted matrix that each element equals the column sum of $\boldsymbol{W}$, $\boldsymbol{D}_{ii}=\sum_j \boldsymbol{W}_{ji}$, where $\boldsymbol{L}$ indicates the Laplacian matrix, $\boldsymbol{L}=\boldsymbol{D}-\boldsymbol{W}$.

When processing LPP, there are three procedures:

1. Constructing the adjacency graph. Two widely used methods are ε-balls method and the k-nearest neighbors.
2. Weighting the edges for the adjacency graph. There are two variations for choosing the weights $\boldsymbol{W}_{ij}$, heat kernel and simple-minded.
3. Calculate the projection matrix.

The projection matrix is generalized as:

$$\boldsymbol{X}\boldsymbol{L}\boldsymbol{X}^T\boldsymbol{A}=\lambda\boldsymbol{X}\boldsymbol{D}\boldsymbol{X}^T\boldsymbol{A}$$

Based on the theory of LPP, there are serials of works have been done to improve the accuracy, such as the Local Discriminant Embedding Approach (LDE) that was proposed by (Chen, Chang, & Liu, 2005), the Unsupervised Discriminant Projection Approach (UDP) that was proposed by (Yang J. , Zhang, Yang, & Niu, 2007), Two-Dimensional Locality Preserving Projections Approach (2D-LPP) that was proposed by (Chen, Zhao, Kong, &

Luo, 2007), and the Null Space Discriminant Locality Preserving Projections Approach (NSDLPP) that was proposed by (Yang, Gong, Gu, Li, & Liang, 2008).

2.4.5 Limitations

Despite of the widely usages in many applications of these non-linear dimensionality reduction techniques, they share certain limitations:

- They are able to properly determine the data manifold structure and effectively reconstructs the data structure, yet they separate the structure learning and obtaining informative features into two independent steps.
- These approaches solely describe the data local neighborhood relationships and ignore the global structure, except ISOMAP. However, the computational complexity of ISOMAP is extensively high to be carried in large scale datasets.

2.4.6 Comparisons

Table (2-1) demonstrates six properties of different techniques, including Global/Local, Supervised/Unsupervised, Parametric/Non-parametric, Parameters/No Parameters, Computational Complexity and Memory Complexity.

Table 2-1: The Attributes of Conventional Approaches

Technique	*Global or Local*	*Supervised or Unsupervised*	*Parametric*	*Parameters*	*Computational Complexity*	*Memory*
PCA	Global	Unsupervised	Yes	None	$O(D^3)$	$O(D^2)$
LLE	Local	Unsupervised	No	k	$O(pr^2)$	$O(pr^2)$
ISOMAP	Global	Unsupervised	No	k	$O(r^3)$	$O(r^2)$
LE	Local	Unsupervised	No	k, σ	$O(pr^2)$	$O(pr^2)$

For non-parametric techniques, the dimensionality reduction process needed to be performed when there appears a new test data, this difficulty is named as out-of-sample problem. Besides, these techniques do not have an insight of the data structure retained either. hey cannot evaluate the error between reconstructed structure and the true structure.

For techniques required parameters, the main advantage is they are comparable more flexible to the technique, yet they need to be tuned to optimize the performance.

For the computational and memory complexities, D denotes to the original dimensions, r represents the number of dimensions that will be reduced to, and p is the proportion of nonzero elements in sparse matrix.

2.5 Sparse Representation (SR)

There is enormous literature that has been devoted in the concern of projecting the original high-dimensional images into a lower-dimensional feature space, such as approaches that we have introduced in previous sections, PCA, LDA, LLE, ISOMAP, LE, and LPP, etc. However, these approaches have certain limitations which we have discussed in previous sections, and there is so little consensus comparing which approach performs better.

(Wright, Yang, Ganesh, Sastry, & Ma, 2009) are the first one who employed SR in face recognition domain. Compared with those conventional feature extraction approaches, SR straightly utilizes the data to represent the dataset instead of selecting a limited subset of features to represent. Therefore, the major issue of SR becomes whether there are enough features to represent the sample points, but not the choices of features.

Typically, if there is a sufficient number of training images from each data class, a testing image can be reconstructed by a linear combination of all training images that are coming from the same class which the testing image belongs to. This expression is generally sparse so that most coefficients in the expression are zero. Generally, the percentage of non-zero elements will vary from zero to approximately 30 percent.

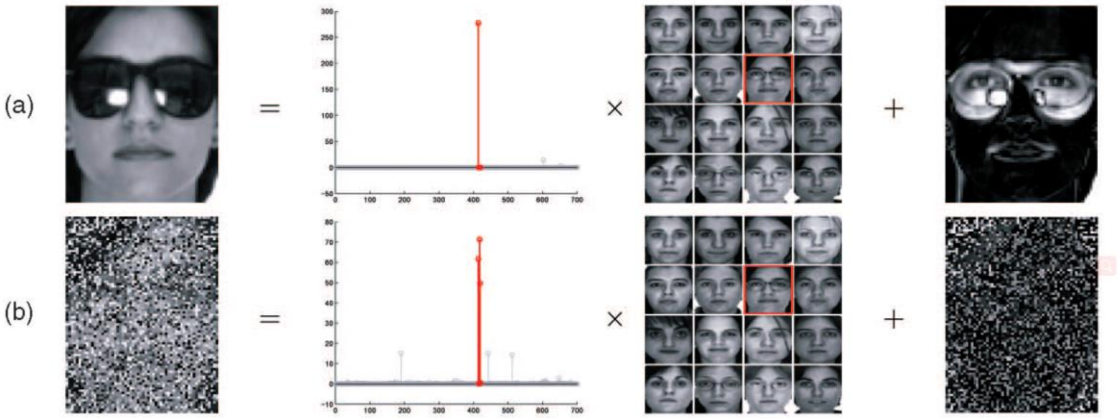


Figure 2-10: Overview of Sparse Representation

Figure (2-10) illustrates two testing samples selected from a dataset with 700 training samples for 100 individuals. Image (a) is an obstructed human face, and image (b) is a corrupted human face. Each testing image can be roughly described by the sparse linear combination of all the training images with a sparse error, as shown in the figure. SR is able to determine the true identity and the individual that has been outlined in the Red Box.

The SR problem is known as a convex optimization problem that the optimal representation should be sufficiently sparse. However, the solving process can be extremely complicated. Since the optimization problem is similar to the Lasso, an easier solution is to penalize the problem using l_1-norm in the linear combination instead of straightly utilizing l_0-norm to penalize the number of nonzero coefficients.

2.5.1 SR Algorithm

Given a dataset $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_n\} \in \mathbb{R}^m$, where n indicates to the total amount of samples in the dataset, $m = d \times n$ expresses to the original number of dimensions of $\boldsymbol{X}$, and $\boldsymbol{x}_i \in \mathbb{R}^d$ denotes to the i-th sample in the dataset. The objective of SR is to use the fewest number of samples to describe a randomly selected test sample $\boldsymbol{y} \in \mathbb{R}^d$ from j-th class, and the problem can be resolved by:

$$\boldsymbol{y} = \boldsymbol{\alpha}\boldsymbol{X} + \boldsymbol{\xi}$$

(2-14)

In Function (2-14), $\boldsymbol{\alpha} \in \mathbb{R}^m$ is the vector of sparse coefficients, $\boldsymbol{\xi}$ is the noise term. Ideally, $\boldsymbol{\alpha}$ will only be non-zero entries whenever the representative training images are coming from the same data class as the testing image from. That is:

$$\boldsymbol{\alpha} = \left\{0, \dots, 0, \alpha_{j,1}, \dots, \alpha_{j,n_j}, 0, \dots, 0\right\}^T$$

(2-15)

In the Function (2-15), $\boldsymbol{\alpha}_{j,i} \in \mathbb{R}^m$ denotes to the corresponding coefficient of the training sample $\boldsymbol{x}_{j,i}$, and $j = 1, \dots, c$ indicates the total quantity of classes for the dataset.

Since it is expected that all elements in $\boldsymbol{\alpha}$ to be zero except the coefficient corresponding to the specific class that testing sample is from, $\boldsymbol{\alpha}$ has the attribution of sparse, therefore, the problem can be described as:

$$\min_{\alpha} \|\boldsymbol{\alpha}\|_0$$

$$s.t. \boldsymbol{y} = \boldsymbol{\alpha X} + \boldsymbol{\xi}$$

(2-16)

As mentioned in Section 2.2, the l_0 -minimization problem is a non-deterministic polynomial-time hard (NP-hard) puzzle, and it can be resolved by solving an l_1 -minimization problem:

$$\min_{\alpha} \|\boldsymbol{\alpha}\|_1$$

$$s.t. \boldsymbol{y} = \boldsymbol{\alpha X} + \boldsymbol{\xi}$$

(2-17)

Despite of the wide usages of SR in many applications, there are certain limitations:

- SR uses l_1 graph for subspace segmentation that only individually considers the sparsest representation for each sample.
- SR utilizes the entire training set as a dictionary to describe an input signal, which is computationally expensive.
- The performances are deteriorating when the dataset is contaminated, such as occlusion, illumination, disguise variations, etc.
- SR is solely describing the data global structure but ignores the data local neighborhood relationships.

2.6 Low-Rank Representation (LRR)

In the pattern recognition and signal processing domain, data are often lying in a high-dimensional space. However, in various real-world applications, including in image recognition, motion, and texture, it is sufficient to use a small set of data features to represent the original high-dimensional input dataset, and it is common for data from the same class to lie in or near a low-rank subspace.

The rank minimization problem attracts a large amount of awareness in recent years, and it has been successfully implemented in different domains, such as in image denoising, motion prediction, matrix fulfillment, and recovery, etc. In 2009, (Wright, Ganesh, Rao, & Ma, 2009) introduced the Robust Principal Component Analysis (RPCA) to deal with the outlying or corrupted observations. Then in 2010, (Liu, et al., 2010) introduced the LRR for subspace structure learning. In 2011, (Lin, Liu, & Su, 2011) proposed the Linearized Alternating Direction Method with Adaptive Penalty for Low-Rank Representation. Followed by (Zhang, Jiang, & Davis, 2013), proposed the Learning Structured Low-Rank Representations for image classification, and (Tang, Liu, Su, & Zhang, 2014) proposed a Structure-Constrained Low-Rank Representation to analyze the structure of various disjoint subspaces.

The main idea of LRR is easy to be understood. Suppose there is a dataset that each sample is able to be reconstructed by a linear combination of the bases of all other training samples, LRR is aiming to find out the Lowest-Rank Representation of all data jointly. The LRR problem can be solved by employing a nuclear-norm regularized optimization problem, which is convex and is able to be resolved within polynomial time.

2.6.1 LRR Algorithm

Suppose there is a dataset $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_n] \in R^{d\times n}$ to be a group of m-dimensional data that was extracted from a collection of linear subspaces $\{\boldsymbol{S}_i\}_{i=1}^{d}$, and the dimension of each subspace is denoted by r_i. Then, each sample image of the dataset can be reconstructed by a linear combination of the base dictionary, $\boldsymbol{A} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \dots, \boldsymbol{a}_n]$, thus we have:

$$\min_{\boldsymbol{Z},\boldsymbol{E}} rank(\boldsymbol{Z}) + \lambda\|\boldsymbol{E}\|_l$$

$$s.t.\boldsymbol{X} = \boldsymbol{AZ} + \boldsymbol{E}$$

(2-18)

In the Function (2-18), $\boldsymbol{Z} = [\boldsymbol{z}_1, \boldsymbol{z}_2, \dots, \boldsymbol{z}_n]$ is the coefficient matrix, $\boldsymbol{E}$ denotes the corrupted errors, $\lambda > 0$ represents a parameter, and $\|\cdot\|_l$ expresses particular regularization strategy.

The optimization Function (2-18) can be transferred into following rank minimization problem:

$$\min_{\boldsymbol{Z}}\|\boldsymbol{Z}\|_*$$

$$s.t.\boldsymbol{X} = \boldsymbol{AZ}$$

(2-19)

Despite of the wide usages of LRR in many applications, LRR is solely describing the class structure of the dataset.

2.8 Datasets

There are different kinds of datasets have been involved in my researches, including human face image, handwriting, and standard machine learning datasets, etc.

(1) The ORL database

The ORL face image dataset is provided by AT&T Laboratories, Cambridge University. The dataset consists of 400 images for 40 people with 10 images per person. The images are taken with various time, illumination, facial expressions, and facial details. Followings are some samples from the ORL database:

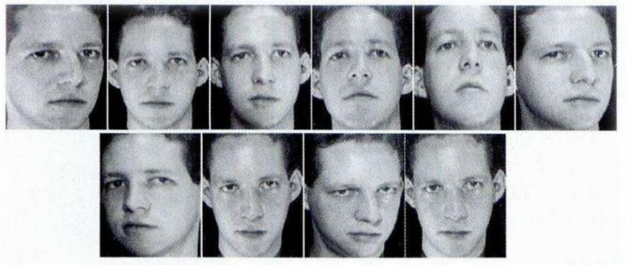


Figure 2-11: The ORL Database samples

The dataset can be retrieved from:

(http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html)

(2) The Yale database

Yale face image dataset is provided by Yale University. The dataset contains 165 grayscale images for 15 individuals with 11 images per individual. The images are taken with different facial expression or configuration. Followings are some samples from the Yale database:

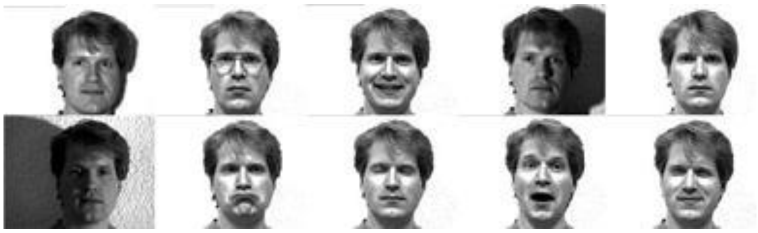


Figure 2-12: The Yale Face Database samples

The dataset can be retrieved from:

(http://www.cad.zju.edu.cn/home/dengcai/Data/FaceData.html)

(3) The Extended Yale Face Database B

The Extended Yale B face image dataset is produced by Yale University. The dataset consists of 5760 images for 10 individuals. Each individual has been taken 576 images under various viewing conditions (9 poses with 64 lighting conditions). Followings are some samples from the Extended Yale B database:

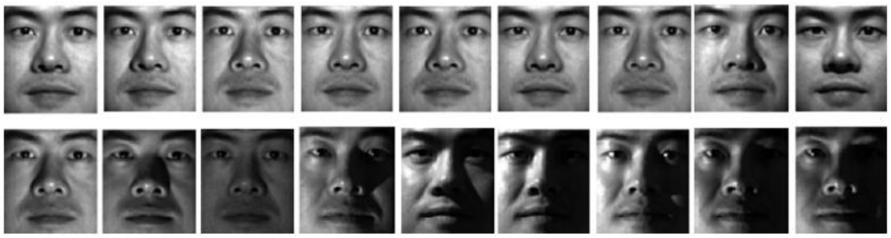


Figure 2-13: The Extended Yale Face Database B samples

The dataset can be retrieved from:

(http://www.cad.zju.edu.cn/home/dengcai/Data/FaceData.html)

(4) The Sheffield Face Database (UMIST Face Database)

UMIST face image dataset is provided by the University of Sheffield. The dataset consists of 564 images for 20 people. Each person has been taken a series of images that covers a range of postures from profile to frontal. The individuals selected are from different race/sex/appearance. Followings are some samples from the UMIST database:

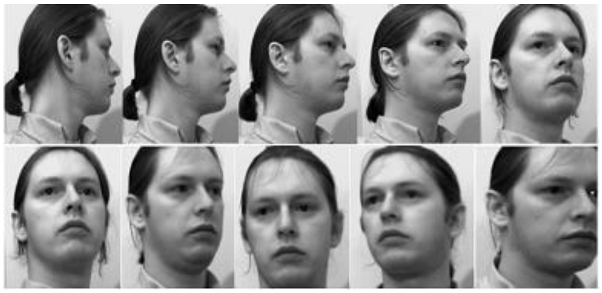


Figure 2-14: The UMIST Face Database samples

The dataset can be retrieved from:

(https://www.sheffield.ac.uk/eee/research/iel/research/face)

(5) The Japanese Female Facial Expression (JAFFE) Database

The JAFFE dataset consists of 213 images for 10 female Japanese students. Each student has been taken photographs of 7 facial expressions, including angry, disgust, fear, happy, sad, surprise, neutral. Followings are some samples from the JAFFE database:

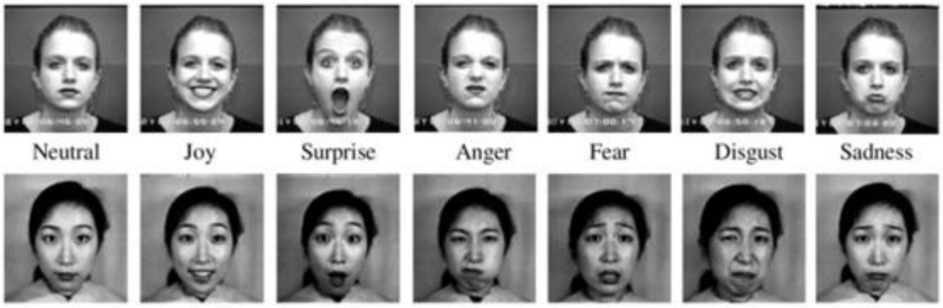


Figure 2-15: The JAFFE Database samples

The dataset can be retrieved from:

(http://www.face-rec.org/databases/)

(6) The AR Face Database

The AR face image dataset is provided by the Robot Vision Lab at Purdue University, USA. The dataset consists of 4000 color photographs for 126 individuals (70 gentlemen and 56 gentlewomen). Every individual has been taken a series of images that covers a set of facial expressions, lighting conditions, and occlusions. Followings are some samples from the AR database:

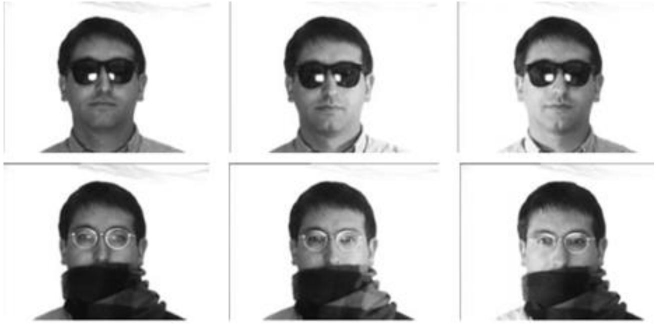


Figure 2-16: The AR Face Database samples

The dataset can be retrieved from:

(http://www2.ece.ohio-state.edu/~aleix/ARdatabase.html)

(7) The COIL-20 Database

The COIL-20 object image dataset is provided by Columbia University. The dataset consists 1440 grayscale images for 20 objects. Every object has been taken 72 photographs from various angles. Followings are some samples from the COIL-20 database:

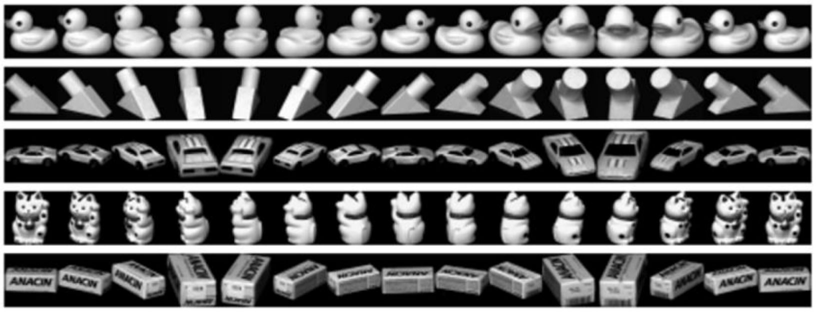


Figure 2-17: The COIL-20 Database samples

The dataset can be retrieved from:

(http://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php)

(8) MFEAT Handwritten Database

The MFEAT handwritten dataset is extracted from the UCI repository. The dataset contains a range of handwritten images for numerals (0-9). Followings are some samples from the MFEAT handwritten database:

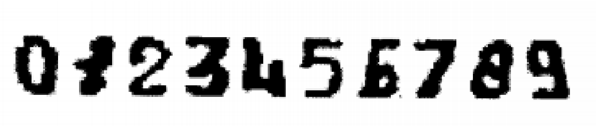


Figure 2-18: The MFEAT Database samples

The dataset can be retrieved from:

(https://archive.ics.uci.edu/ml/datasets/Multiple+Features)

(9) The USPS Handwritten Database

The USPS handwritten digit dataset is extracted from the UCI repository. The dataset contains 9298 handwritten digit images. Followings are some samples from the USPS database:

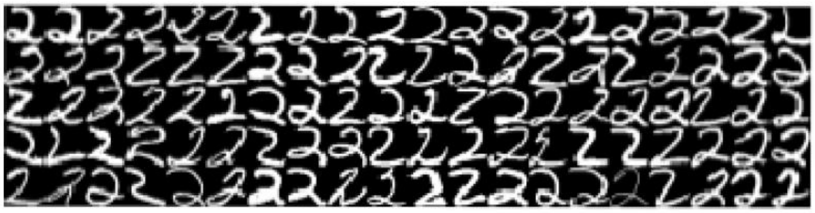


Figure 2-19: The USPS Database samples

The dataset can be retrieved from:

(http://www-stat-class.stanford.edu/~tibs/ElemStatLearn/data.html)

(10) The LUNG Cancer Database

The LUNG image dataset was published by Hong and Young in 1992. The dataset consists of 203 images from 5 classes. Each class has 139, 21, 20, 6, 17 samples respectively, and each sample has 12600 genes in it. Following is an example of the LUNG cancer database:

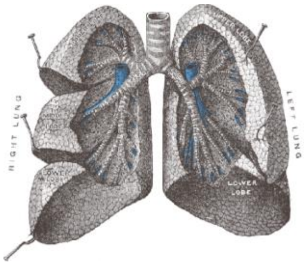


Figure 2-20: The LUNG Cancer Database sample

The dataset can be retrieved from:

(https://archive.ics.uci.edu/ml/datasets/lung+cancer)

(11) TOX Database

The TOX image dataset consists of 171 image samples from 4 classes (four main human being cardiomyocytes ionic currents).

The dataset can be retrieved from:

(http://featureselection.asu.edu/datasets.php)

2.9 Conclusion

To sum up, this chapter has represented a global vision of the popularly used learning methods in image recognition filed, including the most typical dimensionality reduction approaches (PCA and LDA), the graph-based reconstruction approaches (LLE, ISOMAP, LE, and LPP). Then, there is an introduction of two efficient performance representation approaches, SR and LRR. These two approaches have been successfully applicated in computer vision and machine learning with remarkable achievements. All these proposed approaches are excellent starting points and footstone to our research.

Chapter 3 Joint Sparse Representation and Locality Preserving Projection

The Sparse Representation (SR) algorithm is among the mostly widely adopted techniques in different domains such as pattern recognition, gene expression, and machine learning. Numerous literature has demonstrated that SR works favorably in representing a target test sample. However, SR reconstruct the data structure without considering the data local neighborhood structure. Consequently, the reconstruct data graph may not properly represent the data intrinsic structure. On the other side, despite Locality Preserving Projection (LPP) operates excellent in recognizing data neighborhood relationship, the technique separates data structure learning and valuable feature determination apart from each other, and as a result, the accuracy of obtained feature subset is largely depending on the graph previously constructed. We propose a novel technique, named Joint Sparse Representation and Locality Preserving Projection (JSRLPP). This technique joints SR and LPP, and performs a similarity matrix to process the graph learning adaptively during feature extraction. We conduct extensive experiments on different datasets, and the results show that JSRLPP outperforms the state-of-the-art techniques.

3.1 Introduction

Today, data in the real applications are represented by increasingly high-dimensional feature space, and there are multiple linear dimensionality reduction approaches having been proposed in recent decades to reduce the dataset dimensions, such as PCA (Turk & Pentland, 1991), LDA (Belhumeur, Hespanha, & Kriegman, 1997), and Independent Component Analysis (ICA) (Bartlett, 1998), etc. Although these linear approaches are simplistic and straightforward to calculate, the performance of these methods degenerates when the dataset is nonlinear. To deal with the nonlinear dataset, a set of kernel-based approaches (Muller, Mika, Ratsch, Tsuda, & Scholkopf, 2001) (Yang M. , 2002) (Yang, Frangi, Yang, Zhang, & Jin, 2005) (Yang, Gao, Zhang, & Yang, 2005) have been proposed in this domain. In those kernel-based approaches, the nonlinear data is transformed into a higher-dimensional feature space to exert the linear algorithms. Kernel-based approaches solved the problem of processing nonlinear dataset. However, it induces dimensional disasters during the calculation. In addition, there are various selections of kernel-based approaches, and it is difficult to define the optimal one. Then, in the following years, plenty of manifold exploring approaches blossom, such as the Neighborhood Preserving Embedding (NPE) (He, Cai, Yan, & Zhang, 2005), LPP (He & Niyogi, 2003) (Xu, Zhong, Yang, & Zhang, 2010), LE (Belkin & Niyogi, 2003), LLE (Roweis & Saul, 2000), and following with their improved extension approaches. These approaches successfully resolved the problem of preserving the fundamental data structure, and out-of-sample problem. However, these approaches employ a k-Nearest Neighbor or ε-Ball to form the data structure, which is sensitive to parameters and noises. Then, (Wright, Yang, Ganesh, Sastry, & Ma, 2009) realized that SR achieves impressive performance on image recognition. Based on the SR, numbers of extension efforts have been made: Non-Negative Low-Rank and Sparse Graph (NNLRS-graph) (Zhuang, et al., 2012) approach learns the data structure graph adaptively with sparse and Low-Rank Constraints; Unsupervised Large Graph Embedding (ULGE) (Nie, Zhu, & Li, 2017) combines a Low-Rank Representation approximation with traditional data structure graph. In SR, the data structure graph is built by applying the l_1 graph. Most of the SR approaches construct the graph adaptively, namely, the neighborhoods and weights are self-chosen, there is no pre-determined parameter required in the construction. Numbers of extensions, such as Sparsity Preserving Projection (SPP) (Qiao, Chen, & Tan, 2010), and Locality Preserving Projection based on the l_1 graph (LPPl_1) (Liu, Yin, & Jin, 2010), have been proposed afterward.

The approaches stated above are sharing a common problem. All these graph-based approaches have two separated procedures: graph learning, and projection learning. They all construct a data structure graph (using k-nearest neighbor, ε-ball or l_1 graph) first, and then the projecting matrix can be calculated based on the fixed data structure graph. Therefore, the accuracy of feature extraction result is strongly relying on the graph constructed. In other word, the fixed predetermined data structure graph may not be the optimal preference for a specific task. In order to overcome this problem, a framework that can simultaneously carry the graph construction and feature extraction is expected.

In this chapter, we proposed a novel unsupervised feature extraction approach, the JSRLPP. Since the SR can obtain the similarity matrix adaptively, it is appreciable to implement this similarity matrix in LPP to preserve both the global structure and local neighborhood relationships for the dataset. In addition, by combining the SR with LPP, we can simultaneously learn the sparse similarity matrix and the projection matrix in a single framework. The experimental outcomes illustrated that the approach we proposed is performing superior to other similar approaches.

The contributions of JSRLPP are as followings:

1. JSRLPP has joint the traditional two-steps of graph-based feature extraction into one single step. Therefore, the data structure graph learning and feature extraction can be conducted simultaneously. Namely, the data structure graph is continuously updated with the changing value of feature extraction variable. As a consequence, JSRLPP is not sensitive to parameters and variables in this framework, which enhances the performance with each other.
2. JSRLPP unifies SR and LPP for feature extraction. It effectively captures the data intrinsic structure and preserves this structure while projecting the image data into a lower-dimensional space.
3. JSRLPP is able to be extended to other graph-based dimensionality reduction approaches. LPP approach in this framework can be replaced by other graph-based approaches while holding the graph construction of JSRLPP unchanged.

This chapter is organized as followings:

An overview of widely used similar learning approaches will be introduced in Section 3.2. The basic idea of JSRLPP framework will be provided in Section 3.3. In Section 3.4, we are going to present the optimization processes, and Section 3.5 will discuss the computation complexity and convergence of the learning approach. Then, Section 3.6 will demonstrate the comprehensive experiments and experimental results of four public datasets. Ultimately, the conclusion will be given in Section 3.7.

3.2 Related Methods

On the one hand, SR is an approach that has been wildly implemented in different applications, yet it only focusses on data global structure in the reconstruction process. On the other hand, previous studies on LPP are separately processing the data structure construction and valuable features determination.

JSRLPP jointly integrates the SR algorithm and LPP algorithm where SR works to maintain data global structure and LPP is used to preserve data local manifold structure. In addition, in JSRLPP, we employ a similarity matrix to learn the data structure such that the graph can be obtained adaptively.

We have introduced the concept of SR in Section 2.5, and LPP in Section 2.4.4, this section we focus on the implementation of similarity matrix.

3.2.1 Similarity Matrix

Suppose there is a training set $\boldsymbol{X} = [x_1, x_2, \dots, x_n] \in R^{d \times n}$, n denotes the number of samples, and the i-th sample is expressed as $\boldsymbol{x}_i \in R^d, i = 1, 2, \dots, n$.

SR algorithm is solving the problem:

$$\min_{\boldsymbol{s}} \|\boldsymbol{s}\|_1 \qquad s.t. \;\; \boldsymbol{x} = \boldsymbol{X}\boldsymbol{s}$$

(3-1)

where $\|\cdot\|_1$ is a l_1-norm.

Then the LPP problem is shown as:

$$\min \frac{1}{2} \sum_{ij} \|\boldsymbol{y}_i - \boldsymbol{y}_j\|_2^2 S_{ij}$$

(3-2)

In the Function (3-1) $\boldsymbol{y}_i, \boldsymbol{y}_j \in R^c$ are the projection points of original $\boldsymbol{x}_i, \boldsymbol{x}_j$, c represents the dimensional space that will be reduced to. $\boldsymbol{S} \in R^{n\times n}$ denotes a similarity matrix. It is noticeable that in this function, the similarity matrix is the same matrix that we implement in SR. The elements in S are calculated by using k-Nearest Neighbors or ε-Balls Method:

$$S_{ij} = \begin{cases} \exp\left(-\left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|^2 / t\right), & \text{if } \boldsymbol{x}_i \in N_k(x_j), \text{or } \boldsymbol{x}_j \in N_k(x_i); \\ 0, & \text{otherwise} \end{cases}$$

(3-3)

Or,

$$S_{ij} = \begin{cases} \exp\left(-\left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|^2 / t\right), & \text{if } \left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|^2 < \varepsilon; \\ 0, & \text{otherwise} \end{cases}$$

(3-4)

where $\boldsymbol{x}_i \in N_k(x_j)$ represents that $\boldsymbol{x}_i$ is one of the k neighbors of $\boldsymbol{x}_j$. ε is a small pre-set constant. Since it is expected that the neighborhood structure between data samples to be maintained after projection, the objective function of LPP is:

$$\min \frac{1}{2} \sum_{ij} \left\| \boldsymbol{W}^T \boldsymbol{x}_i - \boldsymbol{W}^T \boldsymbol{x}_j \right\|_2^2 S_{ij}$$

(3-5)

The Problem (3-5) is under the constraint, $\boldsymbol{y}^T \boldsymbol{D} \boldsymbol{y} = \boldsymbol{I}$, where $\boldsymbol{D}$ denotes a diagonal matrix that the entries are the sum of columns of $\boldsymbol{S}$. Thus $D_{ii} = \sum_j S_{ij}$ or $D_{ii} = \sum_j S_{ji}$. Then, the Function (3-5) can be rewritten as:

$$\min_{\boldsymbol{W}} \boldsymbol{W}^T \boldsymbol{X} \boldsymbol{L} \boldsymbol{X}^T \boldsymbol{W}$$

$$\text{s.t.} \boldsymbol{W}^T \boldsymbol{X} \boldsymbol{D} \boldsymbol{X}^T \boldsymbol{W} = \boldsymbol{I}$$

(3-6)

$\boldsymbol{L}$ is the Laplacian matrix that $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{S}$. The Problem (3-6) can be resolved by using the Lagrange function.

Definitions and Proofs:

Definition 9.: Let $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_n] \in R^{d \times n}$, *n is the number of samples.* $\forall \boldsymbol{x}_i \in \boldsymbol{X}$ *is a random data point. The locality preserving projection is defined as:*

$$min\frac{1}{2}\sum_{ij}\|\boldsymbol{y}_i - \boldsymbol{y}_j\|_2^2 S_{ij}$$

Lemma 1: Given two samples, $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, are from the same class, the result of $\left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|^2$ will be small by employing the Euclidean distance.

Proof: The distance between sample points can be defined as neighborhood measurement. The smaller the distance, the more similar the sample points are. When sample $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are from the same class, we have $\left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|^2 < \varepsilon$, and the Euclidean distance between them is small.

Lemma 2: Given two samples, $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, are from the same class, the reconstructed point $\sum_{i=1}^{n}\boldsymbol{Xs}_i$ and $\sum_{j=1}^{n}\boldsymbol{Xs}_j$ will be classified into the same class by applying Locality Preserving Projection mapping.

Proof: According to Locality Preserving Projection, the distance between samples after projection is going to be consisted with the original structure. Thus, we have:

$$if\ \left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|^2 \le \varepsilon$$

$$\therefore \quad \left\|\sum_{i=1}^{n}\boldsymbol{Xs}_i - \sum_{j=1}^{n}\boldsymbol{Xs}_j\right\|^2 \le \varepsilon'$$

In addition, based on Lemma 1, if $\sum_{i=1}^{n}\boldsymbol{Xs}_i$ and $\sum_{j=1}^{n}\boldsymbol{Xs}_j$ are in the same class, $\left\|\sum_{i=1}^{n}\boldsymbol{Xs}_i - \sum_{j=1}^{n}\boldsymbol{Xs}_j\right\|^2$ will be small.

Theorem 1: Given two samples, $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, are from the same class, $\sum_{i=1}^{n}\boldsymbol{Xs}_i$ and $\sum_{j=1}^{n}\boldsymbol{Xs}_j$ will be in the same class, and $\left\|\sum_{i=1}^{n}\boldsymbol{Xs}_i - \sum_{j=1}^{n}\boldsymbol{Xs}_j\right\|^2$ will be small by applying Locality Preserving Projection.

Proof: Refers to Lemma 1 and Lemma 2, $\sum_{i=1}^{n}\boldsymbol{Xs}_i$ and $\sum_{j=1}^{n}\boldsymbol{Xs}_j$ are from the same class, and the result of $\left\|\sum_{i=1}^{n}\boldsymbol{Xs}_i - \sum_{j=1}^{n}\boldsymbol{Xs}_j\right\|^2$ is small.

3.3 Joint Sparse Representation and Locality Preserving Projection for Feature Extraction

As discussed in Section 3.1, the traditional LPP feature extraction approach is split into two independent steps, and the accuracy of the feature extraction result is profoundly relying on the predefined data structure graph. To overcome this drawback, JSRLPP combines SR and LPP to learn the similarity matrix $\boldsymbol{S}$ and projection matrix $\boldsymbol{W}$ concurrently and adaptively. With this adaptive framework, the data intrinsic structure can be maintained.

3.3.1 JSRLPP

JSRLPP takes the benefits from SR and LPP, to adaptively learn a unified framework that assures the consistency between original data structure and the projection data structure. The framework has taken the whole training set as an over-complete dictionary, and reconstruct each training sample with all other samples based on the similarity matrix $\boldsymbol{S}$. Since $\boldsymbol{S}$ represents the role of revealing the similarity between sample points, JSRLPP regards this coefficient as a weight coefficient as well. In more detail, it is expected that if image $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are similar, $\boldsymbol{x}_j$ should be chosen in the base to reconstruct $\boldsymbol{x}_i$, and the weight $\boldsymbol{x}_j$ will be heavy and represented by the value of S_{ij}. Table (3-1) is presenting the variables and definitions in JSRLPP:

Table 3-1: Variables of JSRLPP

Variables	Definitions
$\boldsymbol{X} = [\boldsymbol{x_1}, \boldsymbol{x_2}, \dots, \boldsymbol{x_n}] \in \boldsymbol{R}^{\boldsymbol{d \times n}}$	Training Set
$\boldsymbol{x_i} \in \boldsymbol{R^d}$	A training sample in X
$\boldsymbol{d}$	The original dimension of dataset
$\boldsymbol{n}$	The number of training samples
$\boldsymbol{S} \in \boldsymbol{R}^{\boldsymbol{n \times n}}$	Similarity matrix of dataset
$\boldsymbol{W} \in \boldsymbol{R}^{\boldsymbol{d \times c}}$	Projection matrix
$\boldsymbol{c}$	The number of dimensions after reconstruction

JSRLPP framework is shown as:

$$\min_{\boldsymbol{S},\boldsymbol{W}} \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{1}{2} \left\| \boldsymbol{W}^T \boldsymbol{x}_i - \boldsymbol{W}^T \boldsymbol{x}_j \right\|_2^2 S_{ij} + \frac{\lambda_1}{2} \|\boldsymbol{X} - \boldsymbol{XS}\|_F^2 + \lambda_2 \|\boldsymbol{S}\|_1$$

$$\text{s.t.} \qquad \boldsymbol{W}^T \boldsymbol{XDX}^T \boldsymbol{W} = \boldsymbol{I}, \boldsymbol{S} \geq \boldsymbol{0}, S_{ii} = 0$$

(3-7)

In the Function (3-7), $\|\cdot\|_F$ denotes a Frobenius norm, λ_1 and λ_2 are two balance parameters. $\boldsymbol{S}$ is an asymmetric matrix, it will be replaced by $\boldsymbol{S_s} = (\boldsymbol{S} + \boldsymbol{S}^T)/2$ in calculating $\boldsymbol{D}$. Thus, $\boldsymbol{D}_{S_{ii}} = \sum_j \boldsymbol{S}_{S_{ij}}$ or $\boldsymbol{D}_{S_{ii}} = \sum_j \boldsymbol{S}_{S_{ji}}$.

The main intention of JSRLPP is as if two samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are from the same class, then, according to the idea of SR, $\boldsymbol{x}_j$ should be selected in representing $\boldsymbol{x}_i$. In addition, since two points are close enough, according to the concept of LPP, the value (weight) of S_{ij} will be relatively large.

The first term in Function (3-7), $\left\|\boldsymbol{W}^T\boldsymbol{x}_i - \boldsymbol{W}^T\boldsymbol{x}_j\right\|_2^2$, measures the distance between samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. $\boldsymbol{S}$ is the similarity matrix, and the element S_{ij} in $\boldsymbol{S}$ denotes the value of weight for sample $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. The second term in Function (3-7) followed by λ_1 is a SR, which aims to minimize representation error in the process of learning similarity matrix $\boldsymbol{S}$. The third term in Function (3-7) followed by λ_2 is a constraint term, which aims to sparse the similarity matrix $\boldsymbol{S}$. This framework as a whole maintains the data intrinsic relationships in the projection process.

3.4 Optimization

The Problem (3-7) is hard to solve directly since the projection matrix $\boldsymbol{W}$ and the similarity matrix $\boldsymbol{S}$ are unknown. Inspired by (Zhuang, et al., 2012) (Xu, Fang, Wu, Li, & Zhang, 2016), we would prefer to employ an auxiliary variable $\boldsymbol{Z} = \boldsymbol{S}$ to resolve the problem. Then the objective problem can be transformed into:

$$\min_{\boldsymbol{S},\boldsymbol{Z},\boldsymbol{W}} \sum_{i=1}^{n}\sum_{j=1}^{n}\frac{1}{2}\left\|\boldsymbol{W}^T\boldsymbol{x}_i - \boldsymbol{W}^T\boldsymbol{x}_j\right\|_2^2 S_{ij} + \frac{\lambda_1}{2}\|\boldsymbol{X}-\boldsymbol{XS}\|_F^2 + \lambda_2\|\boldsymbol{Z}\|_1$$

$$\text{s.t.} \quad \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{D}_s\boldsymbol{X}^T\boldsymbol{W} = \boldsymbol{I}, \boldsymbol{Z} = \boldsymbol{S}, \boldsymbol{S} \geq \boldsymbol{0}, S_{ii} = 0$$

(3-8)

The problem can be solved by using an Augmented Lagrange Multiplier (ALM):

$$\boldsymbol{L}(\boldsymbol{S},\boldsymbol{Z},\boldsymbol{W}) = \min_{\boldsymbol{S},\boldsymbol{Z},\boldsymbol{W}} \sum_{i=1}^{n}\sum_{j=1}^{n}\frac{1}{2}\left\|\boldsymbol{W}^T\boldsymbol{x}_i - \boldsymbol{W}^T\boldsymbol{x}_j\right\|_2^2 S_{ij} + \frac{\lambda_1}{2}\|\boldsymbol{X}-\boldsymbol{X}\boldsymbol{S}\|_F^2 + \lambda_2\|\boldsymbol{Z}\|_1$$

$$+\frac{\mu}{2}\left\|\boldsymbol{S}-\boldsymbol{Z}+\frac{\boldsymbol{P}}{\mu}\right\|_F^2$$

$$s.t. \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{D}_{\boldsymbol{s}}\boldsymbol{X}^T\boldsymbol{W} = \boldsymbol{I}, \boldsymbol{S} \geq \boldsymbol{0}, S_{ii} = 0$$

(3-9)

In the Function (3-9), $\mu > 0$ is a penalty parameter, and $\boldsymbol{P}$ is a Lagrange Multiplier. This ALM Function (3-9) can be resolved by employing the Alternating Direction Method of Multiplier (ADMM) (Boyd, Parikh, Chu, Peleato, & Eckstein, 2010). In ADMM, each variable will be updated with all other variables unchanged. The main steps are shown as followings.

3.4.1 Update $\boldsymbol{W}$:

The problem regarding $\boldsymbol{W}$ can be described as:

$$\boldsymbol{L}(\boldsymbol{W}) = \min_{\boldsymbol{W}} \sum_{i=1}^{n}\sum_{j=1}^{n}\frac{1}{2}\left\|\boldsymbol{W}^T\boldsymbol{x}_i - \boldsymbol{W}^T\boldsymbol{x}_j\right\|_2^2 S_{ij}$$

$$s.t. \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{D}_{\boldsymbol{s}}\boldsymbol{X}^T\boldsymbol{W} = \boldsymbol{I}$$

(3-10)

Then, we take the partial derivative of $\boldsymbol{W}$:

$$\boldsymbol{L}(\boldsymbol{W}) = \min_{\boldsymbol{W}} Tr\,(\boldsymbol{W}^T\boldsymbol{X}\boldsymbol{L}_{\boldsymbol{s}}\boldsymbol{X}^T\boldsymbol{W})$$

$$\text{s.t.}\ \ \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{D}_{\boldsymbol{s}}\boldsymbol{X}^T\boldsymbol{W} = \boldsymbol{I}$$

(3-11)

In the Function (3-11), $\boldsymbol{L_s} = \boldsymbol{D_s} - \boldsymbol{S_s}$ is the Laplacian matrix. It is apparent that $\boldsymbol{W}$ can be obtained by solving the above-generalized eigenvector problem corresponding to the c smallest eigenvalues that larger than 0.

3.4.2 Update $\boldsymbol{S}$:

The problem regarding $\boldsymbol{S}$ can be described as:

$$\boldsymbol{L}(\boldsymbol{S}) = \min_{\boldsymbol{S}} \sum_{i=1}^{n}\sum_{j=1}^{n}\frac{1}{2}\left\|\boldsymbol{W}^T\boldsymbol{x}_i - \boldsymbol{W}^T\boldsymbol{x}_j\right\|_2^2 S_{ij} + \frac{\lambda_1}{2}\|\boldsymbol{X} - \boldsymbol{X}\boldsymbol{S}\|_F^2 + \frac{\mu}{2}\left\|\boldsymbol{S} - \boldsymbol{Z} + \frac{\boldsymbol{P}}{\mu}\right\|_F^2$$

$$\text{s.t.}\qquad \boldsymbol{S} \geq \boldsymbol{0}, S_{ii} = 0$$

(3-12)

Since we assumed $\boldsymbol{W}$ and $\boldsymbol{Z}$ are fixed, we let

$$\boldsymbol{K}_{ij} = \frac{1}{2}\left\|\boldsymbol{W}^T\boldsymbol{x}_i - \boldsymbol{W}^T\boldsymbol{x}_j\right\|_2^2$$

(3-13)

Then, the problem can be converted into:

$$\boldsymbol{L}(\boldsymbol{S}) = \min_{\boldsymbol{S}} Tr(\boldsymbol{K}^T\boldsymbol{S}) + + \frac{\lambda_1}{2}\|\boldsymbol{X} - \boldsymbol{X}\boldsymbol{S}\|_F^2 + \frac{\mu}{2}\left\|\boldsymbol{S} - \boldsymbol{Z} + \frac{\boldsymbol{P}}{\mu}\right\|_F^2$$

$$\text{s.t.} \qquad \boldsymbol{S} \geq \boldsymbol{0}, S_{ii} = 0$$

(3-14)

We take the partial derivative of $\boldsymbol{S}$, and apply the nonnegative projection on $\boldsymbol{S}$ (Zhuang, et al., 2012), then obtain $\boldsymbol{S}^*$ using:

$$\begin{cases} \boldsymbol{S}^* = (\lambda_1\boldsymbol{X}^T\boldsymbol{X} + \mu\boldsymbol{I})^{-1}(\lambda_1\boldsymbol{X}^T\boldsymbol{X} + \mu\boldsymbol{Z} - \boldsymbol{P} - \boldsymbol{K}) \\ \boldsymbol{S}^* = \max(0, \boldsymbol{S}^*) \\ \boldsymbol{S}_{ii}^* = 0, i = 1, 2, \dots, n \end{cases}$$

(3-15)

3.4.3 Update *Z*:

The problem regarding $\boldsymbol{Z}$ can be described as:

$$\boldsymbol{L}(\boldsymbol{Z}) = \min_{\boldsymbol{Z}} \lambda_2\|\boldsymbol{Z}\|_1 + \frac{\mu}{2}\left\|\boldsymbol{S} - \boldsymbol{Z} + \frac{\boldsymbol{P}}{\mu}\right\|_F^2$$

(3-16)

Problem (3-16) can be solved by using the soft-threshold (shrinkage) operator (Cai, Cands, & Shen, 2008).

$$\boldsymbol{Z}^* = shrink\left(\boldsymbol{S} + \frac{\boldsymbol{P}}{\mu}, \frac{\lambda_2}{\mu}\right)$$

(3-17)

where $shrink(x, a) = signmax(|x| - a, 0)$.

3.4.4 update $\boldsymbol{P}$ and parameter $\boldsymbol{\mu}$:

The problems regarding $\boldsymbol{P}$ and μ are:

$$\begin{cases} \boldsymbol{P} = \boldsymbol{P} + \mu(\boldsymbol{S} - \boldsymbol{Z}) \\ \mu = \min(\rho\mu, \mu_{max}) \end{cases}$$

(3-18)

the $\rho > 1$ is the iteration step, and μ_{max} is a pre-set constant. Therefore, we can obtain the value of **P**.

To sum up, in JSRLPP, SR and LPP have been integrated to operate with each other. Our objective is to resolve the Problem (3-7), and the computational processes are demonstrated in **Algorithm 3-1**. During the optimization, we update the projection matrix $\boldsymbol{W}$ by solving the Problem (3-11) while holding all other variables unchanged. Then, we update $\boldsymbol{S}$ by solving the Problem (3-15) while keeping other variables fixed. In the iteration, the result of $\boldsymbol{W}$ will be renewed by the updated value of $\boldsymbol{S}$, while $\boldsymbol{S}$ will also be refreshed based on the updated value of $\boldsymbol{W}$. Therefore, in JSRLPP, $\boldsymbol{W}$ will affect $\boldsymbol{S}$, and $\boldsymbol{S}$ will influence $\boldsymbol{W}$ during the iteration. However, in previous proposed conventional approaches, $\boldsymbol{S}$ will be calculated and fixed at the very first step, then, following with the feature extraction operation. Therefore, the result of feature extraction is strongly relying on the predefined data structure graph. Since JSRLPP updates $\boldsymbol{S}$ in each iteration, the graph is running updated towards the optimal, an impressive feature extraction result is expected.

Algorithm 3-1: Solving the JSRLPP by IALM

```
Input: $X \in R^{d\times n}$, $\lambda_1$, $\lambda_2$, $c$;
Initialization: $S = S_{knn}$, $Z = S$, $P = 0$, $\mu_{max} = 10^7$, $\mu = 0.1$, $\rho = 1.01$, $\varepsilon = 10^{-7}$;
1. While not converged, do:
    2. Fixing other variables, update the projection matrix $W$ by solving problem (3-11)
    3. Fixing other variables, update the projection matrix $S$ by solving problem (3-15)
    4. Fixing other variables, update the projection matrix $Z$ by solving problem (3-17)
    5. Update the multipliers and parameters by solving problem (3-18)
    6. Check the convergence condition by (3-19):
           $\|S - S_{old}\|_F + \|Z - Z_{old}\|_F + \|W - W_{old}\|_F < \varepsilon$
7. end while
Output: $W$, $S$
```

3.5 Analysis of JSRLPP

The more comprehensive analysis of JSRLPP will be presented in this section, including the computational complexity, convergence, similarity matrix variations and the relationship with other similar approaches.

3.5.1 Computational Complexity

The most challenging parts in the computation are step 2 and step 3 in the **Algorithm (3-1)**. This difficulty is mainly because these parts contain the eigenvalue decomposition (EVD) and inverse operation. In step 2, the EVD is applied to a matrix that lies in a $d \times d$

dimensional space, and the computational complexity is $\boldsymbol{O}(d^3)$. Then, in step 3, the matrix inversion is applied in an $n \times n$ dimensional space, and the computational complexity is $\boldsymbol{O}(n^3)$. As a result, the total computational complexity of JSRLPP is $\boldsymbol{O}(\tau(d^3 + n^3))$, where τ is the number of iterations.

3.5.2 Convergency Analysis

The convergence of JSRLPP is tested by running it with the 1-Nearest Neighbor classifier (1NN) on four image datasets, including face images (Extended Yale B database, AR database, ORL database) and object images (COIL-20 database). These datasets have been introduced in Section 2.4. The convergence can be measured by the objective Function (3-19), which sums up all the variable changes:

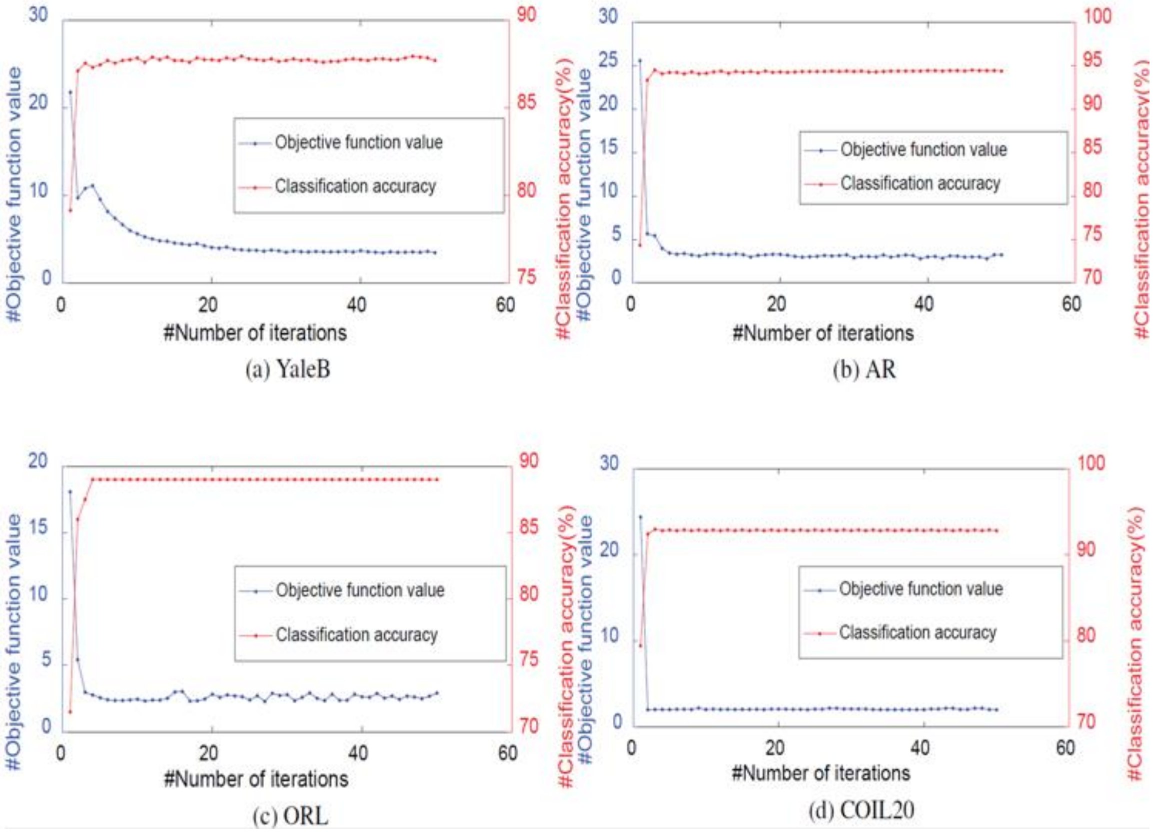
$$obj = \|\boldsymbol{S} - \boldsymbol{S}_{old}\|_F + \|\boldsymbol{Z} - \boldsymbol{Z}_{old}\|_F + \|\boldsymbol{W} - \boldsymbol{W}_{old}\|_F$$

(3-19)

where $\boldsymbol{S}_{old}$, $\boldsymbol{Z}_{old}$, and $\boldsymbol{W}_{old}$ are the values set in the previous iteration step, and $\boldsymbol{S}$, $\boldsymbol{Z}$, and $\boldsymbol{W}$ are variable values calculated in the current iteration.

The maximum iteration of this framework has been set to be 50 times. Figure (3-1) demonstrates the value of Problem (3-19) and classification accuracy along with iterations. It is apparent that the result of the objective Function (3-19) declines dramatically during the initial few steps, and then, fluctuations steadily. At the meantime, the value of classification accuracy increases rapidly within the first three iterations and stays stable.

Especially for the dataset COIL20 in Figure 3-1(d), JSRLPP is converging after two iterations. To sum up, JSRLPP has fast convergence capacity and is robust on different types of datasets.



(a) YaleB

(b) AR

(c) ORL

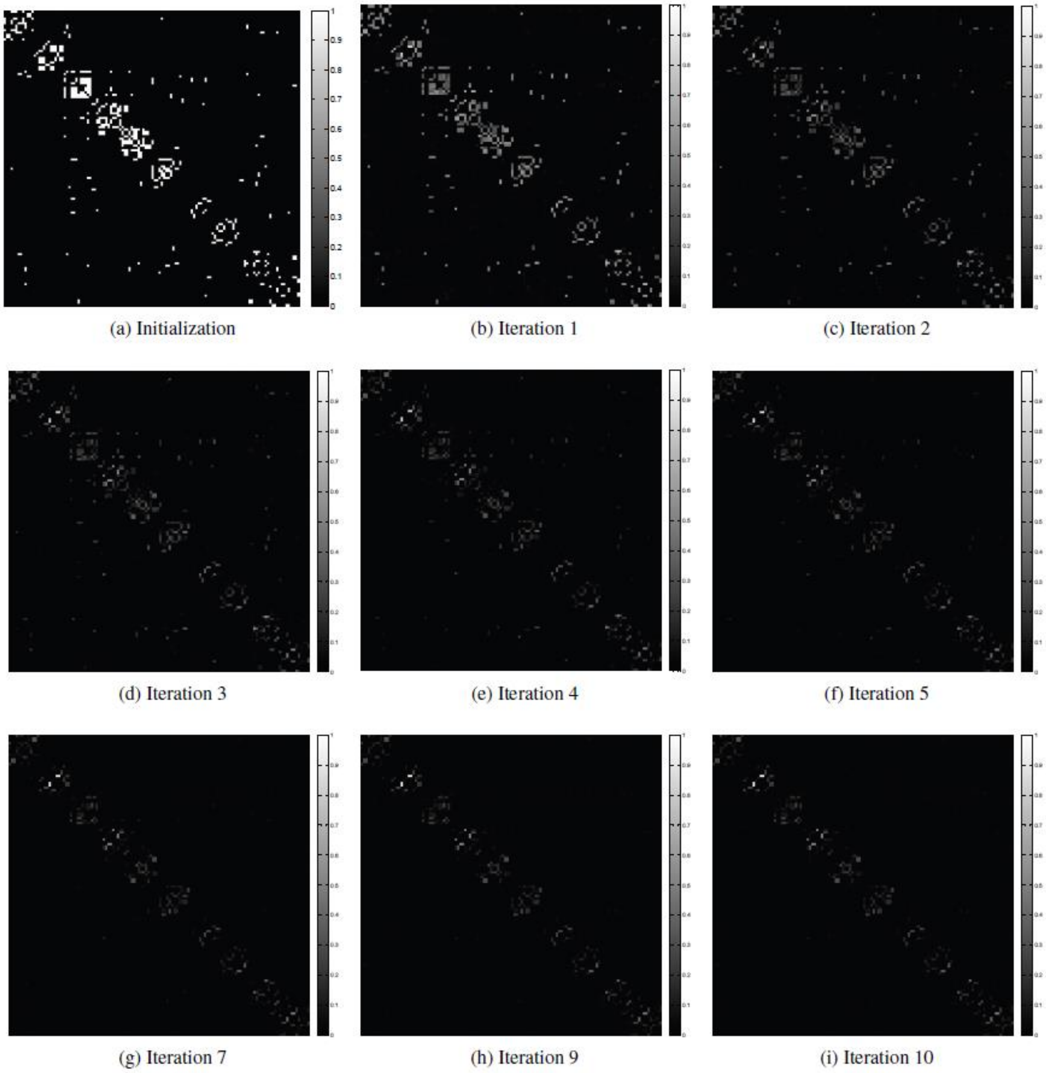
(d) COIL20

Figure 3-1: The Objective Function Value and Classification Accuracy (%)

3.5.3 Similarity Matrix Analysis

The similarity matrix ***S*** measures the relationship between two sample points. Compared with conventional LPP, in which the similarity matrix will be calculated at the very beginning step and will be unchanged in the subsequence procedures, JSRLPP undertakes the data structure learning and projection matrix learning simultaneously. Therefore, the data structure graph keeps adjusted with the entire framework during the calculation processes. Figure (3-2) demonstrates the changing in value of $\boldsymbol{S}$ in the iterations on the AR face image database. The figure only displays the first 100 rows and columns for $\boldsymbol{S}$. During the experiment processes in Figure (3-2), 10 sample images have been randomly selected from each class to build the training set. Since $\boldsymbol{S}$ is the similarity matrix that consists of weights between samples, the value of $\boldsymbol{S}$ will be relatively considerable for those between within-class samples than those between-class samples. Thus, the similarity matrix presents a block-diagonal.

In Figure (3-2) (a), it is clear that there are few undesirable similarities between samples that contaminate the data structure initially. Then, from the fourth iteration, the scatters aside disappear, which indicates the effective elimination of similarities between samples from different classes of JSRLPP. Along with the iterations, $\boldsymbol{S}$ becomes optimal and stays stable.



(a) Initialization (b) Iteration 1 (c) Iteration 2

(d) Iteration 3 (e) Iteration 4 (f) Iteration 5

(g) Iteration 7 (h) Iteration 9 (i) Iteration 10

Figure 3-2: The Value of Similarity Matrix on AR Face Dataset

3.5.4 Comparison with Related Approaches

JSRLPP has shown its impressive property on fast convergence and robustness on different kinds of datasets in previous sections. In this section, we are going to compare JSRLPP with similar approaches, LPP and LPPl_1.

- Compared with LPP

LPP is one of the most popular dimensionality reduction approaches that intends to maintain local neighborhood relationships. However, in LPP, the projection matrix accuracy is unilaterally affected by the pre-defined similarity matrix. JSRLPP is built based on the LPP framework. In JSRLPP, besides its ability to preserve the locality information of original dataset, it also has an advanced property of learning the data structure and projection matrix within a single step. Instead of creating the similarity matrix using k-nearest neighbors or ε-balls, JSRLPP determines the similarity matrix by using the entire training set with a sparsity constraint. JSRLPP will be transformed into LPP when parameter λ_1, λ_2 are set to be zero and the value of the similarity matrix is fixed.

- Compared with LPP$\mathbf{l_1}$

In LPPl_1, it constructs an l_1-graph first, and then, the projection matrix will be computed based on the pre-built graph. LPPl_1 determines neighborhood relationships and weights about the data structure graph adaptively, which indicates that it does not need to employ the k-Nearest Neighbors or ε-Balls in structure reconstruction. But, in LPPl_1, the graph plotted is staying consistent as other conventional methods after the graph has been determined.

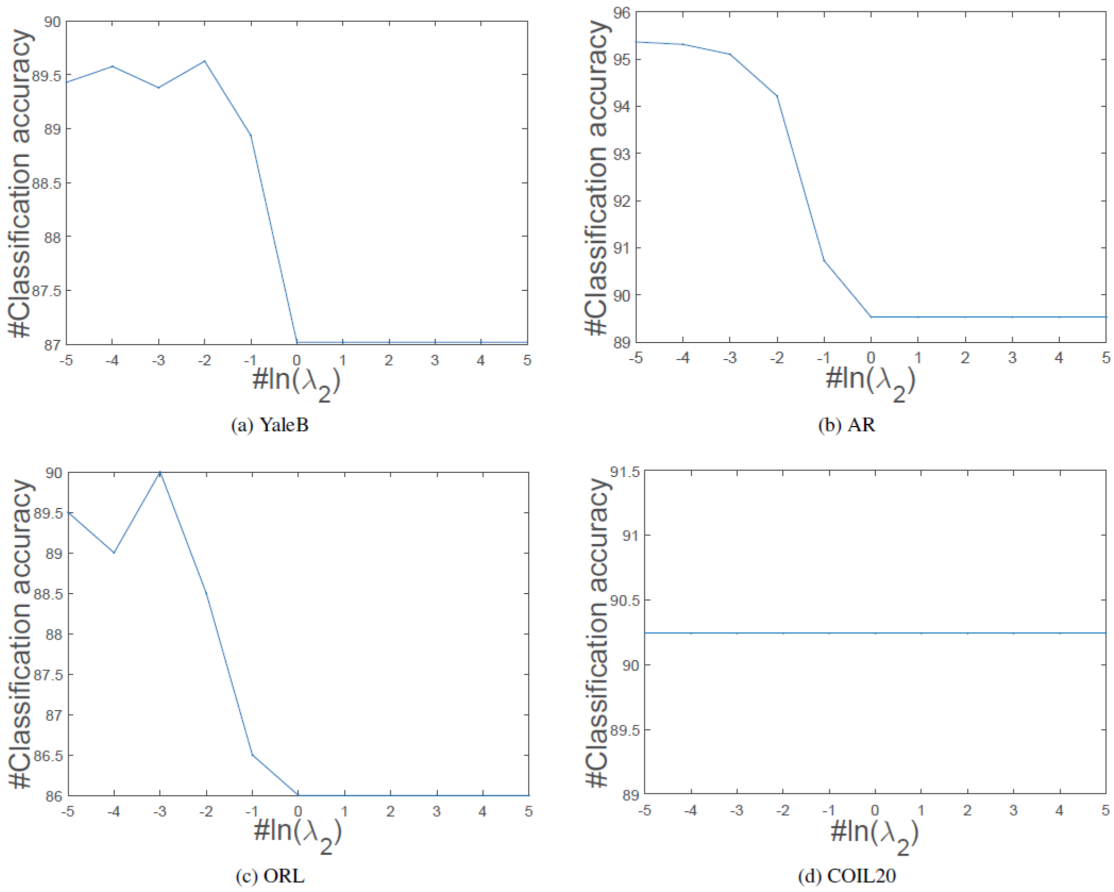
In JSRLPP, the learned graph will be modified along with the iterations and is going to obtain the optimal solution to satisfy the optimization objective eventually. JSRLPP will be transformed into LPPl_1 when the parameter λ_1, λ_2 are set to be zero and the value of the similarity matrix to an l_1-graph is fixed.

3.6 Experiments and Analysis

JSRLPP has been tested on four publicly datasets, i.e., Extended Yale B face database, AR face database, ORL face database, and COIL-20 object database. This section will discuss the selection of parameter first, then will compare the result of JSRLPP with PCA, LPP, NPE, and LPPl_1.

3.6.1 Parameter Selection

JSRLPP has three parameters, λ_1, λ_2, and c. Since there are no guidelines to value parameters, the parameters are determined by employing a grid-search strategy for each database. In the experiments, we randomly select a certain number of samples from each class to build the training set. The values of parameter λ_1 and λ_2 are selected from the range $\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}, 10^{4}, 10^{5}\}$, c is selected from candidate set $\{50, 100, 150, 200, 250, 300, 350, 400, 450, 500\}$. The range will become smaller and detailed in the calculation based on the performance. Each framework will operate on all the combinations of parameters and selects an appropriate combination for each database.



(a) YaleB

(b) AR

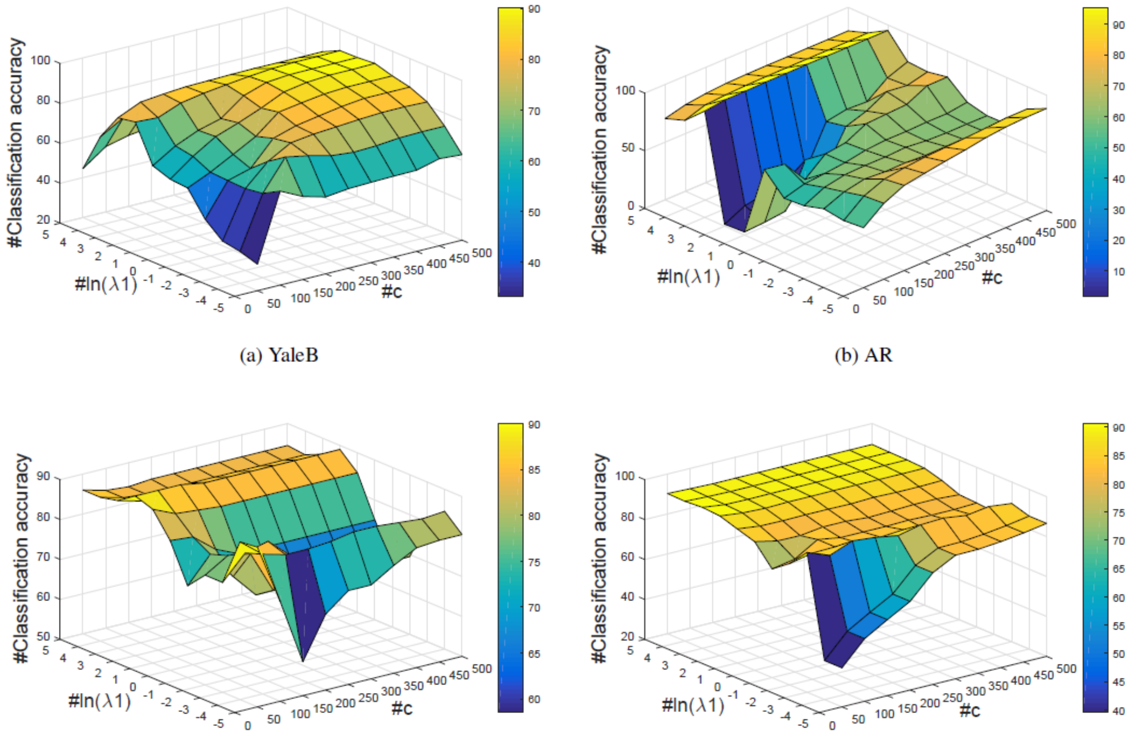
(c) ORL

(d) COIL20

Figure 3-3: The Classification Accuracy with Different Value of Lambda2

During the calculation, we found that the parameter λ_2 has a weak impact on the framework when we fix the value of λ_1 and c. Figure (3-3) illustrates the classification accuracy with the value selection of λ_2.

In Figure (3-3), it is clear that the parameter λ_2 is stable in COIL-20 database and performs better in small values on the Yale B database, AR database, and ORL database. Then, Figure (3-4) shows the grid-search strategy on λ_1 and c, while holding the value of λ_2 fixed.



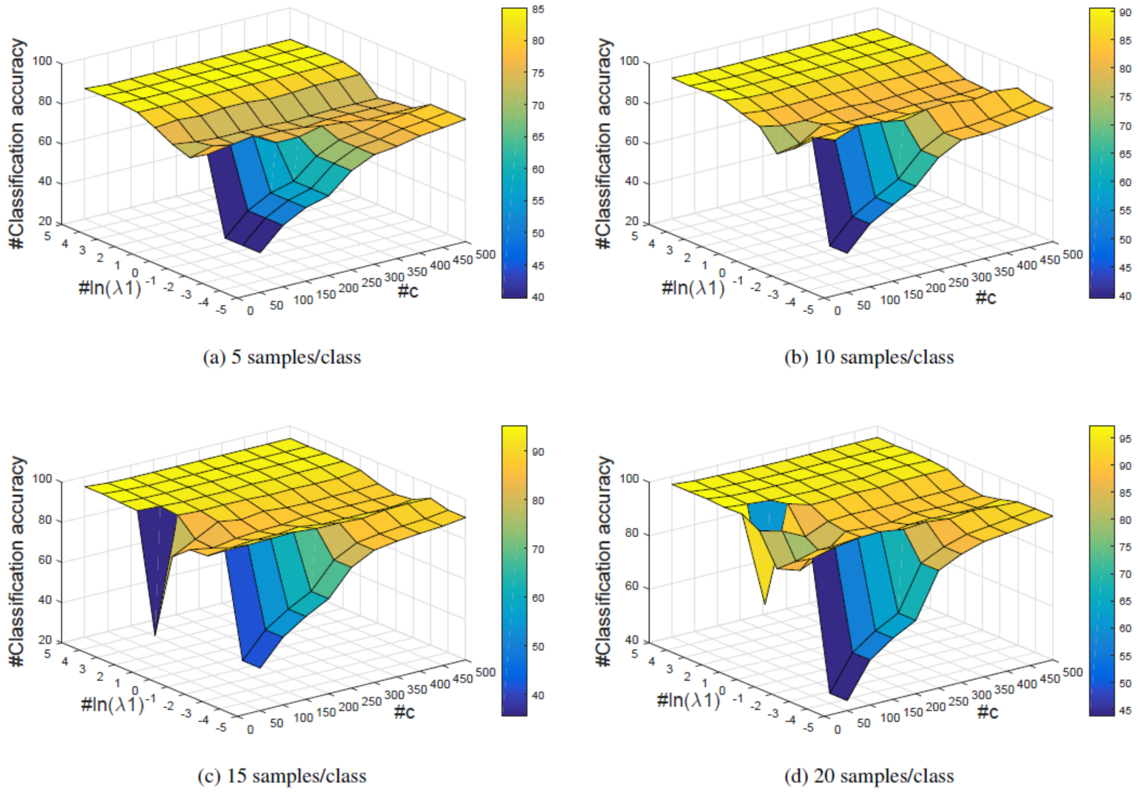
(a) YaleB

(b) AR

(c) ORL

(d) COIL20

Figure 3-4: The Classification Result with Different Value of Lambda1 and c



(a) 5 samples/class

(b) 10 samples/class

(c) 15 samples/class

(d) 20 samples/class

Figure 3-5: The Classification Results with Different Training Set Size

Figure (3-5) demonstrates the experimental outcomes on COIL-20 database regarding different volumes of the training set. Experiments have been conducted on selecting 5, 10, 15, and 20 samples per class respectively. From the figure, we observed that each experiment obtains approximately the same distribution of classification accuracy regardless of the number of samples that have been randomly selected.

By employing the grid-search strategy, we obtain the optimal parameter combinations for each database, i.e., λ_1, λ_2, and c are $10^1, 10^{-5}$ and 420 for database Yale B; $10^3, 10^5$ and 300 for database AR; $10^2, 10^{-3}$ and 50 for database ORL; and $10^5, 10^2$ and 30 for database COIL20. The comparison experiments in the next section are conducted based on these combinations of parameters.

3.6.2 Experimental Comparison

In this section, there is a presentation of extensive experiments on JSRLPP, and also, on those most similar approaches, including PCA, LPP, NPE, LPPl_1, and SPP. In these experiments, we employed k-nearest neighbors to map the data structure under the 'Binary' way and using 1-nearest neighbor classifier (NN) to measure the performs.

- Experiment on the Extended Yale B Face Database

Yale B is a face image database that contains 2414 frontal face images, which are taken from 38 individuals under various illumination conditions. Figure 3-6 demonstrates some samples from the database. Each individual has been taken 59-64 pictures and all have been transformed into grayscale 32×32 pixels.

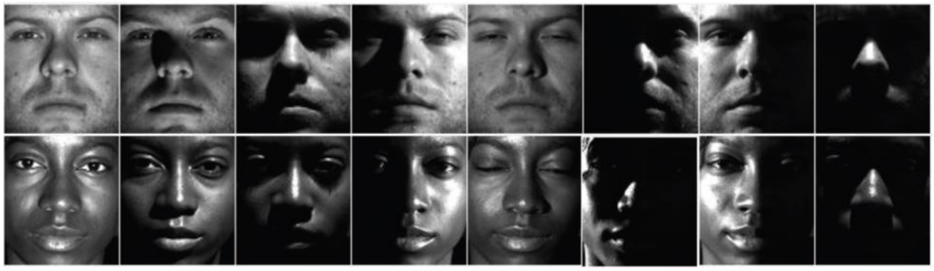


Figure 3-6: Samples from Yale B Database

We have conducted the experiments for four times per approach, and in each time, we randomly select 5, 7, 10, and 13 samples per class to build the training data respectively.

The experimental results and corresponding standard deviations have been listed in the Table (3-2):

Table 3-2: Experimental Results for Different Approaches on Yale B Database

No.	k	NN	PCA	LPP	NPE	LPPl_1	SPP	JSRLPP
5	5	37.12 ± 1.42	37.12 ± 1.42	62.51 ± 2.66	58.11 ± 2.04	71.30 ± 1.81	62.22 ± 1.98	**75.75** **± 1.54**
7	7	44.83 ± 1.26	44.83 ± 1.26	71.80 ± 2.09	67.82 ± 1.99	78.70 ± 1.20	71.08 ± 1.16	**83.10** **± 1.24**
10	7	53.24 ± 1.05	53.24 ± 1.05	79.13 ± 1.23	76.12 ± 1.45	84.83 ± 1.06	78.53 ± 1.01	**88.75** **± 0.76**
13	7	59.85 ± 1.11	59.80 ± 1.11	82.19 ± 1.20	79.27 ± 1.17	88.21 ± 1.07	82.86 ± 0.80	**90.36** **± 0.80**

In Table (3-2), the “No.” denotes to the number of training samples per class and “K” means the number of neighbors in employing the k-nearest neighbors.

From the above table, it is clear that the accuracy results are growing along with the expanding number of the training set. The performance of JSRLPP is much superior to all other methods, which presents an impressive classification accuracy. Then, it is LPPl_1 and SPP, which also obtained comparable good results. Additionally, it is worthwhile to mention that the result of PCA is faraway lower than other approaches, which is approximately the same as a baseline using 1-nearest neighbor classifier (NN) on the original data directly. The possible reason might be because PCA disregards the structure between classes and only focus on preserving the sample points information. Overall, JSRLPP achieves the highest accuracy rate, which indicates that the unified framework operates better in maintaining the dataset intrinsic structure, and is robust to the changing in illuminations.

- Experiment on AR Face Database

AR is a face image database that contains 4000 face images for 126 people. Every individual has been taken images on different expressions, illuminations, and occlusions. Figure (3-7) demonstrates some samples from the AR database.

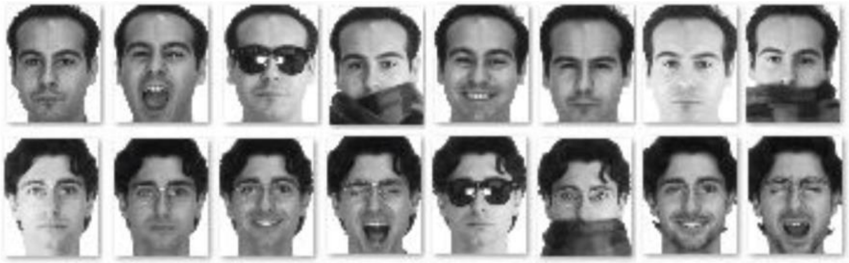


Figure 3-7: Samples from AR Database

In the experiments, we collected a subset of the database to be the input dataset, in which there are 26 faces images for 120 individuals to build the training set, 3120 images in total. All images have been transformed into grayscale 50 × 40 pixels. We have conducted the experiments for four times per approach, and in each time, we randomly select 8, 9, 10, and 11 images from each class to build the training set respectively. The experimental results and corresponding standard deviations have been presented in Table (3-3).

Table 3-3: Experimental Results for Different Approaches on AR Database

No.	**k**	**NN**	**PCA**	**LPP**	**NPE**	**LPPl_1**	**SPP**	**JSRLPP**
8	7	73.81 ± 1.06	72.95 ± 1.04	70.49 ± 1.36	74.26 ± 1.27	89.97 ± 0.95	88.06 ± 1.03	**90.91 ± 0.83**
9	7	76.81 ± 1.16	75.89 ± 1.11	72.31 ± 1.26	72.53 ± 1.77	91.99 ± 0.92	90.51 ± 0.75	**93.13 ± 0.99**
10	7	78.45 ± 1.00	77.60 ± 0.94	75.20 ± 1.23	68.16 ± 1.83	92.86 ± 0.86	91.75 ± 0.85	**94.15 ± 0.68**
11	7	80.62 ± 1.09	79.77 ± 1.14	76.92 ± 1.29	64.45 ± 1.38	94.27 ± 0.85	93.34 ± 1.03	**95.28 ± 0.82**

It is clear that JSRLPP achieves the highest accuracy rate, and indicates that JSRLPP is effective in preserving the data structure and is robust to various expressions and occlusions.

- Experiment on ORL Face Database

ORL is a face database that contains 400 face images for 40 individuals. Each individual has been taken images under different expressions and occlusions. In addition, these images are taken with a 20 degrees rotation. Figure (3-8) demonstrates some samples from the AR database. All images have been transformed into grayscale 32 × 32 pixels.

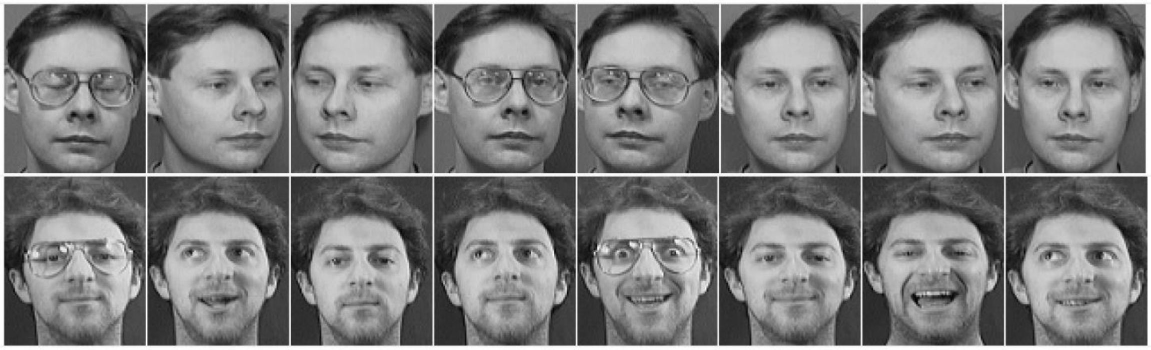


Figure 3-8: Samples from ORL Database

We have conducted the experiments for four times per approach. In every single time, we randomly select 3, 5, 6, and 7 samples from each class to build the training set respectively. The experimental results and corresponding standard deviations have been demonstrated in Table (3-4).

Table 3-4: Experimental Results for Different Approaches on ORL Database

No.	k	NN	PCA	LPP	NPE	LPPl_1	SPP	JSRLPP
8	7	76.17 ± 2.46	74.86 ± 2.67	63.24 ± 2.69	67.60 ± 3.44	77.48 ± 2.74	77.57 ± 2.53	$\mathbf{79.26}$ $\mathbf{\pm 2.43}$
9	7	85.93 ± 2.41	84.67 ± 2.29	71.83 ± 3.00	70.95 ± 3.73	87.77 ± 2.60	87.38 ± 2.37	$\mathbf{88.80}$ $\mathbf{\pm 2.31}$
10	7	88.98 ± 1.90	87.77 ± 1.89	77.69 ± 2.66	73.73 ± 3.04	89.87 ± 2.31	89.44 ± 2.81	$\mathbf{90.23}$ $\mathbf{\pm 2.29}$
11	7	91.44 ± 1.97	90.69 ± 1.92	81.33 ± 4.16	76.78 ± 4.03	92.78 ± 2.30	92.78 ± 1.99	$\mathbf{92.89}$ $\mathbf{\pm 1.83}$

From the above Table (3-4), it is apparent that JSRLPP has the highest classification accuracy rate. The result indicates the significance of data structure graph learning. In addition, these experiments demonstrate that JSRLPP is robust to various expressions and changing rotations.

- Experiment on COIL20 Face Database

COIL-20 (Columbia Object Image Library) is an article database that includes 20 objects images of 360 rotations with intervals of 5 degrees. There are 72 images per object, and Figure (3-9) illustrates some samples from COIL-20 database. All photographs have been cropped into 32 × 32 pixels.

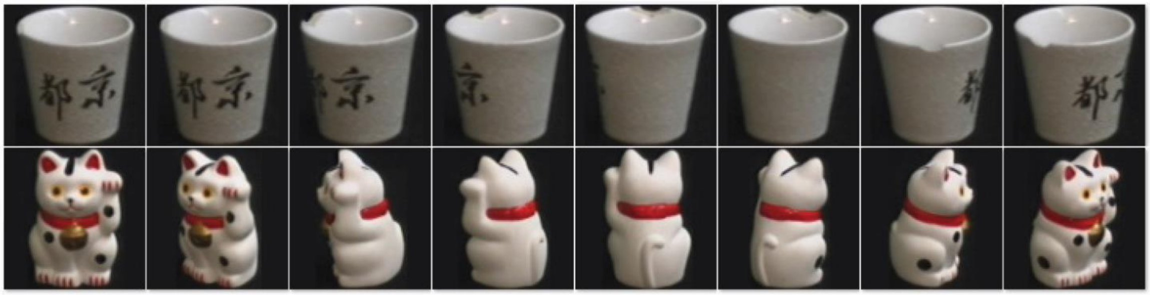


Figure 3-9: Samples from COIL20 Database

We have conducted the experiments for four times per approach, and in each time, we randomly picked 5, 10, 15, and 20 samples from each class to build the training data respectively. The experimental results and corresponding standard deviations have been listed in Table (3-5):

Table 3-5: Experimental Results for Different Approaches on COIL20 Database

No.	k	NN	PCA	LPP	NPE	LPPl_1	SPP	JSRLPP
5	5	82.33 ± 1.55	82.33 ± 1.55	73.11 ± 1.82	71.90 ± 2.09	81.39 ± 1.89	81.91 ± 2.25	**83.29** **± 1.74**
10	7	89.55 ± 1.16	89.55 ± 1.16	80.99 ± 1.13	79.54 ± 1.66	88.92 ± 1.10	89.07 ± 1.14	**91.24** **± 1.27**
15	7	92.99 ± 0.83	92.99 ± 0.83	86.19 ± 1.21	83.99 ± 1.28	92.58 ± 0.86	92.51 ± 0.85	**94.75** **± 0.77**
20	7	95.08 ± 0.86	95.06 ± 0.85	88.27 ± 1.06	85.02 ± 1.35	94.57 ± 0.84	94.33 ± 0.83	**96.60** **± 0.69**

The above Table (3-5) demonstrates the comparison results about various methods on the database COIL-20. Compared to other databases, COIL-20 has achieved better overall performance than others. Possible reasons might be because the database is more straightforward than the other three databases, and the data structure is more likely a kind of linear data structure. However, JSRLPP still achieves the highest accuracy rates on this database.

From the experiments above, we can conclude that JSRLPP performs superior to other similar approaches with different databases. The impressive performance indicates the effectiveness and robustness of JSRLPP.

3.7 Conclusion

This chapter proposed a novel unsupervised feature extraction approach that jointed SR and LPP into a unified framework, JSRLPP. In the framework, the data structure learning and feature extraction are processed concurrently. JSRLPP adaptively captures the primary data intrinsic structure and obtains the informative features. Experiments on various public databases demonstrate that the approach outperforms other state-of-art approaches. In the future, it is expected that the framework to be extended on other dimensionality reduction methods for better interpretability.

Chapter 4 Joint Sparse Representation and Data Residual Relationship

Most feature selection approaches are graph-based approaches that construct a graph to reveal the dataset structure during the process of data analysis. However, it is inevitable that sorts of original data relationships loss will be generated in the graph construction. The Data Residual Relationship (DRR) is an essential consideration, which will be presented in this chapter. For the purpose to properly maintain the relationships between data sample points, we propose a novel unified learning approach, called Unsupervised feature selection with Adaptive Residual Preserving (UFSARP). UFSARP integrates the data reconstruction, local residual relationships preserving and feature selection into a single procedure that allows all tasks to be performed simultaneously.

There are three significant improvements regarding UFSARP: (1) The similarity matrix in the UFSARP ensures similar samples holding similar reconstruction coefficient, and protects the data reconstruction residual relationships. (2) The data reconstruction residual encourages UFSARP to preserve the residual relationships between sample points, as so, similar samples will have similar residuals, and it will further improve the *data structure reconstruction. (3) The similarity matrix and reconstruction coefficient are going to be mutually promoted by the other to obtain better classification result.*

We have conducted comprehensive experiments and confirm that UFSARP is superior over other similar approaches.

4.1 Introduction

The dimensionality reduction approaches can be generally classified into two categories: feature selection (Cai, Zhang, & He, 2010) (Fang, et al., 2014) (He, Cai, & Niyogi, 2005) (He, Ji, Zhang, & Bao, 2011) (Nie, Huang, Cai, & Ding, 2010) (Zhao, Wang, Liu, & Ye, 2013) and feature extraction (Ghassabeh, Rudzicz, & Moghaddam, 2015) (Zheng, Lin, & Wang, 2014) (Krzanowski, 1987). In feature selection, the initial data representation will not be modified, and it tries to obtain a subset of features which are capable to represent the entire original dataset. While, in feature extraction, a set of newly formed features are using to represent the original dataset.

In addition, based on the availability of label information, dimensionality reduction studies can be grouped into supervised learning (Huang, 2015), semi-supervised learning (Kong & Yu, 2010) (Ren, Qiu, Fan, Cheng, & Yu, 2008), and unsupervised learning (Hou, Nie, Li, Yi, & Wu, 2017) (Hou, Nie, Li, Yi, & Wu, 2011) (Mitra, Murthy, & Pal, 2002) (Wang, Tang, & Liu, 2015) (Yang, Shen, Ma, Huang, & Zhou, 2011). Most dataset available in real-world applications are unlabeled, and due to the absence of label information, unsupervised learning is comparable more challenged than supervised and semi-supervised learning. This chapter is about to introduce a novel learning model regarding feature selection in unsupervised learning.

In unsupervised learning, feature selection approaches can be mainly assorted into filters, wrappers, and embeddings.

Filter approaches obtain the optimal subset of features based on a pre-determined ratio score (Zeng & Cheung, 2011) (Liu, Wu, & Zhang, 2014). There are multiple wildly used filter methods, including Laplacian Score (He, Cai, & Niyogi, 2005), Spectral-Based Feature Selection (Zhao & Wang, 2007), and Filter-Based Multivariate Method (Tabakhi, Moradi, & Akhlaghian, 2014), etc. The computation and underlying theory are straightforward. But, these approaches are not tailored to a specific problem, thus, they solely provide a universal selection decision and fail to choose the most appropriate features for a particular decision making (Ghassabeh, Rudzicz, & Moghaddam, 2015).

In wrapper approaches, a pre-defined learning framework is required. Then, the methods can evaluate the performs by ‘wrapping’ the feature selection (Maldonado & Weber, 2009). There are numbers of researches having been proposed in this domain, including the achievements of (Ma, et al., 2017) (Guyon, Weston, Barnhill, & Vapnik, 2002) (Maldonado & Weber, 2009), etc. Compared with filter approaches, wrapper approaches are able to obtain better performant results, but, with much more expensive calculation time. Therefore, it is impractical to implement wrapper approaches in such large-scale datasets.

There are numerous of embedding-based approaches having been proposed in recent decades. Since from 2003, (Weston, Elisseeff, Scholkopf, & Tipping, 2003) introduced the l_0-norm regularization to achieve sparse resolution for feature selection and classifications. Then, in the year 2009, (Liu, Ji, & Ye, 2009) proposed an $l_{2,1}$-norm framework that reached a similar goal. (Zhu, Wu, Ding, & Zhang, 2013) proposed an unsupervised feature selection model by embedding a graph into the framework of joint sparse coding for retaining the local manifold structure of the dataset. (Wang, Chen, Hong, & Zeng, 2018) proposed an unsupervised feature selection framework that employed a similarity matrix and an l_2-norm. All these approaches have successfully achieved impressive performances. However, these approaches separated the data structure construction and feature selection into two independent steps. Namely, if the construction of the data structure fails to represent the original dataset, feature selection based on the construction will be unable to achieve considerable performance. To end this problem, many researchers dedicated to working on frameworks that allow processing the two procedures simultaneously. Most of the frameworks are employing a matrix for both reconstruction and local similarity. In 2010, (Qiao, Chen, & Tan, 2010) applied l_1 -norm to preserve the local neighborhood relationships as well as sparse the representation during the dimensionality reduction. (Du & Shen, 2015) proposed an adaptive learning framework to learn the data structure and select informative features. (Lu, et al., 2016) proposed a framework that used Low-Rank Regularization to preserve the original data structure. (Fang, et al., 2017) introduced a framework that could obtain the feature representation and intrinsic data similarity structure simultaneously by using an orthogonal self-guided approach, etc. During the data structure construction, it is expected that similar samples have similar properties. Although these approaches above have unified the data structure construction and feature selection into a single step, they disregarded the residual relationships between sample points.

This chapter is going to represent a novel unsupervised feature selection approach, UFSARP, that will not only process the subspace learning and feature selection simultaneously but also maintains the local residual relationships during data structure reconstruction. UFSARP is built based on the idea that similar samples hold similarity reconstructed residuals. It introduces a new term, adaptive local residuals, to learn the data local structure. Therefore, with a more precise the data structure and a refined reconstruction structure, UFSARP is capable to obtain a better feature selection result.

The contributions of UFSARP can be listed as followings:

1. UFSARP has unified the feature selection, data reconstruction, and local residual preserving into one framework, which allows all tasks to be completed concurrently.
2. Since UFSARP learns the reconstruction residual relationships adaptively, the framework assures that similar samples have similar reconstruction coefficients and similar reconstruction residuals.
3. UFSARP maintains both local manifold relationships and data global structure, as so, the framework carries a better data reconstruction structure. Experimental outcomes have proved that UFSARP achieves favorable classification performances.
4. The newly proposed term can be implemented in other frameworks or to other research fields to preserve the local manifold relationships.

The remainder of this chapter is designed as followings:

An introduction of recent relevant researches will be provided in section 4.2, and a detail description of UFSARP will be given in section 4.3. Section 4.4 will demonstrate the optimization processes, and Section 4.5 will discuss the computational complexity and convergence. Then, Section 4.6 will list the comprehensive experiments. Lastly, there is a conclusion in section 4.7.

4.2 Related Works

Given that Sparse Representation (SR) has been introduced in Section 2.5, this section will present relevant works in graph embedding and Unsupervised Feature Selection with Adaptive Structure Learning (FSASL) that was proposed by (Du & Shen, 2015).

4.2.1 Graph Embedding

There is numerous literature have proved that graph embedding is working effectively in preserving the manifold structure of datasets (Yan, et al., 2007), and many frameworks have been successfully proposed in this domain, including, ISOMAP (Tenenbaum, Silva, & Langford, 2000), LDA (Martlnez & Kak, 2001), LLE (Roweis & Saul, 2000), LE (Belkin & Niyogi, 2003), etc.

In these conventional graph embedding approaches, the data structure will be constructed by employing different measurement methods. For instance, in the determination of neighborhood relationships, k-nearest neighbors and ε-balls approaches are generally selecting to construct the neighborhood relations. In addition, the weights between neighbors can be calculated by employing the Heat Kernel, inverse Euclidean distance, or Local Linear Reconstruction Coefficient. Since these measurement methods and coefficients are predefined, the data structure graph will be created and fixed at the very beginning, then, the graph construction and feature selection are independently processed.

4.2.2 Unsupervised Feature Selection Framework (FSASL)

In 2015, (Du & Shen, 2015) proposed a unified framework that can process the data reconstruction and feature selection simultaneously under a unified unsupervised learning model.

The objective function of FSASL is:

$$\min_{\boldsymbol{W},\boldsymbol{S},\boldsymbol{P}}(\|\boldsymbol{W}^T\boldsymbol{X}-\boldsymbol{W}^T\boldsymbol{X}\boldsymbol{S}\|_F^2+\alpha\|\boldsymbol{S}\|_1)+\beta\sum_{i,j}^{n}\left(\boldsymbol{W}^T\boldsymbol{x}_i-\boldsymbol{W}^T\boldsymbol{x}_j\right)^2\boldsymbol{P}_{ij}+\mu\boldsymbol{P}_{ij}^2+\gamma\|\boldsymbol{W}\|_{2,1}$$

$$s.t.\ \ s_{ii}=0,\boldsymbol{P}\boldsymbol{1}_n=\boldsymbol{1}_n,\boldsymbol{P}\geq\boldsymbol{0},\boldsymbol{W}^T\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{W}=\boldsymbol{I}$$

(4-1)

In the above Problem (4-1), $\boldsymbol{1}_n\in R^{n\times 1}$ is a vector that the sum of all entries equal to 1.

In FSASL, the first term is a SR, which helps to determine the global structure of the dataset, and $\alpha\|\boldsymbol{S}\|_1$ is the regularization term that intends to balance the sparsity and the reconstruction error. The second term of FSASL has employed a variable $\boldsymbol{P}$, which helps to determine the local manifold structure of the dataset. $\boldsymbol{P}$ denotes to the probability of neighborhood. $\|\boldsymbol{W}\|_{2,1}$ in the function encourages the rows of $\boldsymbol{W}$ to be zero, and with the sparsity of $\boldsymbol{W}$, noises and irrelevant features from the original dataset can be eliminated. β and γ are the regularization parameters.

FSASL has recognized both data global structure and local manifold structure of the original dataset and has integrated the separated processes of data structure construction and feature selection into a unified procedure. Compared with other conventional graph-embedding approaches, FSASL avoids the problem of feature selection based on a pre-built similarity matrix, and as a result, FSASL obtains more appropriate features. However, FSASL defined the local manifold structure by directly employing a neighborhood probability variable. This employed variable can neither guarantee that similar samples to have similar reconstruction residuals nor guarantee similar samples to have similar reconstruction coefficient. Therefore, inspired by FSASL, a framework with DRR is a foreseeable more powerful approach, which can better define the neighborhood relationships between sample points.

4.3 Algorithm

It is essential for graph-based learning approaches to build an informative data structure. UFSARP constructs the data structure by employing SR and DRR. In this section, there is a comprehensive representation of UFSARP.

4.3.1 Similarity Matrix Learning

Similarity matrix can help to maintain the data local manifold structure. Instead of employing the Laplacian graph with a predefined neighborhood relationship, we would prefer to utilize the similarity matrix to unify the graph construction and feature selection into one framework.

The similarity matrix $\boldsymbol{P}$ is used to solve the problem of FSASL as:

$$\min_{\boldsymbol{P}} \sum_{i,j}^{n} \left(\boldsymbol{W}^T\boldsymbol{x}_i - \boldsymbol{W}^T\boldsymbol{x}_j\right)^2 \boldsymbol{P}_{ij} + \mu \boldsymbol{P}_{ij}^2$$

$$\text{s.t. } \boldsymbol{P}\boldsymbol{1}_n = \boldsymbol{1}_n, \boldsymbol{P} \geq \boldsymbol{0}$$

(4-2)

where $\boldsymbol{P}$ denotes the neighborhood probability that reflects the local manifold data structure. From the above function, it is clear that a great distance between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ will increase the value of $\left(\boldsymbol{W}^T\boldsymbol{x}_i - \boldsymbol{W}^T\boldsymbol{x}_j\right)^2$. Holding the outcome of Problem (4-2) to be minimized as a whole, the rate of $\boldsymbol{P}$ will be small. $\mu\boldsymbol{P}_{ij}^2$ in (4-2) is the regularization term that avoids the trivial solution and can be regarded as a prior uniform distribution.

4.3.2 Residual Preserving Learning

Section 4.2.1 has introduced that SR works favorably in obtaining the global representation structure of the dataset. Besides, SR can also be used to eliminate noise and irrelevant features as well. Based on the robustness and effective attributions of SR, our novel framework is going to be built based on it. In SR, the target sample $\boldsymbol{x}_i$ is expressed by a linear combination of all other samples:

$$\min_{\boldsymbol{S}} \sum_{i=1}^{n} \|\boldsymbol{x}_i - \boldsymbol{X}\boldsymbol{s}_i\|^2 + \alpha\|\boldsymbol{S}\|_1$$

$$s.t.\ s_{ii} = 0$$

(4-3)

In Problem (4-3), it is apparent that the target sample can be sparsely represented by adding the regularization term $\alpha\|\boldsymbol{S}\|_1$. However, SR only reflects the data global structure but disregards the local structure and residual relationships between samples. In the UFSARP, we are going to employ a matrix $\boldsymbol{P}$ to preserve DRR. Then, the Problem (4-3) will be reformed as:

$$\min_{\boldsymbol{S},\boldsymbol{P}} \sum_{i,j}^{n} \left\|\boldsymbol{x}_i - \boldsymbol{X}\boldsymbol{s}_j\right\|^2 \boldsymbol{P}_{ij} + \alpha\|\boldsymbol{S}\|_1$$

$$s.t.\ s_{ii} = 0$$

(4-4)

$\boldsymbol{X}\boldsymbol{s}_j$ is the reconstructed point of the original sample $\boldsymbol{x}_j$. Function (4-4) describes that if $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are alike, then, $\boldsymbol{x}_i$ and the reconstructed $\boldsymbol{x}_j$ should still be alike.

Definitions and Proofs:

Lemma 1: Given two samples, $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, are similar, the result of $\left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|^2$ will be small by employing the Euclidean distance.

Proof: The distance between sample points can be interpreted as dissimilarity. The shorter the distance, the more similar the sample points are. When sample $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are alike, then the Euclidean distance between them will be small.

Lemma 2: Given two samples, $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, are similar, the reconstructed points, $\sum_{i=1}^{n} \boldsymbol{X}\boldsymbol{s}_i$ and $\sum_{j=1}^{n} \boldsymbol{X}\boldsymbol{s}_j$, will be similar in the Euclidean distance.

Proof: According to the trigonometric inequality:

$$\left\|\sum_{i=1}^{n} \boldsymbol{X}\boldsymbol{s}_i - \sum_{j=1}^{n} \boldsymbol{X}\boldsymbol{s}_j\right\|^2 = \left\|\sum_{i=1}^{n} \boldsymbol{X}\boldsymbol{s}_i - \sum_{j=1}^{n} \boldsymbol{X}\boldsymbol{s}_j + \boldsymbol{x}_i - \boldsymbol{x}_i + \boldsymbol{x}_j - \boldsymbol{x}_j\right\|^2$$

$$\leq \left\|\boldsymbol{x}_i - \sum_{i=1}^{n} \boldsymbol{X}\boldsymbol{s}_i\right\|^2 + \left\|\boldsymbol{x}_j - \sum_{j=1}^{n} \boldsymbol{X}\boldsymbol{s}_j\right\|^2 + \left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|^2$$

Since $\sum_{i=1}^{n} \boldsymbol{X}\boldsymbol{s}_i$ and $\sum_{j=1}^{n} \boldsymbol{X}\boldsymbol{s}_j$ are the reconstructed points of $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, $\|\boldsymbol{x}_i - \sum_{i=1}^{n} \boldsymbol{X}\boldsymbol{s}_i\|^2$ and $\left\|\boldsymbol{x}_j - \sum_{j=1}^{n} \boldsymbol{X}\boldsymbol{s}_j\right\|^2$ will be small. In addition, based on Lemma 1, $\left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|^2$ is small, then $\sum_{i=1}^{n} \boldsymbol{X}\boldsymbol{s}_i$ and $\sum_{j=1}^{n} \boldsymbol{X}\boldsymbol{s}_j$ will be similar as well.

Theorem 1: Given two samples, $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, are similar, and the reconstructed points, $\sum_{i=1}^{n} \boldsymbol{X}\boldsymbol{s}_i$ and $\sum_{j=1}^{n} \boldsymbol{X}\boldsymbol{s}_j$, are similar to the original points, $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, respectively, then the reconstruction coefficient matrix $\boldsymbol{S}$ is going to be not only enforce similar samples to be similar, but also can preserve residuals relationships between similar samples.

Proof: Refers to Lemma 1 and Lemma 2, given $\sum_{i=1}^{n} \boldsymbol{X}\boldsymbol{s}_i$ and $\sum_{j=1}^{n} \boldsymbol{X}\boldsymbol{s}_j$ are similar, $\left\|\sum_{i=1}^{n} \boldsymbol{X}\boldsymbol{s}_i - \sum_{j=1}^{n} \boldsymbol{X}\boldsymbol{s}_j\right\|^2$ is small. Then, the reconstruction coefficient matrix $\boldsymbol{S}$ is going to ensure that similar samples to be similar after reconstruction, in addition, it preserves residuals relationships between similar samples.

4.3.3 UFSARP

The purpose of USFARP is to choose informative features. According to the prementioned inferences above, the framework should be built as:

$$\min_{\boldsymbol{S},\boldsymbol{P},\boldsymbol{W}} \sum_{i,j}^{n} \left\| \boldsymbol{W}^T \boldsymbol{x}_i - \boldsymbol{W}^T \boldsymbol{X} \boldsymbol{s}_j \right\|^2 \boldsymbol{P}_{ij} + \alpha \|\boldsymbol{S}\|_1$$

$$s.t.\ \ s_{ii} = 0$$

(4-6)

Based on the Function (4-6), a regularization term of $\boldsymbol{P}_{ij}$ should be added, and the problem will be further converted into:

$$\min_{\boldsymbol{S},\boldsymbol{P},\boldsymbol{W}} \sum_{i,j}^{n} \left\| \boldsymbol{W}^T \boldsymbol{x}_i - \boldsymbol{W}^T \boldsymbol{X} \boldsymbol{s}_j \right\|^2 \boldsymbol{P}_{ij} + \alpha \|\boldsymbol{S}\|_1 + \beta \left(\sum_{i,j}^{n} \left\| \boldsymbol{W}^T \boldsymbol{x}_i - \boldsymbol{W}^T \boldsymbol{x}_j \right\|^2 \boldsymbol{P}_{ij} + \mu {\boldsymbol{P}_{ij}}^2 \right)$$

$$s.t.\ \ \boldsymbol{P}\boldsymbol{1}_n = \boldsymbol{1}_n, \boldsymbol{P} \geq \boldsymbol{0}, s_{ii} = 0$$

(4-7)

In Problem (4-7), the variable $\boldsymbol{P}$ preserves the local manifold data structure, and the term $\left\| \boldsymbol{W}^T \boldsymbol{x}_i - \boldsymbol{W}^T \boldsymbol{x}_j \right\|^2$ represents the reconstruction residual relationships between sample points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. $\left\| \boldsymbol{W}^T \boldsymbol{x}_i - \boldsymbol{W}^T \boldsymbol{x}_j \right\|^2$ is known as a regularization term that encourages similar sample has similar reconstruction residuals during data reconstruction.

The above Problem (4-9) is a semi-finished framework. In order to maintain the attributes of DRR and obtain informative features, an $l_{2,1}$-norm constraint should be attached to the reconstruction matrix $\boldsymbol{W}$ as $l_{2,1}$-norm has the desirable property of row sparsity. Then, the UFSARP framework is finally becoming:

$$\min_{\boldsymbol{S},\boldsymbol{P},\boldsymbol{W}} \sum_{i,j}^{n} \left\| \boldsymbol{W}^T \boldsymbol{x}_i - \boldsymbol{W}^T \boldsymbol{X} \boldsymbol{s}_j \right\|^2 \boldsymbol{P}_{ij} + \alpha \|\boldsymbol{S}\|_1 + \beta \left(\sum_{i,j}^{n} \left\| \boldsymbol{W}^T \boldsymbol{x}_i - \boldsymbol{W}^T \boldsymbol{x}_j \right\|^2 \boldsymbol{P}_{ij} + \mu {\boldsymbol{P}_{ij}}^2 \right) + \gamma \|\boldsymbol{W}\|_{2,1}$$

$$s.t. \quad \boldsymbol{P}\boldsymbol{1}_n = \boldsymbol{1}_n, \boldsymbol{P} \geq \boldsymbol{0}, s_{ii} = 0, \boldsymbol{W}^T \boldsymbol{X}\boldsymbol{X}^T \boldsymbol{W} = \boldsymbol{I}$$

(4-8)

where α, β, and γ are regularization parameters that are operated to balance the influences of corresponding terms.

USFARP has integrated SR with the DRR for obtaining the true structure of the dataset. The first term in the framework, $\left\| \boldsymbol{W}^T \boldsymbol{x}_i - \boldsymbol{W}^T \boldsymbol{X} \boldsymbol{s}_j \right\|^2 \boldsymbol{P}_{ij}$, learns the local manifold structure of the dataset. More detailed, suppose there are two similar sample points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, the value of $\left\| \boldsymbol{W}^T \boldsymbol{x}_i - \boldsymbol{W}^T \boldsymbol{X} \boldsymbol{s}_j \right\|^2$ will be small and which will automatically impact the value of $\boldsymbol{P}_{ij}$ to be relatively large. Since $\boldsymbol{P}_{ij}$ can be adaptively learned in this framework, it is not necessary to employ other neighborhood measurements to compute $\boldsymbol{P}_{ij}$. In addition, with the variable $\boldsymbol{P}_{ij}$, USFARP ensures that similar samples have similar reconstruction residuals which assist to precisely preserve the relationships between data points.

4.4 Optimization

In UFARP, all variables $\boldsymbol{W}$, $\boldsymbol{P}$ and $\boldsymbol{S}$ are unknown and it is challenged to find the optimal resolution of the problem directly, particularly during the calculation of the sub-derivative regarding $\boldsymbol{S}$. To simplify the problem, I am going to transferred the problem into a minimization problem of an Augmented Lagrange Multiplier (ALM):

$$\begin{aligned}
&\Gamma(\boldsymbol{W},\boldsymbol{S},\boldsymbol{P},\boldsymbol{Q},\boldsymbol{T},\boldsymbol{Y}_1,\boldsymbol{Y}_2,\boldsymbol{Y}_3,\mu_1)\\
&\qquad = \min_{\boldsymbol{S},\boldsymbol{P},\boldsymbol{W}} \sum_{i,j}^{n} \left\|\boldsymbol{W}^T\boldsymbol{x}_i - \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{s}_j\right\|^2 \boldsymbol{P}_{ij} + \alpha\|\boldsymbol{ST}\|_1\\
&\qquad + \beta\left(\sum_{i,j}^{n} \left\|\boldsymbol{W}^T\boldsymbol{x}_i - \boldsymbol{W}^T\boldsymbol{x}_j\right\|^2 \boldsymbol{P}_{ij} + \mu{\boldsymbol{P}_{ij}}^2\right) + \gamma\|\boldsymbol{W}\|_{2,1} + \langle \boldsymbol{Y}_1, \boldsymbol{P}\boldsymbol{1}_n - \boldsymbol{1}_n\rangle\\
&\qquad + \langle \boldsymbol{Y}_2, \boldsymbol{P}-\boldsymbol{Q}\rangle + \langle \boldsymbol{Y}_3, \boldsymbol{S}-\boldsymbol{T}\rangle + \frac{\mu_1}{2}\left(\|\boldsymbol{P}\boldsymbol{1}_n - \boldsymbol{1}_n\|_F^2 + \|\boldsymbol{P}-\boldsymbol{Q}\|_F^2\right)\\
&\qquad + \frac{\mu_2}{2}\|\boldsymbol{S}-\boldsymbol{T}\|_F^2\\
&\qquad\qquad s.t. \quad s_{ii} = 0, \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{W} = \boldsymbol{I}
\end{aligned} \tag{4-9}$$

where $\langle \boldsymbol{A},\boldsymbol{B}\rangle = Tr(\boldsymbol{A}^T\boldsymbol{B})$. $\boldsymbol{Y}_1$, $\boldsymbol{Y}_2$, and $\boldsymbol{Y}_3$ are Lagrange multiplier. $\mu_1 > 0$ denotes to the penalty parameter. To solve this problem, I am going to employ the Alternating Direction Method of Multipliers (ADMM) (Boyd, Parikh, Chu, Peleato, & Eckstein, 2010). In ADMM, each variable is going to be updated while keeping all other variables unchanged in every single iteration. The solving steps are shown as followings:

4.4.1 Updating *W*

In the process of updating $\boldsymbol{W}$, the relevant function is:

$$\Gamma(\boldsymbol{W}) = \min_{\boldsymbol{W}} \sum_{i,j}^{n} \left\|\boldsymbol{W}^T\boldsymbol{x}_i - \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{s}_j\right\|^2 \boldsymbol{P}_{ij} + \beta\left(\sum_{i,j}^{n} \left\|\boldsymbol{W}^T\boldsymbol{x}_i - \boldsymbol{W}^T\boldsymbol{x}_j\right\|^2 \boldsymbol{P}_{ij} + \mu \boldsymbol{P}_{ij}{}^2\right) + \gamma\|\boldsymbol{W}\|_{2,1}$$

$$s.t. \quad \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{W} = \boldsymbol{I}$$

(4-10)

Let $\boldsymbol{L_{SP}} = \left(\boldsymbol{D_P} + \boldsymbol{S}\boldsymbol{D}_{\boldsymbol{P}^T} - 2\boldsymbol{P}\boldsymbol{S}^T + \beta(\boldsymbol{D_P} + \boldsymbol{D}_{\boldsymbol{P}^T} - 2\boldsymbol{P})\right)$, $\boldsymbol{D_P}$ and $\boldsymbol{D}_{\boldsymbol{P}^T}$ are diagonal matrices that $\boldsymbol{D}_{P_{ii}} = \sum_j P_{ij}$ and $\boldsymbol{D}_{\boldsymbol{P}^T} = \sum_i P_{ij}$. Then, the Problem (4-12) can be rewritten into:

$$\Gamma(\boldsymbol{W}) = \min_{\boldsymbol{W}} Tr(\boldsymbol{W}^T\boldsymbol{X}\boldsymbol{L_{SP}}\boldsymbol{X}^T\boldsymbol{W}) + \gamma(\boldsymbol{W}^T\boldsymbol{D_W}\boldsymbol{W})$$

$$s.t. \quad \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{W} = \boldsymbol{I}$$

(4-11)

where $\boldsymbol{D_W} \in R^{d\times d}$ denotes to a diagonal matrix that the i-th element is $\boldsymbol{D}_{W_{i,i}} = \frac{1}{2\|\boldsymbol{W}_i\|_2}$. Then, the problem can be transferred into:

$$\Gamma(\boldsymbol{W}) = \min_{\boldsymbol{W}} Tr\boldsymbol{W}^T(\boldsymbol{X}\boldsymbol{L}_{\boldsymbol{SP}}\boldsymbol{X}^T + \gamma\boldsymbol{D}_{\boldsymbol{W}})^T\boldsymbol{W}$$

$$s.t. \quad \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{W} = \boldsymbol{I}$$

(4-12)

It is clear that the optimal resolutions of $\boldsymbol{W}$ are the eigenvectors corresponding to the $\boldsymbol{c}$ smallest eigenvalues of $(\boldsymbol{X}\boldsymbol{L}_{\boldsymbol{SP}}\boldsymbol{X}^T + \gamma\boldsymbol{D}_{\boldsymbol{W}})^T\boldsymbol{W} = \boldsymbol{\Lambda}\boldsymbol{X}\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{W}$.

4.4.2 Updating *P*

During the process of updating $\boldsymbol{P}$, the corresponding function is:

$$\Gamma(\boldsymbol{P}) = \min_{\boldsymbol{P}} \sum_{i,j}^{n} \left\|\boldsymbol{W}^T\boldsymbol{x}_i - \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{s}_j\right\|^2 \boldsymbol{P}_{ij} + \beta\left(\sum_{i,j}^{n}\left\|\boldsymbol{W}^T\boldsymbol{x}_i - \boldsymbol{W}^T\boldsymbol{x}_j\right\|^2\boldsymbol{P}_{ij} + \mu\boldsymbol{P}_{ij}{}^2\right)$$
$$+\frac{\mu_1}{2}\left(\left\|\boldsymbol{P}\boldsymbol{1}_n - \boldsymbol{1}_n + \frac{\boldsymbol{Y}_1}{\mu_1}\right\|_F^2 + \left\|\boldsymbol{P} - \boldsymbol{Q} + \frac{\boldsymbol{Y}_2}{\mu_1}\right\|_F^2\right)$$

(4-13)

Take the partial derivative of $\Gamma(\boldsymbol{P})$:

$$\frac{(2\mu + \mu_1)}{\mu_1}\boldsymbol{P} + \boldsymbol{P}\boldsymbol{1}_n\boldsymbol{1}_n^T - \frac{\boldsymbol{E}}{\mu_1} = \boldsymbol{0}$$

(4-14)

Then, the value of $\boldsymbol{P}$ can be solved by:

$$\boldsymbol{P} = \frac{\boldsymbol{E}}{2\mu + \mu_1}\left(\boldsymbol{I} + \frac{\mu_1}{2\mu + \mu_1}\mathbf{1}_n\mathbf{1}_n^T\right)^{-1}$$

(4-15)

where $\boldsymbol{E} = \mu_1\mathbf{1}_n\mathbf{1}_n^T + \mu_1\boldsymbol{Q} - \boldsymbol{A} - \beta\boldsymbol{B} - \boldsymbol{Y}_1\mathbf{1}_n^T - \boldsymbol{Y}_2$, $\boldsymbol{A}_{ij} = \left\|\boldsymbol{W}^T\boldsymbol{x}_i - \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{s}_j\right\|^2$ and $\boldsymbol{B}_{ij} = \left\|\boldsymbol{W}^T\boldsymbol{x}_i - \boldsymbol{W}^T\boldsymbol{x}_j\right\|^2$.

4.3.4.3 Updating *S*

In the process of updating $\boldsymbol{S}$, the relevant function is:

$$\Gamma(\boldsymbol{S}) = \min_{\boldsymbol{S}} \sum_j^n \left\|\boldsymbol{W}^T\boldsymbol{x}_i - \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{s}_j\right\|^2\boldsymbol{P}_{ij} + \alpha\|\boldsymbol{T}\|_1 + \frac{\mu_1}{2}\left\|\boldsymbol{S} - \boldsymbol{T} + \frac{\boldsymbol{Y}_3}{\mu_1}\right\|_F^2$$

$$s.t. \quad s_{ii} = 0$$

(4-16)

Let the partial derivative of $\frac{\partial\Gamma}{\partial\boldsymbol{S}}$ to be zero, then $\boldsymbol{S}$ can be solved by a Sylvester equation:

$$\frac{\boldsymbol{C}}{\mu_1}\boldsymbol{S} + S\boldsymbol{D}_{\boldsymbol{P}^T}^{-1} + \frac{(\boldsymbol{C}\boldsymbol{P} - \mu_1\boldsymbol{T} + \boldsymbol{Y}_3)\boldsymbol{D}_{\boldsymbol{P}^T}^{-1}}{\mu_1} = \mathbf{0}$$

(4-17)

where $\boldsymbol{C} = 2\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{W}^T\boldsymbol{X}$.

4.3.4.4 Updating $\boldsymbol{Q}, \boldsymbol{T}$ and Lagrange Multipliers

$\boldsymbol{Q}$ can be solved by:

$$\boldsymbol{Q}^* = \arg\min_{\boldsymbol{Q}} \frac{\mu_1}{2} \left\| \boldsymbol{P} - \boldsymbol{Q} + \frac{\boldsymbol{Y}_2}{\mu_1} \right\|_F^2$$

$$s.t. \quad \boldsymbol{Q} > \boldsymbol{0}$$

(4-18)

The solution will be set as $\boldsymbol{Q}^* = \max\left(\boldsymbol{P} + \frac{\boldsymbol{Y}_2}{\mu_1}, \boldsymbol{0}\right)$.

$\boldsymbol{T}$ can be solved by:

$$\boldsymbol{T}^* = \arg\min_{\boldsymbol{T}} \alpha \|\boldsymbol{T}\|_1 + \frac{\mu_1}{2} \left\| \boldsymbol{S} - \boldsymbol{T} + \frac{\boldsymbol{Y}_3}{\mu_1} \right\|_F^2$$

(4-19)

Then, a shrinkage operator has been employed:

$$\boldsymbol{T}^* = shrink\left(\boldsymbol{S} + \frac{\boldsymbol{Y}_3}{\mu_1}, \frac{\alpha}{\mu_1}\right)$$

(4-20)

where $shrink(x, a) = sign(x) \max(|x| - a, 0)$.

The Lagrange Multiplier can be solved by:

$$\begin{cases} \boldsymbol{Y}_1 = \boldsymbol{Y}_1 + \mu_1(\boldsymbol{P}\mathbf{1}_n - \mathbf{1}_n) \\ \boldsymbol{Y}_2 = \boldsymbol{Y}_2 + \mu_1(\boldsymbol{P} - \boldsymbol{Q}) \\ \boldsymbol{Y}_3 = \boldsymbol{Y}_3 + \mu_1(\boldsymbol{S} - \boldsymbol{T}) \\ \mu_1 = \min(\mu_{max}, \rho\mu_1) \end{cases} \tag{4-21}$$

In the above Problem (4-21), $\rho > 0$ is the step size in each iteration and μ_{max} is a pre-defined constant.

To sum up, the optimization process can be shown as **Algorithm 4-1**:

Algorithm 4-1: Solving the USFARP by ADMM

```
Input: $\boldsymbol{X} \in \boldsymbol{R}^{d\times n}$, $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\mu}$, $\boldsymbol{c}$;
Initialization: $\boldsymbol{P} = \boldsymbol{P_{knn}}$, $\boldsymbol{Q} = \boldsymbol{P}$, $\boldsymbol{S} = \frac{\mathbf{1}}{\boldsymbol{n}}\mathbf{1}$, $\boldsymbol{T} = \boldsymbol{S}\ \boldsymbol{\mu_{max}} = \mathbf{10^7}$, $\boldsymbol{Y_1} = \mathbf{0}$, $\boldsymbol{Y_2} = \boldsymbol{Y_3} = \mathbf{0}$, $\boldsymbol{\mu} = \mathbf{0.1}$, $\boldsymbol{\rho} = \mathbf{1.01}$, $\boldsymbol{\varepsilon} = \mathbf{10^{-6}}$;
    While not converged do
    Fix other variables, update the projection matrix $\boldsymbol{W}$ by solving problem (4-14)
    Fix other variables, update the projection matrix $\boldsymbol{P}$ by solving problem (4-17)
    Fix other variables, update the projection matrix $\boldsymbol{S}$ by solving problem (4-19)
    Fix other variables, update the projection matrix $\boldsymbol{Q}$ by solving problem (4-20)
    Fix other variables, update the projection matrix $\boldsymbol{T}$ by solving problem (4-22)
    Update the multipliers and parameters by solving problem (4-23)
    Check the convergence condition by:
        $\|\boldsymbol{P}\mathbf{1}_{\boldsymbol{n}} - \mathbf{1}_{\boldsymbol{n}}\|_{\boldsymbol{F}} < \boldsymbol{\varepsilon}$, $\|\boldsymbol{P} - \boldsymbol{Q}\|_{\boldsymbol{F}} < \boldsymbol{\varepsilon}$ and $\|\boldsymbol{S} - \boldsymbol{T}\|_{\boldsymbol{F}} < \boldsymbol{\varepsilon}$
    end while
Output: $\boldsymbol{W}$
```

4.5 Discussion

4.5.1 Computational Complexity

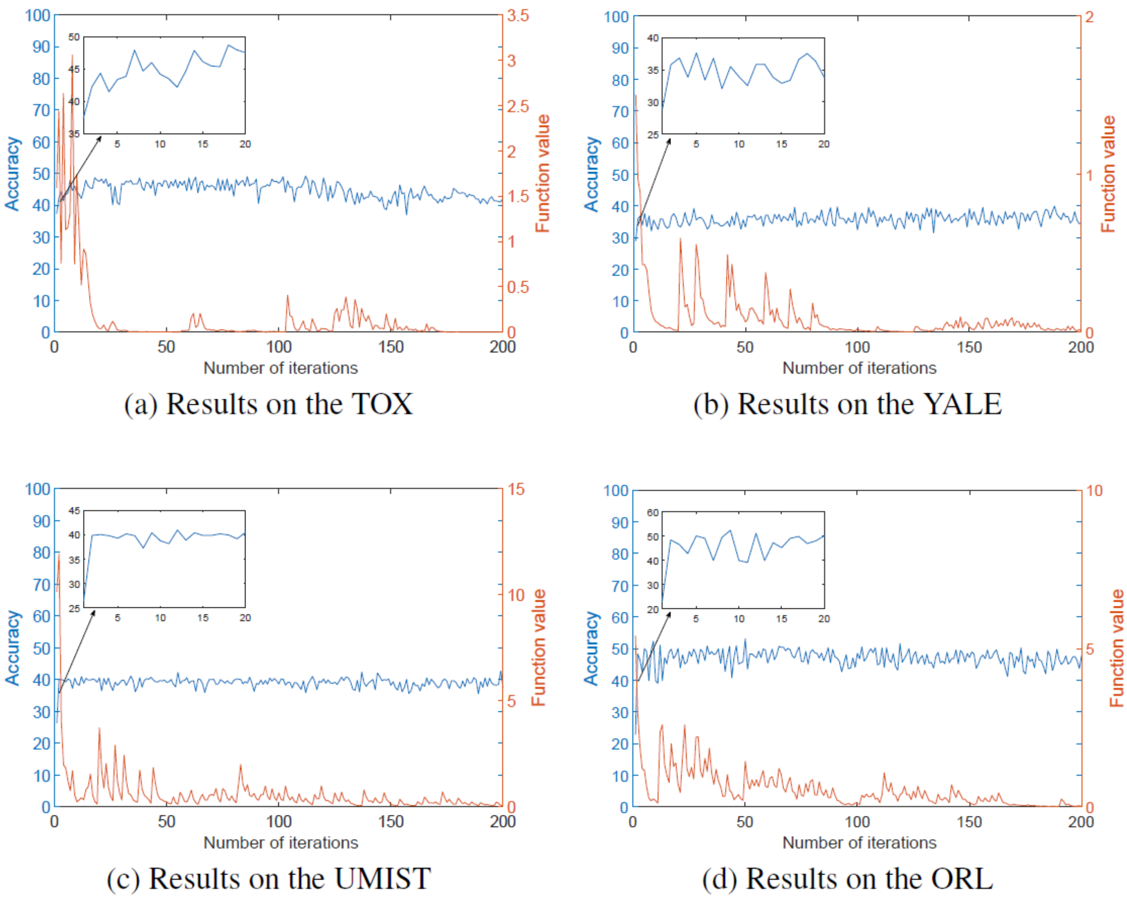
The computational burden of UFSARP is in Problem (4-12), (4-15) and (4-17) as all of them have employed the Eigenvalue Decomposition (EVD) and Sylvester equation problem. Especially in Problem (4-12), which involves a $d \times d$ matrix, the computational complexity is $\boldsymbol{O}(d^3)$. The Problem (4-15) and (4-17) involves $d \times d$ matrices that the computational complexity is $\boldsymbol{O}(2n^3)$. Consequently, the computational complexity of the problem as a whole is $\boldsymbol{O}(\tau(d^3 + 2n^3))$, and τ denotes the number of iterations.

4.5.2 Convergence Analysis

This section will introduce the convergence behavior of UFSARP. The framework has been used to test on four image datasets, including TOX, YALE, UMIST, and ORL. The statue of convergence has been defined as when all variables in the framework are stable. The relative convergence objective function has been determined as:

$$obj = \|\boldsymbol{P} - \boldsymbol{Q}\|_F + \|\boldsymbol{S} - \boldsymbol{T}\|_F < \varepsilon \tag{4-22}$$

Problem (4-22) calculates the sum of changes, and it is going to be operated for 300 iterations to show the numerical value. The result has been provided in Figure (4-1):



(a) Results on the TOX

(b) Results on the YALE

(c) Results on the UMIST

(d) Results on the ORL

Figure 4-1: The Convergence Behavior

From Figure (4-1), it is clear that the value of objective function declines extensively at the beginning, and then it fluctuates within a small range. This unstable result is probably because of the impact of regularization terms in the framework. The main reasons may because: (1) it is not ensuring whether $\boldsymbol{X}$ and $\boldsymbol{X}^T$ are nonsingular, which may direct the fluctuation of the pseudo-inverse of $\boldsymbol{X}\boldsymbol{X}^T$in the Problem (4-12); (2) the Problem (4-17) is using Sylvester, which may be a cause of fluctuation as well.

Although the value of the objective function fluctuates, it is still going to reach the convergence ultimately. In addition, the accuracy of UFSARP also has some waves, which may because the clustering using k-means is mainly depending on the initialization.

4.5.3 Discussion of Parameter μ

μ in the framework is used to balance the tradeoff between the trivial solutions ($\mu = 0$) and the uniform distribution ($\mu = \infty$). Based on the researches of (Du & Shen, 2015) (Nie, Wang, & Huang, 2014), μ can be defined as:

$$\mu = \frac{1}{n}\sum_{i}^{n}\left(\frac{k}{2}d_{i,k'+1}^{\boldsymbol{W}} - \frac{1}{2}d_{ik'}^{\boldsymbol{W}}\right) \tag{4-23}$$

where k is a pre-defined parameter that represents the number of neighbors. Since the result of μ depends on the value of k, it is more intuitive and simpler to tune.

4.5.4 Comparison with FSASL

As prementioned in Section 4.2.3, FSASL is a framework that can simultaneously process the feature selection and data structure recognition. FSASL was proposed by (Du & Shen, 2015) in 2015, and the objective function is demonstrated as Function (4-3). In this section, a discussion of the main differences between the framework FSASL and USFARP is going to be presented.

In FSASL, there is only one variable considering the neighborhood structure, and it has further been used to maintain the local manifold structure of datasets. However, this variable can only capture partial intrinsic structures. Inspired by the idea of FSASL, USFARP framework has been proposed.

The same as FSASL, USFARP has unified the data structure learning and feature selection into a single procedure. However, USFARP introduces a new concept in the framework, the DRR between data samples. USFARP encourages similar samples to have similar reconstruction coefficients and similar reconstruction residuals by attaching DRR into the learning model. With these superior attributes, USFARP maintains the local manifold data structure better.

4.6 Experiments

In this section, there are a series of experiments that have been conducted on ten public datasets from diverse research domains.

4.6.1 Datasets

The datasets that have been involved in the experiments are including the handwritten datasets and spoken digit/letter recognition datasets (MFEA and USPS), face image datasets (UMIST, JAFFE, AR, YALE, and ORL), object dataset (COIL), and biomedical datasets (LUNG and TOX).

Section 2.4 presented comprehensive datasets information, and Table (4-1) briefly summarized the benchmark datasets that USFARP has carried experiments on.

Table 4-1: Summary of the Benchemark Datasets

Datasets.	***Total #. of Samples***	***Total #. of Features***	***Total #. of Classes***	***Selected Features***
MFEA 200	200	240	10	$[5, 10, \dots, 50]$
USPS200	200	256	2	$[5, 10, \dots, 50]$
UMIST	575	644	20	$[5, 10, \dots, 50]$
JAFFE	213	676	10	$[5, 10, \dots, 50]$
AR	840	768	120	$[5, 10, \dots, 50]$
COIL	1440	1024	20	$[5, 10, \dots, 50]$
ORL	400	1024	40	$[5, 10, \dots, 50]$
YALE	165	1024	15	$[5, 10, \dots, 50]$
LUNG	203	3312	5	$[10, 20, \dots, 100]$
TOX	171	5748	4	$[10, 20, \dots, 100]$

4.6.2 Experiment Setup

In order to prove the superior attributions of UFSARP, the framework has been compared with other state-of-art unsupervised feature selection approaches and one base-line (AllFea):

- LapScore (He, Cai, & Niyogi, 2005): this approach is mainly focusing on the ability to preserve locality manifold structure

- MCFS (Cai, Zhang, & He, 2010): this approach has adopted spectral regression and l_1-norm regularization
- LLCFS (Zeng & Cheung, 2011): this approach incorporates data features in a built-in regularization function
- UDFS (Yang, Shen, Ma, Huang, & Zhou, 2011): this approach processes the local data structure and feature correlations simultaneously
- NDFS (Li, Yang, Liu, Zhou, & Lu, 2012): this approach has joined the nonnegative spectral analysis and $l_{2,1}$-norm regularization
- RUFS (Qian & Zhai, 2013): this approach selects the most critical features by processing the robust clustering and feature selection at the same time
- JELSR (Hou, Nie, Li, Yi, & Wu, 2011): this approach has joined the graph embedding method with a SR to perform feature selection
- GLSPFS (Liu, Wang, Zhang, Yin, & Liu, 2014): this approach has combined the global structure and local geometric data structure in one framework
- FSASL (Du & Shen, 2015): this approach using SR and neighborhood variable to determine the global structure and local structure respectively and processes feature selection at the same time
- URAFS (Li, Zhang, Zhang, Liu, & Nie, 2018): this approach has employed an uncorrelated regression function to perform the feature selection and spectral clustering simultaneously

Several parameters have been preset for computational convenience. In the experiments, $k = 5$ is the number of neighbors for most approaches, while for the framework GLSPFS, it is using the Gaussian Kernel to determine the neighborhood relationships. In Gaussian

Kernel, the kernel width is going to be determined using grid-search from the range of $\{2^{-3}, 2^{-2}, \dots, 2^{3}\} \cdot \delta_0$, where δ_0 equals to the mean value of two samples. To fairly compare the feature selection results, all approaches are employing the grid-search strategy to define the parameters. For approaches other than UFSARP, we set the grids to be $\{10^{-5}, 10^{-4}, \dots, 10^{5}\}$, while in the framework UFSARP, the grids have been set as: $\{10^{-3}, 5 \times 10^{-3}, 10^{-2}, \dots, 5, 10\}$ for parameter α and β, $\{10^{-3}, 10^{-2}, \dots, 10^{2}, 10^{3}\}$ for γ. In addition, Accuracy (ACC) and Normalized Mutual Information (NMI) have been employed to evaluate the performance of each approach. The experimental results are the average performance for twenty times repeat clustering with randomly selected training samples.

4.6.3 Clustering

During the experiments, the different number of features have been set to evaluate the performances of each method, as shown in Table (4-1). The performances are measured by ACC and NMI that have been demonstrated in Table (4-2) and (4-3). Each cell outlines the $average \pm standard\ deviation$, and the last column is the average of the performances regarding each approach over six different datasets.

Table 4-2: Perfromances Measured by ACC (%)

Datasets	**UMIST**	**JAFFE**	**AR**	**COIL**	**LUNG**	**TOX**	**Ave.**
AllFea	42.40	71.57	30.26	59.17	72.46	43.65	53.25
LapScore	36.73 ± 1.18	67.62 ± 8.49	25.29 ± 2.89	45.60 ± 6.16	58.97 ± 5.24	40.25 ± 0.65	45.74
MCFS	44.46 ± 3.26	73.56 ± 4.83	29.05 ± 1.19	51.50 ± 5.38	70.42 ± 3.41	43.10 ± 1.86	52.02
LLCFS	47.31 ± 0.83	64.79 ± 4.08	34.22 ± 2.70	50.84 ± 3.76	71.58 ± 5.85	39.28 ± 0.49	51.34
UDFS	48.04 ± 1.92	75.48 ± 1.63	30.87 ± 0.35	48.40 ± 16.89	65.46 ± 3.88	47.14 ± 0.75	52.57
NDFS	52.80 ± 2.26	74.98 ± 2.15	32.34 ± 1.52	52.22 ± 6.33	75.52 ± 1.57	38.28 ± 1.64	54.36
URAFS	45.77 ± 2.89	79.86 ± 8.63	40.67 ± 1.30	56.68 ± 3.84	66.85 ± 7.65	49.80 ± 1.68	56.61
RUFS	50.87 ± 1.95	75.75 ± 2.53	34.84 ± 1.90	59.20 ± 3.28	77.35 ± 2.62	49.17 ± 0.83	57.86
JELSR	53.52 ± 1.54	77.77 ± 1.87	34.19 ± 2.52	59.53 ± 4.01	77.86 ± 3.12	43.96 ± 1.56	57.81
GLSPFS	50.53 ± 0.59	75.46 ± 1.61	34.12 ± 1.60	57.96 ± 2.27	77.83 ± 2.70	47.38 ± 1.93	57.21
FSASL	54.92 ± 1.89	79.29 ± 2.24	36.11 ± 0.75	60.93 ± 2.50	81.93 ± 1.63	50.12 ± 0.67	60.55
UFSARP	53.99 ± 4.14	81.39 ± 9.11	39.32 ± 0.87	61.87 ± 5.91	83.33 ± 2.58	52.57 ± 3.36	62.08

Table 4-3: Performances Measured by NMI (%)

Datasets	UMIST	JAFFE	AR	COIL	LUNG	TOX	Ave.
AllFea	64.15	81.52	65.48	75.58	60.37	15.87	60.50
LapScore	55.57 ± 2.32	77.28 ± 8.89	63.59 ± 2.36	62.21 ± 4.98	50.14 ± 4.13	10.92 ± 0.68	53.29
MCFS	63.46 ± 4.93	79.04 ± 5.88	66.41 ± 0.85	66.19 ± 6.78	55.68 ± 2.31	16.53 ± 2.68	57.89
LLCFS	63.42 ± 1.42	66.97 ± 3.47	69.01 ± 1.45	64.04 ± 4.34	60.12 ± 4.65	9.68 ± 0.75	55.54
UDFS	65.19 ± 2.96	84.25 ± 1.74	67.49 ± 0.27	44.27 ± 12.61	54.88 ± 4.21	22.16 ± 1.36	56.37
NDFS	71.19 ± 2.77	82.53 ± 3.49	67.89 ± 0.89	56.29 ± 6.91	60.57 ± 1.54	9.07 ± 1.87	57.92
URAFS	62.53 ± 2.23	81.37 ± 3.56	70.42 ± 0.59	69.75 ± 2.17	51.97 ± 4.22	26.16 ± 2.22	60.37
RUFS	68.19 ± 2.61	82.00 ± 3.56	69.54 ± 1.10	70.54 ± 4.48	65.47 ± 1.87	25.79 ± 1.60	63.59
JELSR	71.33 ± 2.06	85.23 ± 3.31	69.02 ± 1.32	71.37 ± 4.97	63.54 ± 2.94	17.46 ± 3.36	62.99
GLSPFS	69.16 ± 0.97	83.20 ± 3.17	69.44 ± 0.84	69.89 ± 4.00	63.50 ± 2.99	23.49 ± 2.77	63.11
FSASL	72.39 ± 2.39	86.42 ± 3.34	70.78 ± 0.63	72.93 ± 4.44	66.78 ± 1.72	27.37 ± 1.62	66.11
UFSARP	66.30 ± 2.17	86.42 ± 3.98	69.92 ± 0.50	73.81 ± 2.06	61.34 ± 2.92	28.30 ± 3.24	64.35

From Tables (4-2) and (4-3), compared with using all features in clustering, it is apparent that the ACC and NMI performances have grown extensively after the feature selection.

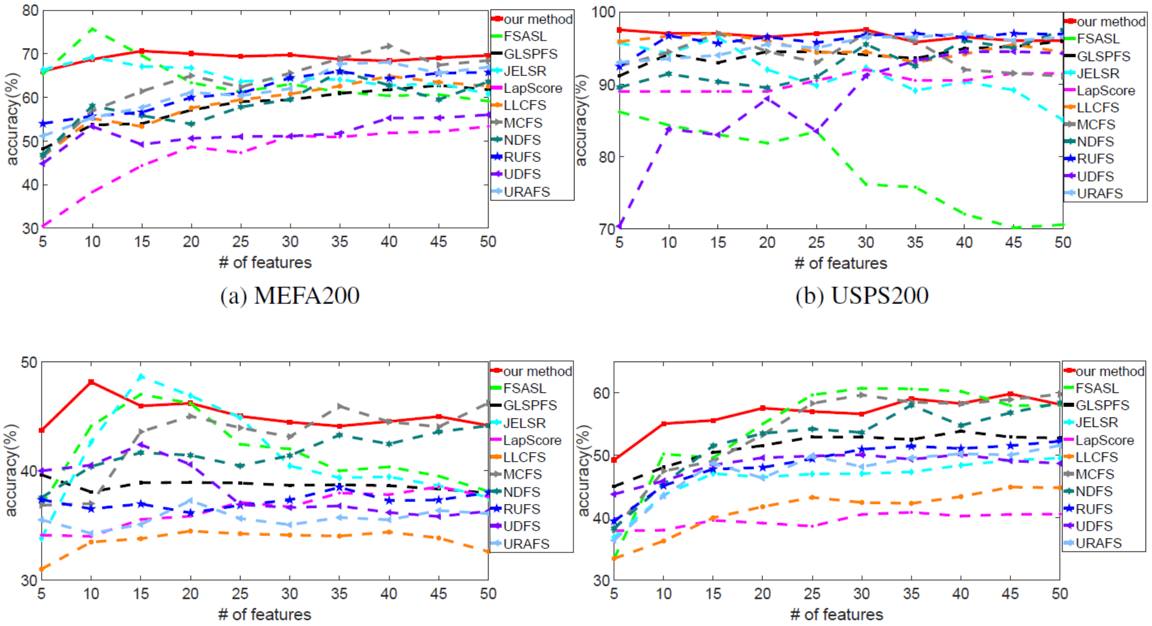
These experimental results demonstrated the significance of processing feature selection to reduce the redundancies and noises in the datasets. In Tables (4-2) and (4-3), the experimental outcomes of UFSARP are superior to the LapScore, MCFS, LLCFS, UDFS, RUFS, JELSR, URAFS and GLSPFS.

In Table (4-2), the ACC rates of UFSARP is slightly larger than the framework FSASL and URAFS on the datasets JAFFE, COIL, LUNG, and TOX. In the experiments on dataset UMIST, FSASL has the highest ACC, and the experiments on AR dataset, URAFS has obtained the biggest ACC. In addition, the UFSARP framework achieved the most favorable average performance outcome measured by ACC across six various datasets.

In Table (4-3), it is clear that UFSARP achieved better performances in NMI than the approaches FSASL and URAFS, on the datasets JAFFE, COIL, and TOX. While, the FSASL obtained a better rate on the datasets UMIST, AR, and LUNG.

Both Table (4-2) and (4-3) indicated that a framework achieved the highest ACC rate cannot guarantee that the framework is going to obtain the best NMI result. Also, it is essential to specify that the UFSARP framework has produced 16.58% and 6.37% higher performance rates (measured by ACC and NMI respectively) at the time of using less than 10% of data features.

A series of detailed experiments have been conducted considering the influences of choosing the different number of features, on the datasets ORL, YALE, USPS200, and MEFA200. The performance results were illustrated in Figures (4-2) and (4-3), in the value of ACC rate and NMI rate respectively. In Figures (4-2) and (4-3), the Red-Line describes the performances of UFSARP, which operates better than other conventional approaches, particularly in lower-dimensional spaces. Potential inferences may because that UFSARP involves a term of data reconstruction residuals which attempts to preserve the intrinsic local manifold structure. From above figures, it is clear that the ACC and NMI rates are continuously growing along with the rising number of features, however, the optimal outcome does not present at the time of the highest number of features. This indicates that most of the data information can be expressed by a small subset of features, and conversely, each dataset has certain redundant and noisy features. These experiments conclude that a substantial number of features in the computation does not guarantee a higher ACC/NMI result.



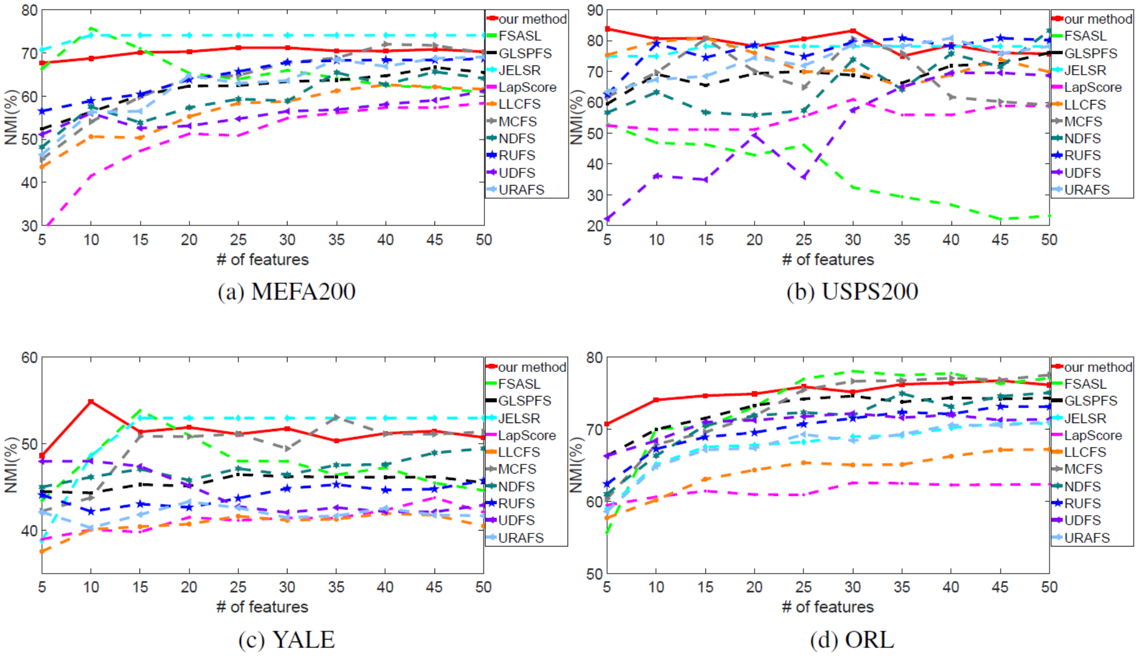
(a) MEFA200

(b) USPS200

(c) YALE

(d) ORL

Figure 4-2: The ACC (%) Result versus #. of Features



(a) MEFA200 (b) USPS200

(c) YALE (d) ORL

Figure 4-3: The NMI (%) Result versus #. of Features

To sum up, based on the extensive experiments have been conducted in this section, USFARP achieved a steady and higher ACC performance on average. It is apparent that the experiments as followings:

(1) The performances of clustering with a feature selection approach are generally better than those performances involving all features. This emphasizes that the feature selection approaches work effectively in eliminating the redundancies and noises.

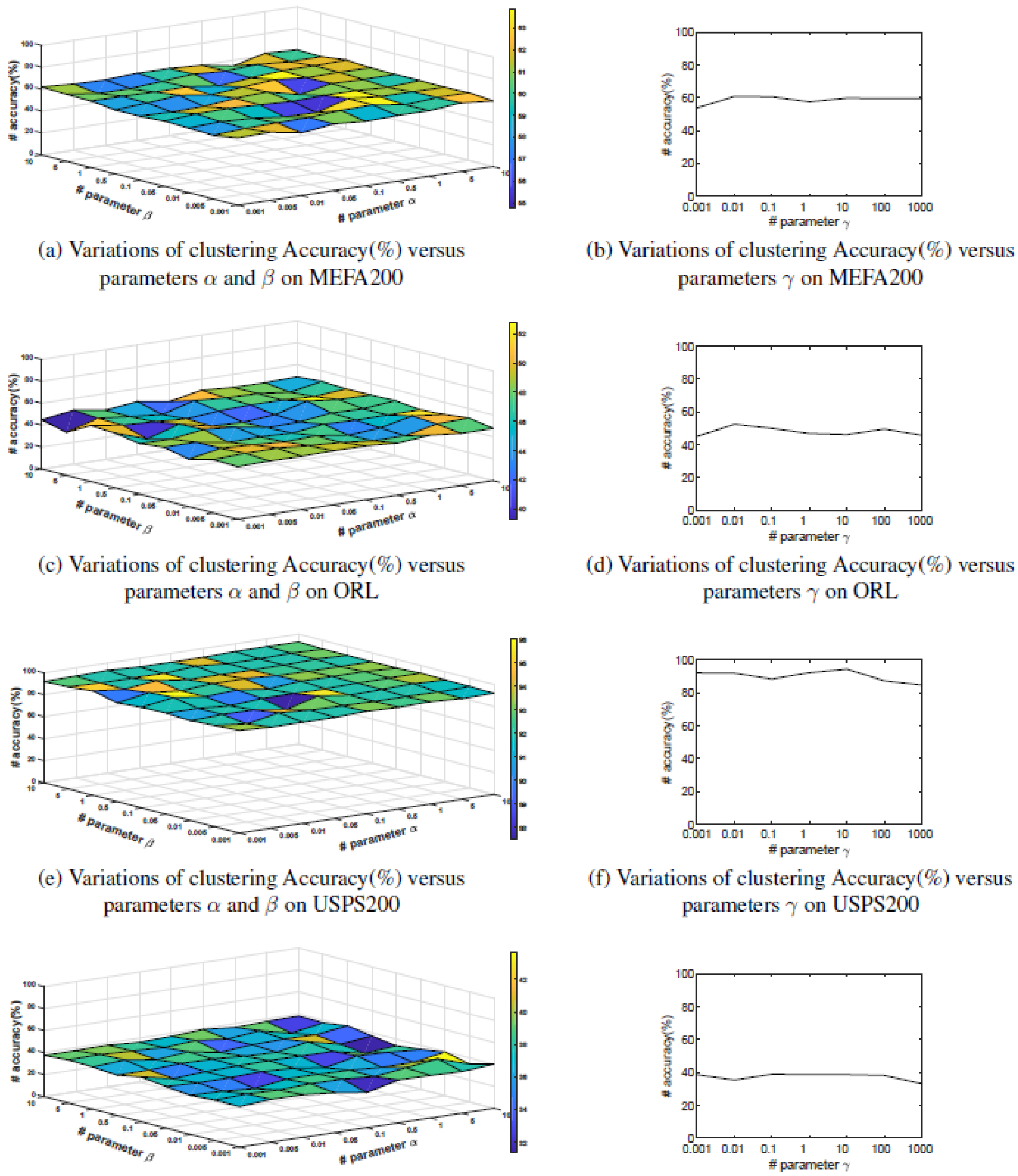
(2) The performance of UFSARP is better than other similar approaches on most datasets. The main reason may be because with the introduction of DRR, UFSARP is capable of precisely retain the data local manifold structure.

(3) Generally speaking, the ACC/NMI rates are increasing along with the growing number of features. However, the experimental results turned into the stable at a certain level of feature numbers. Furthermore, UFSARP achieves high-performance results even in a low dimensional space.

4.6.4 Parameter Sensitivity

The sensitivity of parameters α, β, and γ are evaluated separately by keeping other parameters unchanged. During the process of testing the sensitivity of α and β, the value of γ is set to be fixed. The Figure (4-4) (a), (c), (e) and (g) have demonstrated the performances measured by ACC rate at a different value of α and β. From those figures, it is clear that the ACC rates are generally smooth, and indicates that the UFSARP framework is not sensitive to the value of parameters α and β. Then, holding the values of α and β unchanged, the changes in ACC corresponding to the changes of γ is shown in Figure (4-4) (b), (d), (f) and (h). It is obvious that the clustering results are not sensitive to the value of parameter γ either. But a slightly better result can be obtained in the ranges of $[10^{-2}, 10^{-1}]$ or $[10, 10^{2}]$.

In addition, there is an experiment regarding the sensitivity concerning the number of features, as shown in Figures (4-5) and (4-6) on the datasets JAFFE and USPS200 respectively. The clustering performance results are generally rising with the increasing number of selected features, and it is straightforward to discover that UFSARP framework is robust to the value of parameters α, β, and γ when the feature numbers are at a certain level. In Figure (4-5), the ACC rates fluctuate and depress at the range of small feature numbers. The reason for this may be because the selected feature numbers are too little to represent the information of face images. However, in Figure (4-6), the clustering performance is robust to the number of features selected. Even at the level of only 5 features, the results are still impressive. This may be because the dataset USPS200 is the handwritten dataset, which is much clearer to distinguish than those face images.



(a) Variations of clustering Accuracy(%) versus parameters α and β on MEFA200

(b) Variations of clustering Accuracy(%) versus parameters γ on MEFA200

(c) Variations of clustering Accuracy(%) versus parameters α and β on ORL

(d) Variations of clustering Accuracy(%) versus parameters γ on ORL

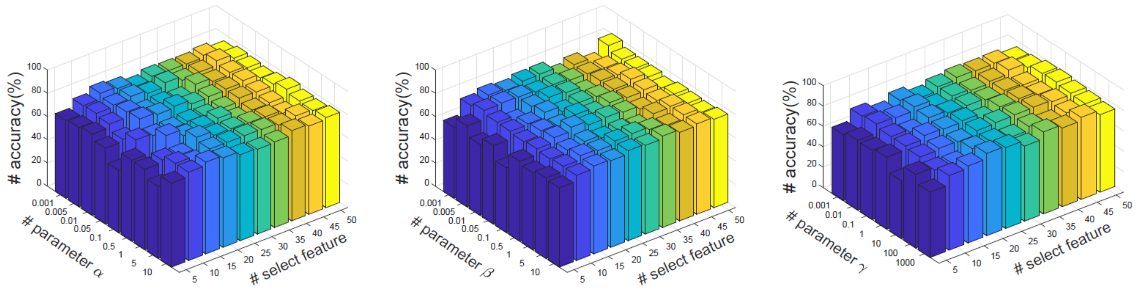
(e) Variations of clustering Accuracy(%) versus parameters α and β on USPS200

(f) Variations of clustering Accuracy(%) versus parameters γ on USPS200

(g) Variations of clustering Accuracy(%) versus parameters α and β on YALE

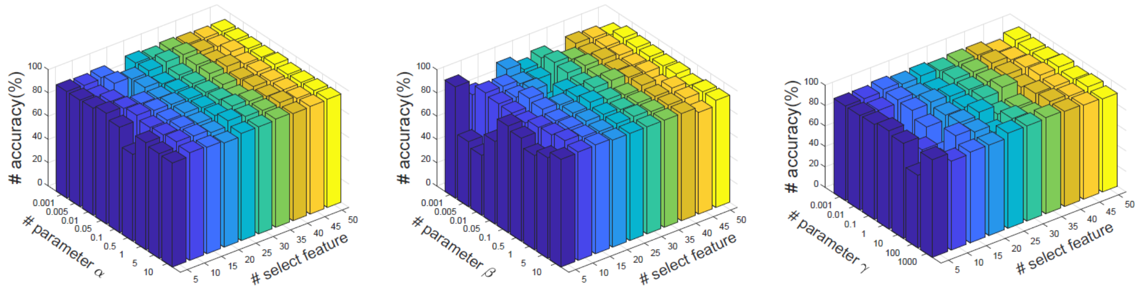
(h) Variations of clustering Accuracy(%) versus parameters γ on YALE

Figure 4-4: Clustering ACC versus Different Value of α, β and γ



(a) α (b) β (c) γ

Figure 4-5: The Clustering ACC versus Different #. of Features on JEFFE Database



(a) α (b) β (c) γ

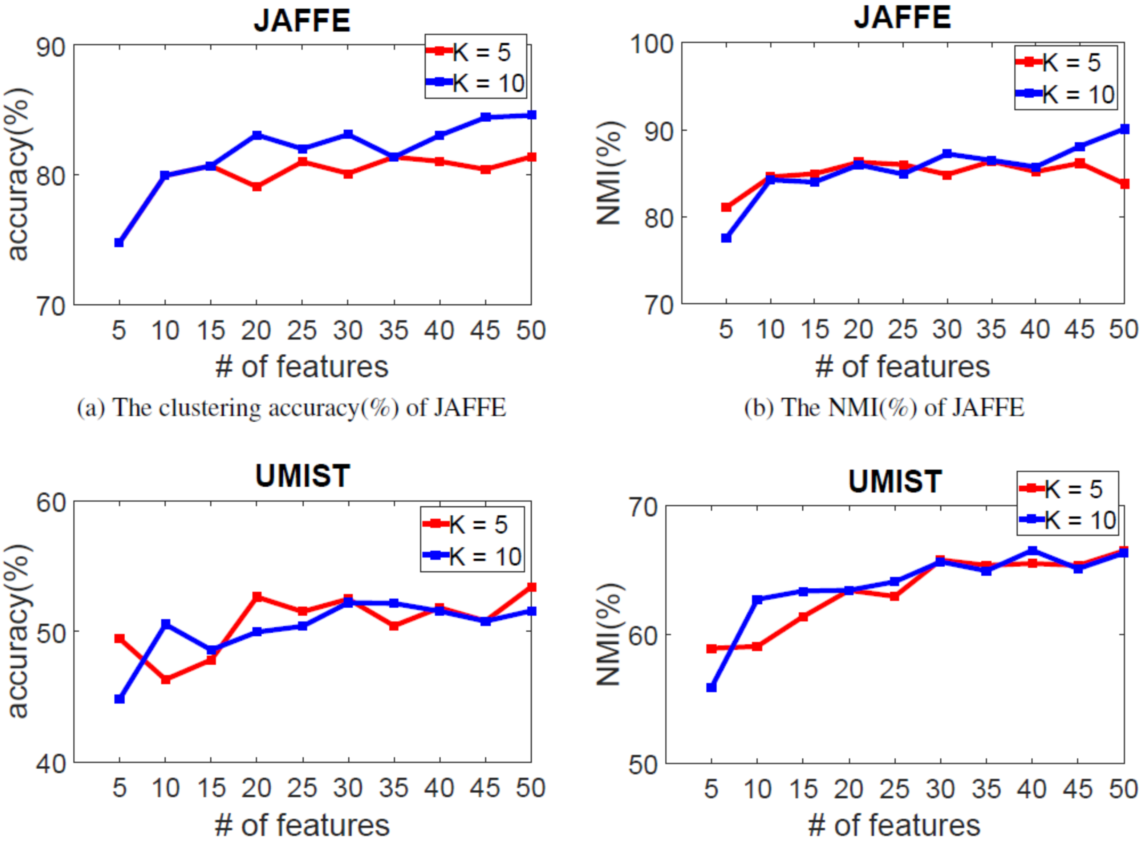
Figure 4-6: The Clustering ACC versus Different #. of Features on USPS200 Database

In short, the framework UFSARP is either sensitive to the value of parameters α, β, and γ, nor to the number of features selected, which indicate the robustness of UFSARP.

4.6.5 Effect of Neighborhood Size and Running Time

All the above experiments are developed based on $k = 5$, which means the number of neighbors has been set to be 5 to construct the data structure. This section will test the sensitivity of the UFSARP regarding with the changing in the size of neighbors. Experiments of $k = 10$ on the datasets JAFFE and UMIST have been conducted, and the experimental results are showing as in Figure (4-7).

In the figure, it is clear that UFSARP performs better when the number of neighbors has been changed to $k = 10$ on the dataset JAFFE. While, on the dataset UMIST, UFSARP achieved a higher ACC result in $k = 5$, and a higher NMI result in $k = 10$. To sum up, the overall performances of UFSARP in $k = 5$ and $k = 10$ are similar.



(a) The clustering accuracy(%) of JAFFE

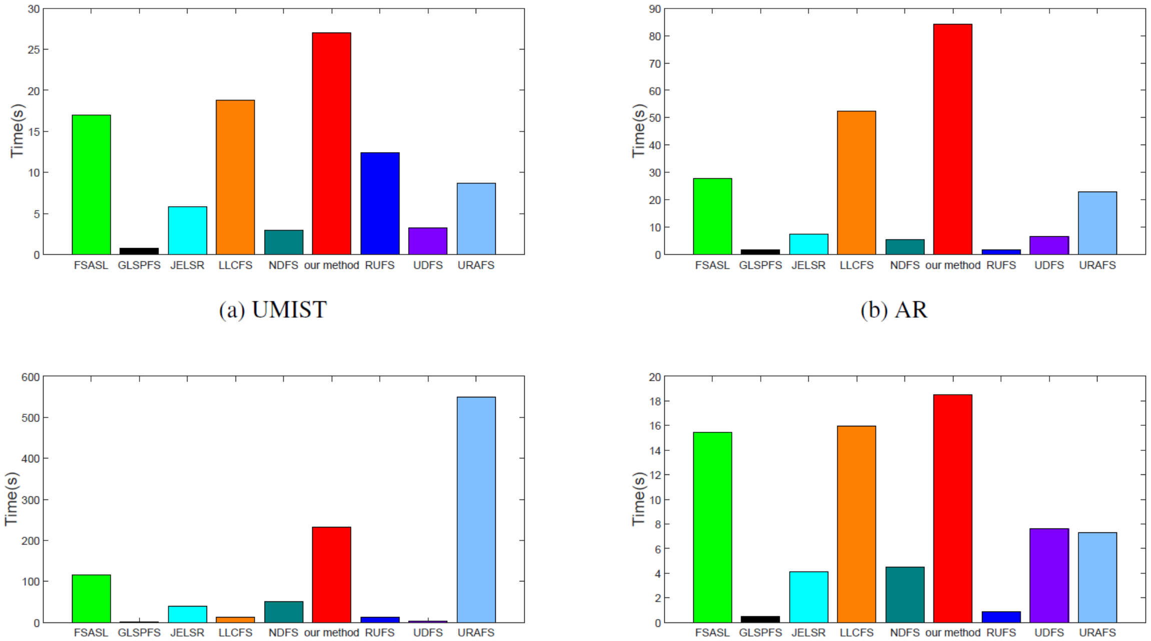
(b) The NMI(%) of JAFFE

(c) The clustering accuracy(%) of UMIST

(d) The NMI(%) of UMIST

Figure 4-7: Clustering Results at k=5 and k=10

Furthermore, Figure (4-8) shows the operating time of each approach on different datasets, UMIST, AR, LUNG and ORL. All experiments that have been carried in this chapter are implemented on MATLAB R2014b, and the codes were run on a Windows 10 Laptop with 2.80-GHz i7-7700HQ CPU, 16 GB main memory. From Figure (4-8), it is apparent that UFSARP takes a slightly higher amount of running time than other approaches. With the concern of the robustness and effectiveness attributes of UFSARP, the longer running time is recognized to be reasonable and is within an acceptable range.



(a) UMIST

(b) AR

(c) LUNG

(d) ORL

Figure 4-8: The Running Time of Each Approach

4.7 Conclusion

This chapter has proposed a novel unsupervised feature selection approach, UFSARP, which enhances the credibility of data reconstruction structure. The learning model allows processing the construction of data manifold structure and the feature selection simultaneously, which avoids the shortcoming of the pre-defined data graph structure. In addition, a new idea, DRR, has been added into the framework. With the consideration of DRR, the framework ensures the quality of data local manifold reconstruction structure. The extensive experiments have shown that UFSARP is superior to all other similar approaches.

In UFSARP, the computational complexity is relatively high. It is expected to apply the DRR to other uncomplicated dimensionality reduction frameworks in order to reduce the computational complexity.

Chapter 5 Joint Sparse Representation and Low-rank Constraint

This work has been done and submitted to an academic journal. It is under the process of waiting for reviewers' responses.

Recently, numerous dimensionality reduction methods have been proposed. However, these approaches are experiencing a common problem that they are all disregarding the within-class and between-class structure of the datasets. As a consequence, a randomly selected testing sample would be described by a combination of samples from multiple sample classes, and it will further influence the outcome of classification.

In order to overcome the problem, we propose a novel supervised feature extraction approach on face image classification, named Sparse Representation based Classifier with Low-rank Constraint (SRCLC). In SRCLC, Low-rank Representation (LRR) is operating not only in maintaining the within-class data structure but also in preserving the between-class data structure. Therefore, the class structure in the datasets is going to be more distinguishable. With the remarkable performances of data reconstructed structure, the accuracy rate of classification is going to be improved.

The extensive experimental results have demonstrated that SRCLC is superior over the state-of-the-art approaches.

5.1 Introduction

During the past decades, multiple algorithms have been proposed, and researchers tend to employ regularization norms to obtain simple but identifiable data representation. SR and LRR are two wildly used approaches that regularize the framework employing l_1-norm, l_2-norm, and nuclear norm.

There is plenty of literature has proved that norms help to recognize construct the data structure. (Wright, Yang, Ganesh, Sastry, & Ma, 2009) are the first people that implemented SR in the face recognition domain. With the idea of automatically recognizing human faces, Wright, et. al. discussed and proved the effectiveness and robustness regarding the sparse signal representation. Then in 2011, (Zhang, Yang, & Feng, 2011) argued that it is the Collaborative Representation having the efficacy in classification. They introduced a CR based classification method with a regularization term that built with least square. However, these conventional approaches are not able to differentiate similar samples from different classes. Therefore, an unreliable classification decision may be derived when the samples are coming from two different classes but are similar to each other, or when the dictionary is not over-completed.

In order to resolve this problem, approaches that discriminate different groups have been proposed. In 2006, (Yuan & Lin, 2006) extended the LASSO, LARS and non-negative Garrotte algorithms with a factor of grouped variables, and have conducted a set of extensive experiments to show the superior performances of the extensions. Then in 2009, (Majumdar & Ward, 2009) proposed two regularization approaches, Elastic Net and Sum-Over-l_2-norm to select a set of most representative training samples from the whole training set. In 2015, (Yang, Kong, Fu, Li, & Zhao, 2015) introduced a semi-supervised learning approach that is using grouped sparsity to automatically discover pairwise comparisons. With the idea of group sparsity, training samples from a particular class that have the closest properties to the test sample would approximately form a linear representation for the test sample. Thus, in the Group Sparse Representation (GSR), non-zero weights only work in a particular data class.

In addition, the LRR has been introduced to resolve the problem of unavoidable effects of redundant and noisy features. Plenty of literature has determined that the combinations of features are more distinguishable than individual features. Namely, a set of features lied in a new subspace are more representative. In LRR, data from the same class is considered to lie in the same Low-Rank subspace. In 2013, (Zhang, Jiang, & Davis, 2013) presented a framework that employed LRR for image classification. Zhang et al. constructed a discriminative dictionary that is using a Low-Rank matrix recovery to train samples from all classes for classification tasks. Then, in 2014, (Tang, Liu, Su, & Zhang, 2014) proposed Structure-Constrained LRR to address the problem that LRR can only be effective when all subspaces are independent.

Inspired by the GSR and LRR, this chapter is going to propose a novel supervised approach that combines the GSR and LRR into a unified framework, the SRCLC. SRCLC is built with two LRR terms, which does not only regularize the within-class sample points to be more similar but also distinguish the between-class sample points from each other. Therefore, a more compact and discriminative data representation will be obtained.

The contributions of this framework are as followings:

1. SRCLC obtains a compact data representation from a new subspace that can effectively eliminate the negative effects of redundancies and noises.
2. LRR constraints are operating in regularizing the within-class data structure and between-class data structure concurrently, which further enable the framework to determine more effective data class information and enhance the group discriminating capacity.
3. The newly attached LRR constraints can be extended and utilized in other frameworks with its robust attribute.

The rest of this chapter is organized as followings:

An introduction of recent relevant researches will be presented in Section 5.2, and the detail explanation of SRCLC is going to be shown in Section 5.3. Section 5.4 will discuss comprehensive experiments that have been conducted. Lastly, a conclusion of the entire chapter is going to be provided in Section 5.5.

5.2 Related Works

Given that LRR has been introduced in section 2.6, we are going to introduce the concept of GSR in this section.

5.2.1 Group Sparse Representation

In GSR, the randomly selected testing sample will be represented as a linear combination of the training set from a particular class as the testing sample belongs to. This assumption can be written formally as the followings:

Given a testing sample $\boldsymbol{y}_{k,j} \in R^d$, which belongs to the class k, then,

$$\boldsymbol{y}_{k,j} = \alpha_{k,1}\boldsymbol{x}_{k,1} + \alpha_{k,2}\boldsymbol{x}_{k,2} + \cdots + \alpha_{k,n}\boldsymbol{x}_{k,n} + \varepsilon \tag{5-1}$$

In the above Problem (5-1), $\boldsymbol{x}_{k,i}$ are the training samples coming from the k-th class.

In terms of expressing the function regarding all the training samples, the problem (5-1) can be rewritten as:

$$\boldsymbol{y}_{k,j} = \boldsymbol{X\alpha} + \varepsilon \tag{5-2}$$

where $\boldsymbol{X} = \left[\boldsymbol{x}_{1,1}, \dots, \boldsymbol{x}_{1,n_1}, \dots, \boldsymbol{x}_{c,1}, \dots, \boldsymbol{x}_{c,n_c}\right]$, $\boldsymbol{\alpha} = \left[\alpha_{1,1}, \dots, \alpha_{1,n_1}, \dots, \alpha_{c,1}, \dots, \alpha_{c,n_c}\right]^T$, and c is the total number of classes in the dataset.

In the GSR, the Equation (5-2) should satisfy two implications:

1. The vector $\boldsymbol{\alpha}$ should be sparse;
2. All elements of $\boldsymbol{\alpha}$ should equal to zero, except elements corresponding to the class that the testing sample belongs to.

These lead the target has been changed to solve the following objective problem:

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_{2,0}$$

$$s.t. \left\|\boldsymbol{y}_{k,j} - \boldsymbol{X\alpha}\right\|_2 < \varepsilon$$

(5-3)

Problem (5-3) is an NP-hard problem, it can be approximately solved by using an $l_{2,1}$-norm regularization.

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_{2,1}$$

$$s.t. \left\|\boldsymbol{y}_{k,j} - \boldsymbol{X\alpha}\right\|_2 < \varepsilon$$

(5-4)

Under the perception of GSR, the selected samples that used to represent the testing sample are assumed to share similar properties. However, with the unavoidable redundancies and noises in datasets, the reconstructed structure may be unreliable. SRCLC is a novel proposed learning model that generates new constraints to eliminate the impacts of irrelevant information.

5.3 Sparse Representation-based Classifier with Low-rank Constraints (SRCLC)

SRCLC framework combines SR and LRR to satisfy the following expectations:

1. Data from the same class is sharing certain attributions, and the data features should be lying in the same low-rank subspace;
2. Data from different classes are distinguishable from each other, and so each feature subspace is discriminable.

To satisfy the abovementioned two expectations, SRCLC is going to impose two LRR terms. One implements to regularize the within-class samples while the other LRR serves to discover the between-class structure of the dataset. As a result, SRCLC acknowledges both the within-class structure and between-class structure.

The SRCLC framework is designated as:

$$\min_{\boldsymbol{W},\boldsymbol{Z}} \frac{1}{2}\|\boldsymbol{W}^T\boldsymbol{Y}-\boldsymbol{W}^T\boldsymbol{X}\boldsymbol{Z}\|_F^2+\frac{\lambda_1}{2}\|\boldsymbol{Z}\|_{2,1}+\left(\lambda_2\sum_{k=1}^{c}\left\|\boldsymbol{W}^T\boldsymbol{X}^{(k)}\right\|_*-\lambda_3\|\boldsymbol{W}^T[\boldsymbol{X}\ \boldsymbol{Y}]\|_*\right)$$

$$s.t. \quad \boldsymbol{W}^T\boldsymbol{W}=\boldsymbol{I}$$

(5-5)

In the above framework, the term $\left(\lambda_2\sum_{k=1}^{c}\left\|\boldsymbol{W}^T\boldsymbol{X}^{(k)}\right\|_*-\lambda_3\|\boldsymbol{W}^T\boldsymbol{X}\|_*\right)$ regularized samples from the same class to be closed whilst samples from different classes to be far apart. $\boldsymbol{X}^{(k)}$ represents the scatters within one class, and $\sum_{k=1}^{c}\left\|\boldsymbol{W}^T\boldsymbol{X}^{(k)}\right\|_*$ indicates that the scatters come from the same class to be similar. However, without the constraint $\|\boldsymbol{W}^T[\boldsymbol{X}\ \boldsymbol{Y}]\|_*$, inappropriate classification outcomes may be generated especially when two classes are sharing similar attributes. To solve this problem, SRCLC attaches another LRR, a negative term $\lambda_3\|\boldsymbol{W}^T\boldsymbol{X}\|_*$ into the learning model. This LRR expresses a negative low-rank structure for the entire dataset. Under the expectation of minimizing the total value of the framework, the negative LRR term is going to be maximized which intends to distinguish every individual sample point. Later, following the impact of former minimization LRR, $\lambda_2 \sum_{k=1}^{c} \left\| \boldsymbol{W}^T \boldsymbol{X}^{(k)} \right\|_*$, that regularized the within-class structure, the LRR term as a whole, $\left(\lambda_2 \sum_{i=1}^{c} \left\| \boldsymbol{W}^T \boldsymbol{X}^{(k)} \right\|_* - \lambda_3 \| \boldsymbol{W}^T \boldsymbol{X} \|_*\right)$, attempts to lessen the distance of within-class samples and to extend the distance of between-class samples.

Definitions and Proofs:

Suppose there is a linear regression model:

$$\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{\varepsilon}$$

where $\boldsymbol{y}$ is a dependent variable, $\boldsymbol{y} \in R^n$, $\boldsymbol{X}$ is a matrix, $\boldsymbol{X} \in R^{n \times (p+1)}$, $\boldsymbol{\beta}$ is a vector that $\boldsymbol{\beta} \in R^{(p+1)}$, and $\boldsymbol{\varepsilon}$ is a random vector, $\boldsymbol{\varepsilon} \in R^n$.

Definition 11.: Let $\boldsymbol{y}$ to be an actual value, and $\widehat{\boldsymbol{y}}$ to be a fitted value, the residual of the linear regression model is defined as:

$$\hat{\boldsymbol{\varepsilon}} = \boldsymbol{y} - \widehat{\boldsymbol{y}}$$

Suppose there is a matrix $\boldsymbol{A} = \left[a_{11}, a_{12}, \dots, a_{1n_1,} a_{21}, a_{22}, \dots, a_{2n_2,} \dots, a_{c1}, a_{c2}, \dots, a_{cn_{c,}}\right]^T$, a set of data $\boldsymbol{X} = \left[x_{11}, x_{12}, \dots, x_{1n_1,} x_{21}, x_{22}, \dots, x_{2n_2,} \dots, x_{c1}, x_{c2}, \dots, x_{cn_{c,}}\right] \in R^{d \times n}$, c is the number of classes, and n_i is the number of samples in every class.

Definition 12.: For $\forall \boldsymbol{y}_k \in \boldsymbol{X}$ and $\boldsymbol{y}_k$ belongs to the class k, the group expression is defined as:

$$\boldsymbol{y}_k = \boldsymbol{X}\boldsymbol{a} = \sum_{n_i} a_{kn_i} x_{kn_i}$$

Definition 13.: For $\forall \boldsymbol{y}_k \in \boldsymbol{X}$, $\boldsymbol{y}_k$ *belongs to the class* k, $\boldsymbol{y}_k$ *is the group sparse expression if it conforms to the following two conditions:*

1. *The vector* $\boldsymbol{\alpha}$ *is sparse*
2. *All elements in* $\boldsymbol{\alpha}$ *should equal to zero, except elements corresponding to the class that the* $\boldsymbol{y}_k$ *belongs to.*

Lemma 1: Given two samples, $\boldsymbol{x}_i$ *and* $\boldsymbol{x}_j$, *are adjacent, the value of* $\left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|^2$ *will be small.*

Proof: Since the samples, $\boldsymbol{x}_i$ *and* $\boldsymbol{x}_j$, *are adjacent (ε-balls) with each other, which means the sample* $\boldsymbol{x}_i$ *is inside ε-balls of* $\boldsymbol{x}_j$, *we have* $\left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|^2 < \boldsymbol{\varepsilon}$ *or* $\left\|\boldsymbol{x}_j - \boldsymbol{x}_i\right\|^2 < \boldsymbol{\varepsilon}$, *and the value of* $\left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|^2$ *will be small.*

Lemma 2: Given samples $\boldsymbol{y}_k$ *to be expressed by the group of sparse expression of dataset* $\boldsymbol{X}$, *the value of* $||\boldsymbol{y}_k - \sum_{n_i} a_{kn_i} x_{kn_i}||$ *will be small.*

Proof: Since $\boldsymbol{y}_k$ *can be expressed by the group sparse expression of* $\boldsymbol{X}$, *we have:*

$$\boldsymbol{y}_k = \boldsymbol{X}\boldsymbol{a} + \boldsymbol{\varepsilon} = \sum_{n_i} a_{kn_i} x_{kn_i} + \boldsymbol{\varepsilon}$$

where, $\boldsymbol{X} = \left[x_{11}, x_{12}, \ldots, x_{1n_1,} x_{21}, x_{22}, \ldots, x_{2n_2,} \ldots, x_{c1}, x_{c2}, \ldots, x_{cn_c,}\right]$,

and $\boldsymbol{a} = \left[0, \ldots, 0, a_{k1}, a_{k2}, \ldots, a_{kn_k,} 0, \ldots, 0\right]^T$.

Therefore, the value of $||\boldsymbol{y}_k - \sum_{n_i} a_{kn_i} x_{kn_i}||$ *is small.*

Lemma 3: Given two samples, $\boldsymbol{y}_s$ and $\boldsymbol{y}_k$, is adjacent (ε-balls) with each other, where the $\sum_{n_i} a_{sn_i} x_{sn_i}$ and $\sum_{n_i} a_{kn_i} x_{kn_i}$ are group sparse expressions of $\boldsymbol{y}_s$ and $\boldsymbol{y}_k$ respectively, then, the value of $\left\|\sum_{n_i} a_{sn_i} x_{sn_i} - \sum_{n_i} a_{kn_i} x_{kn_i}\right\|$ will be small.

Proof: According to the trigonometric inequality: $\left\|\sum_{n_i} a_{sn_i} x_{sn_i} - \sum_{n_i} a_{kn_i} x_{kn_i}\right\|$,

$$\left\|\sum_{n_i} a_{sn_i} x_{sn_i} - \sum_{n_i} a_{kn_i} x_{kn_i}\right\|^2 = \left\|\sum_{n_i} a_{sn_i} x_{sn_i} - \sum_{n_i} a_{kn_i} x_{kn_i} + \boldsymbol{y}_s - \boldsymbol{y}_s + \boldsymbol{y}_k - \boldsymbol{y}_k\right\|^2$$

$$\leq \left\|\boldsymbol{y}_s - \sum_{n_i} a_{sn_i} x_{sn_i}\right\|^2 + \left\|\boldsymbol{y}_k - \sum_{n_i} a_{kn_i} x_{kn_i}\right\|^2 + \|\boldsymbol{y}_s - \boldsymbol{y}_k\|^2$$

Given that $\sum_{n_i} a_{sn_i} x_{sn_i}$ and $\sum_{n_i} a_{kn_i} x_{kn_i}$ are group sparse expression of $\boldsymbol{y}_s$ and $\boldsymbol{y}_k$, $\left\|\boldsymbol{y}_s - \sum_{n_i} a_{sn_i} x_{sn_i}\right\|^2$ and $\left\|\boldsymbol{y}_k - \sum_{n_i} a_{kn_i} x_{kn_i}\right\|^2$ are small, and additional to Lemma 1, the value of $\|\boldsymbol{y}_s - \boldsymbol{y}_k\|^2$ is small, then $\left\|\sum_{n_i} a_{sn_i} x_{sn_i} - \sum_{n_i} a_{kn_i} x_{kn_i}\right\|^2$ will be small as well.

Theorem 3: Given $\boldsymbol{y}_s$ and $\boldsymbol{y}_k$ are adjacent (ε-balls) with each other, and $\sum_{n_i} a_{sn_i} x_{sn_i}$ and $\sum_{n_i} a_{kn_i} x_{kn_i}$ are group sparse expressions of $\boldsymbol{y}_s$ and $\boldsymbol{y}_k$ respectively, then the group sparse expressions will be adjacent to each other.

Proof: Refers to Lemma 1, Lemma 2, and Lemma 3, $\left\|\sum_{n_i} a_{sn_i} x_{sn_i} - \sum_{n_i} a_{kn_i} x_{kn_i}\right\|^2$ is small, $\sum_{n_i} a_{sn_i} x_{sn_i}$ and $\sum_{n_i} a_{kn_i} x_{kn_i}$ are adjacent to each other.

5.4 Optimization

5.4 Optimization

To solve the Problem (5-5), the Alternating Direction Method of Multipliers will be employed (ADMM) (Boyd, Parikh, Chu, Peleato, & Eckstein, 2010). During the calculation, every variable will be updated by holding the value of all other variables unchanged in each iteration.

It is worthwhile to mention that the Problem (5-5) is a non-differentiable and non-convex problem, and it is possible that developing a more advanced optimization technique will further improve the performance of the framework. In this section, we choose a simple projected sub-gradient based optimization algorithm that drives to fast convergence.

5.4.1 Update *W*

The value of $\boldsymbol{W}$ can be updated by keeping $\boldsymbol{Z}$ unchanged, and the problem regarding $\boldsymbol{W}$is:

$$J = \arg\min_{\boldsymbol{W}} \frac{1}{2} \|\boldsymbol{W}^T\boldsymbol{Y} - \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{Z}\|_F^2 + \left(\lambda_2 \sum_{k=1}^{c} \left\|\boldsymbol{W}^T\boldsymbol{X}^{(k)}\right\|_* - \lambda_3 \|\boldsymbol{W}^T[\boldsymbol{X}\ \boldsymbol{Y}]\|_*\right)$$

$$s.t. \quad \boldsymbol{W}^T\boldsymbol{W} = \boldsymbol{I}$$

(5-6)

$\|\cdot\|_*$ in the framework is a nuclear norm. Let $\boldsymbol{A} = \lambda_2 \sum_{k=1}^{c} \partial\left\|\boldsymbol{W}^T\boldsymbol{X}^{(k)}\right\|_* - \lambda_3\partial\|\boldsymbol{W}^T[\boldsymbol{X}\ \boldsymbol{Y}]\|_*$, where $\partial\|\cdot\|$ is the subdifferential of the norm $\|\cdot\|$. Suppose there is a matrix $\boldsymbol{B}$, the subdifferential $\partial\|\boldsymbol{B}\|$ can be approximately calculated as the **Algorithm 5-1**.

Algorithm 5-1: Solving the sub-gradient of matrix with nuclear norm

```
Input: An $\boldsymbol{m} \times \boldsymbol{n}$ matrix $\boldsymbol{B}$, a small threshold value $\boldsymbol{\varepsilon}$
    1. Perform singular value decomposition: $\boldsymbol{B} = \boldsymbol{U\Sigma V}$;
    2. Let $\boldsymbol{s}$ to be the number of singular values that smaller than $\boldsymbol{\varepsilon}$;
    3. Partition $\boldsymbol{U}$ and $\boldsymbol{V}$, let $\boldsymbol{U} = \left[\boldsymbol{U}^{(\mathbf{1})}, \boldsymbol{U}^{(\mathbf{2})}\right]$ and $\boldsymbol{V} = \left[\boldsymbol{V}^{(\mathbf{1})}, \boldsymbol{V}^{(\mathbf{2})}\right]$, where $\boldsymbol{U}^{(\mathbf{1})}$ and $\boldsymbol{V}^{(\mathbf{1})}$ have $\boldsymbol{n} - \boldsymbol{s}$ columns;
    4. Generate a random matrix $\boldsymbol{C}$ with the size $\left((\boldsymbol{m} - \boldsymbol{n} + \boldsymbol{s}) \times \boldsymbol{s}\right)$, $\boldsymbol{C} = \frac{\boldsymbol{C}}{\|\boldsymbol{C}\|}$;
    5. $\boldsymbol{\partial}\|\boldsymbol{B}\|_* = \boldsymbol{U}^{(\mathbf{1})}\left(\boldsymbol{V}^{(\mathbf{1})}\right)^{\boldsymbol{T}} + \boldsymbol{U}^{(\mathbf{2})}\boldsymbol{B}\left(\boldsymbol{V}^{(\mathbf{2})}\right)^{\boldsymbol{T}}$;
    6. Return $\boldsymbol{\partial}\|\boldsymbol{B}\|_*$;
Output: The sub-gradient of nuclear norm $\boldsymbol{\partial}\|\boldsymbol{B}\|_*$;
```

Inspired by the research of (Wen & Yin, 2012), the value of $\boldsymbol{W}$ can be approximately determined by applying gradient during the iterations. Suppose $\boldsymbol{W}(t)$ is the outcome of t-th iteration, then, the skew-symmetric matrix equals to $\nabla = \boldsymbol{GW}(t)^T - \boldsymbol{W}(t)\boldsymbol{G}^T$, and $\boldsymbol{G}$ is the gradient of $\boldsymbol{W}$.

The gradient $\boldsymbol{G}$ can be derived:

$$\boldsymbol{G} = \boldsymbol{WYY}^T - \boldsymbol{WXZY}^T - \boldsymbol{WYZ}^T\boldsymbol{X}^T + \boldsymbol{WXZZ}^T\boldsymbol{X}^T + \boldsymbol{A} \tag{5-7}$$

Then, $\boldsymbol{W}$ can be updated by using the technique that has been introduced in (Wen & Yin, 2012):

$$\boldsymbol{W}(t+1) = \left(\boldsymbol{I} + \frac{\tau}{2}\boldsymbol{\nabla}\right)^{-1}\left(\boldsymbol{I} - \frac{\tau}{2}\boldsymbol{\nabla}\right)\boldsymbol{W}(t)$$

(5-8)

5.4.2 Update *Z*

The value of $\boldsymbol{Z}$ can be updated while keeping $\boldsymbol{W}$ unchanged. The problem regarding with $\boldsymbol{Z}$ is:

$$J = \arg\min_{\boldsymbol{Z}} \frac{1}{2}\|\boldsymbol{W}^T\boldsymbol{Y} - \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{Z}\|_F^2 + \frac{\lambda_1}{2}\|\boldsymbol{Z}\|_{2,1}$$

(5-9)

Letting $E_{ii} = \frac{1}{2\|z_i\|_2}$, where z_i is the i-th row of $\boldsymbol{Z}$. Then the Problem (5-9) can be solved by taking the partial derivative to $\boldsymbol{Z}$:

$$\boldsymbol{Z} = (\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{W}^T\boldsymbol{X} + \lambda_1\boldsymbol{E})^{-1}(\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{W}^T\boldsymbol{Y})$$

(5-10)

5.5 Computational Complexity and Convergence

The most time-consuming parts in the computation are the processes in resolving $\boldsymbol{W}$ and $\boldsymbol{Z}$. In determining $\boldsymbol{W}$, it involves an EVD operation and an inverse operation. The SVD operator runs on an $r \times n$ matrix, and the computational complexity is $\boldsymbol{O}(n^3)$. The inverse operator works on a $d \times d$ matrix, and the computational complexity is $\boldsymbol{O}(d^3)$. While in solving $\boldsymbol{Z}$, there is an inverse operation works on an $n \times n$ matrix that the computational complexity is $\boldsymbol{O}(n^3)$. Accordingly, the total computational complexity of the SRCLC as a whole is $\boldsymbol{O}\left(t(n^3 + d^3 + n^3)\right)$, where t denotes the number of iterations.

For the convergence, the SRCLC framework has experimented on two datasets, the YALE B and COIL20. Figure (5-1) and (5-2) illustrate the convergence status of SRCLC.

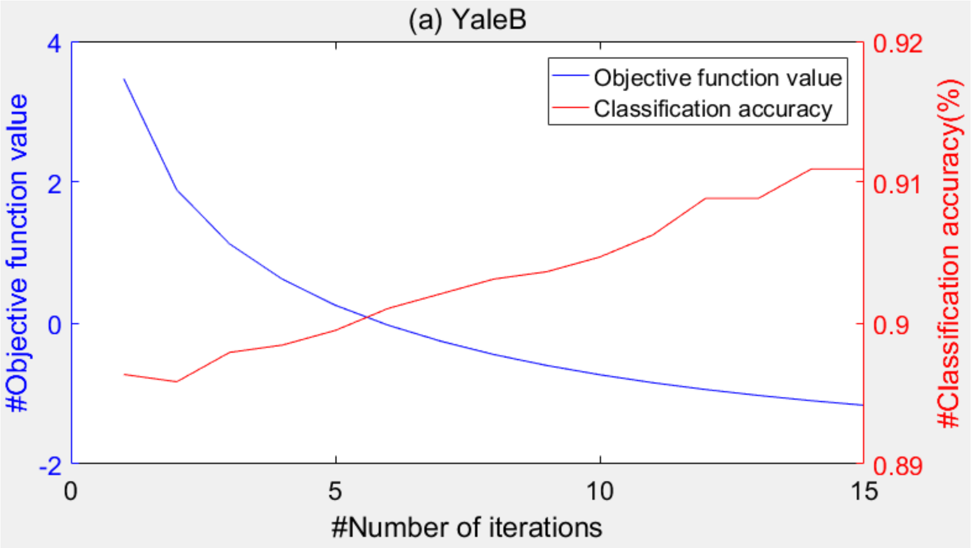


Figure 5-1: The Convergency of SRCLC on YALE B Database

In the above Figure (5-1), it is clear that the SRCLC is going to reach its convergence within 15 iterations on the Extended Yale B database. The value of the objective function is decreasing with a large scale at the beginning and converting smooth at the level around 10 iterations.

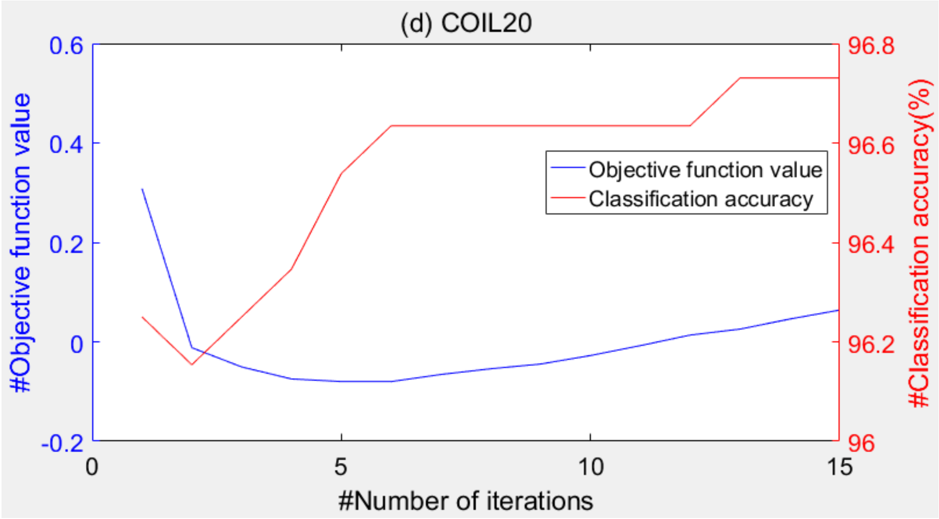


Figure 5-2: The Convergency of SRCLC on COIL20 Database

In Figure (5-2), the objective value and the classification precision rates were changing dramatically at the beginning. Then, after 5 iterations, the rates turned into steady.

Since SRCLC presents fast convergence capacity on both YALE B dataset and COIL20 dataset, it is reasonable to conclude that the framework has fast convergence capability. Therefore, SRCLC can obtain the classification result within small iterations. Consequently, SRCLC is a powerful learning model that can save a significant amount of computational time.

5.6 Experiments

As the article is under reviewed and waiting for the reviewers' responses, only a small part of the experiments has been demonstrated in this section.

SRCLC has been experimented on different datasets and has achieved impressive performances. Followings are two experiments that have been conducted on the face image datasets, Yale B and object image dataset COIL20. The comparison algorithms are PCA, LPP, NPE, LPP_L1, SPP, and JSRLPP. In addition, we have run 1-Nearest Neighbor Classifier (NN) as a benchmark to testify the effectiveness of the algorithms.

- PCA (Turk & Pentland, 1991): One of the most popular dimensionality reduction approach that projects the original dataset to the subspace that maximizes the data variances.
- NPE (He, Cai, Yan, & Zhang, 2005) & LPP (He, Yan, Hu, Niyogi, & Zhang, 2005): The linearized version of LE and LLE.
- SPP (Qiao, Chen, & Tan, 2010) & LPP_L1 (Liu, Yin, & Jin, 2010): The extensive method of L1-graph.
- JSRLPP (Zhang, Kang, Fang, Teng, & Han, 2018): A unified framework that combines SR and LPP.

Table 5-1: Classification ACC (%) for Various Methods on the COIL20 Database (Bold Numbers Denote the Best Result)

	5	**10**	**15**	**20**
NN	82.33 ± 1.55	89.55 ± 1.16	92.99 ± 0.83	95.08 ± 0.86
PCA	82.33 ± 1.55	89.55 ± 1.16	92.99 ± 0.83	95.06 ± 0.85
LPP	73.11 ± 1.82	80.99 ± 1.13	86.19 ± 1.21	88.27 ± 1.06
NPE	71.90 ± 2.09	79.54 ± 1.66	83.99 ± 1.28	85.02 ± 1.35
LPP_L1	81.39 ± 1.89	88.92 ± 1.10	92.58 ± 0.86	94.57 ± 0.84
SPP	81.91 ± 2.25	89.07 ± 1.14	92.51 ± 0.85	94.33 ± 0.83
JSRLPP	83.29 ± 1.74	91.24 ± 1.27	94.75 ± 0.77	96.60 ± 0.69
SRCLC	**84.17 ± 1.87**	**92.00 ± 1.40**	**95.32 ± 1.04**	**97.01 ± 0.70**

From the above Table (5-1), it is apparent that SRCLC has achieved the highest ACC rate, and the experimental result is improved along with the increment in the number of samples selected. In addition, we discovered that the NN and PCA are obtaining excellent performances without any dimensionality reduction methods. This may because that in the object dataset, each object is highly varied from the others, hence, the dataset has the attribute of distinction itself. However, SRCLC still enhances the classification performance, which indicates the effective in reducing the redundancies.

Table 5-2: Classification ACC (%) for Various Methods on the Yale B Database (Bold Numbers Denote the Best Result)

	5	**7**	**10**	**13**
NN	37.12 ± 1.42	44.83 ± 1.26	53.24 ± 1.05	59.85 ± 1.11
PCA	37.12 ± 1.42	44.83 ± 1.26	53.24 ± 1.05	59.85 ± 1.11
LPP	62.51 ± 2.66	71.80 ± 2.09	79.13 ± 1.23	82.19 ± 1.20
NPE	58.11 ± 2.04	67.82 ± 1.99	76.12 ± 1.45	79.27 ± 1.17
LPP_L1	71.30 ± 1.81	78.70 ± 1.20	84.83 ± 1.06	88.21 ± 1.07
SPP	62.22 ± 1.98	71.08 ± 1.16	78.53 ± 1.01	82.86 ± 0.80
JSRLPP	75.75 ± 1.54	83.10 ± 1.24	88.75 ± 0.76	90.36 ± 0.80
SRCLC	$\mathbf{77.66 \pm 1.52}$	$\mathbf{84.31 \pm 1.01}$	$\mathbf{89.39 \pm 0.81}$	$\mathbf{91.61 \pm 0.76}$

From the above experimental results, it is clear that SRCLC has the highest ACC rate on the Extended Yale B database as well. In addition, with the increase in the number of samples selected from each class, the ACC rate is increasing. Also, different from the COIL20 database, NN and PCA in the Extended Yale B database with all features obtained bad performances. This experiment demonstrated the significance of having an effective dimensionality reduction method.

5.7 JSRLPP vs. SRCLC

JSRLPP has been extensively discussed in chapter 3 which is a novel feature extraction approach that combines sparse representation and LPP in a unified framework. In the framework, LPP is using to maintain the data neighborhood structure while the global structure was barely determined by using sparse representation, as the major task for sparse representation is to eliminate noisy and irrelevant data points. However, JSRLPP is simple and easy calculated, namely, it saves a lot of computational time. The comprehensive experiences have proved that the JSRLPP has impressive performances.

Different from JSRLPP, SRCLC has employed two low-rank constraints to maintain the data structure. One is using to close the neighborhood relationships while the other one is using to further the distances between different classes. Thus, SRCLC can preserve and distinct the classes' structure. However, SRCLC has a higher computational complexity compared with JSRLPP, which means it requires more time in processing the datasets.

5.8 Conclusion

This chapter has introduced a novel dimensional reduction method that jointed SR with Low-Rank Constraints. The major innovation of this framework is the Low-Rank Constraints are not only working on the within-class scatters but also regularizing the between-classes scatters. Therefore, this approach can better represent the original data structure, which further improves the accuracy of the classification result. The comprehensive experiments have shown the superior of SRCLC.

This chapter has demonstrated experiments on two databases and compared the SRCLC framework with seven popular algorithms. The experimental results demonstrated the effectiveness and efficiency of this approach.

Due to the robustness and effectiveness attribution of Low-Rank Constraint, it is expected to further extend it to other framework or other areas.

Chapter 6 Future Works and Expectations

This chapter is going to describe the future researches that will be finished in the near future. It has been divided into two sections.

There are still plenty of works can be done on this topic. Kernel Trick is one of the most applicable ideas that can be attached to face recognition framework to improve the classification performance.

The aforementioned approaches have shown promising performance. However, since all of the individual sessions in the frameworks are designed to represent data from linear subspaces, these methods may not gain satisfactory outcomes when dealing with data from nonlinear subspaces. At the meantime, the Kernel Trick maps the original data samples into a high dimensional kernel space that the non-linear structure can be interpreted as linear combinations without destroying the original data intrinsic structure. Considering most of the real-world data come from nonlinear spaces, it is expected to add the Kernel Trick into learning models to enhance the data discrimination attribute and can further direct to higher classification accuracy. Therefore, this chapter is going to first introduce a novel learning model that combines SR, Low-Rank Constraint and Kernel Trick.

Then, in the second section, as Deep Learning has achieved successful outcomes in the Machine Learning domain. It is expected to apply the above researches into Deep Learning to enhance the performance of decision making. Hence, the second part of this chapter will present the future expectations of extending this research to Deep Learning domain.

6.1 Joint Sparse Representation, Low-Rank Constraints, and Kernel Trick Learning Approach

It is common to perform high-dimensional data clustering in various image recognition and data mining applications. The high-dimensions are represented by a large number of features in the dataset. However, discrimination among data classes is often impeded by the abundance of features. Such as in genomic data analysis, only a small subset of features from thousands of gene expression coefficients is capable to distinguish the different tissue classes.

In image dataset that is compressible, sparsity is a useful principle during the image processing. The testing sample is represented as a sparse combination of the whole training set, and the test sample is classified to the class that has minimum residual between. SR reflects the original data global relationships, and it is expected to build a joint framework that can determine both the global and local structure of the dataset. Inspired by the research presented in Chapter 5, which has introduced a framework that attached LRR to regularize the within-class and between-class data structure. In the framework, SR determines the data global structure, the within-class LRR pushes samples from the same class to be closed while the between-class LRR helps to distinguish data from different classes. Thus, SRCLC improves the discrimination of class structure of the dataset. However, SR algorithm assumes that the test sample to be a linear combination of the training sample set, and most image data are inherent with non-linear relationships. To solve this problem, this chapter is going to introduce research on adding the kernel-trick into the framework to improve the classification result.

There is numerous literature having been proposed on the research of kernel-trick in pattern recognition and machine learning. In 1998, (Schölkopf, Smola, & Müller, 1998) proposed the Kernel Principal Component Analysis (KPCA) and then, they have further added the Kernel Trick in LDA, which has been extended to Kernel Fisher Discriminant Analysis (KFD) (Mika, R¨atsch, Weston, Schölkopf, & Müller, 1999). In 2002, (Yu, Ji, & Zhang, 2002) applied Kernel Trick to the nearest-neighbor algorithm. By employing Kernel Trick, the samples that are linearly inseparable in the original feature space can be linearly separated after mapping the datasets into the high dimensional feature space. Since the Kernel Trick enables to extract the most discriminatory nonlinear features in the dataset, it has been found to be an effective approach in many real-world applications. However, the selection of appropriate Kernel Trick is important and time-consuming. To alleviate the efforts of selecting the most suitable kernel for a particular task, (Du, et al., 2015) proposed the Robust Multiple Kernel k-means using $l_{2,1}$-norm. In the framework, the clustering label, clustering membership and combination of multiple kernels are processed simultaneously.

Inspired by SRCLC in Chapter 5, it is expected to add the kernel trick into the framework to improve the performance.

6.1.1 The Framework

In the SRCLC, the learning model is built by SR with two different Low-Rank Constraints, and the function is as followings:

$$\min_{\boldsymbol{W},\boldsymbol{Z}} \frac{1}{2}\|\boldsymbol{W}^T\boldsymbol{Y} - \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{Z}\|_F^2 + \frac{\lambda_1}{2}\|\boldsymbol{Z}\|_{2,1} + \left(\lambda_2 \sum_{k=1}^{c} \left\|\boldsymbol{W}^T\boldsymbol{X}^{(k)}\right\|_* - \lambda_3\|\boldsymbol{W}^T[\boldsymbol{X}\ \boldsymbol{Y}]\|_*\right)$$

$$s.t. \quad \boldsymbol{W}^T\boldsymbol{W} = \boldsymbol{I}$$

(6-1)

where $\boldsymbol{W}$ is the dimensional reduction parameter, $\boldsymbol{Z}$ is the similarity matrix. The term $\frac{\lambda_1}{2}\|\boldsymbol{Z}\|_{2,1}$ is an $l_{2,1}$-norm that drives the elements in $\boldsymbol{Z}$ to be sparse. $\lambda_2 \sum_{i=1}^{c}\left\|\boldsymbol{W}^T\boldsymbol{X}^{(i)}\right\|_*$ is an LRR which ensures samples from the same class to be closer with each other. $(-\lambda_3\|\boldsymbol{W}^T[\boldsymbol{X}\ \boldsymbol{Y}]\|_*)$ is a negative LRR that attempts to distinguish each individual data sample. Therefore, the LRR as a whole in Function (6-1) In addition, SRCLC reduces the dimensions of sample data into a linear low dimensional feature space directly and ignores the inherent non-linear relationships in the dataset.

Inspired by SRCLC, a framework that copes with nonlinear relationships, jointed SR with Low-Rank Constraint and Kernel-Trick (KSRLC), is proposed in this chapter.

The dataset is going to be mapped into a high dimensional kernel space which enables samples present as linear combinations without destroying the original data intrinsic structure. KSRLC framework is represented as followings:

$$\min_{\boldsymbol{W},\boldsymbol{Z}} \frac{1}{2}\|\boldsymbol{W}^T\boldsymbol{\Phi}(\boldsymbol{Y}) - \boldsymbol{W}^T\boldsymbol{\Phi}(\boldsymbol{X})\boldsymbol{Z}\|_F^2 + \frac{\lambda_1}{2}\|\boldsymbol{Z}\|_{2,1} + \lambda_2 \sum_{i=1}^{c} \left\|\boldsymbol{W}^T\boldsymbol{\Phi}(\boldsymbol{X}^{(i)})\right\|_*$$

$$s.t. \quad \boldsymbol{W}^T\boldsymbol{W} = \boldsymbol{I}$$

(6-2)

In the framework, $\mathbf{\Phi}$ maps the sample data into a kernel feature space. The first term in the function is mapping the training set and testing sample into kernel space and using the kernel space training set to reconstruct the kernel space testing sample. $\boldsymbol{W}^T$ is a feature extraction matrix that reduces the feature spaces dimensions. The second term, $\frac{\lambda_1}{2}\|\boldsymbol{Z}\|_{2,1}$, is an $l_{2,1}$-norm regularization term. Since $l_{2,1}$-norm has the attribution of row sparsity, this term can make the elements in the reconstruction matrix $\boldsymbol{Z}$ to be sparse. Therefore, the testing sample is going to be reconstructed by the most representative training samples. The last term, $\lambda_2 \sum_{i=1}^{c}\left\|\boldsymbol{W}^T\mathbf{\Phi}(\boldsymbol{X}^{(i)})\right\|_*$, is a Low-Rank Constraint. It regularizes the samples from the same class to be closer, thus the classes are discriminant from each other.

6.1.2 Optimization

The choice of optimization technique impacts the accuracy of performance result, and how to optimize the framework is still under working processes. In this section, the transformation of the framework will be provided.

Since $\boldsymbol{W}$ can be seemed as a function of x,

$$\boldsymbol{W}_j = \sum_{i=1}^{n} a_{ji}\phi(\boldsymbol{x}_i) = \boldsymbol{\Phi}(\boldsymbol{X})\boldsymbol{a}_j$$

$$\boldsymbol{a}_j = [a_{j1}, a_{j2}, \dots, a_{jn}]^T$$

$$\boldsymbol{W} = \boldsymbol{\Phi}(\boldsymbol{X})\boldsymbol{A}$$

$$\boldsymbol{W}^T = \boldsymbol{A}^T\boldsymbol{\Phi}^T(\boldsymbol{X})$$

(6-3)

Substitute $\boldsymbol{W}$ to $\boldsymbol{\Phi}(\boldsymbol{X})\boldsymbol{A}$ and $\boldsymbol{W}^T$ to $\boldsymbol{A}^T\boldsymbol{\Phi}^T(\boldsymbol{X})$ we have:

$$\min_{\boldsymbol{W},\boldsymbol{Z}} \frac{1}{2}\|\boldsymbol{A}^T\boldsymbol{\Phi}^T(\boldsymbol{X})\boldsymbol{\Phi}(\boldsymbol{Y}) - \boldsymbol{A}^T\boldsymbol{\Phi}^T(\boldsymbol{X})\boldsymbol{\Phi}(\boldsymbol{X})\boldsymbol{Z}\|_F^2 + \frac{\lambda_1}{2}\|\boldsymbol{Z}\|_{2,1}$$

$$+ \lambda_2 \sum_{i=1}^{c} \left\|\boldsymbol{A}^T\boldsymbol{\Phi}^T(\boldsymbol{X})\boldsymbol{\Phi}(\boldsymbol{X}^{(i)})\right\|_*$$

$$s.t. \quad \boldsymbol{A}^T\boldsymbol{\Phi}^T(\boldsymbol{X})\boldsymbol{\Phi}(\boldsymbol{X})\boldsymbol{A} = \boldsymbol{I}$$

(6-4)

Given a Mercer kernel $\boldsymbol{K} : (\boldsymbol{X} \cdot \boldsymbol{Y}) \in \mathbb{R}$, which can be represented as: $\boldsymbol{K}_{(\boldsymbol{X},\boldsymbol{Y})} = \boldsymbol{\Phi}^T(\boldsymbol{X})\boldsymbol{\Phi}(\boldsymbol{Y})$, then function (6-4) can be rewritten as:

$$\min_{\boldsymbol{W},\boldsymbol{Z}} \frac{1}{2}\left\|\boldsymbol{A}^T\boldsymbol{K}_{(\boldsymbol{X},\boldsymbol{Y})} - \boldsymbol{A}^T\boldsymbol{K}_{(\boldsymbol{X},\boldsymbol{X})}\boldsymbol{Z}\right\|_F^2 + \frac{\lambda_1}{2}\|\boldsymbol{Z}\|_{2,1} + \lambda_2\sum_{i=1}^{c}\left\|\boldsymbol{A}^T\boldsymbol{K}_{(\boldsymbol{X},\boldsymbol{X}^{(i)})}\right\|_*$$

$$s.t. \quad \boldsymbol{A}^T\boldsymbol{K}_{(\boldsymbol{X},\boldsymbol{X})}\boldsymbol{A} = \boldsymbol{I}$$

(6-5)

Function (6-5) is the final form of KSRLC, and the optimization techniques are under processes. However, from the explanation of the framework, it is expected that the framework will outperform others.

6.2 Future Expectations

Aforementioned are frameworks have been proposed based on the SR and matrix theory. Besides the SR, there are still various techniques that can be applied in the field of image processing and machine learning. One of the hottest and most popular topics that have been attracted lots of attention in recent years is deep learning. Many kinds of literature have proved that deep learning has successfully obtained impressive performance in coping with image dataset. Especially the neural network that is building based on simple and easy understanding theoretical knowledge achieves superior performance.

During the study of neural networks, it has been found that although the basic theories are easy understanding, different choices of parameters can largely impact the study efficiency and results. In addition, it is a new idea to join the theory that has been applied in the abovementioned framework with neural networks to improve the performance of decision making. The future plan can be shown as:

1. Finish the work in chapter six first and extend the work to other frameworks. Since the most real-world dataset and applications are inherent with nonlinear relationships, and kernel trick is advanced in dealing with the nonlinear dataset. It is expected that kernel trick helps to better reflect data intrinsic structure, and then, a more accurate result.
2. Adding previously proposed theories in chapter 3-6 to the neural network domain. Inspired by (Kang, et al., 2019), the constraints work effectively in the networks and which is expected to further extend to other areas.

Chapter 7 Conclusion

Sparse Representation (SR) is a significant impact on the research of image recognition and computer vision analysis. Since SR is advance in analyzing data that is naturally sparse to fixed bases, SR becomes an extremely powerful tool for obtaining, representing and compressing the high-dimensional image datasets. Although the images are naturally lying in very high-dimensional space, the signification information is often belonging to the same class exhibit degenerate structure. Compared with the conventional techniques, SR using the whole training samples as a dictionary to represent the test sample, thus, the framework with SR has comparable more physical significance. On the other hand, LPP preserves the data neighborhoods relations that can represent the original data with high-fidelity, Residual Relationship Preserving is a novel concept that maintains the data reconstruction residual relationships, and Low-rank Constraint works effectively on learning the classes information. This thesis aims to focus on the applications of SR regarding subspace learning, pattern classification, and data structure analysis in different scenarios. The research innovations and outcomes are shown as followings.

1. Chapter three has proposed an unsupervised feature extraction framework that joint SR with LPP, JSRLPP. Since LPP shares the common drawback of separating the

graph structure learning and feature extraction in two steps, the feature extraction result is highly-depending on the graph learned previously. JSRLPP extends the LPP technique and combines it with SR to avoid the drawback of the pre-defined graph. In the framework, instead of employing the similarity matrix, a newly adaptively learned matrix is employed to ensure the data structure during data reconstruction. Namely, the matrix $\boldsymbol{S}$ in JSRLPP plays the role of the reconstruction matrix as well as the similarity matrix. It ensures framework reconstructs data structure with the attribute of data local relationships. In addition, with the SR, the framework is robustness to the effects of redundant and noise data. We carried out extensive experiments on four image datasets and compare JSRLPP with the state-of-the-art approaches. The results demonstrated that our framework achieved an impressive performance.

2. Chapter four has proposed an unsupervised feature selection framework that joint SR with DRR, UFSARP. It introduced a novel term, DRR, which represents the residual relationships between sample data. The residual relationship is an effective term to reflect the data relationships. It works not only in preserving the data local relationships, but also present effective in retain the data global structure. Adding DRR into the framework maintains the residual relationships during data reconstruction. We conducted comprehensive experiments, compared UFSARP with eleven state-of-the-art methods along with six datasets. The results demonstrated that UFSARP is superior than others.
3. Chapter five has proposed a supervised feature extraction framework that joint SR with Low-Rank Constraint, SRCLC. Since data from the same class should stay

closer with each other and share the same subset of features, it is expected that the within-class scatters to have a low-rank structure. In SRCLC, we have added two Low-Rank Constraints to preserve both within-class data structure and between-class data structure. Therefore, it effectively distinguishes the data classes' structure and greatly improves the accuracy of the classification result.

There are still many extensions can be done based on these researches in the near future:

1. The novel introduced DRR term can be applied to other frameworks or other areas as it is robustness to the different dataset and is powerful to maintain the data intrinsic structure.
2. The Low-Rank Constraint term that uses for representing the between-class structure can also be applied to other frameworks or other areas. It is a powerful term to distinguish the data class structure, which is effective in classification.
3. Chapter six has introduced the kernel trick to the dimensional reduction framework. As most image data are lying in non-linear dimensional spaces, and kernel trick is advanced in dealing the non-linear relationships, it is expected that kernel-trick can better preserve the original data structure.
4. Chapter seven has briefly introduced future study directions and expectations.

Bibliography

Abecasis, G. R., Cherny, S. S., Cookson, W. O., & Cardon, L. R. (2002). Merlinłrapid Analysis of Dense Genetic Maps Using Sparse Gene Flow Trees. *Nature genetic*, 97.

Amaldi, E., & Kann, V. (1998). On the Approximability of Minimizing Nonzero Variables or Unsatisfied Relations in Linear System. *Theoretical Computer Science*, 237-260.

Baker, S., & Nayar, S. K. (1996). Pattern Rejection. *Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (pp. 544-549).

Bartlett, M. (1998). Independent Component Representations for Face Recognition. *Proceeding of SPIE Symposium on Electronic Imaging.*

Belhumeur, P. N., Hespanha, J. P., & Kriegman, D. J. (1997). Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 711-720.

Belkin, M., & Niyogi, P. (2003). *Laplacian Eigenmaps for Dimensionaltiy Reduction and Data Representation.* MIT Press.

Bishop, C. M. (1995). *Neural Networks for Pattern Recognition.* Oxford university press.

Boyd, S., Parikh, N., Chu, E., Peleato, B., & Eckstein, J. (2010). Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Foundations and Trends in Machine Learning*, 1-122.

Burges, C. (1998). A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining Knowledge Discovery*, 121-167.

Cai, D., He, X., & Han, J. (2007). Semi-supervised Discriminant Analysis. *IEEE International Conference on Computer Vision*, (pp. 1-7).

Cai, D., Zhang, C., & He, X. (2010). Unsupervised Feature Selection for Multi-cluster Data. *In Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (pp. 333-342).

Cai, J., Cands, E., & Shen, Z. (2008). A Singular Value Thresholding Algorithm for Matrix Completion. *Siam Journal on Optimization*, 1956-1982.

Cai, Z., & Zhu, W. (2017). Multi-label Feature Selection via Feature Manifold Learning and Sparsity Regularization. *International Journal of Machine Learning and Cybernetics*, 1-14.

Chen, H.-T., Chang, H.-W., & Liu, T.-L. (2005). Local Discriminant Embedding and Its Variants. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR).*

Chen, S., Zhao, H., Kong, M., & Luo, B. (2007). 2D-LPP: A Two-dimensional Extension of Locality Preserving Projections. *Neurocomputing*, 912-921.

Cheng, X., Yang, J., Yan, S., Fu, Y., & Huang, T. S. (2010). Learning with L1-graph for Image Analysis. *IEEE Transactions on Image Processing*, 858-866.

Cheng, Y., Liu, K., Yang, J., Zhuang, Y., & Gu, N. (1991). Human Face Recognition Method Based on The Statistical Model of Small Sample Size. *Intelligent Robots and Computer Vision X: Algorithms and Techniques*, 85-95.

Cui, Y., Swets, D. L., & Weng, J. (1995). Learning-based Hand Sign Recognition Using SHOSLIF-M. *Proceedings of IEEE International Conference on Computer Vision*, (pp. 631-636).

Donoho, D. L. (2006). For Most Large Underdetermined Systems of Linear Equations the Minimal L1-norm Solution is Also the Sparsest Solution. *Communications on Pure and Applied Mathematics*, 907-934.

Du, L., & Shen, Y. (2015). Unsupervised Feature Selection with Adaptive Structure Learning. *In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (pp. 209-218).

Du, L., Zhou, P., Shi, L., Wang, H., Fan, M., Wang, W., & Shen, Y. (2015). Robust Multiple Kernel k-means Using L2,1-norm. *Proceedings of the 24th International Conference on Artificial Intelligence (AAAI Press)*, (pp. 3476-3482).

Fan, Z., Xu, Y., & Zhang, D. (2011). Local Linear Discriminant Analysis Framework Using Sample Neighbors. *IEEE Transaction on Neural Networks*, 1119-1132.

Fan, Z., Xu, Y., Zuo, W., Yang, J., Tang, J., Lai, Z., & Zhang, D. (2014). Modified Principal Component Analysis: An Integration of Multiple Similarity Subspace Models. *IEEE Transactions on Neural Networks and Learning Systems*, 1538-1552.

Fang, X., Xu, Y., Li, X., Fan, Z., Liu, H., & Chen, Y. (2014). Locality and Similarity Preserving Embedding for Feature Selection. *Neurocomputing*, 304-315.

Fang, X., Xu, Y., Li, X., Lai, Z., Teng, S., & Fei, L. (2017). Orthogonal Self-guided Similarity Preserving Proection for Classification and Clustering. *Neural Networks*, 1-8.

Feng, G., Hu, D., Zhang, D., & Zhou, Z. (2006). An Alternative Formulation of Kernel LPP with Application to Image Recognition. *Neurocomputing*, 1733-1738.

Fisher, R. A. (1936). The Effect of A Few Toxic Substances Upon The Total Blood Cell Count in The Cockroach, Blatta Orientalis Linn,. *Annals of The Entomological Society of America*, 335-340.

Ghassabeh, Y. A., Rudzicz, F., & Moghaddam, H. A. (2015). Fast Incremental LDA Feature Extraction. *Pattern Recognition*, 1999-2012.

Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). Gene Selection for Cancer Classification Using Support Vector Machines. *Machine Learning*, 389-422.

He, X., & Niyogi, P. (2003). Locality Preserving Projections. *Neural Information Processing Systems Conference.*

He, X., Cai, D., & Niyogi, P. (2005). Laplacian Score for Featrue Selection. *International Conference on Neural Information Processing System* (pp. 507-514). MIT Press.

He, X., Cai, D., Yan, S., & Zhang, H. (2005). Neighborhood Preserving Embedding. *IEEE International Conference on Computer Vision*, (pp. 1208-1213).

He, X., Ji, M., Zhang, C., & Bao, H. (2011). A Variance Minimization Criterion to Feature Selection Using Laplacian Regularization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013-2025.

He, X., Yan, S., Hu, Y., Niyogi, P., & Zhang, H. (2005). Face Recognition Using Laplacianfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 328-340.

Hou, C., Nie, F., Li, X., Yi, D., & Wu, Y. (2011). Feature Selection via Joint Embedding Learning and Sparse Regression. *International Joint Conference on IJCAI*, (pp. 1324-1329).

Hou, C., Nie, F., Li, X., Yi, D., & Wu, Y. (2017). Joint Embedding Learning and Sparse Regression: A Framework for Unsupervised Feature Seletion. *IEEE Transactions on Cybemetics*, 793-804.

Huang, S. H. (2015). Supervised Feature Selection: A Tutorial. *Artificial Intelligence Research*.

Kang, P., Lin, Z., Yang, X., Fang, X., Li, Q., & Liu, W. (2019). Deep Semantic Space with Intra-class Low-rank Constraint for Cross-modal Retriebal. *ACM International Conference on Multimedia Retrieval (ICMR).*

Kong, X., & Yu, P. S. (2010). Semi-supervised Feature Selection for Graph Classification. *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (pp. 793-802).

Kreutz-Delgado, K., & Murray, J. F. (2007). Visual Recognition and Inference Using Dynamic Overcomplete Sparse Learning. *Neural Computation*, 2301-2352.

Krzanowski, W. J. (1987). Selection of Variables to Preserve Multivariate Data Structure, Using Principal Components. *Journal of the Royal Statistical Society*, 22-23.

Lai, Z., Xu, Y., Yang, J., Tang, J., & Zhang, D. (2013). Sparse Tensor Discriminant Analysis. *IEEE Transactions on Image Processing*, 3904-3915.

Li, H., Jiang, T., & Zhang, K. (2006). Efficient and Robust Feature Extraction by Maximum Margin Criterion. *IEEE Transactions on Neural Networks*, 157-165.

Li, X., Zhang, H., Zhang, R., Liu, Y., & Nie, F. (2018). Generalized Uncorrelated Regression with Adaptive Graph for Unsupervised Feature Selection. *IEEE Transactions on Neural Networks and Learning Systems*.

Li, Z., Yang, Y., Liu, J., Zhou, X., & Lu, H. (2012). Unsupervised Feature Selection Using Nonnegative Spectral Analysis. *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, (pp. 1026-1032).

Lin, Z., Liu, R., & Su, Z. (2011). Linearized Alternating Direction Method with Adaptive Penalty for Low-rank Representation. *Advances in Neural Information Processing System*, (pp. 612-620).

Liu, G., & Yan, S. (2011). Latent Low-rank Representation for Subspace Segmentation and Feature Extraction. *IEEE International Conference on Computer Vision*, (pp. 1615-1622).

Liu, G., Lin, Z., Yan, S., Sun, J., Yu, Y., & Ma, Y. (2010). Robust Recovery of Subspace Structures by Low-rank Representation. *CoRR*.

Liu, G., Lin, Z., Yan, S., Sun, J., Yu, Y., & Ma, Y. (2013). Robust Recovery of Subspace Structures by Low-rank Representation. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 171-184.

Liu, H., Wu, X., & Zhang, S. (2014). A New Superviased Feature Selection Method for Pattern Classification. *Computational Intelligence*, 342-361.

Liu, J., Ji, S., & Ye, J. (2009). Multi-task Feature Learning via Efficient L2,1-norm Minimization. *In Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence*, (pp. 339-348).

Liu, X., Wang, L., Zhang, J., Yin, J., & Liu, H. (2014). Global and Local Structure Preservation for Feature Selection. *IEEE Transactions on NNLS*, 1083-1095.

Liu, Z., Yin, J., & Jin, Z. (2010). Locality Preserving Projections Based on L1 Graph. *Pattern Recognition*, 1-4.

Lu, Y., Lai, Z., Xu, Y., Li, X., Zhang, D., & Yuan C. (2016). Low-rank Preserving Projections. *IEEE Transactions on Cybernetics*, 1900-1913.

Ma, L., Li, M., Gao, Y., Chen, T., Ma, X., & Qu, L. (2017). A Novel Wrapper Approach for Feature Selection in Object-based Image Classification Using Ploygon-based Cross-validation. *IEEE Geoscience and Remote Sensing Letters*, 409-413.

Majumdar, A., & Ward, R. K. (2009). Classification via Group Sparsity Promoting Regularization. *IEEE International Conference on Acoustics, Speech and Signal Processing.*

Maldonado, S., & Weber, R. (2009). A Wrapper Method for Feature Selection Using Support Vector Machines. *Information Sciences*, 2208-2217.

Martlnez, A. M., & Kak, A. C. (2001). PCA versus LDA. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 228-233.

Mika, S., R¨atsch, G., Weston, J., Schölkopf, B., & Müller, K. R. (1999). Fisher Discriminant Analysis with Kernels. *Neural Networks for Signal Processing IX: Proceedings of the 1999 IEEE Signal Processing Society Workshop.*

Mitra, P., Murthy, C. A., & Pal, S. K. (2002). Unsupervised Feature Selection Using Feature Similarity. *Pattern Analysis and Machine Intelligence IEEE Tractions*, 301-312.

Moses, Y., Adini, Y., & Ullman, S. (1994). Face Recognition: The Problem of Compensating for Changes in Illumination Direction. *European Conf. Computer Vision*, (pp. 286-296).

Muller, K., Mika, S., Ratsch, G., Tsuda, K., & Scholkopf, B. (2001). An Introduction to Kernel-based Learning Algorithms. *IEEE Transactions on Neural Networks*, 181-201.

Nie, F., Huang, H., Cai, X., & Ding, C. H. (2010). Efficient and Robust Feature Selection via Joint L2,1-norms Minimization. *In Advances in Neural Information Processing Systems*, (pp. 1813-1821).

Nie, F., Wang, X., & Huang, H. (2014). Clustering and Projected Clustering with Adaptive Neighbors. *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (pp. 977-986).

Nie, F., Xu, D., Tsang, I., & Zhang, C. (2010). Flexible Manifold Embedding: A Framework for Semi-supervised and Unsupervised Dimension Reduction. *IEEE Transactions on Image Processing*, 1921-1923.

Nie, F., Zhu, W., & Li, X. (2017). Unsupervised Large Graph Embedding. *Proceeding of the Thirty-First AAAI Conference on Aritficial Intelligence*, (pp. 2422-2428).

Peng, M., Xie, Q., Wang, H., Zhang, Y., & Tian, G. (2018). Bayesian Sparse Topical Coding. *IEEE Transactions on Knowledge and Data Engineering*.

Qian, M., & Zhai, C. (2013). Robust Unsupervised Feature Selection. *IJCAI*, (pp. 1621-1627).

Qiao, L., Chen, S., & Tan, X. (2010). Sparsity Preserving Projections with Applications to Face Recognition. *Pattern Recognition*, 331-341.

Ren, J., Qiu, Z., Fan, W., Cheng, H., & Yu, P. S. (2008). Forward Semi-supervised Feature Selection. *Pacific-asia Conference PAKDD*, (pp. 970-976). Osaka.

Roweis, S. T., & Saul, L. K. (2000). Nonlinear Dimensionality Reduction by Locally Linear Embedding. *SCIENCE*, 2323-2326.

Schölkopf, B., Smola, A., & Müller, K. R. (1998). Nonlinear Component Analysis as a Kernel Eigenvalue Problem. *Neural Computing*, 1299-1319.

Shalkoff, R. L. (1989). *Digital Image Processing and Computer Vision.* New York: Wiley.

Shang, R., Chang, J., Jiao, L., & Xue, Y. (2017). Unsupervised Feature Selection Based on Self-Representation Sparse Regression and Local Similarity Preserving. *International Journal of Machine Learning and Cybernetics*, 1-14.

Sirovich, I., & Kirby, M. (1987). Low-dimensional Procedure for the Caracterization of Human Faces. *Journal of Optical Society of America*, 519-524.

Skoˇcaj, D., Leonardis, A., & Bischof, H. (2007). Weighted and Robust Learning of Subspace Representations. *Pattern Recognition*, 1556-1569.

Smola, A. J., & Schölkopf, B. (2004). A Tutorial on Support Vector Regression. *Statistics and Computing*, 199-222.

Swets, D. L., & Weng, J. (1996). Using Discriminant Eigenfeatures for Image Retrieval. *IEEE Transactions on pattern analysis and machine intelligence*, 831-836.

Tabakhi, S., Moradi, P., & Akhlaghian, F. (2014). An Unsupervised Feature Selection Algorithm Based on Ant Colony Optimization. *Engineering Applications of Artificial Intelligence*, 112-123.

Tang, K., Liu, R., Su, Z., & Zhang, J. (2014). Structure-constrained Low-rank Representation. *IEEE Transactions on Neural Networks and Learning System*, 2167-2179.

Tenenbaum, J. B., Silva, V. D., & Langford, J. C. (2000). A Global Geometric Framework for Nonlinear Dimensionality Reduction. *SCIENCE*, 2319-2323.

Turk, M., & Pentland, A. (1991). Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, 71-86.

Vapnik, V. N. (1998). *Statistical Learning Theory*. New York: Wiley.

Wang, S., Tang, J., & Liu, H. (2015). Embedded Unsupervised Feature Selection. *AAAI*, (pp. 470-476).

Wang, X., Chen, R., Hong, C., & Zeng, Z. (2018). Unsupervised Feature Analysis with Sparse Adaptive Learning. *Pattern Recognition Letters*, 89-94.

Wen, J., Fang, X., Cui, J., Fei, L., Yan, K., Chen, Y., & Xu, Y. (2018). Robust Sparse Linear Discriminant Analysis. *IEEE Transactions on Circuits and Systems for Video Technology*.

Wen, Z., & Yin, W. (2012). A Feasible Method for Optimization with Orthogonality Constraints. *Mathematical Programming*, 397-434.

Weston, J., Elisseeff, A., Scholkopf, B., & Tipping, M. (2003). Use of the Zero-norm with Linear Models and Kernel Methods. *Journal of Machine Learning Research*, 1439-1461.

Wright, J., Ganesh, A., Rao, S., & Ma, Y. (2009). Robust Principal Component Analysis: Exact Recovery of Corrupted Low-rank Matrices via Convex Optimization. *Advances in Neural Information Processing Systems*, (pp. 2080-2088).

Wright, J., Yang, A. Y., Ganesh, A., Sastry, S. S., & Ma, Y. (2009). Robust Face Recognition via Sparse Representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 210-227.

Xiong, H., Swamy, M. N., & Ahmad, M. O. (2005). Two-dimensional FLD for Face Recognition. *Pattern Recognition*, 1121-1124.

Xu, Y., Fang, X., Wu, J., Li, X., & Zhang, D. (2016). Discriminative Transfer Subspace Learning via Low-rank and Sparse Representation. *IEEE Transacitions on Image Processing*, 850-963.

Xu, Y., Zhong, A., Yang, J., & Zhang, D. (2010). LPP Solution Schemes for Use with Face Recognition. *Pattern Recognition*, 4165-4176.

Xu, Y., Zhu, Q., Fan, Z., Wang, Y., & Pan, J. (2013). From the Idea of "Sparse Representation" to a Representation-based Transformation Method for Feature Extraction. *Neurocomputing*, 168-176.

Yan, S., Xu, D., Zhang, B., Zhang, H. J., Yang, Q., & Lin, S. (2007). Graph Embedding and Extensions: A General Framework for Dimensionality Reduction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40-51.

Yang, H., Kong, X., Fu, H., Li, M., & Zhao, G. (2015). Semi-supervised Learning Based on Group Sparse for Relative Attributes. *IEEE International Conference on Image Processing (ICIP).*

Yang, J., Frangi, A. F., Yang, J., Zhang, D., & Jin, Z. (2005). KPCA Plus LDA: A Complete Kernel Fisher Discriminant Framework for Feature Extraction and Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 230-244.

Yang, J., Gao, X., Zhang, D., & Yang, J. (2005). Kernel ICA: An Alternative Formulation and Its Application to Face Recognition. *Pattern Recognition*, 1784-1787.

Yang, J., Zhang, D., Frangi, A. F., & Yang, J. (2004). Two-dimensional PCA: A New Approach to Face Representation and Recognition. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 131-137.

Yang, J., Zhang, D., Yang, J.-y., & Niu, B. (2007). Globally Maximizing, Locally Minimizing: Unsupervised Discriminant Projection with Applications to Face and

Palm Biometrics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 650-664.

Yang, L., Gong, W., Gu, X., Li, W., & Liang, Y. (2008). Null Space Discriminant Locality Preserving Projections for Face Recognition. *Neurocomputing*, 3644-3649.

Yang, M. (2002). Kernel Eigenfaces vs. Kernel Fisherfaces: Face Recognition Using Kernel Methods. *IEEE International Conference on Automatic Face and Gesture Recognition*, (pp. 215-220).

Yang, W., & Dai, D. (2009). Two-dimensional Maximum Margin Feature Extraction for Face Recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 1002-1012.

Yang, Y., Shen, H. T., Ma, Z., Huang, Z., & Zhou, X. (2011). L2,1-norm Regularized Discriminative Feature Selection for Unsupervised Learning. *In IJCAI Proceedings-international Joint Conference on Artificial Intelligence*, (p. 1589).

Yu, K., Ji, L., & Zhang, X. (2002). Kernel Nearest-Neighbor Algorithm. *Neural Processing Letters*, 147-156.

Yu, S., Tranchevent, L., Liu, X., Glanzel, W., Suykens, J. A., De Moor, B., & Moreau, Y. (2012). Optimized Data Fusion for Kernel k-means Clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1031-1039.

Yuan, M., & Lin, Y. (2006). Model Selection and Estimation in Regression with Grouped Variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 49-67.

Zeng, H., & Cheung, Y. M. (2011). Feature Selection and Kernel Learning for Local Learning-based Clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1532-1547.

Zhang, L., Yang, M., & Feng, X. C. (2011). Sparse Representation or Collaborative Representation: Which Helps Face Recognition? *IEEE International Conference on Computer Vision*, (pp. 471-478).

Zhang, W., Kang, P., Fang, X., Teng, L., & Han, N. (2018). Joint Sparse Representation and Locality Preserving Projection for Feature Extraction. *International Journal of Machine Learning and Cybernetics*, 1-15.

Zhang, Y., Jiang, Z., & Davis, L. S. (2013). Learning Structured Low-rank Representations for Image Classification. *IEEE Conference on Computer Vision and Pattern Recognition*, (pp. 676-683).

Zhao, Z., & Wang, L. (2007). Spectral Feature Selection for Superviased and Unsuperviased Learning. *In Proceedings of the 24th International Conference on Machine Learning*, (pp. 1151-1157).

Zhao, Z., Wang, L., Liu, H., & Ye, J. (2013). On Similarity Preserving Feature Selection. *IEEE Transaction on Knowledge and Data Engineering*, 619-632.

Zheng, W., Lin, Z., & Wang, H. (2014). L1-norm Kernel Discriminant Analysis via Bayes Error Bound Optimization for Robust Feature Extraction. *IEEE Transactions on Neural Networks and Learning Systems*, 793.

Zhu, X., Wu, X., Ding, W., & Zhang, S. (2013). Feature Selection by Joint Graph Sparse Coding. *In Proceedings of the 2013 SIAM International Conference on Data Mining*, (pp. 803-811).

Zhuang, L., Gao, H., Lin, Z., Ma, Y., Xhang, X., & Yu, N. (2012). Non-negative Low Rank and Sparse Graph for Semi-supervised Learning. *IEEE International Conference on Computer Vision and Pattern Recognition*, (pp. 2318-2335).

Zou, H., Hastie, T., & Tibshirani, R. (2006). Sparse Principal Component Analysis. *Journal of Computational and Graphical Statistics*, 265-286.