ECE Technical Reports

Electrical and Computer Engineering

11-1-2001

Design of an Adaptive Classification Procedure for the Analysis of High-Dimensional Data with Limited Training Samples

Qiong Zhang Jackson
Purdue University School of ECE

David Landgrebe
Purdue University School of ECE

DESIGN OF AN ADAPTIVE CLASSIFICATION PROCEDURE FOR THE ANALYSIS OF HIGH-DIMENSIONAL DATA WITH LIMITED TRAINING SAMPLES

QIONG ZHANG JACKSON
DAVID LANDGREBE

TR-ECE 01-5
NOVEMBER 2001

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

DESIGN OF AN ADAPTIVE CLASSIFICATION PROCEDURE FOR THE ANALYSIS OF HIGH-DIMENSIONAL DATA WITH LIMITED TRAINING SAMPLES

Qiong Zhang Jackson
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.

TABLE OF CONTENTS

ABSTRACT

In a typical supervised classification procedure the availability of training samples has a fundamental effect on classifier performance. For a fixed number of training samples classifier performance is degraded as the number of dimensions (features) is increased. This phenomenon has a significant influence on the analysis of hyperspectral data sets where the ratio of training samples to dimensionality is small. Objectives of this research are to develop novel methods for mitigating the detrimental effects arising from this small ratio and to reduce the effort required by an analyst in terms of training sample selection. An iterative method is developed where semi-labeled samples (classification outputs) are used with the original training samples to estimate parameters and establish a positive feedback procedure wherein parameter estimation and classification enhance each other in an iterative fashion. This work is comprised of four discrete phases. First, the role of semi-labeled samples on parameter estimates is investigated. In this phase it is demonstrated that an iterative procedure based on positive feedback is achievable. Second, a maximum likelihood pixel-wise adaptive classifier is designed. Third, a family of adaptive covariance estimators is developed that combines the adaptive classifiers and covariance estimators to deal with cases where the training sample set is extremely small. Finally, to fully utilize the rich spectral and spatial information contained in hyperspectral data and enhance the performance and robustness of the proposed adaptive classifier, an adaptive Bayesian contextual classifier based on the Markov random field is developed.

CHAPTER 1: INTRODUCTION

1.1 Statement of Problem

Remote sensing technology involves the measurement and analysis of the electromagnetic radiation reflected or emitted from the earth's surface by a passive or an active source. The radiation responses in various wavelengths reveal the types or properties of the materials on the surface being measured and collectively form a multispectral image. Previously, multispectral scanners were developed which measured radiation in 3 to 12 spectral bands. Current sensors can collect data in hundreds of spectral bands and then generate hyperspectral data. For instance, the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) gathers data in 210 spectral bands covering 0.4-2.5 um wavelength region with 20 m spatial resolution. The objective of analysis is to associate each pixel in a multispectral image with a class category of interest. Using a statistical pattern recognition approach, the spectrum of a pixel in a multispectral image is represented as an n-dimensional random process and analyzed subsequently. Fig. 1.1 illustrates a portion of hyperspectral image (210 bands) and a data representation of one of pixels.

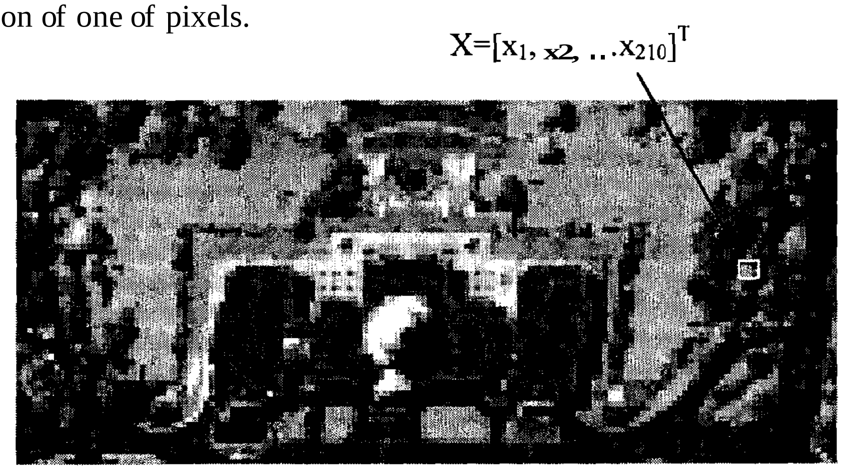
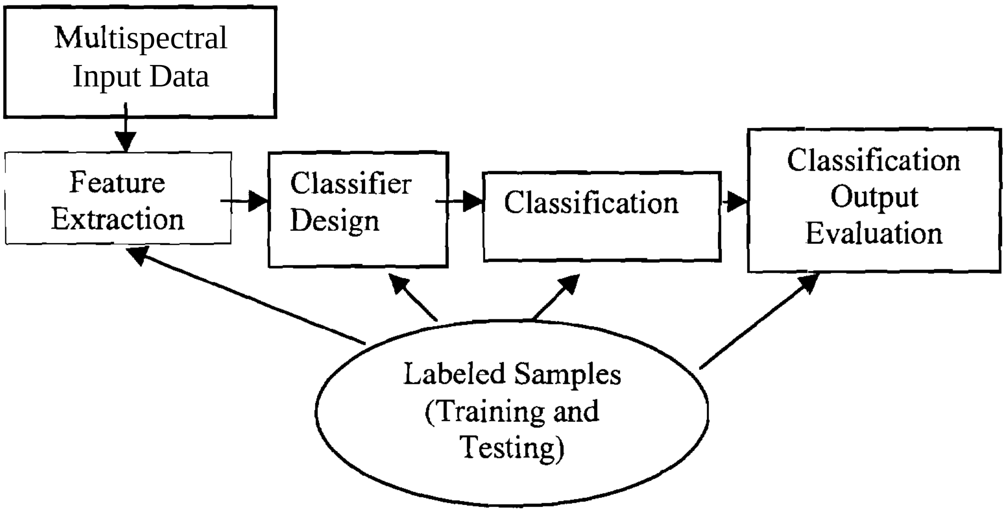


Fig. 1.1. A portion of Hyperspectral image. (in color)

Usually class statistics of interest are unknown and they may be estimated from pixels with known class origin. The pixel or sample with known class origin is referred as a labeled sample, and subsequently a sample with unknown class origin is referred as an unlabeled sample. A labeled sample can be used either to estimate class statistics (a training sample) and to test the quality of estimated statistics (a testing sample). **A** process using training samples to estimate class statistics is called a supervised learning. A typical supervised classification process for analyzing multispectral data is shown in Figure 1.2.



Fig, 1.2. **A** typical supervised classification process

Before classifying the multispectral data, some form of processing is usually performed on the data. The purpose of this process is to obtain a better representation of data based on the available training samples in preparation for classification. If the probability density functions (pdf's) of the classes are assumed, a better representation usually means a good set of parameter estimates for the pdf's. Due to limited training sample size, the common approach in remote sensing is to estimate class statistics up to second order, and consequently a sequence of normal distributions of classes are usually assumed. The processing stage may then involve, for example, (1) regularized covariance estimation which the number of estimated parameters for covariance matrices are reduced to decrease the variance of estimation, or (2) statistics enhancement using an expectation maximization (EM) algorithm where unlabeled samples in additional to ,trainingsamples are used to estimate parameters, and (3) feature extraction which extracts the features most significant to discriminate classes.

The classifier types can be broadly divided into two categories: pixel-wise classifier and spectral-spatial classifier. A pixel-wise classifier assigns each pixel to one of the candidate classes by a pre-specified decision rule based on the spectral measurement only. In general, the decision rule can be written as a function of a set of parameters contained in the pdf. In a spectral-spatial classifier, in addition to the spectral information, spatial information which represents the interaction of adjacent pixels is either incorporated into a statistical test rule to group adjacent pixels into an object or into the decision rule to represent a prior about the class distribution. The former is usually referred as the sample classifier where the resulting objects are eventually classified, and the latter is called a contextual classifier.

After a classifier is designed, it is usually tested by measuring the error probability. This might be estimated by using the labeled samples. In practical situations, the number of these labeled samples is limited, so one must decide how to divide them between both the design and test of the classifier. An unbiased estimator is obtained by using a set of samples to design a classifier and the other statistically independent set of samples for testing the classifier. This approach, called the holdout method [1], will be adopted for this thesis.

The increase in spectral resolution provided by the new sensor technology has brought about new potentials and challenges to data analysts. On one hand, the availability of a large number of spectral bands makes it possible to identify more detailed classes with higher accuracy than would be possible with the data from earlier sensors. On the other hand, in order to fully utilize the information contained in the new feature measurements, training samples are needed from all of the classes of interest. A large number of classes of interest, and a large number of spectral bands to be used, require a large number of labeled samples. For remote sensing applications, the ground truth information and hence labeled samples may be obtained in any of several ways. For example, by visual inspection of the actual site at the time the data are being gathered, or by matching the spectral responses of the samples against the responses of known samples [2], or by visually inspecting pixels from the image with high spatial resolution. Unfortunately, in any case, the necessary number of labeled samples for designing and testing the classifier are usually very expensive or time consuming to acquire. As a result, the class statistics have to be estimated by a limited training sample set. When the ratio of the number of training samples to the number of features is small, the parameter estimates become highly variable. These poorly estimated statistics might cause both feature extraction and classification performance to deteriorate.

Typically, the performance of the classifier improves up to a certain points as additional features are added, and then deteriorates. This is referred as the Hughes phenomenon [3] or peak phenomenon as shown in Figure 1.3. The Hughes phenomenon can be simply explained as follows: The most commonly used supervised classifiers estimate the unknown parameters and plug them in for the true parameters in the class-conditional densities. For a fixed sample size, as the number of features is increased (with a corresponding increase in the number of unknown parameters), even though the separability may increase as illustrated in Figure 1.4a and it may potentially improve classifier performance, the reliability of the parameter estimates decreases as shown in Figure 1.4b. Consequently, the performance of the resulting plug-in classifiers, for a fixed sample size, may degrade with an increase in the number of features as illustrated in Figure 1.4c.

The number of training samples required for different classifiers to obtain reasonable parameter estimates has been studied [1]. Loosely speaking, for a linear classifier the number of training samples should be proportional to the number of features; and for a quadratic classifier, the number of training samples should be proportional to the square of the number of features.

The additional problem that usually exists in remote sensing applications is the unrepresentative training sample problem. Since usually training samples are selected from spatially adjacent regions, they may not be good representatives of the samples of the entire same class that might exist in all regions in the scene. This problem further aggravates the difficulties in analyzing multispectral data.

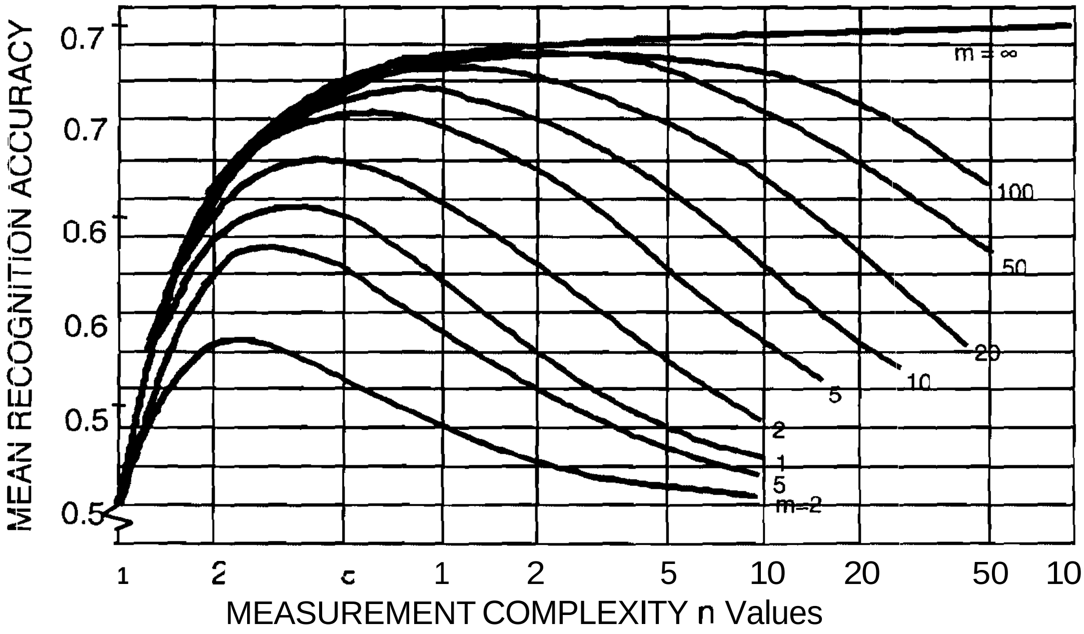


Fig. 1.3. Hughes phenomenon

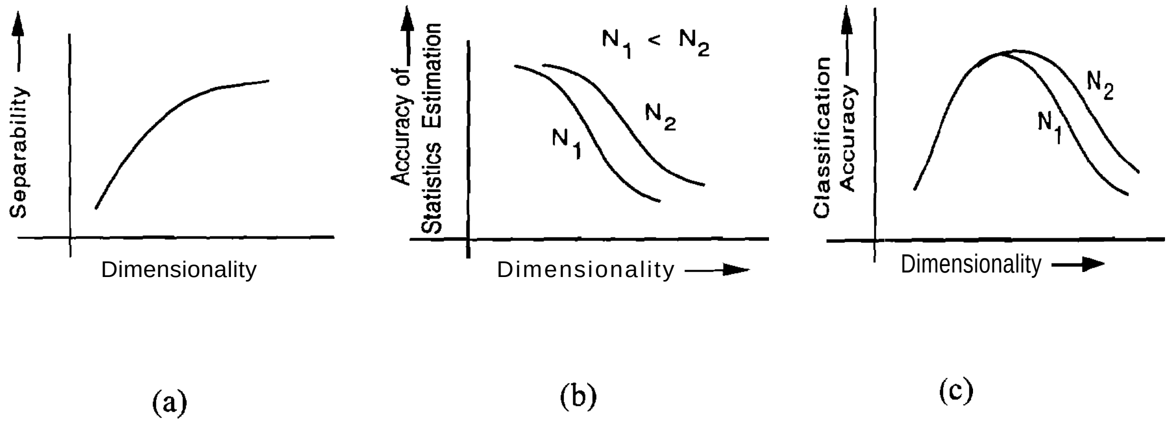


Fig. 1.4. Simple illustration of variation of (a) Separability (b) Reliability of statistics estimation and (c) Classification accuracy as function of dimensionality

1.2 Organization of Thesis

The objective of this research is: 1) to alleviate the Hughes phenomenon by extracting additional class label information from data and utilize it to enhance the statistics estimates and then improve classification accuracy; 2) to design a robust classification procedure where only minimum analyst effort is required in terms of the quantity and quality of training samples selected.

With this goal in mind, in this thesis a general adaptive classification procedure and then three specific ways to implement this procedure are developed to accommodate various training sample sizes. In this adaptive classification procedure, the semi-labeled sarnples (classification outputs) in addition to the original training samples are utilized to estimate parameters in order to establish a positive feedback procedure established where parameter estimation and classification enhanced each other at each iteration. Eventually, a higher classification accuracy may be achieved iteratively starting with a small training sample set. This thesis is organized as follows.

In Chapter 2, the rule of semi-labeled samples on parameter estimates and the feasibility of establishing the positive feedback procedure are investigated. In Chapter 3, a maximum likelihood pixel-wise adaptive classifier is designed. In order to control the influence of semi-labeled samples, the proposed method gives full weight to the training samples and reduced weight to semi-labeled samples. This method works well for the case when the number of training samples is slightly higher than the number of dimensions.

When the number of training samples is comparable or even smaller than the number dimensions (referred as to ill-posed problem), covariance estimates become highly variable and using semi-labeled samples alone is not adequate to maintain the classification performance. In Chapter 4, a family of adaptive covariance estimators is developed that combines the adaptive classifiers and covariance estimators, where the semi-labeled samples (whose labels are determined by a decision rule) are incorporated in the process of determining the optimal regularized parameters and estimating those supportive covariance matrices that form final covariance estimators.

In Chapter 5, to full utilize the rich spectral and spatial information contained in hyperspectral data and enhance the performance and robustness of the proposed adaptive classification procedure, an adaptive Bayesian contextual classifier based on the Markov random field is then developed. This method is advantageous when segmented image has strong short distance statistics dependence and when different classes have similar spectral response but may be able to be differentiated by their locations.

Finally, general conclusions and potentials for future research clevelopment future research are suggested in Chapter 6.

CHAPTER 2: EFFECT OF SEMI-LABELED SAMPLES IN REDUCING THE SMALL SAMPLE SIZE PROBLEM AND MITIGATING THE HUGHES PHENOMENON

2.1 Introduction

In a typical supervised classification problem, the objective is to assign a class label, from a set of candidate labels, to an incoming observation. The minimum expected error that can be achieved in performing the classification process is referred to as the Bayes error [1], which can be achieved by a decision rule that assigns a sample to the class with maximum a posteriori probability (The MAP classifier). In order to design such a classifier, knowledge of the a posteriori probabilities and the class-conditional probability density functions is required. If such knowledge is available then by increasing the dimensionality of the data, one would expect to enhance the performance. In other words, the Bayes error is a non-increasing function of the dimensionality of the data. After all, a new feature can only add information about a sample and thus, one would expect to do at least as well as if such information is not available. In practice, however, class conditional probability density function (pdf's) need to be estimated from a set of training samples. When these estimates are used in place of the true values of the pdf's the resulting decision rule is sub-optimal and hence has higher probability of error. The expected value of the probability of error taken over all training sample sets of a particular size is, therefore, larger than the Bayes error. When a new feature is added to the data the Bayes error decreases, but at the same time the bias of the classification error increases. This increase is due to the fact that more parameters must be estimated from the same number of training samples. If the increase in the bias of the classification error is more than the decrease in the Bayes error, then the use of the additional feature degrades the performance of the decision rule. This effect is called the Hughs phenomenon [3]. The larger the number of the parameters that need to be estimated, the more severe the Hughes phenomenon can become. Therefore, when the dimensionality of data and complexity of the decision rule increase, the Hughs effect becomes more severe.

In this chapter, a quantitative comparison between training samples and semi-labeled samples (definition will be given in the following section) on their value in reducing the classification error is provided. The information available:for estimating the parameters of a mixture of two normal distributions is examined for training samples and semi-labeled samples. The error bounds of some classifiers are obtained when supervised, semi-supervised, and combined supervised-semi-supervised learning (definition will be given in the following section) are used to perform the classification.

2.2 Definitions

- Labeled samples: samples whose class labels are correctly known
- Training samples: labeled samples which are used for training a classifier, i.e., estimating class conditional statistics if the class pdf are assumed
- Testing samples: labeled samples which are used to test performance of a classifier
- Unlabeled samples: samples whose class labels are completely unknown
- Semi-labeled samples: samples whose class labels are determined by a decision rule. They are unlabeled samples before classification is performed and their class label information partially obtained after classification. The label for a semi-labeled sample can be either right or wrong.
- Supervised learning: the training samples are used to train a classifier, estimating the parameters in the decision rule
- Semi-supervised learning: the semi-labeled samples are used to train a classifier
- Combined supervised-semi-supervised learning: the semi-labeled samples together with the training samples are used to train a classifier

2.3 Effect of Additional Semi-Labeled Samples

Consider a classification problem involving L classes with prior probabilities P_i and probability density functions $f_i(x)$, and each class is Gaussianly distributed. We denote e^* the Bayes error achieved by using the MAP classifier with given P_i and $f_i(x)$. Let θ denote the vector of parameters of the MAP classifier, e.g. mean vectors and covariance matrices and the associated prior probabilities. Let θ^* denote the true value of θ, and e 'the Bayes error obtained by using θ^* in the decision rule is e *. Assuming that $\hat{\theta}$is an estimate of θ^* and the deviation of θ from θ^* is small, we can approximate the error corresponding to the decision rule obtained by using $\hat{\theta}$ by a Taylor series expansion of up to the second term:

$$\hat{e} = e(\hat{\theta}) \approx e^* + \frac{\partial^T e(\theta)}{\partial \theta}\Big|_{\theta=\theta^*} (\hat{\theta} - \theta^*) + \frac{1}{2} tr\{\frac{\partial e^2(\theta)}{\partial \theta^2}\Big|_{\theta=\theta^*} (\hat{\theta} - \theta^*)(\hat{\theta} - \theta^*)^T\} \qquad (2.1)$$

Where $tr(A)$ designates the trace of the matrix A, and T stands for transpose. The second term vanishes because θ^* is an extreme point of $e(\theta)$. If the bias of $\hat{\theta}$ is zero or can be ignored ($E\{\hat{\theta}\} \approx \theta^*$), Then the expected value of e can be approximated at follows:

$$E\{\hat{e}\} \approx e^* + \frac{1}{2} tr\{\frac{\partial^2 e(\theta)}{\partial \theta^2}\Big|_{\theta=\theta^*} \operatorname{cov}(\hat{\theta})\} \qquad (2.2)$$

Notice that the bias term on the right side of the above equation (2.2) is non-negative, because it is the trace of the product of two positive semi-definite matrices [4]. With the increase of the number of the parameters (B), the covariance estimate becomes more variable, which causes the number of terms in the bias to go up and hence the expected value of the error increases, too. If this increase is not canceled by the decrease in the Bayes error that the additional parameters may provide, the overall classification performance degrades. Hence the Hughes phenomenon occurs. However, if additional information is utilized, such as the information contained in the semi-labeled samples, more accurate estimates with lower covariance matrices may be obtained, and the bias in the classification error may be reduced and then the Hughes phenomenon may be mitigated.

Consider two different estimators, $\hat{\theta}$ and $\tilde{\theta}$ with negligible biases, and assume that $\operatorname{cov}(\hat{\theta}) \geq \operatorname{cov}(\tilde{\theta})$ (i.e., $\operatorname{cov}(\hat{\theta}) - \operatorname{cov}(\tilde{\theta})$is positive semi-definite). Since θ^* is the global minimum of $e(\theta)$, the Hessian $\frac{\partial^2 e(\theta)}{\partial^2 \theta}\Big|_{\theta=\theta^*}$ is positive definite and we have:

$$tr\{\frac{\partial^2 e(\theta)}{\partial\theta^2}\bigg|_{\theta=\theta^*}.(\mathrm{cov}(\hat{\theta})-\mathrm{cov}(\tilde{\theta}))\}\geq 0$$

Furthermore, since the trace of the produce of two positive semi-definite matrices is non-negative [4], the above can be written as:

$$tr\{\frac{\partial^2 e(\theta)}{\partial\theta^2}\bigg|_{\theta=\theta^*}.\,\mathrm{cov}(\hat{\theta})\}-tr\{\frac{\partial^2 e(\theta)}{\partial\theta^2}\bigg|_{\theta=\theta^*}.\,\mathrm{cov}(\tilde{\theta})\}=tr\{\frac{\partial^2 e(\theta)}{\partial\theta^2}\bigg|_{\theta=\theta^*}.(\mathrm{cov}(\hat{\theta})-\mathrm{cov}(\tilde{\theta}))\}\geq 0$$

Therefore, the expected error by using $\hat{\theta}$ is greater than the expected error by using $\tilde{\theta}$ in the decision rule, i.e.:

$$E\{\hat{e}\}\geq E\{\tilde{e}\} \qquad (2.3)$$

In the following we will show that, by using additional semi-labeled samples, estimates with smaller covariance matrices can be found. Therefore, better performance can be obtained without the extra cost of selecting more training samples.

Assume that an estimate $\hat{\theta}$ of θ^* is obtained by using the training samples. Furthermore, assume that $\ddot{\theta}$ is asymptotically unbiased and efficient (for example, maximum likelihood estimates always posses these properties [5]). In other words, for a moderately large sample size we have $E\{\hat{\theta}\}\approx\theta^*$ and $\mathrm{cov}(\hat{\theta})\approx I_s^{-1}$, where I_s^{-1} is the inverse of the Fisher information matrix [5]. The subscript "s" denotes that the Fisher information matrix corresponding to a supervised estimator obtained by using training samples that are drawn separately from each class. The Fisher information matrix is positive semi-definite and is defined as follows:

$$I=E\{[\frac{\partial}{\partial\theta}\log f(x)][\frac{\partial}{\partial\theta}\log f(x)]^T\} \qquad (2.4)$$

Now, assume that $\tilde{\theta}$ is another estimate of θ^* obtained by using some semi-labeled samples in addition to the training samples. The semi-labelled samples are selected separately from each class. If $\tilde{\theta}$ is also asymptotically unbiased and efficient, the we have $\mathrm{cov}(\hat{\theta})\approx I_c^{-1}$, where I_c is the Fisher information matrix corresponding to the estimate obtained by using both training samples and semi-labeled samples. Provided that the semi-labeled and training samples are independent, one can write:

$$I_c = I_s + I_{sl}$$

where I_{sl} denotes another Fisher information matrix corresponding to the information contained in the semi-labeled samples for estimating θ^*. Since all of the Fisher information matrices are positive semi-definite one can obtain $I_c \geq I_s$, and hence $\mathrm{cov}(\hat{\theta}) - \mathrm{cov}(\tilde{\theta})$. Therefore, one can conclude that using additional semi-labeled samples, a smaller expected error may be obtained.

2.4 Information of Two Normal Distributions

In this section, the information available for estimating the parameters of a mixture of two normal distributions is examined in terms of the Fisher information matrix, denoted by I,. According to Crame-Rao inequality [5], if $\hat{\theta}$ is any absolutely unbiased estimator of θ based on measure data, then the covariance of the error in the estimator is bounded below by the inverse of the Fisher information matrix, assuming it exists. Furthermore, if $\ddot{\theta}$ is asymptotically (a large sample size) unbiased and efficient (for example, maximum likelihood estimates always possess these properties [5]), then $\mathrm{cov}(\hat{\theta}) \approx I_s^{-1}$. Loosely speaking, with more information available, then the determinant and trace of the inverse of the Fisher information matrix become smaller, and correspondingly, the covariance of an unbiased estimator is smaller too. In other words, the estimator becomes more stable.

Consider a classification problem involving two multivariate classes that can be represented as Gaussian distributions with probability density functions (pdf's) $f_i(x|\mu_i, \Sigma_i), i = 1,2$, where μ_i, and Σ_i denote the mean vector and covariance matrix of class i. The prior probabilities associated with the two classes are designated by P_1 and P,. We consider the following case: n independent unlabeled observations ($X_1, X_2, \ldots, X_n$) are drawn from the mixture of these two classes, and are subsequently classified as class one (C,) and class two (C,) based on the Bayes decision rule which assigns an observation to the class with the highest a posteriori probability for minimizing the total classification error:

$$\begin{aligned} X \in \Omega_1 &\Leftrightarrow P_1 f_1(x) \geq P_2 f_2(x) \\ X \in \Omega_2 &\Leftrightarrow P_1 f_1(x) < P_2 f_2(x) \end{aligned} \tag{2.5}$$

where Ω_1 and Ω_2 are two sub-spaces corresponding to class one and class two respectively. Suppose n, samples are correctly classified, and n_2 samples are rnisclassified, i.e., n, + n_2 = n. Denoting I,, as the Fisher information matrix for this case, using the definition of Fisher information matrix given by Eq. (2.4), then we have:

$$\begin{aligned} I_{sl} &= nE\left\{[\frac{\partial}{\partial\theta}\log f(x,\theta)][\frac{a}{ae}\log f(x,\theta)]^T\right\} \\ &= n_1P_1E\left\{[\frac{\partial}{\partial\theta}\log f(x,\theta)][\frac{\partial}{\partial\theta}\log f(x,\theta)]^T \middle| x\in\Omega_1, x \text{ is } C_1\right\} \\ &+n_1P_2E\left\{[\frac{\partial}{\partial\theta}\log f(x,\theta)][\frac{\partial}{\partial\theta}\log f(x,\theta)]^T \middle| x\in\Omega_2, x \text{ is } C_2\right\} \\ &+n_2P_1E\left\{[\frac{\partial}{\partial\theta}\log f(x,\theta)][\frac{a}{\partial\theta}\log f(x,\theta)]^T \middle| x\in\Omega_2, x \text{ is } C_1\right\} \\ &+n_2P_2E\left\{[\frac{\partial}{\partial\theta}\log f(x,\theta)][\frac{\partial}{\partial\theta}\log f(x,\theta)]^T \middle| x\in\Omega_1, x \text{ is } C_2\right\} \end{aligned} \tag{2.6}$$

Without loss of generality, consider the canonical form where μ_l=0, and μ_2=[Δ 0...0]^T and $\Sigma_l=\Sigma_2=I_d$, Δ>0, Δ^2 is the Mahalanobis distance between the: two classes, and I_d is a d × didentity matrix (d is the dimension of the feature space). Since the error rate of probability is the subject of our study in the next section and is invariant under nonsingular linear transformation, the canonical form can be used here without loss of generality. Any other two-class problem for which $\Sigma_l=\Sigma_2$ can be transformed into the above form through a linear transformation [1]. Using these conditions, Eq. (2.6) can be simplified as follows (the detailed derivation is shown at appendix **A):**

$$I_{sl} = n\begin{bmatrix} P_1k_1 & & & \\ & P_1k_2I_{d-1} & & \\ & & P_2k_3 & \\ & & & P_2k_4I_{d-1} \end{bmatrix} \tag{2.7}$$

where

$$k_1 = r_c\alpha_1 + (1-r_c)(1-a,)$$

$$k_2 = r_c\beta_1 + (1-r_c)(1-\beta_1)$$

$$k_3 = r_c\alpha_2 + (1 - r_c)(1 - \alpha_2)$$

$$k_4 = r_c\beta_2 + (1 - r_c)(1 - \beta_2)$$

$$r_c = \frac{n_1}{n}$$

$$\alpha_1 = \Phi(t) - t\phi(t)$$

$$\beta_1 = \Phi(t)$$

$$\alpha_2 = \Phi(\Delta - t) - (t - \Delta)\phi(t - \Delta)$$

$$\beta_2 = \Phi(\Delta - t)$$

$$\Phi(t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{t} e^{-\frac{x^2}{2}} dx$$

$$\phi(t) = \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}}$$

Here $\Phi(t)$ and $\phi(t)$are the cumulative distribution function (cdf) and probability density function (pdf) of the standard normal distribution respectively, and r_c is the classification accuracy. From equation (2.7) we can derive the following interesting results:

1) If two classes are quite separated, i.e., **A** $>> 1$, then $t >> 1$ and hence $\Phi(t) \approx 1$ and $t\phi(t) \approx 0$, $a, \approx a, \approx \beta_1 = \beta_2 \approx 1$. **In** this case, equation (2.7) can be simplified as:

$$I_{st} \approx n_1 \begin{bmatrix} P_1 I_d & 0 \\ 0 & P_2 I_d \end{bmatrix} \leq n \begin{bmatrix} P_1 I_d & 0 \\ 0 & P_2 I_d \end{bmatrix} \qquad (2.8)$$

where the above inequality is a matrix inequality indicating that the right hand side minus the left hand side is a positive semi-definite matrix. Notice that the right hand side of the above inequality is the Fisher information matrix for estimating θ if the n randomly drawn samples have been labeled. In particular, let I_s be the information matrix for this case. One can write:

$$\begin{aligned} I_s &= n\{P_1 E\left\{\left[\frac{\partial}{\partial\theta}\log f_1(x)\right]\left[\frac{\partial}{\partial\theta}\log f_1(x)\right]^T \mid x \text{ is } \mathrm{C}_1\right\} \\ &+ P_2 E\left\{\left[\frac{\partial}{\partial\theta}\log f_2(x)\right]\left[\frac{\partial}{\partial\theta}\log f_2(x)\right]^T \mid x \text{ is } \mathrm{C}_2\}\right\}\} \\ &= n\begin{bmatrix} P_1 I_d & 0 \\ 0 & P_2 I_d \end{bmatrix} \end{aligned} \tag{2.9}$$

Therefore, inequality (2.8) reveals the conceptually appealing fact that the information contained in n classified observations based on the Bayes decision rule is less than or equal to that of n labeled observations. The missing information in this case using only semi-labeled samples (referred as semi-supervised learning) is due to the mis-assigned labels. From now on we refer to the right hand side of (2.8) as the "supervised bound" for I_{sl}. Usually, classification accuracy achieved by Bayes rule with known class condition probability density functions goes up with the separation of classes. Therefore, if two classes are quite separated, we have $n_1 >> n_2$ or n, ≈ n, leading to I„ ≈ I,, which implies more information can be gained from more correctly classified samples.

2) At the worst case where half of the samples are correctly classified and the remaining half are rnisclassified, i.e., $n_1 = n_2 = n/2$, I_{sl} can be written as:

$$I_{sl} = \frac{n}{2}\begin{bmatrix} P_1 I_d & 0 \\ 0 & P_2 I_d \end{bmatrix} = \frac{1}{2} I_s \tag{2.10}$$

This indicates that at least 50% of class label information is generated after classification.

In summary, for the canonical two component normal mixtures with unknown means, after the classification is performed based on the Bayes decision rule, the Fisher information matrix I„ is bounded as follows:

$$\frac{n}{2}\begin{bmatrix} P_1 I_d & 0 \\ 0 & P_2 I_d \end{bmatrix} \le I_{sl} \le \mathrm{n}\begin{bmatrix} P_1 I_d & 0 \\ 0 & P_2 I_d \end{bmatrix} \tag{2.11}$$

Under suitable regularity conditions the inverse of the Fisher information matrix (I-') is the asymptotic (large sample) variance-covariance matrix for the maximum likelihood estimates [4]. For the equal prior probability case (P_1=P_2=0.5), by inverting the bounds in **Eq.** (2.11), the asymptotic covariance of the ML (Maximum Likelihood) estimate of $0 = [\mu_1^T, \mu_2^T]^T$ can be bounded from above and below. Notice that for any two positive definite matrices **A** and B, if A≥B, then $B^{-1} \geq A^{-1}$ [5]. Denoting $\bar{\theta}$ as the ML estimate of 0 obtained by using semi-labeled samples, then $\text{cov}(\bar{\theta})$ is bounded as follows:

$$\text{cov}(\bar{\theta}) \leq \frac{1}{2n}\begin{bmatrix} \frac{1}{P_1} I_d & \\ & \frac{1}{P_2} I_d \end{bmatrix} \tag{2.12a}$$

and

$$\text{cov}(\bar{\theta}) \geq \frac{1}{n}\begin{bmatrix} \frac{1}{P_1 k_1} & & & \\ & \frac{1}{P_1 k_2} I_{d-1} & & \\ & & \frac{1}{P_2 k_3} & \\ & & & \frac{1}{P_2 k_4} I_{d-1} \end{bmatrix} \tag{2.12b}$$

Using || and tr to denote the determinant and trace operators respectively then $| I^{-1} |$ and $tr(I^{-1})$ represent the asymptotic generalized and total variance [6]. Using **Eq.** (2.11) we can obtain the trace and determinant of $(I_{sl})^{-1}$:

$$\begin{aligned} tr((I_{sl})^{-1}) &= \frac{1}{n}\left(\frac{1}{P_1 k_1} + \frac{d-1}{P_1 k_2} + \frac{1}{P_2 k_3} + \frac{d-1}{P_2 k_4}\right) \\ &= \frac{1}{n}\left[\left(\frac{1}{P_1 k_1} + \frac{1}{P_2 k_3}\right) + (d-1)\left(\frac{1}{P_1 k_2} + \frac{1}{P_2 k_4}\right)\right] \end{aligned} \tag{2.13a}$$

and

$$| (I_{sl})^{-1} | = \left(\frac{1}{P_1 P_2 n}\right)^d \left(\frac{1}{k_1 k_3}\right)\left(\frac{1}{k_2 k_4}\right)^{d-1} \tag{2.13b}$$

Fig. (2.1a) and (2.1b) illustrate the variation of asymptotic total variance with the accuracy, the number of samples, separations for semi-supervised learning (only semi-labeled samples are used) and supervised learning (only labeled samples are used). Note that accuracy achieved by Bayes rule is approximately 69% for Δ=1, and 99% Δ=5 with equal prior probabilities [1]. From these figures it is seen that 1) asymptotic total variance decreases with increase of classification accuracy. It drops faster when two classes are more separated; 2) The asymptotic total variance increases with increase of dimensionality, but decreases dramatically with increase of the number of samples; 3) The difference of asymptotic total variance using labeled and semi-labeled samples reduces with classification accuracy and separability of two classes.

The above results imply that when semi-labeled samples are used, 1) the improvement of classification accuracy may reduce the total variance and hence enhance the estimation of statistics, and in return, the enhanced statistics can further improve the classification accuracy. This implies when semi-labeled samples are used to integrate statistics estimation with classification, a positive feedback can be established where 2) Tht: large number of semi-labeled samples may significantly reduce the total variance and therefore mitigate the effect of small training sample size problem. 3) Semi-labeled samples can provide comparable class label information when two classes are quite separable and classification accuracy is high.

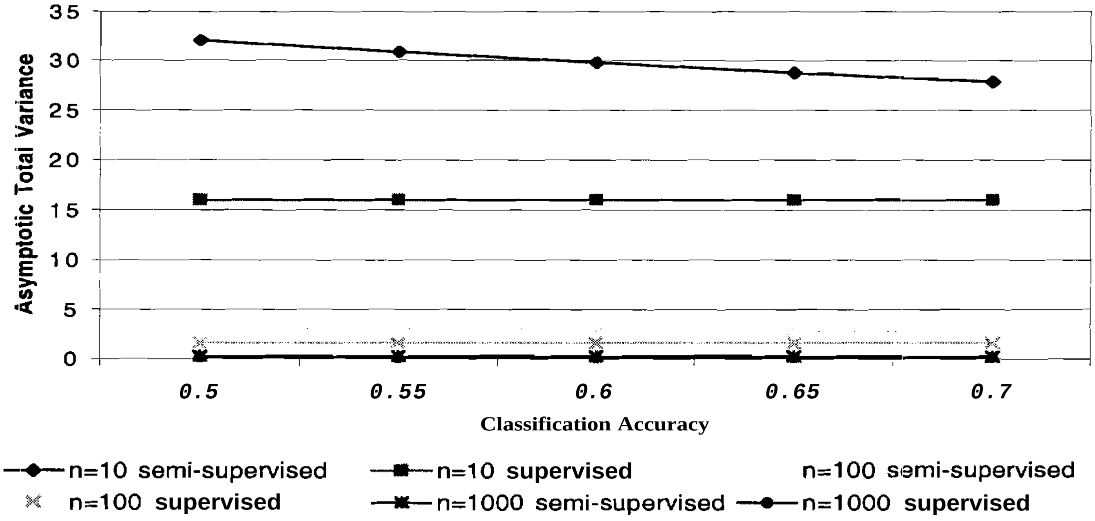
2.5 Bound on Probability of Error

2.5.1 Semi-supervised learning

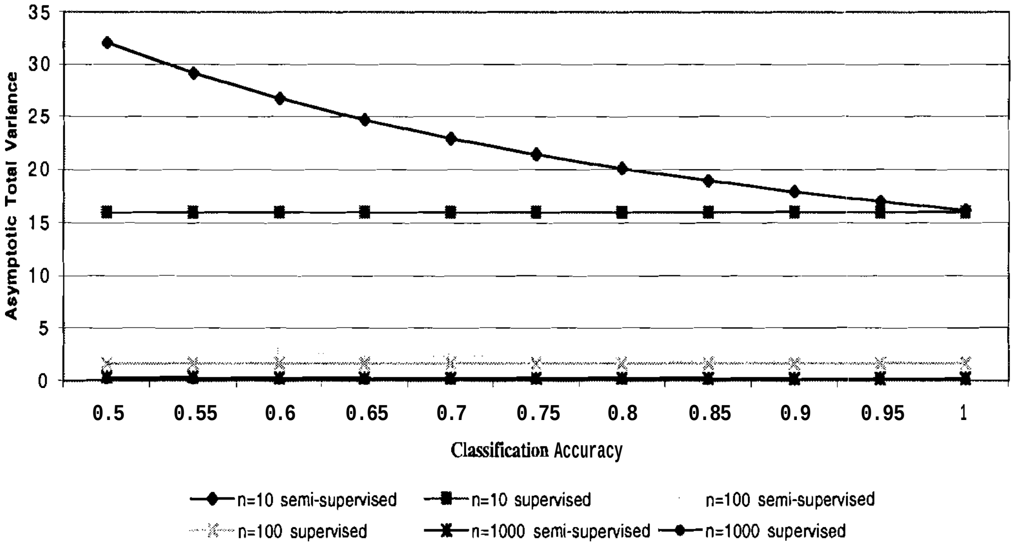
In the equal covariance case ($\Sigma_1 = \Sigma_2 = \mathrm{C}$), the optimal classifier is linear:

$$
\begin{aligned}
&h(x) = (\mu_2 - \mu_1)\Sigma^{-1} + \frac{1}{2}(\mu_1^T\Sigma \;\; \mu_1 - \mu_2^T\Sigma^{-1}\&) + \log\frac{P_2}{e} \\
&h\begin{cases} < 0 \Rightarrow class1 \\ > 0 \Rightarrow class2 \end{cases}
\end{aligned}
\tag{2.14}
$$

when the true parameter values are used to evaluate h(x), the above linear classifier minimizes probability of error, which is referred to as the Bayes probability of error.



(a) delta=1, d=40



(b) delta=5, d=40

Fig. 2.1. Asymptotic total variance using the semi-labeled. samples

If the parameters are replaced by their estimates in h(x), the error rises. The expected probability of error using estimated parameters can be written as [1]:

$$
\begin{aligned}
E\{\hat{err}\} &\approx \mathrm{err}^{*} + \frac{1}{2} tr\left\{ \frac{\partial^2 err}{\partial \theta^2}\bigg|_{\theta=\theta^*} \mathrm{cov}(\hat{\theta}) \right\} \\
&= err^{*} + \frac{1}{2} tr\left\{ \int \int_{-\infty}^{\infty} \frac{1}{j\omega} \left[\frac{\partial^2}{\partial \theta^2} e^{j\omega h(x)} \right]_{\theta=\theta^*} [P_1 f_1(x) - P_2 f_2(x)] dx d\omega \, \mathrm{cov}(\hat{\theta}) \right\} \qquad (2.15) \\
&= err^{*} + \frac{1}{2\pi} \int \int_{-\infty}^{\infty} \frac{1}{2} tr\left\{ \left[\frac{\partial^2 h(x)}{\partial \theta^2} + j\omega \frac{\partial h(x)}{\partial \theta} \frac{\partial h^T(x)}{\partial \theta} \right]_{\theta=\theta^*} \mathrm{cov}(\hat{\theta}) \right\} \\
&\times e^{j\omega h(x)} [P_1 f_1(x) - P_2 f_2(x)] dx d\omega
\end{aligned}
$$

For the canonical form where μ_1=0, and $\mu_2=[\Delta\ 0\ldots0]^T$, and $\Sigma_1=\Sigma_2=I_d$, $\Delta>0$, we have:

$$
\left[\frac{\partial h(x)}{\partial \theta} \frac{\partial h^T(x)}{\partial \theta} \right]_{\theta=\theta^*} = \begin{bmatrix} xx^T & -x(x-\mu_2)^T \\ -(x-\mu_2)x^T & (x-\mu_2)(x-\mu_2)^T \end{bmatrix} \qquad (2.16a)
$$

$$
\text{and} \quad \frac{\partial^2 h(x)}{\partial \theta^2}\bigg|_{\theta=\theta^*} = \begin{bmatrix} I_d & \\ & -I_d \end{bmatrix} \qquad (2.16b)
$$

The integrals in (2.16) can be computed by the method provided in [1]. Replacing $\mathrm{cov}(\hat{\theta})$ in (2.15) by its upper and lower bounds described in Eq. (2.12a) through Eq. (2.12b) leads to the following inequalities for the bias of $\hat{err}$:

$$
bias(\hat{err})\ 2\, \frac{1}{n\sqrt{2\pi}\Delta} e^{-\frac{\Delta^2}{8}} \left[\frac{\Delta^2}{4} + d - 1 \right] \text{(supervised lower bound)} \qquad (2.17a)
$$

$$
bias(\hat{err}) \geq \frac{1}{n\sqrt{2\pi}\Delta} e^{-\frac{\Delta^2}{8}} \left[\frac{\Delta^2}{8}\left(\frac{1}{k_3} + \frac{1}{k_1}\right) + \frac{(d-1)}{2}\left(\frac{1}{k_2} + \frac{1}{k_4}\right) \right] \qquad (2.17b)
$$

$$
bias(\hat{err}) \leq \frac{2}{n\sqrt{2\pi}\Delta} e^{-\frac{\Delta^2}{8}} \left[\frac{\Delta^2}{4} + d - 1 \right] \qquad (2.17c)
$$

Here the supervised lower bound is applied for supervised learning where n samples are labeled. It is possible to show that the variance of $\hat{err}$is $O(\frac{1}{n^2})$[5] and is therefore negligible.

Fig. (2.2a) and (2.2b) show the bounds on the number of semi-labeled samples required to maintain the bias of classification error to less than 1% when dimensionality varies. Fig. (2.3) shows the upper and lower bounds of the bias of the probability of error (in percent) versus A (Square root of the Mahalanobis distance), when P_1=P_2, d=4, and n=1000. Notice that as A goes up the semi-supervised curves gets closer to the supervised lower bound indicating when classes are far away from each other, semi-supervised learning can achieve comparable performance to supervised learning.

2.5.2 Combined Supervised and Semi-supervised learning

In practical applications, usually both training and semi-labeled samples are available. Assuming that the training and semi-labeled samples are statistically independent, one can write the Fisher information matrix corresponding to the combined supervised and semi-supervised learning as the sum of the Fisher information matrices corresponding to the training and semi-labeled samples. This implies that if both training samples and semi-labeled samples are used simultaneously to estimate: the parameters of tht: decision rule, better performance with lower bias and variance can be achieved than when using training samples alone [7]. By using the bounds obtained for the Fisher information matrix corresponding to the semi-labeled samples in equation (2.8), similar bounds can be obtained for the combined supervised and semi-supervised learning case. These bounds can then be utilized to determine the upper and lower bounds for bias of classification error as is done in the pervious section for the semi-supervised case.

Assume that in addition to the n semi-labeled samples, n_{1t} labeled samples from class 1 and n„ labeled samples from class 2 are also available for training the classifier. If the estimate of the parameter set $\theta = [\mu_1^T \ \mu_1^T]^T$ obtained by using all of these samples in the decision rule (10), the bias of the classification error, for the case P_1=P_2, is bounded as:

$$bias(e\hat{r}r) \geq \left(\frac{1}{n_{1t} + n/2} + \frac{1}{n_{2t} + n/2} \right) \frac{1}{4\sqrt{2\pi}\Delta} e^{-\frac{1}{8}\Delta^2} \left[\frac{\Delta^2}{4} + d - 1 \right] \quad (2.18a)$$

(supervised lower bound)

$$bias(\hat{err}) \geq \frac{1}{4\sqrt{2\pi}\Delta} e^{-\frac{\Delta^2}{8}}$$

$$\left[\frac{\Delta^2}{4} \left(\frac{1}{n_{1t} + \frac{n}{2}k_1} + \frac{1}{n_{2t} + \frac{n}{2}k_3} \right) + (d-1)\left(\frac{1}{n_{1t} + \frac{n}{2}k_2} + \frac{1}{n_{2t} + \frac{n}{2}k_4} \right) \right] \quad (2.18b)$$

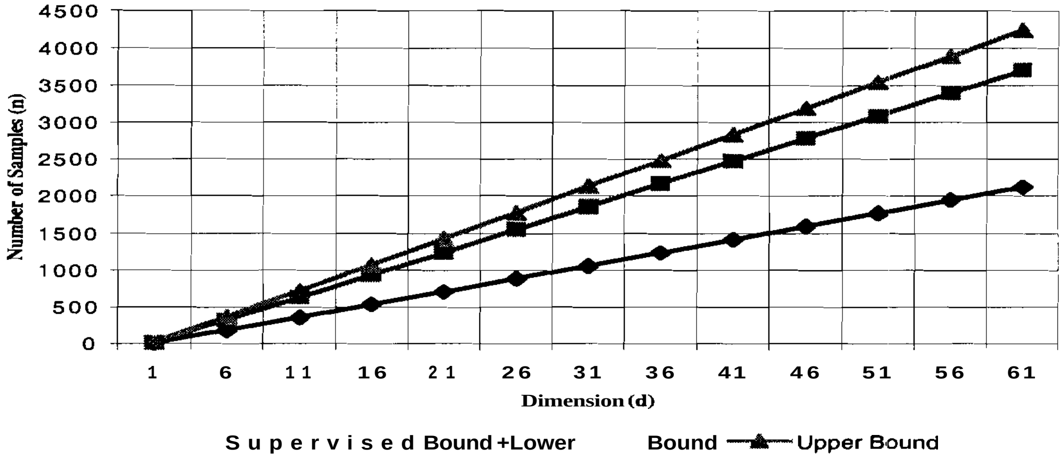
$$\geq \left(\frac{1}{n_{1t} + n/2} + \frac{1}{n_{2t} + n/2} \right) \frac{1}{4\sqrt{2\pi}\Delta} e^{-\frac{1}{8}\Delta^2} \left[\frac{\Delta^2}{4} + d - 1 \right]$$

$$bias(\hat{err}) \leq \left(\frac{1}{n_{1t} + n/4} + \frac{1}{n_{2t} + n/4} \right) \frac{1}{4\sqrt{2\pi}\Delta} e^{-\frac{1}{8}\Delta^2} \left[\frac{\Delta^2}{4} + d - 1 \right] \quad (2.18c)$$

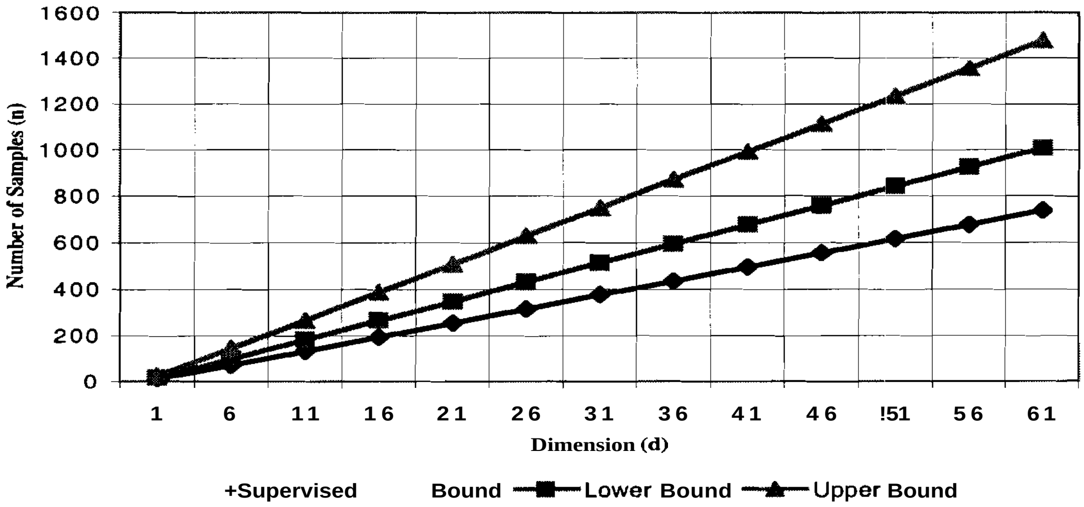
The variance of $\hat{err}$is again negligible since it is inversely proportional to the square of the number of training samples.

Figure (2.4) shows the bounds of the bias of the probability of error versus A when P_1=P_2, d=4, n=100, and n_{1t}=n_{2t}=10. The no-semi-labeled curve in this figure refers to the case when only labeled samples are used. It is seen that by using additional semi-labeled samples, the bias of the classification error is substantially reduced. The amount of the reduction depends on the separation between two classes as characterized by A.

In conclusion, semi-supervised learning can achieve comparable performance to supervised learning when the classes are relatively separated. When the classes are highly overlapped, a large number of semi-labeled samples are necessary for designing a classifier which matches the performance of the one designed by supervised learning. When both training and semi-labeled samples are available, the combined supervised and semi-supervised learning that uses these two kinds of samples can outperform supervised learning. This result is significant for the remote sensing applications where the number of training samples is usually limited compared to the dimensionality of data obtained by high spectral resolution sensors, while a large amount of semi-labeled samples are available after the classification is performed without additional effort. In such cases, utilizing semi-labeled samples may mitigate the Hughes phenomenon [1]. If we know which samples have been correctly classified and use them accordingly to re-estimate statistics in addition to original training samples, the estimated statistics should be more precise because the actual training samples have been enlarged. Since usually we have no knowledge of classification accuracy for each individual sample, the key is to design a scheme that is able to apply a control factor that is related to the likelihood of a semi-labeled sample to a class. In the next chapter, an adaptive classifier is designed to achieve this goal.



(a) delta=1



(b) delta=2

Fig. 2.2. Number of training samples for supervised learning and semi-labeled samples for semi-supervised learning required having bias (error) <1%.

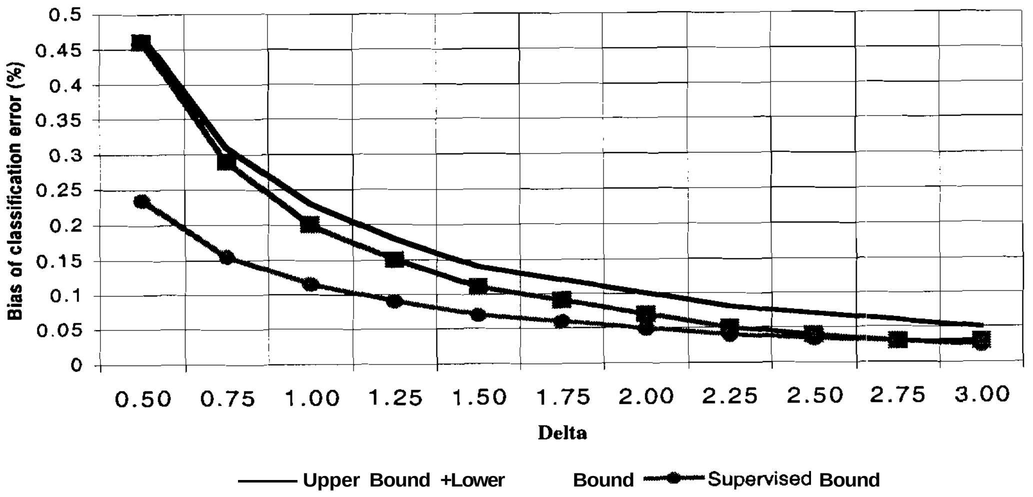


Fig. 2.3. Bounds on the bias of the classification error for semi-supervised learning. (P1=P2, d=4, and n=1000)

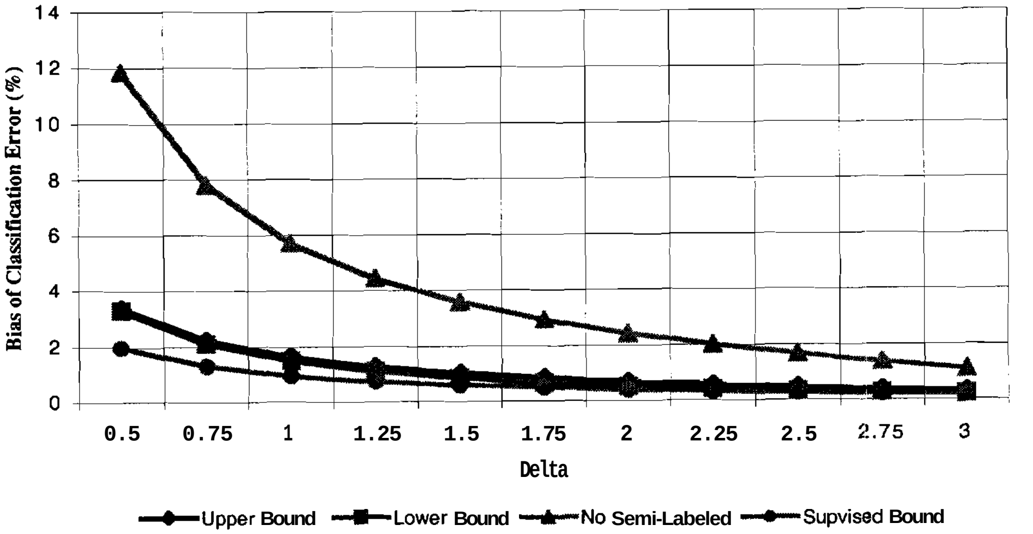


Fig. 2.4. Bounds on the bias of the classification error for combined learning.

CHAPTER 3: DESIGN OF AN ADAPTIVE CLASSIFIER

3.1 Introduction

In remote sensing applications, increased spectral resolution brought about by the current sensor technology has offered new potentials and challenges to data analysts. On one hand, the availability of a large number of spectral bands makes it possible to identify more detailed classes with higher accuracy than would be possible with the data from earlier sensors. On the other hand, a large number of classes of interest and a large number of spectral bands available require a large number of training samples, which unfortunately are expensive or tedious to acquire. As a result, the class statistics must be estimated from the limited training sample set. When the ratio of the number of training sarnples to the number of spectral features is small, the parameter estimates become highly variable, causing classification performance to deteriorate with increasing dimensionality. This phenomenon where with finite training samples, classifier performance raises with dimensionality at first and then declines, was studied in detail by Hughes [1], and is later referred to as the Hughes phenomenon.

An additional problem that usually exists in remote sensing applications is the unrepresentative training sample problem. Since usually training samples are selected from spatially adjacent regions, they may not be good representatives of the samples of the entire class, which is likely distributed over the entire scene. This problem further aggravates the difficulties in analyzing remote sensing data.

In Chapter 2, it has shown that using semi-labeled samples may reduce the variance of the parameter estimation. To mitigate the small training sample problem, a self-learning and self-improving adaptive classifier is proposed in this paper. This adaptive classifier enhances statistics estimation and hence improves classification accuracy iteratively by utilizing the semi-labeled samples, in addition to the original training samples, in subsequent statistics estimation. In this iterative process, samples are initially classified based on the estimated statistics using the original training samples only. Then the classified results are subsequently used with the original training samples to update class statistics, and the samples are reclassified by the updated statistics. This process is repeated until convergence is reached.

The proposed adaptive classifier potentially has the following benefits:

1) The large number of semi-labeled samples can enhance the statistics estimates, decreasing the estimation error and therefore reduce the effect of the small sample size problem, because the semi-labeled samples in effect enlarge the training sample size.

2) The estimated statistics are more representative of the true class distribution, because samples used to estimate statistics are from a larger portion of the entire data set.

3) This classifier is adaptive in the sense that it can improve the accuracy by using the information extracted from its output. With proper conditions, a positive feedback system can be formed, whereby better statistics estimation leads to higher classification accuracy, and in return, higher classification accuracy results in even better parameter estimation.

4) In a way, this approach augments automation of the classifier. It is possible that to start with a small number of training samples (minimum input from the analyst) this classifier may be able to continuously extract useful information from the data and adjust itself accordingly, and eventually evolve automatically to an optimal classifier which produces optimal classification accuracy with a given data set. Hence analyst's effort can be greatly reduced.

5) Since the semi-labeled samples can be fed back before or after any feature extraction is performed, it offers flexibility of implementation, that is, depending on the requirement of accuracy and the computation load, the semi-labeled samples can be used in more than one way.

3.2 Design of Adaptive Classifier

If we assume every sample in the data set is unique, i.e. it belongs only to one class, we would expect it should only contribute to statistics of the only class to which it belongs. In the EM algorithm [9] and its application in remote sensing [7][8], each unlabeled sample has a certain amount of membership for each class and correspondingly has weighted contribution to the statistics of every class. Even though this is reasonable at this point because the sample labels are completely unknown, the contribution of the sarnple to the class to which it does not belong is definitely undesired. This negative influence may be significant enough to cause the estimated statistic to deviate from the true one, especially when a large number of unlabeled samples are used, or there exists a class whose statistics are quite different from the rest of classes. For example, if the class proportion is quite unbalanced, i.e., a few classes are quite dominant in the given data set, then the large number of unlabeled samples used may be mostly frorn these dominant classes. With small numbers of training samples, the estimated statistics of minority classes will be overwhelmed by the unlabeled samples and consequently may deviate from the true one. This phenomenon has been observed in practice,, and it has been noticed that better classification accuracy could be achieved by using approximately the same number of unlabeled samples as the number of training samples, which is small. This is unfortunate because more information can be obtained and utilized with additional unlabeled samples [7][8].

In this section, an adaptive classifier based on the Maximum Likelihood (ML) rule is proposed to enhance the statistics estimation by using semi-labeled samples in addition to training samples. In this new classifier, the partial information of the class label obtained in the process of classification is utilized in such a way that each semi-labeled sample only affects the statistics of the class into which it has been partitioned. Furthermore this classifier assigns full weight to training samples, but automatically gives reduced weight to semi-labeled samples. Therefore, it utilizes the additional class label information provided by correctly classified semi-labeled samples and at the same time limits the undesired influence from misclassified samples. Before we describe the proposed adaptive classifier, we first provide a brief review of Expectation Maximization (EM) algorithm.

The EM algorithm is an iterative method for numerically approximating the maximum likelihood (ML) estimates of the parameters in a mixture model. Under the mixture model, the distribution of an observation $x \in R^P$ is given as:

$$f(x \mid \Phi) = \sum_{i=1}^{L} \alpha_i f_i(x \mid \phi_i)$$

where α_1, ..., α_L are the class prior probabilities and thus the mixing proportions, f_I is the component density parameterized by ϕ_I and L is the total number of components. The mixture density f is then parameterized by $\Phi = (\alpha_1, ..., \alpha_L,\ \phi_1, ...\phi_L)$.

Assume that $y = (y, ,..., y_{mi})$ are the m_i training samples from class i. Also, there are L classes and a total of n unlabeled samples denoted by $x = (x_1, ..., x_n)$. The parameter set Φ then contains all the prior probabilities, mean vectors and covariance matrices. Assume that $\phi_1, ..., \phi_L$ are mutually independent. The EM algorithm can then be expressed as the following iterative equation [9]:

E-step:

$$\tau_{ij}^c = \tau_i(x_j \mid \phi_i^c) = \alpha_i^c f_i(x_j \mid \phi_i^c) / \sum_{i=1}^{L} \alpha_i^c f_i(x_j \mid \phi_i^c) \tag{3.19}$$

where τ_{ij}^c is the posterior probability that x_j belongs to class i.

M-step:

$$\alpha_i^+ = \sum_{j=1}^{n} \tau_{ij}^c / n \tag{3.20a}$$

$$\begin{aligned} \phi_i^+ \in \arg\max_{\phi_i \in \Omega} (\sum_{k=1}^{m_i} \ln(f_i(y_k \mid \phi_i)) \\ + \sum_{k=1ik} \ln(f_i(x_k \mid \phi_i))) \end{aligned} \tag{3.20b}$$

Equation (3.20b) indicates that the optimal ϕ_i maximizes the weighted summation of the: log likelihood of training samples and unlabeled samples. For every training sample, the: weighting factor is one, and for every unlabeled sample, the weighting factor is the posterior probability. If L classes can be represented as Gaussian distributions, Eq. (3.20a) and (3.20b) yield:

$$\begin{aligned}
\mu_i^+ &= \frac{\sum_{j=1}^{m_i} y_{ij} + \sum_{j=1}^{n} \tau_{ij} x_j}{m_i + \sum_{j=1}^{n} \tau_{ij}} \\
&= \frac{m_i}{m_i + \sum_{j=1}^{n} \tau_{ij}} \frac{\sum_{j=1}^{m_i} y_{ij}}{m_i} + \frac{\sum_{j=1}^{n} \tau_{ij}}{m_i + \sum_{j=1}^{n} \tau_{ij}} \frac{\sum_{j=1}^{n} \tau_{ij} x_j}{\sum_{j=1}^{n} \tau_{ij}}
\end{aligned} \qquad (3.21a)$$

$$\begin{aligned}
\Sigma_i^+ &= \frac{\mathop{}\limits_{j=1}^{m_i} (y_{ij} - \mu_i^+)(y_{ij} - \mu_i^+)^T + \sum_{j=1}^{n} \tau_{ij} (x_j - \mu_i^+)(x_j - \mu_i^+)^T}{m_i + \sum_{j=1}^{n} \tau_{ij}} \\
&= \frac{m_i}{m_i + \sum_{j=1}^{n} \tau_{ij}} \frac{\sum_{j=1}^{m_i} (y_{ij} - \mu_i^+)(y_{ij} - \mu_i^+)^T}{m_i} \\
&+ \frac{\sum_{j=1}^{n} \tau_{ij}}{m_i + \sum_{j=1}^{n} \tau_{ij}} \frac{\sum_{j=1}^{n} \tau_{ij} (x_j - \mu_i^+)(x_j - \mu_i^+)^T}{\sum_{j=1}^{n} \tau_{ij}}
\end{aligned} \qquad (3.21b)$$

In [7][8], the EM algorithm has been studied and applied to remote sensing data. It was shown that by assuming a mixture model and using both training samples and unlabeled samples in obtaining the statistics estimates, the classification performance can be improved, and the Hughes phenomenon can then be delayed to a higher dimensionality and hence more features can be applied to achieve better performance. In addition, the parameter estimates represent the true class distribution more completely.

As indicated by **Eq.** (3.19) through **Eq.** (3.21b), in the EM algorithm each unlabeled sample contributes to the statistics of all classes selected, and the amount of contribution is weighted by the sample's posterior probability. This is reasonable because at this stage the class label information of an unlabeled sample is completely missing. However, if we assume each sample has a unique class label, apparently the influence from one of the unlabeled samples k of the j^{th} class to the i^{th}class statistics ($i \neq j$) is undesired, specifically, if i^{th} and j^{th} are quite different, and it is possible sample k has a large posterior probability for i^{th} class. This negative influence may be significant enough to cause the estimated statistics to deviate from the true ones. As a result, the iteration may converge to erroneous solutions. This situation can become very severe when a large number of unlabeled samples are used. For example, if the class proportion is quite unbalanced, i.e., there are a few classes that are quite dominant in the given data set, then the large number of unlabeled samples used may be mostly from these dominant classes.

An alternative way is to replace unlabeled samples by semi-labeled samples, which contain partial information of class origin obtained by a decision rule in the classification process. With the additional information of class labels, one can limit the effect of a semi-labeled sample to one class to which it has been assigned with the highest likelihood. In addition, by using semi-labeled samples, parameter estimation and classification can be integrated in an iterative way such that they enhance each other consistently. In this process, every bit of improvement from classification is fed back to the ;process of parameter estimation and hence leads to better statistic estimation, and in return a better classification accuracy can be achieved. In other words, a self-learning and self-adapting process can then be established. This is advantageous for the analysis of high(-dimensional data with limited training samples. In high dimensional space, in general, samples are more separable, and higher classification accuracy can be achieved if class statistics can be estimated more precisely. In the following section, an adaptive classifier will be proposed using both training samples and semi-labeled samples to obtain close to optimal statistics estimation and classification iteratively.

The proposed adaptive classifier is an iterative method to numerically find close to optimal performance for given data by integrating parameter. estimation with classification. Denote $y = (y_{i1},...,y_{im_i})$ as the training samples for the i^{th} class, whose pdf is $f_i(x|\phi_i)$, and $x = (x,,,...,x_{in_i})$ are the semi-labeled samples that have been classified to the i^{th} class. Among these semi-labeled samples, there are two types of samples, the correctly classified samples and misclassified samples. Correctly classified samples can play a role as equivalent to training samples and enhance statistics estimation. On the other hand, misclassified samples introduce undesired effects as information noise to the estimated statistics. Ideally, one would like to just use those semi-labeled samples that have been correctly classified. However, information about the classification accuracy for individual sample is not available at this point. Therefore, in order to control the effect from semi-labeled samples, a weighting factor is applied to represent this influence.

With this in mind, an adaptive classifier is designed, which obtains close to optimal performance by maximizing the weighted log likelihood of training samples and semi-labeled samples. Similar to the EM algorithm, it is an iterative approach and achieves the optimal statistics estimation and classification by starting with initial estimate ϕ^0 and classification based on training samples only and repeating the following steps at each iteration using training samples and semi-labeled samples:

1) Compute Weighting Factors:

$$w_{ij}^c = \frac{f_i(x_{ij} \mid \phi_i^c)}{\sum_{k=1}^{L} f_k(x_{ij} \mid \phi_k^c)} \tag{3.22a}$$

2) Maximize the mixed log likelihood:

$$\begin{aligned} \phi_i^+ = \arg\max_{\phi_i \in \Omega}(&\sum_{k=1}^{m_i} \ln(f_i(y_k \mid \phi_i)) \\ &+ \sum_{k=1}^{n_i} w_{ik}^c \ln(f_i(x_{ik} \mid \phi_i))) \end{aligned} \tag{3.22b}$$

3) Perform classification based on the maximum likelihood (ML) classification rule:

$$\mathrm{x} \in \mathrm{i} \Leftrightarrow \mathrm{i} = \arg\max_{1 \le i \le L}(\ln(f_i(x \mid \phi_i^+))) \qquad (3.22c)$$

Here the superscript "c" and "+" designate the current and next value respectively. If all L classes are Gaussian distributed, Eq. (3.22b) yields,:

$$\begin{aligned} \mu_i^+ &= \frac{\sum_{j=1}^{m_i} y_{ij} + \sum_{j=1}^{n_i} w_{ij}^c x_{ij}}{m_i + \sum_{j=1}^{n_i} w_{ij}^c} \\ &= \frac{m_i}{m_i + \sum_{j=1}^{n_i} w_{ij}^c} \frac{\sum_{j=1}^{m_i} y_{ij}}{m_i} + \frac{\sum_{j=1}^{n_i} w_{ij}^c}{m_i + \sum_{j=1}^{n_i} w_{ij}^c} \frac{\sum_{j=1}^{n_i} w_{ij}^c x_{ij}}{\sum_{j=1}^{n_i} w_{ij}^c} \end{aligned} \qquad (3.23a)$$

$$\begin{aligned} \Sigma_i^+ &= \frac{\sum_{j=1}^{m_i} (y_{ij} - \mu_i^+)(y_{ij} - \mu_i^+)^T + \sum_{j=1}^{n_i} w_{ij}^c (x_{ij} - \mu_i^+)(x_{ij} - \mu_i^+)^T}{m_i + \sum_{j=1}^{n_i} w_{ij}^c} \\ &= \frac{m_i}{m_i + \sum_{j=1}^{n_i} w_{ij}^c} \frac{\sum_{j=1}^{m_i} (y_{ij} - \mu_i^+)(y_{ij} - \mu_i^+)^T}{m_i} \\ &+ \frac{\sum_{j=1}^{n_i} w_{ij}^c}{m_i + \sum_{j=1}^{n_i} w_{ij}^c} \frac{\sum_{j=1}^{n_i} w_{ij}^c (x_{ij} - \mu_i^+)(x_{ij} - \mu_i^+)^T}{\sum_{j=1}^{n_i} w_{ij}^c} \end{aligned} \qquad (3.23b)$$

and Eq. (18c) yields:

$$x \in i \Leftrightarrow i = \arg\min_{1 \le i \le L} d_i(x)$$

where d_i is a discriminant function [1] given by:

$$d_i(x) = (x - \mu_i^+)(\Sigma_i^+)^{-1}(x - \mu_i^+)^T + \ln\left|\Sigma_i^+\right|$$

Note that in a manner similar to the EM algorithm, the rnean vectors and covariance matrices are weighted mixtures of ML estimates from training samples and semi-labeled samples, and the weight for each sample is related to the relative likelihood, which is less than one. But in this proposed adaptive classifier, unique membership is assumed and each semi-labeled sample only has contribution to the same class to which is classified. In addition, in this iterative process, the membership of each training sample remains the same. However, the membership of each semi-labeled sample is being updated at every iteration through the whole procedure.

3.3 Experimental Results

In the following experiments, we test the performance of the proposed adaptive classifier using both simulated and real multispectral data. The first two experiments use simulated data of dimensionality of 6, 20, and 40. The third uses 12 dimensional real data.

In experiment 1 and 2, there are three simulated classes with Gaussian distributions. Three sets of labeled samples are generated independently. In the first set, there are 1000 samples for each class, and 10 samples are selected ranclomly from 1000 samples and subsequently used for training; the other 990 samples are then classified and become semi-labeled samples, which are used to estimate statistics at the following iteration. In the second data set, there are 10,000 random samples for each class and they are used for testing the performance of the classifier. The third data set is generated to benchmark the performance of the proposed adaptive classifier. In this data set, there are 1000 random samples for each class, and then all of them are used for designing a classifier, which is then tested by using 10,000 test samples from the second data set. The convergence criterion is that the relative difference of classification accuracy between two consecutive iterations is less than 0.01%. Each experiment is repeated ten times, and the rnean classification accuracy and standard deviation are then estimated.

3.3.1 Experiment 1: equal spherical covariance

1) d=6: In this experiment, the covariance matrix of all three classes is the identity matrix, but each class had a slightly different mean vector. The mean of the first class is at the origin; the mean of the second class is 3.0 in the first variable and zero in the other variables. The dimension is d=6. The mean classification accuracy versus iteration number is graphed in Fig. (3.5a).

Here SC represents the mean classification accuracy and standard deviation of the data where a sample covariance estimate is used as the initial estimate from training samples, and the mixed sample covariance shown in Eq. (3.23b) is used for the later estimation. The SC_Test represents the results for the testing data. LOOC represents the results where a mixed covariance estimator, LOOC, is used to estimate covariance matrices [2], and, similar to SC case, the mixed sample covariance shown in Eq. (3.23b) is then used for the following covariance estimation. LOOC_Test represents the results of the testing data.

The results show that with additional semi-labeled samples, the mean accuracy of data and testing data increases steadily with iterations until it reaches convergence. Note that in this data set, in the supervised learning process the mean classification accuracy for training data (resubstution accuracy [1]) is 91.01% with a standard deviation 0.66%, and for testing (hold out accuracy [1]) it is 90.67% with a standard deviation 0.15%. The Bayes accuracy (optimal) is bounded between these two. Therefore, we believe the final convergence solution is optimal within a range of standard deviation. Also, it is observed that the difference of the mean accuracy between data and test data are within a standard deviation. Further, the standard deviation is reduced with iterations. The final one is reduced by about five folds. Additional results not shown here indicate that the estimated statistics become more and more representative to the true ones and more robust. This, then, is a self-improving adaptive classifier where statistics estimation and classification enhance each other.

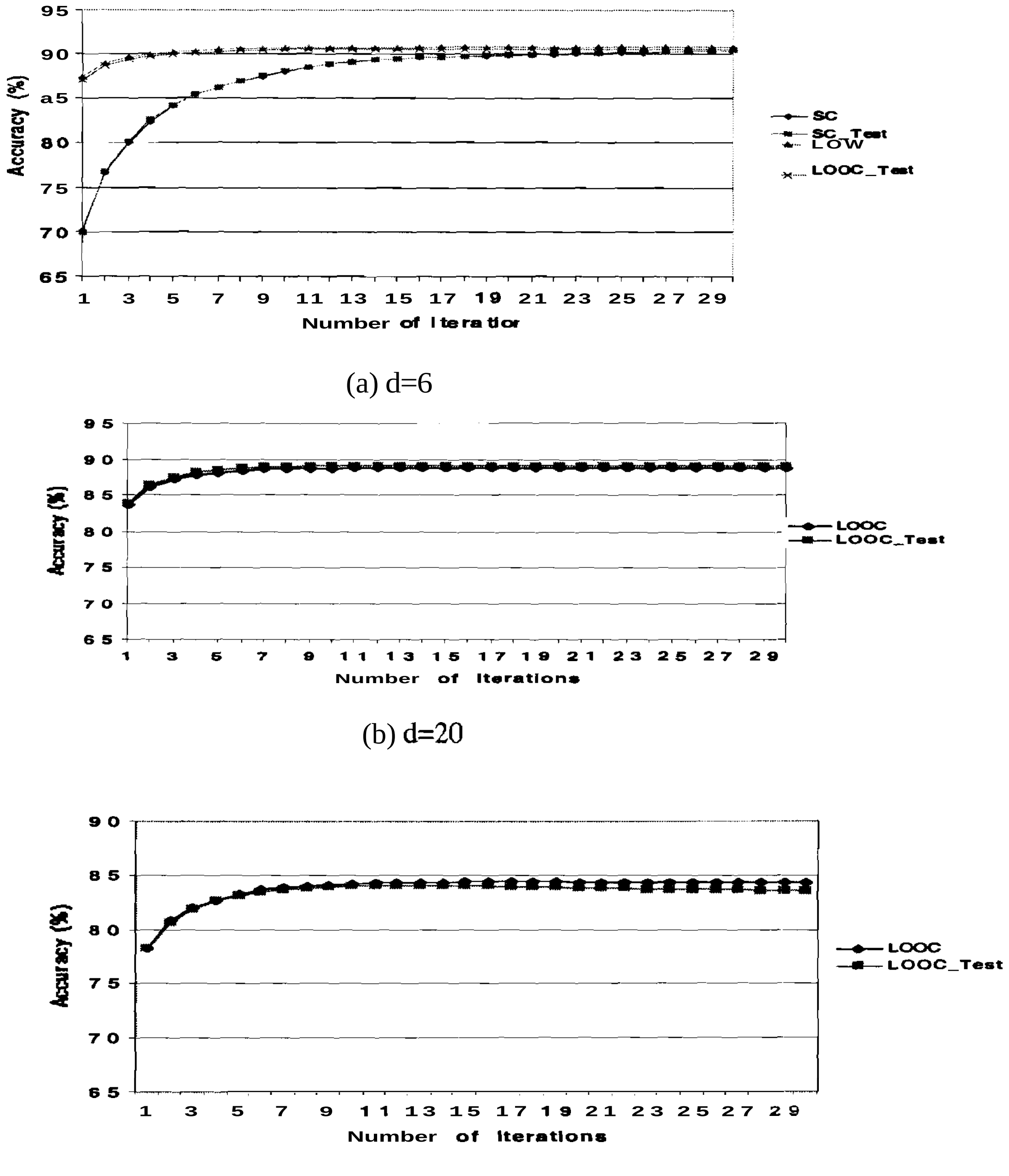
Also, it is seen that without LOOC, the initial accuracy is lower, and as a result convergence is attained more slowly but the final accuracy is very close to that with

LOOC. This further indicates that eventually semi-labeled samples can compensate for the deterioration of classifier performance caused by lack of training samples.

2) d=20: In this experiment, the synthetic data from the experiment 1a is used with the exception that the dimensionality is raised from 6 to 20. Hence, the number of dirnension is now greater than the number of class training samples but smaller than the total number of training samples. This case represents a poorly posed problem where the dimension size is greater than the training sample size. Mean classification accuracy is plotted in Fig. 3.5b. Since the number of dimension is greater than the class training sarnple size, the sample covariance matrix becomes singular and uninvertible. The covariance estimator LOOC must be used for the initial iteration. In this experiment, for supervised learning, the mean accuracy for data is 91.51% (std. dev. 0.59%) and for test data is 90.12 (std. dev. 0.12%).

Comparing with experiment one, even though the initial classification accuracy reduces about 3% relatively, the classification accuracy still steadily increases and final classification accuracy is only about 2% lower. These results indicate that even with the poorly posed problem, this proposed adaptive classifier is still able to perform well.

3) d=40: Again, in this experiment the synthetic data from the experiment 1a is used with the exception that the dimension is increased to 40. Hence, the number of dimensions is much greater than the number of class training samples and even greater than the total number of training samples. This case represents an ill-posed problem where the number of dimensions exceeds the total number of training samples, and the number of parameters (2000) is twice the number of samples available. Mean classification accuracy is plotted in Fig. 3.5c. Again, since the number of dimension is greater than the class training sample size, the sample covariance matrix is singular and uninvertible. The covariance estimator LOOC is again used for the initial iteration. In this experiment, for supervised learning, the mean accuracy for data is 93.46% (std. dev. 0.57%) and for test data is 88.33 (std. dev. 0.28%).



(a) d=6

(b) d=20

(c) d=40

Fig. 3.1. Mean Accuracy for **Experiment** 1

Compared to the results of LOOC in experiment one, even though the initial classification accuracy is reduced about 10% relatively, the classification accuracy for the data still steadily increases. Final classification is about 7% less, and the standard deviation reduces with iterations as well. For testing data, the classification accuracy converges more slowly, and the final value is a little lower than previous accuracy. But overall these results show that this proposed adaptive classifier still is able to perform relatively well even for an ill-posed problem.

3.3.2 Experiment 2: unequal spherical covariance matrices

1) d=6: In this experiment, the three classes have unequal rnean vectors and spherical covariance matrices. The mean vectors are the same as those in the experiment one:. The covariance matrices of class one, two and three are I, 21 and 3I respectively. In this case, these three classes overlap more and are more difficult to discriminate than the equal covariance case. Mean accuracy is plotted in Figure 3.6a. It is observed that these results are similar to those in experiment la. In this experiment, for supervised learning, the mean accuracy for data is 88.68% (std. dev. 0.75%) and for test data is 85.99 (std. dev. 0.20%).

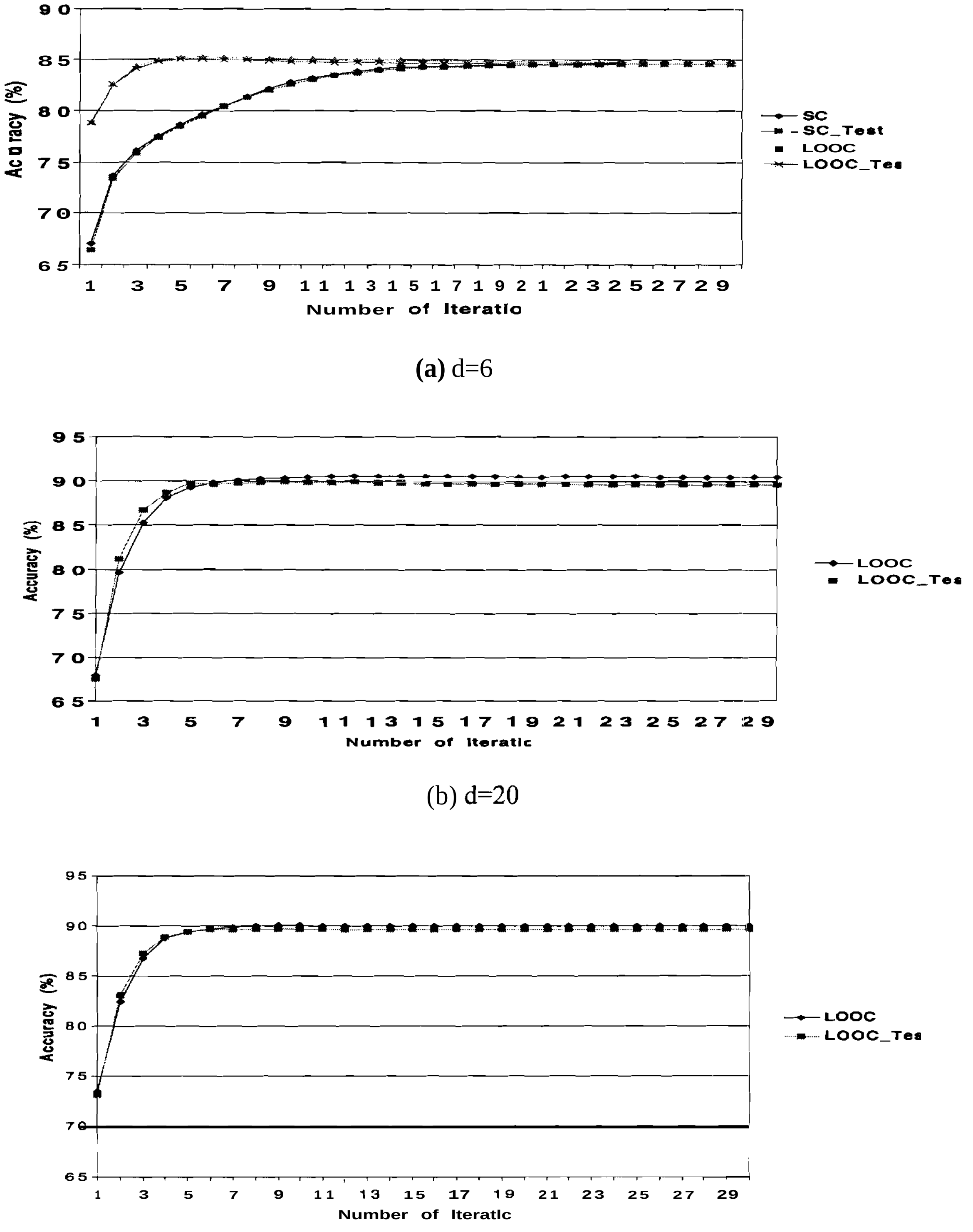
2) d=20: In this experiment, the simulated data in Experiment 2a is used with exception that the dimension is twenty, which is greater than the number of training samples. This is thus again a poorly posed problem. Mean accuracy is plotted in Figure 6b. In this experiment, for supervised learning, the mean accuracy for data is 92.48% (std. 0.56%) and for test data is 90.98 (std. 0.13%).

It is worth noting that even though the initial classification mean accuracy is reduced by 7% relatively, the final increases by 5%. This shows the appealing fact that with semi-labeled samples the proposed adaptive classifier is able to utilize the increment of separability provided by additional dimensions, and then improve the classification accuracy. In other words, Hughes phenomenon is mitigated.

3) d=40: In this experiment, the simulated data in Experiment 2a is used with exception that the dimension is forty. Mean accuracy is plotted in Figure 3.6c. In this

experiment, for supervised. learning, the mean accuracy for data is 96.27% (std. **0.40%)** and for test data is **93.07** (std. 0.14%).

With such a high ratio of the number of dimensions to the number of samples, it is seen due to the Hughes phenomenon, the accuracy with only ten training samples is greatly reduced, about **10%.**However, with additional semi-labeled samples being fed back to statistics estimation, the classification accuracy is able to climb up and quickly coriverges to a value that is just slightly lower than the optimal with diminishing standard deviation.



(a) d=6

(b) d=20

(c) d=40

Fig. 3.2. Mean Accuracy for Experiment 2

3.3.3 Experiment 3: Flight line C1

This experiment is conducted using real samples from data designated Flightline C1 (FLC1), which is 12-band multispectral data taken over Tippecanoe County, Indiana by the M7 scanner [10] in June, 1966. The number of training samples and testing sarnples in each class is listed in Table 1. The training sample size was deliberately chosen to be very small, representing a poorly-posed problem where the number of training samples for each class is comparable to dimensions. Since the resting data in this experiment is very large, and in particular for some of classes with small numbers of samples, almost all samples of such class are included in the testing data. For this reason, the testing samples and a majority of training samples are independent, and there are small overlaps on the testing data and training data. Also, for the same reason, test samples that are not training samples are used as semi-labeled samples and are used to update the class statistics. Otherwise, there may not be sufficient semi-labeled samples to modify the class statistics for some minority classes. The classification results are plotted in Fig. 7, based on available ground truth for the area, a test field map is provided in Fig. 8a, and thematic map for the initial and final classifications are shown in Fig. 8b and 8c. It is seen from Fig. 7, the classification accuracy increases and converge:; quickly, and the final accuracy is slightly lower than 94.7%, the resubstution classification accuracy that is obtained by using all testing samples as training samples. Also, comparing Fig. 8b with Fig. 8c, the speckle error has been greatly reduced.

Table 3.1

Training and testing samples for Flight line C1

Class Name	No. of Testing samples	No. of training samples
Alfalfa	3,375	12
Rr Soil	1,230	8
Corn	10,625	16
Oats	5,781	8
Red Clover	12,147	12
Rye	2,385	4
Soybeans	25,174	16
Water	18	4
Wheat-1	7,827	12
Wheat-2	2,091	16
Total	70,653	104

To illustrate how this proposed classifier improves itself iteratively by reducing the class statistics estimation error, the close up snapshots of the classified map for two crops are presented in Fig. 3.9 and 3.10. Fig. 3.9 is of the rye field a little below the middle of the flightline (Figure 3.8). As shown in Fig. 3.9a, the rye tsaining field of 4 pixels was selected in it. As illustrated in figure 3.9b, due to poorly estimated statistics using limited training samples, the majority of pixels have been misclassified as something other than rye. However, at the second iteration when semi-labeled samples are added to enhance the statistics, there are more pixels around the training field classified as rye. This trend continues and at the last iteration, a majority of pixels in the field are eventually correctly classified as rye. In fact, some of the pixels in this rye field are not actually rye.

The second close up example involves the field of oats within a doughnut shaped wheat field just above the middle of the flightline. There are no training fields for oats in this field, and instead oats training is located elsewhere in the flightline:. As expected, at the first iteration, on the test field for oats only very few pixels are correctly classified as oats. However, at the second iteration, more pixels around those pixels that have been previously classified as oats have been identified as oats. As this process continues, more and more pixels on this test field for oats have been correctly identified as oats. In figure 3.10f, at the fifth iteration a group of pixels of the shape of a strip across the oats field has been misclassified as wheat, this is not an error of omission for the class oats. Instead, this area is really a sod water way unplowed by the farmer. Since there are no training samples for this class of ground cover, this result further indicates that the proposed adaptive classifier adjusted itself to the next nearest class based on the information provided by the semi-labeled samples.

To show how representative the estimated parameters are, the probability map [11] associated with the classification is obtained. The probability map is determined by color coding the Mahalanobis distance of each pixel for the class to which it is classified. Blue pixels are ones that classified with low conditional probabilities. The color/likelihood scale indicates increasing likelihood from blue to yellovv to red with red pixels being the ones that are classified with the highest likelihood. Figure 3.11 shows the probability map for the rye field of Figure 3.9. It is seen from this figure that when only the initial supervised learning is used the only bright spots are near the training fields. In other words, the rest of the data are not represented well. By adding semi-.labeled samples to the estimation process, more representative estimates are obtained, and thus the probability maps indicate increased likelihood by the brighter, red color..

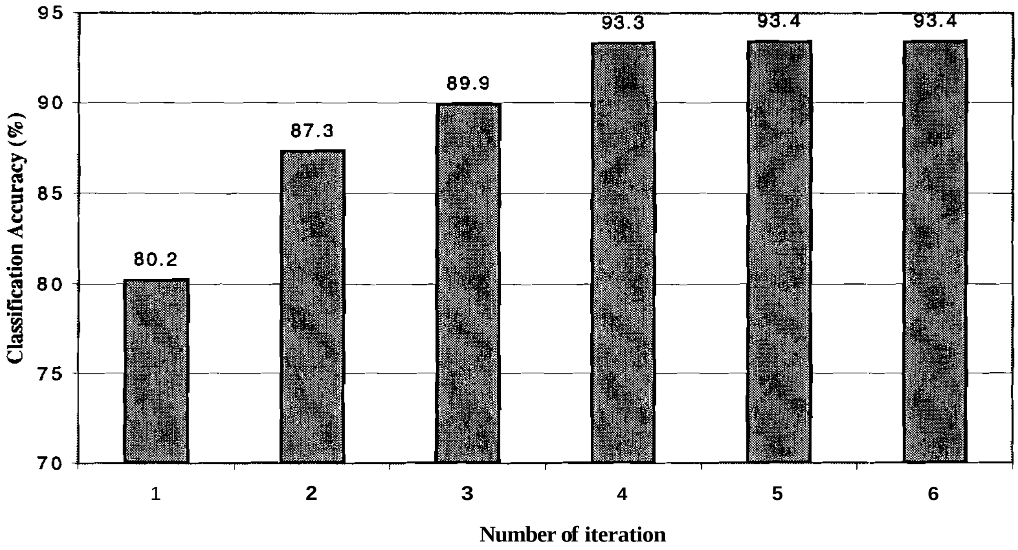
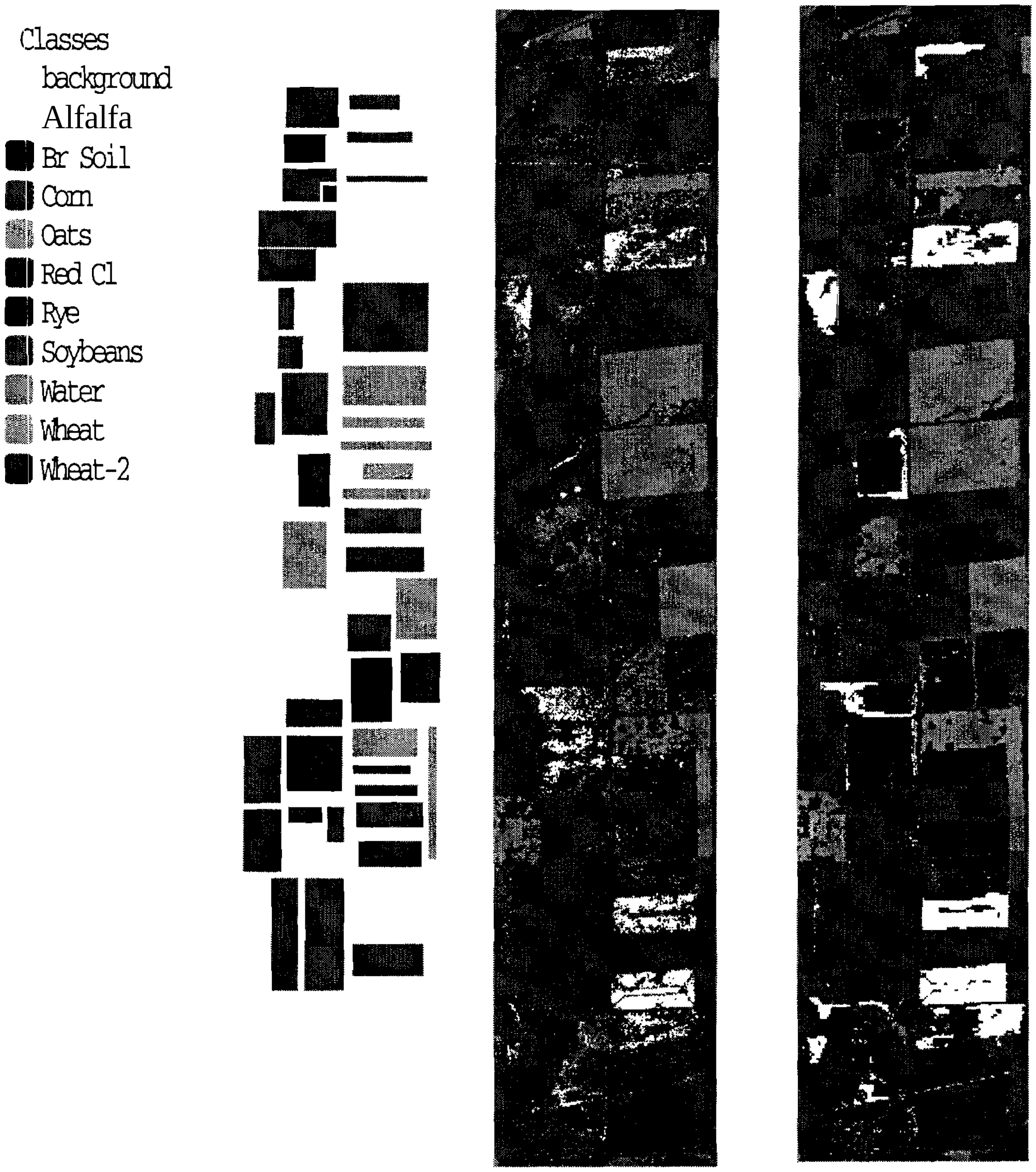
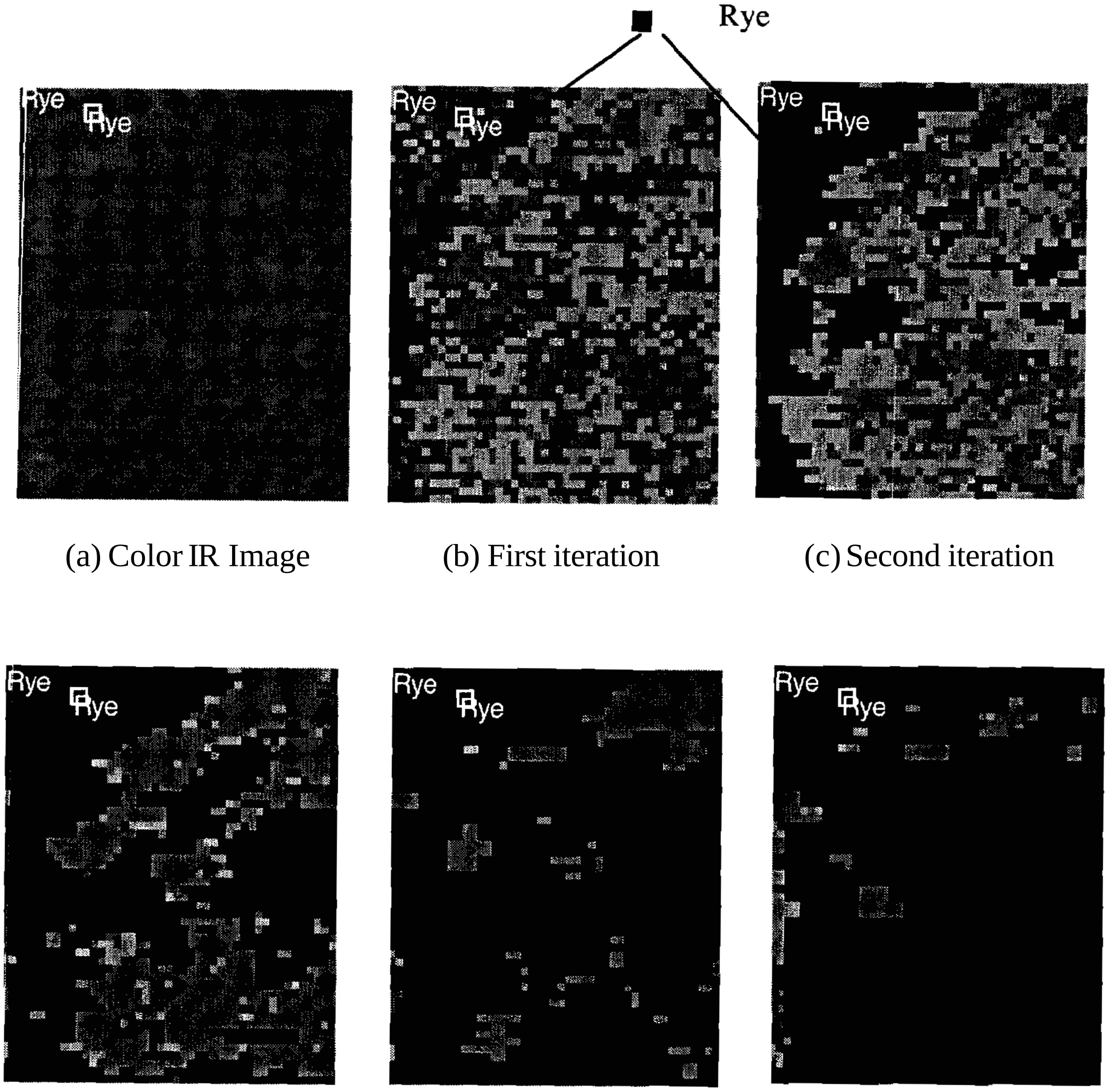


Fig. 3.3. Classification accuracy for flight line C1



(a) Testing field (b) Initial classification (c) Finial Classification

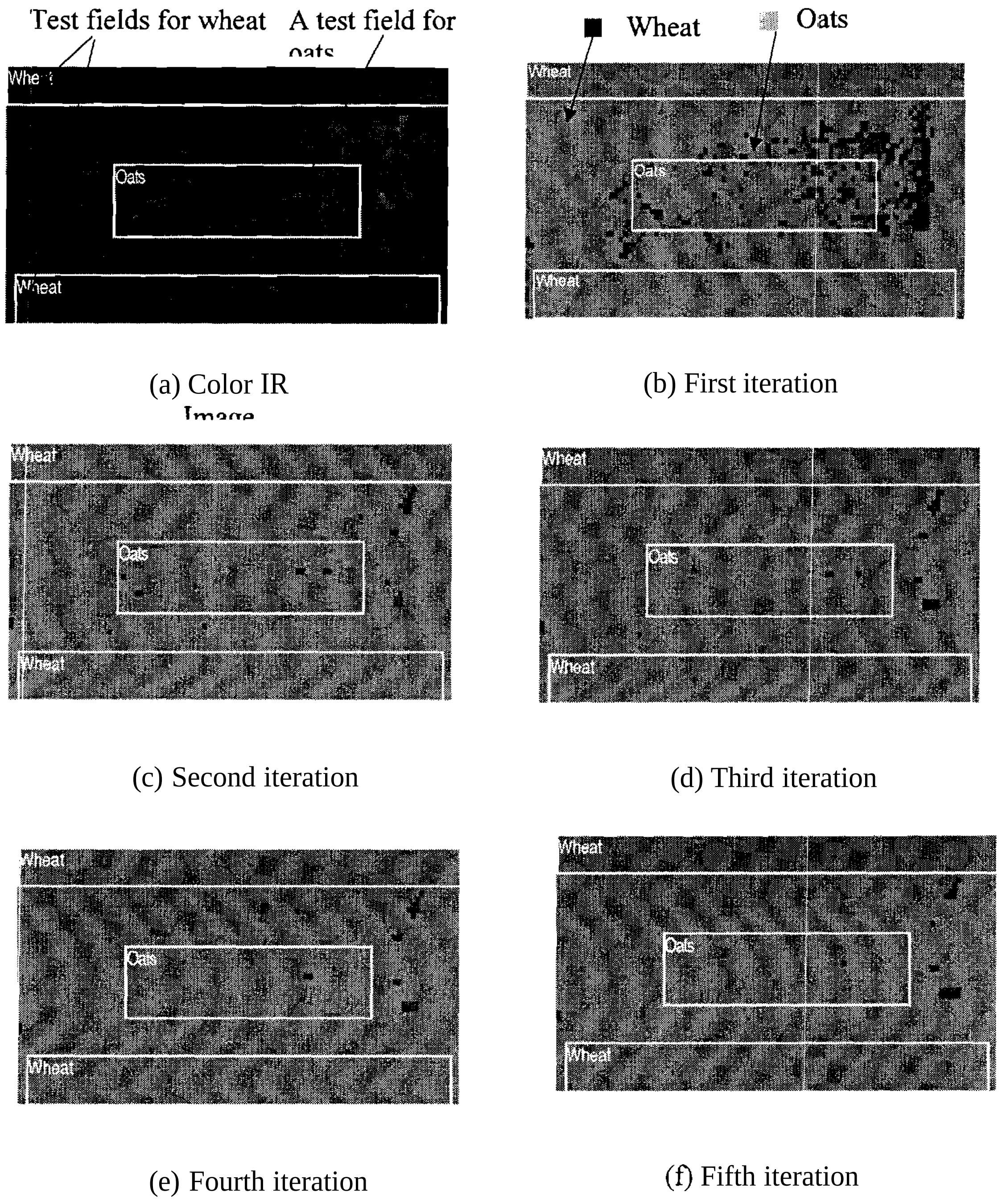
Fig. 3.4. Test and classification map for flight line C1. (In color)



(a) Color IR Image (b) First iteration (c) Second iteration

(d) Third iteration (e) Fourth iteration (f) Fifth iteration

Fig. 3.5. Original image and classification map for a rye test field at each iteration. (In color)



(a) Color IR Image

(b) First iteration

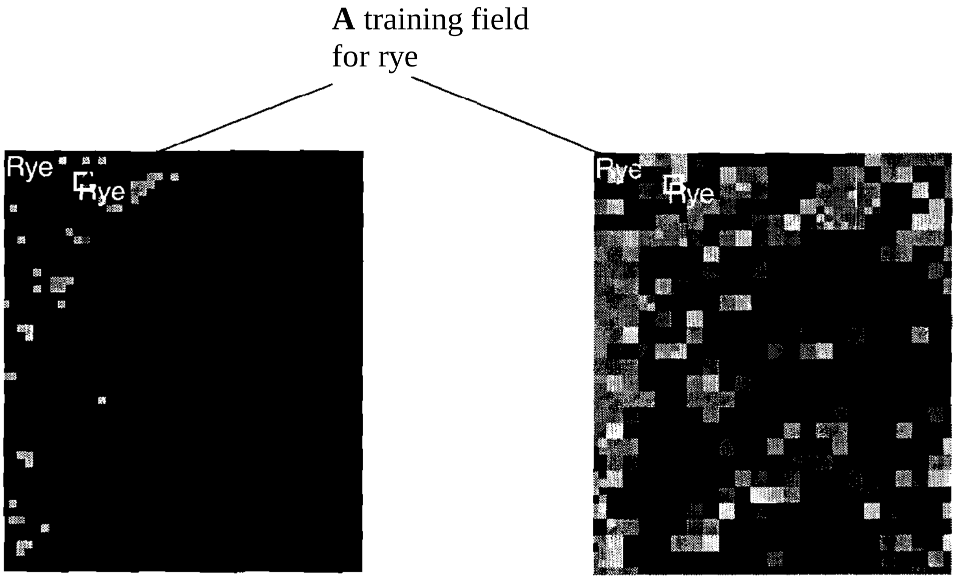
(c) Second iteration

(d) Third iteration

(e) Fourth iteration

(f) Fifth iteration

Fig. 3.6. Original image and classification map of wheat and oats fields at each iteration. (In color)



(a) Initial iteration

(b) Final iteration

Fig. 3.7. Portion of Probability map for Flight Line C1. (In color)

3.4 Conclusion

This thesis is begun by investigating the information contained in semi-labeled samples of two Gaussian distributions in terms of the Fisher Information Matrix. Results show that higher classification accuracy can provide more useful class label information for statistical estimation, and so do the number of samples. The probability of error for semi-supervised learning and combined learning process is also investigated. Results indicate that when semi-labeled samples are fed back to the statistical estimation process, higher accuracy and more semi-labeled samples may enhance statistics significantly and consequently reduce the probability of error for the following classification.

Based on the above findings, a self-improving adaptive process is proposed which integrates statistical estimation and classification using semi-labeled samples. It may mitigate the Hughes phenomenon by iteratively utilizing the additional class label information extracted from classification process.

The experimental results further reveal several benefits of this classifier. First, all experiments show that the proposed adaptive classifier is able to raise classification accuracy steadily and eventually drive it close to the optimal value. Higher initial classification accuracy accelerates the rate of convergence but has little effect on the final classification.

Second, as is shown in experiment results 6a and 6b, when the separability increases with dimensionality, with semi-labeled samples, the peak performance is enhanced. In other words, the information in the new feature measurements can be used to further reduce the error. Without the semi-labeled samples, the peak performance occurs at a lower dimension after which no further improvement can be obtained from new feature measurements; instead performance deteriorates with dimensions.

Third, the estimated statistics are approaching the true ones with iterations. As is shown through all the experiments, the standard deviation is greatly reduced with iterations, indicating the estimated statistics are more and more robust. In particular, as shown in the last experiment with semi-labeled samples, most of samples are classified with high likelihood.

Despite the promising results, the proposed adaptive classifier has limitations. In particular, for a very ill-posed problem, where the number of dimensions are far greater than the number of training samples and the number of parameters are even greater than the number of all semi-labeled samples, the initial classification can be very bad. As a result a positive feedback could hardly be established and the proposecl adaptive classifier may not converge. This necessitates the use of an adaptive covariance estimator, where semi-labeled samples are incorporated into the process to determine the optimal covariance mixture.

CHAPTER 4: AN ADAPTIVE METHOD FOR COMBINED COVARIANCE ESTIMATION AND CLASSIFICATION

4.1 Introduction

In quadratic maximum likelihood classification, the mean vector and covariance matrix are usually unknown and must be estimated by the sample mean and sample covariance matrix based on training samples. When the training sample size is quite small relative to the dimensionality, the sample estimates, especially the sample covariance matrix becomes highly variable and consequently, this greatly deteriorates the classifier performance. In particular, when the number of training samples; is less than the dimensionality, the sample covariance matrix becomes singular and hence quadratic classifiers cannot be used. This poses limitations on the number of dimensions (or features) that can be used in remote sensing applications where training samples are usually small compared to the number of dimensions available. This is unfortunate because larger numbers of features provided by new generation sensors make it possible to identify more classes while training samples remain difficult and expensive to acquire. **An** adaptive classifier has been proposed to mitigate the small training sample problem by using semi-labeled samples in chapter **3.** This method works well for the case that the number of dimensions is not too large. However, when the number of dimensions is very high (up to a few hundreds), the number of parameters in the covariance matrix estimation process increases dramatically (approximate to the square of the dimensions). In such cases, using additional semi-labeled samples alone may not be adequate to reduce the variance of covariance estimation. On the other hand, regularization methods attempt to reduce the variance of these estimates by biasing them toward values that are deemed more "physically plausible" [12]. Therefore, the variance is reduced by limiting the number of parameters needed to be estimated and at the expense of potentially increased bias. The extent of this bias-variance trade-off is controlled by one or more regularization parameters.

In this chapter, a method of combining the adaptive classifier and regularized covariance estimations is proposed. Depending on the method of selecting support covariance matrices and the regularization parameters, a group of new adaptive covariance estimators are then introduced. The regularized parameters and support covariance matrices used in a covariance mixture are determined based on both training samples and semi-labeled samples, and they are repeatedly updated until the highest classification accuracy is reached. Extensive experiments are performed using simulated data and real, aircraft-acquired hyperspectral data. With simulated data, the experimental results indicate the proposed sequential covariance estimators can achieve equivalent classification performance with a small training sample size to that obtained using large training sample size. With hyperspectral data, the proposed adaptive covariance estimators can improve the classification performance significantly with limited training samples.

4.2 Gaussian Maximum Likelihood Classification

The objective of classification is to assign unlabeled samples to one of several groups or classes based on certain decision rules. In the typical quadratic maximum likelihood (QML) classifier, the decision rule can be represented by a discriminate function and an unlabeled sample is partitioned to the class with the: smallest value. A multivariate Gaussian distribution has a pdf as:

$$f_i(x \mid \mu_i, \Sigma_i) = (2\pi)^{-\frac{p}{2}} \mid \Sigma_i \mid^{-1/2} \exp\left[-\frac{1}{2}(x-\mu_i)^T(\Sigma_i)^{-1}(x-\mu_i)\right] 1 \le i \le L$$

where μ_i and Σ_i are ith class mean vector and covariance matrix, respectively, L is the number of classes and $x \in \mathrm{R}^{\mathrm{P}}$. Assuming a [0,1] loss function and equally likely classes, the maximum likelihood classification rule then is given by:

$$x \in i \Leftrightarrow i = \arg\min_{1 \le i \le L} d_i(x) \qquad (4.1)$$

where d_i is a discriminate function with a form as: $d_i(x) = (\mathrm{x} - \mu_i)\Sigma_i^{-1}(x - \mu_i)^T + \ln|\Sigma_i|$

The measure $d_i(x)$ is sometime referred as the (true) general dlistance between x and μ_i with a unit prior probability. The first term is the familiar Mahalanobis distance between x and the mean vector for the i^th^ class, while the latter term is adjustment factors based on the generalized variance.

In practical situations, the true class distributions are not known and hence mean vectors and covariance matrices must be estimated from training samples. The mean is typically estimated by the sample mean, which is also a maximum likelihood estimate, $\hat{\mu}_i^{ML} = m_i = \frac{1}{M_i}\sum_{j=1}^{M_i} y_{i,j}$,where $y_{i,}$, is the jth training sample from class i, and M_i is the total number of training samples in class i, The covariance is typically estimated by the sample covariance $\hat{\Sigma}_i = S_i = \frac{1}{M_i - 1}\sum_{j=1}^{M_i}(y_{ij} - m_i)(y_{ij} - m_i)^T$or by the maximum likelihood covariance estimate $\hat{\Sigma}_i^{ML} = \frac{1}{M_i}\sum_{j=1}^{M_i}(y_{ij} - m_i)(y_{ij} - m_i)^{\mathrm{T}}$.

When sample mean vectors and sample covariance matrices art substituted in Eq. (1), the quadratic discriminate rule (QD) is still asymptotically (large sample) optimal. However, when the size of training sets is small compared to the number of dimensions, the performance of QD can be seriously degraded because sample estimates are unstable. In particular, the sample covariance estimate S_i becomes highly variable, and it is singular if fewer than p+1 training samples from class i are available. Therefore, QD cannot be used in this case.

When the dimensionality is large compared to the number of training samples, the estimated covariance can be highly variable and classifier performance can deteriorate severely. Specifically, when the number of dimensions is greater than the number of training samples, the sample covariance is singular and hence uninvertible. This type of problems is referred as a poorly-posed problem. In particular, when the number of dimensions is even greater than the number of entire training samples, the problem becomes ill-posed.

To deal with the poorly or ill posed problems or those nearly so, the number of the parameters to be estimated must be reduced. One way to deal with this is to employ a linear classifier that is obtained by replacing sample covariance matrices for all classes by their weighted average:

$$S_w = \frac{1}{N-L}\sum_{i=1}^{L}(N_i - 1)S_i \tag{4.2}$$

where N is the total number of training samples from all classes. Once S_w is used, the number of parameters is substantially reduced, and the variance of the. elements of S_w is smaller than the variance of the corresponding elements of S_i. Even if each Σ_i differs greatly, using S_w can sometime lead to better performance for small training sets because $S_{w'}$ reduces the number of parameters to be estimated and decreases the variance. This has been verified by several studies [15][16][17].

Even though a linear classifier often performs better than a quadratic classifier for small training set size, the choice between these two is quite restrictive. Several more flexible methods are proposed in which a sample covariance estimate is replaced by partially pooled covariance matrices of various form, and a varying degree of regularization is applied to control the number of parameters to be estimated and consequently improve the classifier performance based on training samples.

4.3 Regularized Covariance Estimation

Regularization methods attempt to reduce the variance of these estimates by reducing the number of parameters. Usually, there are two tasks in the regularization procedures: 1) select the covariance mixture models, and 2) select a model to determine the appropriate value for regularized parameters.

Normally, in a regularized scheme a covariance mixture of the following form is assumed:

$$\hat{\Sigma}_i = (1 - w_i)S_i + w_i S_p \qquad 0 \le w_i \le 1 \tag{4.3}$$

The regularized (mixing) parameter w_i controls the biasing of individual class covariance sample S_i toward a pooled covariance matrix S_p. However, when the total number of training samples N is comparable to or is less than the: dimension p, S_p becomes problematic. Hence, more regularization is required and usually a non-singular diagonal matrix A is used to replace S_p and a covariance mixture becomes:

$$\hat{\Sigma}_i = (1 - w_i)S_i + w_i\Lambda \qquad 0 \le w_i \le 1 \tag{4.4}$$

Usually, in the model selection the mixing parameter(s) is determined by minimizing a loss function based on the training samples. A popular minimization criterion is based on cross-validated estimation of classification error. In the leave-one-out cross-validation error procedure, the classification rule is to use the classifier designed using $N_i - 1$ training samples excluding the training samples x_i, to classify $x_{i,k}$, and then a parameter is selected that minimizes the classification error rate. This criterion has an advantage of being directly related to classification accuracy. However, the process of estimating the covariance of each class requires the covariance estimates of all classes, which implies the same mixing parameter has to be used for all classes. Apparently, the same choice of mixing parameter might not be optimal for all classes. Furthermore, the parameter values are not unique to achieve the same classification error rate, and therefore a tie-breaking method is required.

An alternative maximization criterion is to maximize the sum of average leave-one-out likelihood values of each class. In this procedure, the leave-.one-out likelihood corresponding to training sample $x_{i,k}$ is obtained by using $N_i - 1$ training samples excluding $x_{i,k}$. This criterion requires less computation than the leave-one-out classification error procedure. Also, it allows different mixing parameters for each class, which provides the flexibility of applying a varying degree of regularization to each class. However, the major shortcoming of this criterion is lack of a direct relationship with classification accuracy.

4.3.1 Regularized discriminant analysis (RDA)

In [12], a procedure referred as " regularized discriminate analysis" (RDA) is proposed, which is a two-dimensional optimization over four-way mixtures as shown in the following:

$$\hat{\Sigma}_i(\lambda,\gamma) = (1-\gamma)\hat{\Sigma}_i(\lambda) + \gamma\left(\frac{tr\left(\hat{\Sigma}_i(\lambda)\right)}{p}\right)I \qquad 0 \le \gamma \le 1 \tag{4.5}$$

where

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

and the common covariance S_w is given by Eq. (4.2). As indicated in Eq. (4.5), the mixing parameter λ controls the amount of shrinkage of the sample covariance S_i toward a common pooled covariance S_i, and the mixing parameter γ regularizes the shrinkage of eigenvalues of S_i toward equality as $tr(\hat{\Sigma}_i(\lambda))/p$ is equal to the average of the eigenvalues $\hat{\Sigma}_i(\lambda)$. Shrinking the eigenvalues of S_i toward equality compensates for the well-known upward bias of the large eigenvalues and downward bias of the smaller eigenvalues of the sample covariance matrix S_i. This is particularly advantageous when the true covariance matrices are some multiples of the identity matrix.

As mentioned before, the pair of mixing parameters is selected by cross-validating on the total number of misclassifications based on available training samples. Even though this procedure has the benefit of directly relating the classification accuracy, it is computationally expensive, and the same mixing parameters must be used for all classes. Moreover, the same classification accuracy can occur on the extensive range of values of the pair (A,γ)[16]. Hence a tie-breaking technique is required.

4.3.2 Leave-one-out covariance (LOOC) Estimator

In [17], a new covariance estimator is proposed which examines the following pair-wise mixtures of the estimators: diagonal sample covariance-sample covariance, sample covariance-common covariance, and common covariance-diagonal common covariance. Thus, it has the following form:

$$\hat{\Sigma}_i(\alpha_i)=\begin{cases}(1-\alpha_i)diag(S_i)+\alpha_i S_i & 0\le a,\le 1\\(2-\alpha_i)S_i+(\alpha_i-1)S & 1\le\alpha_i\le 2\\(3-\alpha_i)S+(a,-2)diag(S) & 2\le a, I\ 3\end{cases}\tag{4.6}$$

where

$$S=\frac{1}{L}\sum_{i=1}^{L}S_i\tag{4.7}$$

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

$$LOOL_i=\frac{1}{N}\sum_{k=1}^{N_i}\ln[f(x_{i,k}|m_{i/k},\hat{\Sigma}_{i/k}(\alpha_i)]\tag{4.8}$$

As aforementioned, in the process of selecting mixing parameter by maximizing leave-one-out average log likelihood, the covariance estimate can be determined independently and then each class can have a mixing parameter that is optimal in terms of available training samples. Overall, classes with more training samples only need a small arnount of bias, while classes with very few training samples need moire bias. In addition, using an approximation on the diagonal matrices, LOOC requires less computation than RDA. However, without this approximation, LOOC needs more computation than RDA. Another major drawback of this criterion is having no direct relationship to classification accuracy.

4.3.3 Empirical Bayesian Covariance Estimate

In [16], a middle-of-road approach between LD and QD is proposed, in which the covariance mixture has the form:

$$\hat{\Sigma}_i(\hat{w}_i)=(1-\hat{w}_i(m))S_i+\hat{w}_iS_p^*(\mathrm{m})\qquad 0\le\hat{w}_i\le 1\tag{4.9}$$

where S_p^* is an appropriate pooled covariance matrix similar to S_p. Here the sample covariance matrices S_i are modeled as outcomes of a common inverted Wishart prior distribution. The parameter m is determined by maximizing the sum of average leave-one-out class likelihood, which has the merit of less computation than RDA and avoiding tie breaking. Also, since the pooled covariance matrices S_p^* and the mixing parameter $\hat{w}_i$ are selected in an empirical Bayes setting, they accommodate the variability of S_i and the training sample size of each class. In particular, under Bayes context, a large magnitude in the variability of the S_i and/or large training sample size N_i leads to small $\hat{w}_i$, while similar S_i and/or small N_i result from large $\hat{w}_i$. However, this approach requires the training sample size be larger than the number of dimensions, which cannot apply to ill- or poorly-posed problems.

4.3.4 Bayesian Leave-One-Out Covariance Estimation (BLOOC)

In [18], a new covariance estimator is developed which virtually is the combination of RDA, LOOC, and empirical Bayesian approach. There are two forms of this new covariance estimation depending on the form of covariance matrices used. When the ridge estimator is adopted, the proposed estimator is called as (bLOOC1) and has the following form:

$$\hat{\Sigma}_i(\alpha_i) = \begin{cases} (1-\alpha_i)\dfrac{tr(S_i)}{p}I + \alpha_i S_i & 0 \le \alpha_i \le 1 \\ (2-\alpha_i)S_i + (\alpha_i - 1)S_p^*(\mathrm{t}) & 1 \le \alpha_i < 2 \\ (3-\alpha_i)S + (\alpha_i - 2)\dfrac{tr(S)}{p}I & 2 < \alpha_i \le 3 \end{cases} \tag{4.10}$$

where t can be expressed as the function of α_i, $t = \dfrac{(\alpha_i - 1)f_i - \alpha_i(p+1)}{2-\alpha_i}$, where p is the dimensionality and $f_i = \mathrm{N}_i - 1$, which represents the degree of freedom in Wishart distributions, and the pooled covariance matrices S_p^* are determined. under a Bayesian context and can be represented as:

$$S_p^*(t) = \left[\sum_{i=1}^{L} \frac{f_i}{f_i + t - p - 1}\right]^{-1} \sum_{i=1}^{L} \frac{f_i S_i}{f_i + t - p - 1} \tag{4.11}$$

When the mixture of covariance and covariance-diagonal covariance matrices is used, the proposed estimator is referred as (bLOOC2) and is defined as the following

$$\hat{\Sigma}_i(\alpha_i)=\begin{cases}(1-\alpha_i)diag(S_i)+\alpha_i S_i & 0\le\alpha_i\le 1\\ (2-\alpha_i)S_i+(\mathrm{a},-1)S_p^*(\mathrm{t}) & 1\le\alpha_i<2\\ (3-\alpha_i)S+(\alpha_i-2)diag(S) & 2\,\mathbf{I}\,\mathrm{a},\le 3\end{cases}\qquad(4.12)$$

The mixing parameters α_i are determined by maximizing average leave-one-out log likelihood.

As an extension of RDA, LOOC, the Empirical Bayesian covariance estimators, bLOOC1 and bLOOC2 have appealing benefits possessed by these methods. For example, like LOOC, bLOOCl and bLOOC2 are quite flexible on the training sample size. They can deal with a broad range of limited training sample sizes, from well-posed, to poorly posed, and ill-posed problems, and the mixing parameters are customized for each class. Also, with the approximation on the diagonal matrices, bLOOCl and bLOOC2 are computational more efficient than RDA and the Empirical Bayesian covariance estimators. In addition, like the Empirical Bayesian covariance estimators, bLOOC1 and bLOOC2 can accommodate the variability of S_i. Using ridged estimators, bLOOC1 has the additional advantage of reducing the larger eigenvalues and increasing the smaller ones.

However, bLOOCl and bLOOC2 suffer from drawbacks inherited in RDA, LOOC and the Empirical Bayesian covariance estimators. First of all, even though the average leave-one-out likelihood used in LOOC, bLOOCl and bLOOC2 provides the flexibility of selecting different mixing parameters for each class and avoiding tie-breaking, it has a major disadvantage of having no direct relation with classification accuracy. Most important of all, even though instability of covariance estimates posed by limited training samples can be reduced using a covariance mixture in the aforementioned approaches, the degree of improvement is certainly limited. This is true because covariance matrix estimates and mixing parameters used in the covariance mixture are all based on limited training samples only. In particular, when the training sample sets are so small, the estimated covariance matrices can be over-tuned to accommodate training samples only and they may not be good representatives of statistics for the entire data.

On the other hand, in the chapter 3 we proposed an adaptive iterative classifier, where the limited training samples problem is alleviated by using additional semi-labeled samples to enhance statistics estimation. In this proposed classifier, the class label of a semi-labeled sample is updated in the classification process at each iteration. We have shown that in an adaptive classifier, starting with a reasonable good initial accuracy achieved by using training samples only, a positive feedback process can be established where semi-labeled samples can provide additional useful class label information and, when they are used, the estimation of statistics can be enhanced and the classification accuracy can be improved. In return, the class label information from semi-labeled samples can be further enhanced in the later stage when better statistics estimation and higher classification accuracy are achieved. However, when the number of dimensions is very high (up to a few hundreds), the number of parameters in the covariance matrix estimation process increases dramatically (approximate to the square of the dimensions). In such cases, using additional semi-labeled samples alone may not be sufficient to reduce the variance of covariance estimation.

4.4 Adaptive Covariance Estimators

A new method is then developed in this section that combines an adaptive classifier with various regularized covariance estimation methods, i.e., LOOC, bLOOC1 and bLOOC2. As an adaptive classifier, this method is an iterative approach, i.e., initially the regularized covariance matrices are determined by using training, samples only, and then they are continuously updated using training samples in addition to currently updated semi-labeled samples until a convergence is reached where the classification accuracy changes very little.

Denote $y = (y_{i1},...,y_{im_i})$ as the training samples for the i^{th} class, whose pdf is $f_i(y\,|\,\Phi_i)$, and $\mathbf{x} = (x_{i1}^c,...,x_{in_i}^c)$ as the current semi-labeled samples that have been classified to belong to the i^{th} class. Depending on the covariance estimator with which the adaptive classifier is combined, the proposed estimators have various forms:

1) ADAPTIVE LOOC

(1) Compute the Weighting Factors

$$w_{ij}^{c} = \frac{f_i(x_{ij} \mid \phi_i^c)}{\sum_{k=1}^{L} f_k(x_{ij} \mid \phi_k^c)}$$

$$= \frac{1}{1 + \sum_{k=1,k\neq j}^{L} f_i(x_{ij} \mid \phi_k^c) / f_k(x_{ij} \mid \phi_i^c)}$$

$$= \frac{1}{1 + \sum_{k=1,k\neq j}^{L} \exp(-\frac{1}{2}(d_k(x_{ij}) - d_i(x_{ij})))}$$

where $d_i(x)$ is the general distance of x to the ith class and is defined as:

$$d_i(x) = (x - \mu_i^c)(C_i^c)^{-1}(x - \mu_i^c)^T + \ln\left|C_i^c\right|$$

(2) Estimate the mean vectors and supportive covariance matrix, i.e. sample covariance and common covariance matrices:

$$\mu_i^+ = \frac{\sum_{j=1}^{m_i} y_{ij} + \sum_{j=1}^{n_i} w_{ij}^c x_{ij}^c}{m_i + \sum_{j=1}^{n_i} w_{ij}^c}$$

$$\Sigma_i^+ = S_i^+ = \frac{\sum_{j=1}^{m_i} (y_{ij} - \mu_i^+)(y_{ij} - \mu_i^+)^T + \sum_{j=1}^{n_i} w_{ij}^c (x_{ij}^c - \mu_i^+)(x_{ij}^c - \mu_i^+)^T}{m_i + \sum_{j=1}^{n_i} w_{ij}^c}$$

$$S^+ = \frac{1}{L}\sum_{i=1}^{L} S_i^+$$

(3) Estimate the regularized covariance mixture:

$$C_i^+(\alpha_i) = \begin{cases} (1-\alpha_i^+)diag(S_i^+) + \alpha_i^+ S_i^+ & 0 \le \alpha_i^+ \le 1 \\ (2-\alpha_i^+)S_i^+ + (\alpha_i^+ - 1)S^+ & 1 \le \alpha_i^+ \le 2 \\ (3-\alpha_i^+)S^+ + (\alpha_i^+ - 2)diag(S^+) & 2 \le \alpha_i^+ \le 3 \end{cases} \tag{4.13}$$

The way to select the optimal mixing parameter α_i^+will be given later.

(4) Perform classification based on the ML rule:

$$x \in \mathrm{i} \Leftrightarrow \mathrm{i} = \underset{1 \le i \le L}{\arg\max}(\ln(f_i(x \mid \mu_i^+, C_i^+(\alpha_i^+))))$$

Steps (1) through step (4) are repeated until a convergence is attained.

Except at step (3), Adaptive BLOOC1 and BLOOC2 have the same procedures as in the Adaptive LOOC. In step (3), adaptive BLOOCl and BLOOC2 have steps with the following forms, respectively:

2) Adaptive BLOOCl

$$C_i(\alpha_i) = \begin{cases} (1-\alpha_i)\dfrac{tr(S_i)}{p}I + \alpha_i S_i & 0 \le \alpha_i \le 1 \\ (2-\alpha_i)S_i + (\alpha_i - 1)S_p^*(\mathrm{t}) & 1 \le \alpha_i < 2 \\ (3-\alpha_i)S + (\alpha_i - 2)\dfrac{tr(S)}{P}\mathbf{I} & 2 < \alpha_i \le 3 \end{cases} \tag{4.14a}$$

3) Adaptive BOOC2

$$C_i(\alpha_i^+) = \begin{cases} (1-\alpha_i^+)diag(S_i^+) + \alpha_i^+ S_i^+ & 0 \le \alpha_i^+ \ 51 \\ (2-\alpha_i^+)S_i^+ + (\alpha_i^+ - 1)S_p^{*+}(\mathrm{t}) & 1 \le \alpha_i^+ < 2 \\ (3-\alpha_i^+)S^+ + (\alpha_i^+ - 2)diag(S^+) & 2 \ \mathbf{I} \ \alpha_i^+ \le 3 \end{cases} \tag{4.14b}$$

Correspondingly, similar to the results from [18], the pooled covariance matrix $S_p^*(t)$ is given by:

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

where t is related to α_i by the following expression:

$$t = \frac{(\alpha_i - 1)((f_i + \sum_{k=1}^{n_i} w_{ik}) - \alpha_i (p+1))}{2 - \alpha_i}$$

4) Selecting an appropriate regularized parameter

For the proposed estimators, since the semi-labeled samples are used in addition to training samples, the leave-one-out average likelihood is modified and used as the criterion to select the appropriate mixture model. In other words, the mixing parameters α_i are selected so that the best fit to the training samples and semi-labeled samples is achieved, which implies the best classification accuracy may then be able to be accomplished correspondingly. The technique is to remove a sample.,estimate the mean and covariance from the remaining samples, and then to compute the mixed log likelihood of the sample that is left out, given the mean and covariance estimates. Each sample is removed in turn, and the average mixed log likelihood is computed. By changing the value of α_i, the value of α_i that maximizes the average mixed log likelihood is selected.

Denote $m_{i/k}$ and $S_{i/k}$ as the mean and the sample covariance of class iwithout sample k, respectively. Depending on whether k is a training sample or semi-labeled sample, $m_{i/k}$ and $S_{i/k}$can be computed as follows:

If k is a training sample

$$\mu_{i/k} = \frac{\sum_{j=1}^{m_i} y_{ij} - y_{ik} + \sum_{j=1}^{n_i} w_{ij}^c x_{ij}^c}{(m_i - 1) + \sum_{j=1}^{n_i} w_{ij}^c}$$

$$\Sigma_{i/k} = \frac{\sum_{\substack{j=1 \\ j \neq k}}^{m_i} (y_{ij} - \mu_{i/k})(y_{ij} - \mu_{i/k})^T + \sum_{j=1}^{n_i} w_{ij}^c (x_{ij} - \mu_{i/k})(x_{ij} - \mu_{i/k})^T}{(m_i - 1) + \sum_{j=1}^{n_i} w_{ij}^c}$$

For k a semi-labeled sample

$$\mu_{i/k} = \frac{\sum_{j=1}^{m_i} y_{ij} + \sum_{\substack{j=1 \\ j \neq k}}^{n_i} w_{ij} x_{ij}^c}{m_i + \sum_{\substack{j=1 \\ j \neq k}}^{n_i} w_{ij}}$$

$$\Sigma_{i/k} = \frac{\sum_{j=1}^{m_i} (y_{ij} - \mu_{i/k})(y_{ij} - \mu_{i/k})^T + \sum_{\substack{j=1 \\ j \neq k}}^{n_i} w_{ij} (x_{ij} - \mu_{i/k})(x_{ij} - \mu_{i/k})^T}{m_i + \sum_{\substack{j=1 \\ j \neq k}}^{n_i} w_{ij}}$$

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

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

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

$$C_{i/k}(\alpha_i) = \begin{cases} (1 - \alpha_i) diag(\Sigma_{i/k}) + \alpha_i \Sigma_{i/k} & 0 \leq \alpha_i \leq 1 \\ (2 - \alpha_i)\Sigma_{i/k} + (\alpha_i - 1) S_{i/k} & 1 < \alpha_i \leq 2 \\ (3 - \alpha_i) S_{i/k} + (\alpha_i - 2) diag(S_{i/k}) & 2 < \alpha_i \leq 3 \end{cases} \qquad (4.15)$$

Next the mixed average log likelihood of $y_{i,j}$ and $x_{i,k}$, is computed as follows:

$$LOOL_i = \frac{1}{m_i + \sum_{k=1}^{n_i} w_{ik}} \left\{ \begin{array}{l} \sum_{k=1}^{m_i} \ln(f(y_{ik} \mid \mu_{i/k}, C_{i/k}(\alpha_i)) \\ + \sum_{k=1}^{n_i} w_{ik} \ln(f(x_{ik} \mid \mu_{i/k}, C_{i/k}(\alpha_i))) \end{array} \right\} \tag{4.16}$$

This computation is repeated for several values of α_i over the range $0 \le \alpha_i 5 3$, and the value of α_i with the highest mixed average log likelihood is selected. Once the appropriate value of α_i has been evaluated, the estimated covariance matrix mixture is calculated with all the training samples and semi-labeled samples (step **3)** and is used in the Quadratic ML classifier (step 4). For adaptive BLOOC1 and BLOOC2, the optimal value of α_i is determined in the similar way except that (4.14a) and (4.14b) are used to calculate the covariance mixture.

The direct implementation of the leave-one-out likelihood function for each class with n_i training samples and m_i semi-labeled samples would require the computation of (n_i+m_i) matrix inverse and determinants at each value of α_i. Fortunately, a more efficient implementation can be derived using the rank-one down-date of the covariance matrix [17], where the leave-one-out covariance matrix can be represented as the function of the covariance matrix. In addition, the computation of optimal can be further simplified if one assumes that $diag(S) \approx diag(S_{i/k})$ in the adaptive LOOC or adaptive bLOOC2 estimators and the approximation of $\frac{tr(S_{i/k})}{p} I \approx \frac{tr(S)}{p} I$ in the adaptive bLOOC1 estimator.

4.5 Computational Consideration

4.5.1 Efficient implementation of the adaptive LOOC estimator

Efficient Implementation of the Adaptive LOOC estimator for $1 \le \alpha_i 5 2$

If implemented directly, the computation of the proposed estimate would require computing the inverse and determinant of the (p by p) matrix $C(\alpha_i)$ for each sample, which would be quite computational expensive. Fortunately, a significant reduction in the required computation can be accomplished by writing the matrix in a fbrm that allows the determinant and inverse to be computed efficiently.

Denote $z = (y_{i1},\ldots,y_{im_i};x_{i1},\ldots,x_{im_i})$, the combination of training samples and semi-labeled samples from class i, and redefine the weighting factor w_{ij} related with sample z_{ij} as follows:

$$w_{ij} = \begin{cases} \dfrac{p(x_{ij} \mid \mu_i,\hat{\Sigma}_i(\alpha_i))}{\sum_{k=1}^{L} p(x_{ij} \mid \mu_i,\hat{\Sigma}_i(\alpha_i))} & \text{j is a semi - labeled sample} \\ 1 & \text{j is a training sample:} \end{cases} \qquad \textbf{(4.27)}$$

Then

$$\text{'ilk} = \frac{\sum_{\substack{j=1 \\ j\neq k}}^{nm_i} w_{ij}(z_{ij} - m_{i/k})(z_{ij} - m_{i/k})^T}{w_i - w_{ik}} = \frac{w_i}{w_i - w_{ik}}\left[\Sigma_i - \frac{w_i w_{ik}}{(w_i - w_{ik})^2} vv^T\right] \qquad (4.28)$$

where

$$w_i = m_i + \sum_{j=1}^{n_i} w_{ij}$$

$$nm_i = n_i + m_i$$

$$v = z_{ik} - m_i$$

The common covariance estimate without sample k from class ican be written as:

$$\begin{aligned} S_{i/k} &= \frac{1}{L}\sum_{\substack{j=1\\ j\neq i}}^{L}\Sigma_j + \frac{1}{L}\Sigma_{i/k} \\ &= \frac{1}{L}\sum_{j=1}^{L}\Sigma_j - \frac{1}{L}\Sigma_i + \frac{1}{L}\Sigma_{i/k} \\ &= S - \frac{1}{L}\Sigma_i + \frac{w_i}{L(w_i - w_{ik})}\left[\Sigma_i - \frac{w_i w_{ik}}{\left(w_i - w_{ik}\right)^2} vv^T\right] \\ &= S + \frac{w_{ik}}{L(w_i - w_{ik})}\Sigma_i - \frac{w_i^2 w_{ik}}{L\left(w_i - w_{ik}\right)^3} vv^T \end{aligned} \tag{4.29}$$

Then *the proposed estimate for* $1 \leq \alpha \leq 2$becomes

$$\begin{aligned} C_{i/k}(\alpha_i) &= (2-\alpha_i)\Sigma_{i/k} + (\alpha_i - 1)S_{i/k} \\ &= (2-\alpha_i)\frac{w_i}{w_i - w_{ik}}\left[\Sigma_i - \frac{w_i w_{ik}}{(w_i - w_{ik})^2} vv^T\right] \\ &\quad +(\alpha_i - 1)\left[S_i + \frac{w_{ik}}{L(w_i - w_{ik})}\Sigma_i - \frac{w_i^2}{L(w_i - w,)^{3}} vv^T\right] \\ &= \left[(2-\alpha_i)\frac{w_i}{w; - w_{ik}} + \frac{(\alpha_i - 1)w_{ik}}{L(w_i - w_{ik})}\right]\Sigma_i + (\alpha_i - 1)S_i \\ &\quad -\left[(2-\alpha_i)\frac{w_i^2 w_{ik}}{(w_i - w_{ik}{}^{3}} + (\alpha_i - 1)\frac{w_i^2 w_{ik}}{L(w_i - w_{ik})^3}\right] vv^T \\ &= G_1 - z_1 z_1{}^T \end{aligned} \tag{4.30}$$

where

$$G_1 = \left[(2-\alpha_i)\frac{w_i}{w_i - w_{ik}} + \frac{(\alpha_i - 1)w_{ik}}{L(w_i - w_{ik})}\right]\Sigma_i + (\alpha_i - 1)S_i$$

$$z_1 = \sqrt{k_1} v$$

$$k_1 = (2-\alpha_i)\frac{w_i^2 w_{ik}}{(w_i - w_{ik})^3} + (\alpha_i - 1)\frac{w_i^2 w_{ik}}{L(w_i - w_{ik})^3}$$

Then the inverse of $C_{i/k}(\alpha_i)$can *be computed efficientlyusing the* Sherman-Morrison-Woodbury *formula:*

$$
\begin{aligned}
C_{i/k}(\alpha_i)^{-1} &= G_1^{-1} + \frac{G_1^{-1} z_1 z_1^{T} G_1^{-1}}{1 - z^{T} G^{-1} z} \\
&= G_1^{-1} + \frac{k_1 G_1^{-1} v v^{T} G_1^{-1}}{1 - k_1 v^{T} G_1^{-1} v}
\end{aligned}
$$

$$
\begin{aligned}
z_{ik} - m_{i/k} &= z_{ik} - \left[m_i - \frac{w_i}{w_i - w_{ik}} (z_{ik} - m_i) \right] \\
&= \frac{w_i}{w_i - w_{ik}} (z_{ik} - m_i)
\end{aligned}
$$

Then the quadratic term in the Gaussian density function becomes:

$$
\begin{aligned}
d_{i/k} &= (z_{ik} - m_{i/k})^{T} C_{i/k}^{-1}(\alpha_i)(z_{ik} - m_{i/k}) \\
&= \left[\frac{w_i}{w_i - w_{ik}} \right]^2 v^{T} C_{i/k}^{-1}(\alpha_i) v \\
&= \left[\frac{w_i}{w_i - w_{ik}} \right]^2 v^{T} \left[G^{-1} + \frac{k_1 G^{-1} v v^{T} G^{-1}}{1 - k_1 v^{T} G^{-1} v} \right] v \\
&= \left[\frac{w_i}{w_i - w_{ik}} \right]^2 \frac{d}{1 - k_1 d}
\end{aligned}
\tag{4.31}
$$

where

$$
d = v^{T} G^{-1} v
$$

The determinant can also be computed efficiently:

$$
\begin{aligned}
\left| C_{i/k}(\alpha_i) \right| &= \left| G_1 - z_1 z_1^{T} \right| \\
&= |G_1| (1 - z_1^{T} G^{-1} z_1) \\
&= |G_1| (1 - k_1 v^{T} G_1^{-1} v) \\
&= |G_1| (1 - k_1 d)
\end{aligned}
$$

Finally, the log likelihood function for 15 α_i ≤ 2can be computed efficiently as follows:

$$\begin{aligned}
\ln\left[f(z_{ik})\mid m_{i/k},C_{i/k}(\alpha_i)\right] &= -\frac{p}{2}\ln(2\pi)-\frac{1}{2}\ln\left|C_{i/k}(\alpha_i)\right| \\
&\quad -\frac{1}{2}(z_{ik}-m_{i/k})C_{i/k}^{-1}(\alpha_i)(z_{ik}-m_{i/k}) \\
&= -\frac{p}{2}\ln(2\pi)-\frac{1}{2}\ln\left[|G_1|(1-k_1d)\right]\ldots \\
&\quad \ldots -\frac{1}{2}\left[\frac{w_i}{w_i-w_{ik}}\right]^2\frac{d}{1-k_1d} \\
&= -\frac{p}{2}\ln(2\pi)-\frac{1}{2}\ln(|G_1|)-\frac{1}{2}\ln(1-k_1d)\ldots \\
&\quad \ldots -\frac{1}{2}\left[\frac{w_i}{w_i-w_{ik}}\right]^2\frac{d}{1-k_1d}
\end{aligned} \qquad (4.32)$$

As indicated in equation (4.32), instead of inverting a (p by p) matrix and finding its determinant for every sample in the class, one only needs to compute the inverse and the determinant of matrix G once, and then calculate $\mathrm{d}=v^TG_1^{-1}v$ for each sample, which is relatively simple.

Efficient Implementation of the Adaptive LOOC estimator for $0\le\alpha_i$ ***5*** 1

Unfortunately, there isn't a similar method to avoid inverting a large matrix for each sample in the diagonal sample covariance-sample covariance mixture. However, if one makes the approximation that the diagonal covariance matrix changes only a little when a single sample is excluded, one can significantly reduce computation. Experiments presented later confirm the validity of this assumption when at least a moderate number of samples are available. With this assumption, the proposed estimate for 0 ***5*** $\alpha_i\le 1$can be written as follows:

$$\begin{aligned}
C_{i/k}(\alpha_i) &= (1-\alpha_i)diag(\Sigma_{i/k})+\alpha_i\Sigma_{i/k} \\
&\approx (1-\alpha_i)diag(\Sigma_i)+\alpha_i\Sigma_{i/k} \\
&= (1-\alpha_i)diag(\Sigma_i) \\
&\quad +\alpha_i\frac{w_i}{w_i-w_{ik}}\left[\Sigma_i-\frac{w_iw_{ik}}{(w_i-w_{ik})^2}vv^T\right] \\
&= (1-\alpha_i)diag(\Sigma_i)+\frac{\alpha_iw_i}{w_i-w_{ik}}\Sigma_i-\left[\frac{\alpha_iw_i^2w_{ik}}{L(w_i-w_{ik})^3}\right]vv^T \\
&= G_2-z_2z_2^{\,T}
\end{aligned}$$

$$G_2 = (1-\alpha_i)diag(\Sigma_i) + \frac{\alpha_i w_i}{w_i - w_{ik}}\Sigma_i$$
$$d_2 = v^T G_2^{-1} v$$
$$z_2 = \sqrt{k_2}v$$
$$k_2 = \frac{\alpha_i w_i^2 w_{ik}}{L(w_i - w_{ik})^3}$$

Using the same steps as in the previous section, the log likelihood function is

$$\begin{aligned}
\ln\left[f(z_{ik}) \mid m_{i/k}, C_{i/k}(\alpha_i)\right] &= -\frac{p}{2}\ln(2\pi) - \frac{1}{2}\ln\left|C_{i/k}(\alpha_i)\right| \\
&\quad -\frac{1}{2}(z_{ik} - m_{i/k})C_{i/k}^{-1}(\alpha_i)(z_{ik} - m_{i/k}) \\
&= -\frac{p}{2}\ln(2\pi) - \frac{1}{2}\ln\left[\left|G_2\right|(1-k_2 d)\right]\ldots \\
&\quad \ldots -\frac{1}{2}\left[\frac{w_i}{w_i - w_{ik}}\right]^2 \frac{d}{1-k_2 d} \\
&= -\frac{p}{2}\ln(2\pi) - \frac{1}{2}\ln(\left|G_2\right|) - \frac{1}{2}\ln(1-k_2 d)\ldots \\
&\quad \ldots -\frac{1}{2}\left[\frac{w_i}{w_i - w_{ik}}\right]^2 \frac{d}{1-k_2 d}
\end{aligned} \tag{4.33}$$

Efficient Implementation of the Adaptive LOOC estimator for $2 \le \alpha_i \le 3$

Similarly, the computation of the common covariance-diagonal common covariance mixture can be simplified by assuming the change of diagonal common covariance can be ignored when a sample is removed ($diag(S) \approx diag(S_{i/k})$). Experiments presented later confirm the validity of this assumption for moderate sample sizes.

$$
\begin{aligned}
C_{i/k}(\alpha_i) &= (3-\alpha_i)S_{i/k} + (\boldsymbol{a},- 2)diag(S_{i/k}) \\
&\approx (3-\boldsymbol{a},)S_{i/k} + (\alpha_i - 2)diag(S) \\
&= (3-\alpha_i)\left[S_i + \frac{w_{ik}}{L(w_i - w_{ik})}\Sigma_i - \frac{w_i^2 w_{ik}}{L(w_i - w_{ik})^3}vv^T\right] \\
&\quad +(\alpha_i - 2)diag(S) \\
&= G_3 - z_3 z_3^T
\end{aligned}
\qquad (4.34)
$$

where

$$
\begin{aligned}
G_3 &= (3-\alpha_i)\left[S_i + \frac{w_{ik}}{L(w_i - w_{ik})}\Sigma_i\right] + (\alpha_i - 2)diag(S) \\
z_3 &= \sqrt{k_3}v \\
k_3 &= \frac{(3-\boldsymbol{a},)w_i^2 w_{ik}}{L(w_i - w_{ik})^3} \\
d_3 &= v^T G_3^{-1} v
\end{aligned}
$$

Similarly, the log likelihood function is

$$
\begin{aligned}
\ln\left[f(z_{ik}) \mid m_{i/k}, C_{i/k}(\alpha_i)\right] &= -\frac{p}{2}\ln(2\pi) - \frac{1}{2}\ln\left|C_{i/k}(\alpha_i)\right| \\
&\quad -\frac{1}{2}(z_{ik} - m_{i/k})C_{i/k}^{-1}(\alpha_i)(z_{ik} - m_{i/k}) \\
&= -\frac{p}{2}\ln(2\pi) - \frac{1}{2}\ln\left[\left|G_3\right|(1 - k_3 d)\right]\ldots \\
&\quad \ldots -\frac{1}{2}\left[\frac{w_i}{w_i - w_{ik}}\right]^2 \frac{d}{1 - k_3 d} \\
&= -\frac{p}{2}\ln(2\pi) - \frac{1}{2}\ln(\left|G_3\right|) - \frac{1}{2}\ln(1 - k_3 d)\ldots \\
&\quad \ldots -\frac{1}{2}\left[\frac{w_i}{w_i - w_{ik}}\right]^2 \frac{d}{1 - k_3 d}
\end{aligned}
\qquad (4.35)
$$

4.5.2 Efficient Implementation of the Adaptive BLOOCl Estimator

Efficient Implementation of the Adaptive BLOOCl estimator for $1 \le \alpha_i < 2$

When the sample k is removed form class i, the sample covariance-common covariance mixture is then given by as follows:

$$C_{i/k}(\alpha_i) = (2-\alpha_i)S_{i/k} + (\alpha_i - 1)S^{*}_{p/k}(\mathrm{t})$$
$$= \frac{w_i - w_{ik}}{(w_i - w_{ik}) + t - p - 1} S_{i/k} + \frac{t-p-1}{(w_i - w_{ik}) + t - p - 1} S^{*}_{p/k}(\mathrm{t}) \qquad (4.36)$$

where ***t*** can be derived as follows:

$$t = \frac{(\alpha_i - 1)(w_i - w_{ik})}{2-\alpha_i} + p + 1$$

The pooled covariance matrix $S^{*}_{p}(t)$ without sample kfrom class i can then be derived as:

$$S^{*}_{p/k}(t) = \left[\sum_{\substack{j=1\\ j\neq i}}^{L} \frac{w_j}{w_j + t - p - 1} + \frac{w_i - w_{ik}}{w_i - w_{ik} + t - p - 1}\right]^{-1}$$
$$\left[\sum_{\substack{j=1\\ \mathrm{jti}}}^{L} \frac{w_j \Sigma_j}{w_j + t - p - 1} + \frac{w_i - w_{ik}}{w_i - w_{ik} + t - p - 1}\mathrm{i}_{ilk}\right]$$
$$= \left[\sum_{j=1}^{L} \frac{w_j}{w_j + t - p - 1} - \frac{w_i}{w_i + t - p - 1} + \frac{w_i - w_{ik}}{w_i - w_{ik} + t - p - 1}\right]^{-1} \qquad (4.37)$$
$$\left[\sum_{j=1}^{L} \frac{w_j \Sigma_j}{w_j + t - p - 1} - \frac{w_i \Sigma_i}{w_i + t - p - 1} + \frac{w_i - w_{ik}}{w_i - w_{ik} + t - p - 1}\Sigma_{i/k}\right]$$

Define:

$$c_1 = \sum_{j=1}^{L} \frac{w_j}{w_j + t - p - 1}$$

$$c_2 = \frac{w_i}{w_i + t - p - 1}$$

Then equation **(37)** can be written as follows:

$$
\begin{aligned}
S^*_{p/k}(t) &= [c, - c_2 + 2 - \alpha_i]^{-1} \\
&\quad \left[c_1 S^*_p(t) - c_2\Sigma_i + (2-\alpha_i)\Sigma_{i/k}\right] \\
&= \frac{1}{c_1 - c_2 + 2 - \alpha_i} \\
&\quad \left[c_1 S^*_p(t) - c_2\Sigma_i + \frac{(2-\alpha_i)w_i}{w_i - w_{ik}}\Sigma_i - (2-\alpha_i)\frac{w_i^2 w_{ik}}{(w_i - w_{ik})^3} vv^T\right] \qquad (4.38) \\
&= \frac{c_1}{c_1 - c_2 + 2 - \boldsymbol{a},} S^*_p(t) + \frac{-c_2 + \frac{(2-\boldsymbol{a},)w,}{w_i - w_{ik}}}{c_1 - c_2 + 2 - \boldsymbol{a},}\Sigma_i \\
&\quad - \frac{(2-\alpha_i)\frac{w_i^2 w_{ik}}{(w_i - w_{ik})^3}}{c_1 - c_2 + 2 - \alpha_i} vv^T
\end{aligned}
$$

and subsequently equation (24a) becomes:

$$
\begin{aligned}
C_i(\alpha_i) &= b_1 S^*_p(t) + b_2\Sigma_i - k_3 vv^T \\
&= G_4 - z_4 z_4^T
\end{aligned}
\qquad (4.39)
$$

where

$$G_4 = b_1 S^*_p(t) + b_2\Sigma_i$$

$$z_4 = \sqrt{k_4}\, v$$

$$b_1 = c_1(\alpha_i - 1)(c_1 - c_2 - 2 - \alpha_i)^{-1}$$

$$b_2 = \frac{(2-\alpha_i)w_i}{w_i - w_{ik}} + \frac{\alpha_i - 1}{c_1 - c_2 + 2 - \alpha_i}\left[\frac{2-\alpha_i}{w_i - w_{ik}} - c_2\right]$$

$$k_4 = \frac{(2-\alpha_i)w_i^2 w_{ik}}{(w_i - w_{ik})^3} + \frac{(\alpha_i - 1)(2-\alpha_i)}{c_1 - c_2 + 2 - \alpha_i}\frac{w_i^2 w_{ik}}{(w_i - w_{ik})^3}$$

Then the log likelihood function is given as follows:

$$
\begin{aligned}
\ln\left[f(z_{ik}) \mid m_{i/k}, C_{i/k}(\alpha_i)\right] &= -\frac{p}{2}\ln(2\pi) - \frac{1}{2}\ln\left|C_{i/k}(\alpha_i)\right| \\
&\quad -\frac{1}{2}(z_{ik} - m_{i/k})C_{i/k}^{-1}(\alpha_i)(z_{ik} - m_{i/k}) \\
&= -\frac{p}{2}\ln(2\pi) - \frac{1}{2}\ln\left[\left|G_4\right|(1-k_4 d)\right]\ldots \\
&\quad \ldots -\frac{1}{2}\left[\frac{w_i}{w_i - w_{ik}}\right]^2 \frac{d}{1-k_4 d} \\
&= -\frac{p}{2}\ln(2\pi) - \frac{1}{2}\ln(\left|G_4\right|) - \frac{1}{2}\ln(1-k_4 d)\ldots \\
&\quad \ldots -\frac{1}{2}\left[\frac{w_i}{w_i - w_{ik}}\right]^2 \frac{d}{1-k_4 d}
\end{aligned}
\tag{4.40}
$$

Efficient Implementationof the Adaptive BLOOCl Estimator for 0 5 α_i 5 1

With sample kfrom class i, the adaptive BLOOCl estimator for $0 \le \alpha_i \le 1$ becomes:

$$
\begin{aligned}
C_{i/k}(\alpha_i) &= (1-\alpha_i)\frac{tr(\Sigma_{i/k})}{p}I + \alpha_i \Sigma_{i/k} \\
&= \frac{1-\alpha_i}{p} tr\left[\frac{w_i}{w_i - w_{ik}}\Sigma_i - \frac{w_i^2 w_{ik}}{(w_i - w_{ik})^3}vv^T\right] I \\
&\quad + \alpha_i\left[\frac{w_i}{w_i - w_{ik}}\Sigma_i - \frac{w_i^2 w_{ik}}{(w_i - w_{ik})^3}vv^T\right] \\
&= \alpha_i \frac{w_i}{w_i - w_{ik}}\Sigma_i + (1-\alpha_i)\frac{w_i}{p(w_i - w_{ik})}tr(\Sigma_i)I \\
&\quad -(1-\alpha_i)\frac{w_i^2 w_{ik}}{p(w_i - w_{ik})^3}\left\|v\right\|^2 I \\
&\quad -\alpha_i \frac{w_i^2 w_{ik}}{p(w_i - w_{ik})^3}vv^T \\
&= G_5 - z_5 z_5^T
\end{aligned}
\tag{4.41}
$$

where

$$G_5 = \alpha_i \frac{w_i}{w_i - w_{ik}} \Sigma_i + (1-\alpha_i) \frac{w_i}{p(w_i - w_{ik})} tr(\Sigma_i) I$$
$$-(1-\alpha_i) \frac{w_i^2 w_{ik}}{p(w_i - w_{ik})^3} \|v\|^2 I$$

$$z_5 = \sqrt{k_5} v$$

$$k_5 = \frac{\alpha_i w_i^2 w_{ik}}{p(w_i - w_{ik})^3}$$

Therefore, the log likelihood of class iwithout sample $k$can be computed as follows:

$$\begin{aligned}
\ln\left[f(z_{ik}) \mid m_{i/k}, C_{i/k}(\alpha_i)\right] &= -\frac{p}{2}\ln(2\pi) - \frac{1}{2}\ln\left|C_{i/k}(\alpha_i)\right| \\
&\quad -\frac{1}{2}(z_{ik} - m_{i/k}) C_{i/k}^{-1}(\alpha_i)(z_{ik} - m_{i/k}) \\
&= -\frac{p}{2}\ln(2\pi) - \frac{1}{2}\ln\left[|G_5|(1-k_5 d)\right] \ldots \\
&\quad \ldots -\frac{1}{2}\left[\frac{w_i}{w_i - w_{ik}}\right]^2 \frac{d}{1-k_5 d} \\
&= -\frac{p}{2}\ln(2\pi) - \frac{1}{2}\ln(|G_5|) - \frac{1}{2}\ln(1-k_5 d) \ldots \\
&\quad \ldots -\frac{1}{2}\left[\frac{w_i}{w_i - w_{ik}}\right]^2 \frac{d}{1-k_5 d}
\end{aligned} \tag{4.42}$$

Efficient Implementation of the Adaptive BLOOC1 Estimator for 2 I α_i I 3

With sample kfrom class i, the adaptive BLOOC1 estimator for 2 **I** $a, \leq$ **3** becomes:

$$\begin{aligned}C_{i/k}(\alpha_i) &= (3-\alpha_i)S_{i/k} + (\alpha_i - 2)\frac{tr(S_{i/k})}{p}I \\ &= (3-\alpha_i)(S + \frac{w_{ik}}{L(w_i - w_{ik})}\Sigma_i - \frac{w_i^2 w_{ik}}{L\left(w_i - w_{ik}\right)^3}vv^T) \\ &+ \frac{(\alpha_i - 2)}{p}tr(S + \frac{w_{ik}}{L(w_i - w_{ik})}\Sigma_i - \frac{w_i^2 w_{ik}}{L\left(w_i - w_{ik}\right)^3}vv^T)I \\ &= G_6 - z_6 z_6^T\end{aligned} \tag{4.43}$$

where

$$\begin{aligned}G_6 &= (3-\alpha_i)S + \frac{(3-\alpha_i)w_{ik}}{L(w_i - w_{ik})}\Sigma_i \\ &+ \frac{\alpha_i - 2}{p}\left[-\frac{w_i^2 w_{ik}}{L\left(w_i - w_{ik}\right)^3}\|v\|^2 + tr(s) - \frac{w_{ik}tr(\Sigma_i)}{L(w_i - w_{ik})}\right]I\end{aligned}$$

$$z_6 = \sqrt{k_6}\,v$$

$$k_6 = \frac{(3-\alpha_i)w_i^2 w_{ik}}{L(w_i - w_{ik})^3}$$

The log likelihood function is then given as:

$$\begin{aligned}\ln\left[f(z_{ik}) \mid m_{i/k}, C_{i/k}(\alpha_i)\right] &= -\frac{p}{2}\ln(2\pi) - \frac{1}{2}\ln\left|C_{i/k}(\alpha_i)\right| \\ &-\frac{1}{2}(z_{ik} - m_{i/k})C_{i/k}^{-1}(\alpha_i)(z_{ik} - m_{i/k}) \\ &= -\frac{p}{2}\ln(2\pi) - \frac{1}{2}\ln\left[\left|G_6\right|(1-k_6 d)\right]\ldots \\ &\ldots -\frac{1}{2}\left[\frac{w_i}{w_i - w_{ik}}\right]^2 \frac{d}{1-k_6 d} \\ &= -\frac{p}{2}\ln(2\pi) - \frac{1}{2}\ln(\left|G_6\right|) - \frac{1}{2}\ln(1-k_6 d)\ldots \\ &\ldots -\frac{1}{2}\left[\frac{w_i}{w_i - w_{ik}}\right]^2 \frac{d}{1-k_6 d}\end{aligned} \tag{4.44}$$

The above computation can be further simplified if one assumes the trace of the common covariance estimate changes little when a single sample is removed ($\frac{tr(S_{i/k})}{P} I \approx \frac{tr(S)}{P} I$)

$$\begin{aligned} C_{i/k}(\alpha_i) &= (3-\alpha_i)S_{i/k} + (\alpha_i - 2)\frac{tr(S_{i/k})}{p} I \\ &\approx (3-\alpha_i)S_{i/k} + (\alpha_i - 2)\frac{tr(S)}{p} I \\ &= (3-\alpha_i)(S + \frac{w_{ik}}{L(w_i - w_{ik})}\Sigma_i - \frac{w_i^2 w_{ik}}{L(w_i - w_{ik})^3} vv^T) \qquad (4.45) \\ &+ \frac{(\alpha_i - 2)}{p} tr(S) I \\ &= G_7 - z_7 z_7^T \end{aligned}$$

$$G_7 = (3-\alpha_i)S + \frac{(3-\alpha_i)w_{ik-}}{L(w_i - w_{ik})}\Sigma_i + (\alpha_i - 2)\frac{tr(S)}{P} I$$

$$z_7 = \sqrt{k_7}\, v$$

$$k_7 = \frac{(3-\alpha_i)w_i^2 w_{ik}}{L(w_i - w_{ik})^3}$$

The log likelihood function is then given as:

$$\begin{aligned} \ln\left[f(z_{ik}) \mid m_{i/k}, C_{i/k}(\alpha_i)\right] &= -\frac{p}{2}\ln(2\pi) - \frac{1}{2}\ln\left|C_{i/k}(\alpha_i)\right| \\ &\quad -\frac{1}{2}(z_{ik} - m_{i/k})C_{i/k}^{-1}(\alpha_i)(z_{ik} - m_{i/k}) \\ &= -\frac{p}{2}\ln(2\pi) - \frac{1}{2}\ln\left[\left|G_7\right|(1-k_7 d)\right]\ldots \\ &\qquad \ldots -\frac{1}{2}\left[\frac{w_i}{w_i - w_{ik}}\right]^2 \frac{d}{1-k_7 d} \qquad (4.46) \\ &= -\frac{p}{2}\ln(2\pi) - \frac{1}{2}\ln(\left|G_7\right|) - \frac{1}{2}\ln(1-k_7 d)\ldots \\ &\qquad \ldots -\frac{1}{2}\left[\frac{w_i}{w_i - w_{ik}}\right]^2 \frac{d}{1-k_7 d} \end{aligned}$$

For notational purposes, in the following sections and experiments, the adaptive BLOOC1 without approximation is denoted as AbLOOC1-exact (Adaptive Bayesian Leave One Out Covariance Estimation), whereas the implementation with approximation is designated as AbLOOC1.

4.6 Experimental Results

In this section, the experimental results from computer-generated data are presented. Six proposed covariance estimates, namely, ALOOC, ALOOC–Exact, AbLOOC1, AbLOOC1-Exact, AbLOOC2, and AbLOOC2-Exact are used. The values of the regularized parameter are chosen to be 0, 0.25, 0.5, 0.75, 1.0,1.25. 1.5, 1.75, 2, 2.25, 2.5, 2.75, and 3. The data distributions are generated from three different covariance structures as adapted from [2]. These simulated data represent the two extremes where one covariance matrix is spherical and the other is highly elliptical. The purpose of using these different types of covariance matrices is to demonstrate that the performance of the covariance estimation techniques is affected by the underlying class covariance structure. The dimensions p are chosen to be 10, 20, 40, and 60, which represent low, medium and high dimensions.

Three sets of labeled samples are generated independently. In the first set, there are 1000 samples for each class; 10 samples are selected randonnly from the 1000 samples and subsequently used for training. The other 990 samples are: then classified and become semi-labeled samples, which are used to estimate statistics at the following iteration. In the second data set, there are 10,000 random samples for each class and they are used for testing the performance of the classifier. The third data set is generated to benchmark the performance of the proposed adaptive covariance estimator. In this data set, there are 1000 random samples for each class, then all of them are used for designing a classifier, which is then tested by using the 10,000 test samples from the second data set.

The classification accuracy obtained by classification itself is referred to as re-substitution accuracy and the classification accuracy for testing data is called hold out accuracy [1]. The convergence criterion is that the relative difference of classification accuracy between two consecutive iterations is less than 0.1%. Each experiment is repeated 10 times from which the mean and variance of the classification accuracy are computed. Since only 10 training samples are used for dimensions ranging from p=10 to p=60, the training set size is small compared to dimensionality, in particular, for p=60, the problem becomes very ill-posed because the dimension is twice the total number of training samples.

4.6.1 Experiment one: equal spherical covariance matrices

All three classes have the identity covariance matrix. The mean of the first class is at the origin. The mean of the second class is taken to be 3.0 in the first variable and zeros in the others, and the mean of the third class is 3.0 in the seconcl variable and zeros in the rest. The mean accuracy and the standard deviation in parentheses and the number of iterations to reach convergence are shown in Table 4.1, and the increment classification accuracy is highlighted by bold letters, and the mean accuracy is also plotted in Figure 4.1.

Table 4.1

Mean Classification Accuracy (%) for Experiment 1

	Accuracy	p=10	p=20	p=40	p=60
ALOOC	Initial	82.64(3.32)	75.54(5.4)	67.98(4.9)	65.32(5.26)
	Final	90.76(0.2)	90.65(0.14)	90.42(0.17)	90.62(0.16)
	Difference	**8.12(-3.12)**	**15.14(-5.26)**	**22.44(-4.73)**	**25.3(-5.1)**
	Iterations	9	10	11	13
ALOOC-exact	Initial	85.4(4.17)	84.27(2.24)	79.93(3.98)	75.5(5.12)
	Final	90.74(0.17)	90.6(0.17)	90.65(0.22)	90.51(0.11)
	Difference	**5.27(-4)**	**6.4(-2.07)**	**10.72(-3.76)**	**15(-5.01)**
	Iterations	6	8	6	10
AbLOOC1	Initial	86.88(3.8)	85.42(1.48)	79.35(2.28)	78.92(1.52)
	Final	90.68(0.2)	90.77(0.13)	90.54(0.17)	90.66().14)
	Difference	**3.8(-0.82)**	**5.35(-1.35)**	**10.72(-2.11)**	**11.74(-1.38)**
	Iterations	4	6	6	6
AbLOOC1-exact	Initial	86.65(1.51)	84.71(2.19)	80.6(1.91)	76.69(2.52)
	Final	90.76(0.14)	90.7(0.16)	90.6(0.12)	90.47(0.17)
	Difference	**4.11(-1.37)**	**5.99(-2.03)**	**10(-1.79)**	**13.78(-2.35)**
	Iterations	7	7	6	6
AbLOOC2	Initial	80.16(5.7)	77.61(4.15)	68.63(5.03)	66.5(6.42)
	Final	90.63(0.18)	90.53(0.22)	90.65(0.19)	90.48(0.17)
	Difference	**10.47(-5.52)**	**12.92(-3.93)**	**22.02(-4.84)**	**23.98(-6.25)**
	Iterations	9	10	12	12
AbLOOC2-exact	Initial	86.53(2.65)	83.77(3.34)	80.29(3.85)	75.96(3.93)
	Final	90.68(0.14)	90.77(0.14)	90.66(0.12)	90.75(0.13)
	Difference	**4.15(-2.51)**	7(-3.2)	**10.37(-3.73)**	**14.79(-3.8)**
	Iterations	6	6	6	7
Supervised	Hold out	90.55(0.11)	90.12(0.12)	88.33(0.28)	85.26(0.45)
Sample Cov.	Resubstitution	91.05(0.66)	91.51(0.59)	93.06(0.57)	95.01(0.45)

Table 4.1 cont.

The value of regularized parameters for Experiment 1

			p=10	p=20	p=40	p=60
ALOOC	class one	Initial	0(0)	0.03(0.08)	0(0)	0(0)
		Final	2.63(0.94)	2.4(1.26)	1.5(1.58)	3.0(0)
	class two	Initial	0(0)	0(0)	0(0)	0(0)
		Final	0(0)	0.03(0.08)	0(0)	0(0)
	class three	Initial	0.03(0.08)	0(0)	0(0)	0(0)
		Final	0(0)	0(0)	0(0)	0(0)
ALOOC-Exact	class one	Initial	2.60(0.92)	3.0(0)	3.0(0)	3.0(0)
		Final	2.87(0.18)	2.98(0.08)	2.98(0.08)	3.0(0)
	class two	Initial	2.90(0.18)	2.98(0.08)	3.0(0)	3.0(0)
		Final	0.30(0.78)	0.58(1.21)	1.50(1.58)	1.50(1.58)
	class three	Initial	2.60(0.93)	2.95(0.11)	3.0(0)	2.98(0.08)
		Final	0.35(0.94)	0.60(1.27)	1.50(1.58)	1.50(1.58)
AbLOOC1	class one	Initial	2.3(1.23)	2.05(1.42)	0.90(1.45)	1.80(1.55)
		Final	2.4(0.18)	2.73(0.14)	2.98(0.08)	3.0(0)
	class two	Initial	1.80(1.50)	1.75(1.51)	1.50(1.58)	1.20(1.55)
		Final	0.78(0.98)	0.60(1.20)	0(0)	0(0)
	class three	Initial	2.08(1.38)	1.73(1.48)	2.10(1.45)	1.50(1.58)
		Final	0.18(1.06)	0.03(0.08)	0(0)	0(0)
AbLOOC1-Exact	class one	Initial	1.48(1.46)	1.45(1.53)	2.40(1.27)	0.90(1.45)
		Final	2.33(0.17)	2.68(0.17)	2.95(0.11)	3.0(0)
	class two	Initial	2.23(1.12)	2.40(1.20)	1.78(1.53)	2.10(1.45)
		Final	0.60(1.14)	0.05(0.11)	0.03(0.08)	0(0)
	class three	Initial	2.08(1.43)	2.38(1.18)	0.9(1.45)	1.45(1.43)
		Final	0.73(1.08)	0.38(0.93)	0(0)	0(0)
AbLOOC2	class one	Initial	0(0)	0(0)	0(0)	0(0)
		Final	2.40(1.27)	2.10(1.45)	2.70(0.95)	2.10(1.45)
	class two	Initial	0(0)	0(0)	0(0)	0(0)
		Final	0(0)	0(0)	0(0)	0(0)
	class three	Initial	0(0)	0(0)	0(0)	0(0)
		Final	0(0)	0(0)	0(0)	0(0)
AbLOOC2-Exact	class one	Initial	2.85(0.21)	2.98(0.08)	3.0(0)	3.0(0)
		Final	2.83(0.24)	2.95(0.11)	3.0(0)	3.0(0)
	class two	Initial	2.60(0.84)	2.98(0.08)	3.0(0)	3.0(0)
		Final	0.25(0.79)	0(0)	0.3(0.95)	1.20(1.55)
	class three	Initial	0.21(0.86)	3.0(0)	3.0(0)	3.0(0)
		Final	0.24(0.08)	0.6(1.27)	1.78(1.53)	2.10(1.45)

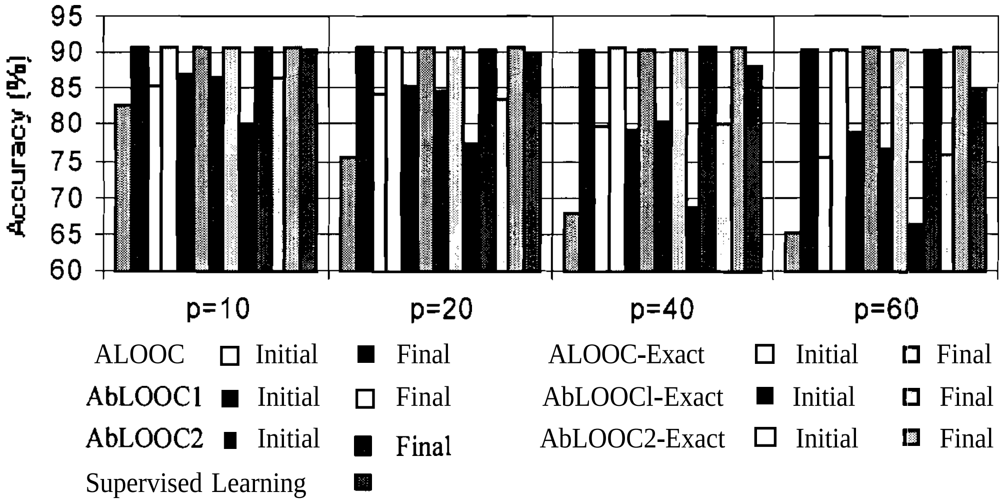


Fig. 4.1. Mean classification accuracy for experiment 1. (In color)

It is observed that even though the initial accuracy resulted from different covariance estimators where only training samples are used has large variation, with additional semi-labeled samples the final accuracy is approximately equal with a value close to the optimal. This suggests that with additional semi-labeled samples, the various proposed covariance estimators can provide comparable performance. However, it is seen the higher initial accuracy leads to slightly faster convergence. This indicates that the value of the initial accuracy has little influence on the final value of the accuracy, but it has slight effect on the rate of convergence.

Furthermore, when the number of dimensions varies from 10 to 60, even though the initial accuracy steadily decreases due to Hughes phenomenon [3], the final accuracy remains almost unchanged, and improvement on accuracy is more pronounced with higher dimensions. Also, for higher dimensions, for example 20, 40 and 60, the final accuracy is even better than the accuracy from supervised sample covariance estimators. This indicates the Hughes phenomenon [3] has been greatly alleviated. This result is very significant in that for this data set, the separability of classes remain unchanged with the number of dimensions. In other word, the increase of dimensions has no effect on discriminant power. As a result, the Hughes phenomenon [3] is severe which can be seen by the fact the dramatic decrease of the initial accuracy and the classification accuracy for supervised learning with dimensions. Further, the standard deviation is reduced with iterations. The final one is reduced by about 10-50 fold, which indicates the estimated statistics are more representative of the true ones.

Even though the initial and final mean values of the selected regularized parameter have noticeable variation, they are similar within the standard deviation. This may suggest that improvement may mostly result from the improvement of supporting covariance matrices used in the covariance estimators.

4.6.2 Experiment two: unequal spherical covariance matrices

In experiment 2, all three classes have different spherical covariance matrices and different mean vectors. The covariance of class one, two, and three is I, 21, and 31, respectively. The mean of the first class is the origin. The mean of the second class is taken to be 3.0 in the first variable and zeros in the others, and the mean of the third class is 3.0 in the second variable and zeros in the rest. The results of the experiment are presented in Table 4.2 and the mean classification accuracy for each estimator is graphed in Fig. 4.2.

Note that contrary to the first experiment, for this data set the separability of classes increases with dimensions. This suggests the potential of dramatic improvement of' accuracy as long as the class statistics can be estimated precisely in the high dimension space. It is seen that with the exception of the methods AbLOOC1 and AbLOOC1-exact, the initial accuracy from the other four methods deteriorate to various degrees when the number of dimensions increases. For instance, in the method of ALOOC-Exact the decrease of the initial accuracy is up to 15%. However, the final accuracy from all proposed methods increases with dimensions, and values of the final accuracy are quite close and much higher than the initial ones.

Table 4.2

Mean Classification Accuracy (%) for Experiment 2

	Accuracy	p=10	p=20	p=40	p=60
ALOOC	Initial	77.87(4.28)	76.26(3.97)	74.8(6.02)	74.94(4.13)
	Final	87.56(0.29)	91.28(0.24)	94.48(0.16)	96.12(0.21)
	Difference	**9.69(-3.99)**	**15.02(-3.73)**	**19.68(-5.89)**	**21.18(3.92)**
	Iterations	7	12	9	9
ALOOC -Exact	Initial	76.68(2.71)	72.68(3.17)	66.55(4.01)	61.01(4.11)
	Final	87.45(0.3)	91.22().17)	94.55(0.17)	96.24(0.11)
	Difference	**10.77(-2.41)**	**18.54(-3)**	**28(-3.84)**	**35.23(-4)**
	Iterations	11	11	7	6
AbLOOCl	Initial	82.88(2.8)	85.94(1.73)	88.27(1.14)	89.96(1.24)
	Final	87.24(0.24)	91.09(0.19)	94.54(0.16)	96.27(0.14)
	Difference	**4.36(-2.56)**	**5.15(-1.7)**	**6.27(-0.98)**	**6.31(-1.1)**
	Iterations	8	5	5	4
AbLOOC1 -Exact	Initial	82.57(2.79)	86.03(1.96)	88.88(1)	89.79(1.46)
	Final	87.72(0.2)	91.04(0.26)	94.61(0.14)	96.31(0.11)
	Difference	**5.15(-2.59)**	**5.01(-1.7)**	**5.73(-0.86)**	**6.52(-1.35)**
	Iterations	10	6	8	5
AbLOOC2	Initial	79.57(3.19)	75.59(2.97)	73.23(2.5)	72.92(4.84)
	Final	87.5(0.32)	91.23(0.11)	94.5(0.16)	96.16(0.21)
	Difference	**7.93(-2.87)**	**15.64(-2.86)**	**21.27(-2.34)**	**23.34(-4.83)**
	Iterations	6	11	9	13
AbLOOC -Exact	Initial	77.52(3.14)	70.82(5.65)	73.25(5.47)	66.45(7.04)
	Final	87.62().34)	91.18(1.81)	94.54().14)	96.29(0.17)
	Difference	**10.1(-2.8)**	**20.36(-3.84)**	**21.29(-5.33)**	**29.84(-6.87)**
	Iterations	8	7	6	5
Supervised Sample Cov.	Hold out	87.98(0.13)	92.48(0.13)	93.07(0.14)	93.22(0.2)
	Resubstitution	88.68(0.52)	92.48(0.56)	96.27(0.4)	98.26(0.2)

Table 4.2 cont.

The value of the Regularized Parameters for Experiment 2

			p=10	p=20	p=40	p=60
ALOOC	class one	Initial	0(0)	0(0)	0(0)	0(0)
		Final	0(0)	0(0)	0(0)	0(0)
	class two	Initial	0.05(0.11)	0(0)	0(0)	0(0)
		Final	2.33(0.77)	2.95(0.11)	2.70(0.95)	0.9(1.55)
	class three	Initial	0(0)	0.03(0.08)	0(0)	0(0)
		Final	0.1(0.12)	0(0)	0(0)	0(0)
ALOOC -Exact	class one	Initial	2.0(1.07)	2.90(0.18)	2.4(1.27)	3.0(0)
		Final	0.4(0.6)	0(0)	0(0)	0(0)
	class two	Initial	2.98(0.08)	2.98(0.08)	3.0(0)	3.0(0)
		Final	2.33(0.90)	2.83(0.12)	3.0(0)	3.0(0)
	class three	Initial	1.70(1.47)	1.20(1.55)	1.2(1.55)	0.9(1.45)
		Final	0.28(0.19)	0.05(0.11)	0(0)	0(0)
AbLOOC1	class one	Initial	0.28(0.87)	0.05(0.11)	0.03(0.08)	0(0)
		Final	0.85(0.54)	0.03(0.08)	0(0)	0(0)
	class two	Initial	1.18(1.42)	1.53(1.56)	1.8(1.55)	1.50(1.58)
		Final	2.03(0.30)	2.75(0.12)	3.0(0)	3.0(0)
	class three	Initial	0(0)	0.03(0.08)	0(0)	0(0)
		Final	0.33(0.17)	0(0)	0(0)	0(0)
AbLOOC1 -Exact	class one	Initial	0.05(0.11)	0(0)	0(0)	0(0)
		Final	1.25(0)	0(0)	0(0)	0(0)
	class two	Initial	1.45(1.49)	1.83(1.52)	1.8(1.55)	1.8(1.55)
		Final	2.33(0.12)	2.80(0.11)	3.0(0)	3.0(0)
	class three	Initial	0.03(0.08)	0.05(0.11)	0(0)	0(0)
		Final	0.3(0.20)	0.03(0.08)	0(0)	0(0)
AbLOOC2	class one	Initial	0.03(0.08)	0(0)	0(0)	0(0)
		Final	0.03(0.08)	0(0)	0(0)	0(0)
	class two	Initial	0.03(0.08)	0(0)	0(0)	0(0)
		Final	2.23(0.9)	2.98(0.08)	3(0)	1.20(1.55)
	class three	Initial	0(0)	0(0)	0(0)	0(0)
		Final	0.13(0.13)	0(0)	0(0)	0(0)
AbLOOC2 -Exact	class one	Initial	2.05(1.23)	2.93(0.17)	2.32(1.23)	2.98(0.08)
		Final	0.35(0.50)	0(0)	0.03(0.08)	0(0)
	class two	Initial	2.58(0.91)	2.93(0.12)	3.0(0)	3.0(0)
		Final	2.55(0.18)	2.85(0.13)	3.0(0)	3.0(0)
	class three	Initial	0.58(1.21)	1.23(1.52)	0.6(1.26)	0.6(1.26)
		Final	0.23(0.08)	0.03(0.08)	0(0)	0(0)

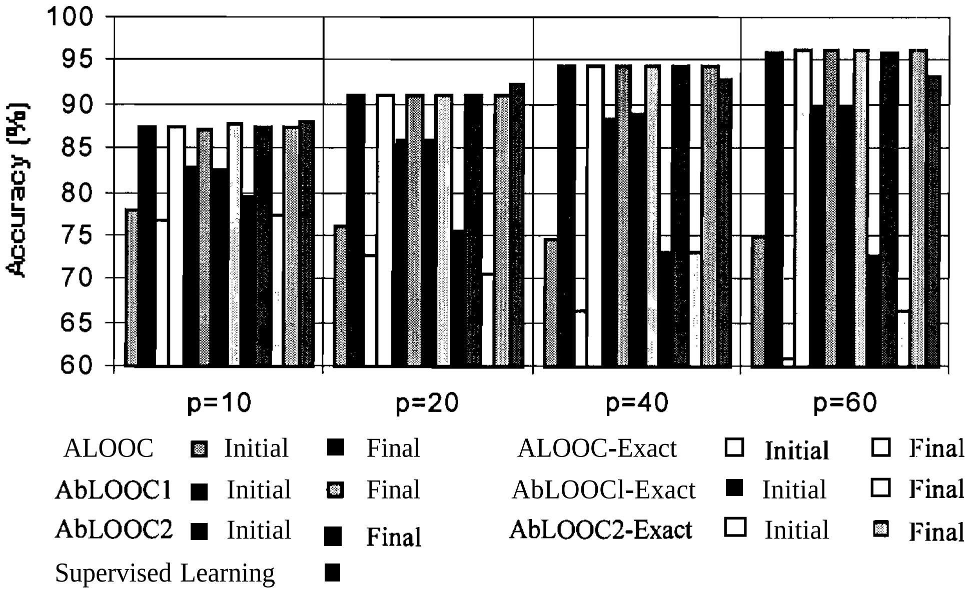


Fig. 4.2. Mean Classification Accuracy for Experiment 2. (In color)

4.6.3 Experiment three: equal elliptical covariance matrices with mean difference in the variable with low variance

In this experiment, all three classes have the same highly elliptical covariance matrix, and the primary difference in the mean vectors is in the variables with low variance. The covariance matrix for all three classes is a diagonal matrix whose diagonal elements are given by $\sigma_i = \left[\frac{9(i-1)}{(p-1)}+1\right]^2$ $1 \le i \le p$. The mean vector of the first class is the origin, the elements of the mean vector of the second class are given by $\mu_{2,i} = 2.5\sqrt{\frac{\sigma_i}{P}}\left[\frac{p-1}{\frac{p}{2}-1}\right]$ $1 \le i \le \text{p}$, and the mean of class three is defined by $\mu_{3,i} = (-1)^i \mu_{3,i}$ $1 \le i \le p$. See Table 4.3 and Figure 4.3 for the results.

Table 4.3

Mean Classification Accuracy (%) for Experiment 3

	Accuracy	p=10	p=20	p=40	p=60
ALOOC	Initial	86.6(2.34)	85.12(1.38)	78.82(3)	76.09(3.25)
	Final	93.53(0.12)	91.56().14)	90.4(0.18)	90.1(0.2)
	Difference	**6.93(-2.22)**	**6.44(-1.24)**	**11.58(-2.82)**	**14.01(-3.05)**
	Iterations	7	5	6	8
ALOOC -Exact	Initial	90.39(1.36)	84.35(3.43)	79.48(2.51)	74.73(1.88)
	Final	93.51(0.13)	91.53(0.2)	90.48(0.16)	90.06(0.17)
	Difference	**3.12(-1.23)**	**7.18(-3.23)**	**11(-2.35)**	**15.33(-1.71)**
	Iterations	5	4	6	6
AbLOOC1	Initial	77.71(7.83)	67.09(3.8)	56.77(2.95)	53.77(1.89)
	Final	93.41(0.11)	91.28(0.18)	89.81(0.2)	87.05(6.65)
	Difference	**15.7(-7.72)**	**24.19(-3.62)**	**33.04(-2.75)**	**33.28(4.76)**
	Iterations	9	13	23	40
AbLOOC1 -Exact	Initial	80.31(3.28)	67.3(3.29)	56.77(2.96)	53.39(2.22)
	Final	93.49(0.15)	91.42(0.17)	89.78(0.22)	85.17(12.46)
	Difference	**13.18(-3.13)**	**24.12(-3.12)**	**33.01(-2.74)**	**31.78(10.24)**
	Iterations	7	11	20	29
AbLOOC2	Initial	86.37(3.78)	79.07(4.68)	69.85(2.39)	64.54(3.04)
	Final	93.52(0.11)	91.51(0.17)	90.4(0.13)	89.95(0.18)
	Difference	**7.15(-3.67)**	**12.44(-4.51)**	**20.55(-2.26)**	**25.41(-2.86)**
	Iterations	8	10	5	5
AbLOOC -Exact	Initial	90.79(1.4)	85.47(1.9)	80.16(2.54)	75.29(2.65)
	Final	93.61(0.14)	91.52(0.15)	90.44(0.18)	90.06(0.21)
	Difference	**2.82(-1.26)**	**6.05(-1.75)**	**10.28(-2.36)**	**14.77(-2.44)**
	Iterations	7	7	5	7
Supervised Sample Cov.	Hold out	93.32(0.11)	90.94(0.09)	88.04(0.17)	84.74(0.33)
	Resubstitution	93.47(0.44)	92.2(0.33)	93.09(0.30)	94.82(0.27)

Table 4.3 cont.

The Value of the Regularized Parameters for Experiment 3

			p=10	p=20	p=40	p=60
ALOOC	class one	Initial	0.05(0.11)	2.95(0.11)	3.0(0)	3.0(0)
		Final	2.40(0.27)	2.73(0.18)	2.95(0.11)	3.0(0)
	class two	Initial	0(0)	2.95(0.11)	3.0(0)	3.0(0)
		Final	0.03(0.08)	0.95(1.36)	1.48(1.56)	1.50(1.58)
	class three	Initial	0(0)	2.95(1.23)	2.93(0.12)	3.0(0)
		Final	0.08(0.12)	1.23(1.43)	1.78(1.53)	1.20(1.55)
ALOOC -Exact	class one	Initial	2.86(0.20)	3.0(0)	3.0(0)	3.0(0)
		Final	2.28(0.15)	2.68(0.21)	2.95(0.11)	3.0(0)
	class two	Initial	2.88(0.18)	2.98(0.08)	3.0(0)	3.0(0)
		Final	2.20(1.05)	1.70(1.42)	0.6(1.26)	1.20(1.55)
	class three	Initial	2.65(0.94)	3.0(0)	3.0(0)	3.0(0)
		Final	0.95(1.19)	1.78(1.38)	1.80(1.55)	1.20(1.55)
AbLOOC1	class one	Initial	2.23(0.08)	2.38(0.13)	2.73(0.08)	2.75(0)
		Final	1.93(0.12)	2.0(0)	2.0(0)	1.98(0.08)
	class two	Initial	2.23(0.08)	2.40(0.13)	2.68(0.12)	2.75(0)
		Final	1.73(0.19)	1.85(0.13)	1.98(0.08)	1.95(0.11)
	class three	Initial	2.15(0.18)	2.48(0.15)	2.70(0.11)	2.75(0)
		Final	1.70(0.20)	1.88(0.13)	2.0(0)	1.98(0.08)
AbLOOC1 -Exact	class one	Initial	2.23(0.08)	2.48(0.19)	2.73(0.08)	2.75(0)
		Final	1.98(0.08)	2.0(0)	2.0(0)	2.0(0)
	class two	Initial	2.18(0.12)	2.48(0.15)	2.68(0.12)	2.75(0)
		Final	1.73(0.14)	1.93(0.12)	1.95(0.11)	1.95(0.11)
	class three	Initial	2.23(0.08)	2.43(0.17)	2.70(0.11)	2.75(0)
		Final	1.80(0.16)	1.88(0.13)	2.0(0)	1.98(0.08)
AbLOOC2	class one	Initial	0.05(0.11)	0.0(0)	0(0)	0(0)
		Final	0.03(0.08)	2.80(0.16)	2.95(0.11)	3.0(0)
	class two	Initial	0(0)	0.03(0.08)	0(0)	0(0)
		Final	0.33(0.77)	0.05(0.11)	0(0)	0(0)
	class three	Initial	0(0)	0(0)	0(0)	0(0)
		Final	0.15(0.13)	0.03(0.08)	0(0)	0(0)
AbLOOC2 -Exact	class one	Initial	2.60(0.93)	2.95(0.11)	3.0(0)	3.0(0)
		Final	2.30(0.35)	2.60(0.13)	2.93(0.12)	3.0(0)
	class two	Initial	2.88(0.21)	2.96(0.11)	3.0(0)	3.0(0)
		Final	1.20(1.23)	1.40(1.38)	0.9(1.45)	0.6(1.27)
	class three	Initial	2.83(0.21)	2.93(0.13)	3(0)	3.0(0)
		Final	0.70(1.03)	0.93(1.32)	0.88(1.42)	1.8(1.55)

Fig. 4.3 reveals that both the initial accuracy and the final accuracy decrease with dimensions. However, the value of the final accuracy is still much higher than the initial one, and the increment increases with dimensions, indicating that the Hughes phenomenon [3] has been alleviated to some degree.

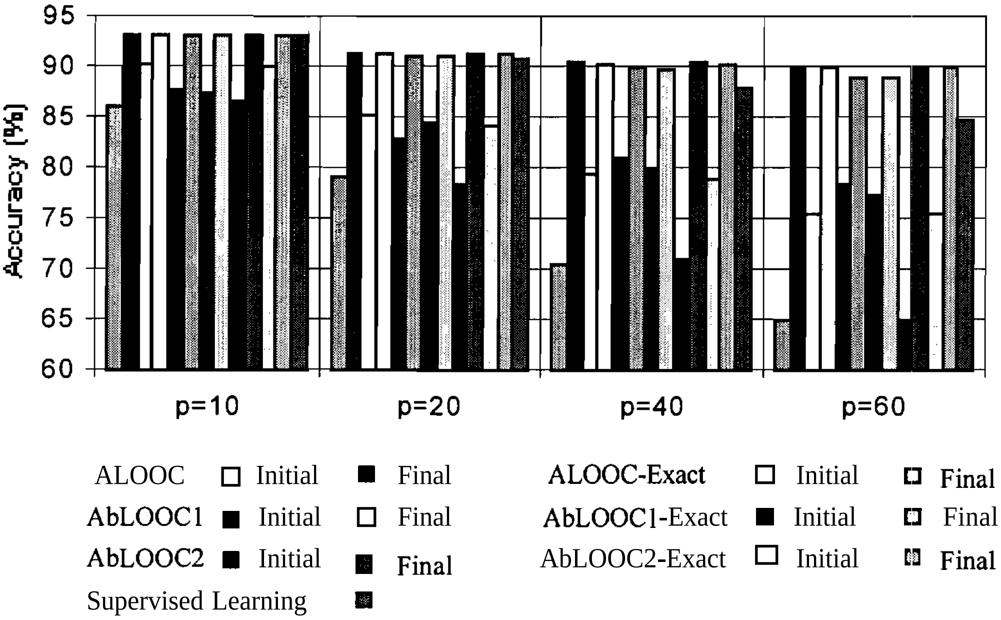


Fig. 4.3. Mean classification accuracy for experiment 3. (In color)

The initial accuracy of AbLOOCl and AbLOOC1-Exact is much lower than that from the other four methods at the high dimension where p=40 and p=60, for example about 10-20%. As a result, the final accuracy from these two methods is noticeably lower too, about 1% at p=40 and 5% at p=60, and convergence rate (not shown here) has been very slow because the initial accuracy is too low. However, the improvement of the accuracy is still very significant, about 33% at p=40 and 31% at p=60.

4.6.4 Experiment four: equal elliptical covariance matrices with mean difference in the variable with high variance

In this experiment, the same highly elliptical covariance matrix from experiment 3 is again used for all three classes. However, the difference in mean vectors occurs in the variables that have high variance. The mean vector of the first class is again the origin, the elements of the mean vector of the second class are given by

$\mu_{2,i} = 2.5\sqrt{\frac{\sigma_i}{p}}\left[\frac{i-1}{\frac{p}{2}-1}\right].$ 1≤ i ≤ p, and the mean of class three is defined by

, =(− 1) 1■ i ≤ p. See Table 4.4 and Fig. 4.4 for the results.

Due to the difficulty of statistics estimation of this data set at the high dimensions, the initial accuracy from all method drops dramatically, about 20% drop from p=10 to p=:60. As a result the final accuracy decrease slightly, about 4% drop from p=10 to p=60. However, all final ones are much higher than the initial ones, and the increment of the classification accuracy increases with the dimensionality. Again, ABLOOC1 and AHLOOC1-Exact slightly under-perform all other four methods.

Table 4.4

Mean Classification Accuracy (%) for Experiment 4

	Accuracy (%)	p=10	p=20	p=40	p=60
ALOOC	Initial	86.02(2.08)	79.19(2.71)	70.53(2.51)	65.04(3.09)
	Final	93.2(0.13)	91.29(0.12)	90.41(0.12)	89.96(0.16)
	Difference	**7.18(-1.95)**	**12.1(-2.59)**	**19.88(-2.39)**	**24.92(-2.93)**
	Iterations	5	6	7	7
ALOOC -Exact	Initial	90.37(1.3)	85.24(2.11)	79.35(1.85)	75.55(2.12)
	Final	93.19(0.09)	91.29(0.17)	90.33(0.23)	89.99(0.24)
	Difference	**2.82(-1.21)**	**6.05(-1.94)**	**10.98(-1.62)**	**14.44(-1.88)**
	Iterations	7	8	7	7
AbLOOC1	Initial	87.57(5.62)	82.88(2.88)	80.94(9.01)	78.37(2.74)
	Final	93.19(2.34)	91.1(0.2)	89.95(3.16)	88.84(0.27)
	Difference	**15.7(-7.72)**	**8.22(-2.68)**	**9.01(-2.95)**	**10.47(-2.47)**
	Iterations	8	6	7	8
AbLOOC1 -Exact	Initial	87.42(3.23)	84.57(1.71)	79.94(3.04)	77.36(2.56)
	Final	93.07(0.6)	91.03(0.19)	89.77(0.28)	88.9(0.22)
	Difference	**5.65(-3.07)**	**6.46(-1.52)**	**9.83(-2.76)**	**11.54(-2.34)**
	Iterations	7	7	8	9
AbLOOC2	Initial	86.45(1.76)	78.52(2.91)	71.02(3.21)	64.97(2.77)
	Final	93.24(0.14)	91.37(0.17)	90.41(0.16)	90.02(0.15)
	Difference	**6.79(-1.62)**	**12.85(-2.74)**	**19.39(-3.05)**	**25.05(-2.62)**
	Iterations	7	6	7	7
AbLOOC2 -Exact	Initial	89.9(1.46)	84.33(2.51)	79.01(2.94)	75.4(2.5)
	Final	93.28(0.9)	91.33(0.15)	90.35(0.19)	90(0.17)
	Difference	**3.38(-0.55)**	**7(-2.36)**	**11.34(-2.75)**	**14.6(-2.33)**
	Iterations	5	7	6	7
Supervised Sample Cov.	Hold out	93.26(0.24)	92.16(0.36)	92.96(0.46)	94.87(0.30)
	Resubstitution	93.07(0.21)	90.74(0.17)	87.98(0.17)	84.76(0.4)

Table 4.4 cont.

The Value of the Regularized Parameters for Experiment 4

			p=10	p=20	p=40	p=60
ALOOC	class one	Initial	0(0)	0(0)	0(0)	0(0)
		Final	2.58(0.24)	2.53(0.81)	3.0(0)	3.0(0)
	class two	Initial	0.03(0.08)	0(0)	0(0)	0(0)
		Final	0.08(0.12)	0.03(0.08)	0(0)	0(0)
	class three	Initial	0(0)	0(0)	0(0)	0(0)
		Final	0.38(0.94)	0(0)	0(0)	0(0)
ALOOC -Exact	class one	Initial	2.83(0.21)	2.95(0.11)	3.0(0)	3.0(0)
		Final	2.38(0.27)	2.63(0.18	2.88(0.13)	3.0(0)
	class two	Initial	2.90(0.18)	2.95(0.11)	3.0(0)	3.0(0)
		Final	1.03(1.07)	2.28(1.08)	0.88(1.41)	1.80(1.55)
	class three	Initial	2.88(0.24)	2.93(0.12)	3.0(0)	1.20(1.55)
		Final	1.03(1.14)	1.98(1.32)	1.50(1.58)	1.20(1.55)
AbLOOC1	class one	Initial	2.23(0.08)	2.23(0.71)	2.65(0.13)	2.75(0)
		Final	1.98(0.08)	1.98(0.08)	2.0(0)	2.0(0)
	class two	Initial	2.10(0.49)	2.20(0.61)	2.68(0.12)	2.48(0.87)
		Final	1.65(0.17)	1.85(0.13)	2.0(0)	2.0(0)
	class three	Initial	2.23(0.08)	2.38(0.13)	2.73(0.08)	2.75(0)
		Final	1.73(0.22)	1.88(0.14)	1.98(0.08)	2.0(0)
AbLOOC1 -Exact	class one	Initial	2.25(0)	2.33(0.12)	2.68(0.12)	2.76(0)
		Final	1.98(0.08)	2.0(0.12)	2.0(0)	2.0(0)
	class two	Initial	2.23(0.18)	2.35(0.13)	2.68(0.12)	2.76(0)
		Final	1.73(0.14)	1.85(0.13)	1.98(0.08)	2.0(0)
	class three	Initial	2.07(0.47)	2.40(0.13)	2.68(0.12)	2.76(0)
		Final	1.80(0.11)	1.88(0.13)	2.0(0)	2.0(0)
AbLOOC2	class one	Initial	0.03(0.08)	0(0)	0(0)	0(0)
		Final	2.03(1.04)	2.73(0.08)	3.0(0)	3.0(0)
	class two	Initial	0.03(0.08)	0(0)	0(0)	0(0)
		Final	0.13(0.14)	0.03(0.08)	0(0)	0(0)
	class three	Initial	0.05(0.11)	0.03(0.08)	0(0)	0(0)
		Final	0.08(0.12)	0.03(0.08)	0(0)	0(0)
AbLOOC2 -Exact	class one	Initial	2.90(0.17)	2.98(0.08)	3.0(0)	3.0(0)
		Final	2.35(0.27)	2.63(0.14)	2.95(0.11)	2.98(0.08)
	class two	Initial	2.88(0.18)	2.98(0.08)	3.0(0)	3.0(0)
		Final	1.73(1.28)	1.45(1.43)	0(0)	1.5(1.58)
	class three	Initial	2.65(0.86)	2.98(0.08)	3.0(0)	2.98(0.08)
		Final	1.43(1.35)	1.73(1.44)	1.15(1.49)	0.60(1.27)

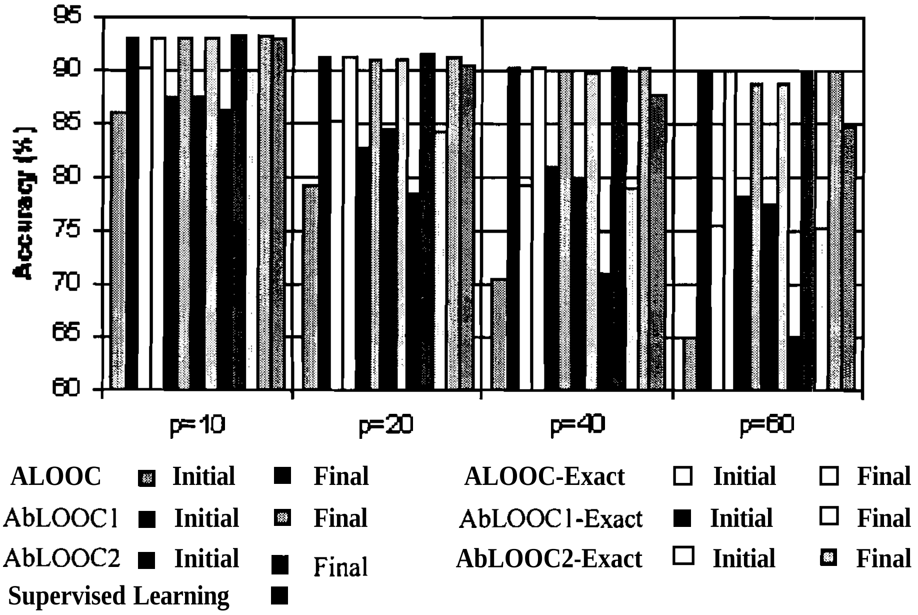


Fig. 4.4. Mean Accuracy for Experiment 4. (In color)

The following experiments are performed on AVIRIS data collected in 1992. Several samples of various ground cover classes are identified in each of the scenes. Initially a small percentage of the samples are selected at random and used to estimate the class mean vectors and covariance matrices, and the remaining samples are classified. For the following iterations, all the classified samples (semi-labeled samples) in addition to the training samples are used to enhance the mean vectors and covariance matrices, and then all the samples are reclassified. The iteration is repeated until convergence is reached. Convergence is assumed to have occurred when the classification accuracy has less than 0.1% change. The experiment is repeated ten times, and the mean and standard deviation of the ten classification accuracies are obtained.

The previous results from simulation data indicate that estimators AbLOOCl and AbLOOC1-Exact do not perform as well as the other four estimators in some cases. For this reason, these two estimators are not considered in the following experiments. In the analysis of the hyperspectral data, feature extraction is often employed to reduce dimensionality. Hence, discriminant analysis feature extraction (DAFE) [1] is incorporated in this experiment to demonstrate the effect of covariance estimators on the classification process.

4.6.5 Experiment 5: Cuprite, Nevada scene data

In experiment 4, Cuprite, Nevada scene is used, which 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 ym) are used in the experiment, and then 7 features are extracted using DAFE and classification is performed in the subspace formed by these features. 1% labeled samples are randomly selected as training samples, and the rest are used as testing samples. The number of labeled and training samples in each class is shown in table 4.5, and the experiment results are shown in table 4.6. The overall mean classification accuracy is depicted in Fig. 4.5.

Table 4.5

Training Samples Information for Experiment 5

	Labeled Samples	Training Samples
Alunite	729	7
Buddingtonite	71	1
Kaolinite	232	2
Quartz	385	4
Alluvium	689	7
Playa	252	3
Tuff	293	3
Argillized	93	1
Total Samples	2744	27

In this experiment, extremely small training sets are deliberately selected. The total number of training samples is much less than the original 191 bands, and it is just slighter greater than the number of the extracted features, which is 7. In addition, there are two classes, i.e., Buddingtonite and Argilized, which have only one training sample. For this reason, the initial overall mean classification accuracy is low and for most of the classes, individual classification accuracies are quite low, too. However, with the adaptive process, the final overall mean classification accuracy is increased by about 10%-15%, up to above 90% with much smaller standard deviation. In particular, for most of classes, the individual mean classification accuracy improvement is very impressive, for instance, up to 20%-30%. The reduction of standard deviation is significant, too, except for one class, Kaolinite. Here the final standard deviation is higher than the initial one. The reason for this exception is that the initial classification accuracy is quite bad, making it very difficult to improve final accuracy to the near optimal value. We observed that at each iteration, if the initial accuracy is quite low then final accuracy value tends to be. low too. This indicates that the initial accuracy not only has an effect on the convergence rate but also the final convergence value too. This can also observed by the overall classification accuracy achieved by the combination of the adaptive process with different covariance estimators, except for AbLOOC2+DAFE, which starts with slightly lower initial accuracy, but achieves highest final accuracy with low standard deviation. However, even though ALOOC and AbLOOC have the lower initial accuracy, they generate highest classification increment.

Table 4.6

Mean Accuracy (%) for Experimental 5

	Accuracy (%)	Overall	Alunite	Budding-tonite	Kaolinite	Quartz	Alluvium	Playa	Tuff	Argillized
ALOOC + DAFE	Initial	74.6(17.2)	98.4(1.2)	71.5(20.3)	46.0(19.1)	64.4(26.1)	79.6(16.4)	100(0)	70.6(13.8)	55.3(19.3)
	Final	90.2(7.9)	99(0.7)	93.66(19.0)	72.2(25.8)	92.2(7.5)	81.7(15.6)	100(0)	93.4(5.1)	93.1(6.3)
	Difference	**15.3(-9.3)**	**0.6(-0.5)**	**22.1(-1.4)**	**26.2(6.7)**	**27.8(-18.6)**	**2.2(-1.0)**	**0(0)**	**22.8(-8.6)**	**37.9(-13.0)**
	Iteration	15	2	10	14	15	13	2	14	14
ALOOC-Exact +DAFE	Initial	82.2(5.9)	98.0(1.5)	66.3(22.5)	53.5(18.2)	72.8(14.5)	83.1(13.6)	100(0)	70.8(19.4)	60.5(21.3)
	Final	93.3(2.7)	98.1(1.4)	95.2(14.4)	81.2(23.2)	95.8(6.4)	87.8(11.2)	100(0)	95.6(4.9)	89.4(18.9)
	Difference	**11.1(-3.2)**	**0.1(-0.1)**	**28.9(-8.2)**	**27.6(5.0)**	**23.0(-8.2)**	**4.7(-2.3)**	**0(0)**	**24.9(-14.5)**	**28.8(-2.4)**
	Iteration	15	15	7	15	15	13	2	15	11
AbLOOC2 +DAFE	Initial	78.8(3.8)	95.3(10.1)	64.8(22.8)	55.5(12.8)	50.8(21.2)	86.9(12.5)	100(0)	65.9(18.5)	57.1(24.3)
	Final	94.1(3.8)	96.5(6.2)	93.5(13.9)	87.5(15.6)	96.2(6.9)	91.2(11.3)	100(0)	94.5(4.8)	85.9(3.1)
	Difference	**15.3(-0.0)**	**1.2(-3.9)**	**28.7(-8.7)**	**32.0(2.8)**	**45.4(-14.2)**	**4.3(-1.2)**	**0(0)**	**28.6(-13.7)**	**28.8(3.1)**
	Iteration	15	15	15	15	15	15	2	15	15
AbLOOC2 -Exact2 +DAFE	Initial	80.2(4.1)	92.0(9.1)	55.6(17.7)	42.8(16.8)	63.6(16.7)	90.3(10.2)	100(0)	71.3(19.3)	67.4(22.7)
	Final	91.5(5.3)	97.8(3.0)	87.8(22.4)	67.2(23.9)	91.7(8.7)	90.2(9.1)	100(0)	92.1(15.2)	93.0(10.6)
	Difference	**11.4(1.2)**	**5.9(-6.1)**	**32.1(4.7)**	**24.4(7.1)**	**28.1(-7.9)**	**-0.1(-1.0)**	**0(0)**	**20.8(-4.1)**	**25.6(-12.1)**
	Iteration	14	7	11	14	14	12	2	14	14

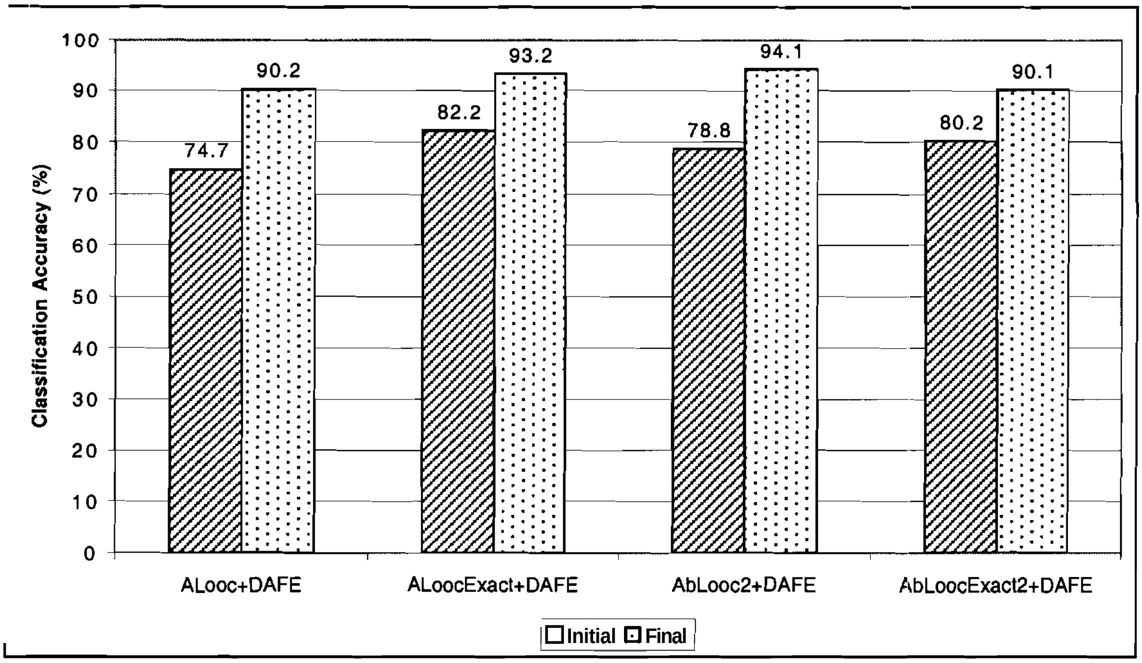


Fig. 4.5. Overall Mean Accuracy for Experiment 5

4.6.6 Experiment 6: Jasper ridge site data

In this experiment, data taken over the Jasper Ridge site is used. This is a biological preserve in San Mateo County, California. In all, 3207 labeled samples are used. The 193 spectral bands (0.40-1.34, 1.43-1.80, and 1.95-2.47 μm) outside the water absorption bands are used. Using DAFE, five features are selected and subsequently classification is performed in the subspace. 0.5% labeled samples are randomly selected as training samples, and the rest samples are used as testing samples. The number of labeled and training samples in each class is shown in Table 4.7, and the classification results are shown in Table 4.8. The overall mean classification accuracy is graphed in Fig. 4.6.

As in experiment 5, a very small training set is used in this experiment to simulate a very ill-posed problem. However, the initial mean accuracy is relatively high because the classes might be more separated. For this reason, the final mean accuracy is able to reach near optimal value with a much smaller standard deviation and with fewer iterations. Again, it is seen that the initial mean accuracy affects the final value of the accuracy. AbLOOCExact2 produces the highest final classification accuracy with lowest standard deviation even though it starts with slightly lower initial classification accuracy and highest standard deviation.

Table 4.7

Training Samples for Experiment 6

	Labeled Samples	Training Samples
Evergreen	900	5
Serpentine	202	1
Green-stone	810	4
Water	208	1
Deciduous	495	2
Chaparral	592	3
Total Samples	3207	16

Table 4.8

Classification Results for Experimental 6

	Accuracy (%)	Overall	Evergreen	Serpentine	Greenstone	Water	Deciduous	Chaparral
ALOOC+ DAFE	Initial	91.7(3.1)	95.8(5.5)	84.3(16.4)	96.1(7.5)	61(17.0)	91.0(13.6)	93.3(5.5)
	Final	97.4(1.3)	99.6(0.2)	98.1(0.9)	98.6(3.1)	70.8(17.0)	100(0)	99.5(0.2)
	Difference	**5.7(-1.9)**	**3.8(-5.3)**	**13.8(-15.5)**	**2.5(-4.4)**	**9.8(0.0)**	**9.0(-13.6)**	**6.2(-5.3)**
	Iterations	12	6	9	11	6	6	12
ALOOC-Exact +DAFE	Initial	92.9(2.3)	98.9(1.7)	91.0(14.2)	95.0(4.0)	58.1(19.6)	95.6(4.4)	91.7(4.5)
	Final	97.3(1.8)	99.7(0.2)	98.8(0.3)	97.6(3.0)	71.4(28.3)	100(0)	99.3(0.3)
	Difference	**4.4(-0.5)**	**0.8(-1.6)**	**8.3(-13.8)**	**2.64(-1.0)**	**13.2(8.7)**	**4.4(-4.3)**	**7.7(-4.2)**
	Iterations	12	9	11	10	5	3	11
ABLOOC2 + DAFE	Initial	88.1(5.5)	90.7(13.7)	81.9(25.7)	88.4(21.8)	58.9(17.4)	96.4(3.0)	89.4(10.3)
	Final	95.7(4.6)	99.7(3.1)	90.4(25.3)	92.7(17.9)	74.3(18.2)	100(0)	99.6(0.1)
	Difference	**7.6(-0.9)**	**9.1(-10.6)**	**8.5(-0.4)**	**4.2(-3.8)**	**15.4(0.8)**	**3.64(-3.0)**	**10.2(-10.2)**
	Iterations	12	12	12	13	7	3	9
ABLOOC Exact2+ DAFE	**Initial**	90.7(10.0)	90.1(19.8)	89.8(14.0)	97.1(2.8)	66.7(10.9)	96.7(3.9)	86.6(21.3)
	Final	98.2(1.1)	99.6(0.2)	98.5(0.8)	99.1(2.0)	90.9(12.6)	100(0)	99.4(0.4)
	Difference	**7.5(-8.8)**	**9.6(-19.6)**	**8.7(-13.2)**	**2.0(-0.9)**	**24.2(1.7)**	**3.3(-3.9)**	**12.8(-20.8)**
	Iterations	13	13	9	13	12	3	9

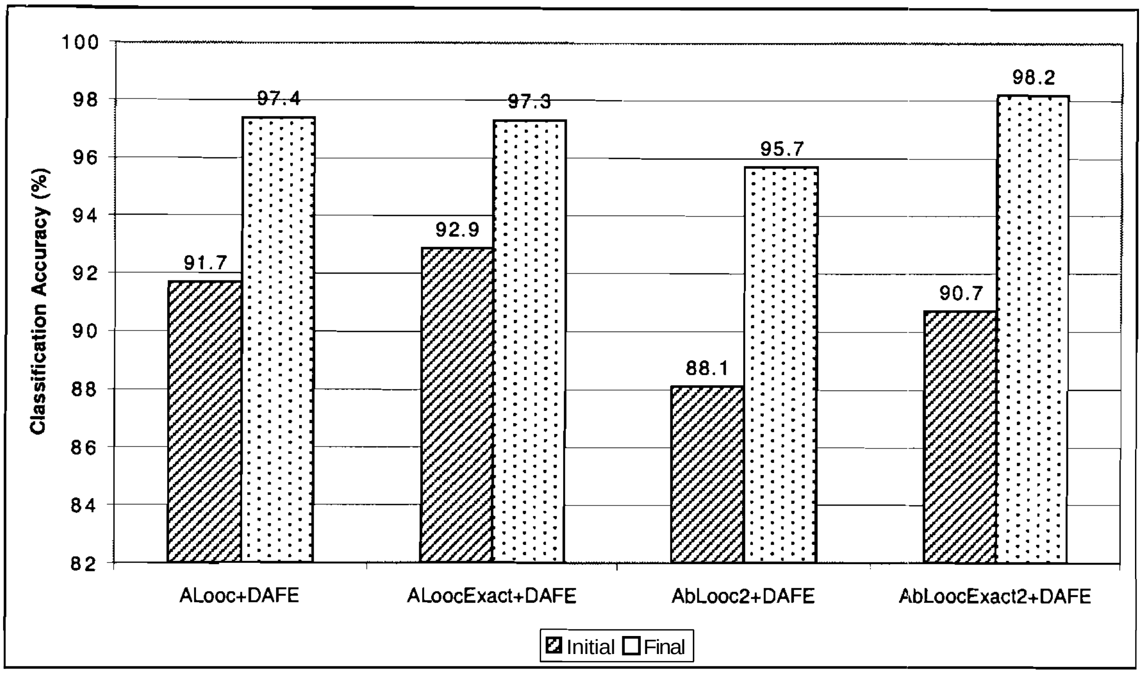


Fig. 4.6. Overall Mean Accuracy for Experiment 6

4.6.7 Experiment 7: Indian pine

In this experiment, the data taken over the Indian Pine test site is used. This is a mixed forest/agricultural area in Indiana. The water absorption bands (104-108, 150-163, 220) have been discarded, and 5 features from among the total of 191 bands are extracted using DAFE. Of the total of 2521 labeled samples, 1% labeled samples are used as training samples. See table 4.9 for the number of training samples in each class in detail. Table 4.10 shows the results, and Fig. 4.7 illustrates the overall mean classification accuracy.

The classes in this data set are highly overlapped, making classification quite challenging, because the initial overall classification accuracy and most of individual initial class ones are quite low; some of them are even below 50%. ALOOCExact produces the highest final classification accuracy with the highest initial one, while AbLOOC2 leads to the highest increment of the classification accuracy for the lowest initial one.

Table 4.9

Training Samples for Experiment 7

	Labeled Samples	Training Samples
Beans/Corn Residue	520	5
Corn/No Residue	450	5
Corn/Bean Residue	372	4
Beans/No Residue	490	5
Corn/Wheat Residue	388	4
Wheat/No Residue	301	3
Total Samples	2521	25

Table 4.10

Mean Classification Accuracy (%) for Experiment 7

	Accuracy (%)	Overall	Beans/Corn Residue	Corn/No Residue	Corn/Bean Residue	Beans/No Residue	Corn/Wheat Residue	Wheat/No Residue
ALOOC+ DAFE	Initial	52.4(3.8)	81.6(15.1)	52.7(23.2)	22.5(19.3)	36.9(24.3)	29.6(18.7)	93.3(8.8)
	Final	67.83(5.2)	73.77(9.1)	90.1(14.1)	44.5(34.0)	62.0(16.6)	40.9(22.9)	100(0)
	Difference	**15.4(1.4)**	**-7.8(-6.1)**	**37.4(-9.0)**	**22.0(14.7)**	**25.1(-7.7)**	**11.3(4.2)**	**6.7(-8.8)**
	Iterations	15	13	15	15	15	10	5
ALOOC EXACt+ DAFE	Initial	67.2(4.1)	66.9(11.7)	80.8(17.6)	61.7(11.5)	49.8(15.2)	55.9(10.0)	97.4(2.6)
	Final	74.1(4.7)	62.5(10.9)	97.1(3.3)	73.6(0.4)	59.0(16.5)	63.4(11.5)	100(1.6)
	Difference	**6.9(0.6)**	**-4.4(-0.9)**	**16.3(-14.2)**	**12.0(-11.1)**	**9.2(1.4)**	**7.5(1.5)**	**2.6(-1.6)**
	Iterations	12	9	9	9	12	13	4
ABLOOC2 + DAFE	Initial	53.0(9.1)	78.4(17.7)	41.4(24.5)	30.9(25.6)	36.8(17.7)	4.9(20.8)	0.4(13.9)
	Final	70.9(7.5)	73.3(9.6)	93.1(4.1)	49.4(34.1)	60.1(21.3)		

	Difference	**17.9(-1.6)**	**-5.1(-8.1)**	**51.7(-20.4)**	**18.5(8.5)**	**23.3(3.7)**	**9(-18.0)**
							44.9(20.8)
								6(-13.9)
								90.4(13.9)

	Iterations	15	15	15	15	15	52.8(2.8)	100(0)
ABLOOC EXACT2 +DAFE	Initial	64.6(4.6)	57.6(16.6)	83.5(21.O)	53.9(17.8)	50.7(11.6)	**7.9(-18.0)**	**9.6(-13.9)**
	Final	72.6(6.1)	57.9(10.9)	96.6(5.5)	73.8(2.9)	57.8(17.6)	15	4
	Difference	**8.0(1.6)**	**0.3(-5.8)**	**13.1(-15.6)**	**20.0(-15.0)**	**7.1(6.0)**	**6.8(-5.5)**	**2.0(-1.7)**
	Iterations	15	15	15	15	15	15	15

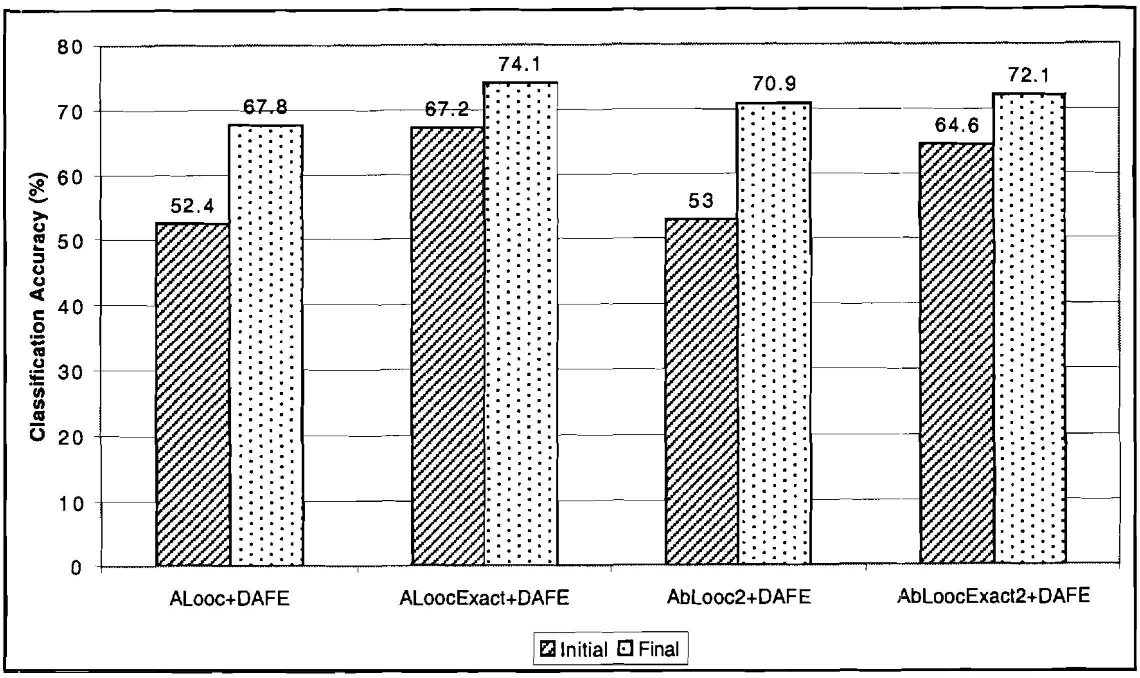


Fig. 4.7. Overall Mean Accuracy for Experiment 7

4.7 Conclusions

A new family of adaptive covariance estimators are presented which are produced by combining an adaptive classification process with various regularized covariance estimators, i.e., LOOC, bLOOC1 and bLOOC2. They are proposed as a means to mitigate small training sample problems, in particular, for the poorly or ill-posed problem where for high dimension data the number of training samples is comparable to the number of features or where the sum of all training samples is even smaller than the number of features. A set of experiments on simulated data and real hyperspectral data are performed and reported.

For simulated data, the proposed adaptive covariance estimators offer similar performance, i.e., starting with various initial classification accuracies, all of them led to higher final classification accuracy. They also appear more robust against variations in training sets as indicated by the decreased standard deviation among the repeated test trials. In addition, the increment of mean classification accuracy increases with dimensionality.

For real data, all proposed adaptive covariance estimators are able to improve the classification accuracy significantly. However, performance of the adaptive covariance estimators depends on the specific data and the initial classification accuracy. Higher initial classification accuracy tends to lead to higher final classification accuracy. However, the net increment of classification accuracy is higher with the lower initial ones.

In conclusion, the proposed adaptive covariance estimators have the advantage of both an adaptive classifier and a regularized covariance estimator and are able to produce higher classification accuracy than either of them used alone. This method is also robust because, from all experiments performed where training samples are randomly selected, the mean classification accuracy has been improved and for most of them the standard deviation of multiple trials has been reduced.

The capability of improving the classification accuracy of these: proposed adaptive covariance estimators also offers a robust classification procedure that can significantly reduce the user's effort in terms of the quantity and quality of training samples selected, which usually are difficult or tedious to achieve. This implies that, as long as a user can correctly select a few training samples for each class with this method, the classification, accuracy may be significantly improved to a value that could only have been achieved previously with large number of training samples using a common ML classifier. These characteristics suggest that the procedures tend to reduce the dependence on the skill level of the analyst.

Regarding the computation expense of these adaptive covariance estimators, at first glance, they appear computationally somewhat costly, because at each iteration, all semi-labeled samples and training samples must be checked to find the optimal regularized parameters. If there are a number of semi-labeled samples, the computation could be immense. However, in the practical application, the computation can be greatly reduced and becomes affordable for several reasons without much compromise in the classification accuracy. First of all, as was mentioned before, the determination of the optimal regularized parameter can be efficiently implemented using the rank-one down-date of the covariance matrix. Secondly, as shown in experiments the approximation of the adaptive covariance estimators, i.e., ALOOC, AbLOOC1, and AbLOOC2, produce comparable performance in most cases. Thirdly, from our experience, the major increment of classification accuracy occurs at the first a few iterations. As a matter of fact, almost 50%-60% increment occurs at the second iteration when the semi-labeled samples are used at the first time, and additional 20-10% increment occurs at the third iteration. For this reason, if computational efficiency is a major concern, one only needs to perform the first few iterations to obtain the majority increment of classification accuracy. The computation time for the hyperspectral data reported in this paper is about 45 CPU seconds for a Macintosh G4, which is affordable for practical applications.

CHAPTER 5: ADAPTIVE BAYESIAN CONTEXTUAL CLASSIFICATION BASED ON MARKOV RANDOM FIELDS

5.1 Introduction

Hyperspectral image data acquired by new generation sensors contain extremely rich spectral attributes, which offer the potential to discriminate more detailed classes with the high classification accuracy using a conventional Maximum Likelihood Pixel Classifier (MLC). However, two difficulties inhibit this potential. First of all, the large number of classes of interest combined with the large number of spectral bands available requires a large number of training samples. Unfortunately training samples are generally expensive and tedious to obtain. As a result, the class statistics estimated from the limited training sample set are less accurate and the subsequent classifier performance deteriorates. Additionally, in a conventional MLC, it is explicitly assumed that the spectral properties are independent of the properties of all other pixels. Consequently, the MLC has difficulty distinguishing the pixels that come from different land-cover classes but have very similar spectral properties. The result is usually a snow-like classification map.

Since, in general, certain ground cover class may be more likely to be placed adjacently than others, there is more than trivial information available from the relative assignments of the classes of neighboring pixels. Also, in many remotely sensed images, objects on the ground are much greater than the pixel element size so neighboring pixels are more likely to come from the same class and form a homogeneous region. Therefore, a classifier that utilizes both spectral and spatial contextual information may be able to better discriminate the pixels with similar spectral attributes but located in different regions, and subsequently reduce the speckle error and improve the classification performance significantly. However, this type of classifier also faces the problem of the srnall training sample size.

In chapter 3, it has been demonstrated that a proposed adaptive pixel MLC may alleviate the small training sample problem by including semi-labeled samples along with the training samples during the process of statistics estimation. The key to successful performance of this classifier is to establish a positive feedback process wherein during each iteration the statistics estimation can be improved based on the higher classification accuracy of the previous iteration. In return, much higher classification accuracy can be achieved in the current iteration, and so on. As with a conventional MLC, performance of this adaptive pixel MLC is limited by using just spectral information.

In this chapter, an adaptive Bayesian contextual classifier. that utilizes both spectral and spatial interpixel dependency contexts in statistics estimation and classification is proposed. In this classifier, only interpixel class dependency context is considered, and the joint prior probabilities of the classes of each pixel and its spatial neighbors are modeled by the Markov Random Field. The statistics estimation and classification are performed in a recursive manner to allow the establishment of the positive feedback process in a computationally efficient manner. Experiments with real hyperspectral data show that starting with a small training sample set this classifier can reach classification accuracies similar to that obtained by a pixel wise MLC with a very large training sample set. Additionally, classification maps are produced which have significantly less speckle error.

5.2 Previous Work and Background

There are generally two main types of contextual information [19]. i.e., interpixel class dependency context and interpixel correlation context. Both of these exist spatially and temporally. Spatial correlation coefficients between pixels generally differ according to the distance between pixels and the spectral bands. The exploitation of this spatial correlation context can make it possible to differentiate classes in more detail. This would not be possible without additional spatial correlation contextual information. However this requires paying the price of increased computational complexity as compared to pixel wise classification [20]-[22]. In this study, only interpixel class dependency context is considered.

Generally speaking, the methodologies for taking spatial context into account can be categorized into four different groups [19]. The postprocessing type contextual classifiers perform postprocessing such as filtering or applying syntactic rules after the pixelwise classification. An example of a filter for postprocessing is a majority filter [23], which counts the votes of classification results inside a given window. A common problem of this approach is that its performance relies heavily on the initial classification accuracy achieved by the pixel wise MLC. That is, the postprocessing proceedure can lead to degraded performance if the initial classification accuracy is poor. What is more, this method tends to bias a pixel into a class to which its neighbors belong. Sometimes this biasing can be overdone and as a result the segmented image may loose details unnecessarily.

The approaches in the second category are based on a region growing process. A given scene is divided into distinct homogeneous regions by using an appropriately chosen criterion and each homogeneous region is classified on a sample or per-field basis. One procedure in this category is ECHO, which uses a conjunctive, object-seeking method as the tool for region finding [24, 25]. ECHO is able to capture the homogenous behavior of regions with different sizes and utilize it to reduce speckle error. This capability depends heavily on the true size of each homogenous region. ECHO is particularly successful in applications where statistics of pixels in an image have long distance dependence. That is, neighboring pixels are more likely to come from the same class and form a large homogeneous parcel. An example of this arises with remote sensing of agriculture fields that have large regions of identical crops. However, since class statistics have been estimated to form appropriate criterion, ECHO also suffers the limited training sample problem in the analysis of hyperspectral data.

The third type of approach is the so-called stacked vector approach. This adds to the original spectral feature vector new components of features that can carry spatial contexts. Additional components can be derived, for example, from texture descriptors such as Fourier coefficients or coocurrent matrices [26]. This approach has an inherent problem of excessive dimensionality of augmented feature vectors and poor performance at the object boundaries since the texture measures are based on a certain size of region.

The final category is a model-based approach that tries to incorporate contextual information through modeling of the scene. Example models are the spatial stochastic model [20] and the two-dimensional Markovian model [27]. These approaches assume a local dependency of a pixel on its neighbors and it is incorporated into the decision rule in addition to the spectral information. As a result, these are also referred to as simultaneous contextual classification methods [20, 27-30], or Bayesian contextual classification because the theoretical foundation of simultaneous classification is based on the Bayesian formulation. Bayesian contextual approaches involve the formulation of a distribution model for both the underlying class labels and the class-conditional model so that the estimated class labels can be derived from optimizing a posterior cost function. In other words, in contextual classification, the image is classified by finding a Maximum A Posterior (MAP) estimate of the unknown field of class labels.

In the study of contextual classification, the prior probability mass function for the underlying entities (class labels) is modeled as a discrete Markov Random Field (MRF) or equivalently Gibbs distribution according to the Hammersley-Clifford theorem [29]. These models are very popular in image segmentation and restoration because they only require the specification of spatially local interaction (short distance statistical dependence) using a set of local parameters. This greatly reduces the complexity of the model. It has been shown that classification performance of multispectral remotely sensed images has been improved with these approaches [30]-[35].

Although the Bayesian contextual MAP estimation is neatly formulated, the MAP estimation still involves huge computational complexity due to the: size of the image lattice wherein the image is confined. Also, the exact maximization of the posterior probability is intractable. As a result, methods for approximately maximizing the true MAP estimate must be used. In [36] a simulated annealing has been used and it has been shown that the method will converge to the global optimum, but it is generally too slow to be practically useful. An alternative approach called Iterated Conditional Modes (ICM), which is rather crude compared with simulated annealing but computationally efficient, was developed in [37]. This method is known to yield relatively good results when textures can be discriminant over small regions containing few pixels, but in high resolution images, where larger numbers of pixels are necessary to discriminant, the method is prone to being trapped in a local minimum. In [30] an algorithm is suggested which successively classifies the image from coarse resolution to finer resolution until individual pixels are classified. This method is known to be faster than ICM [37] when distinct textures exist, and is less likely to be trapped in local minima. In [38] an approach is developed which replaces the MRF model with a novel MultiScale Random Field (MSRF), and replaces the MAP estimator with a sequential MAP (SMAP) estimator derived from a novel estimation criteria. This method is not iterative and computational efficient, and has better performance than MAP' estimation using simulated annealing.

In the analysis of hyperspectral data (up to a few hundred spectral bands), supervised MAP also face the challenge of precisely estimating the class conditional statistics with limited training sample size. In chapter 3, it is demonstrated that a proposed adaptive pixel MLC is able to alleviate the small training sample problem by including semi-labeled samples in the process of statistics estimation in addition to training samples. The key to successful performance of this classifier is to establish a positive feedback process where, at each iteration, the statistics estimation can be improved based on the higher classification accuracy of the previous iteration. This allows much higher classification accuracy to be achieved during the current iteration and those that follow. With a few iterations, eventually more accurate statistics and higher classification accuracy can be achieved. Higher classification accuracy makes establishment of the positive feedback more likely and results in faster convergence. However, like a conventional MLC, performance of this adaptive pixel MLC is limited by just using spectral information alone. Therefore, it would be advantageous to integrate a MAP classifier with the adaptive classification procedure in that performance of MAP can be enhanced because of the better class statistics provided by the adaptive method. In return, performance of the adaptive method can be further improved by the better classification accuracy produced by MAP where the spatial information is exploited in addition to spectral information. In other words, a combination of the MAP cllassifier with an adaptive procedure should outperform a pixel-wise adaptive MLC. From now on, we will refer to this method as Adaptive Bayesian Contextual Classification Based on Markov Random Field (ABCC-MRF).

Although there are methods [30] [37] which perform better than ICM [38] in the application of image segmentation, the ICM method is selected in ABCC-MRF to approximately maximize the MAP estimate of the unknown field of class labels for three reasons. First of all, the ICM is an iterative process and it starts with a pixel-wise ML process. Therefore, it is easier to integrate with an adaptive pixel-wise MLC. Secondly, and most important of all, ICM has demonstrated adequate performance in the application of multispectral data analysis [28] [33] [35]. The reason is as follows. In the classification process of the ICM [29] where the class label is assigned to each pixel, maximizing the joint posterior probability is approximated as maximiizing the individual class posterior probability. Since multispectral data contains more spectral attributes than spatial ones, the spectral information plays the major rule in classification. In contrast, spatial information is subsidiary, and it is only used to enhance. the classification performance. With hyperspectral data, which has many more spectral bands than multispectral data, if the class statistics can be more accurately estimated, the rich spectral information contained in data can be better utilized. Consequently, higher classification performance can be achieved. Better class statistics estimates may be achieved by an adaptive method. In other words, in the analysis of hyperspectral data, high classification performance doesn't require one to estimate MAP more precisely using more elaborate methods [36] [37]. An important advantage of ICM is that it is conceptually simple and computationally efficient. As a result, ABCC-MRF also has the advantage of computational efficiency. This is a highly desirable feature in the analysis of the hyperspectral data. In the next section, the Bayesian formulation and ICM are presented.

5.2.1 Bayesian formulation of image in markov random field

Multivariate image X is composed of p-dimensional pixels where $X_k(s)$, and {k=1, 2, ..., p}, and s=(i,j) denotes a two-dimensional index, an image lattice point at the i^{th} row and j^{th} column. Let u denote the field that contains the classification of each pixel in X. Points in u can take values in the set {1, 2, ..., L}, where L is the number of classes. The multivariate image X is then classified by finding a field of class labels $\hat{u}_{MAP}$ such that

$$\hat{u}_{MAP} = \arg\max_{u}\{p(u \mid X)\} = \arg\max_{u}\{p(X \mid u)p(u)\} \tag{5.1}$$

where $\hat{u}_{MAP}$ is referred to as a MAP estimate of the field of class labels which maximizes the posterior cost function (5.1). Therefore, the modeling of both the prior probability distribution p(u) and class-conditional distribution p(X | u) becomes an essential task. Note that the estimate Eq.(5.1) becomes the pixel-wise noncontextual classifier if the prior probability does not have any consequence in formulating Eq.(5.1).

In most vision problems, available information stems from two different sources: observation on image sites for a given occurrence of the problem, and a priori knowledge about the restrictions imposed on the simultaneous labeling of connected neighboring units. This second source of information reflects statistical dependencies between the labels of neighboring sites. Markov random field (MRF) theory [29] [36] [38] [39] [40] provides a convenient and consistent way to model such context-dependent information. The MRF's-Gibbs equivalence, established by Hammersley and Clifford, and further developed by Besag [29], gives an explicit formula for the joint distribution of MRF's.

For a Markov random field u, the conditional distribution of a point in the field, given all other points, is only dependent on its neighbors: $p\{u(s)\,|\,u(S-\mathrm{s})\} = p\{u(s)\,|\,u(\partial s)\}$. Here S is an image lattice and S-s denotes a set of points in S excluding s, ∂s denotes the neighboring pixels of s. The first order neighborhood system is usually defined as the four pixels surrounding a given pixel, and higher orders are defined by adding comer pixels to a lower order neighborhood system. A clique is defined as a subset of points in S such that if s and r are two points contained in a clique c, then s and r are neighbors, and the order of a clique is the number of points (sites) in the clique. The neighborhood system and the corresponding cliques are illustrated in Figure 5.1.

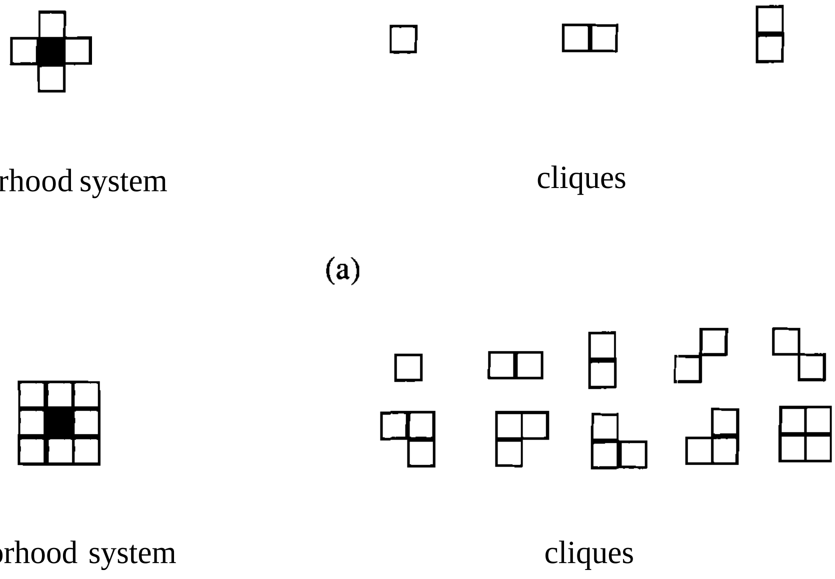
5.2.2 Prior model

The a priori probability of the labeling p(u) defines an MRF. According to the Hammersley-Clifford theory [29], for a given neighbor system, p(u) can be expressed as a Gibbs distribution:

$$p(u) = \frac{1}{Z}\exp[-\sum_{c} V^{c}(u)] \tag{5.2}$$

where Z is a normalizing constant called a partition coefficient, and V^c is an arbitrary function of u on the clique c. C is defined as the set of all cliques.

Together with the joint class-conditional distribution p{X|u} and prior distribution of (5.2), the MAP estimates of true class labels as given by (5.1) becornes:



4-neighborhood system

cliques

(a)

8-neighborhood system

cliques

(b)

Fig. 5.1. Neighborhoods system and corresponding cliques

$$\hat{u}_{MAP} = \arg\min_{u}\{-\ln p(X \mid \mathrm{u}) + \sum_{c} V^{c}(\mathrm{u}))\qquad (5.3)$$

The minimization of (5.3) is essential in order to derive a MAP estimate of u, $\hat{u}_{MAP}$. In [30], it is pointed out that the one dimensional dynamic programming in [31] or simulated annealing method in [36] are computational expensive, and the global minimization still suffers from falling into a local minimum. In [38], a method called ICM is developed to approximate $\hat{u}_{MAP}$ using assumptions to reduce: the computational complexity. Instead of attempting to optimize in one step by the above suggested methods, the ICM is computationally feasible since it updates the class assignments iteratively so that inverting a huge matrix is avoided. To apply the ICM method, (5.1) is modified to conform to the task based on two main assumptions, which are:

(1) Each pixel value is class-conditionally independent, such that:

$$p(X \mid u) = \prod_{s=1}^{N} p\{X(s) \mid u(s)\}$$

(2) The class labels are the realization of a Markov random field, and their probability mass functions are identical, i.e.,

$$p(u(s) \mid u(S-s)) = p\{u(s) \mid \hat{u}(\partial s)\}$$

Suppose that the objective is to estimate the class label of' a pixel given the estimates of class labels for all other pixels inside the rectangular lattice. Then the optimization of (5.3) becomes:

$$\hat{u}(s) = \arg\max_{u}\{p(u(s) \mid X, \hat{u}(S-s)\} \tag{5.4}$$

Note that u(s) denotes a class label at s ∈ S. Applying the Bayes' rule and considering the Markov property of (2), the argument of (5.4) becomes

$$p(u(s) \mid X, \hat{u}(S-s) \propto p\{X(u) \mid u(s), \hat{u}(S-s)\} p_s\{u(s) \mid \hat{u}(\partial s)\} \tag{5.5}$$

The first term of the right hand side of (5.5) becomes

$$p\{X \mid u(s), \hat{u}(S-s)\} = p\{X(u) \mid u(s)\} p\{X(S-s) \mid \hat{u}(S-s)\} \tag{5.6}$$

by virtue of (1). Since the class assignment of all other pixels except u(s) inside the lattice are already made, the term $p\{X(S-s) \mid \hat{u}(S-s)\}$ is not a factor affecting the optimization. Therefore, (5.4) in connection with (5.5) and (5.6) becomes

$$
\begin{aligned}
\hat{u}(s) &= \underset{u(s)}{\arg\max}\{p(u(s)\mid X,\hat{u}(S-s)\} \\
&= \underset{u(s)}{\arg\max}[p\{X\mid u(s),\hat{u}(S-s)\}p_{S}\{u(s)\mid \hat{u}(S-s)\}] \\
&= \underset{u(s)}{\arg\max}[p\{X(s)\mid u(s)\}p\{u(s)\mid \hat{u}(\partial s)\}]
\end{aligned}
\tag{5.7}
$$

Assuming the class conditional distribution can be represented by Gaussian distribution, i.e.,

$$
p\{X(s)\mid u(s)\} = \frac{1}{(2\pi)^{\frac{p}{2}}|\Sigma_u|^{\frac{1}{2}}}\exp[-\frac{1}{2}\{(X(s)-\mu_u)^T\Sigma_u^{-1}(X(s)-\mu_u)\}] \tag{5.8}
$$

Concerning energies of cliques of order 2 (2-point clique) and restricting to 4-neighborhood system, for the sake of mathematical and computational convenience, most MRF vision models are assumed to be homogeneous and isotropic. Then V^c is independent of the location of clique c in S and independent of the orientation of c. Under these assumptions, the M-Level MRF model is frequently used for an image segmentation problem:

$$
V(u(s),u(s')) = \begin{cases} 0 & \text{if } \mathrm{u(s)=u(s')} \\ \beta & \text{otherwise} \end{cases}
$$

where β is a constant coefficient, which can be estimated from the image or empirically determined. It is a weight emphasizing the significance of interaction among adjacent pixels inside a clique. Therefore, the class conditional probability mass function of $p\{u(s)\mid \hat{u}(\partial s)\}$ becomes

$$
p\{u(s)\mid \hat{u}(\partial s)\} = \frac{1}{z}\exp[-\beta\sum_{s'\in c(s)}\{1-\delta(u(s)-u(s'))\}]
$$

Then (5.7) is equivalent to:

$$\begin{aligned}\hat{u}(s) &= \arg\max_{1\le u\le L}[p\{X(s)\,|\,u(s)\}p\{u(s)\,|\,\hat{u}(\partial s)\}[\\ &= \arg\min_{1\le u\le \mathrm{L}}[-\mathrm{In}\,p\{X(s)\,|\,u(s)\} - \ln p\{u(s)\,|\,\hat{u}(\partial s)\}] \qquad (5.9)\\ &= \arg\min_{1\le u\le L}[\ln|\Sigma_s| + (X(s)-\mu_u)^T\Sigma_u^{-1}(X(s)-\mu_u) + 2m\beta + const.]\end{aligned}$$

Here, m is the number of occurrences of the class different from u(s) in the clique containing s. The term const. doesn't depend on the particular class assignment to the pixels.

5.3 Adaptive Bayesian Contextual classifier: the Combination of an Adaptive Classifier with Bayesian Contextual Iteration Conditional Modes (ICM)

In this section, the new adaptive Bayesian contextual classifier is developed that combines the adaptive procedure proposed in chapter **3** with the Bayesian Contextual Iteration Conditional Modes (ICM) [38]. In this new classifier, contextual information is incorporated into the process of weighting factor computation and classification. There are two reasons for this operation. One is to further emphasize the positive effect from the correctly classified semi-labeled samples and discourage the negative influence from the mis-classified semi-labeled ones, and the second is to enhance the classification using contextual information in addition to the likelihood.

Similar to the adaptive procedure and ICM, this new method is also an iterative process that achieves the optimal statistics estimation and classification by starting with initial estimate ϕ^0 and the classification based on training samples only and repeating the following steps at each iteration using training samples and semi-labeled samples. Assume the initial class conditional statistics and classification ha!; been obtained by using the training samples, and all L classes can be represented by Gaussian distributions. Denote $y = (y_{i1},\ldots,y_{im_i})$ as the training samples for the i^{th} class, whose pdf is $f_i(x|\phi_i)$, and $x = (x_{i1},\ldots,x_{in_i})$ are the semi-labeled samples that have been classified to the i^{th} class. The procedure of this method is defined as follows the initial ML classification using training samples:

Cycle 1:

la) Perform classification using a MAP classifier based on the classification map from the ML:

$$x(s) \in u \Leftrightarrow \hat{u}(s) = \arg\max_{u(s)}[p\{X(s) \mid u(s)\} p\{u(s) \mid \hat{u}(\partial s)\}]$$

or

$$X(s) \in \mathrm{s} \Leftrightarrow \hat{u}(s) = \arg\min_{1 \leq u \leq L}[\ln | \mathrm{C}, | + (X(s) - \mu_u)^T \Sigma_u^{-1}(X(s) - \mu_u) + 2m\beta] \quad (5.10)$$

where β is empirically determined

lb) Perform classification using a postprocessing classifier based on the classification map from the ML

$$x(s) \in u \Leftrightarrow \hat{u}(s) = \arg\max_{u(s)}[p\{u(s) \mid \hat{u}(\partial s)\}] = \arg\min_{1 \leq u(s) \leq L}[2m\beta] \quad (5.11)$$

The purpose of using the postprocessing classifier is to compare the results from the MAP classifier

Cycle ***2:***

1) Compute weighting factors using contextual information together with the likelihood based on the classification results from MAP classifier in step (la) from the previous cycle

$$w_{uj}^c = \frac{p(x_{uj} \mid \phi_u^c) p(u(s) \mid \hat{u}(\partial s))}{\sum_{k=1}^{L} p(x_{uj} \mid \phi_k^c) p(k(s) \mid \hat{k}(\partial s))} \quad (5.12)$$

Note that the unit weight is assigned to each training sample.

2) Obtain the class conditional statistics by maximizing the mixed log likelihood based on the classification results from the MAP classifier in step (1a) from the previous cycle:

$$\phi_i^+ = \arg\max_{\phi_i \in \Omega}\Big(\sum_{k=1}^{m_i} \ln(f_i(y_k \mid \phi_i)) + \sum_{k=1}^{n_i} w_{ik}^c \ln(f_i(x_{ik} \mid \phi_i))\Big)$$

or

$$\begin{aligned} \mu_i^+ &= \frac{\sum_{j=1}^{m_i} y_{ij} + \sum_{j=1}^{n_i} w_{ij}^c x_{ij}}{m_i + \sum_{j=1}^{n_i} w_{ij}^c} \\ &= \frac{m_i}{m_i + \sum_{j=1}^{n_i} w_{ij}^c} \frac{\sum_{j=1}^{m_i} y_{ij}}{m_i} + \frac{\sum_{j=1}^{n_i} w_{ij}^c}{m_i + \sum_{j=1}^{n_i} w_{ij}^c} \frac{\sum_{j=1}^{n_i} w_{ij}^c x_{ij}}{\sum_{j=1}^{n_i} w_{ij}^c} \end{aligned} \qquad (5.13a)$$

$$\begin{aligned} \Sigma_i^+ &= \frac{\sum_{j=1}^{m_i} (y_{ij} - \mu_i^+)(y_{ij} - \mu_i^+)^T + \sum_{j=1}^{n_i} w_{ij}^c (x_{ij} - \mu_i^+)(x_{ij} - \mu_i^+)^T}{m_i + \sum_{j=1}^{n_i} w_{ij}^c} \\ &= \frac{m_i}{m_i + \sum_{j=1}^{n_i} w_{ij}^c} \frac{\sum_{j=1}^{m_i} (y_{ij} - \mu_i^+)(y_{ij} - \mu_i^+)^T}{m_i} \qquad (5.13b) \\ &\quad + \frac{\sum_{j=1}^{n_i} w_{ij}^c}{m_i + \sum_{j=1}^{n_i} w_{ij}^c} \frac{\sum_{j=1}^{n_i} w_{ij}^c (x_{ij} - \mu_i^+)(x_{ij} - \mu_i^+)^T}{\sum_{j=1}^{n_i} w_{ij}^c} \end{aligned}$$

3) Performing classification based on the maximum likelihood (ML) classification rule:

$$x \in i \Leftrightarrow i = (x - \mu_i^+)(\Sigma_i^+)^{-1}(x - \mu_i^+)^T + \ln|\Sigma_i^+| \qquad (5.14)$$

4a) Perform classification using the MAP classifier based on the classification map from the ML classifier:

$$x(s) \in u \Leftrightarrow \hat{u}(s) = \arg\min_{1 \le u \le L}[\ln|\Sigma_u^+| + (X(s) - \mu_u^+)^T(\Sigma_u^+)^{-1}(X(s) - \mu_u^+) + 2m\beta] \qquad (5.15)$$

4b) Perform classification using the postprocessing classifier based on the classification map from the ML

$$x(s) \in u \Leftrightarrow \hat{u}(s) = \arg\max_{u(s)}[p\{u(s) \mid \hat{u}(\partial s)\}] = \arg\min_{1 \le u(s) \le L}[2m\beta]$$

The steps of the cycle 2 are repeated until convergence is reached where the classification results have small change. The flow chart in figure 5.2 illustrates one complete cycle of the adaptive contextual classifier.

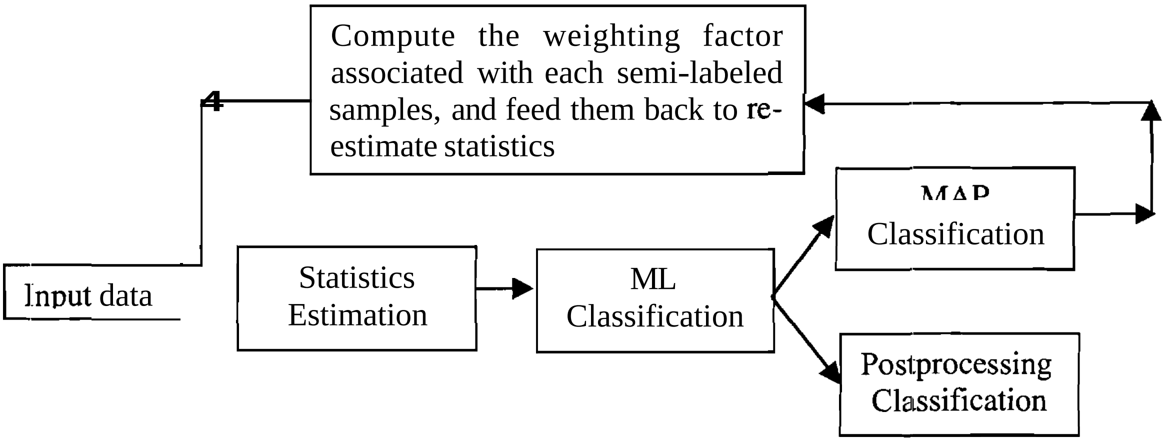

Compute the weighting factor associated with each semi-labeled samples, and feed them back to re-estimate statistics
Input data
Statistics Estimation
ML Classification
MAP Classification
Postprocessing Classification

Fig. 5.2. One complete cycle of the adaptive contextual classifier

Note that as an adaptive pixel-wise ML classifier, in this adaptive contextual classifier, the label of each semi-labeled sample is updated after each classification, including NIL, MAP, and postprocessing classification at each cycle, and the weight of each semi-labeled sample is updated after each cycle. Correspondingly, the class conditional statistics are updated at each cycle too.

However, two modifications have been made in this new proposed classifier. First, the contextual information in addition to likelihood is utilized to enhance the performance of semi-labels in terms of their influence of class conditional statistics estimation and to improve the classifier performance. Second, the semi-labeled samples generated from the MAP classifier instead of a ML classifier at the previous cycle in addition to training samples are used to update the current class conditional statistics, and each cycle is started with a ML classifier instead of a MAP classifier. The reason for this choice is as follows. First, it has been shown that in ICM starting with the classification results from a ML classifier, in general the MAP classifier outperforms the ML classifier [33] [34]. Even though a postprocessing process may be able to improve classification accuracy also by reducing the speckle error, it is more likely to be overdone and lead to loss of details. Therefore, semi-labeled samples generated from the MAP classifier should contain more correctly classified samples, and better statistics estimation may result than to use those from a ML classifier or a postprocessing process. Second, with good statistics estimation, a ML classifier may be able to catch more details, and it is less likely to bias the minority class with small numbers of pixels than a MAP classifier or a postprocessing process. Since the ultimate objective here is to generate a classification map with high quality, i.e., high classification accuracy with less speckle but with adequate detail, a NIL classifier is chosen to start each cycle to produce the classification results with as much detail as possible. After that a MAP, or a postprocessing process is used to further improve classification accuracy by removing the speckle error that usually can be corrected by using contextual information, for instance, spatial proximity. In the following section, the experiment with the proposed algorithm is applied to analysis of hyperspectral data and the results are presented.

5.4 Experimental Results and Discussion

In this experiment, the analyzed data is part of an airborne hyperspectral data flighline over the Washington DC mall, which was collected with the HYDICE system.

In this case there were **210** bands in the **0.4** to **2.4** μm region of the visible and infrared spectrum. In the analysis, the water absorption bands are removed and **191** bands are used. There are **11** classes with about **50** training samples for each class selected. Since the data has high spatial resolution (about **5** meters), the testing samples are manually selected. The detailed information about training and testing samples are shown in Table **5.1** and thematic maps of the training and testing fields are illustrated.in Figure 5.3a and Figure 5.3b, respectively. Note that there is no overlap between training fields and testing fields. The training samples size is deliberately chosen to be small. With Discriminant Analysis Feature Extraction (DAFE) [1], **10** out of **191** features are selected to form a new subspace, and then the classification is performed in this new subspace.

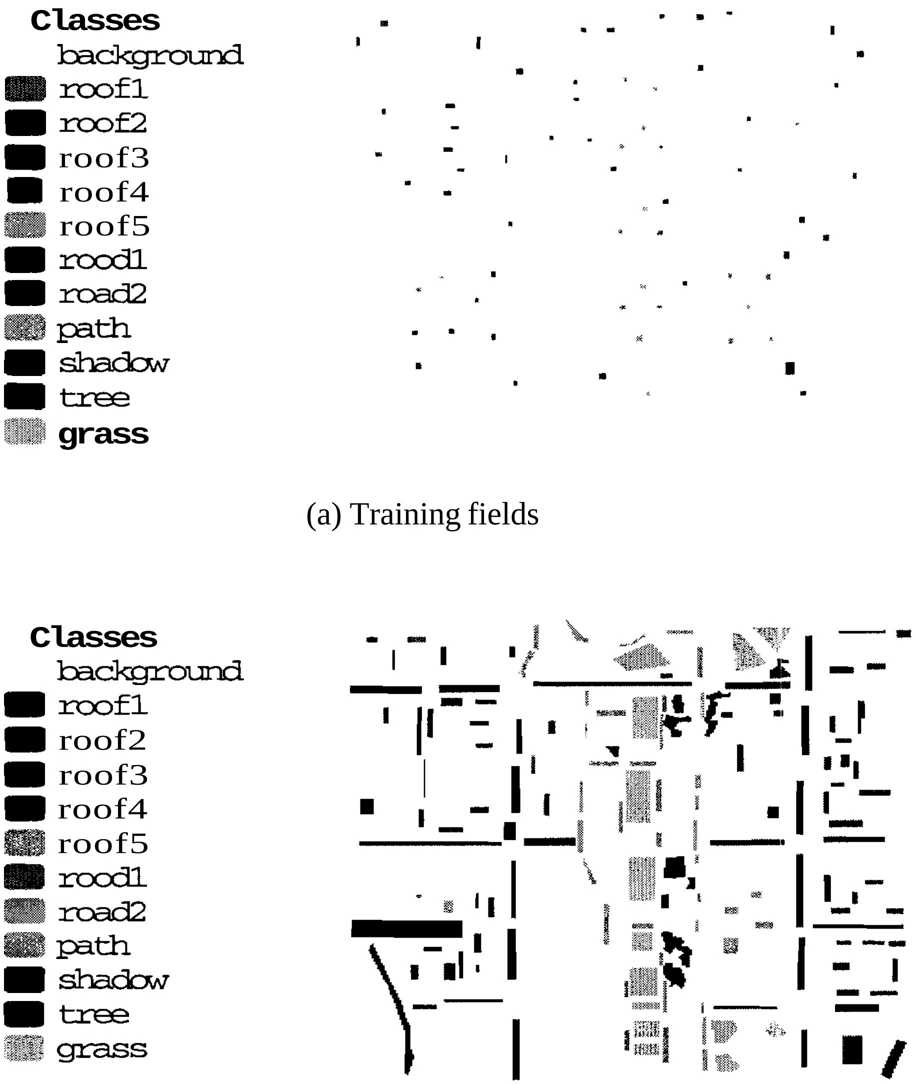
Table **5.1**

Training and Testing Samples

Class	Training Samples	Testing Samples
roof1	**54**	**121**
roof2	**54**	**1433**
roof3	**58**	**348**
roof4	**46**	**290**
roof5	**55**	**243**
rood1	**61**	**4737**
road2	**52**	**855**
path	**56**	**552**
shadow	**46**	**215**
tree	**55**	**1109**
grass	**54**	**3193**
Total	**647**	**13096**

This data set is a challenge to analyze for several reasons. First, classes are complex. For example, the information class ***roof*** consists of five types of subclasses, and the ***road*** class consists of two types of subclasses. Usually, even though some of the subclasses are spectrally quite different, some are quite similar. Second, the classes ***roof, road*** and ***path*** are spectrally similar in that they may be made of similar materials, for instance, asphalt. Third, this data was collected at the dry season; most of lawns are not well grown and as a result, the class ***grass*** and ***path*** are hard to differentiate in the regions where the path is located on the lawn.

In table 5.2, the overall classification accuracy obtained by three types of classifiers during each cycle with various values of β is illustrated. The resubstitution accuracy [1] represents the accuracy where, for the ML classifier, all test samples are used to train and test the classifier. Usually it is biased to a value higher than the true one. The Kappa statistic of each result is listed in parentheses next to the corresponding overall classification accuracy. Figure 5.4 illustrates the classification accuracy at each cycle, and Figure 5.5 shows the variation in classification accuracy with β.



(a) Training fields

(b) Testing fields

Fig. 5.3. Thematic maps for training and testing fields. (In color)

Table 5.2

The Performance of the Adaptive Contextual Classifier

		Accuracy (%)	Belta=1	Belta=2	Belta=4	Belta=8	Belta=16	Belta=32
Cycle 1	ML	Class	80.5(76.5)	80.5(76.5)	80.5(76.5)	80.5(76.5)	80.5(76.5)	80.5(76.5)
		Group	90.1(86.6)	90.1(86.6)	90.1(86.6)	90.1(86.6)	90.1(86.6)	90.1(86.6)
	MAP	Class	82.6(78.9)	83.0(79.4)	83.0(79.4)	82.9(79.2)	82.9(79.3)	83.4(79.8)
		Group	90.5(87.2)	91.0(87.7)	91.0(87.8)	91.3(88.)	91.4(88.3)	91.8(88.8)
	Post-Processing	Class	85.0(81.6)	85.0(81.6)	85.0(81.6)	85.0(81.6)	85.0(81.6)	85.0(81.6)
		Group	91.8(88.9)	91.8(88.9)	91.8(88.9)	91.8(88.9)	91.8(88.9)	91.8(88.9)
Cycle 2	ML	Class	87.6(84.7)	87.9(85.1)	88.1(85.4)	88.4(85.7)	88.5 (85.9)	88.8(86.2)
		Group	93.0(90.4)	93.2(90.7)	93.2(90.7)	93.4(90.9)	93.3(90.9)	93.3(90.9)
	MAP	Class	89.1(86.6)	89.9(87.5)	90.4(88.1)	91.0(88.8)	91.4(89.3)	91.7(89.6)
		Group	93.9(91.6)	94.3(92.1)	94.5(92.4)	94.9(93.0)	95.1(93.2)	95.1(93.3)
	Post-Processing	Class	92.4(90.5)	92.5(90.6)	92.6(90.8)	92.7(90.8)	92.7(90.9)	92.9(91.1)
		Group	95.4(93.6)	95.3(93.6)	95.3(93.5)	95.3(93.6)	95.3(93.6)	95.3(93.5)
Cycle 3	ML	Class	88.2(85.4)	88.7(86.1)	89.3(86.7)	89.7(87.2)	90.0(87.6)	90.4(88.1)
		Group	92.9(90.3)	93.2(90.7)	93.4(90.9)	93.6(91.2)	93.6(91.3)	93.8(91.5)
	MAP	Class	89.4(86.9)	90.2(87.9)	91.2(89.0)	92.0(90.0)	92.7(90.9)	93.2(91.4)
		Group	93.8(91.4)	94.2(92.0)	94.7(92.7)	95.1(93.3)	95.4(93.7)	95.6(94.0)
	Post-Processing	Class	92.6(90.8)	92.9(91.1)	93.2(91.5)	93.7(92.1)	94.1(92.6)	94.4(93.0)
		Group	95.5(93.8)	95.6(94.0)	95.7(94.1)	95.8(94.2)	95.9(94.3)	95.9(94.4)
Resubstitution		Class	95.6(94.5)					
		Group	96.7(95.5)					

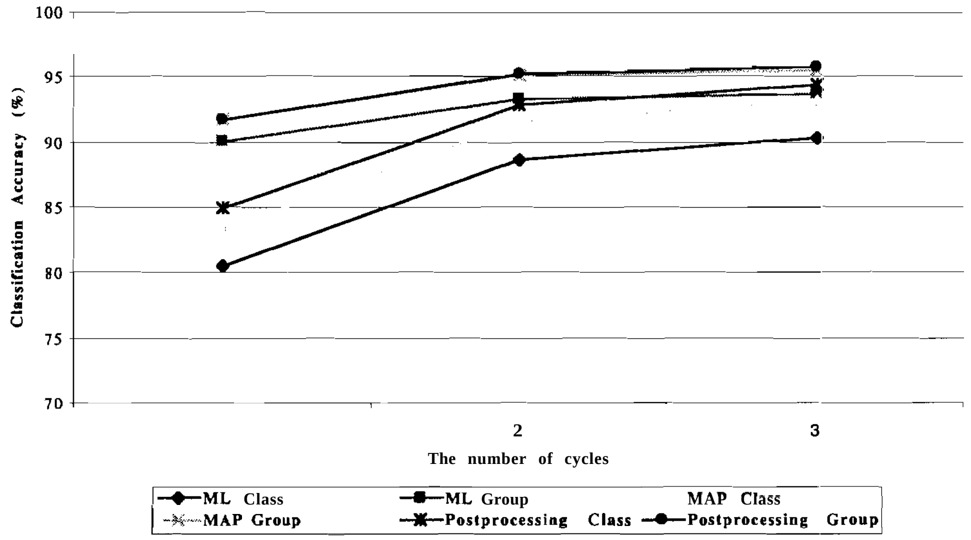


Fig. 5.4. Progression of the classification accuracy with β=32

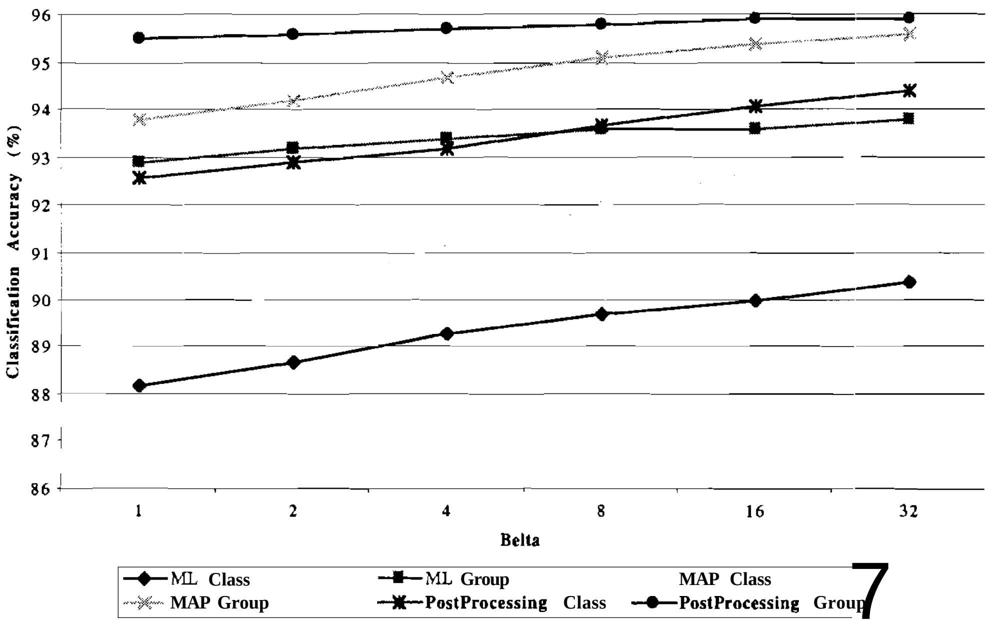
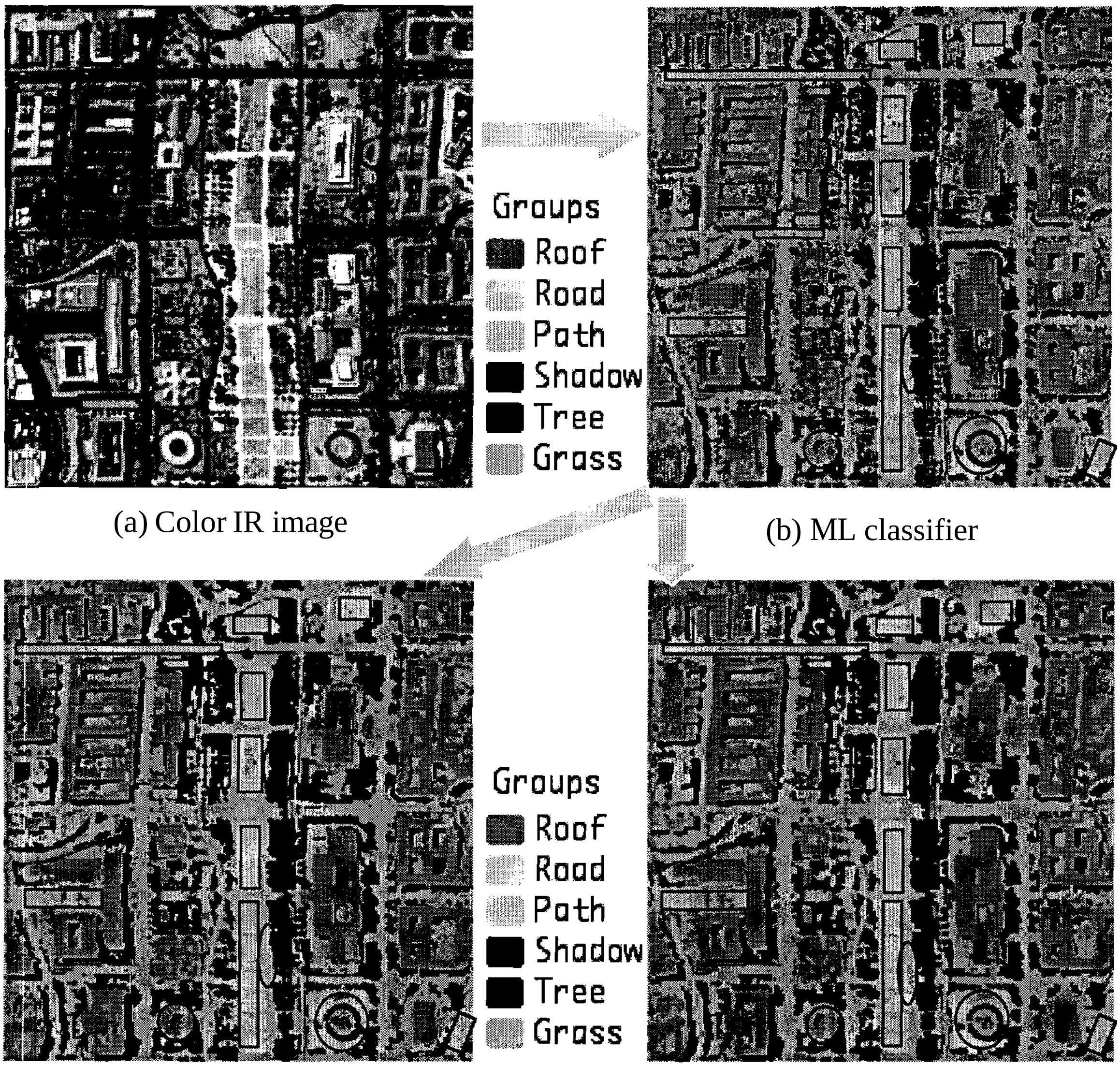


Fig. 5.5. Classification accuracy versus β in the adaptive contextual classifier at the last cycle

From the Table 5.2 and Figures 5.4 and 5.5, the following results may be observed: 1) For all classifiers, both the overall class and group classification accuracies have been improved as iteration progresses. After just three cycles the classification accuracy converges with net increment about 13% for the class, and about 6% for the group. 2) At each cycle, the MAP and the postprocessing classifier achieve the higher overall class and group classification accuracies than the ML classifier does. This indicates that contextual information does help to reduce the speckle error and accordingly improve classification performance. 3) During the first cycle the classification accuracy increment from ML to MAP is about 2.6% for the class and 0.4% for the group. However, the classification accuracy increase for the ML at the second cycle is about 7.1% for the class and 2.5% for the group. This indicates that using additional contextual information does improve the classification performance, but the irnprovement is limited. Essentially, the significant improvement of the classification performance may stem from better statistics estimates produced by the adaptive method. 4) Even though classifier performance increases as β becomes large, the improvement is not significant. This indicates that the classification result is not very sensitive to the value of β if it is large enough.

Even though there are a large number of samples (13,906) in the test fields, this is only about one seventh of the total number of samples (95,456) in the data set. Therefore, the classification accuracy for test fields can only provide an incomplete characterization this proposed adaptive Bayesian contextual classification procedure's performance. It is worthwhile to examine the thematic maps of the segmented images. Figures (5.6a) through (5.6c) depict the classification results during the first cycle with β=32.

During the first cycle, with limited training samples the initial statistics estimates are not very good. In this case, the total number of training samples is greater than the original bands and much smaller than the total number of parameters needed to be estimated in the original space. Therefore, feature extraction is not very good either, thus the classification performance. It may be seen in Figure 5.6a that classification errors occur in many places and some of them are highlighted by rectangles and ovals. These errors are mostly due to incorrectly estimated statistics and, to a lesser extent, the spectral similarity (class overlap) between classes. For instance, there is a great deal of similarity in the spectral response between roof and road or path, between *path* and grass, and between tree and grass. In Figures 5.6b and 5.6c it may be observed that the latter type of errors (speckle errors) are greatly reduced by the **MAP** or the postprocessing classifiers. However, errors of the first type (highlighted by ovals) still remain. In some areas the **MAP** or the postprocessing create additional errors (highlighted by rounded rectangles) beyond those generated by the ML. These errors lead to loss of details. This indicates that with additional contextual information the classification performance may be improved. However, this improvement is certainly limited if the initial classification accuracy is poor. The limitation is represented in the classification accuracy increment shown in Table 5.2, as has been pointed out previously.



(a) Color IR image

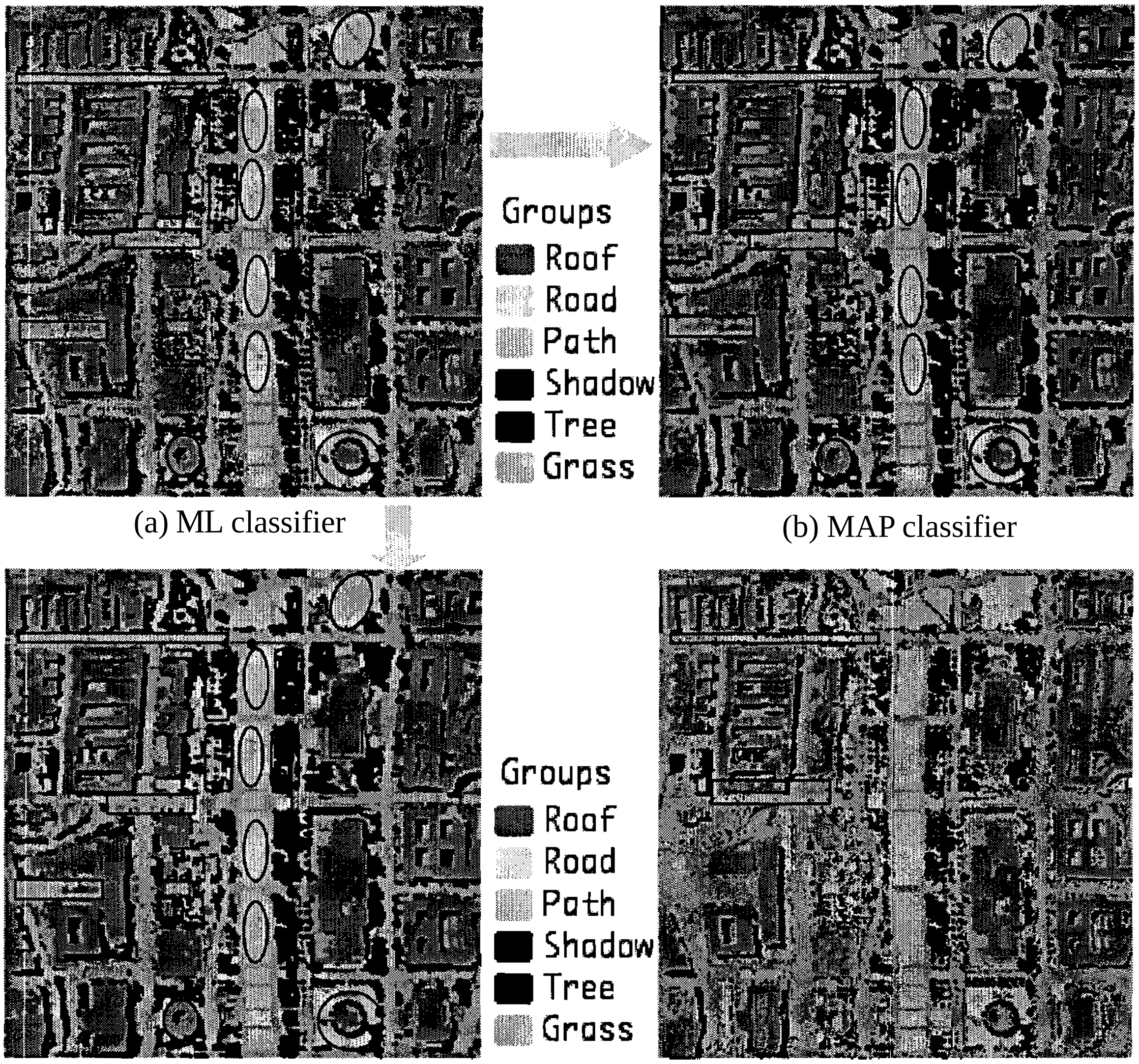
(b) ML classifier

(c) MAP classifier

(d) Postprocessing

- Regions highlighted by the rectangles: speckle errors here may be due to confusions between classes generated by the ML classifier, but most of them are corrected by the MAP and the postprocessing classifiers.
- Regions highlighted by the rounded rectangles: partial details achieved by the ML classifier, but then lost by the MAP and the postprocessing classifiers
- Regions highlighted by the ovals: classification errors here may be due to bad statistics estimation with limited training samples which occurs in the ML classifier, and could not be corrected by the MAP and the postprocessing classifiers

Fig. 5.6. The original image and the thematic maps of the segmented images from the first cycle with β=32. (In color)



(a) ML classifier

(b) MAP classifier

(c) Postprocessing

(d) ML classifier with all testing samples as training samples

- Regions highlighted by the ovals: classification errors occurring during the first cycles may be due to bad statistics estimates but have been corrected by the ML classifier at this cycle with improved statistics estimates.
- Regions highlighted by the rounded rectangles: details lost in the first cycle and then are recovered by the ML classifier during this cycle, then most of them have been maintained in the subsequent MAP and the postprocessing classifiers
- Regions highlighted by the rectangles: speckle errors remain in the results of the ML classifier at this cycle, but corrected by the MAP and the postprocessing classifier

Fig. 5.7. The thematic maps (grouped classes) of the segmented images at the third cycle with β=32 by the adaptive contextual classifier and by a MLC. (In color)

During the third cycle, which is shown in Fig. 5.7a, the classification errors have been greatly reduced and the details lost in the first cycle have been recovered by the ML classifier. This may be attributed to improved statistics estimation. However, speckle errors still remain in certain regions, for example, the regions which are highlighted by rectangles. As a result, even with good statistics, the ML could not differentiate the classes with similar spectral responses very well. On the other hand, with additional contextual information, this type of error can be removed by the MAP or the postprocessing classifiers. Results of this approach are displayed in Figures 5.7b and 5.7c which depict thematic maps that are visually clean and pleasant.

To benchmark the performance of the adaptive Bayesian contextual classification method, all testing samples are used as training samples, and then classification is performed by a ML classifier. Subsequently, the ML classifier performance is tested by the same set of testing samples. The thematic map of the segmented image is shown in Fig. 5.6d. With the large training sets, three information classes, grass, tree, and road are nicely identified. However, there are some undesired effects. There are many pixels from path, and five subclasses of roofs that are incorrectly identified as road. The possible explanation is as follows: the classes, *roof1* through *roof5*, *road1* and *road2*, might be made of similar materials and therefore pixels from these classes may contain similar spectral response. However, since the testing samples are manually selected, and roads may be readily identified by their long and relatively narrow aspect ratio it is relatively easy to select numerous pixels for the road testing samples. On the contrary, it is relatively difficult to select pixels as testing samples for the classes path and roofs, which are limited and distributed in the narrow and short blocks. As a result, there are much more testing samples for roads than for roofs and path. Therefore, the statistics estimates for roads may be more accurate than ones for roofs and path with smaller variances. Consequently, the pixels from the classes, roofs or path, might be more likely incorrectly classified as roads.

In addition, there are many speckle errors that are mostly scattered on the regions where roads are located. This further indicates the essential drawback of a ML classifier, that is, even with pretty good statistics estimates, speckle errors may be unavoidable.

Upon comparing Figures 5.7b and 5.7c with Figure 5.7d one can see that identification is improved. Specifically the classification of roofs, *path* and shadow has improved. In other words, the detailed information about these three classes lost in Fig. 5.7d has been recovered and is displayed in Figures 5.7b and 5.7c. In addition, most of speckle errors exiting in Fig. 5.7d do not appear on the Fig. 5.7b and Fig. 5.7c. Overall, the images in Fig. 5.7b and Fig. 5.7c are visually more appealing than the one on Fig. 5.7d.

Even though the classification accuracies corresponding to images in Figures 5.7b and 5.7c were achieved by the MAP and the postprocessing procedure during the third cycle, are slightly lower than the one corresponding to the Fig. 5.7d produced by the ML with all testing samples as training ones (resubstitution accuracy [1]), this doesn't mean that the proposed Adaptive Bayesian Contextual Classifier underperforms the ML classifier. Usually the value of the resubstitution accuracy [12] often biases to a higher one than the true accuracy . Therefore, it is possible that the classification accuracies achieved by the MAP and the Postprocessing classifiers may be higher than the resubstitution accuracy if a different testing data set is used. This :indicates that even starting with a limited training data set, the adaptive Bayesian Contextual Classification procedure can achieve high quality classification results. In other words, the final classification accuracy is high and the resulted segmented image is visually pleasant. Furthermore, it may even outperform a conventional one pass ML classifier with large number of training samples.

5.5 Conclusion

In this chapter, an adaptive Bayesian contextual classification procedure based on Markov Random Field is developed. In this procedure, the adaptive classification classifier and the Bayesian contextual classifier that is approxirnated by ICM are integrated. As a result, advantages of both classifiers are incorporated. As an adaptive ML classifier, the proposed procedure can iteratively enhance statistics estimates and improve classification performance with a limited training sample set. As with a contextual classifier, it can therefore improve the classification accuracy by reducing the speckle errors due to spectral similarity between classes that are very difficult to differentiate by a pixel-wise ML classifier.

The experimental results with hyperspectral data further reveals the benefits of this classification procedure. Starting with a limited training sample set, this method is able to steadily raise classification accuracy and eventually drive it close to the optimal value. The total improvement in the classification accuracy is significant and the convergence rate is fast even though a simple sub-optimal contextual classifier is used. This is significant because the classifier ICM has a reputation of slow convergence when it is used alone.

Overall, the proposed procedure is conceptually simple, easy to implement, fast to run, and has high performance. Here, the very simple and effiicient sub-optimal contextual classifier, ICM, is integrated with the simple ML classifier. The high performance is achieved because these techniques are combined in a constructive way so that their individual shortcomings can be reduced and their advantages can be amplified. It is specifically advantageous when the pixels have strong local (short distance) statistics independence.

As with the adaptive ML classifier developed in Chapter 3, and the adaptive covariance estimator developed in Chapter 4, the adaptive Bayesian contextual classification procedure provides a means to mitigate the limitations imposed by Hughes phenomenon. In addition, it offers a robust classification procedure that can significantly reduce the analyst's effort in terms of the quantity and quality of training samples selected. This is important because training samples are generally difficult or tedious to o'btain. Also, this means the dependence on the skill level of the analyst may be greatly reduced.

CHAPTER 6: CONCLUSION

6.1 Summary

In a typical supervised classification procedure, training samples play a fundamental rule on performance of a classifier. When the number of training samples is finite, the classification accuracy first increases then decreases with dimensionality. This is often referred to as the Hughes phenomenon, or the peaking phenomenon. The degradation of classification performance with dimensionality is particular severe for the analysis of hyperspectral data where the ratio of the number of training samples to the number of dimensions is small.

For the purpose of mitigating the Hughes phenomenon and to reduce the effort of an analyst in terms of training sample selection, in this thesis a general adaptive classification procedure and then three specific methods to implement this procedure are developed to accommodate various training sample sizes. In this adaptive classification procedure, the semi-labeled samples (classification outputs) in addition to the original training samples are utilized to estimate class statistics in order to establish a positive feedback procedure where statistics estimation and classification enhance each other during each iteration. Eventually, a more accurate statistics estimation and higher classification accuracy can be achieved iteratively.

In Chapter 2, the role of semi-labeled samples on statistics estimation and feasibility of establishing the positive feedback procedure are investigated. Theoretical results show that when semi-labeled samples are used, statistics estimation may be enhanced. With the enhanced statistics estimates, classification performance may then be further improved. In other words, the positive feedback may be established. The degree of improvement of classification performance depends on the following factors: the number of semi-labeled samples, the classification accuracy (or the number of correctly classified semi-labeled samples) during each iteration, and the separability between classes. In other words, the more semi-labeled samples, the higher classification accuracy, and the more separation between classes, the more likely positive feedback is to be established, and the faster the final classification accuracy can reach the close to optimal value with given data set.

In Chapter 3, based on the theoretical results from the chapter 2, a self-learning and self-improving adaptive classifier is proposed. This adaptive classifier enhances statistics estimation and hence improves classification accuracy iteratively by utilizing the semi-labeled samples, in addition to the original training samples, in subsequent statistics estimation. In this iterative process, samples are initially classified based on the estimated statistics using the original training samples only. Then semi-labeled samples are subsequently used with the original training samples to update class statistics, and the samples are reclassified by the updated statistics. This process is repeated until convergence is reached where the classification accuracy changes a little. Since the class label accuracy of each sample is unknown, in order to control the influence of semi-labeled samples, the proposed method gives full weight to the training samples and reduced weight to semi-labeled samples. When this classifier is combined with ECHO, it is particularly advantageous on analysis of data where long statistics spatial dependency is strong

When the training sample size is extremely small, i.e., the number of entire training samples is comparable or even smaller than the number of (dimensions(poorly posed or ill-posed cases), using the adaptive method or a regularized covariance estimation method alone may not adequate. In Chapter 4, to deal with poorly posed or ill-posed cases, a family of adaptive covariance estimators is developed. This method combines the adaptive classification method and regularized covariance estimators. The semi-labeled samples (whose labels are determined by a decision rule) are incorporated in the process of determining the optimal regularized parameters and estimating those supportive covariance matrices that formulate final covariance estimators.

Finally, to fully utilize the rich spectral and spatial information contained in hyperspectral data, and to enhance the performance and robustness of the proposed adaptive classifier, in Chapter 5 an adaptive Bayesian contextual classifier based on the Markov random field is then developed. In this classifier, only interpixel class lable dependency context is considered. The joint prior probabilities of the classes of each pixel and its spatial neighbors are modeled by the Markov Random Field. The statistics estimation and classification are performed in a recursive mariner to allow the establishment of a positive feedback process in a computationally efficient manner.

All experimental results with the above three types of adaptive classifiers show that with a small training sample size, the statistics estimation can be enhanced, and classification accuracy can be improved iteratively. For most of experiments, the final classification accuracy can reach a close to optimal value. These classifiers can even outperform a supervised Maximum Likelihood classifier with a large training sample size.

6.2 Suggestions for Further Work

Extension of the adaptive classification procedure: the general philosophy of this adaptive classification procedure is to improve classification performance iteratively. During each iteration, information from the classification outputs is extracted and then it is utilized to update the process before classification, i.e., re-extract features, re-estimate statistics, and classification is performed with updated information. Semi-labeled samples (classification outputs) bridge the iterative process. Since semi-labeled samples contain partial class label information, they can be used wherever the training samples are used in the supervised classification process. The adaptive classification can be combined with any methods used in the steps of the classification process, i.e. preprocessing (Project Pursuit [41]), feature extraction (DBFE [42] and DAFE), subclass determination (LOOL [2]). The key to successfully use of semi-labeled samples is to control their effect appropriately.

Quantitative study on convergence of this adaptive classification procedure: from the experiments performed studied, we observed that the number of training samples for each class, the initial classification accuracy, and the number of semi-labeled samples for each class are the factors affecting the convergence rate and the final value of the classification accuracy. How these factors exactly determine convergence characteristics of this adaptive classification procedure is still open question. The study will provide valuable guidelines to use this adaptive classification procedure properly, and determine the minimum effort necessary from an analyst in terms of training sample selection.

REFERENCES

[1] K. Fukunaga, Zntro. Statistical Pattern Recognition, San Diego: Academic Press Inc., 1990

[2] 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

[3] G. F. Hughes, " On the mean accuracy of statistical pattern recognition", ZEEE Trans. *Information* Theory, 1968, Vol. IT-14, No. 1, pp 55-63

[4] F.A. Graybill, Matrices With Applications in Statistics, Belmont: Wadsworth Inc., 1983

[5] H.W. Sorenson, Parameter Estimation: Principles and Problems, New York: M. Dekker, 1980

[6] D.W. Hosmer, jr., "Information and Mixtures of two normal Distributions", *J.* Statics. Cbmput. Simul., 1997, Vol. 6, pp. 137-148

[7] B.M. Shahshahani, PhD dissertation, Purdue University, December 1993

[8] B.M. Shahshahani and D.A. Landgrebe, " The Effect of Unlabeled Samples in Reducing the Small Sample Size Problem and Mitigating the Hughes Phenomenon", *IEEE* Trans. On Geoscience and Remote Sensing, Vol. 32, No. 5, pp 1087-1095, September 1994

[9] R.A. Redner, H.F. Walker, " Mixture Densities, Maximum Likelihood and the EM Algorithm,"SZAM Review, Vol. 26, No. 2, pp 195-239, 1984

[10] Swain, P.H. and S.M. Davis, eds., Remote Sensing: The *Quantitative* Approach, McGraw Hill, 1978, Chapter 2

[11] D. A. Landgrebe and L. Biehl, An Zntroduction to *MultiSpec*, School of Electrical Engineering, Purdue University, IN. 47907-1285

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

[13] S.P. Lin and M.D. Perlman, " A Monte Carlo comparison of four estimators of a covariance matrix," Multivariate analysis—VI : Proceedings of the Sixth International Symposium on Multivariate Analysis, P.R. Krishnaiah, ed., Amsterdam: Elsevier Science Pub. Co., 1985, pp. 411-429

[14] P.W. Wahl and R.A. Kronmall, " Discriminant Functions when Covariances are Equal and Sample Sizes are Moderate," *Biometrics*, vol. 33, pp. 479-484, 1977

[15] S. Marks and O.J. Dunn, " Discriminant Functions when the Covariance Matrices are unequal," Journal of the American Statistical Association, vol., 69, pp. 555-559,1974

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

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

[18] Saldju Tadjudin and David Landgrebe, "Covariance Estimation With Limited Training Samples," ZEEE Transactions on Geoscience and Remote Sensing, Vol. 37, No. 4, pp. 2113-2118, July 1999

[19] J. Kitter and J. Foglein, " Contextual classification of multispectral pixel data," *Image* and Vision Computing, vol. 2, no. 1, pp. 13-29, Feb. 1984

[20] T. S. Yu and K.S. Fu, " Recursive contextual classification using a spatial stochastic model," Pattern Recognition, vol, 16, pp. 89-108, 1983

[21] N. Khazenie and M. M. Crawford, "Spatial-temporal autocorrelated model for contextual classification," ZEEE Trans. Geosci. Remote Sensing, vol. 28, pp. 529-539, July 1990

[22] B. Jeon and D. A. Landgrebe, " Spatio-temperoal contextual classification of remotely sensed multispectral data," Proc. of 1990 IEEE Intern. Conf. On Syst., Man, and Cybern., Los Angeles, CA, pp. 342-344

[23] N. A. Drake et al., "The development of improved algorithms for image processing and classification," Final Report of NERC, Dept. of Geography, University of Reading, U.K., 1987

[24] D. A. Landgrebe, " The Development of a Spectral-Spatial Classifier for Earth Observational Data," Pattern Recognition, vol. 12, pp. 165-175

[25] R. L. Kettig and D.A. Landgrebe, " Classification of Multispectral Image Data by Extraction and Classification of Homogenous Objects," IEEE *Trans.* Geosci. Remote Sensing, vol. GE-14, pp. 19-26, Jan. 1976

[26] R.M. Haralick et al., " Textural Features for Image Classification," IEEE Trans. *Syst.* Man, Cybern., pp. 610-621, 1973

[27] J. Haslett, " Maximum Likelihood Discriminant Analysis on the Plane Using a Makovian Model of Spatial Context," Pattern Recognition, vol. 18, no. 3, pp. 287-296, 1985

[28] P.H. Swain, S.B. Vardeman, and J.C. Tilton, " Contextual classification of rnultispectral image data," Pattern Recognition, vol. 13, no. 6, pp. 429-441, 1982

[29] J. Besag, " Spatial interaction and the statistical analysis of lattice: systems." J. Royal *Statist. Soc.,* vol. 36, no. 2, pp. 192-236, 1974

[30] C. Bouman and B. Liu, " Multiple resolution segmentation of' textured images," *IEEE* Trans. Pattern. Anal. Machine *Intell.,* vol. 13, no. 2, pp. 99-113, 1991

[31] B. Gidas, " A renormalization group approach to image processing problems," IEEE Trans. Pat. An. Mach, *Intell.*, vol. 11, no. 2, pp. 164-180, Feb. 1989

[32] P. Perez and F. Heitz, " Multiscale markov random fields and constrained relaxation in low level image analysis," Pro. IEEE. Int'l Conf. Acoust., Speech and Sig. Proc., vol. 3.,pp. 61-64, San Francisco, CA, March 23-26, 1992

[33] B. Jeon and D. A. Landgrebe, " Sptio-temporal contextual classification of remotely sensed multispectral data," Proc. of 1990 IEEE Intern. Conf. on Syst., Man, and Cybern., Los Angeles, CA, pp. 342-344

[34] Yonhong Jhung and Philip H. Swain, " Bayesian contextual classification based on modified M-estimates and markov random fields", IEEE Trans. Geosci. Remote Sensing, vol.34, no. 1, pp. 68-75, Jan. 1996

[35] Shan Yu, M. Berthod, and G. Giraudon,"Toward robust analysis of satellite images using map information-application to urban area detection", IEEE *Trans.* Geosci. Remote Sensing, vol.17, no. 4, pp.1925-1938, July 1999

[36] S. German and D. Geman, " Stochastic relaxation, Gibbs distributions, and the Bayesian restoration," IEEE Trans. Pat. An. Mach, *Intell,* vol. PAMI-6, no. 6, pp. 721-741, Nov. 1984

[37] C. A. Bouman and M. Shapiro, "A multiscale random field model for Bayesian image segmentation," IEEE Trans. on Image Processing, vol. 3, no. 2, pp. 162-177, March 1994

[38] J. Besag, " On the statistical analysis of dirty pictures," J. Royal Statist. *Soc.,* vol. 68, pp.259-302, 1986

[39] R. Kinderman and J. L. Snell, " Markov random fields and their applications," Amer. Math. *Soc.,* vol. 1, pp. 1-142, 1980

[40] S. Z. LI, Markov Random Field Modeling in Computer Vision,,Berlin, Germany: Spring-Verlag, 1995

[41] W.K. Hastings," Monte Carlo sampling methods using Markov chains and their applications,"Biomertrika, vol. 57, pp, 97-109, 1970

[42] Jimenez, Luis, and David Landgrebe, "Supervised Classification in High Dimensional Space: Geometrical, Statistical, and Asymptotical Properties of Multivariate Data," IEEE Transactions on System, Man, and Cybernetics, Volume 28, Part C, No. 1, pp. 39-54, February 1998

[43] Chulhee Lee and David A. Landgrebe, "Feature Extraction Based On Decision Boundaries," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 15, No. 4, pp. 388-400, April 1993

Appendix A: Derivation of Fisher Information Matrix for Two Normal Distributions

The Fisher information matrix expressed in Eq. (2.2) can be written as:

$$
\begin{aligned}
I_{sl} = {} & n_1 P_1 \int_{\Omega_1} [\frac{\partial}{\partial\theta}\log f(x,\theta)][\frac{\partial}{\partial\theta}\log f(x,\theta)]^T f_1(x|\mu_1,\Sigma_1)dx \\
& + n_1 P_2 \int_{\Omega_2} [\frac{\partial}{\partial\theta}\log f(x,\theta)][\frac{\partial}{\partial\theta}\log f(x,\theta)]^T f_2(x|\mu_2,\Sigma_2)dx \\
& + n_2 P_1 \int_{\Omega_2} [\frac{\partial}{\partial\theta}\log f(x,\theta)][\frac{\partial}{\partial\theta}\log f(x,\theta)]^T f_1(x|\mu_1,\Sigma_1)dx \\
& + n_2 P_2 \int_{\Omega_1} [\frac{\partial}{\partial\theta}\log f(x,\theta)][\frac{\partial}{\partial\theta}\log f(x,\theta)]^T f_2(x|\mu_2,\Sigma_2)dx
\end{aligned}
$$

Since the vector of unknown parameters is $\theta = [\mu_1^T, \mu_2^T]^T$, therefore:

$$
\frac{\partial}{\partial\theta}\log f_1(x) = \frac{1}{f_1(x)}\frac{\partial}{\partial\theta}f_1(x) = \frac{1}{f_1(x)}\begin{bmatrix} f_1(x)(x-\mu_1)\Sigma_1^{-1}(x-\mu_1)^T \\ 0 \end{bmatrix}
$$

$$
\frac{\partial}{\partial\theta}\log f_2(x) = \frac{1}{f_2(x)}\frac{\partial}{\partial\theta}f_2(x) = \frac{1}{f_2(x)}\begin{bmatrix} 0 \\ f_2(x)(x-\mu_2)\Sigma_2^{-2}(x-\mu_2)^T \end{bmatrix}
$$

With $\mu_1 = 0$ and $\Sigma_1 = \Sigma_2 = \mathbf{I}$,, the above can be simplified as:

$$
\frac{\partial}{\partial\theta}\log f_1(x) = \frac{1}{f_1(x)}\frac{\partial}{\partial\theta}f_1(x) = \frac{1}{f_1(x)}\begin{bmatrix} f_1(x)xx^T \\ 0 \end{bmatrix}
$$

$$
\frac{\partial}{\partial\theta}\log f_2(x) = \frac{1}{f_2(x)}\frac{\partial}{\partial\theta}f_2(x) = \frac{1}{f_2(x)}\begin{bmatrix} 0 \\ f_2(x)(x-\mu_2)(x-\mu_2)^T \end{bmatrix}
$$

Also, in the canonical case under consideration, the subspaces Ω_1 and Ω_2 can be determined as:

$$x \in \Omega_1 \Leftrightarrow x_1 \leq t$$

$x \in \Omega_2$ e x , >t

where

$$t = \frac{1}{\Delta}\log(\frac{P_1}{P_2}) + \frac{1}{2}\Delta$$

If we define:

$$I_1 = \int_{\Omega_1}[\frac{\partial}{\partial\theta}\log f_1(x)][\frac{\partial}{\partial\theta}\log f_1(x)]^T f_1(x|\mu_1,\Sigma_1)dx$$

$$I_2 = \int_{\Omega_2}[\frac{\partial}{\partial\theta}\log f_2(x)][\frac{\partial}{\partial\theta}\log f_2(x)]^T f_2(x|\mu_2,\Sigma_2)dx$$

$$I_3 = \int_{\Omega_2}[\frac{\partial}{\partial\theta}\log f_1(x)][\frac{\partial}{\partial\theta}\log f_1(x)]^T f_1(x|\mu_1,\Sigma_1)dx$$

$$I_4 = \int_{\Omega_1}[\frac{\partial}{\partial\theta}\log f_2(x)][\frac{\partial}{\partial\theta}\log f_2(x)]^T f_2(x|\mu_2,\Sigma_2)dx$$

then we have:

$$I_1 = \begin{bmatrix} \alpha_1 & 0 & 0 \\ 0 & \beta_1 I_{d-1} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$I_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \alpha_2 & 0 \\ & 0 & \beta_2 I_{d-1} \end{bmatrix}$$

$$I_3 = \begin{bmatrix} 1-\alpha_1 & 0 & 0 \\ 0 & (1-\beta_1)I_{d-1} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$I_4 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1-\alpha_2 & 0 \\ 0 & 0 & (1-\beta_2)I_{d-1} \end{bmatrix}$$

$$k_1 = r_c\alpha_1 + (1-r_c)(1-\alpha_1)$$

$$k_2 = r_c\beta_1 + (1-r_c)(1-\beta_1)$$

$$k_3 = r_c\alpha_2 + (1 - r_c)(1 - a,)$$

$$k_4 = r_c\beta_2 + (1 - r_c)(1 - \beta_2)$$

$$r_c = \frac{n_1}{n}$$

$$\alpha_1 = \Phi(t) - t\phi(t)$$

$$\beta_1 = \Phi(t)$$

$$\alpha_2 = \Phi(\Delta - t) - (t - \Delta)\phi(t - \Delta)$$

$$\beta_2 = \Phi(\Delta - t)$$

$$\Phi(t) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{t} e^{-\frac{x^2}{2}}dx$$

$$\phi(t) = \frac{1}{\sqrt{2\pi}}e^{-\frac{t^2}{2}}$$