6-1-2001

IMPROVED STATISTICS ESTIMATION AND FEATURE EXTRACTION FOR HYPERSPECTRAL DATA CLASSIFICATION

Bor-Chen Kuo
Purdue University School of ECE

David Landgrebe
Purdue University School of ECE

IMPROVED STATISTICAL ESTIMATION AND FEATURE EXTRACTION FOR HYPERSPECTRAL DATA CLASSIFICATION

BOR-CHEN KUO
DAVID LANDGREBE

TR-ECE 01-6
NOVEMBER 2001

SCHOOL OF ELECTRICAL
AND COMPUTER ENGINEERING
PURDUE UNIVERSITY
WEST LAFAYETTE, INDIANA 47907-1285

IMPROVED STATISTICS ESTIMATION AND FEATURE EXTRACTION FOR HYPERSPECTRAL DATA CLASSIFICATION

Bor-Chen Kuo and
David Landgrebe

December 2001

School of Electrical & Computer Engineering
Electrical Engineering Building
Purdue University
West Lafayette, Indiana 47907-1285

This work was sponsored in part by
the U.S. Army Research Office under
Grant Number DAAH04-96-1-0444 and
The National Imagery and Mapping Agency
Under Contract NMA201-01-C-0023

TABLE OF CONTENTS

Page

TABLE OF CONTENTS.....iii

ABSTRACT.....vii

CHAPTER 1: INTRODUCTION.....1
1.1 Statement of the Problem.....1
1.2 Organization of This Report.....4

CHAPTER 2: MIXED LEAVE-ONE-OUT COVARIANCE ESTIMATOR.....7
2.1 Introduction.....7
2.2 Background and Previous Works.....7
2.2.1 Regularized Discriminant Analysis (RDA; [2]).....8
2.2.2 Leave-One-Out Covariance Estimator (LOOC; [4],[5]).....9
2.2.3 Bayesian Leave-One-Out Covariance Estimator (BLOOC; [6]).....10
2.2.4 The Comparison of Performances of RDA. LOOC. and BLOOC.....10
2.3 Mixed Leave-One-Out Covariance (Mixed-LOOC) Estimators.....11
2.3.1 Mixed-LOOC1.....11
2.3.2 Mixed-LOOC2.....12
2.4 Experiment Design for Comparing LOOC. Mixed-LOOC1, and Mixed-LOOC2.....15
2.5 Experiment Results.....16

CHAPTER 3: Discriminate Analysis Feature Extraction Based on Mixed-LOOC2.....19
3.1 Discriminate Analysis Feature Extraction (DAFE).....19
3.2 Comparison of DAFE and DAFE Based on Mixed-LOOC2.....20

3.3 Concluding Comments....22

CHAPTER 4: GAUSSIAN MIXTURE CLASSIFIER BASED ON MIXED-LOOC2....23

4.1 Introduction....23

4.2 Parameter Estimation Methods....23

4.2.1 Normal Mixture Density....23

4.1.2 Nearest Means Clustering....24

4.1.3 EM clustering....24

4.3 Model Selection Indices....25

4.3.1 Akaike information criterion (AIC)....26

4.3.2 Bayesian information criterion (BIC)....26

4.3.3 Normalized Entropy Criterion and Classification Likelihood Criterion....27

4.3.4 Integrated Classification Likelihood Criterion....28

4.4 Gaussian Mixture Classifier Based on Mixed-LOOC....29

4.4.1 Mixture Classifier Using Mixed-LOOC and Nearest Means Clustering....29

4.4.2 Mixture Classifier Using Mixed-LOOC and EM clustering....29

4.5 Simulated and Real Data Experiments....30

4.5.1 Simulation Data Experiment Design....30

4.5.1 Real Data Experiment Design....32

4.6 Experiment Results....32

4.6.1 Simulation Experiment Results....32

4.6.2 Real Data Experiment Results....33

4.7 Concluding Comments....33

CHAPTER 5: Nonparametric Weighted Feature Extraction....41

5.1 Introduction....41

5.2 Previous Works....42

5.2.1 Discriminant Analysis Feature Extraction (DAFE)....42

5.2.2 aPAC Linear Dimension Reduction (aPAC-LDR)....43

5.2.3 Decision Boundary Feature Extraction (DBFE)....44

5.2.4 Nonparametric Discriminant Analysis (NDA; [8],[23])....45

5.3 Nonparametric Weighted Feature Extraction (NWFE)....48

5.4 Simulated and Real Data Experiments....49

5.4.1 Simulation Data Experiment Design....49
5.4.2 Real Data Experiment Design....50
5.5 Experiment Results....51
5.5.1 Simulation Experiment Results....51
5.5.2 Real Data Experiment Results....58
5.6 Concluding Comments....64

CHAPTER 6: Using Mixture Classifier Based on Mix-LOOC2 after Feature Extraction....67
6.1 Introduction....67
6.2 Experiment Design....67
6.2 Experiment Results....68
6.3 Concluding Comments....74

CHAPTER 7: CONCLUSIONS....75
7.1 Summary....75
7.2 Suggestions for Further Work....76

APPENDIX A: THE MAXIMUM LIKELIHOOD ESTIMATOR OF MIXTURE PARAMETER IN LOOC AND BLOOC....77

APPENDIX B: THE INFORMATION ABOUT SIMULATION DATA SETS AND REAL DATA SETS....81
B.1 Experiment Design of Simulation Studies....81
B.1.1 The Mean Vector and Covariance Matrix....81
B.2 Dimensionality and Sample Size of Real Data Sets....83
B.2.1 Cuprite. Nevada scene data....83
B.2.2 Jasper Ridge Data....83
B.2.3 Indian Pine Data....84
B.2.4 DC Mall Data....84

REFERENCES....85

ABSTRACT

For hyperspectral data classification, the avoidance of singularity of covariance estimates or excessive near singularity estimation error due to limited training data is a key problem. This study is intended to solve problem via regularized covariance estimators and feature extraction algorithms. **A** second purpose is to build a robust classification procedure with the advantages of the algorithms proposed in this study but robust in the sense of not requiring extensive analyst operator skill.

A pair of covariance estimators called Mixed-LOOCs is proposed for avoiding excessive covariance estimator error. Mixed-LOOC2 has advantages over LOOC and BLOOC and needs less computation than those two. Based on Mixed-LOOC2, new DAFE and mixture classifier algorithms are proposed.

Current feature extraction algorithms, while effective in some circumstances, have significant limitations. Discriminate analysis feature extraction (DAFE) is fast but does not perform well with classes whose mean values are similar, and it produces only N-1 reliable features where N is the number of classes. Decision Boundary Feature Extraction does not have these limitations but does not perform well when training sets are small, A new nonparametric feature extraction method (NWFE) is developed to solve the problems of DAFE and DBFE. NWFE takes advantage of the desirable characteristics of DAFE and DBFE, while avoiding their shortcomings.

Finally, experimental results show that using NWFE features applied to a mixture classifier based on the Mixed-LOOC2 covariance estimator has the best performance and is a robust procedure for classifying hyperspectral data.

CHAPTER 1: INTRODUCTION

1.1 Statement of the Problem

As new sensor technology has emerged over the past few years, high dimensional multispectral data with hundreds of bands have become available. For example, the AVIRIS system[i] gathers image data in 210 spectral bands in the 0.4-2.4 μm range. Compared to the previous data of lower dimensionality (less than 20 bands), this hyperspectral data potentially provides a wealth of information. However, it also raises the need for more specific attention to the data analysis procedure if this potential is to be fully realized.

Among the ways to approach hyperspectral data analysis, a useful processing model that has evolved in the last several years [1] is shown schematically in Figure 1.1. Given the availability of data (box 1), the process begins by the analyst specifying what classes are desired, usually by labeling training samples for each class (box 2). New elements that have proven important in the case of high dimensional data are those indicated by boxes in the diagram marked 3 and 4. These are the focus of this work and will be discussed in more detail shortly, however the reason for their importance in this context is as follows. Classification techniques in pattern recognition typically assume that there are enough training samples available to obtain reasonably accurate class descriptions in quantitative form. Unfortunately, the number of training samples required to train a classifier for high dimensional data is much greater than that required for conventional data, and gathering these training samples can be difficult and expensive. Therefore, the assumption that enough training samples are available to accurately estimate the class quantitative description is frequently not satisfied for high dimensional data. There are

---

i Airborne Visible and Infrared Imaging Spectrometer system, built and operated by the NASA Jet Propulsion Center.

many types of classification algorithms used on such data. Perhaps the most common is the quadratic maximum likelihood algorithm.

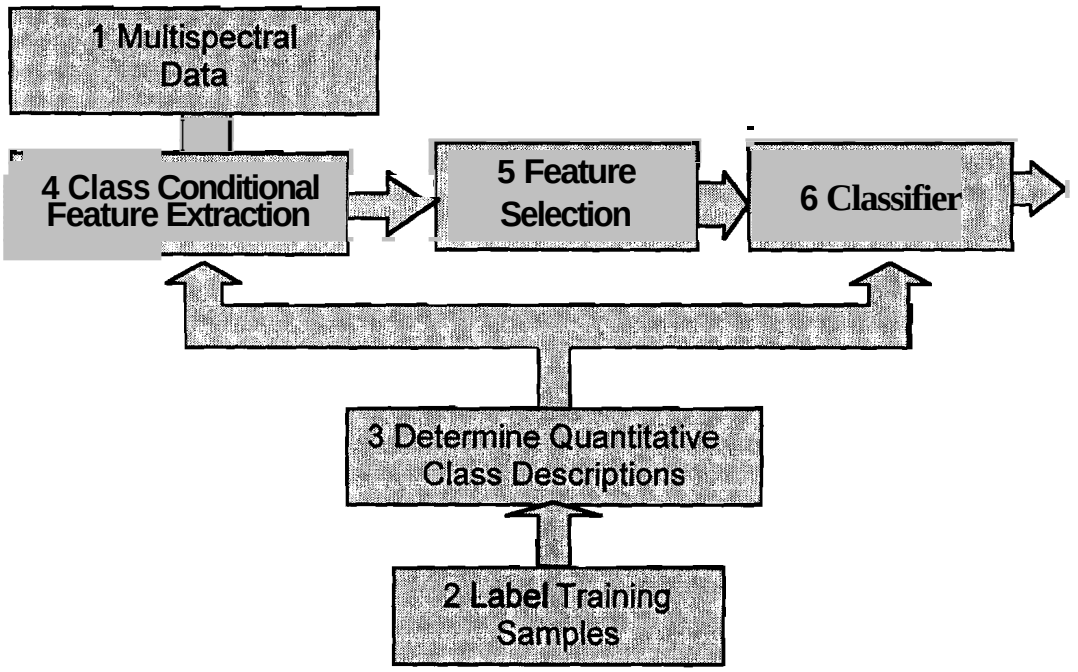
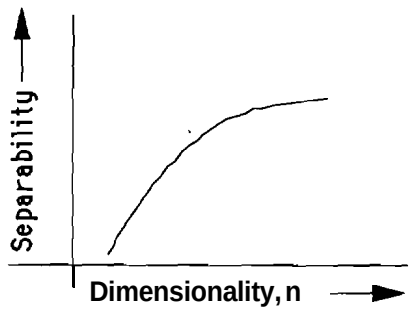


Figure 1.1 A schematic diagram for a hyperspectral data analysis procedure.

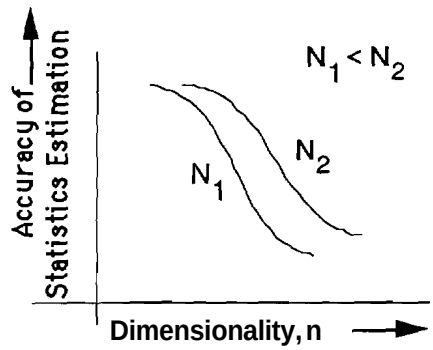
In the stochastic approach, the characteristics of a class are modeled with a set of parameters, which are estimated based on some prior knowledge, often in the form of pre-labeled samples. The pre-labeled samples used to estimate class parameters and design a classifier are called training samples. The accuracy of parameter estimation depends substantially on the ratio of the number of training samples to the dimensionality of the feature space. As the dimensionality increases, the number of training samples needed to characterize the classes increase as well. If the number of training samples available fails to catch up with the need, which is the case for hyperspectral data, parameter estimation becomes inaccurate.

Consider the case of a finite and fixed number of training samples. The accuracy of statistics estimation decreases as dimensionality increases, leading to a decline of the classification accuracy (Figure 1.2(b)). Although increasing the number of spectral bands (dimensionality) potentially provides more information about class separability

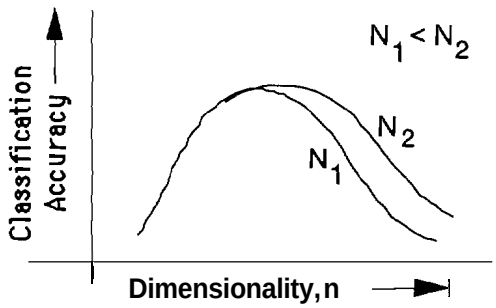
(Figure 1.2(a)), this positive effect is diluted by poor parameter estimation. As a result, the classification accuracy first grows and then declines as the number of spectral bands increases (Figure 1.2(c)), which is often referred to as the Hughes phenomenon (or the peaking phenomenon).



(a) High dimensionality (the number of spectral bands) potentially provides better class separability.



(b) With a finite and fixed number of samples, the accuracy of statistics estimation decreases as dimensionality increases. As the number of training samples, denoted by N, increases, statistics estimation improves.



(c) The peaking phenomenon results from the combination of the two opposite effects shown in (a) and (b).

Figure 1.2 Conceptual presentation of classification accuracy vs. measurement complexity in finite and fixed training cases (The Hughes phenomenon).

There are several ways to overcome this difficulty. In [2], these techniques are categorized into three groups:

a. Dimensionality reduction by feature extraction or feature selection,

b. Regularization of sample covariance matrix (e.g. [3], [4]), and

c. Structurization of a true covariance matrix described by a small number of parameters [2].

The objectives of this research are

1. To improve the major steps of hyperspectral data classification (box 3, 4 and 6 of the Figure 1.1).

2. To find a robust and easy classification process for users.

1.2 Organization of This Report

Chapter 2: Two regularized covariance estimators with the advantages of LOOC [5], [6] and BLOOC [7] are developed. The results of several experiments with computer generated data and AVIRIS data sets are presented that test their performances.

Chapter 3: Discriminate analysis feature extraction (DAFE) is improved in this chapter by using one of the regularized covariance estimators developed in Chapter 2. The improved DAFE relieves one of the limitations of DAFE that total training sample size should be greater than the dimensionality. Different combinations of feature extraction methods and classifiers are tested by using AVIRIS data sets.

Chapter 4: Gaussian mixture classifiers with different parameter estimation and model selection methods are improved in this chapter by using one of the regularized covariance estimators developed in Chapter 2. The results of several experiments with computer generated data and AVIRIS data sets are presented that test their performances.

Chapter 5: **A** nonparametric feature extraction method is developed to solve those problems in **DAFE.** The results of several experiments with computer generated data and **AVIRIS** data sets are presented that test its performance.

Chapter 6: The performances of combining feature extraction **(DAFE** and NWFE) and a mixture classifier based on Mixed-LOOC2 procedures are tested in this Chapter. The results of several experiments with computer generated data sets, **AVIRIS** data sets, and HyMap data sets are presented that test its performance.

Chapter 7: General conclusions and potentials for future research development future research are suggested in this chapter.

CHAPTER 2: MIXED LEAVE-ONE-OUT COVARIANCE ESTIMATOR

2.1 Introduction

For a quadratic classifier, the mean vector and covariance matrix of each class are the parameters that must be estimated from training samples. Usually the ML estimator is used. When the dimensionality of data exceeds the number of training samples, the ML covariance estimate is singular and cannot be used, however even in cases where the number of training samples is only two or three times the number of dimensions, estimation error can be a significant problem.

The purpose of this chapter is to define an improved regularized covariance estimator of each class that is invertible and with the advantages of LOOC [5], [6] and BLOOC [7] (box *3* of Figure *1.1).*

2.2 Background and Previous Works

The decision rule in a quadratic classifier is to label the (p by *1)* vector x as class k if the likelihood of class k is the greatest among the classes:

$$x \in \text{class } k, \text{ if } \arg\max_i [f(m_i, \Sigma_i \mid x)] = k$$

$$f(x \mid m_i, \Sigma_i) = \frac{1}{\sqrt{(2\pi)^p |\Sigma_i|}} \exp\left(\frac{-1}{2} (x - m_i)^T \Sigma_i^{-1} (x - m_i) \right)$$

where m_i is the mean vector, and Σ_i is the covariance matrix. Usually in practice the true values of the mean and covariance are not known and must be estimated from training

samples. The mean is typically estimated by the sample mean $m_i = \frac{1}{N_i}\sum_{j=1}^{N_i} x_{i,j}$, where $x_{i,j}$ is sample j from class i. The covariance matrix is typically estimated by the maximum likelihood covariance estimate $S_i = \frac{1}{N_i}\sum_{j=1}^{N_i}(x_{i,j} - m)(x_{i,j} - m_i)^T$.

The maximum likelihood mean and covariance estimates have the property that they maximize the joint likelihood of the training samples, which are assumed to be statistically independent.

$$m_i = \arg\max_m \prod_{j=1}^{N_i} f(x_{i,j} \mid m, \Sigma_i) \quad \text{and} \quad S_i = \arg\max \prod_{j=1}^{N_i} f(x_{i,j} \mid m_i, \Sigma).$$

2.2.1 Regularized Discriminant Analysis (RDA; [2])

Regularized discriminant analysis (RDA) is a two-dimensional optimization over four-way mixtures of the sample covariance, common covariance, the identity matrix times the average diagonal element of the common covariance, and the identity matrix times the average diagonal element of the sample covariance.

$$\hat{\Sigma}_i(\lambda,\gamma) = (1-\gamma)\hat{\Sigma}_i(\lambda) + \gamma\left(\frac{tr(\hat{\Sigma}_i(\lambda))}{p}\right)I \qquad 0 \le \gamma \le 1$$

where

$$\hat{\Sigma}_i(\lambda) = \frac{(1-\lambda)(N_i - 1)S_i + \lambda(N-L)S_w}{(1-\lambda)N_i + \lambda N}, \quad 0 \le \lambda \le 1$$

$$S_w = \frac{1}{N_i - L}\sum_{i=1}^{L}(N_i - 1)S_i$$

The criterion function that is maximized is the leave-one-out classification error. Since the criterion function depends on the covariance estimates of the other classes, the same values of the mixing parameters are used for all classes.

2.2.2 Leave-One-Out Covariance Estimator (LOOC; [4],[5])

$$\hat{\Sigma}_i(\alpha_i) = \begin{cases} (1-\alpha_i)diag(S_i) + \alpha_i S_i & 0 \le \alpha_i \le 1 \\ (2-\alpha_i)S_i + (\alpha_i - 1)S & 1 \le \alpha_i \le 2 \\ (3-\alpha_i)S + (\alpha_i - 2)diag(S) & 2 \le \alpha_i \le 3 \end{cases}$$

$$\text{where} \quad S = \frac{1}{L}\sum_{i=1}^{L} S_i$$

The mean of class i, without sample k, is $m_{i/k} = \frac{1}{N_i - 1}\sum_{\substack{j=1 \\ j+k}}^{N_i} x_{i,j}$, where the notation /k indicates the quantity is computed without sample k. The sample covariance of class i, without sample k, is

$$\Sigma_{i/k} = \frac{1}{N_i - 2}\sum_{\substack{j=1 \\ j \neq k}}^{N_i} (x_{i,j} - m_{i/k})(x_{i,j} - m_{i/k})^T$$

and the common covariance, without sample k from class i, is

$$S_{i/k} = \left(\frac{1}{L}\sum_{\substack{j=1 \\ j \neq i}}^{L} \Sigma_j \right) + \frac{1}{L}\Sigma_{i/k} \ .$$

The proposed estimate for class i, without sample k, can then be computed as follows:

$$C_{i/k}(\alpha_i) = \begin{cases} (1-\alpha_i)diag(\hat{\Sigma}_{i/k}) + \alpha_i \hat{\Sigma}_{i/k} & 0 \le \alpha_i \le 1 \\ (2-\alpha_i)\hat{\Sigma}_{i/k} + (\alpha_i - 1)S_{i/k} & 1 < \alpha_i \le 2 \\ (3-\alpha_i)S_{i/k} + (\alpha_i - 2)diag(S_{i/k}) & 2 < \alpha_i \le 3 \end{cases}$$

The mixing parameter a, is determined by maximizing the average leave-one-out log likelihood of each class:

$$LOOL_i = \frac{1}{N_i}\sum_{k=1}^{N_i} \ln[f(x_k \mid m_{i/k}, C_{i/k}(\alpha_i)]$$

2.2.3 Bayesian Leave-One-Out Covariance Estimator (BLOOC; [6])

LOOC was found to work well for well trained classifiers, however, it was sensitive to outliers. In practice this frequently occurs in cases where the class list is not exhaustive, such that the missing classes constitute outliers to the defined classes. Thus the following scheme was devised.

$$\hat{\Sigma}_i(\alpha_i) = \begin{cases} (1-\alpha_i)\dfrac{tr(S_i)}{p} I + \alpha_i S_i & 0 \le \alpha_i \le 1 \\ (2-\alpha_i)S_i + (\alpha_i - 1)S_p^*(t) & 1 \le \alpha_i < 2 \\ (3-\alpha_i)S + (\alpha_i - 2)\dfrac{tr(S)}{p} I & 2 < \alpha_i \le 3 \end{cases}$$

where p is the dimensionality and $f_i = N_i - 1$,

$$S_p^*(t) = \left[\sum_{i=1}^{L} \frac{f_i}{f_i + t - p - 1}\right]^{-1} \sum_{i=1}^{L} \frac{f_i S_i}{f_i + t - p - 1}$$

The criterion function of BLOOC is the same as that of LOOC. BLOOC tends to mitigate the outlier problem.

2.2.4 The Comparison of Performances of RDA, LOOC, and BLOOC

Table 1 is a summary of demonstrations in [5] and [6]. The following are the rules and notation of this summary.

1. Compute the differences of the performances of RDA vs. LOOC in [5] and LOOC vs. BLOOC in [6].

2. If the difference is greater or equal than the standard deviation of LOOC, then round to the hundreds' and display in Table 2.1 in the form x(y). x(y) means that, in case x, the accuracy of this method is y% better than that of other method.

3. Blank cells in two methods mean that both methods have the same performance in this situation within one standard deviation.

4. Expi*E* means Experiment **i** with equal sample size design. Expi*U* means Experiment *i* with unequal sample size design. Reali means real data set i.

Table 2.1. The Summary of Hoffbeck and Tadjudin's Research Experiments

Hoffbeck(1995)			Tadjudin(1998)		
Experiment	RDA(%)	LOOC(%)	Experiment	LOOC(%)	BLOOC1(%)
Exp1			Exp1E		
Exp2			Exp1U		
Exp3		b(4)c(11)d(22)	Exp2E		c(12)d(20)
Exp4			Exp2U		d(8)
Exp5		a(7)b(11)c(9)d(7)	Exp3E	b(12)c(17)d(23)	
Exp6		a(3)b(4)c(6)d(5)	Exp3U	b(3)c(5)d(8)	
Real1			Exp5E	a(13)b(14)c(19)d(21)	
Real2	d(2)		Exp5U	b(4)c(3)d(3)	
Real3		d(1)	Real4		c(22)

In Exp1–6 and T3.2–3.8
a: dim=6
b: dim=10
c: dim=20
d: dim=40
In Real1: Cuprite Site and
Real2: Indian Pine Site
a: dim=10
b: dim=50
c: dim=100
d: dim=191

In Real3: Jasper Ridge site
a: dim=10
b: dim=50
c: dim=100
d: dim=193
In Real4: Indian Pine site (small segment), dim=200
a: training sample size =1% of labeled data
b: training sample size =5% of labeled data
c: training sample size =8% of labeled data
d: training sample size =10% of labeled data

From Table 2.1, we see that

1. LOOC results are better than RDA in most situations.

2. In simulation data, BLOOC is only better than LOOC in experiment 2 (both equal and unequal case).

In addition, computation time is decreasing in the order RDA, BLOOC, and LOOC. According to both accuracy and computation, LOOC is a better choice than the others. However, BLOOC has an advantage of being more resistant to outliers in the training set.

2.3 Mixed Leave-One-Out Covariance (Mixed-LOOC) Estimators

2.3.1 Mixed-LOOC1

LOOC and BLOOC are the linear combination of two of the three matrices, and in some situations, BLOOC is better than LOOC, elsewhere LOOC is better. The difference between LOOC and BLOOC is in those matrices that they use to formulate the

regularized covariance estimator. So we know that only using some of the six matrices will not get good results in all situations. The basic idea of Mixed-LOOC is to use all six matrices to gain the advantages of both LOOC and BLOOC. Hence the first proposed regularized covariance estimator, Mixed-LOOC1, is

$$\hat{\Sigma}_i(a_i,b_i,c_i,d_i,e_i,f_i) = a_i\frac{tr(S_i)}{p}I + b_i diag(S_i) + c_i S_i + d_i\frac{tr(S)}{p}I + e_i diag(S) + f_i S$$

where $a, + b_i + c_i + d_i + e_i + f_i = 1$ and $i = 1,2,...,L$

L : number of classes

p : number of dimensions

S_i : covariance matrix of class i

S :common covariancematrix (pooled)

The mixture parameters are determined by maximizing the average leave-one-out log likelihood of each class:

$$LOOL_i = \frac{1}{N_i}\sum_{k=1}^{N_i}\ln[f(x_k\,|m_{i/k},\hat{\Sigma}_{i/k}(\theta_i))] \quad , \quad \text{where } \vartheta_i = (a_i,b_i,c_i,d_i,e_i,f_i)$$

2.3.2 Mixed-LOOC2

Since using Mixed-LOOC1 is computationally intensive, finding a more simplified estimator will be more practical. Appendix A shows that given two known matrices, the ML estimate of mixture parameters in LOOC and BLOOC are at the end points (α_i =0, 1, 2, or 3). Figures 2.1, 2.2, 2.3, and 2.4 illustrate the relationship between LOOL and the mixture parameter, α_i. The first three figures are generated from simulated data sets; Figure 5 is based on a real data set. The detail information about simulated and real data set is in experiment design (section 2.4). In the case of Figure 2.1, the sample size is greater than the dimensionality. For Figure 2.2, 2.3, and 2.4, the sample sizes are less than the dimensionality. Figure 2.2, 2.4, and 2.4 show that when the ML covariance estimator is singular, the optimal choice of LOOC parameter under LOOL criteria is around the boundary points.

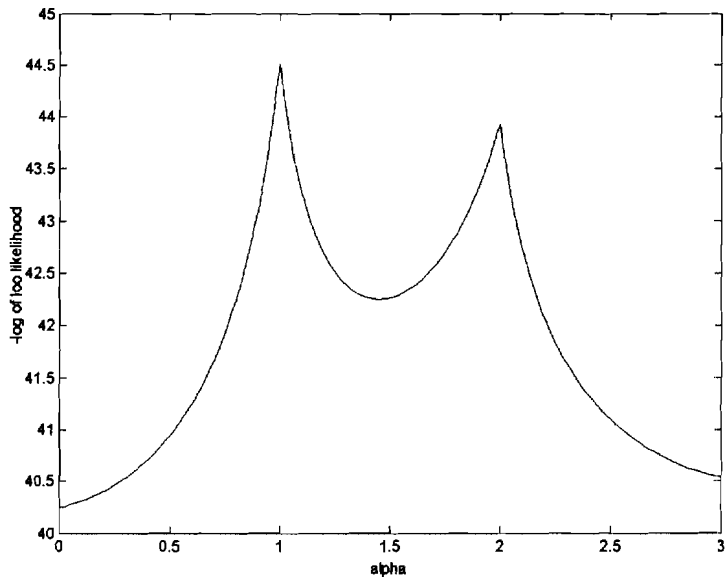


Figure 2.1 -LOOL of class1 in experiment 10 (p=10) and the minimum of -LOOL occurs at alpha= 0

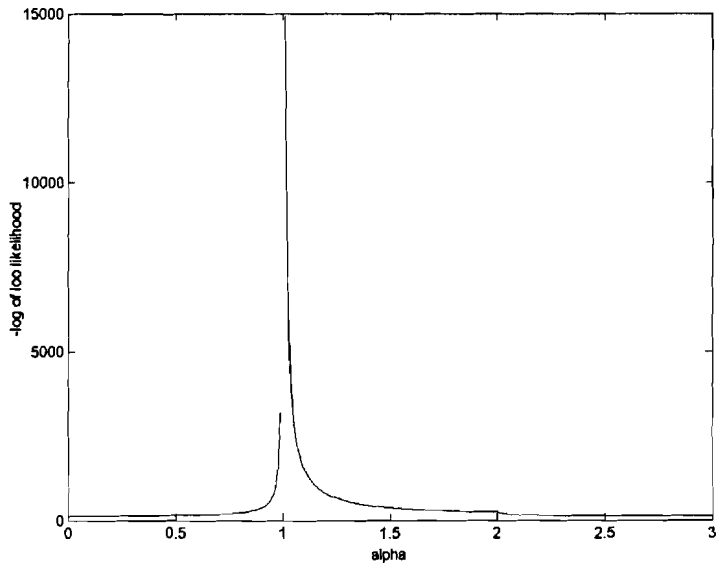


Figure 2.2 -LOOL of class2 in experiment 10 (p=30) and the minimum of -LOOL occurs at alpha=3

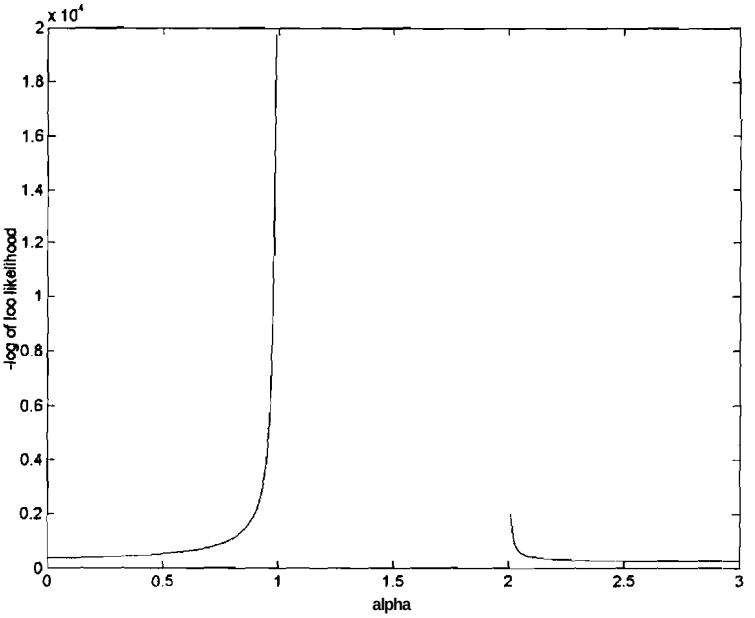


Figure 2.3 -LOOL of class3 in experiment 10 (p=60) and the minimum of -LOOL occurs at alpha= 2.97

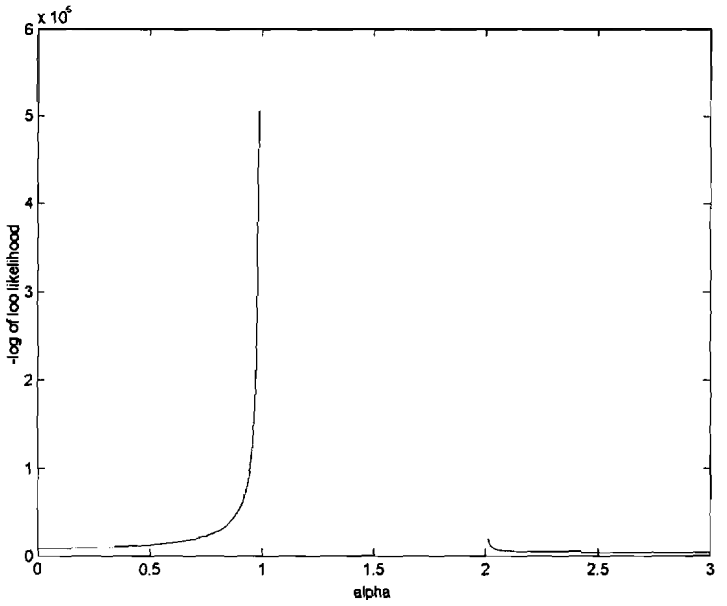


Figure 2.4 -LOOL of class 6 in DC data (p=191) and the minimum of -LOOL occurs at alpha= 2.99

Since a closed form solution for the parameter α_i under the LOOL criteria is not available, and based on the above observations, one of the six support matrices is chosen

to be the covariance estimator to reduce the computation time. The Mixed-LOOC2 is proposed as the following form:

$$\hat{\Sigma}_i(\alpha_i) = \alpha_i \mathrm{A} + (1-\alpha_i)\mathrm{B}$$

where $\mathrm{A} = \frac{tr(S_i)}{P} I$, $diag(S_i)$, S_i, $\frac{tr(S)}{P} I$, $diag(S)$, or S , $\mathrm{B} = S_i$, or $diag(S)$ and α_i is closed to 1. $\mathrm{B} = S_i$, or $diag(S)$ is chosen because if a class sample size is large, S_i will be a better choice. If total training sample size is less than the dimensionality then the common (pooled) covariance **S** is singular but has much less estimation error than S_i . For reducing estimation error and avoiding singularity, $diag(S)$ will be a good choice. The selection criteria is the log leave-one-out likelihood function:

$$LOOL_i = \frac{1}{N_i} \sum_{k=1}^{N_i} \ln[f(x_k \,|m_{i/k}, \hat{\Sigma}_{i/k}(\alpha_i))]$$

The algorithm to decide the Mixed-LOOC2 of each class is to compute LOOL of the 12 covariance estimator combinations, then choose the maximal one. This method needs less computation time than the LOOC proposed in [5].

2.4 Experiment Design for Comparing LOOC, Mixed-LOOC1, and Mixed-LOOC2

In the following experiments, the grid method is used to estimate the mixture parameters of LOOC and Mixed-LOOC1. The range of the parameter a in LOOC is from 0 to 3 and the grids are a = [0, 0.25, 0.5, ..., 2.75, 3]. There are six parameters in Mixed-LOOC1 and the ranges of them are from 0 to 1. The grids of Mixed-LOOC1 are [0, 0.25, 0.5, 0.75, 1]. For Mixed-LOOC2, the parameter a is set to 0.05. In the simulation experiments, performances of all three covariance estimators are compared. Based on computational consideration, only the performances of LOOC and Mixed-LOOC2 are compared for the real data experiments.

Experiments 2.1 to 2.12 are based on simulated data sets. Experiments 2.1 to 2.6 and experiments 2.7 to 2.12 are generated from the same normal distributions respectively. The mean vectors and covariance matrices of experiments 2.1 to 2.6 (and 2.7 to 2.12) are the same as those six experiments in [2] Their mean vectors and covariance matrices are in Appendix B. The only difference between these two set

experiments is that experiment 2.1 to 2.6 are with equal training sample sizes in each class but experiments 2.7 to 2.12 are with different sample sizes in each class. Training and testing sample sizes of these experiments are in Table 2.2. Ther are three different dimensionalities, p=10, 30, 60, in every experiment. At each situation, 10 random training and testing data sets are generated for computing the testing sample accuracies of algorithms, and the standard deviations of the accuracies.

Table 2.2 The Design of Sample Size

	Experiments 2.1 ~ 2.6			Experiments 2.7 ~ 2.12		
Sample Size	Class 1	Class 2	Class 3	Class 1	Class 2	Class 3
Training	10	10	10	30	10	5
Testing	200	200	200	600	200	100

There are four different real data sets, the Cuprite site, which is an area of geologic interest, Jasper Ridge, an ecological site, Indian Pine, an agricultural/forestry site, and DC Mall, an urban site, in experiment 2.13 to 2.16 respectively. All real data sets have 191 bands. There are 8, 6, 6, and 7 classes used in the Cuprite site, Jasper Ridge site, Indian Pine site, and DC Mall, respectively. There are 20 training samples in each class. At each experiment, 10 training and testing data sets are selected for computing the accuracies of algorithms, and the standard deviations of the accuracies.

2.5 Experiment Results

1. In Table 2.3(a), (b), (c), the shadow parts indicate that the differences of performances of LOOC and Mixed-LOOC2 are larger than the standard deviation of Mixed-LOOC2. If the difference is smaller than the standard deviation, we assume that the performances of these methods have no significant difference.

2. All the experiments with significant differences (the shadow parts) indicate that Mixed-LOOC outperformed LOOC.

3. The results of shadow parts show that the differences between Mixed-LOOC and LOOC increase as the number of dimensions increases.

4. When the training sample sizes of classes are unbalance, Mixed-LOOC performed better than LOOC in more situations.

5. Significant differences most often occurred in experiments 2.2, 2.7, and 2.8. Those are the situations in which BLOOC has better performances than LOOC. Since the Mixed-LOOCs are the union version of LOOC and BLOOC, based on these findings, we conclude that the Mixed-LOOCs have advantages over LOOC and BLOOC and can avoid their disadvantages.

6. In most of the experiments, the standard deviations of the Mixed-LOOCs are less than those of LOOC. This suggests Mixed-LOOCs are more stable than LOOC.

7. The results of experiment 2.13 (Cuprite Site) show that Mixed-LOOC2 outperforms LOOC very much. The results of experiment 2.13 and 2.14 (Jasper Ridge site) show that the performances of Mixed-LOOC2 is more stable than those of LOOC

8. The computation time decreases in the order Mixed-LOOC1, LOOC, and Mixed-LOOC2.

Table 2.3(a) The Accuracy of Experiments (p=10)

Experiment	LOOC	Mixed-LOOC1	Mixed-LOOC2
1	0.8630 (0.0425)	0.8632 (0.0243)	0.8602 (0.0466)
2	0.7753 (0.0481)	0.8373 (0.0180)	0.8450 (0.0224)
3	0.8948 (0.0241)	0.8915 (0.0251)	0.8992 (0.0265)
4	0.8875 (0.0309)	0.8893 (0.0263)	0.8837 (0.0386)
5	0.9860 (0.0283)	0.9822 (0.0361)	0.9858 (0.0282)
6	0.9885 (0.0033)	0.9833 (0.0085)	0.9885 (0.0036)
7	0.8500 (0.0286)	0.8622 (0.0252)	0.8641 (0.0249)
8	0.8433 (0.0410)	0.8750 (0.0289)	0.8792 (0.0250)
9	0.9021 (0.0230)	0.9041 (0.0183)	0.9041 (0.0203)
10	0.8928 (0.0247)	0.8948 (0.0204)	0.8940 (0.0245)
11	0.9883 (0.0064)	0.9920 (0.0041)	0.9872 (0.0065)
12	0.9841 (0.0076)	0.9830 (0.0075)	0.9827 (0.0116)

Table 2.3(b) The Accuracy of Experiments (p=30)

Experiment	LOOC	Mixed-LOOC1	Mixed-LOOC2
1	0.8317 (0.0227)	0.8285 (0.0196)	0.8267 (0.0213)
2	0.7263 (0.0510)	0.8700 (0.0205)	0.8813 (0.0204)
3	0.8162 (0.0220)	0.8142 (0.0223)	0.8152 (0.0237)
4	0.7978 (0.0619)	0.7955 (0.0609)	0.7972 (0.0612)
5	0.9993 (0.0014)	0.9975 (0.0037)	0.9993 (0.0014)
6	0.9990 (0.0021)	0.9945 (0.0087)	0.9992 (0.0016)
7	0.8239 (0.0345)	0.8469 (0.0154)	0.8504 (0.0171)
8	0.8718 (0.0311)	0.9210 (0.0130)	0.9189 (0.0118)
9	0.8228 (0.0274)	0.8343 (0.0206)	0.8241 (0.0268)
10	0.8326 (0.0162)	0.8370 (0.0186)	0.8313 (0.0156)
11	0.9976 (0.0021)	0.9994 (0.0008)	0.9984 (0.0018)
12	0.9953 (0.0059)	0.9991 (0.0007)	0.9978 (0.0047)

Table 2.3(c) The Accuracy of Experiments (p=60)

Experiment	LOOC	Mixed-LOOC1	Mixed-LOOC2
1	0.7378 (0.0540)	0.7607 (0.0259)	0.7605 (0.0287)
2	0.6578 (0.0631)	0.8792 (0.0213)	0.8882 (0.0175)
3	0.7632 (0.0265)	0.7615 (0.0235)	0.7583 (0.0281)
4	0.7483 (0.0324)	0.7473 (0.0308)	0.7435 (0.0288)
5	1.0000 (0.0000)	0.9998 (0.0005)	1.0000 (0.0000)
6	1.0000 (0.0000)	1.0000 (0.0000)	1.0000 (0.0000)
7	0.7820 (0.0327)	0.8098 (0.0229)	0.8120 (0.0192)
8	0.8876 (0.0219)	0.9401 (0.0075)	0.9400 (0.0073)
9	0.7947 (0.0216)	0.8024 (0.0150)	0.7958 (0.0203)
10	0.7802 (0.0302)	0.7932 (0.0277)	0.7837 (0.0275)
11	0.9988 (0.0021)	0.9997 (0.0011)	0.9997 (0.0011)
12	1.0000 (0.0000)	1.0000 (0.0000)	1.0000 (0.0000)

Table 2.3(d) The Accuracy of Real Data Sets (p=191)

Real Data Set	LOOC	Mixed-LOOC2
Cuprite	0.7743 (0.1372)	0.9524 (0.0117)
Jasper Ridge	0.9864 (0.0042)	0.9849 (0.0019)
Indian Pine	0.7612 (0.0127)	0.7625 (0.0144)
DC Mall	0.7831 (0.0455)	0.7858 (0.0431)

CHAPTER 3: Discriminate Analysis Feature Extraction Based on Mixed-LOOC2

3.1 Discriminate Analysis Feature Extraction (DAFE)

The purpose of DAFE is to find a transformation matrix A such that the class separability of transformed data $Y=A^TX$ is maximized. Usually within-class, between-class, and mixture scatter matrices are used to formulate the criteria of class separability. A within-class scatter matrix is expressed by [8]:

$$S_w = \sum_{i=1}^{L} P_i E\{(X-m_i)(X-m_i)^T \mid \omega_i\} = \sum_{i=1}^{L} P_i \Sigma_i$$

where L is the number of classes and P_i and m_i are the prior probability and mean vector of the class i, respectively.

A between-class scatter matrix is expressed as

$$S_b = \sum_{i=1}^{L} P_i (m_i - m_0)(m_i - m_0)^T = \sum_{i=1}^{L-1} \sum_{j=i+1}^{L} P_i P_j (m_i - m_j)(m_i - m,)^T$$

where m_0 represents the expected vector of the mixture distribution and is given by

$$m_0 = E\{X\} = \sum_{i=1}^{L} P_i m_i$$

Let $Y = A^t X$,then we have

$$S_{wY} = A^t S_{wX} A \quad \text{and} \quad S_{bY} = A^t S_{bX} A$$

The optimal features are determined by optimizing the criterion given by

$$J_1 = tr(S_{wY}^{-1} S_{bY})$$

The optimum A must satisfy

$$(S_{wX}^{-1} S_{bX}) A = A(S_{wY}^{-1} S_{bY})$$

This is a generalized eigenvalue problem [9] and usually can be solved by the QZ algorithm. But if the covariance is singular, the result will have a poor and unstable performance on classification. In this section, the ML covariance estimate will be replaced by Mixed-LOOC when it is singular. Then the problem will become a simple eigenvalue problem.

3.2 Comparison of DAFE and DAFE Based on Mixed-LOOC2

For convenience, denote DAFE based on ML estimators as DAFE and DAFE based on MLOOC2 as DAFE-Mix2, Gaussian classifier based on ML estimators as GC, and Gaussian classifier based on MLOOC2 estimators as GC-Mix2. Experiments 3.1 to 3.3 are for determining the performances of DAFE-Mix2. The classification process in experiment 3.1 is to use DAFE then GC, in experiment 3.2 use DAFE-Mix2 then GC, and in experiment 3.3 use DAFE-Mix2 then GC-Mix2. The sample sizes of experiment 3.2 and 3.3 are the same as those of experiments 2.13 to 2.16 (N_i=20). Since using those sample sizes in DAFE will cause very poor results, we increase the sample size of each class in Cuprite, Jasper Ridge, Indian Pine, and DC Mall data sets up to 40. The results of those experiments are shown in Table 3.1.

Table 3.1 The Results of DAFE Based on ML Estimators and MLOOC2

	Exp3.1(Ni=40)	Exp3.2(Ni=20)	Exp3.3(Ni=20)
Real **Data** Set	DAFE -GC	DAFE-Mix2 -GC	DAFE-Mix2 -GC-Mix2
Cuprite	0.8943 (0.0205)	0.9474 (0.0194)	0.9627 (0.0196)
Jasper Ridge	0.9127 (0.0243)	0.9782 (0.0120)	0.9876 (0.0036)
Indian Pine	0.5727 (0.0156)	0.7547 (0.0316)	0.7562 (0.0191)
DC Mall	0.7392 (0.0530)	0.8691 (0.0282)	0.8600 (0.0345)

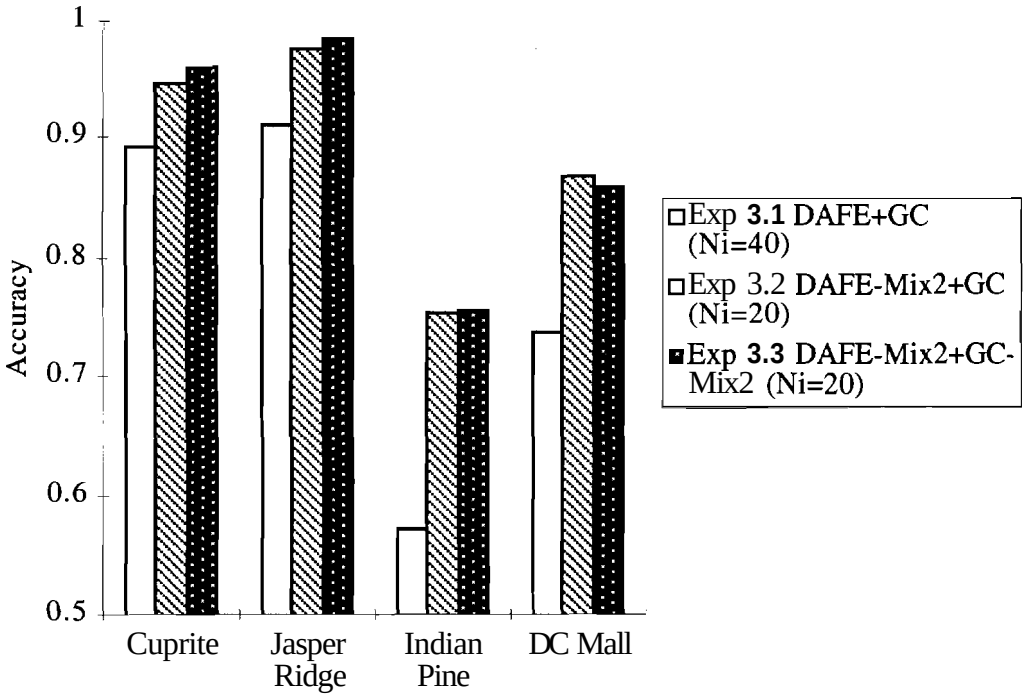


Figure 3.1 The Mean Accuracies of Three Classification Procedures

From above results we find:

1. Using DAFE-Mix2 provides higher accuracy and, in most cases, smaller standard deviation than using only DAFE.

2. Comparing Table 2.3(d) and Table 3.1, we find that in all data sets except the DC Mall sets, using DAFE-Mix2 then GC or GC-Mix2 have similar results with only using GC-Mix2. But the results for DC Mall show that using DAFE-Mix2 then GC or GC-Mix2 gave a significant improvement.

3. From Table 3.1 and Figure 3.1, DAFE-Mix2 -GC-Mix2 looks like the best choice.

3.3 Concluding Comments

The singularity or near-singularity problem often occurs in the case of high dimensional classification. From the above discussion, we know that finding a suitable regularized covariance estimator is a way to mitigate this problem. Further, Mixed-LOOC2 has advantages over LOOC and BLOOC and needs less computation than those two. The problems of class statistics estimation error resulting from training sets of finite size grows rapidly with dimensionality, thus making it desirable to use no larger feature space dimensionality than necessary for the problem at hand, and therefore the importance of an effective, case-specific feature extraction procedure. Usually DAFE cannot be used when the training sample size is less than dimensionality. The new procedure, DAFE-Mix2, overcomes this shortcoming, and can provide higher accuracy when the sample size is limited.

CHAPTER 4: GAUSSIAN MIXTURE CLASSIFIER BASED ON MIXED-LOOC2

4.1 Introduction

The normal mixture density, which models the density as the sum of one or more weighted Gaussian components, is a compromise between Gaussian and non-parametric densities. It allows more flexibility than the Gaussian density, yet requires fewer parameters to be estimated than non-parametric densities. Most methods in this area usually assume that if one class can be divided by several normal distributed subgroups then the sample size of each subgroup should not be less than the dimensionality. The purpose of this section is to provide the evidence that we can divide one class into some subgroups whose sample sizes may be less than the dimensionality, and the classification result could be improved by this way.

There are two steps to design a quadratic mixture classifier. The first is parameter estimation and the second is model selection. In this study, NM (nearest means or K-mean) clustering and EM (expectation-maximization)clustering are used in the parameter estimation part. There are many indices for model selection. In this research, only the performances of AIC, BIC, NEC, and ICL-BIC, described below, are tested.

4.2 Parameter Estimation Methods

4.2.1 Normal Mixture Density

In order to model non-Gaussian classes, consider the quadratic mixture density, which is the weighted summation of L Gaussian density functions:

$$p(x) = \sum_{k=1}^{L} \alpha_k f(x \mid m_k, \Sigma_k) \tag{4.1}$$

$$\text{where } f(x \mid m_k, \Sigma_k) = \frac{1}{\sqrt{(2\pi)^p |\Sigma_k|}} \exp[-\frac{1}{2}(x - m_k)^T \Sigma_k^{-1}(x - m_k)] \tag{4.2}$$

Each term in the summation of (4.1) is called a component of the normal mixture density. The weights α_k, which must sum to unity, are a ***priori*** probabilities of the components. In practice the parameters of the density function (L, α_k, m_k, and Σ_k for k = 1, 2, ...L) are usually not known and must be estimated from the training samples. Multimode classes can be represented by a mixture density with one or more components representing each mode. Since the covariance matrix of each component should be invertible, ordinarily the sample size of each component should not be less than the dimensionality of the data. In this section, the new mixture classifier will relieve this limitation.

4.1.2 Nearest Means Clustering

The nearest means clustering algorithm, which requires the number of clusters to be specified, is used and proceeds as follows [8]:

Step 1. Choose an initial classification of the samples into L clusters and compute the mean of each cluster.

Step 2. Reclassify each sample to the cluster with the lowest Euclidean distance between the mean of the cluster and the sample.

Step 3. If the classification of any sample has changed, calculate new mean vectors and return to step 2; otherwise stop.

4.1.3 EM clustering

The EM (expectation-maximization) algorithm consists of two major steps: an expectation step, followed by a maximization step. The expectation is with respect to the unknown underlying variables, using the current estimate of parameters and conditioned upon the observations. The maximization provides a new estimate of the parameters. It is an iterative method for computing the maximum likelihood estimates of the mean vector, covariance matrix, and a ***priori*** probability of the components in a normal mixture. It can

correctly identify clusters that have the same mean vectors but different covariance matrices. The number of components L must be specified at the outset. The method proceeds as follows [8]:

Step 1. Choose an initial classification of the samples into L clusters.

Step 2. Estimate the a ***priori*** probability α_i , the mean vector m_i , and the sample covariance Σ_i of each cluster.

Step 3. Compute $q_{i,j}$, which is the a ***posteriori*** probability of class **i** given sample j:

$$q_{i,j} = \frac{\alpha_i f(x_j \mid m_i, \Sigma_i)}{\sum_{k=1}^{L} \alpha_k f(x_j \mid m_k, \Sigma_k)} \tag{4.3}$$

Step 4. Compute new estimates of the a ***priori*** probability, mean vector, and sample covariance of each cluster:

$$\alpha_i = \frac{1}{N} \sum_{j=1}^{N} q_{i,j} \tag{4.4}$$

$$m_i = \frac{1}{N\alpha_i} \sum_{j=1}^{N} q_{i,j} x_j \tag{4.5}$$

$$\Sigma_i = \frac{1}{N\alpha_i} \sum_{j=1}^{N} q_{i,j} (x_j - m_i)(x_j - m_i)^T \tag{4.6}$$

where N is the total number of samples to be clustered.

Step 5. If any $q_{i,j}$ changed, repeat steps 3-4, otherwise stop.

4.3 Model Selection Indices

In the multivariate normal mixture model, data $x_1, \ldots, x_n$ in R^p are assumed to be a sample from a probability distribution with density (4.1)

$$p(x) = \sum_{k=1}^{\infty} \alpha_k f(x, a_k)$$

where the α_k's are the mixing proportions $(0 < \alpha_k < 1)$ for all $k = 1,...,L$ and $\Sigma_{k=1}^{L} \alpha_k = 1$ and $\phi(x, a_k)$ denotes the p-dimensional Gaussian density with mean m_k and covariance matrix Σ_k with $a_k = (m_k, \Sigma_k)$. The maximized log likelihood of $\Psi = ((\alpha_1, a_1),...,(\alpha_K, a_K))$ for the sample $x_1,...,x_n$ is denoted

$$L(\Psi) = \sum_{i=1}^{n} \log[\sum_{k=1}^{L} \hat{\alpha}_k f(x_i, \hat{\mu}_k, \hat{\Sigma}_k)]$$

with $\hat{\alpha}_k$ and $\hat{a}_k$ denoting the maximum likelihood estimates of the corresponding parameters.

Various criteria to be minimized have been proposed to measure a model's suitability by balancing model fit and model complexity.

4.3.1 Akaike information criterion (AIC)

The Akaike information criterion (AIC; [10]) is defined as

$$AIC(\Psi) = -2L(\Psi) + 2\nu(\Psi)$$

where $\nu(\Psi)$ is the number of free parameters in the mixture model Ψ.

It was observed that AIC is order inconsistent and tends to overfit models [11]. In the mixture context, that means that AIC trends to overestimate the true number of components [12], [13].

4.3.2 Bayesian information criterion (BIC)

The Bayes factor for one model against another model is the posterior odds for that model against the other when neither model is favored over the other a priori. It is equal to the ratio of marginal or integrated likelihood for each model. In [14], the integrated likelihood of the data $d = (x_1,...,x_n)$ given the model Ψ is

$$P(d \mid \Psi) = \int P(d \mid \Psi, \theta) P(\theta \mid \Psi) d\theta$$

where $P(\theta \mid \Psi)$ is the prior density of θ. A classical way to approximate the integrated likelihood consists in using the Bayesian information criterion [15]. Noting $\hat{\vartheta}$ the maximum likelihood estimate of θ, this approximation is

$$\log P(d \mid \Psi) = \log P(d \mid \Psi, \hat{\theta}) - \frac{v(\Psi)}{2} \log n + O(1).$$

Thus the Bayesian information criterion (BIC) is given by

$$BIC(\Psi) = -2L(\Psi) + v(\Psi) \log n.$$

4.3.3 Normalized Entropy Criterion and Classification Likelihood Criterion

Classification Likelihood Criterion (CLC) was proposed by [17], Normalized Entropy Criterion (NEC) was proposed by [13] and modified by [16]. It was derived from a relation emphasizing the differences between the likelihood and the "fuzzy" classification likelihood of the mixture or, in the same manner, between the likelihood and the classification likelihood of the mixture [17]. Let

$$t_{ik} = \frac{\hat{\alpha}_k \phi(x_i, \hat{a}_k)}{\sum_{j=1}^{k} \hat{\alpha}_j \phi(x_i, \hat{a}_j)}$$

be the estimated conditional probability that x_i rises from the kth mixture component. The fuzzy classification likelihood criterion is defined as

$$CLC(\Psi) = \sum_{k=1}^{L} \sum_{i=1}^{n} t_{ik} \log[\hat{\alpha}_k \phi(x_i, \hat{a}_k)]$$

and the entropy is defined as

$$E(\Psi) = -\sum_{k=1}^{L} \sum_{i=1}^{n} t_{ik} \log t_{ik} \geq 0.$$

Then we have

$$CLC(\Psi) = L(\Psi) - E(\Psi),$$

$CLC(\Psi)$ is related to the fuzzy classification matrix $t = \{t_{ik}\}$. If the mixture components are well-separated, then $E(\Psi) \approx 0$. Otherwise, $E(\Psi)$ will have a large value. Thus, $E(\Psi)$ can be regarded as a measure of the ability of the L-component mixture model to provide a relevant partition of the data $(x_1, \ldots, x_n)$. The relation shows that the classification likelihood term $CLC(\Psi)$ can be regarded as a compromise between the fit

of the data to the mixture model, measured with the log likelihood $L(\Psi)$, and the ability of the mixture model to provide a classification in well-separated clusters, measured with the entropy term $E(\Psi)$ [18].

As a consequence, the entropy of the classification matrix t gives raise to several classification criteria [13], which are $E(\Psi)$, its normalized version

$$NEC(\Psi) = \frac{E(\Psi)}{L(\Psi) - L_1(\Psi)}$$

where $L_1(M)$ denotes the maximized log-likelihood for a single Gaussian distribution. In [13], the entropy term is equal to 0 when the number of components (nc) is 1. According to [16], setting NEC=1 when nc = 1 corrects for the tendency of original version to prefer nc > 1 when the true nc = 1.

4.3.4 Integrated Classification Likelihood Criterion

The Integrated Classification Likelihood Criterion was proposed in [18] and is an attempt to overcome the shortcomings of BIC and CLC. There are two versions of this index [19]. The full version is

$$ICLC(\Psi) = -2L(\Psi) + 2E(\Psi) + \beta \log n + 2n\sum_{i=1}^{L} \hat{\alpha}_i \log \hat{\alpha}_i - 2K(n\hat{\alpha}_1, \cdots, n\hat{\alpha}_L)$$

were $\beta = \nu(\Psi) - (L-1)$ is the number of free parameters in $\hat{a}$ and

$$K(n_1, \cdots, n_L) = \sum_{i=1}^{L} \Gamma(n_i + \delta) + \log \Gamma(n + L\delta) - L \log \Gamma(\delta) + \log \Gamma(L\delta$$

In [19], δ is set as _. When the sample size of each component is large enough, the Gamma function can be replaced by Stirling's formula

$$\Gamma(u) \approx u^{u+\frac{1}{2}} \exp(-u)(2\pi)^{\frac{1}{2}}$$

On setting $\delta = 1$ and neglecting terms of order $O(1)$, we have

$$K(n\hat{\alpha}_1, \cdots, n\hat{\alpha}_L) \approx n\sum_{i=1}^{L} \hat{\alpha}_i \log \hat{\alpha}_i - \frac{1}{2} L \log n$$

Then we can get the reduced version of ICL and it is named ICL-BIC in [19]

$$ICL - BIC(\Psi) = -2L(\Psi) + 2E(\Psi) + v(\Psi)\log n$$

4.4 Gaussian Mixture Classifier Based on Mixed-LOOC

One of limitations of the above model selection indices is that the component sample size should be greater than the dimensionality. The new algorithms based on Mixed-LOOC will release this constraint.

4.4.1 Mixture Classifier Using Mixed-LOOC and Nearest Means Clustering

The algorithm of a mixture classifier using Mixed-LOOC2 and nearest means (NM) clustering is

Step 1. Compute Mixed-LOOC2 of each class and for each class, use nearest means clustering to find the components.

Step 2. Compute Mixed-LOOC2 of each component in classes.

Step 3. Compute the model selection index using Mixed-LOOC2 to replace ML covariance estimate.

Step 4. If the number of components in classes is 1, then use the Mixed-LOOC2 of this class as its covariance estimator.

Step 5. Compute the mixture density function to form the Bayesian mixture classifier.

4.4.2 Mixture Classifier Using Mixed-LOOC and EM clustering

The algorithm of mixture classifier using Mixed-LOOC2 and EM clustering is

Step 1 Compute Mixed-LOOC2 of each class and for each class.

Step 2 Use EM clustering to find the components. But, in the estimating covariance steps of EM clustering, the ML estimator should be replaced by Mixed-LOOC2.

Step 3 Compute the model selection index using Mixed-LOOC2 to replace ML covariance estimate.

Step 4 If the number of components in classes is 1, then using the Mixed-LOOC2 of this class as its covariance estimator.

4.5 Simulated and Real Data Experiments

4.5.1 Simulation Data Experiment Design

In simulation experiment, the performances of mixture classifiers based on NM and EM clustering with model selection indices AIC, BIC, NEC, ICL-BIC and their Mixed-LOOC versions are compared.

In classification problems, there are two kinds of mixture situations. One is the components of each class are grouped together and do not mix with those of other classes, like Figure 4.1(a). The other is that the components of different classes mix together, like Figure 4.1(b). In first case, the mixture classifier may have performance similar to the a simple quadratic classifier if the class sample sizes are large enough. But when the class sample is small then the performance of a mixture classifier may not be as good as that of Gaussian quadratic classifier due to estimation error. In second case, the mixture classifier would be expected to do a better job when the class sample sizes are large enough, but if class sample is small then the mixture classifier may have more severe problems.

The simulation study will focus on the second situation and try to find out which combination of parameter estimation and model selection will give a better result. The class sample sizes and the class mean vectors and covariance matrices of simulated data are in Table 4.1(a). The clustering algorithm used in experiments 4.1 and 4.2 is NM clustering and that used in experiments 4.3 and 4.4 is EM clustering. Five different dimensionality (2,4,10,20,60) and three different class sample sizes are tested. In each situation (Table 4.1(b)), 10 random training and testing data sets are generated for computing the accuracies of algorithms, and the standard deviations of the accuracies.

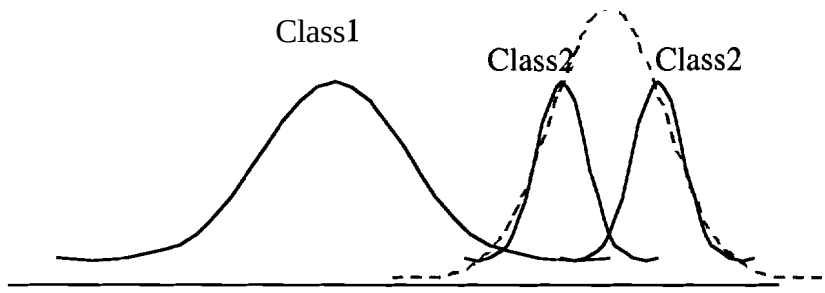


Figure 4.1(a) Class 1 is not between subcomponents of class 2

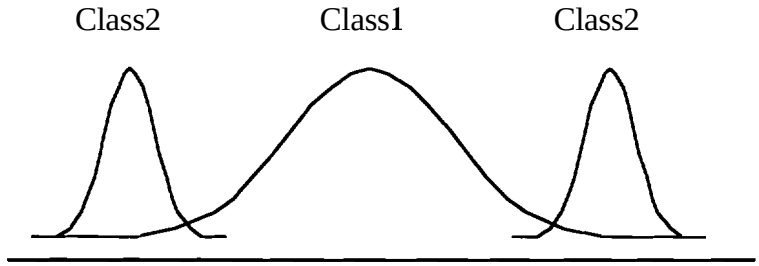


Figure 4.1(b) Class 1 is between subcomponents of class 2

Table 4.1(a) The class mean vectors and covariance matrices of simulated data

		class 1	class 2	
Dim=2,4,10,20,60		component 1	component 1	component 2
Mean Vector		[0,0,…,0]	[1,1, …,1]	[-1,-1, …,-1]
Covariance	Exp4.1 and 4.3	I	0.1I	0.1I
	Exp4.2 and 4.4	I	I	I
Training Class Sample Size		30, 60, 300	15,30,150	15,30,150
Testing Class Sample Size		30, 60, 300	15,30,150	15,30,150

Table 4.1(b) Dimensionality and class sample size of situation 1 to 15

Situation	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Dim	2	4	10	20	60	2	4	10	20	60	2	4	10	20	60
Ni	30	60	300	30	60	300	30	60	300	30	60	300	30	60	300

4.5.1 Real Data Experiment Design

Hyperspectral data from the Washington, DC Mall is used in real data experiments, and the better clustering algorithm, chosen from the results of simulation studies, is used. Two different class sample sizes (20 and 100) and two different dimensionalities (20 and 7) are used in Experiment 4.5. There are 191 bands in the DC Mall image data and every 10-th band and 30-th band, which begins from the first one are selected, for the 20 and 7 bands cases. At each situation, 10 random training and testing data sets are generated for computing the testing sample accuracies of algorithms, and the number of subcomponents in each class..

4.6 Experiment Results

For connivance, denote the mixture classifier built on the original model selection index as the index itself (for example: AIC) and the mixture classifier built on the model selection index based on Mixed-LOOC2 as the index itself with a "Mix" suffix in tables and figures.

4.6.1 Simulation Experiment Results

The results of experiments 4.1 to 4.4 are displayed in tables 4.2(a), (b), (c), (d) and figures 4.2(a), (b), (c), (d). The results displayed in the figures are the accuracies using BIC_Mix in situations 1 to 15 (from top to bottom of the tables). They show that

1. Generally speaking, the mixture classifier BIC_Mix gave better performance than the others.

2. The shadow parts in tables indicate those cases that the performance of mixture classifier BIC_Mix is significantly better than that of the simple quadratic

classifier. In those unmarked situations, these two classifiers have equivalent performances.

3. From tables 4.2(a), (b), (c), (d), the performance of the mixture classifier using NM clustering was better than that of the mixture classifier using EM clustering.

4. The tables 4.2(a) and (b) (NM cases) show that if the subcomponents are well separated (I-0.1I case) then mixture classifiers (with/without using Mixed–LOOC2) have advantages in low dimensionality situations. When the dimensionality goes up, only the mixture classifiers using Mixed_LOOC2 can have similar results with a Gaussian classifier. Those not using Mixed–LOOC2 yield poorer results due to estimation error increasing. If the subcomponents are well separated (I-I case) then increasing the dimensionality will help the mixture classifiers using Mixed–LOOC2 to obtain better performance but will reduce the accuracy of those not using Mixed–LOOC2.

5. For estimating subcomponents, BIC_Mix is still a better choice than the others.

4.6.2 Real Data Experiment Results

The simulation study suggested that NM clustering is a better choice to build a mixture classifier, so NM clustering is used on real data experiment. The results are in Table 4.3. It shows that BIC_Mix still has the better performance than others in all cases.

4.7 Concluding Comments

The above results show that, sometimes, an original mixture classifier outperforms a Gaussian classifier but sometimes not. The proposed mixture classifier using BIC_Mix has the advantages of both classifiers and outperforms those two in some situations. Before classifying hyperspectral image data, feature extraction is usually a preprocessing step. The effect of combining feature extraction and mixture classification will be discussed in Chapter 6.

Table 4.2(a) Results of experiment 4.1 (I-0.1I case) using NM clustering

Accuracy										
Model Selection		1 mode	AIC	AIC Mix	BIC	BIC Mix	NEC	NEC Mix	ICLBIC	ICLBIC Mix
Dimensionality	Sample Size									
2	30	0.7333	0.8333	0.8567	0.8433	0.8567	0.7383	0.7583	0.735	0.7417
2	60	0.7742	0.8617	0.8608	0.8617	0.8608	0.8042	0.8233	0.7708	0.7742
2	300	0.7758	0.8788	0.88	0.8788	0.88	0.8717	0.8702	0.8717	0.869
4	30	0.9167	0.83	0.9617	0.8333	0.9617	0.9167	0.9183	0.9167	0.89
4	60	0.9158	0.9408	0.9625	0.9475	0.9625	0.9092	0.9325	0.9167	0.9142
4	300	0.9225	0.968	0.9703	0.968	0.9703	0.9655	0.9663	0.9655	0.9668
10	30	0.9683	0.7233	0.9017	0.755	0.9617	0.9683	0.9517	0.9683	0.95
10	60	0.99	0.8075	1	0.8075	0.99	0.99	0.99	0.99	0.99
10	300	0.9945	0.9995	0.9997	0.9995	0.9997	0.9975	0.9997	0.9945	0.9947
20	30	0.945	0.7567	0.985	0.7567	0.97	0.945	0.97	0.71	0.74
20	60	0.9967	0.6892	0.9933	0.7342	0.9933	0.9967	0.9933	0.8975	0.8958
20	300	1	0.9228	1	0.9228	1	0.9995	1	1	1
60	30	0.5	0.5	0.9983	0.5	1	0.5	0.9983	0.5	0.9983
60	60	0.5	0.5	0.9992	0.5	1	0.5	0.9992	0.5	0.9992
60	300	1	1	1	1	1	1	1	1	1

Number of Components										
Model Selection			AIC	AIC Mix	BIC	BIC Mix	NEC	NEC Mix	ICLBIC	ICLBIC Mix
Dimension	Sample Size	Class								
2	30	1	1.1	1	1	1	1	1	1	1
		2	2.4	2	2	2	2	2	1.5	1
2	60	1	1	1	1	1	2	1	1	1
		2	2	2	2	2	4	5	1.5	1
2	300	1	1	1	1	1	4.5	5	1	1
		2	2	2	2	2	6	6	6	6
4	30	1	1.8	1	1.4	1	1	1.5	1	1.1
		2	2.9	2	2.9	2	1	3	1	1
4	60	1	1	1	1	1	2.5	2.5	1	1
		2	2.7	2	2	2	2	4	1.5	1
4	300	1	1	1	1	1	1	2	1	1
		2	2	2	2	2	6	6	6	6
10	30	1	1.5	1	1.2	1	1	1.5	1	1
		2	2.1	1.8	1.7	1.3	1	1	1	1
10	60	1	2.4	1	2.3	1	1	3	1	1
		2	3	2	2.6	1	1	1.5	1	1
10	300	1	1	1	1	1	5.5	1	1	1
		2	2	2	2	2	6	6	1	1
20	30	1	1.3	1	1.3	1	1	1	3.5	3.5
		2	1.3	1.6	1.3	1	1	1	1	1
20	60	1	1.4	1	1.4	1	1	1	2	2
		2	2.6	1	2.1	1	1	1	1	1
20	300	1	2	1	2	1	4	1	1	1
		2	2.7	2	2.7	1	2	3.5	1	1
60	30	1	1	1	1	1	1	1	1	5
		2	1.2	1	1.2	2	1	1	1	1
60	60	1	1	1	1	1	1	1	1	1
		2	1.1	1	1.1	2	1	1	1	1
60	300	1	1	1	1	1	1	1	1	1
		2	2	1	1	1	1	1	1	1

Table 4.2(b) Results of experiment 4.2 (I-I case) using NM clustering

Accuracy									
Model Selection		1 mode	AIC	AIC_Mix	BIC	BIC_Mix	NEC	NEC_Mix	ICLBIC	ICLBIC_Mix
Dimensionality	Sample Size									
2	30	0.6333	0.6233	0.645	0.6333	0.6467	0.6033	0.64	0.6333	0.6467
2	60	0.6575	0.6583	0.6608	0.6575	0.6575	0.6658	0.6608	0.6575	0.6575
2	300	0.6773	0.6815	0.6842	0.679	0.682	0.6138	0.6167	0.6773	0.6795
4	30	0.6767	0.6983	0.6933	0.6583	0.6867	0.6767	0.6917	0.6767	0.6867
4	60	0.7358	0.7058	0.7467	0.6967	0.7425	0.6767	0.7467	0.73	0.7425
4	300	0.7785	0.7848	0.7887	0.7848	0.7813	0.7663	0.7733	0.7785	0.7823
10	30	0.71	0.66	0.835	0.63	0.835	0.71	0.835	0.6983	0.805
10	60	0.745	0.7333	0.8433	0.685	0.8433	0.745	0.8275	0.745	0.8433
10	300	0.8735	0.8632	0.9193	0.8537	0.9032	0.8832	0.9137	0.8735	0.9032
20	30	0.665	0.5983	0.9233	0.5983	0.9233	0.665	0.9233	0.655	0.885
20	60	0.7483	0.655	0.935	0.6442	0.935	0.7483	0.935	0.7483	0.935
20	300	0.8878	0.8607	0.9832	0.805	0.9417	0.8438	0.9648	0.8878	0.9417
60	30	0.5	0.5	0.9717	0.5	0.9717	0.5	0.9717	0.5	0.9717
60	60	0.5	0.5	0.9683	0.5	0.9683	0.5	0.9683	0.5	0.9683
60	300	0.8147	0.8147	0.9683	0.8147	0.9683	0.8147	0.9683	0.8147	0.9683

Number of Components										
Model Selection			AIC	AIC_Mix	BIC	BIC_Mix	NEC	NEC_Mix	ICLBIC	ICLBIC_Mix
Dimension	Sample Size	Class								
2	30	1	1	1	1	1	1.5	1	1	1
		2	1.5	1.1	1	1	2.5	1.5	1	1
2	60	1	1	1	1	1	1	1	1	1
		2	1.2	1.1	1	1	2	1.5	1	1
2	300	1	1	1	1	1	5.5	5.5	1	1
		2	1.9	1.9	1.4	1.1	1	1	1	1
4	30	1	1.8	1	1.4	1	1	1.5	1	1
		2	2.9	1.1	2.2	1	1	2	1	1
4	60	1	2	1	2	1	3.5	1	1	1
		2	3.5	1.3	1.5	1	4.5	4.5	1.5	1
4	300	1	1	1	1	1	2	2	1	1
		2	2	2	2	1.1	1.5	1	1	1
10	30	1	1.7	1	1.7	1	1	1	1.5	1.5
		2	2	1	1.2	1	1	1	1	1
10	60	1	1.4	1	1.4	1	1	3	1	1
		2	2.5	1	1.8	1	1	3	1	1
10	300	1	1	1	1	1	6	1	1	1
		2	2.2	2	1.9	1	6	6	1	1
20	30	1	1.3	1	1.3	1	1	1	1.1	1.1
		2	1.5	1	1.5	1	1	1	1	1
20	60	1	1.8	1	1.8	1	1	1	1	1
		2	1.4	1	1	1	1	1	1	1
20	300	1	1.3	1	1.3	1	2	1.5	1	1
		2	3.1	2	2.4	1	1.5	4	1	1
60	30	1	1	1	1	1	1	1	1	1
		2	1	1	1	1	1	1	1	1
60	60	1	1	1	1	1	1	1	1	1
		2	1	1	1	1	1	1	1	1
60	300	1	1	1	1	1	1	1	1	1
		2	1	1	1	1	1	1	1	1

Table 4.2(c) Results of experiment 4.3 (I-0.1I case) using EM clustering

Accuracy(nm)

Model Selection		1 mode	AIC	AIC_Mix	BIC	BIC_Mix	NEC	NEC_Mix	ICLBIC	ICLBIC_Mix
Dimensionality	Sample Size									
2	30	0.7333	0.745	0.705	0.8067	0.8167	0.815	0.8217	0.8433	0.8267
2	60	0.7742	0.7858	0.825	0.8483	0.8542	0.8583	0.8583	0.8533	0.855
2	300	0.7758	0.833	0.8743	0.8383	0.8778	0.8782	0.8782	0.8782	0.869
4	30	0.9167	0.83	0.8333	0.9167	0.9183	0.915	0.8983	0.92	0.925
4	60	0.9158	0.825	0.9158	0.9383	0.9492	0.93	0.9467	0.9458	0.935
4	300	0.9225	0.9667	0.9622	0.922	0.9683	0.9633	0.9685	0.963	0.9683
10	30	0.9683	0.8717	0.8117	0.9683	0.9683	0.9283	0.9283	0.9683	0.9683
10	60	0.99	0.9567	0.9325	0.99	0.99	0.9658	0.9775	0.99	0.99
10	300	0.9945	0.9787	0.9947	0.9995	0.9988	0.9995	0.9992	0.9985	0.9985
20	30	0.945	0.945	0.945	0.945	0.945	0.945	0.945	0.945	0.945
20	60	0.9967	0.9158	0.9092	0.9967	0.9975	0.9617	0.9925	0.9967	0.9967
20	300	1	0.9997	0.9997	1	1	0.9998	1	1	1
60	30	0.5	0.5	0.9983	0.5	0.9983	0.5	0.9983	0.5	0.9983
60	60	0.5	0.5	0.9992	0.5	0.9992	0.5	0.95	0.5	0.9992
60	300	1	1	1	1	1	1	1	1	1

Number of Components

Model Selection			AIC	AIC_Mix	BIC	BIC_Mix	NEC	NEC_Mix	ICLBIC	ICLBIC_Mix
Dimension	Sample Size	Class								
2	30	1	4.2	5.5	1	1.5	2.7	2.5	1	1
		2	4.3	5.1	2.3	3.3	1.9	1.9	2	1.9
2	60	1	4.1	4.7	1	1	2.9	1	1	1
		2	4.3	4.8	2.4	2.3	2	2.1	2.1	2.1
2	300	1	2.7	3.1	1	1	1	1	1	1
		2	3.9	3.1	2.2	2.4	2	2.1	2.1	1.9
4	30	1	2.9	5.6	1	1	2.3	2.5	1	1
		2	2.9	5	1	2.1	2.1	2.3	1.1	1.1
4	60	1	4	5.5	1	1	3.2	3.4	1	1
		2	3.1	5	1.8	2.1	2.1	2.4	1.9	1.7
4	300	1	3.9	5	1	1	1	1	1	1
		2	3.9	4.7	2.3	2.1	1.9	2	1.9	2.1
10	30	1	1.4	1.6	1	1	1.4	1.5	1	1
		2	1.8	1.9	1	1	1.2	1.4	1	1
10	60	1	1.6	3.6	1	1	2.1	2.7	1	1
		2	2.3	3.7	1	1	1.9	2	1	1
10	300	1	1.8	3.2	1	1	4.1	3.8	1	3.8
		2	2.9	5.1	2	1.9	2.1	2.3	1.9	2.3
20	30	1	1	1	1	1	1	1	1	1
		2	1	1	1	1	1	1	1	1
20	60	1	1	1.8	1	1.1	1.1	1.3	1	1
		2	1.7	2	1	1	1.4	1.1	1	1
20	300	1	1.1	3.5	1	1	1.3	3.3	1	3.3
		2	2.5	4.3	1	1	2	2	1	2
60	30	1	1	1	1	1	1	3.9	1	1
		2	1	1	1	1	1	2.2	1	1
60	60	1	1	1	1	1	1	2.6	1	1
		2	1	1	1	1	1	2	1	1
60	300	1	1	1.1	1	1	1	1.8	1	1
		2	1	3.1	1	1	1	1	1	1

Table 4.2(d) Results of experiment **4.4** (I-I case) using EM clustering

Accuracy(nm)										
Model Selection		1 mode	AIC	AIC_Mix	BIC	BIC_Mix	NEC	NEC_Mix	ICLBIC	ICLBIC_Mix
Dimensionality	Sample size Size									
2	30	0.6333	0.5617	0.5633	0.6117	0.615	0.62	0.6267	0.6267	0.6267
2	60	0.6575	0.5983	0.5875	0.6575	0.6575	0.6333	0.6383	0.6575	0.6575
2	300	0.6773	0.6693	0.6735	0.6802	0.6802	0.6773	0.6773	0.6773	0.6773
4	30	0.6767	0.6033	0.7615	0.6617	0.7782	0.6433	0.782	0.6767	0.7785
4	60	0.7358	0.655	0.6483	0.7358	0.7358	0.6817	0.7158	0.7358	0.7358
4	300	0.7785	0.766	0.7615	0.7782	0.7782	0.7825	0.782	0.7785	0.7785
10	30	0.71	0.6583	0.625	0.71	0.71	0.66	0.6417	0.71	0.71
10	60	0.745	0.7408	0.7008	0.745	0.745	0.69	0.6842	0.745	0.745
10	300	0.8735	0.8953	0.891	0.8735	0.8735	0.8892	0.882	0.8735	0.8735
20	30	0.665	0.665	0.665	0.665	0.665	0.665	0.665	0.665	0.665
20	60	0.7483	0.7242	0.7317	0.7483	0.7483	0.6725	0.6767	0.7483	0.7483
20	300	0.8878	0.9023	0.9023	0.8878	0.8878	0.8282	0.8282	0.8878	0.8878
60	30	0.5	0.5	0.9717	0.5	0.9717	0.5	0.7783	0.5	0.9717
60	60	0.5	0.5	0.9683	0.5	0.9683	0.5	0.6775	0.5	0.9683
60	300	0.8147	0.8033	0.9387	0.8033	0.8147	0.9733	0.7935	0.8033	0.8147
Number of Clusters										
Model Selection			AIC	AIC_Mix	BIC	BIC_Mix	NEC	NEC_Mix	ICLBIC	ICLBIC_Mix
Dimension	Sample Size	Class								
2	30	1	4.8	5.8	1.9	1.3	2.7	3.1	1	3.1
		2	4.8	5.6	1.1	1.6	2.1	2.6	1.1	2.6
2	60	1	5	5	1	1	1.9	1.7	1	1.7
		2	4.6	4.7	1	1	2.7	2.6	1	2.6
2	300	1	2.9	2.7	1	1	1	1	1	1
		2	3.2	3.8	1.1	1.1	1	1	1	1
4	30	1	3.5	4.7	1.2	1	2.3	1	1	1
		2	3.6	4.5	1	1.2	2.8	2	1	2
4	60	1	3.8	4.8	1.1	1	2.7	2.8	1	2.8
		2	4.1	5.3	1	1	3	2.6	1	2.6
4	300	1	3.4	4.7	1	1	1.1	1	1	1
		2	4.3	4.5	1.2	1.2	2.4	2	1	2
10	30	1	1.3	1.6	1	1	1.3	1.7	1	1.7
		2	1.4	1.9	1	1	1.1	1.4	1	1.4
10	60	1	1.1	3.1	1	1	2.1	2.6	1	2.6
		2	1.7	3.7	1	1	2.3	2.7	1	2.7
10	300	1	2.4	2.3	1	1	3.4	4	1	4
		2	3.2	4.5	1	1	2.5	2.7	1	2.7
20	30	1	1	1	1	1	1	1	1	1
		2	1	1	1	1	1	1	1	1
20	60	1	1.3	1.4	1	1	1.4	1.7	1	1.7
		2	1.4	1.4	1	1	1.2	1.5	1	1.5
20	300	1	1.1	1.1	1	1	3.6	3.6	1	1
		2	1.9	1.9	1	1	2.5	2.5	1	1
60	30	1	1	1	1	1	1	3.9	1	1
		2	1	1	1	1	1	5	1	1
60	60	1	1	1	1	1	1	4.1	1	4.1
		2	1	1	1	1	1	3.8	1	3.8
60	300	1	1	4.2	1	1	1	2.5	1	1
		2	1	5.1	1	1	2	2.3	1	1

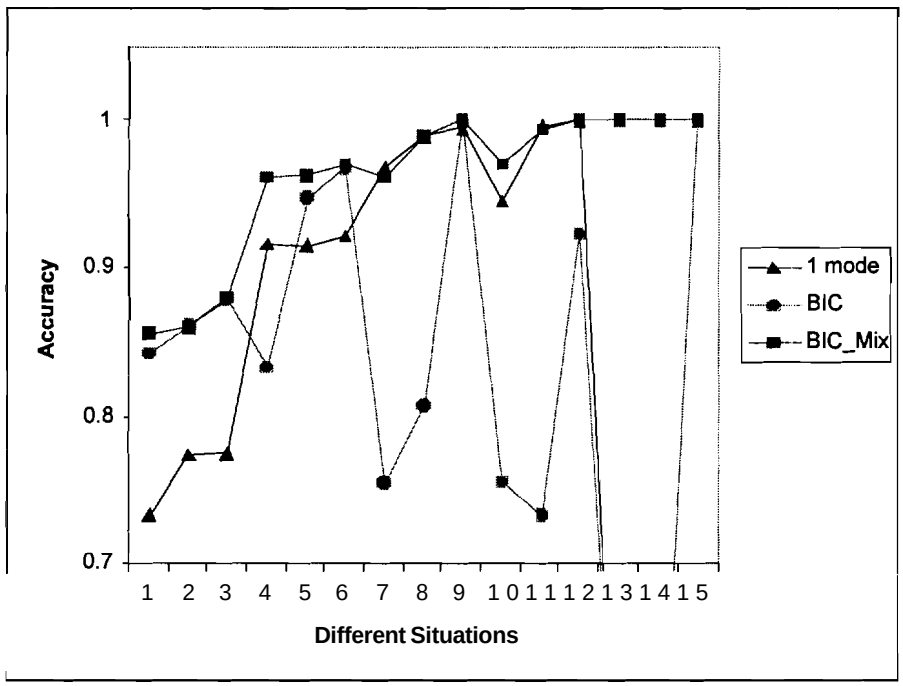


Figure 4.2(a) Some results of experiment 4.1 (I-0.1I case) using NM clustering

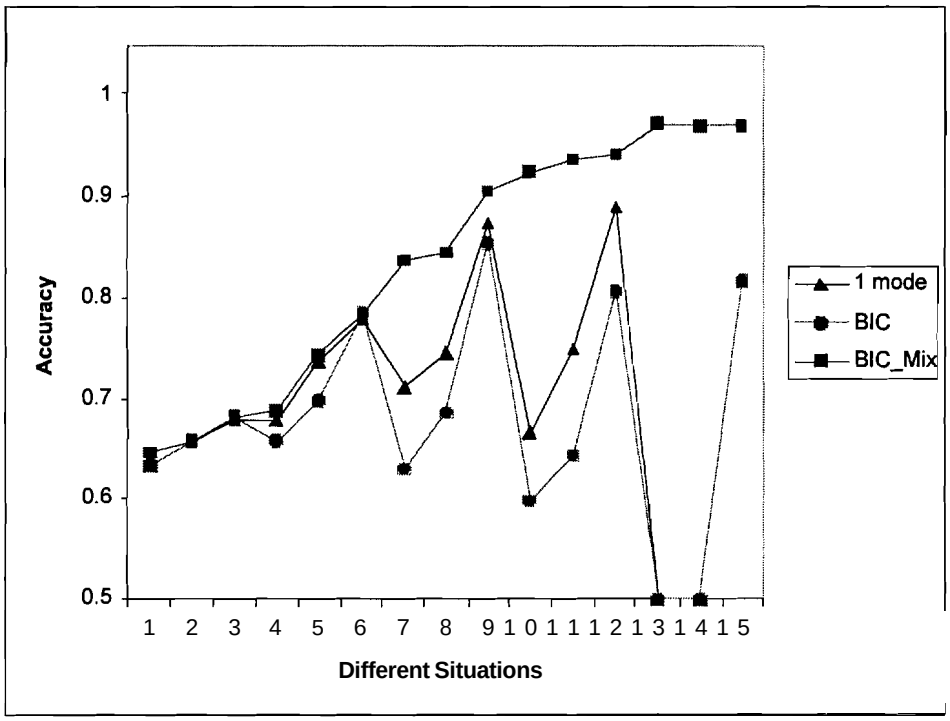


Figure 4.2(b) Some results of experiment 4.2 (I-I case) using NM clustering

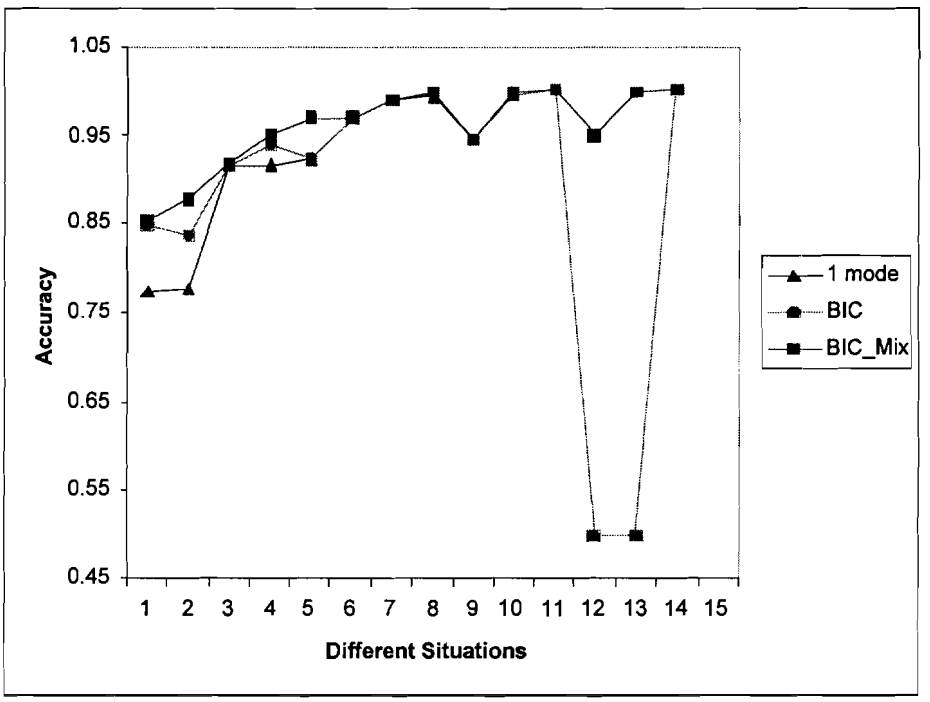


Figure 4.2(c) Some results of experiment 4.3 (I-0.1I case) using EM clustering

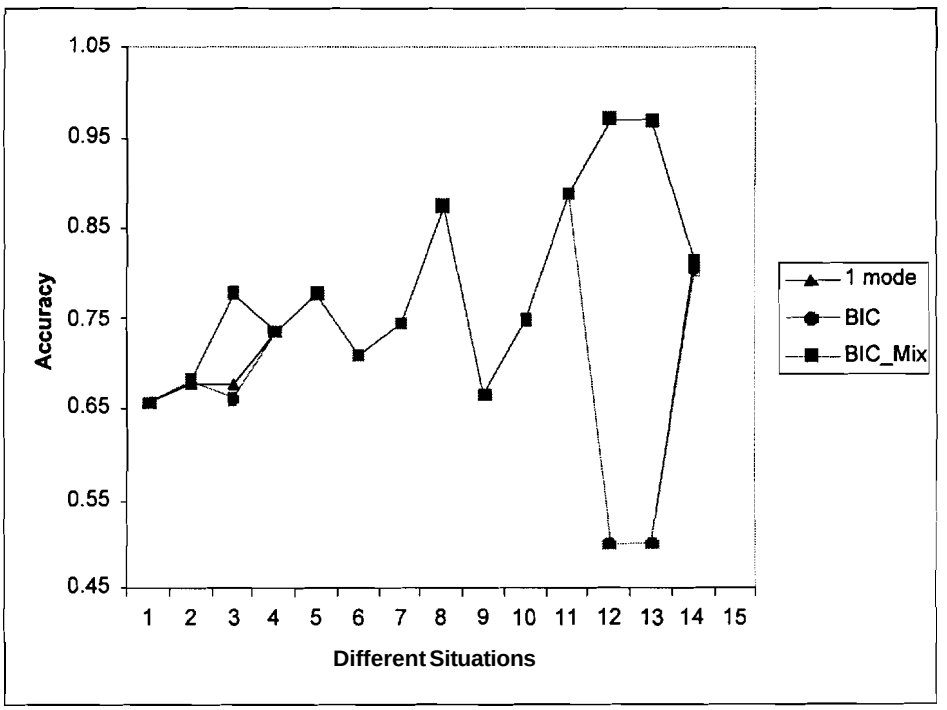


Figure 4.2(d) Some results of experiment 4.4 (I-I case) using EM clustering

Table 4.3 Results of DC Mall real data experiments using NM clustering

Accuracy										
Model Selection		1 mode	AIC	AIC_Mi	BIC	BIC_Mix	NEC	NEC_Mix	ICLBIC	ICLBIC_Mix
Dimensionality	Sample Size									
20	100	0.949	0.9024	0.951	0.925	0.9475	0.7065	0.9199	0.949	0.9312
7	100	0.7394	0.8315	0.8365	0.8314	0.8384	0.7667	0.7503	0.7366	0.7408
20	20	0.4154	0.4154	0.7789	0.4154	0.7789	0.4154	0.6822	0.4154	0.6822
7	20	0.7011	0.6484	0.7163	0.6959	0.7163	0.7011	0.7056	0.7011	0.7056
Number of Components										
Model Selection			AIC	AIC_Mi	BIC	BIC_Mix	NEC	NEC_Mix	ICLBIC	ICLBIC_Mix
Dimensionality	Sample Size	Class								
20	100	1	2.2	2.3	1.1	2.1	3.5	4	1	1
		2	2.6	1.3	1	1.3	4.5	4	1	1
		3	2.7	2.7	1.8	2.6	3.5	5.5	1	1
		4	2.2	2.6	2	2.3	4.5	4	1	1
		5	1.3	1	1	1	3.5	6	1	1
		6	2.9	5.1	1.7	4.3	5	4	1	2.5
		7	2.6	1.9	1	1.7	6	4.5	1	1
20	100	1	2.1	2.3	1	1.3	6	5.5	1	1
		2	4.1	2.4	1	1.2	4.5	6	1	1
		3	3.8	2.7	2.5	2.7	4.5	5.5	1	1
		4	3.2	2.9	2.2	2.7	5.5	5.5	1	1
		5	1	1.1	1	1	5	2	1	1
		6	5.5	4.6	3.2	4.1	5.5	6	2	3.5
		7	5.2	4.5	1.2	4.1	6	6	1	1
39	20	1	1	1.1	1	1.1	1	1	1	1
		2	1	1.2	1	1.2	1	1	1	1
		3	1	1.6	1	1.6	1	1	1	1
		4	1	1.5	1	1.5	1	1	1	1
		5	1	1.6	1	1.6	1	1	1	1
		6	1	2.5	1	2.5	1	1	1	1
		7	1	1.5	1	1.5	1	1	1	1
20	20	1	1.7	1.3	1.1	1.3	1	1	1	1
		2	1.4	1.5	1.1	1.5	1	1	1	1
		3	1.4	1.9	1.2	1.9	1	1	1	1
		4	1.5	1.5	1.3	1.5	1	1	1	1
		5	1.5	1.4	1	1.4	1	1	1	1
		6	1.6	1.9	1.5	1.9	1	1	1	1
		7	1.5	1.3	1.4	1.3	1	1	1	1

CHAPTER 5: Nonparametric Weighted Feature Extraction

5.1 Introduction

Discriminant Analysis Feature Extraction (DAFE, or Linear Discriminant Analysis; LDA) is often used for dimension reduction in classification problems. It is also called the parametric feature extraction method in [8], since DAFE uses the mean vector and covariance matrix of each class. In [20], DAFE is shown to be equivalent to finding the ML estimators of a Gaussian model, assuming that all classes discrimination information resides in the transformed subspace and the within-class distances are equal for all classes. The advantage of DAFE is that it is distribution-free but there are three major disadvantages in DAFE. One is that it works well only if the distributions of classes are normal-like distributions [8]. When the distributions of classes are nonnormal-like or multi-modal mixture distributions, the performance of DAFE is not satisfactory. The second disadvantage of DAFE is the rank of the within-scatter matrix S_b is number of classes (nc) –1, so generally only nc-1 features can be extracted. From [8], we know that unless a posterior probability function is specified, nc–1 features are suboptimal in a Bayes sense, although they are optimal based on the chosen criterion. In real situations, the data distributions are complicated and not normal-like, therefore only using nc-1 features is not sufficient for real data. The third limitation is that if the within-class covariance is singular which usually occurs in high dimensional problems, DAFE will have a poor performance on classification. In this paper, a new nonparametric feature extraction method is developed to solve those problems.

5.2 Previous Works

5.2.1 Discriminant Analysis Feature Extraction (DAFE)

The purpose of feature extraction is to find a transformation matrix A such that the class separability of transformed data (Y) is maximized. A common approach and the one used in DAFE is for within-class, between-class, and mixture scatter matrices to be used to formulate the criteria of class separability. A within-class scatter matrix is expressed by [8]:

$$S_w = \sum_{i=1}^{L} P_i E\{(X - m_i)(X - m_i)^T \mid \omega_i\} = \sum_{i=1}^{L} P_i \Sigma_i \tag{5.1}$$

where P_i means the prior probability of class i, m_i is the class mean and Σ_i is the class covariance matrix. A between-class scatter matrix is expressed as

$$S_b = \sum_{i=1}^{L} P_i (m_i - m_0)(m_i - m_0)^T = \sum_{i=1}^{L-1} \sum_{j=i+1}^{L} P_i P_j (m_i - m_j)(m_i - m_j)^T \tag{5.2}$$

where m_0 represents the expected vector of the mixture distribution and is given by

$$m_0 = E\{X\} = \sum_{i=1}^{L} P_i m_i \tag{5.3}$$

Let Y = AX, then we have

$$S_{wY} = A S_{wX} A^T \quad \text{and} \quad S_{bY} = A S_{bX} A^T \tag{5.4}$$

The optimal features are determined by optimizing the Fisher criteria given by

$$J(A) = tr(S_{wY}^{-1} S_{bY}) \tag{5.5}$$

The optimum **A** must satisfy

$$(S_{wX}^{-1} S_{bX}) A^t = A^t (S_{wY}^{-1} S_{bY}) \tag{5.6}$$

This is a generalized eigenvalue problem **[3]** and usually can be solved by the QZ algorithm.

5.2.2 **aPAC** Linear Dimension Reduction **(aPAC-LDR)**

The approximated Pairwise Accuracy Criterion Linear Dimension Reduction (aPAC-LDR) [21] can be seen as DAFE weighted contributions of individual class pairs according to the Euclidian distance of respective class means. The major difference between DAFE and aPAC-LDR is that the Fisher criteria is redefined as

$$J_{\omega}(A) = \sum_{i=1}^{L-1} \sum_{j=i+1}^{L} P_i P_j \omega(\Delta_{ij}) tr[(AS_{wX}A^T)^{-1}(AS_{ij}A^T)] \ , \tag{5.7}$$

$$\text{where } S_{ij} = (m_i - m_j)(m_i - \text{m,})^{\text{T}} \ , \omega(\Delta_{ij}) = \frac{1}{2\Delta_{ij}} erf(\frac{\Delta_{ij}}{2\sqrt{2}}) \ ,$$

$$\text{and } \Delta_{ij} = \sqrt{(m_i - m_j)^t S_w^{-1}(m_i - \text{m,})} \tag{5.8}$$

The above weighted Fisher criteria is the same as (5.5) by redefining the between-class scatter matrix as

$$S_b = \sum_{i=1}^{L-1} \sum_{j=i+1}^{L} P_i P_j \omega(\Delta_{ij})(m_i - m_j)(m_i - m_j)^T \tag{5.9}$$

Hence the optimization problem is the same as DAFE.

There are one simulated and one real data experiments in [21]. They show that the advantages of this method are

1. It can be designed to confine the influence of outlier classes on the final LDR transformation.

2. aPAC-LDR needs fewer features to reach the optimal accuracy of DAFE, but the best accuracy of aPAC-LDR is almost the same as that of DAFE

aPAC-LDR is the same as DAFE using the mean vector and covariance to formulate the scatter matrix; hence it still suffers from those three major disadvantages of DAFE.

5.2.3 Decision Boundary Feature Extraction (DBFE)

Decision Boundary Feature Extraction (DBFE) [22] is an alternative feature extraction method using boundary information. The following procedure in [22] for the 2-class case has been proposed to determine the transformation needed to find the desired minimal set features (intrinsic discriminant dimensions).

1. Let $\hat{\mu}_i$ and $\hat{\Sigma}_i$ be the estimated mean and covariance of class ω_i. Classify the training samples using full dimensionality. Apply a chi-square threshold test to the correctly classified training samples of each class and delete outliers. In other words, for class ω_i, retain X only if $(X-\hat{\mu}_i)^t \hat{\Sigma}_i^{-1}(X-\hat{\mu}_i) < R_{t1}$. In the following steps, only correctly classified training samples that passed the chi-square threshold test will be used. Let $\{X_1, X_2, \ldots X_{L1}\}$ be such training samples of class ω_1 and $\{Y,, Y_2, \ldots Y_{L2}\}$ be such training samples of class ω_2.

2. Apply a chi-square threshold test of class ω_1 to the samples of class ω_2 and retain Y_j only if $(Y-\hat{\mu}_1)^t \hat{\Sigma}_1^{-1}(Y-\hat{\mu}_1) < R_{t2}$. If the number of the samples of class ω_2 which pass the chi-square threshold test is less than L_{min}, retain the L_{min} samples of class ω_2 that give the smallest values.

3. For X_i of class ω_1, find the nearest samples of class ω_2 retained in STEP2.

4. Find the point P_i where the straight line connecting the pair of the samples found in STEP 3 meets the decision boundary.

5. Find the unit normal vector, N_i, to the decision boundary that can be calculated based on training samples at the point P_i found in STEP 4.

6. By repeating STEP 3 Through STEP 5 for X_i, I=1,..., L,, L, unit normal vectors will be calculated. From the normal vectors, calculate an estimate of the effective decision boundary feature matrix from class ω_1 as follows:

$$\Sigma_{EDBFM}^{1} = \frac{1}{L_1}\sum_{i=1}^{L_1} N_i N_i^t$$

Repeat STEP 2 through STEP *6* for class ω_2.

7. Calculate an estimate of the final effective decision boundary feature matrix as follows:

$$\Sigma_{EDBFM} = \Sigma^{(1,2)}_{EDBFM} = \frac{1}{2}(\Sigma^{1}_{EDBFM} + \Sigma^{2}_{EDBFM})$$

For multiple classes problem,

$$\Sigma_{EDBFM} = \sum_{i=1}^{nc} \sum_{\substack{j=1 \\ j \neq i}}^{nc} P_i P_j \Sigma^{(i,j)}_{EDBFM}$$

After EDBFM is estimated, the intrinsic discriminant dimension can be estimated and the new features can be extracted to achieve the full accuracy at the subspace spanned by these features.

There are a few advantages of DBFE. First, it focuses directly on classification accuracy rather that a surrogate to it. Second, it shows directly how many features are needed to achieve full accuracy and it provides evidence as to which original features were the most important. Finally, it is able to directly treat the problem of outliers. However, there are some shortcomings of this approach. First, it demands a large number of training samples to perform well, which is unfortunately limited in most of practical applications. When the training samples size is not large enough, the performance of DAFE is frequently a little better than that of DBFE. Second, L_{min} is usually decided by "trial and error". Finally, LDBFE needs much computational time.

5.2.4 Nonparametric Discriminant Analysis (NDA; [8],[23])

Nonpararnetric Discriminant Analysis (NDA) is proposed to solve the problems of DAFE. In NDA, the between-class scatter matrix is redefined as a new nonparametric between-class scatter matrix (for the 2 classes problem), denoted S_b , as

$$S_b = P_1 E\{(X^{(1)} - M_2(X^{(1)}))(X^{(1)} - M_2(X^{(1)}))^t \mid \omega_1\}$$

$$+ P_2 E\{(X^{(2)} - M_1(X^{(2)}))(X^{(2)} - M_1(X^{(2)}))^t \mid \omega_2\} \qquad (5.10)$$

where $M_i(X_\ell) = \frac{1}{k}\sum_{j=1}^{k} X_{jNN}^{(i)}$ is called the local kNN mean, $X_{jNN}^{(i)}$ is the jth the nearest neighborhood (NN) from ω_i to the sample X_ℓ, and $X^{(i)}$ refers to samples from class i (ω_i). If $k = N_i$, [8] shows that the features extracted by maximizing $\mathrm{tr}(S_w^{-1}S_b)$ must be the same as the ones from $\mathrm{tr}(S_w^{-1}S_b)$. Thus, the parametric feature extraction obtained by maxirnizing $\mathrm{tr}(S_w^{-1}S_b)$ is a special case of feature extraction with the more general nonparametric criterion $\mathrm{tr}(S_w^{-1}S_b)$.

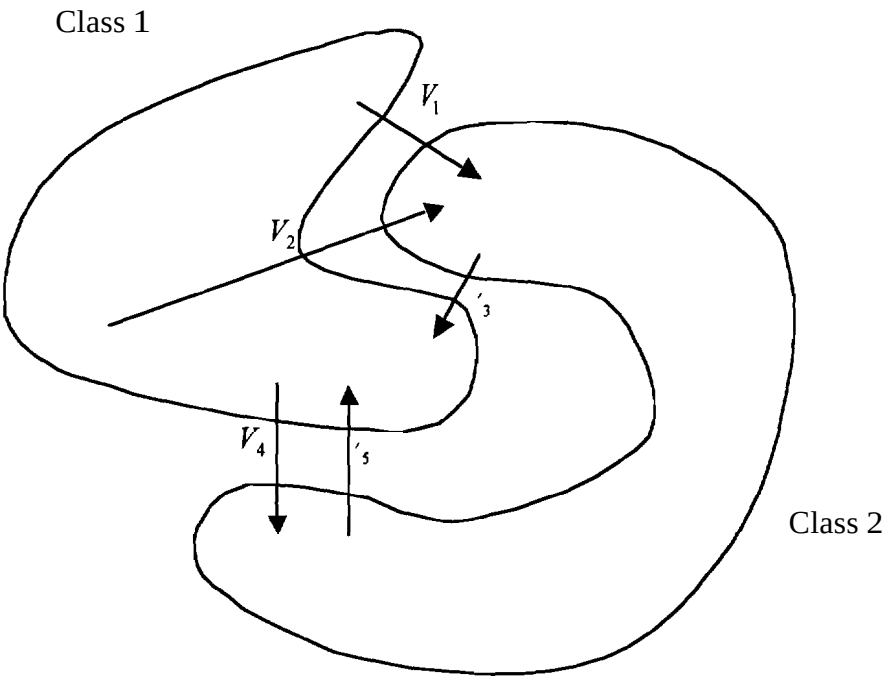


Figure 5.1 The relationship between sample points and their local means.

Further understanding of S_b is obtained by examining the vector $(X_\ell - M_i(X_\ell))$. Figure 5.1 shows the importance of using boundary points and local means. Pointing to the local mean from the other class, each vector indicates the direction to the other class locally. If we select these vectors only from the samples located in the classification boundary (V_1, V_3, V_4, V_5), the scatter matrix of these vectors should specify the subspace in

which the boundary region is embedded. Samples that are far away from the boundary (V_2) tend to have large magnitudes.

These large magnitudes can exert a considerable influence on the scatter matrix and distort the information of the boundary structure. Therefore, some method of de-emphasizing samples far from the boundary seems appropriate. To accomplish this, [8] uses a weighting function for each ($X_\ell - M_i(X_\ell)$). The value of the weighting function, denoted as w_ℓ, for X_ℓ is defined as

$$w_\ell = \frac{\min\{d^\alpha(X_\ell, X_{kNN}^{(1)}), d^\alpha(X_\ell, X_{kNN}^{(2)})\}}{d^\alpha(X_\ell, X_{kNN}^{(1)}) + d^\alpha(X_\ell, X_{kNN}^{(2)})}, \tag{5.11}$$

where a is a control parameter between zero and infinity, and $d(X_i, X_{kNN}^{(i)})$ is the distance from X_ℓ to its kNN from w_ℓ.

The final discrete form for S_b is expressed by

$$S_b = \frac{1}{N}\sum_{\ell=1}^{N_1} w_\ell (X_\ell^{(1)} - M_2(X_\ell^{(1)}))(X_\ell^{(1)} - M_2(X_\ell^{(1)}))^T$$

$$+ \frac{1}{N}\sum_{\ell=1}^{N_2} w_\ell (X_\ell^{(2)} - M_2(X_\ell^{(2)}))(X_\ell^{(2)} - M_2(X_\ell^{(2)}))^T . \tag{5.12}$$

where $N = N_1 + N_2$, and the expectations of ***(5.10)*** are replaced by the sample means and P_i by N_i / N

The disadvantages of NDA are

1. Parameters ***k*** and a are usually decided by rules of thumb. So the better result usually comes after several trails.
2. S_w is still with a parametric form. When the training set size is small, NDA will have the singularity problem.

For solving the above problems, a new feature extraction method is proposed below

5.3 Nonparametric Weighted Feature Extraction (NWFE)

In this section, a new feature extraction method called nonparametric weighted feature extraction (NWFE) is proposed. From NDA (and from DBFE), we know that the "local information" is important and useful for improving DAFE. The main ideas of NWFE are putting different weights on every sample to compute the "local means" and defining new nonparametric between-class and within-class scatter matrices to get more features. In NWFE, the nonparametric between-class scatter matrix is defined as

$$S_b = \sum_{i=1}^{nc} \frac{P_i}{nc-1} \sum_{\substack{j=1 \\ j \neq i}}^{nc} \sum_{k=1}^{n_i} \lambda_k^{(i,j)} (x_k^{(i)} - M_j(x_k^{(i)}))(x_k^{(i)} - M_j(x_k^{(i)}))^T \tag{5.13}$$

where $x_k^{(i)}$ refers to the k-th sample from class i. The scatter matrix weight $\lambda_k^{(i,j)}$ is defined as:

$$\lambda_k^{(i,j)} = \frac{dist(x_k^{(i)}, M_j(x_k^{(i)}))^{-1}}{\sum_{l=1}^{n_j} dist(x_l^{(i)}, M_j(x_l^{(i)}))^{-1}}, \tag{5.14}$$

where $dist(a,b)$ means the distance from a to b.

and $M_j(x_k^{(i)})$ is the local mean of $x_k^{(i)}$ in the class j and defined as:

$$M_j(x_k^{(i)}) = \sum_{l=1}^{n_j} w_l^{(i,j)}(x_k^{(i)}) x_l^{(j)} , \tag{5.15}$$

where $w_l^{(i,j)}(x_k^{(i)}) = \frac{dist(x_k^{(i)}, x_l^{(j)})^{-1}}{\sum_{l=1}^{n_j} dist(x_k^{(i)}, x_l^{(j)})^{-1}}$. (5.16)

The nonparametric between-class scatter matrix is defined as

$$S_w = \sum_{i=1}^{nc} P_i \sum_{k=1}^{n_i} \lambda_k^{(i,i)} (x_k^{(i)} - M_i(x_k^{(i)}))(x_k^{(i)} - M_i(x_k^{(i)}))^T \tag{5.17}$$

The optimal features are determined by optimizing the criteria given by

$$J = tr(S_w^{-1}S_b)$$

To reduce the effect of the cross products of between-class distances and prevent the singularity, we will replace S_w by

$$S_w = 0.5S_w + 0.5diag(S_w)$$

Finally the NWFE algorithm is

1. Compute the distances between each pair of sample points and form the distance matrix.
2. Compute $w_l^{(i,j)}$ using the distance matrix
3. Use $w_l^{(i,j)}$ to compute local means $M_j(x_k^{(i)})$
4. Compute scatter matrix weight $\alpha_k^{(i,j)}$.
5. Compute S_b and S_w .
6. Select the m eigenvectors of $S_w^{-1}S_b$, $\psi_1,\psi_2,\cdots,\psi_m$, which correspond to the m largest eigenvalues to form the transformation matrix $A_m = [\psi_1,\psi_2,\cdots,\psi_m]$

5.4 Simulated and Real Data Experiments

In this section, the simulated and real data set performances of four methods, DAFE, NWFE, aPAC-LDR, and NDA using 1NN and 5NN based on the α=2 , will be compared under several experiment designs.

5.4.1 Simulation Data Experiment Design

Two different designs (experiments 5.1 and 5.2), and three different dimensionality cases (30, 60, 120) are tested. One is that all *6* classes are distributed with normal distributions whose covariance matrices are the same but mean vectors are different. The

other is that all 6 classes are distributed with mixture normal distributions and each class contains two normally distributed components. Their mean vectors, covariance matrices, training and testing sample sizes are in Tables 5.1(a) and 5.2(b). At each situation, 10 random training and testing data sets are generated for computing the accuracies of algorithms, and the standard deviations of the accuracies.

Table 5.1(a) Design of Experiment 5.1 for normal distributions

Dim=30, 60, 120	class 1	class 2	class 3	class 4	class 5	class 6
Mean Vector	[0,…,0]	[1,0,…,0]	[0,1,0,…,0]	[0,0,1,0,…,0]	[1,1,0,…,0]	[1,0,1,0,…,0]
Covariance	0.1I					
Training Sample Size	40	40	40	40	40	40
Testing Sample Size	400	400	400	400	400	400

Table 5.1(b) Design of Experiment 5.2 for mixture distributions

	class 1		class 2		class 3	
Dim=30, 60, 120	component 1	component 2	component 1	component 2	component 1	component 2
Mean Vector	[2,2,0,…,0]	[0,0,…,0]	[2,4,…,0]	[4,-2,0,…,0]	[-2,0,…,0]	[6,0,…,0]
Covariance	0.1I					
Training Sample Size	20	20	20	20	20	20
Testing Sample Size	200	200	200	200	200	200
	class 4		class 5		class 6	
dim=30, 120	component 1	component 2	component 1	component 2	component 1	component 2
Mean Vector	[-2,-2,0,…,0]	[0,6,…,0]	[2,-4,…,0]	[-4,2,0,…,0]	[2,0,…,0]	[-6,0,…,0]
Covariance	0.11					
Training Sample Size	20	20	20	20	20	20
Testing Sample Size	200	200	200	200	200	200

5.4.2 Real Data Experiment Design

There are four different real data sets, Cuprite, which is a site of geologic interest in western Nevada, Jasper Ridge, a site of ecological interest in California, Indian Pine, a mixed forest/agricultural site, and DC Mall, an urban site, in experiment 5.3. There are 8, 6, 6, and 7 classes in Cuprite, Jasper Ridge, Indian Pine, and DC Mall data sets respectively. There are 40 training samples in each class of Cuprite, Jasper Ridge, and Indian Pine experiments, and 50 training samples in the DC Mall experiments. At each experiment, 10 training and testing data sets are selected for computing the testing sample accuracies of algorithms, and the standard deviations of the accuracies.

5.5 Experiment Results

5.5.1 Simulation Experiment Results

The results of experiment 5.1 are displayed in tables 5.2(a), (b), (c), and figures 5.2(a), (b), and (c). The results of experiment 5.2 are displayed in Table 5.3(a), (b), (c), and Figures 5.3(a), (b), and (c). They show that

1. NWFE performs better than the other methods uniformly in both experiments.

2. The differences between NWFE and the other methods increase as the dimensionality of original space increases. And the increasing dimensionality of original space has only a small impact on accuracy of NWFE.

3. When the number of extracted features is greater than nc-1, the performances of DAFE and aPAC-LDR decease rapidly, but NWFE and NDA do not.

4. In mixture distribution data, NWFE is much better than the other methods whether the dimensionality is large or not.

5. Figure 5.3(c) shows that nc-1 features may not be a best choice. Using NWFE, more features can be extracted, and better results are obtained.

Table 5.2(a) Mean and standard deviation of accuracies (normal and dim=30)

	DAFE		NWFE		aPAC_LDR		NDA_1NN		NDA_5NN	
Features	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	0.4545	0.0124	0.4543	0.0436	0.4477	0.0278	0.4025	0.0085	0.4005	0.0082
2	0.7253	0.0138	0.7493	0.0117	0.7212	0.0272	0.7033	0.02	0.6973	0.0185
3	0.8234	0.0124	0.8387	0.0053	0.8235	0.0081	0.8124	0.0052	0.8078	0.007
4	0.8107	0.0169	0.8318	0.0057	0.8098	0.0074	0.8002	0.0097	0.7951	0.0117
5	0.8019	0.0167	0.8264	0.0075	0.8019	0.0064	0.7919	0.0124	0.7881	0.0134
6	0.5603	0.3973	0.8213	0.0089	0.403	0.4147	0.784	0.0088	0.7794	0.0119
7	0.1762	0.2305	0.8144	0.0077	0.1791	0.3222	0.7712	0.0095	0.7692	0.0156
8	0.1257	0.0336	0.807	0.0075	0.1275	0.2332	0.7638	0.0097	0.7638	0.0114
9	0.1618	0.0452	0.8006	0.0088	0.0937	0.0547	0.7518	0.0095	0.7517	0.0116
10	0.1487	0.0486	0.7907	0.0097	0.1301	0.0549	0.74	0.0078	0.7393	0.0083
11	0.2033	0.0455	0.779	0.0135	0.1775	0.0727	0.7303	0.0107	0.7304	0.0087
12	0.2497	0.0414	0.7656	0.0132	0.2233	0.0782	0.7186	0.0111	0.7199	0.012
13	0.2894	0.0428	0.7542	0.0165	0.2638	0.0873	0.7083	0.0143	0.7058	0.0109
14	0.3128	0.0424	0.7405	0.0164	0.2897	0.0955	0.696	0.0148	0.6954	0.0113
15	0.3214	0.0489	0.7215	0.0177	0.3065	0.0842	0.6818	0.0128	0.6833	0.0124

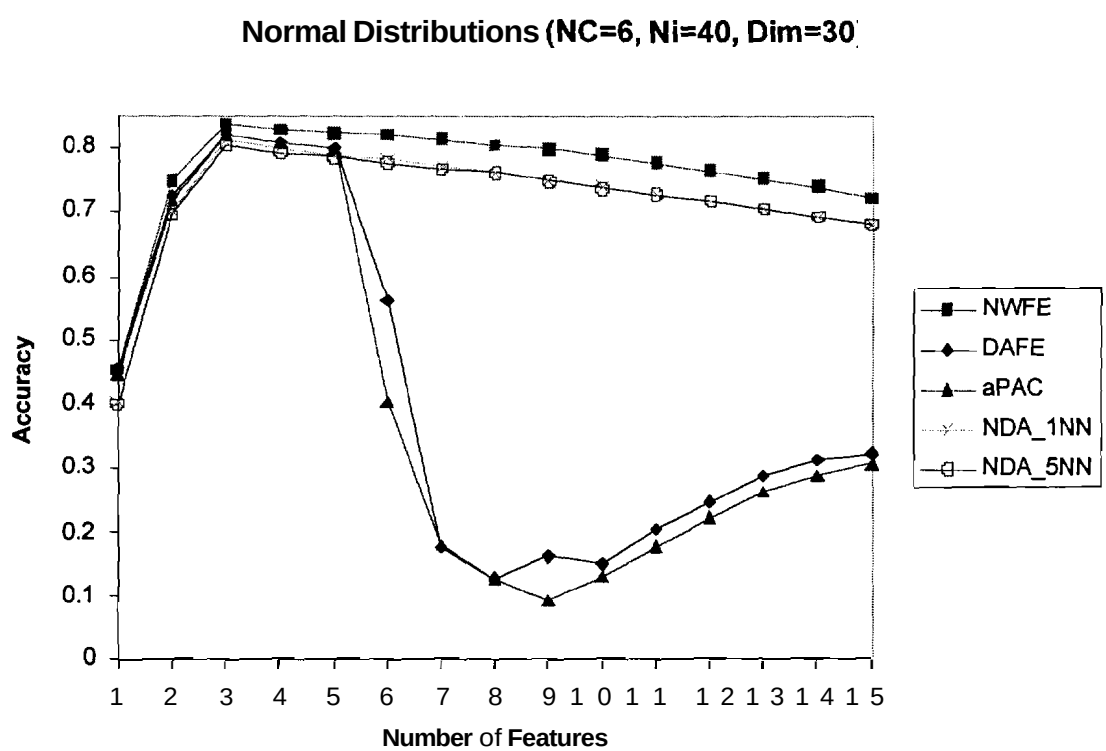


Figure 5.2(a) Mean of accuracies using 1–15 features (normal and dim=30).

Table 5.2(b) Mean and standard deviation of accuracies (normal and dim=60)

	DAFE		NWFE		aPAC_LDR		NDA_1NN		NDA_5NN	
Features	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	0.4309	0.0257	0.4538	0.0259	0.4306	0.0215	0.3789	0.0118	0.3751	0.0123
2	0.6875	0.0137	0.7379	0.0106	0.6848	0.0167	0.6501	0.0261	0.6426	0.0258
3	0.7622	0.0164	0.8155	0.0104	0.7618	0.0166	0.7541	0.0175	0.7457	0.0175
4	0.7464	0.0195	0.8055	0.0155	0.7457	0.02	0.7377	0.0199	0.7318	0.022
5	0.7295	0.0216	0.795	0.0169	0.7295	0.0216	0.724	0.0211	0.7193	0.0241
6	0.4442	0.3775	0.7907	0.0197	0.5767	0.3021	0.7112	0.0236	0.7077	0.0238
7	0.2953	0.3642	0.7882	0.0204	0.2866	0.3508	0.7049	0.0273	0.6993	0.0286
8	0.095	0.2103	0.7815	0.0198	0.165	0.2732	0.6953	0.0269	0.692	0.0272
9	0.1206	0.2016	0.7745	0.0217	0.0529	0.0395	0.6855	0.0257	0.6838	0.03
10	0.0831	0.0444	0.7666	0.0197	0.088	0.053	0.6766	0.0295	0.671	0.0325
11	0.1134	0.0496	0.7595	0.0203	0.118	0.0712	0.6708	0.0297	0.6658	0.0301
12	0.1472	0.0594	0.7514	0.0234	0.144	0.0735	0.6581	0.0304	0.6549	0.0293
13	0.173	0.0658	0.7419	0.0226	0.1663	0.0649	0.6471	0.0296	0.6452	0.0298
14	0.1955	0.0677	0.7313	0.0232	0.1857	0.0634	0.6355	0.0322	0.6363	0.0365
15	0.2074	0.055	0.7181	0.0265	0.209	0.0547	0.6245	0.0306	0.6233	0.0333

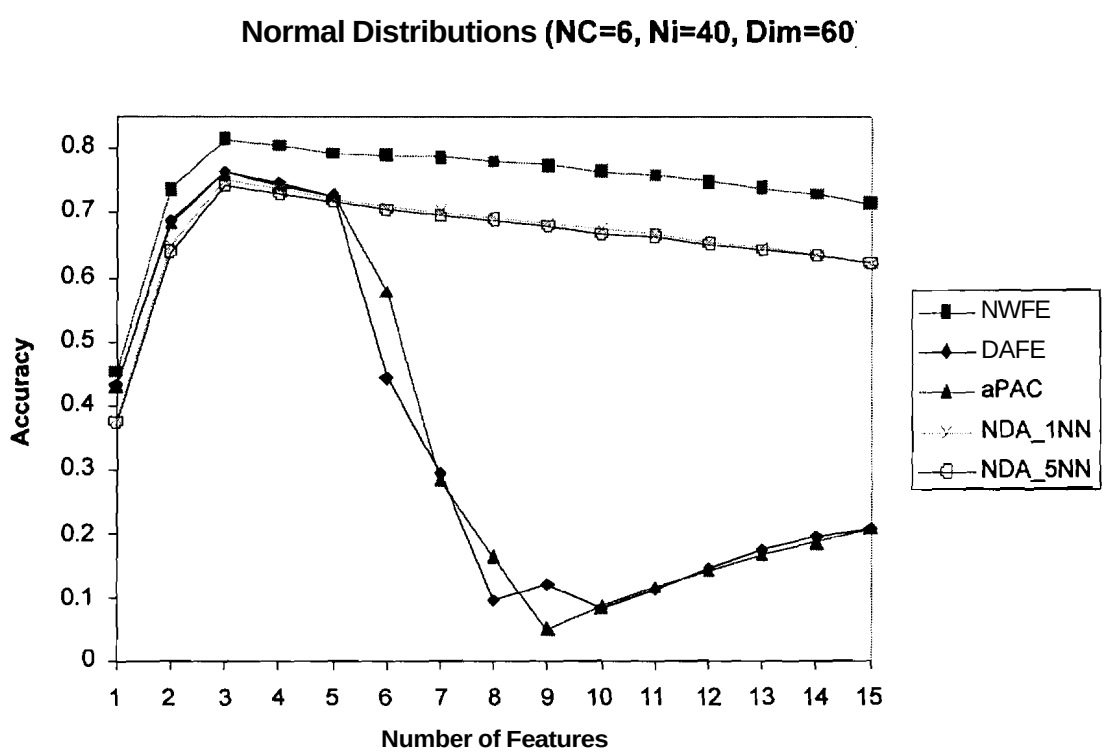


Figure 5.2(b) Mean of accuracies using 1–15 features (normal and dim=60).

Table 5.2(c) Mean and standard deviation of accuracies (normal and dim=120)

	DAFE		NWFE		aPAC_LDR		NDA_1NN		NDA_5NN	
Features	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	0.3554	0.027	0.4091	0.0447	0.3548	0.028	0.3377	0.0204	0.3367	0.0186
2	0.5623	0.0134	0.702	0.0143	0.5615	0.0129	0.5326	0.0335	0.5304	0.0236
3	0.635	0.0169	0.786	0.0073	0.635	0.0176	0.628	0.0203	0.623	0.0218
4	0.6077	0.016	0.7617	0.0114	0.6085	0.0157	0.6054	0.0209	0.6018	0.0198
5	0.5859	0.0197	0.7493	0.011	0.5859	0.0197	0.5892	0.025	0.5876	0.0256
6	0.2957	0.2962	0.7474	0.0109	0.4126	0.2748	0.581	0.0248	0.582	0.0245
7	0.0748	0.1808	0.745	0.0119	0.1828	0.2674	0.5741	0.0223	0.5718	0.0242
8	0.017	0.006	0.7421	0.0122	0.0173	0.0045	0.5689	0.0225	0.5614	0.0255
9	0.0201	0.0063	0.7379	0.0125	0.019	0.0056	0.5628	0.0246	0.5529	0.0244
10	0.024	0.0077	0.7342	0.0125	0.0233	0.0073	0.5581	0.0261	0.5455	0.0258
11	0.0321	0.0124	0.7312	0.012	0.0283	0.0102	0.5512	0.0257	0.5397	0.0277
12	0.0373	0.0095	0.7263	0.0126	0.0342	0.0127	0.5442	0.0268	0.5282	0.0293
13	0.0455	0.0132	0.7225	0.0102	0.0422	0.0147	0.5384	0.0248	0.5207	0.0308
14	0.0515	0.0133	0.7155	0.011	0.0494	0.015	0.5302	0.0251	0.5144	0.0304
15	0.062	0.0137	0.7109	0.0115	0.055	0.0139	0.5256	0.0258	0.5035	0.0326

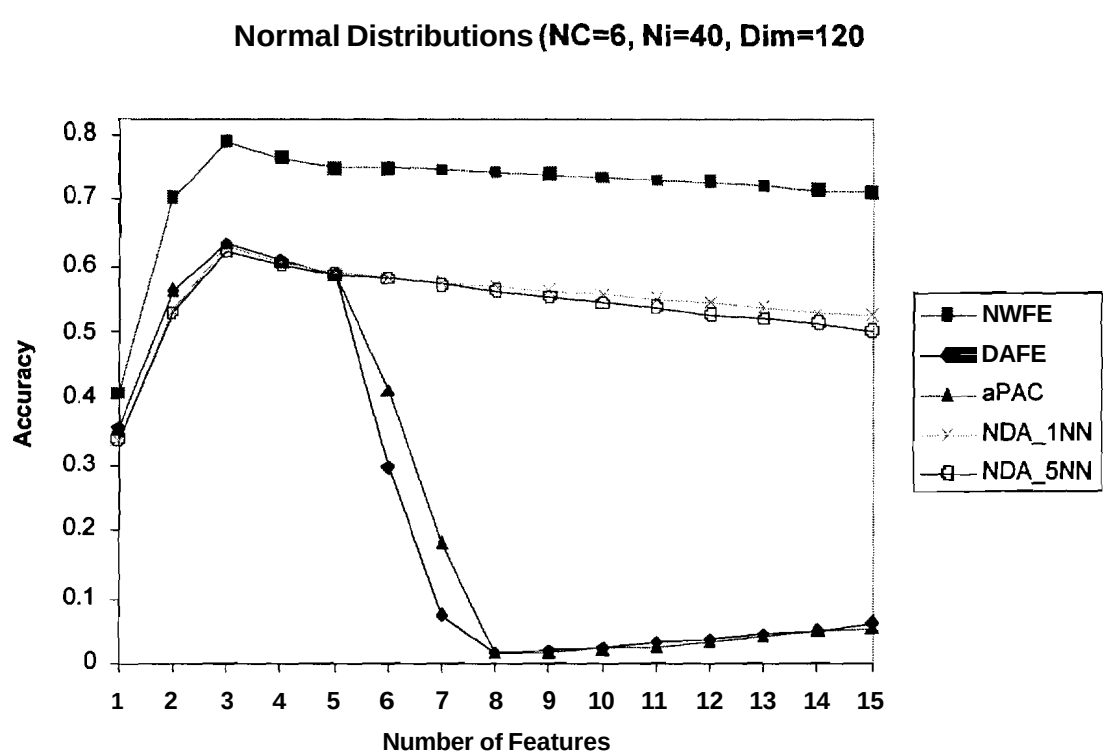


Figure 5.2(c) Mean of accuracies using 1–15 features (normal and dim=120).

Table 5.3(a) Mean and standard deviation of accuracies (mixture and dim=30)

	DAFE		NWFE		aPAC_LDR		NDA_1NN		NDA_5NN	
Features	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	0.3648	0.0206	0.489	0.006	0.3427	0.0297	0.3621	0.0225	0.3388	0.0259
2	0.3924	0.0196	0.7936	0.0088	0.351	0.0386	0.3977	0.0469	0.3525	0.0441
3	0.3755	0.0313	0.7979	0.0147	0.3556	0.0335	0.4258	0.0606	0.3685	0.0348
4	0.362	0.0319	0.7936	0.0124	0.3561	0.0379	0.4892	0.0414	0.4066	0.0434
5	0.3688	0.0195	0.7854	0.0142	0.3688	0.0195	0.5201	0.0318	0.427	0.0284
6	0.3527	0.1685	0.7811	0.0125	0.2802	0.1855	0.5314	0.0271	0.4523	0.0276
7	0.21	0.2063	0.7736	0.0133	0.0828	0.0279	0.5531	0.0228	0.4625	0.0284
8	0.1879	0.2041	0.7701	0.0129	0.0985	0.0353	0.567	0.0327	0.4727	0.0172
9	0.068	0.0391	0.7616	0.0137	0.1091	0.0357	0.5761	0.0325	0.4655	0.0208
10	0.0688	0.0399	0.7571	0.0117	0.0999	0.0373	0.5731	0.0303	0.4698	0.0266
11	0.0685	0.0409	0.7542	0.013	0.1017	0.0373	0.5674	0.0314	0.4781	0.0312
12	0.0656	0.0328	0.7478	0.0118	0.1057	0.0311	0.5753	0.0271	0.4795	0.0251
13	0.0675	0.036	0.7395	0.0118	0.1	0.0306	0.58	0.0273	0.4787	0.0253
14	0.0634	0.0404	0.7311	0.013	0.0979	0.0285	0.5785	0.0283	0.4768	0.0284
15	0.0577	0.0387	0.7223	0.0133	0.0948	0.0284	0.584	0.0316	0.4758	0.0284

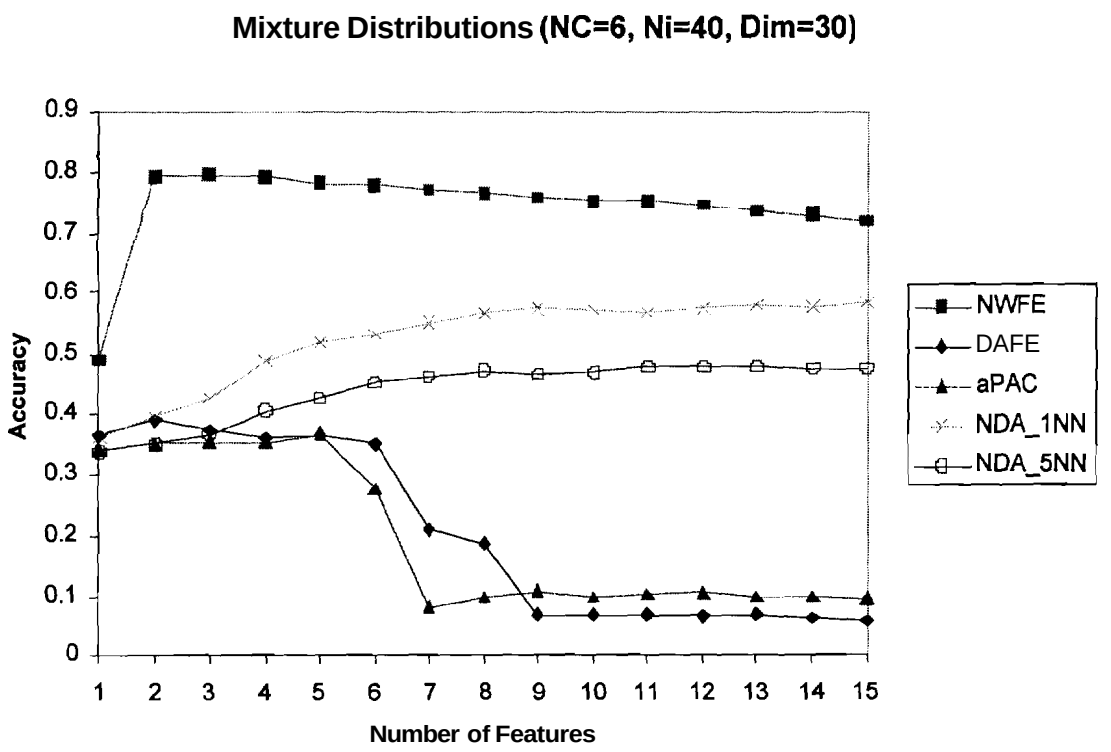


Figure 5.3(a) Mean of accuracies using 1–15 features (mixture and dim=30).

Table 5.3(b) Mean and standard deviation of accuracies (mixture and dim=60)

	DAFE		NWFE		aPAC_LDR		NDA_1NN		NDA_5NN	
Features	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	0.2782	0.0236	0.456	0.0138	0.2233	0.0349	0.2749	0.0203	0.2511	0.0192
2	0.2842	0.016	0.6971	0.0234	0.2457	0.025	0.2547	0.0163	0.2479	0.0109
3	0.2826	0.0107	0.69	0.0236	0.2697	0.0051	0.2551	0.0181	0.2479	0.0125
4	0.2782	0.0118	0.7034	0.022	0.2728	0.0115	0.2643	0.0098	0.2515	0.0087
5	0.2715	0.0131	0.7087	0.0163	0.2715	0.0131	0.2699	0.0126	0.2526	0.0053
6	0.2494	0.0945	0.7129	0.0164	0.2432	0.103	0.3165	0.0315	0.2756	0.0221
7	0.1823	0.1121	0.7104	0.0172	0.1962	0.1144	0.3587	0.0206	0.3053	0.0139
8	0.1366	0.0782	0.7095	0.0195	0.1839	0.1075	0.4022	0.0331	0.3118	0.0206
9	0.115	0.0341	0.7088	0.0192	0.1253	0.0186	0.4147	0.0319	0.323	0.0227
10	0.125	0.0299	0.7046	0.0204	0.1353	0.0178	0.4249	0.0279	0.3274	0.024
11	0.1291	0.0285	0.6999	0.0204	0.144	0.0196	0.4372	0.0219	0.3279	0.0241
12	0.123	0.0325	0.6952	0.0224	0.1513	0.0229	0.4299	0.0181	0.3295	0.0209
13	0.1257	0.0366	0.6894	0.0224	0.1505	0.0283	0.4266	0.0192	0.3316	0.0244
14	0.1245	0.0336	0.6878	0.0243	0.1542	0.0304	0.4257	0.0207	0.3267	0.0234
15	0.1203	0.0363	0.6854	0.0224	0.155	0.0283	0.4234	0.0149	0.3305	0.0168

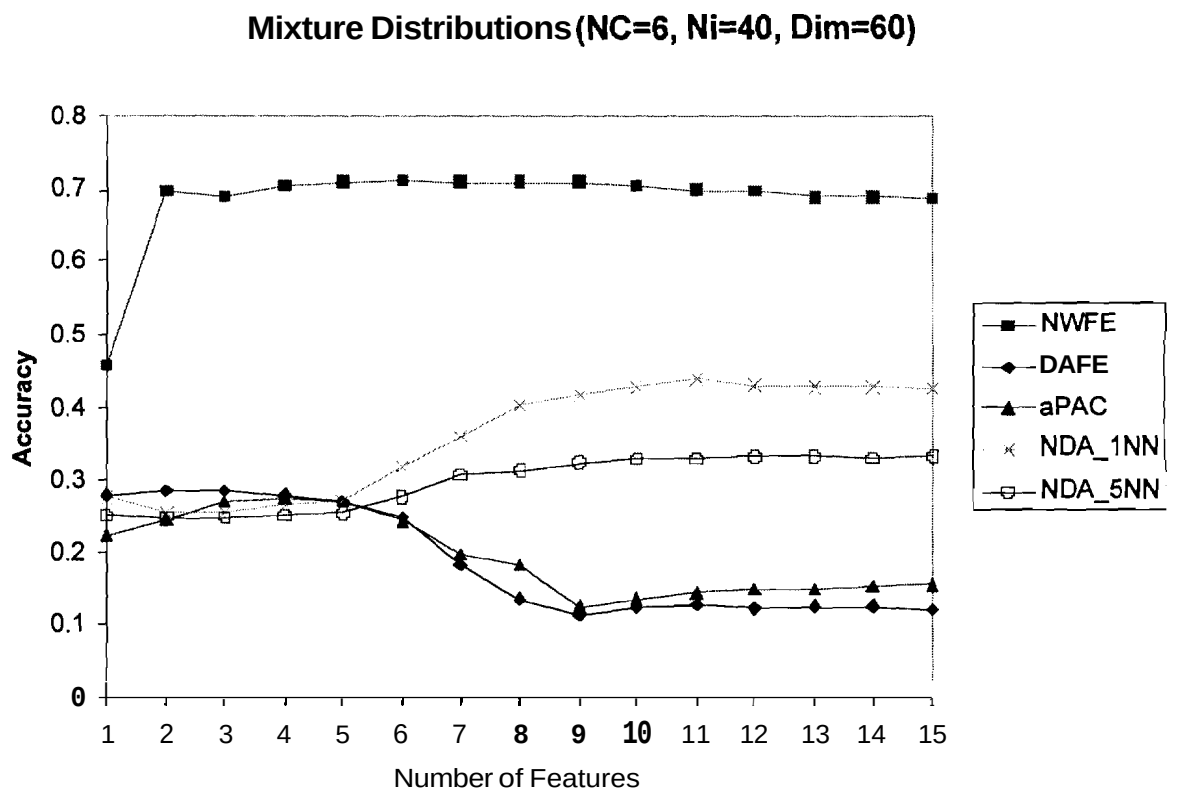


Figure 5.3(b) Mean of accuracies using 1–15 features (mixture and dim=30).

Table 5.3(c) Mean and standard deviation of accuracies (mixture and dim=120)

	DAFE		NWFE		aPAC_LDR		NDA_1NN		NDA_5NN	
Features	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	0.2238	0.0151	0.3254	0.0173	0.1879	0.024	0.2133	0.0159	0.2078	0.0139
2	0.2243	0.0107	0.3534	0.0354	0.209	0.0171	0.2122	0.0146	0.2062	0.0172
3	0.228	0.0136	0.36	0.0245	0.2152	0.0113	0.2113	0.0138	0.2098	0.0167
4	0.2278	0.0044	0.3529	0.0205	0.2227	0.0083	0.2084	0.0112	0.2059	0.0121
5	0.225	0.0067	0.3647	0.0114	0.225	0.0067	0.2099	0.0089	0.2052	0.0134
6	0.1648	0.0602	0.4303	0.0191	0.2084	0.0505	0.2085	0.0054	0.2045	0.0074
7	0.157	0.0615	0.4728	0.027	0.1467	0.0602	0.2462	0.0085	0.224	0.0155
8	0.1294	0.0407	0.4805	0.0251	0.1207	0.015	0.2677	0.0245	0.2305	0.0138
9	0.1147	0.0113	0.4808	0.0262	0.1243	0.0165	0.2815	0.0209	0.2328	0.0117
10	0.1174	0.0094	0.4814	0.0273	0.1281	0.0164	0.2988	0.0177	0.2372	0.0155
11	0.1141	0.0126	0.4822	0.0271	0.1275	0.0194	0.3084	0.0162	0.2454	0.0177
12	0.1164	0.0124	0.4803	0.028	0.1298	0.0179	0.3087	0.0114	0.2465	0.015
13	0.1194	0.012	0.4822	0.0264	0.1333	0.0214	0.3082	0.015	0.2438	0.0166
14	0.1185	0.0131	0.477	0.0204	0.1317	0.0237	0.3024	0.0157	0.2471	0.015
15	0.1203	0.0158	0.4763	0.0221	0.1362	0.0234	0.303	0.02	0.2441	0.0148

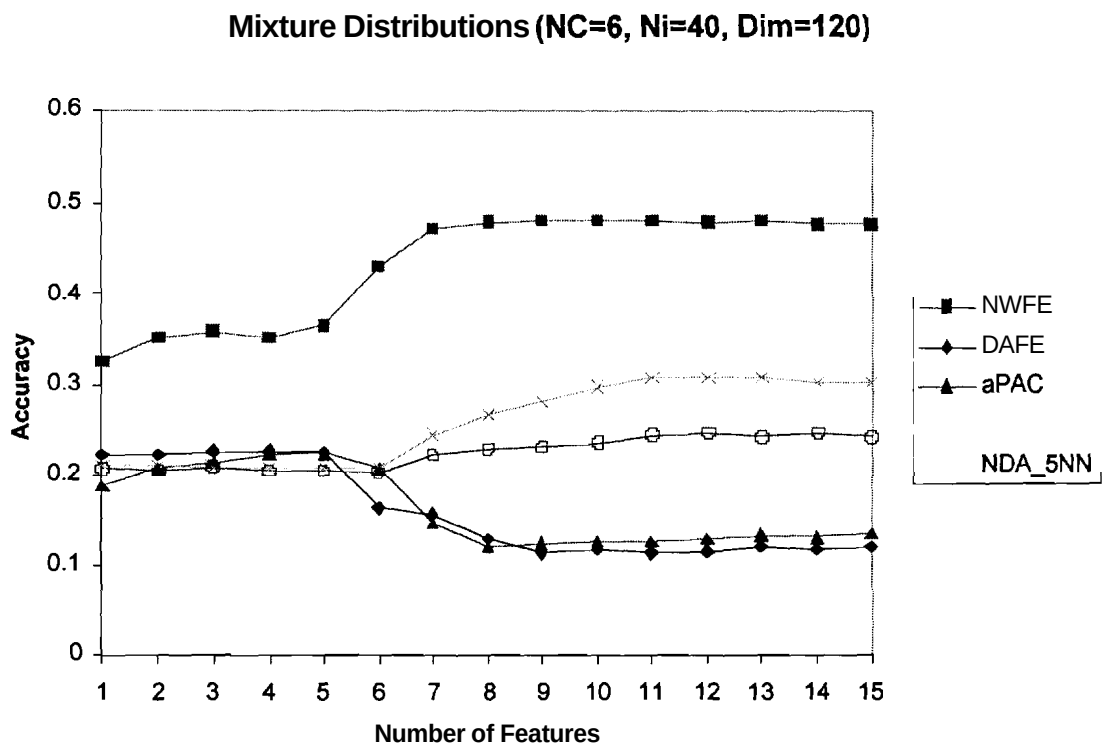


Figure 5.3(c) Mean of accuracies using 1–15 features (mixture and dim=120).

5.5.2 Real Data Experiment Results

The results of experiment 3 are displayed in tables 5.4(a), (b), (c), (d), figures 5.4(a), (b), (c), and (d). Figure 5.5(a) shows a simulated color IR image of a portion of the DC Mall area for reference. Figure 5.5(b), and (c), are the classified DC Mall maps for DAFE and NWFE respectively. These figures show that

1. For all real data experiments, NWFE has better performance than the other methods.

2. When the number of extracted features is greater than nc-1, the performances of DAFE and aPAC-LDR decease rapidly, but NWFE and NDA does not.

3. Figure 5.4(c) shows that if only 5 (nc-1) features are used then the accuracies of DAFE and aPAC-LDR are 57.27% and that of NWFE is 86.16%. But if 7 features of NWFE are used then the accuracy increases to 91.57%. This shows that only using nc-1 features is not enough in this real situation. DAFE cannot do this due to the restriction of the rank of the between-class scatter matrix. NWFE does not have this restriction.

4. Comparing Figure 5.5(b) and 5.5(c), one sees that the performance of NWFE is better than that of DAFE in almost all classes.

Table 5.4(a) Mean and standard deviation of accuracies of Cuprite data sets

	DAFE		NWFE		aPAC_LDR		NDA_1NN		NDA_5NN	
Features	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	0.4297	0.0699	0.2592	0.0281	0.3479	0.0295	0.2301	0.0631	0.2311	0.0629
2	0.6026	0.0569	0.6019	0.0234	0,5547	0.0358	0.4234	0.0739	0.4356	0.109
3	0.74	0.0295	0.8686	0.0124	0.7084	0.0327	0.5805	0.0484	0.6456	0.0464
4	0.8156	0.0246	0.9439	0.0203	0.8035	0.0281	0.7329	0.0231	0.7412	0.0327
5	0.8699	0.0202	0.9671	0.0125	0.8799	0.0176	0.7983	0.0167	0.8007	0.0205
6	0.88	0.0252	0.9827	0.0061	0.8835	0.0222	0.8648	0.0231	0.8516	0.0304
7	0.8943	0.0205	0.984	0.0062	0.8943	0.0205	0.8731	0.0188	0.8492	0.0304
8	0.7076	0.3736	0.9829	0.0054	0.6194	0.4278	0.8786	0.0195	0.8488	0.0307
9	0.3537	0.4567	0.9829	0.0047	0.352	0.4546	0.8812	0.0201	0.8482	0.0303
10	0.1816	0.3829	0.9826	0.0045	0.2643	0.4257	0.8694	0.0209	0.8401	0.0296
11	0.091	0.2878	0.9833	0.0038	0.0889	0.2813	0.8676	0.0202	0.8311	0.0318
12	0	0	0.981	0.0052	0.0888	0.2807	0.8648	0.0259	0.8219	0.0349
13	0	0	0.9806	0.0043	0.0885	0.2798	0.858	0.0237	0.8091	0.0315
14	0	0	0.979	0.0054	0	0	0.8564	0.0247	0.8069	0.031
15	0	0	0.9783	0.0065	0	0	0.8548	0.0218	0.7995	0.0284

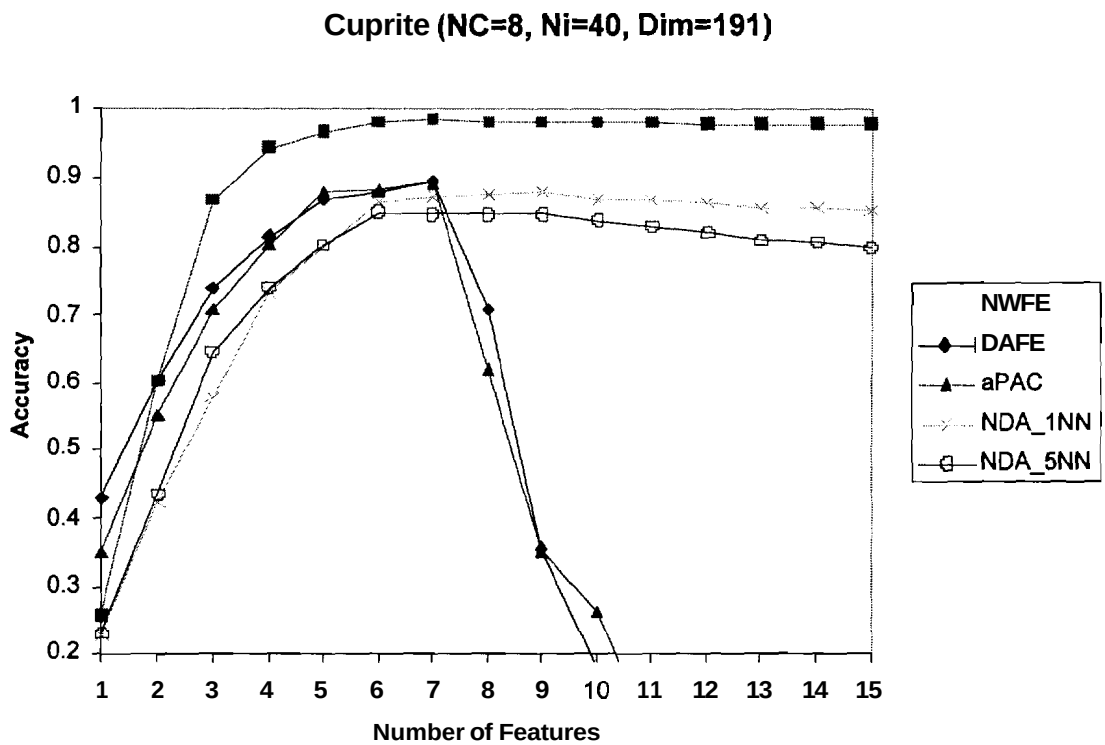


Figure 5.4(a) Mean of accuracies of Cuprite data sets using 1–15 features

Table 5.4(b) Mean and standard deviation of accuracies of Jasper Ridge data sets

	DAFE		NWFE		aPAC_LDR		NDA_1NN		NDA_5NN	
Features	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	0.4869	0.0649	0.8253	0.0261	0.511	0.0722	0.2941	0.0557	0.2941	0.0553
2	0.7783	0.069	0.9742	0.0037	0.7596	0.058	0.6041	0.0433	0.602	0.0425
3	0.8495	0.0587	0.9816	0.0022	0.8047	0.0567	0.8135	0.0466	0.8106	0.047
4	0.8971	0.0335	0.9883	0.0041	0.8772	0.0471	0.8471	0.0509	0.8444	0.051
5	0.9127	0.0243	0.9916	0.0022	0.9127	0.0243	0.8901	0.0322	0.8864	0.0342
6	0.82	0.2892	0.993	0.0015	0.5441	0.4687	0.9072	0.0242	0.9037	0.0271
7	0.6388	0.4414	0.9934	0.0017	0.2716	0.4376	0.9027	0.0225	0.8998	0.0249
8	0.2785	0.4484	0.9932	0.0019	0.1857	0.3912	0.8987	0.0231	0.8949	0.0275
9	0.0937	0.2962	0.9929	0.0023	0.1856	0.3911	0.899	0.0252	0.8937	0.0253
10	0	0	0.9921	0.0035	0.1859	0.3917	0.8893	0.0309	0.893	0.0327
11	0	0	0.993	0.0026	0	0.0001	0.8862	0.0351	0.8862	0.0339
12	0	0	0.9924	0.0031	0	0	0.8848	0.036	0.8821	0.0341
13	0	0	0.9925	0.0029	0.0001	0.0002	0.8765	0.0386	0.8773	0.0274
14	0	0.0001	0.9921	0.0028	0	0.0001	0.872	0.0373	0.8735	0.0314
15	0	0.0001	0.9919	0.0026	0.0001	0.0002	0.8658	0.0338	0.8623	0.0325

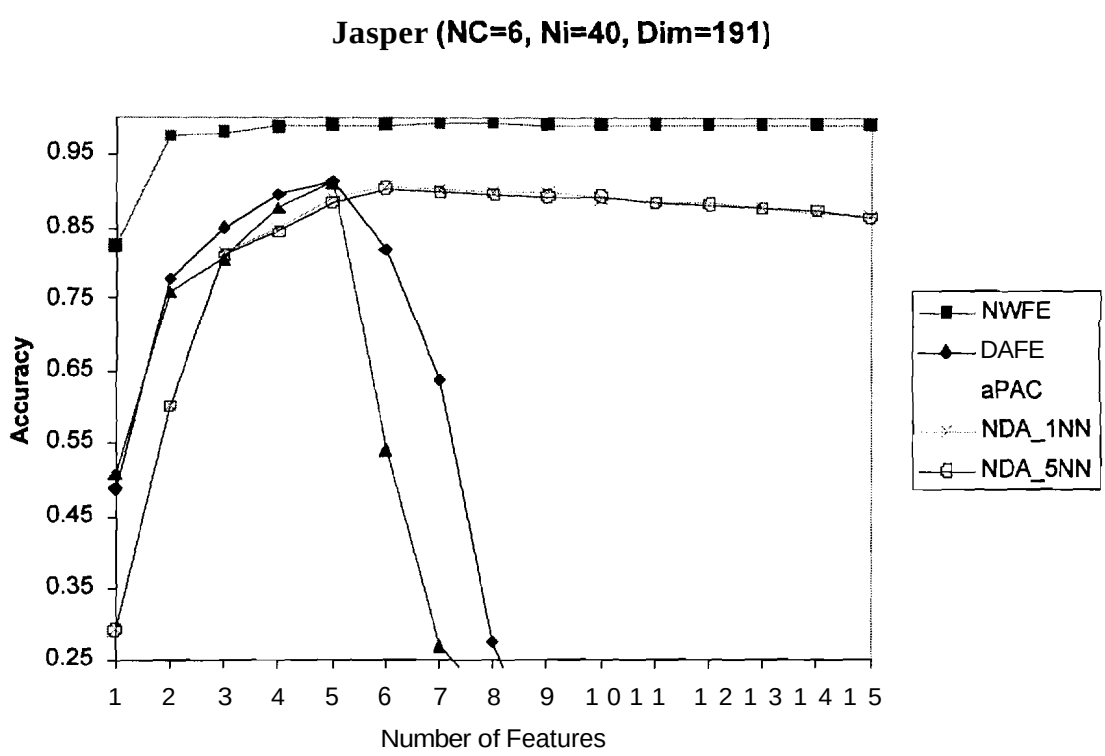


Figure 5.4(b) Mean of accuracies of Jasper Ridge data sets using 1–15 features

Table 5.4(b) Mean and standard deviation of accuracies of Indian Pine data sets

	DAFE		NWFE		aPAC_LDR		NDA_1NN		NDA_5NN	
Features	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	0.3544	0.0224	0.5298	0.0211	0.3225	0.0494	0.2609	0.0523	0.2609	0.0522
2	0.4623	0.0334	0.7171	0.0142	0.4317	0.0307	0.3972	0.0295	0.3986	0.031
3	0.5044	0.0226	0.7946	0.015	0.4921	0.0315	0.468	0.0274	0.4688	0.0219
4	0.5483	0.0187	0.809	0.0177	0.5382	0.0328	0.5081	0.0204	0.5091	0.0207
5	0.5727	0.0156	0.8613	0.0104	0.5727	0.0156	0.5466	0.0247	0.5427	0.0228
6	0.2859	0.3012	0.8908	0.0128	0.458	0.2416	0.5606	0.0192	0.5569	0.0189
7	0.1751	0.2816	0.9156	0.0048	0.2892	0.3048	0.557	0.0241	0.5523	0.0135
8	0.0583	0.1835	0.9114	0.0057	0.059	0.186	0.5539	0.0221	0.5459	0.0174
9	0.0583	0.1832	0.9064	0.0078	0.0003	0.0005	0.5457	0.0194	0.5338	0.0203
10	0.0585	0.1839	0.9062	0.0083	0.0003	0.0005	0.5362	0.0218	0.5259	0.0132
11	0.0004	0.0003	0.9007	0.0083	0.0005	0.0007	0.5299	0.0213	0.5172	0.0174
12	0.0004	0.0003	0.8977	0.0078	0.0004	0.0005	0.5279	0.0181	0.5104	0.0151
13	0.0004	0.0003	0.8944	0.008	0.0003	0.0004	0.5176	0.0155	0.5046	0.0192
14	0.0004	0.0004	0.8922	0.0075	0.0003	0.0004	0.5117	0.016	0.4932	0.0191
15	0.0004	0.0004	0.8876	0.0084	0.0004	0.0004	0.511	0.0217	0.4847	0.0151

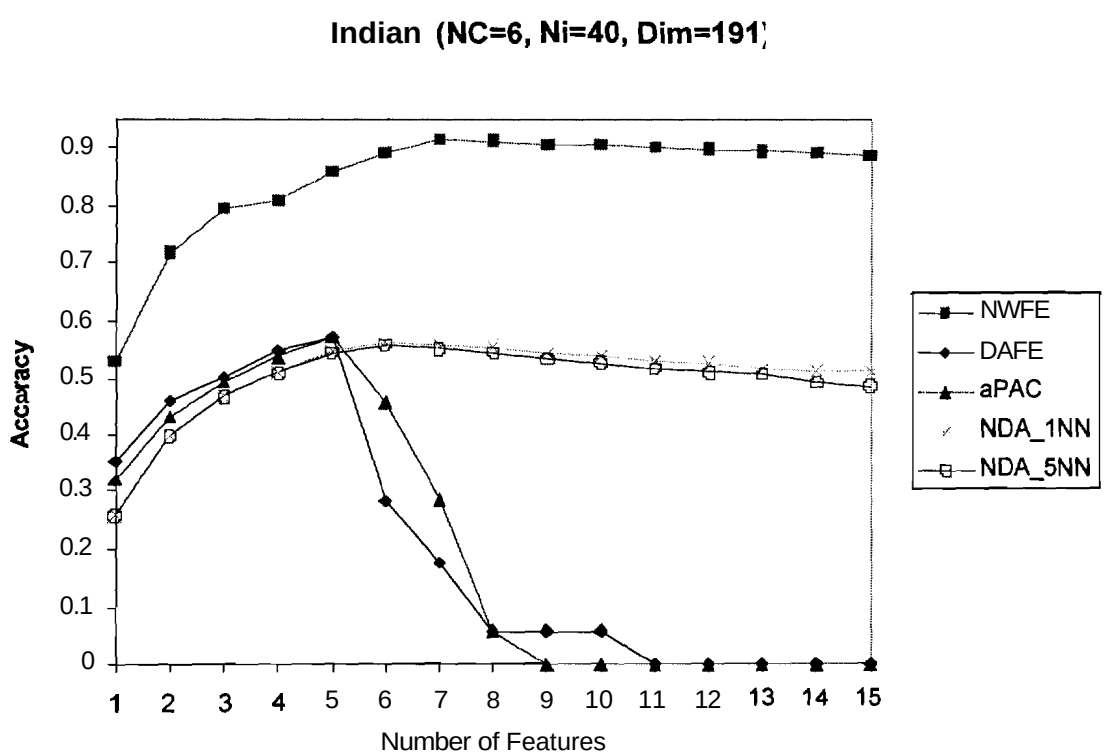


Figure 5.4(c) Mean of accuracies of Indian Pine data sets using 1–15 features

Table 5.4(d) Mean and standard deviation of accuracies of DC Mall data sets

	DAFE		NWFE		aPAC_LDR		NDA_1NN		NDA_5NN	
Features	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	0.4976	0.086	0.7053	0.0813	0.516	0.0535	0.2739	0.0652	0.2739	0.0648
2	0.6739	0.0485	0.853	0.0451	0.6458	0.0354	0.5301	0.0772	0.5262	0.0789
3	0.7325	0.0555	0.9125	0.018	0.7273	0.0604	0.642	0.0491	0.6391	0.0482
4	0.7437	0.0482	0.9179	0.0174	0.7254	0.0708	0.7009	0.0458	0.6934	0.0459
5	0.7424	0.0592	0.9206	0.0161	0.7373	0.0643	0.7151	0.0417	0.7054	0.0423
6	0.7392	0.053	0.92	0.0168	0.7392	0.053	0.708	0.0585	0.7002	0.0585
7	0.3882	0.396	0.9217	0.0157	0.6197	0.3204	0.7037	0.0537	0.6918	0.0556
8	0.0866	0.2345	0.9223	0.0132	0.2517	0.3812	0.6976	0.0538	0.7	0.0496
9	0.0122	0.0076	0.9219	0.0128	0.1677	0.3274	0.6902	0.0557	0.7238	0.035
10	0.0103	0.0045	0.9213	0.0134	0.0891	0.2498	0.6882	0.0583	0.7315	0.0468
11	0.0083	0.0043	0.9209	0.013	0.009	0.0081	0.7086	0.062	0.7386	0.052
12	0.0078	0.0045	0.9218	0.013	0.0095	0.0075	0.7497	0.0402	0.7446	0.0541
13	0.0065	0.0044	0.9229	0.0144	0.0092	0.0065	0.7667	0.0299	0.7402	0.0553
14	0.0073	0.0056	0.9233	0.0128	0.0085	0.0065	0.7599	0.0317	0.7343	0.0652
15	0.0067	0.0049	0.9214	0.0132	0.008	0.0058	0.7491	0.0425	0.7291	0.0638

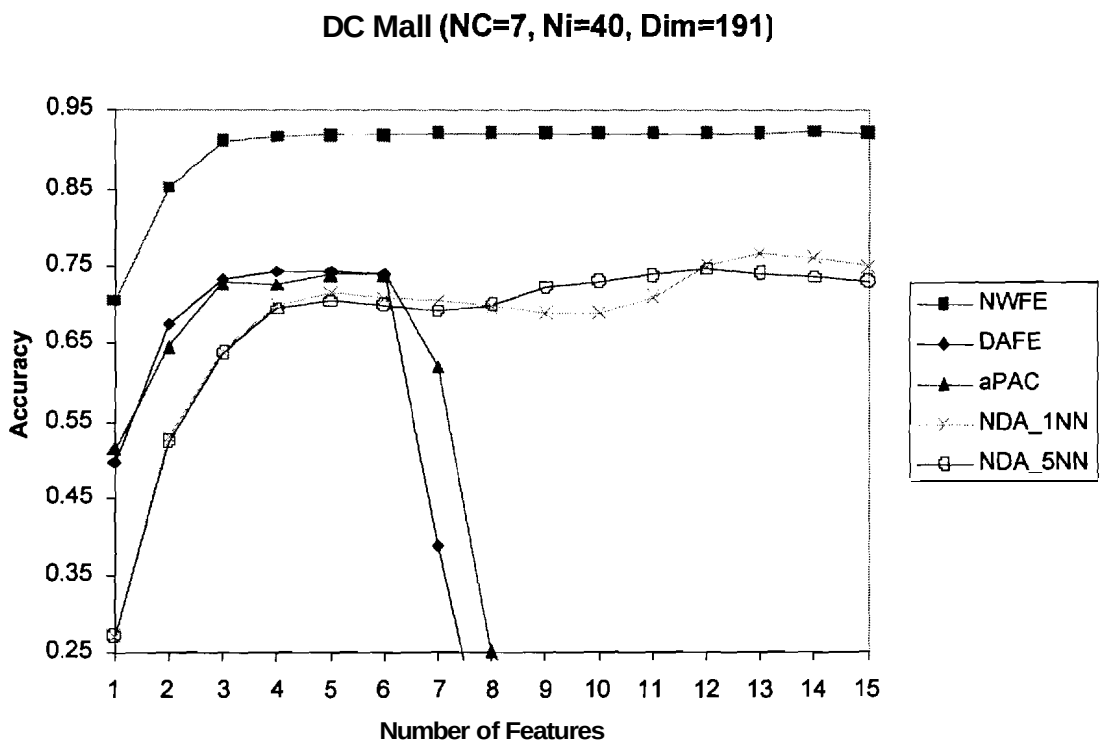


Figure 5.4(d) Mean of accuracies of DC Mall data sets using 1–15 features

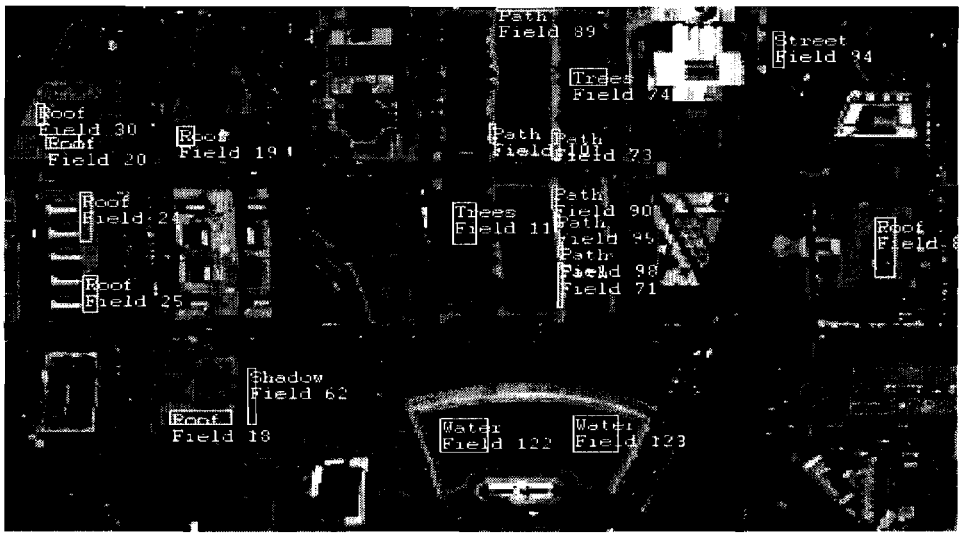


Figure 5.5(a) A color IR image of a portion of the DC data set. (In Color)

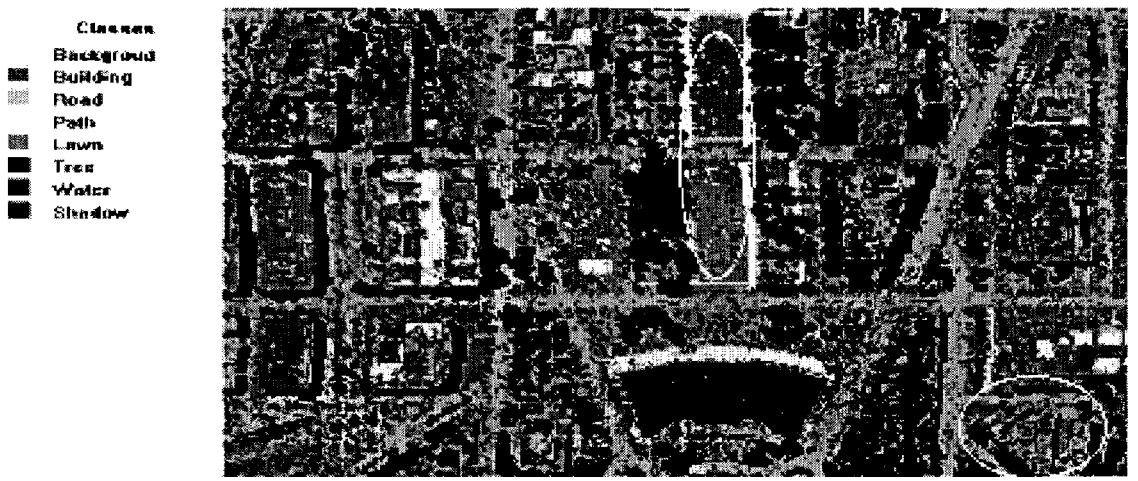


Figure 5.5(b). The thematic map resulting from the classification of the area of Figure 5.4(a) using DAFE features. (In Color)

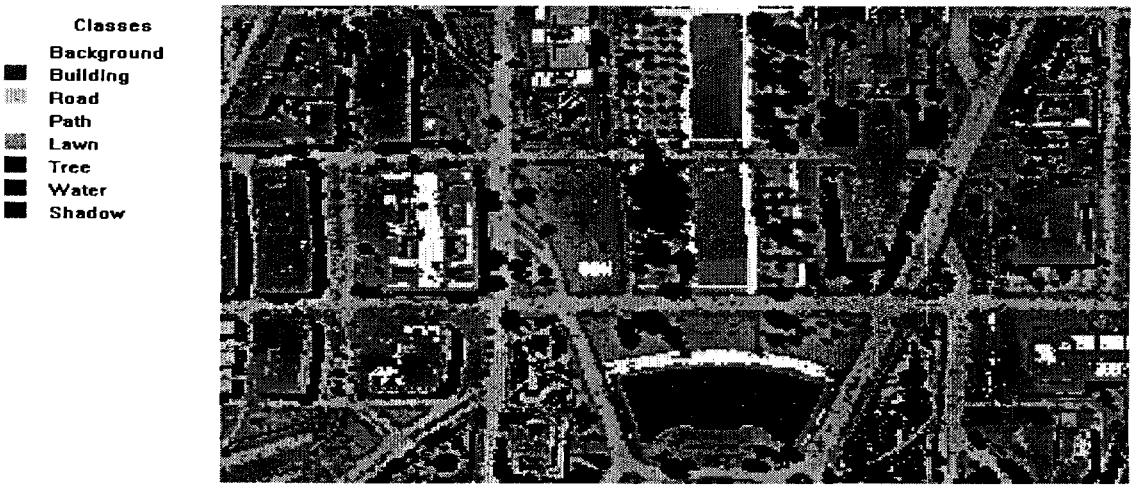


Figure 5.5(c). The thematic map resulting from the classification of the area of Figure 5.4(a) using NWFE features. (In Color)

5.6 Concluding Comments

The volume available in high dimensional feature spaces is very large, making possible the discrimination between classes with only very subtle differences. On the other hand, this large volume makes increasingly challenging the problem of defining ***adequate precisely*** the desired classes in terms of the feature space variables. The problems of class statistics estimation error resulting from training sets of finite size grows rapidly with dimensionality, thus making it desirable to use no larger feature space dimensionality than necessary for the problem at hand, and therefore the :importance of an effective, case-specific feature extraction procedure.

The NWFE algorithm presented here is intended to take advantage: of the desirable characteristics of DAFE and DBFE, while avoiding their shortcomings. DAFE is fast and easy to apply, but its limitation of nc-1 features, its reduced performance particularly when the difference in mean values of classes is small, and the fact that iit is based on the statistical description of the entire training set, making it sensitive to outliers, limit its performance in many cases. DBFE does not have these limitations. It focuses the attention on training samples near the needed decision boundary, but it is a long calculation and does not produce effective results when training sets are small.

NWFE does not have any of these limitations. It appears to have improved performance in a broad set of circumstances, making possible substantially better classification accuracy in the data sets tested, which included sets of agricultural,

geological, ecological and urban significance. This improved performance is perhaps due to the fact that, like DBFE, attention is focused upon training samples that are near to the eventual decision boundary, rather than equally weighted on all training pixels as with DAFE. It also appears to provide feature sets which are relatively insensitive to the precise choice of feature set size, since the accuracy versus dimensionality curves are relatively flat beyond the initial knee of the curve. This characteristic would appear to be significant for the circumstance when this technology begins to be used by general remote sensing practitioners who are not otherwise highly versed in signal processing principles and thus might not realize how to choose the right dimensionality to use.

CHAPTER 6: Using Mixture Classifier Based on Mix-LOOC2 after Feature Extraction

6.1 Cntroduction

From Chapter 4, we know that a mixture classifier based on Mix-LOOC2 is a good choice for classifying data in the original space. But using that mixture classifier in hyper dimensional data is not efficient and will suffer from the Hughes phenomenon more seriously. Before classifying hyper dimensional data, feature extraction is usually used to transform data from the original hyper dimensional space into a lower dimensional feature space. This section is to explore the performances of combining feature extraction and the mixture classifier based on Mixed-LOOC2 procedures.

6.2 Experiment Design

In this section, the performances of the following four classification procedures are compared.

1. Using DAFE features applied to the Gaussian quadratic classifier (DAFE+GC). This is the previous, conventionally used approach and serves as a baseline for comparison.

2. Using DAFE features applied to the mixture classifier based on BIC and Mixed-LOOC2 covariance estimator (DAFE+MC-Mix2).

3. Using *NWFE* applied to the Gaussian quadratic classifier (NWFE+GC).

4. Using NWFE features applied to a mixture classifier based on BIC and Mixed-LOOC2 covariance estimator (NWFE+MC-Mix2).

The experiment data are again in two parts, simulated and real data. Ten simulated data sets in Experiment 5.2 with *30* and *60* dimensions and mixture distributions are used

in Experiment 6.1 to compute the average accuracy of four different procedures. Ten randomly sampled DC Mall and Purdue campus data sets are used in Experiment 6.2 to compute the average accuracy of four different procedures. The dimensionality of the DC Mall data sets is 191 and that of the Purdue campus data sets is 126. The class training sample sizes of all real data experiments are 40 pixels

6.2 Experiment Results

The results of experiment 6.1 are displayed in tables 6.1(a), (b), and figures 6.1(a), (b). The results of experiment 6.2 are displayed in tables 6.2(a), (b)., (c), and figures 6.2(a), (b), and (c). They show that

1. Figures 6.1(a) and (b) show that using 2 features from NWFE and the mixture classifier based on Mixed-LOOC2 yields the best performance. It implies that NWFE may preserve the original data distribution situation better than DAFE does.

2. Figure 6.2(a) shows that the performances of NWFE+GC and NWFE+MC-Mix2 are similar. When the number of features is greater than nc-1, the performance of DAFE+GC will decrease rapidly but DAFE+MC-Mix2 can improve the situation.

3. Figure 6.3(b) shows that the performances of NWFE+GC and NWFE+MC-Mix2 are similar but the performance of DAFE+MC-Mix2 is much better than that of DAFE+GC.

4. Figure 6.3(c) shows that the performances of DAFE+GC and DAFE+MC-Mix2 are similar but the performance of NWFE+MC-Mix2 is better than that of. NWFE+GC.

5. Generally speaking, using the procedure NWFE+MC-Mix2 yielded better results and reduced the Hughes phenomenon but it needs more computation time.

Table 6.1(a) Mean and standard deviation of accuracies of simulated data sets (dim=30)

	DAFE		DAFE+Mixture		NWFE		NWFE+Mixture	
Features	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	**0.3405**	**0.022**	**0.3501**	**0.0383**	**0.5145**	**0.0114**	0.6928	**0.017**
2	**0.3822**	**0.0329**	**0.3866**	**0.0423**	**0.8194**	**0.0118**	**0.9922**	**0.003**
3	0.3818	0.0269	0.3653	0.0279	0.8167	0.0161	0.8447	0.0627
4	0.3798	0.0254	0.3625	0.0257	0.8143	0.0167	0.7905	0.0173
5	0.3732	0.0234	0.3498	0.0219	0.8069	0.0142	0.7865	0.0142
6	0.3857	0.1265	0.4026	0.0861	0.8034	0.015	0.7835	0.0113
7	0.3716	0.1535	0.3908	0.0646	0.7943	0.0147	0.772	0.0155
8	0.3067	0.1395	0.4183	0.0485	0.7893	0.0144	0.7615	0.0164
9	0.265	0.1028	0.3882	0.074	0.7807	0.0152	0.7457	0.0159
10	0.2379	0.0607	0.4118	0.0784	0.7735	0.0148	0.7284	0.0135
11	0.232	0.0554	0.4032	0.0958	0.7685	0.016	0.7113	0.0181
12	0.2251	0.0656	0.4126	0.0732	0.7635	0.0132	0.6605	0.0414
13	0.2191	0.0819	0.3906	0.0892	0.756	0.0126	0.6345	0.0379
14	0.2146	0.0856	0.3871	0.0843	0.7483	0.0149	0.6179	0.0204
15	0.203	0.0884	0.373	0.0948	0.7413	0.017	0.6117	0.0173

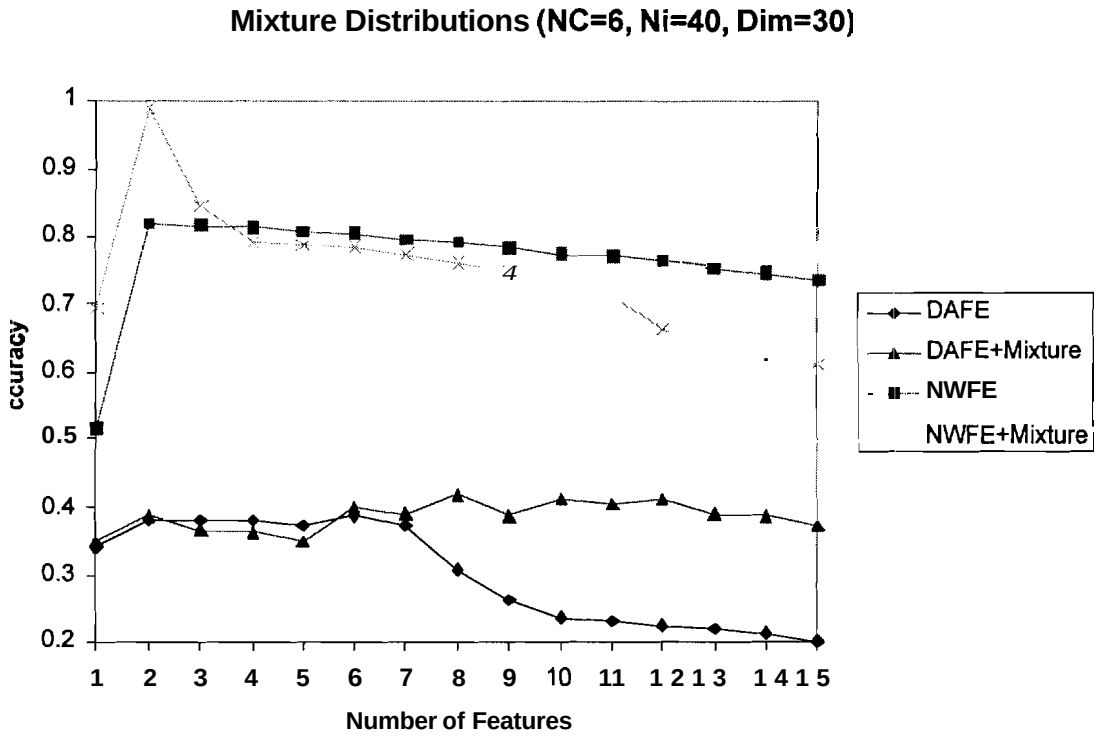


Figure 6.1(a) Mean of accuracies of simulated data sets (dim=30)

Table 6.1(b) Mean and standard deviation of accuracies of simulated data sets (dim=60)

	DAFE		DAFE+Mixture		NWFE		NWFE+Mixture	
Features	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	0.2898	0.0252	0.2759	0.0314	0.4491	0.0299	0.5425	0.0239
2	0.281	0.0232	0.2753	0.0238	0.7262	0.0293	0.89	0.028
3	0.2743	0.0138	0.2702	0.0134	0.7175	0.0246	0.7553	0.0734
4	0.2725	0.0151	0.2658	0.0154	0.7271	0.0346	0.7076	0.0368
5	0.2688	0.0152	0.2601	0.0124	0.7255	0.0266	0.701	0.0215
6	0.2766	0.0509	0.2713	0.0324	0.7315	0.0302	0.7028	0.0308
7	0.2826	0.0565	0.2729	0.0349	0.7333	0.0302	0.7015	0.0247
8	0.268	0.0328	0.2705	0.0305	0.733	0.0266	0.6933	0.0291
9	0.2679	0.0146	0.2733	0.0288	0.7281	0.0253	0.6692	0.028
10	0.2781	0.0175	0.2721	0.0388	0.7215	0.022	0.6587	0.0256
11	0.2865	0.0192	0.2696	0.0173	0.7202	0.022	0.6479	0.0359
12	0.289	0.0232	0.2837	0.0374	0.7122	0.0213	0.6043	0.0447
13	0.2805	0.0361	0.2715	0.0352	0.7067	0.0218	0.5712	0.0433
14	0.2825	0.0332	0.2916	0.0358	0.702	0.0232	0.5516	0.0175
15	0.271	0.0322	0.2788	0.026	0.6939	0.0225	0.5461	0.0206

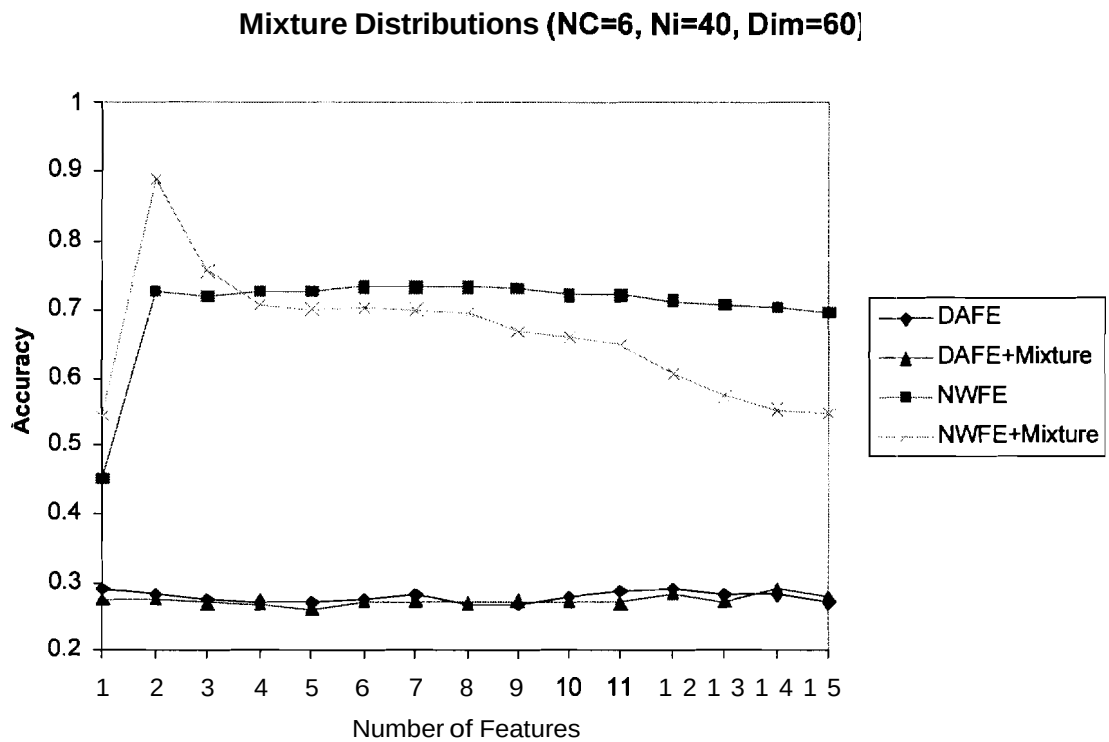


Figure 6.1(b) Mean of accuracies of simulated data sets (dim=60)

Table 6.2(a) Mean and standard deviation of accuracies of Indian Pine data sets (dim=191)

	DAFE		DAFE+Mixture		NWFE		NWFE+Mixture	
Features	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	0.3539	0.0229	0.3544	0.0226	0.5162	0.0269	0.5302	0.0213
2	0.4629	0.0331	0.4627	0.0331	0.7066	0.0067	0.7174	0.0142
3	0.5048	0.0231	0.5075	0.0228	0.777	0.0171	0.7943	0.015
4	0.549	0.0185	0.5532	0.0228	0.7963	0.0202	0.8092	0.0157
5	0.5733	0.0155	0.5836	0.0208	0.8583	0.0172	0.8618	0.0099
6	0.2867	0.302	0.576	0.0272	0.8909	0.0138	0.8918	0.0137
7	0.1754	0.282	0.5421	0.0352	0.9121	0.0067	0.9161	0.0033
8	0.0584	0.1834	0.558	0.0522	0.9098	0.0065	0.9122	0.0041
9	0.0586	0.1841	0.5634	0.0439	0.905	0.0088	0.9076	0.0059
10	0.0585	0.184	0.5636	0.0399	0.9064	0.0082	0.9078	0.0053
11	0.0004	0.0003	0.5329	0.0842	0.9027	0.0069	0.9023	0.0052
12	0.0004	0.0003	0.5315	0.085	0.9007	0.0067	0.8979	0.007
13	0.0004	0.0003	0.5384	0.0836	0.899	0.0067	0.8865	0.0154
14	0.0004	0.0004	0.5429	0.0606	0.8959	0.0048	0.8717	0.0233
15	0.0004	0.0004	0.5631	0.0237	0.8917	0.0059	0.8635	0.0196

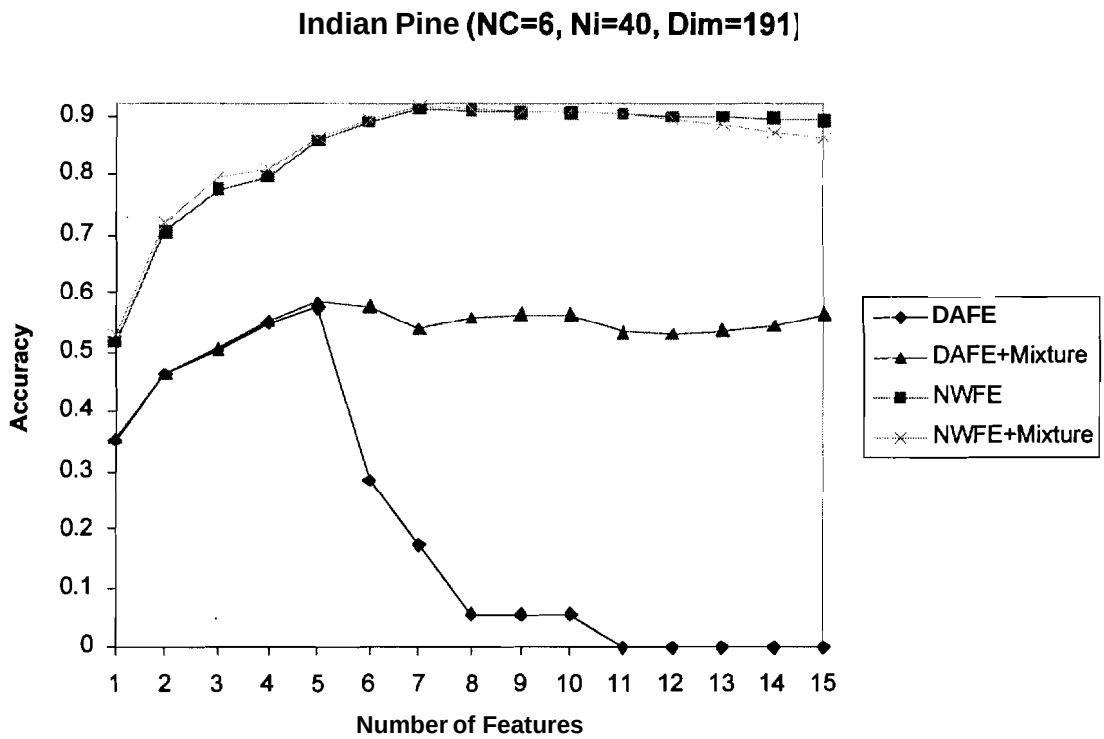


Figure 6.2(a) Mean of accuracies of simulated data sets (dim=60)

Table 6.2(b) Mean and standard deviation of accuracies of DC Mall data sets (dim=191)

	DAFE		DAFE+Mixture		NWFE		NWFE+Mixture	
Features	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	**0.4994**	**0.0895**	**0.5082**	**0.0844**	**0.6934**	**0.0928**	**0.7053**	**0.0813**
2	0.6768	0.0477	0.7041	0.0442	0.85	0.0514	0.8525	0.0444
3	0.7355	0.0549	0.7799	0.0395	0.9123	0.0182	0.9114	0.017
4	0.746	0.0476	0.8029	0.0297	0.9174	0.0176	0.9165	0.017
5	0.7453	0.058	0.8163	0.0275	0.919	0.0169	0.9196	0.0159
6	0.7413	0.0522	0.8276	0.0288	0.9169	0.0172	0.9194	0.0163
7	0.3886	0.3965	0.8225	0.0395	0.9151	0.0169	0.9176	0.0157
8	0.0865	0.2344	0.7738	0.0767	0.9141	0.0129	0.9178	0.0155
9	0.0122	0.0076	0.7568	0.06	0.9155	0.0127	0.9176	0.0154
10	0.0103	0.0045	0.7421	0.1117	0.9154	0.0136	0.917	0.0153
11	0.0083	0.0043	0.7126	0.1111	0.9162	0.0136	0.9178	0.0159
12	0.0078	0.0045	0.7019	0.0919	0.9178	0.0134	0.9173	0.0158
13	0.0065	0.0044	0.6674	0.1475	0.9201	0.0149	0.9164	0.0179
14	0.0074	0.0057	0.7076	0.1187	0.9206	0.0144	0.917	0.018
15	0.0067	0.0049	0.6316	0.1303	0.9192	0.0151	0.9155	0.0177

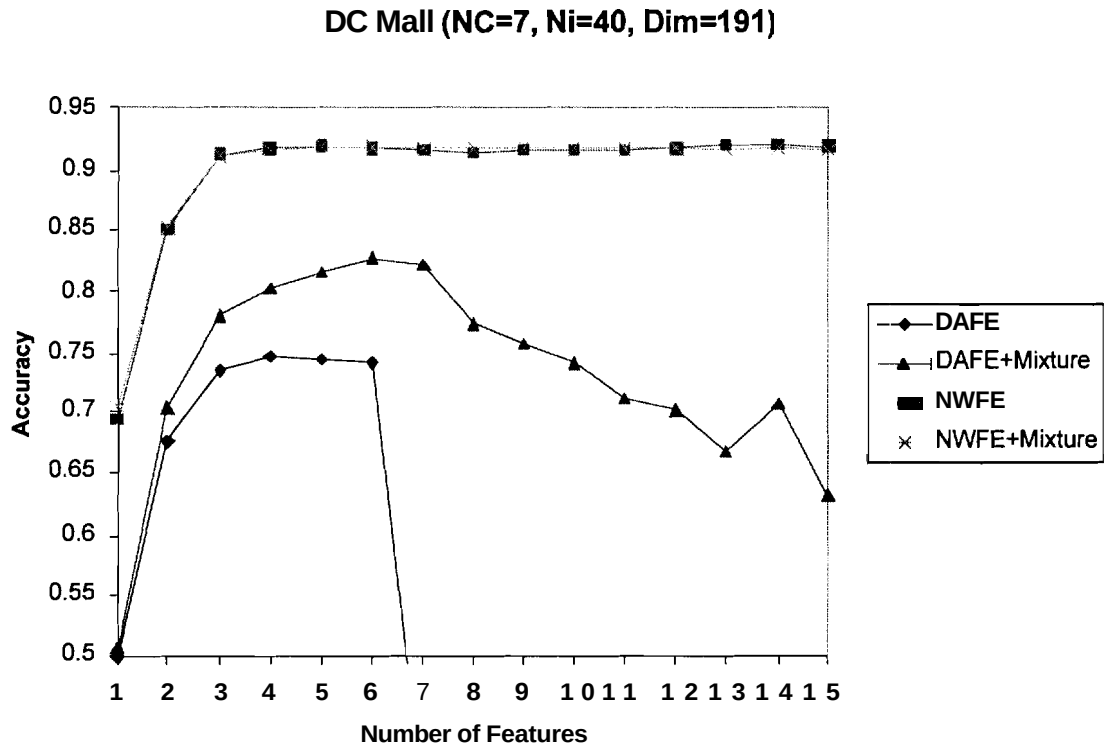


Figure 6.2(b) Mean of accuracies of DC Mall data sets (dim=191)

Table 6.2(c) Mean and standard deviation of accuracies of Purdue campus data sets (dim=126)

	DAFE		DAFE+Mixture		NWFE		NWFE+Mixture	
Features	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	0.495	0.0501	0.4911	0.0458	0.613	0.0362	0.6432	0.0497
2	0.5882	0.0639	0.5883	0.0677	0.7557	0.0167	0.7719	0.0205
3	0.6913	0.0379	0.6922	0.0321	0.7811	0.0258	0.8113	0.0266
4	0.8052	0.0114	0.8071	0.0099	0.8088	0.0267	0.8422	0.0377
5	0.8182	0.0154	0.8198	0.0167	0.8148	0.0343	0.8438	0.0388
6	0.4971	0.4282	0.8235	0.0174	0.8176	0.0328	0.8573	0.0208
7	0.4236	0.4467	0.8074	0.0481	0.8244	0.0264	0.8539	0.0285
8	0.2542	0.4097	0.8036	0.0487	0.8311	0.0278	0.8563	0.026
9	0.171	0.3608	0.7469	0.1357	0.8428	0.0349	0.8689	0.0297
10	0	0.0001	0.683	0.1496	0.8549	0.0324	0.8731	0.0282
11	0.0001	0.0003	0.6485	0.1565	0.8702	0.0287	0.8854	0.0213
12	0.0001	0.0003	0.6344	0.1339	0.885	0.0256	0.8892	0.0176
13	0.0004	0.001	0.5415	0.2131	0.8877	0.0229	0.8857	0.017
14	0.0006	0.0018	0.5519	0.2014	0.8907	0.0237	0.8845	0.0213
15	0.0004	0.0011	0.5489	0.2097	0.8954	0.0167	0.8849	0.0233

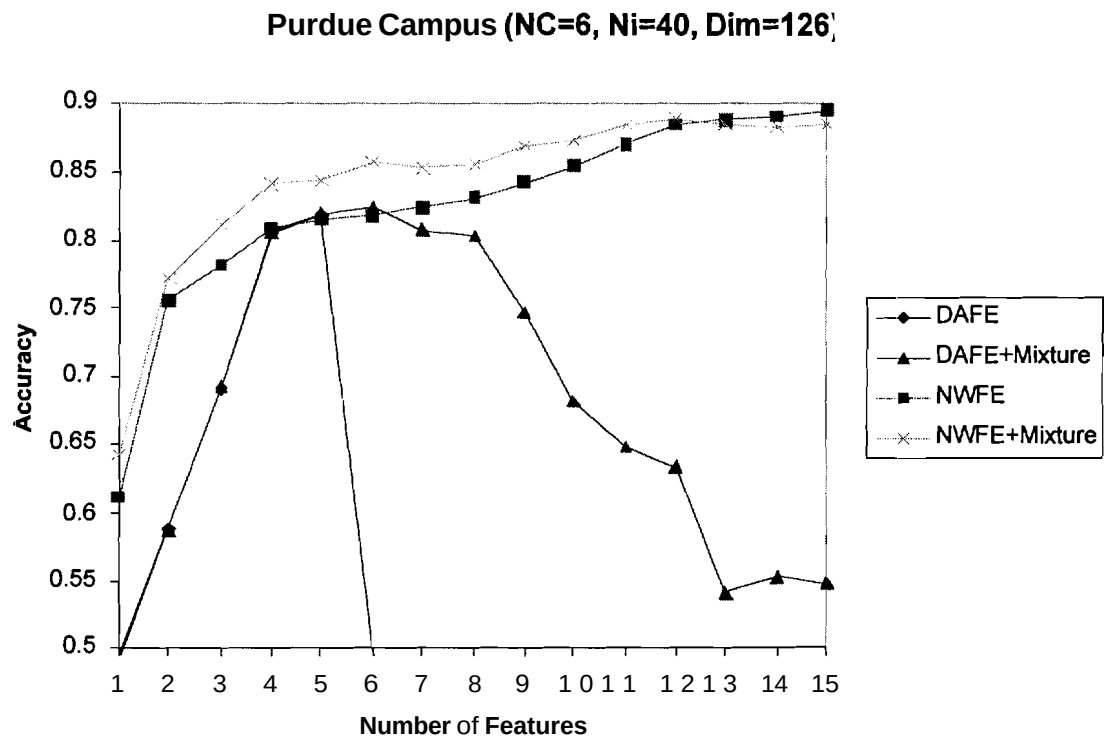


Figure 6.2(c) Mean of accuracies of Purdue campus data sets (dim=126)

6.3 Concluding Comments

It has long been known that modeling each class in a data set with a single mode Gitussian density is rarely a good model. The use of "Gaussian subclasses" to provide a better class model has long been in use, and has shown itself to be an effective way to proceed. This is basically what has been called here a mixture classifier. The problem has been that deciding just how many "subclasses" to use for each class and how to train each has been a substantial challenge to the analyst, Devising an effective scheme for doing this should be a significant aid to the analyst.

The performances of combining feature extraction (DAFE and NWFE) and the mixture classifier based on Mixed-LOOC2 procedures are tested. The simulated and real data results show that using NWFE then the mixture classifier based on nearest mean clustering and BIC_Mix index is a robust classification procedure for hyperspectral data.

CHAPTER 7: CONCLUSIONS

7.1 Summary

In Chapter 2, Mixed-LOOC1 and Mixed-LOOC2 are defined and retain the advantages of LOOC and BLOOC.

In Chapter 3, Mixed-LOOC2 is used with DAFE. Experimental results show that this proposed feature extraction not only can avoid the singularity problem in DAFE but also can get a better result by using fewer training samples.

In Chapter 4, Mixed-LOOC2 is used with parameter estimation and model selection steps of mixture classifiers. Experimental results show that the proposed mixture classifier using nearest mean clustering and BIC_Mix has the advantages of both quadratic and original mixture classifier and outperforms those two in some situations.

In Chapter 5, the proposed nonparametric feature extraction method, NWFE, is defined and takes advantage of the desirable characteristics of DAFE and DBFE, while avoiding their shortcomings.

In Chapter 6, the performances of combining feature extraction (DAFE and NWFE) and the mixture classifier based on Mixed-LOOC2 procedures are tested. The simulated and real data results show that using NWFE then the mixture classifier based on nearest mean clustering and BIC_Mix index is a robust classification procedure for hyperspectral data.

Based on above summary, in feature extraction step, if the total sample size is less than the dimensionality, then DAFE based on Mixed-LOOC2 is suggested; otherwise NWFE is the best choice. In designing classifier step, the mixture classifier based on NM clustering and BIC with Mixed-LOOC2 seems to be the best choice.

In [24], a list of significant factors affecting classification performance includes,

1. The classes of interest and the number of training samples available for each class,
2. The algorithm available by which to estimate especially the covariance matrices.
3. The feature extraction process,
4. The classifier algorithm complexity, and
5. The analyst's skill.

This thesis provides a robust classification procedure that helps the analyst avoid troubles from item 2, 3, and 4.

7.2 Suggestions for Further Work

1. Combine the adaptive classification procedure [26] and the algorithms proposed in this thesis.
2. Find the method to decide how many features should be extracted in NWFE.

.APPENDIXA: THE MAXIMUM LIKELIHOOD ESTIMATOR OF MIXTURE PARAMETER IN LOOC AND BLOOC

The maximum likelihood estimator of the mixing parameter of LOOC or BLOOC will be derived.

Let

$$\tilde{O}_i = \tilde{O}_i(\acute{a}_i) = \acute{a}_i A_i + (1-\acute{a}_i) B_i \quad \text{, where } A_i \text{ and } B_i \text{ aresymmetric}$$

and

$$L_i(\alpha_i) = -\frac{1}{N_i}\sum_{k=1}^{N_i} \log f_i(x_{i,k} \mid \hat{m}_i, \hat{\Sigma}_i)$$

where

$$f_i(x_{i,k} \mid \hat{m}_i, \hat{\Sigma}_i) = \frac{1}{\sqrt{(2\pi)^p \mid \Sigma_i \mid}} \exp\{-\frac{1}{2}(x_{i,k} - \hat{m}_i)^T \hat{\Sigma}_i^{-1}(x_{i,k} - \hat{m}_i)\}, \quad i = 1,2,...,c$$

is the likelihood function of $x_{i,k}$ is the k-th observation in class i, c is the number of classes and A_i and B_i are known P × P matrices.

Since

$$\log f_i(x_{i,k} \mid \hat{m}_i, \hat{\Sigma}_i) = -\frac{p}{2}\log 2\pi - \frac{1}{2}\log \mid \hat{\Sigma}_i \mid - \frac{1}{2}(x_{i,k} - \hat{m}_i)^T \hat{\Sigma}_i^{-1}(x_{i,k} - \hat{m}_i)$$

then

$$L_i(\alpha_i) = -\frac{1}{N_i}\sum_{k=1}^{N_i} \log f_i(x_{i,k} \mid \hat{m}_i, \hat{\Sigma}_i) = \frac{p}{2}\log 2\pi - \frac{1}{2}\log \mid \hat{\Sigma}_i^{-1} \mid + \frac{1}{2N_i}\sum_{i=1}^{N_i}(x_{i,k} - \hat{m}_i)^T \hat{\Sigma}_i^{-1}(x_{i,k} - \hat{m}_i)$$

For an m by n matrix $X = [x_{ij}]$, let

$$\text{vec}\, X = \text{vec}\,(X) = [x_{11}\, x_{21} \ldots x_{m1}\, x_{12}\, x_{22} \ \ldots \ x_{m2} \ldots x_{mn}]^T$$

From [25] **p.176,** we know that

$$\text{vec}\,(\frac{\partial f(X)}{\partial X}) = \frac{\partial f(X)}{\partial \text{vec}(X)} \quad , \quad \text{where} \quad \mathbf{f} : \mathrm{R}^{mn} \rightarrow \mathrm{R}$$

By the chain rule, the first derivative of $L_i(\alpha_i)$ can be written as

$$\frac{\partial L_i(\alpha_i)}{\partial \alpha_i} = \frac{\partial L_i(\alpha_i)}{\partial \text{vec}(H_i)^T} \frac{\partial \text{vec}(H_i)}{\partial \alpha_i} = \text{vec}(\frac{\partial L_i(\alpha_i)}{\partial H_i})^T \text{vec}(\frac{\partial H_i}{\partial \alpha_i}) \quad ,$$

$$\text{where} H_i = \hat{\Sigma}_i^{-1} \text{ and } H_i^{-1} \models | H_i |^{-1}$$

Then we have

$$\frac{\partial L_i(\alpha_i)}{\partial H_i} = -\frac{1}{2}\frac{\partial \log |H_i|}{\partial H_i} + \frac{1}{2N_i}\sum_{k=1}^{N_i} \frac{\partial}{\partial H_i}[(x_{i,k} - \hat{m}_i)^T H_i (x_{i,k} - \hat{m}_i)]$$

Since $H_i = \hat{\Sigma}_i^{-1}$ is symmetric,

$$\frac{\partial \log |H_i|}{\partial H_i} = 2H_i^{-1} - diag(H_i^{-1}) = 2\hat{\Sigma}_i - diag(\hat{\Sigma}_i)$$

and

$$\sum_{k=1}^{N_i} \frac{\partial}{\partial H_i}[(x_{i,k} - \hat{m}_i)^T H_i (x_{i,k} - \hat{m}_i)] = \sum_{k=1}^{N_i} [2(x_{i,k} - \hat{m}_i)(x_{i,k} - \hat{m}_i)^T - diag((x_{i,k} - \hat{m}_i)(x_{i,k} - \hat{m}_i)^T)]$$

$$= 2N_i[S_i - diag(S_i)] \text{ , where } S_i = \frac{1}{N_i}\sum_{k=1}(x_{i,k} - \hat{m}_i)(x_{i,k} - \hat{m}_i)^T$$

Therefore,

$$\frac{\partial L_i(\alpha_i)}{\partial H_i} = -\frac{1}{2}[2(\hat{\Sigma}_i - S_i) - diag(\hat{\Sigma}_i - S_i)]$$

And

$$\frac{\partial H_i}{\partial \alpha_i} \quad \frac{\partial \hat{\Sigma}_i^{-1}}{\partial \alpha_i} = -\hat{\Sigma}_i^{-1}(A_i - B_i)\hat{\Sigma}_i^{-1}$$

Since

$$\text{vec}(X^T)^T \text{vec}(Y) = tr(XY) = tr(YX), \text{ where } X : m \times n \text{ and } Y : n \times m$$

then we have

$$\frac{\partial L_i(\alpha_i)}{\partial \alpha_i} = \text{vec}[(\hat{\Sigma}_i - S_i) - \frac{1}{2} diag(\hat{\Sigma}_i - S_i)]^T \text{vec}[\hat{\Sigma}_i^{-1}(A_i - B_i)\hat{\Sigma}_i^{-1}]$$

$$= tr\{[(\hat{\Sigma}_i - S_i) - \frac{1}{2} diag(\hat{\Sigma}_i - S_i)][\hat{\Sigma}_i^{-1}(A_i - B_i)\hat{\Sigma}_i^{-1}]\}$$

Since A_i and B_i are not the same in LOOC and BLOOC, $\frac{\partial L_i(\alpha_i)}{\partial \alpha_i}$ is not equal to 0 for all α_i. Therefore, we know that the optimal solution of the mixture pararneter occurs at one of the end points.

APPENDIX B: THE INFORMATION ABOUT SIMULATION DATA SETS AND REAL DATA SETS

B.l Experiment Design of Simulation Studies

The experiments 1 to 12 are three-class problems.

B.1.1 The Mean Vector and Covariance Matrix

The followings are some notations used in this study.

Mi: the mean vector of class i;

$\mu_{i,j}$: The j-th element of Mi, $i=1,2,3$.

Covi: The covariance matrix of class i;

$\sigma_{i,j}$: The j-th diagonal element of Covi

p: number of dimensions

N_i : the training sample size of class I; $N = \sum N_i$

[Experiment 2.1 and 2.7]

M1=[0,...,0]; M2=[0,3,0...,0];M3=[0,0,3,0,...,0] ,

Cov1=Cov2=Cov3=I;

[Experiment 2.2 and 2.81

M1=[0,...,0]; M2=[0,3,0...,0];M3=[0,4,0,0,...,0] ,

Cov1=I; Cov2=2I; Cov3=3I;

[Experiment 2.3 and 2.9]

$$\mu_{1,i} = 0,\ \mu_{2,i} = 2.5\sqrt{\frac{\sigma_i}{p}}\left(\frac{p-i}{\frac{p}{2}-i}\right),\ \mu_{3,i} = (-1)^i \mu_{2,i} \quad ,$$

Cov1=Cov2=Cov3=diag $(\sigma_1, \sigma_2, ..., \sigma_p)$;

where $\sigma_i = \left[\frac{9(i-1)}{p-1}+1\right]^2$, i = 1,2,...,p

Experiment 2.4 and 2.101

$$\mu_{1,i} = 0,\ \mu_{2,i} = 2.5\sqrt{\frac{\sigma_i}{p}}\left(\frac{i-1}{\frac{p}{2}-1}\right),\ \mu_{3,i} = (-1)^i \mu_{2,i} \quad ,$$

Cov1=Cov2=Cov3=diag $(\sigma_1, \sigma_2, ..., \sigma_p)$;

where $\sigma_i = \left[\frac{9(i-1)}{p-1}+1\right]^2$, i = 1,2,...,p

[Experiment 2.5 and 2.111

$$\mu_{1,i} = \mu_{2,i} = \mu_{3,i} = 0 \quad ,$$

Cov1 =diag $(\sigma_{1,1}, \sigma_{1,2}, ..., \sigma_{1,p})$; where $\sigma_{1,i} = \left[\frac{9(i-1)}{p-1}+1\right]^2$

Cov2 =diag $(\sigma_{2,1}, \sigma_{2,2}, ..., \sigma_{2,p})$; where $\sigma_{2,i} = \left[\frac{9(i-1)}{p-1}+1\right]^2$

Cov3=diag $(\sigma_{3,1}, \sigma_{3,2}, ..., \sigma_{3,p})$; where $\sigma_{2,i} = \left[\frac{9(i-\frac{p-1}{2})}{p-1}\right]^2$

[Experiment 2.6 and 2.12]

$$\mu_{1,i} = 0, \quad \mu_{2,i} = \frac{14}{\sqrt{p}}, \quad \mu_{3,i} = (-1)^i \mu_{2,i}, \quad i = 1,2,...,p \quad ,$$

All covariance matrices are the same as those in Experiment 5 and 11.

B.2 Dimensionality and Sample Size of Real Data Sets

B.2.1 Cuprite, Nevada scene data

Cuprite, Nevada covers an interesting geological feature called a hydrothermal alteration zone, which is exposed due to sparse vegetation. A total of 2744 samples and 191 bands (0.40-1.34, 1.43-1.80, 1.96-2.46 μm) are used.

Table B.1 Labeled Sample Sizes of Cuprite Data Set

	Labeled Samples
Alunite	729
Buddingtonite	71
Kaolinite	232
Quartz	385
Alluvium	689
Playa	252
Tuff	293
Argillized	93
Total Samples	2744

B.2.2 Jasper Ridge Data

This is a biological preserve in San Mateo County, California. In all, 3207 labeled samples are used. The 191 spectral bands (0.40-1.34, 1.43-1.80, and 1.95-2.47 μm) outside the water absorption bands are used.

Table B.2 Labeled Sample Sizes of Jasper Ridge Data Set

	Labeled Samples
Evergreen	900
Serpentine	202
Green-stone	810
Water	208
Deciduous	495
Chaparral	592
Total Samples	3207

B.2.3 Indian Pine Data

This is a mixed forest/agricultural area in Indiana. The water absorption bands (104-108, 150-163,220) have been discarded,

Table B.3 Labeled Sample Sizes of Indian Pine Data Set

	Labeled Samples
Beans/Corn Residue	520
Corn/No Residue	450
Corn/Bean Residue	372
Beans/No Residue	490
Corn/Wheat Residue	388
Wheat/No Residue	301
Total Samples	2521

B.2.4 DC Mall Data

DC Mall image data is an airborne hyperspectral data flighline over the Washington DC mall, which was collected with the HYDICE system. There were 210 bands in the 0.4 to 2.4 μm region of the visible and infrared spectrum. In the experiments, the water absorption bands are removed and 191 bands are used.

Table B.4 Labeled Sample Sizes of Cuyrite Data Set

	Labeled Samples
Building	3834
Road	680
Path	616
Lawn	1928
Tree	919
Water	1224
Shadow	221
Total Samples	9422

REFERENCES

[1] David Landgrebe, "Information Extraction Principles and Methods for Multispectral and Hyperspectral Image Data," Chapter 1 of *Information Processing* for Remote Sensing, edited by C. H. Chen, published by the World Scientific Publishing Co., Inc., 1060 Main Street, River Edge, NJ 07661, USA 1999.

[2] S. Raudys and A. Saudargiene, "Structures of the Covariance Matrices in Classifier Design", Advances in Pattern Recognition, A. Amin, D. Dori, P. Pudil, and H. Freeman, ed., Berlin Heidelberg: Springer-Verlag pp.583-592,1998.

[3] J.H. Friedman, " Regularized Discriminant Analysis," Journal of the American Statistical Association, vol. 84, pp. 165-175, March 1989

[4] W. Rayens and T. Greene, " Covariance pooling and stabilization for classification." *Computational* Statistics and Data Analysis, vol. 11, pp. 17-42, 1991

[5] J. P. Hoffbeck and D.A. Landgrebe, Classification of High Dimensional Multispectral Data, Purdue University, West Lafayette, IN., TR-EE 95-14, May, 1995, pp.43-71

[6] J. P. Hoffbeck and D.A. Landgrebe, " Covariance matrix estimation and classification with limited training data" IEEE Transactions on Pattern Analysis & Machine Intelligence, vol 18, No. 7, pp. 763-767, July 1996

[7] S. Tadjudin and D.A. Landgrebe, Classification of High Dimensional Data with Limited Training Samples, PhD thesis Purdue University, West Lafayette, IN., ECE Technical Report TR-EE 98-8, April, 1998, pp35-82.

[8] K. Fukunaga, Introduction to Statistical Pattern Recognition, San Diego: Academic Press Inc., 1990.

[9] Moler, C. B. and G.W. Stewart, "An Algorithm for Generalized Matrix Eigenvalue Problems", *SIAM J. Numer.* Anal., Vol. 10, No. 2, April 1973.

[10] H. Akaike, A New Look at the Statistical Identification Model, *ZEEE Trans. On Automatic Control,* vol. 19, pp. 716-732, 1974.

[11] A. B. Koehler, and E. H. Murohee, A Comparison of Akaike and Schwarz Criteria for Selecting Model Order, *Applied Statistics,* vol. 37, pp. 187-195, 1988.

[12] *G.* Soromenho, Comparing Approaches for Testing the Number of Components in a Finite Mixture Model, *Computational Statistics,* vol. 9, pp. 65-78, 1993.

[13] *G.* Celeux, and G. Soromenho, An Entropy Criterion for Assessing the Number of Clusters in a Mixture Model, *Classification Journal,* vol. 13, pp. 195-212, 1996.

[14] R. Kass, and A. E. Raftery, Bayes factors, *Journals of the Anzerican Statistical Association,* vol. 90, pp. 773-795, 1995.

[15] G. Schwarz, Estimating the Dimension of a Model, *Annals of Statistics,* vol. 6, pp. 461-464, 1978.

[16] C. Biernacki, G. Celeux, and G. Govaert, An Improvement of the NEC Criterion for Assessing the Number of Clusters in a Mixture Model. *Pattern Recognition Letter,* vol. 20, pp. 267-272, 1999.

[17] C. Biernacki, and G. Govaert, Using Classification Likelihood to Choose the Number of Clusters, *Computing Science and Statistics,* vol. 29, pp. 451-457, 1997.

[18] C. Biernacki, G. Celeux, and G. Govaert, Assessing a Mixture Model for Clustering with the Integrated Classification Likelihood. *Technical Report No.3521*, Rhone-Alpes:INRIA,1998.

[19] G. McLachlan and D. Peel, Finite Mixture Models, New York: John Wiley & Sons Inc., 2000.

[20] A. Campbell, "Canonical Variate Analysis—A General Model Formulation," *Australian J. Statistics,* vol 26, pp.86-96, 1984.

[21] R, P. W. Duin and R. Haeb-Umbach, "Multiclass Linear Dimension Reduction by Weighted Pairwise Fisher Criteria," *ZEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, pp. 762-766, 2001.

[22] Chulhee Lee and David A. Landgrebe, " Feature Extraction and Classification Algorithms For High Dimensional Data", PhD thesis, Purdue University, December 1992. ECE Technical Report TR-EE 93-1.

[23] K. Fukunaga and M. Mantock, Nonparametric Discriminant Analysis, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 5, pp. 671-678, 1983.

[24] D. Landgrebe, On the Relationship Between Class Definition Precision and Classification Accuracy in Hyperspectral Analysis, Proceedings *of* the International Geoscience and Remote Sensing Symposium, Honolulu, Hawaii, July 24-28, 2000.

[25] H. Lutkepohl, Handbook of Matrices, Chichester: John Wiley & Sons, 1996.

[26] Qiong Jackson and David Landgrebe, "A Self-Improving Classifier Design for High-Dimensional Data Analysis with a Limited Training Data Set," Proceedings of the International Geoscience and Remote Sensing Symposium, Sydney Australia, July 9-13, 2001.