

UC Riverside

UC Riverside Electronic Theses and Dissertations

Title

Alternative Approaches in Multi-label Neutral Zone Classification Problems

Permalink

<https://escholarship.org/uc/item/7cx3b94x>

Author

Le, Rebecca Phuonganh

Publication Date

2012

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA
RIVERSIDE

Alternative Approaches
in Multi-label Neutral Zone Classification Problems

A Dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Applied Statistics

by

Rebecca Phuonganh Le

June 2012

Dissertation Committee:

Dr. Daniel R. Jeske, Chairperson

Dr. Keh-Shin Lii

Dr. James Borneman

Copyright by
Rebecca Phuonganh Le
2012

The Dissertation of Rebecca Phuonganh Le is approved:

Committee Chairperson

University of California, Riverside

ACKNOWLEDGEMENTS

My special thanks and appreciations go to Dr. Daniel Jeske for his passion, dedication, valuable guidance and support to complete my research work.

My sincerest thanks go to Dr. Keh-Shin Lii and Dr. James Borneman for being my dissertation committee and spending time to review this work.

I gratefully thank Paula Lemire and Perla Fabelo for professional administrative supports during my years at UCR. I would also like to thank Dr. Linda Penas and the Statistics department for providing very useful computing resources.

My thanks are addressed to Dr. Paul Ruegger for his unscheduled accesses and numerous helpful discussions on the background of the biological data set. I would like to thank Wei Li for his help to use the tool created from the Computer department even though the result of that work is not presented here.

I warmest thanks go to Ms. Jill Smith for being nice to me all the time. It is my pleasure to work with her.

My big thanks go to Romy Vu for being my best friend more than ten years and taking care my laptop and desktop. My grateful thanks also go to Dr. Brian Vu for lending me his fast computer to run my simulation.

I would like to express my thanks to all my friends in the Statistics department for always being available when I need to talk to them.

Last but not least, my deepest thanks go to my parents, sisters, brother, and four adorable dogs for their forever love and supports.

ABSTRACT OF THE DISSERTATION

Alternative Approaches in Multi-label Neutral Zone Classification Problems

by

Rebecca Phuonganh Le

Doctor of Philosophy, Graduate Program in Applied Statistics
University of California, Riverside, June 2012
Dr. Daniel R. Jeske, Chairperson

Various multi-label classification algorithms are broadly developed in the literature. Each of these existing methods has different strengths and drawbacks, and very few attempts have been performed to address the uncertainty. Recently, neutral zone classifiers have been presented to deal with ambiguous data to improve the accuracy of the estimated classification results by trading off the relatively high penalty of an incorrect classification with a lower penalty for remaining neutral about class membership. The existing neutral zone methods were developed to construct multiple single-label classifiers for multi-label data. In this work, we present different classification alternatives for multi-label data using a standard logistic regression model, a generalized linear mixed model and Markov random field to relax the current underlying assumptions. The proposed neutral zone classification methods are

implemented and tested on simulation data sets and on the biological data set. Their results suggest that our proposed classification approaches are useful alternatives for practical application when working with multi-label data.

Table of Contents

List of Tables

List of Figures

1. Introduction	1
1.1 Multi-label Classifiers	1
1.1.1 Algorithm Transformation Methods	2
1.1.2 Algorithm Adaptation Methods	4
1.2 Illustrative Example	5
1.2.1 Classification Approaches	5
1.2.2 Simulation Study	7
1.3 Previous Work	10
1.3.1 Biological Application	10
1.3.2 Neutral Zone Classification Approach	11
1.3.3 Review of Neutral Zone Classifiers	11
1.3.4 Applications of Neutral Zone Classifiers	13
1.4 Structure of the Dissertation	14
2. Logit-based Neutral Zone Classifier	16
2.1 Model Definition	16
2.2 Parameter Estimation	17
2.3 Classifier Definition	19
2.4 Neutral Zone Boundary	20
2.5 Naive Logit-based Neutral Zone Classifier	23
3. Markov Random Field Neutral Zone Classifier	26
3.1 Markov Random Field	26
3.2 Markov Random Field Neutral Zone Classifier	31
3.2.1 Graph Construction	31
3.2.2 Parameter Estimation	32
3.3 Neutral Zone Decision Algorithm	34
3.3.1 Algorithm	34
3.2.2 Implementation Examples	35

4. Simulation Studies	41
4.1 Simulation Algorithm	41
4.1.1 Generalized Linear Mixed Model Algorithm	43
4.1.2 Markov Random Field Algorithm	44
4.1.3 Convergence Diagnostic for Markov Random Field Approach	47
4.2 Results	47
4.2.1 Findings	47
4.2.2 Discussion	51
5. Biological Application	57
5.1 Data Introduction and Preparation	57
5.2 Analysis of the Logit-based Classifier	60
5.2.1 Parameter Estimate	60
5.2.2 Neutral Zone Limit	60
5.3 Implementation of the Markov Random Field Classifiers	63
5.3.1 Graph Construction and Parameter Estimation	63
5.3.2 Neutral Zone Limit	65
5.4 Results	67
5.4.1 Findings	67
5.4.2 Discussion	67
6. Concluding Remarks and Future Work	70
6.1 Concluding Remarks	70
6.2 Future Work	71
Bibliography	72
Appendices	
A. Proof of the Theorem	77
B. Proof of the Markov-Gibbs Equivalence	78
C. Analysis of Simulated Data	80
D. Analysis of Fungi Data	96

List of Tables

Table 2.1	A Misclassification table for any fixed $L_k \in [0, 1]$	20
Table 4.1	Comparison of the proposed neutral zone classifiers	52
Table 5.1	Structure of a fungi training data set for N clones and K probes	57
Table 5.2	Parameter estimates for 26 probes of fungi data of the logit-based and the naive logit-based classifiers (LNZC and nLNZC)	61
Table 5.3	Parameter estimates of the Markov random field classifier for 26 probes of fungi data	64
Table 5.4	Neutral zone limits for 26 probes of the fungi data for the logit-based, naive logit-based, threshold-based and Gaussian-mixture neutral zone classifiers (LNZC, nLNZC, MNZC1 and MNZC2)	66
Table C.1	Hard and soft error rates of proposed classification methods using the GLMM simulated data when $K = 5$ and $\rho_b = 0$	80
Table C.2	Total error rates of proposed classification methods using the GLMM simulated data when $K = 5$ and $\rho_b = 0$	81
Table C.3	Hard and soft error rates of proposed classification methods using the GLMM simulated data when $K = 5$ and $\rho_b = 0.4$	82
Table C.4	Total error rates of proposed classification methods using the GLMM simulated data when $K = 5$ and $\rho_b = 0.4$	83
Table C.5	Hard and soft error rates of proposed classification methods using the GLMM simulated data when $K = 5$ and $\rho_b = 0.7$	84
Table C.6	Total error rates of proposed classification methods using the GLMM simulated data when $K = 5$ and $\rho_b = 0.7$	85
Table C.7	Hard and soft error rates of proposed classification methods using the GLMM simulated data when $K = 10$ and $\rho_b = 0$	86
Table C.8	Total error rates of proposed classification methods using the GLMM simulated data when $K = 10$ and $\rho_b = 0$	87

Table C.9	Hard and soft error rates of proposed classification methods using the GLMM simulated data when $K = 10$ and $\rho_b = 0.4$	88
Table C.10	Total error rates of proposed classification methods using the GLMM simulated data when $K = 10$ and $\rho_b = 0.4$	89
Table C.11	Hard and soft error rates of proposed classification methods using the GLMM simulated data when $K = 10$ and $\rho_b = 0.7$	90
Table C.12	Total error rates of proposed classification methods using the GLMM simulated data when $K = 10$ and $\rho_b = 0.7$	91
Table C.13	Hard and soft error rates of proposed classification methods using the MRF simulated data when $K = 5$	92
Table C.14	Total error rates of proposed classification methods using the MRF simulated data when $K = 5$	93
Table C.15	Hard and soft error rates of proposed classification methods using the MRF simulated data when $K = 10$	94
Table C.16	Total error rates of proposed classification methods using the MRF simulated data when $K = 10$	95
Table D.1	Data summary of five subsets categorized by hybridization classes	96
Table D.2	Five-fold cross-validation hard error rates of five neutral zone classifiers for the fungi data set	97
Table D.3	Five-fold cross-validation soft error rates of five neutral zone classifiers for the fungi data set	98
Table D.4	Five-fold cross-validation total error rates of five neutral zone classifiers for the fungi data set	99
Table D.5	Distribution parameter estimates for non-binding class of fungi data	100
Table D.6	Distribution parameter estimates for binding class of fungi data	101

List of Figures

Figure 1.1	Performances of two classification methods for two-label simulated data with equal priors	8
Figure 1.2	Performances of two classification methods for three-label simulated data with equal priors	9
Figure 1.3	A graphical illustration of a neutral zone classifier	12
Figure 1.4	Illustrations of the class assignment regions of (a) Three-class Bayes classifier and (b) Three-class neutral zone classifier	13
Figure 2.1	Schematic plot of link function (2.1) vs. covariate with different fixed parameters	18
Figure 2.2	Graph of misclassification error rate vs. neutral zone limit L	22
Figure 2.3	Magnified version of neighboring location of the neutral zone L^*	23
Figure 2.4	Flowchart of the procedure for constructing the logit-based neutral zone classifier using two-fold cross-validation to find L^*	25
Figure 3.1	An undirected graph is constructed by six nodes and seven edges	28
Figure 3.2	Connections between nodes of the graph are constructed depending on Pearson correlation coefficients	32
Figure 3.3	Scree plots of 32 configurations when $K = 5$	36
Figure 3.4	A Markov random field constructed by four nodes and four edges	37
Figure 3.5	Graphical allocation illustration of parameter estimates of a four-node Markov random field	38
Figure 4.1	Flowchart of the procedure for simulating data, constructing proposed classifiers and obtaining misclassification error rates	43
Figure 4.2	Simulated K node graphs associated with different structures of Σ_x	46

Figure 4.3	Trace plots of Gibbs sampler chains with number of labels $K = 5$ based on 1,000 iterations using two different initial states (1,0,1,0,1) and (1,1,1,1,1)	48
Figure 4.4	Trace plots of Gibbs sampler chains with number of labels $K = 10$ based on 1,000 iterations using two different initial states (0,0,0,0,0,0,0,0,0,0) and (1,1,1,1,1,1,1,1,1,1)	49
Figure 4.5	Misclassification error rates of proposed neutral zone classifiers vs. covariate data structure for the GLMM simulated data when $K = 5$	53
Figure 4.6	Misclassification error rates of proposed neutral zone classifiers vs. covariate data structure for the GLMM simulated data when $K = 10$	54
Figure 4.7	Misclassification error rates of proposed neutral zone classifiers vs. covariate data structure for the MRF simulated data when $K = 5$	55
Figure 4.8	Misclassification error rates of proposed neutral zone classifiers vs. covariate data structure for the MRF simulated data when $K = 10$	56
Figure 5.1	Layout of the data preparation process	59
Figure 5.2	Graphical illustration of finding the neutral zone limits of two probes	62
Figure 5.3	An undirected graph for 26 probes of fungi data	63
Figure 5.4	Scree plot of 25 configurations with highest pseudo-likelihoods for one selected clone of fungi data	65
Figure 5.5	Five-fold cross-validation misclassification error rates of all neutral zone classification methods	68
Figure 5.6	Distributions of log(intensity) and five-fold cross-validation misclassification error rates of fungi data for probes 2 and 3	69
Figure D.1	Distributions of log(intensity) of fungi data for 26 probes	102-106

Chapter 1

Introduction

1.1: Multi-label Classifiers

Classification is a very popular approach to many decision-making problems. Examples of classification tasks consist of assigning a certain disease as cancerous or non-cancerous, or categorizing a tumor type as benign or acute. More formally, given an observed measurement of x , a classifier is stated as a function f , built from a learning process on training data, that assigns the discrete class label $\hat{C} = f(x)$. This approach is termed a single-label classification. Multi-label classifiers might be built to map a vector of an observed attribute $x = \{x_k\}_{k=1..K}$ to a vector of predicted class labels $\hat{C} = \{\hat{C}_k\}_{k=1..K}$ corresponding to each subject, where K is the number of labels. Here, $\hat{C}$ represents a membership profile of the subject with multiple class labels. Several multi-label classifiers were developed in the literature for use in many applications, such as emotion detection (Li, T. & Ogihara, M., 2006; Wieczorkowska et al., 2006; Trohidis et al., 2008), image classification (Boutell et al., 2004; Zhang, M.L & Zhou, Z.H., 2007; Yang et al., 2007), text categorization (Ueda & Saito, 2003; Rousu et al., 2006), and bioinformatics (Barutcuoglu et al., 2006; Jin et al., 2008; Schietgat et al., 2010), etc.

1.1.1: Algorithm Transformation Methods

This approach transforms a multi-label problem into one or more single-label problems. One of the most common existing transformation methods is the binary relevance method (Boutell et al., 2003; Tsoumakas et al., 2010). This approach constructs K data subsets, one for each class label with the value of 1 if present and 0 otherwise, that contain all subjects in the original data set. Then, each subset is used to develop a single-label binary classifier for its corresponding class label. Ranking by a pair-wise comparison method (Hullermeier et al., 2008) is another popular alternative and works as follows. The multi-label original data set is split into $K(K - 1)/2$ subsets, one for each pair of labels $(C_k, C_{k'})$, where $1 \leq k, k' \leq K$. For each subset, class labels of each pair are not shown on each subject simultaneously. Then, each data subset is used to develop a binary classifier corresponding to each pair of labels to find a rule to distinguish them. All binary classifiers are applied to predict class labels for a new subject, and the assignment for each label is made based on its highest ranking from the voting scheme. Ranking by a pair-wise comparison technique is extended to a calibrated label ranking method (Furnkranz et al., 2008) by using an artificial label C_0 as the breaking point of the ranking to distinguish relevant and irrelevant groups of class labels, where $K = 5$, as follows:

$$C_{(1)} > C_{(2)} > C_{(3)} > C_{(4)} > C_{(5)} \rightarrow C_{(1)} > C_{(2)} > C_{(3)} > \mathbf{C_0} > C_{(4)} > C_{(5)}$$

This method has the same setting as the binary relevance method. For each class label, a binary classification rule is formed to classify that corresponding label as "positive" or

"negative" and vice versa for the artificial label C_0 , respectively, depending on whether that label is annotated with each subject or not.

The label powerset method or label combination considers each unique subset of labels that appears in the original multi-label training data set as a single of a new single-label classification problem (Santos et al., 2011; Tsoumakas et al., 2010). For example, the sets of binary multiple labels $C = \{C_1, C_2, C_3, C_4, C_5\}$ of the five-subject original training data set defined as $C_{subject\ 1} = \{1,1,0,0,1\}$, $C_{subject\ 2} = \{1,0,1,0,0\}$, $C_{subject\ 3} = \{0,1,1,0,0\}$, $C_{subject\ 4} = \{0,0,0,0,1\}$ and $C_{subject\ 5} = \{0,0,1,1,1\}$ are transformed to five corresponding unique single labels termed $C_{subject\ 1} = C_{125}$, $C_{subject\ 2} = C_{13}$, $C_{subject\ 3} = C_{23}$, $C_{subject\ 4} = C_5$ and $C_{subject\ 5} = C_{345}$, respectively. As in the binary relevance method, one binary classifier is developed for each new single label. These classifiers are applied on a new subject to compute their corresponding probability distributions. A new set of predicted class labels with the highest probability is used for the subject.

Several other label-based transformation methods have been used, such as the random k-label sets method (Tsoumakas et al., 2011), the pruned sets method (Read, 2008), and ensembles of the pruned sets method (Read et al., 2008), etc. These transformation methods appear to have both strengths and weaknesses (Dharmadhikari et al., 2011).

1.1.2: Algorithm Adaptation Methods

All the existing techniques have been developed to handle multi-label data directly. The C4.5 algorithm builds a hierarchy decision tree where multiple labels are located at the tree nodes created using a measure of entropy (Blockeel et al., 2006). Some drawbacks of this method are that it (1) does not consider the possible correlation among labels; (2) cannot apply to non-hierarchy multi-label data, and (3) does not generalize data appropriately if creating complicated trees. The RankSVM method (Elisseeff & Weston, 2002) is another adaptation approach that employs the Support Vector Machine using a Kernel-based model (i.e., ordinal regression) to minimize a ranking loss identified as the number of discordant label pairs. This ranking algorithm is very time-consuming because its computational cost is proportional to NK^2 , where N and K are the number of subjects in the training data set and the number of class labels, respectively.

Multi-label k-Nearest Neighbors (ML-kNN) is developed from the k-Nearest Neighbors (Zhang & Zhou, 2005; Brinker & Hullermeier, 2007). This method works by: (1) ranking all computed distances (i.e. Euclidean metric) between a new object and all objects in the training set; (2) retrieving k nearest neighbors based on these distances; and (3) determining the set of class labels for the new object using a Maximum A Posteriori method based on prior and posterior probabilities obtained from the label sets of neighboring objects. Same as the RankSVM method, ML-kNN is time-consuming and needs a large memory space to bank a big set of labels of all objects in the training set.

Other adaptation algorithms are presented for multi-label data. One of them is AdaBoost.MH (Schapire & Singer, 1999), which is formed to improve accuracy by

minimizing Hamming loss error defined as a sum of incorrect predicted class labels; collective multi-label (CML) is developed to capture label co-occurrences for each pair of labels; and collective multi-label with features (CMLF) is built to capture the influence of an individual feature on the co-occurrence probability of a pair of labels (Ghamrawi & McCallum, 2005).

1.2: Illustrative Example

1.2.1: Classification Approaches

We conduct an example to examine the performances of the joint and marginal classification approaches, which account for and ignore correlations among labels, respectively, for multi-label data. Consider a vector of two binary class labels $C = (C_1, C_2)'$ and a corresponding covariate $x = (x_1, x_2)'$ that are involved in the experiment. Assume the distribution of $x|C$ is $MVN(\mu_{C_1C_2}, \Sigma)$ with priors $\pi_{C_1C_2}$ for all four populations: $(0,0)'$, $(0,1)'$, $(1,0)'$ and $(1,1)'$ of C . To be more specific, the joint density function of C is written as $f_{x,C} = \pi_{C_1C_2}f_{x|C_1C_2}$. Then, the conditional density function of C_1 is:

$$\begin{aligned} f_{x_1|C_1=0} &= \frac{f_{x_1,C_1=0}}{P(C_1 = 0)} = \frac{f_{x_1,C_1=0,C_2=0} + f_{x_1,C_1=0,C_2=1}}{\pi_{00} + \pi_{01}} \\ &= \frac{\pi_{00}f_{x_1|00} + \pi_{01}f_{x_1|01}}{\pi_{00} + \pi_{01}} \\ f_{x_1|C_1=1} &= \frac{f_{x_1,C_1=1}}{P(C_1 = 1)} = \frac{f_{x_1,C_1=1,C_2=0} + f_{x_1,C_1=1,C_2=1}}{\pi_{10} + \pi_{11}} \end{aligned}$$

$$= \frac{\pi_{10}f_{x_1|10} + \pi_{11}f_{x_1|11}}{\pi_{10} + \pi_{11}}$$

Similarly for C_2 :

$$f_{x_2|C_2=0} = \frac{\pi_{00}f_{x_2|00} + \pi_{10}f_{x_2|10}}{\pi_{00} + \pi_{10}}$$

$$f_{x_2|C_2=1} = \frac{\pi_{01}f_{x_2|01} + \pi_{11}f_{x_2|11}}{\pi_{01} + \pi_{11}}$$

The Bayes rule for the class labels of C_1 and C_2 for the joint classification approach is:

$$\hat{C} = \begin{cases} (0, 0) & \text{if } (\pi_{00}f_{x|00} > \pi_{01}f_{x|01}) \& (\pi_{00}f_{x|00} > \pi_{10}f_{x|10}) \& (\pi_{00}f_{x|00} > \pi_{11}f_{x|11}) \\ (0, 1) & \text{if } (\pi_{01}f_{x|01} > \pi_{00}f_{x|00}) \& (\pi_{01}f_{x|01} > \pi_{10}f_{x|10}) \& (\pi_{01}f_{x|01} > \pi_{11}f_{x|11}) \\ (1, 0) & \text{if } (\pi_{10}f_{x|10} > \pi_{00}f_{x|00}) \& (\pi_{10}f_{x|10} > \pi_{01}f_{x|01}) \& (\pi_{10}f_{x|10} > \pi_{11}f_{x|11}) \\ (1, 1) & \text{if } (\pi_{11}f_{x|11} > \pi_{00}f_{x|00}) \& (\pi_{11}f_{x|11} > \pi_{01}f_{x|01}) \& (\pi_{11}f_{x|11} > \pi_{10}f_{x|10}) \end{cases}$$

If C_1 and C_2 are two independent class labels, the marginal classification approach

defines the following two Bayes classifiers:

$$\hat{C}_1 = 1 \quad \text{if } (\pi_{10} + \pi_{11})f_{x_1|C_1=1} > (\pi_{00} + \pi_{01})f_{x_1|C_1=0}$$

$$\hat{C}_2 = 1 \quad \text{if } (\pi_{01} + \pi_{11})f_{x_2|C_2=1} > (\pi_{00} + \pi_{10})f_{x_2|C_2=0}$$

The misclassification error rates for each label can be calculated as:

$$P(\hat{C}_1 = 1|C_1 = 0) = \frac{P(\hat{C}_1 = 1, C_1 = 0)}{P(C_1 = 0)}$$

$$P(\hat{C}_1 = 0|C_1 = 1) = \frac{P(\hat{C}_1 = 0, C_1 = 1)}{P(C_1 = 1)}$$

$$P(\hat{C}_2 = 1|C_2 = 0) = \frac{P(\hat{C}_2 = 1, C_2 = 0)}{P(C_2 = 0)}$$

$$P(\hat{C}_2 = 0|C_2 = 1) = \frac{P(\hat{C}_2 = 0, C_2 = 1)}{P(C_2 = 1)}$$

1.2.2: Simulation Study

A data set of 100,000 observations is simulated to obtain the true values of $C = (C_1, C_2)'$ and corresponding covariate $x = (x_1, x_2)'$ using the means of four populations: $(0, 0)'$, $(0, 1)'$, $(1, 0)'$ and $(1, 1)'$, respectively, as $\mu_{00} = [0.5, 0.5]'$, $\mu_{01} = [1.0, 4.0]'$, $\mu_{10} = [4.0, 0.5]'$ and $\mu_{11} = [5.5, 5.0]'$; variance-covariance matrix Σ contains variances $\sigma_{11}^2 = \sigma_{22}^2 = 5$; and correlation coefficient values varying from 0 to 0.90. Then, Bayes classifiers of joint and marginal approaches described above are employed to get $\hat{C} = (\hat{C}_1, \hat{C}_2)'$ given on the simulated covariate x . We compute misclassification error rates by comparing the predicted class labels $\hat{C}$ with the true value C for two classification approaches. The performances of two classification methods are illustrated in **Figure 1.1** for equal priors $\pi_{C_1 C_2} = 0.25$. As expected, the joint method is consistently and aggressively superior to the marginal approach, since it gives smaller conditional error rates when the correlation coefficient is getting larger.

Extending this example to $C = (C_1, C_2, C_3)'$, we generate a simulated data set using $\mu_{000} = [0.5, 0.5, 0.5]'$, $\mu_{001} = [0.5, 1.0, 4.0]'$, $\mu_{010} = [1.0, 4.0, 0.5]'$, $\mu_{011} = [0.5, 5.5, 5.0]'$, $\mu_{100} = [4.5, 0.5, 0.5]'$, $\mu_{101} = [5.5, 1.0, 4.0]'$, $\mu_{110} = [5.0, 4.0, 0.5]'$, and $\mu_{111} = [5.5, 5.5, 5.0]'$ (the computing algorithm is not shown). The results are in **Figure 1.2** for equal priors $\pi_{C_1 C_2 C_3} = 0.125$. Our simple illustration shows that it can be very important to account for correlation among the components in x . Hence, more sophisticated multi-label classification algorithms are needed for real-world problems.

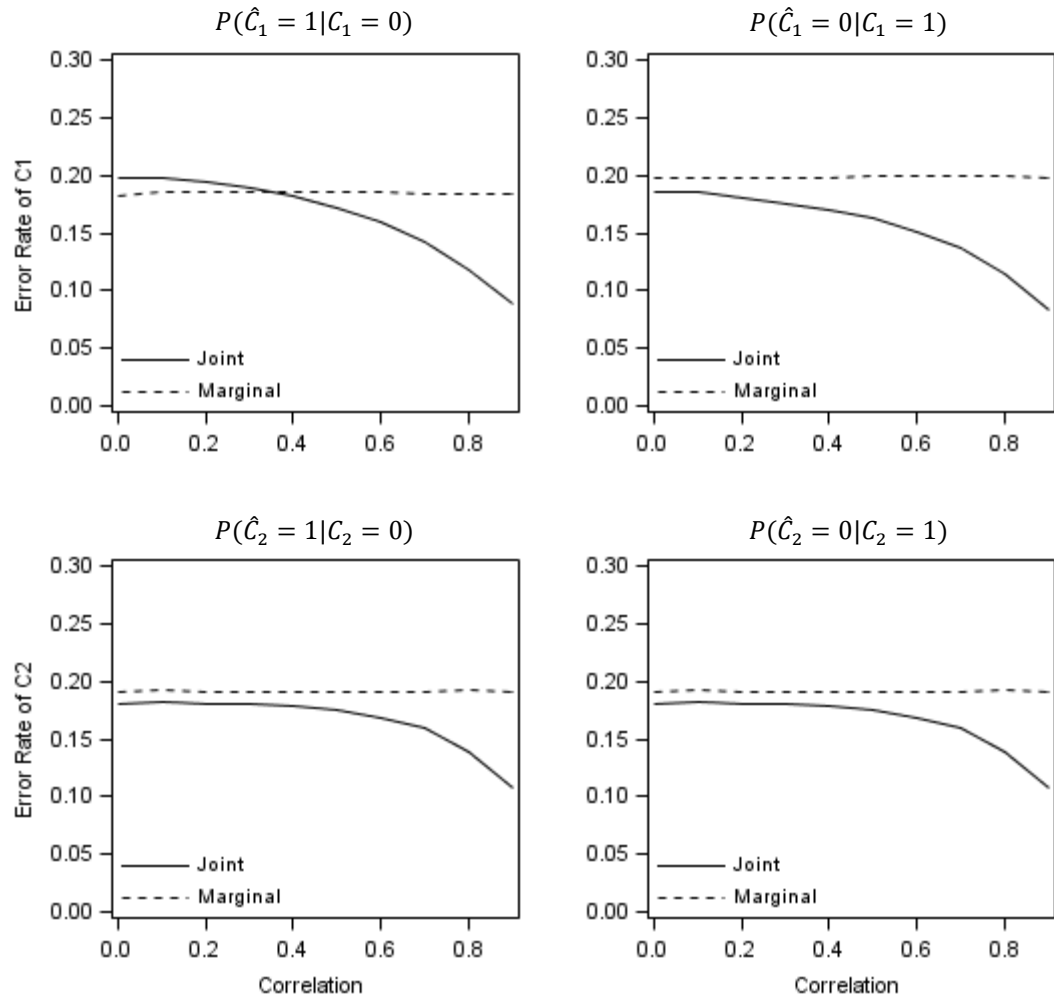


Figure 1.1: Performances of two classification methods for two-label simulated data with equal priors

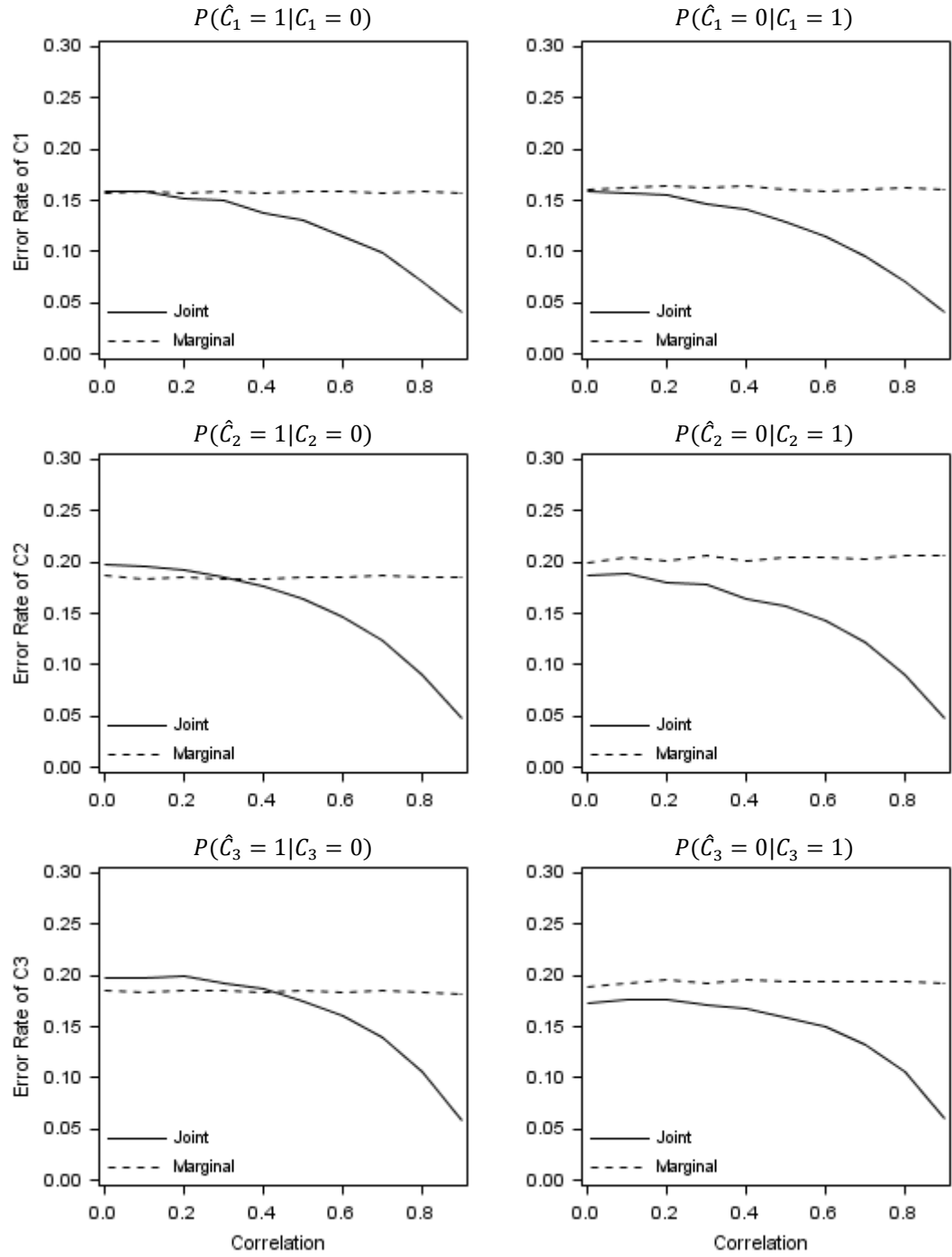


Figure 1.2: Performances of two classification methods for three-label simulated data with equal priors

1.3: Previous Work

1.3.1: Biological Application

An array-based method called Oligonucleotide Fingerprinting of rRNA Genes (OFRG) has been developed to identify a sample of microbial community in a cost-effective manner (Valinsky et al., 2002ab; Bent et al., 2006). Compared to nucleotide sequence analysis, OFRG provides a statistical analysis of microbial community composition with a cost reduction of up to 90 percent. In brief, cloned rRNA genes, or clones, i.e., bacteria or fungi, are obtained through the process of DNA extraction and polymerase chain reaction (PCR) amplification of samples (mouse, human, soil, etc.). The rRNA clones are arrayed on nylon membranes. Then, they undergo a series of microarray hybridization experiments with the optimally selected short DNA probes of ten nucleotides in length. Hybridization signal intensities are observed for each experiment between the rRNA genes and probes, one at the time. This process finally produces a full set of hybridization measurements of rRNA genes after the experiment has been completed with all the probes.

As with many other real-world applications, microbial communities could be identified by applying traditional and unsupervised learning methods directly to these observed hybridization intensities. In the OFRG study, researchers make a first attempt at classifying these intensity measurements into classes of binding and non-binding to form a fingerprint before any clustering is done. One hopes that the clustering using the fingerprint will be more precise and improved by reducing the impact of experimental systematic variations in hybridization intensities. For this biological application, we are

focus on the classification rule used to build the hybridization fingerprint, which is a vector of class labels C , using a joint approach given a vector of observed hybridization measurements x .

1.3.2: Neutral Zone Classification Approach

A neutral zone classification rule is able to identify ambiguities when predicting a class label for an object. It helps in improving the accuracy of the estimated classification results by trading off the relatively high penalty of an incorrect classification with a lower penalty for remaining neutral about class membership. In the OFRG study, this approach hopefully leads to a better set of clusters obtained from the fingerprints as compared to using raw hybridization intensities.

1.3.3: Review of Neutral Zone Classifiers

The first formal neutral zone classifier algorithm was developed by Jeske et al. 2007 for the two-class context. This classifier was later modified and extended to three-class context by Yu et al. 2010. Notations in this review section closely adhere to Yu et al. 2010.

To-date, neutral zone classifiers have been developed by constructing multiple single-label classifiers for multi-label data. The process of building a neutral zone classifier (NZC) includes using a training data set where class labels, C_{jk} , associated with observed measurements, x_{jk} , are theoretically known and, where $j = 1, 2 \dots N$ and $k = 0, 1, \dots K$ are indexes of objects and covariates, respectively. Denote the probability

density function of X conditioned on a class label $C = i$ as $f_i(x) = f_{X|C=i}(x)$. Define $\pi_i \in (0,1)$ as the priori probabilities of the classes where $i = 0, 1$ and $\sum_{i=0}^1 \pi_i = 1$. The Bayes posterior probability, $p_i(x)$, is given as $P(C = i|X = x) = \pi_i f_i(x) / \sum_{i=0}^1 \pi_i f_i(x)$. Note that the traditional Bayes rule classifier is defined as $\hat{C}(x) = \operatorname{argmax}_i p_i(x)$ or $\hat{C}(x) = \operatorname{argmax}_i \pi_i f_i(x)$. Consider a fixed limit $L \in [0,1]$ in the neutral zone frame work (see **Figure 1.3** for an illustration of the class region assignment). The neutral zone classifier (NZC) is stated as:

$$\hat{C}(x; L) = \begin{cases} 0 & \text{if } p_1(x) < 0.5 - L/2 \\ 1 & \text{if } p_1(x) > 0.5 + L/2 \\ N & \text{otherwise} \end{cases} \quad (1.1)$$

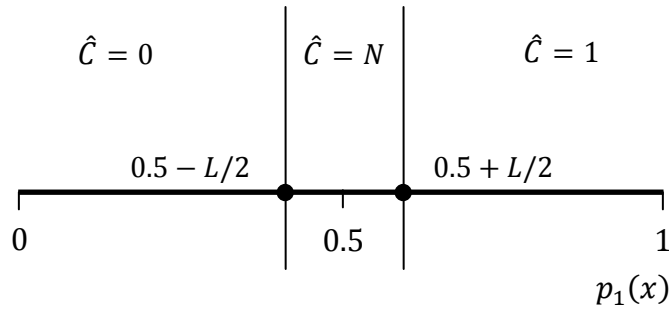


Figure 1.3: A graphical illustration of a neutral zone classifier

The NZC model was further developed by adding a third class. Similarly, denote the Bayes posterior probabilities, $p_i(x)$, which is written as $P(C = i|X = x) = \pi_i f_i(x) / \sum_{i=0}^2 \pi_i f_i(x)$. See **Figure 1.4** for a graphical view of class

assignment rules where regions A, B, and C encompass neutral zone regions between classes 0 and 1, 1 and 2, and 0 and 2, respectively. The form of the extended NZC is:

$$\hat{C}(x; L) = \begin{cases} 0 & \text{if } p_1(x) > 1 - 2p_o(x) + L \text{ and } p_1(x) < p_o(x) - L \\ 1 & \text{if } p_1(x) > p_o(x) + L \text{ and } p_1(x) > -0.5p_o(x) + 0.5(1 + L) \\ 2 & \text{if } p_1(x) < 1 - 2p_o(x) - L \text{ and } p_1(x) < -0.5p_o(x) + 0.5(1 - L) \\ N & \text{otherwise} \end{cases} \quad (1.2)$$

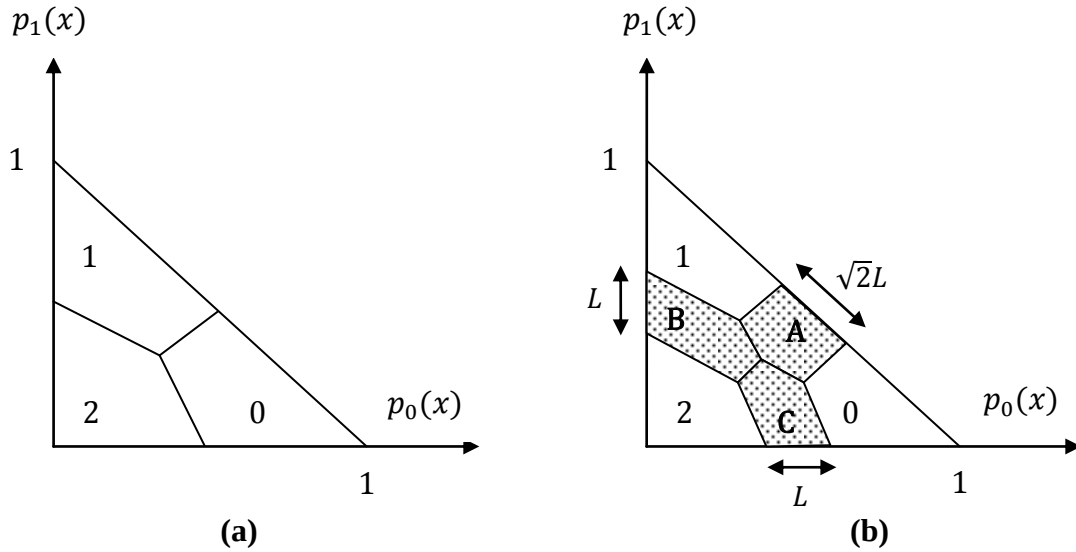


Figure 1.4: Illustrations of the class assignment regions of
(a) Three-class Bayes classifier and (b) Three-class neutral zone classifier

1.3.4: Applications of Neutral Zone Classifiers

In the OFRG study, the binary classes of binding and non-binding are classified base on the observed intensities between probes and subjects (Jeske et al. 2007). When

the third class (partial binding) was added (Yu et al. 2010), the covariate data x_{jk} of each subject j associate with probe k were obtained from the observed data o_{jk} by using the Box-Cox transformation method in the three-class model as:

$$x_{jk} = \begin{cases} \frac{o_{jk}^\lambda - 1}{\lambda} & \text{if } \lambda \neq 0 \\ \log(o_{jk}) & \text{if } \lambda = 0 \end{cases}$$

where $\lambda \in [-2, 2]$ and x_{jk} then follows a class-dependent Gaussian-mixture distribution. For these two papers, the Gaussian-mixture neutral zone classifiers (GNZC) for the two-class and three-class models, respectively, were developed and applied for each probe separately to classify objects into classes for use in the clustering process.

1.4: Structure of the Dissertation

In this dissertation, we are introducing new multi-label neutral zone classifiers to investigate their classification performance for dealing multi-label data directly. The document is organized as follows: Chapter 2 and Chapters 3 provide in detail the development of three neutral zone classification algorithms using a generalized linear mixed model (logit-based), logistic regression (naive logit-based), and Markov random field approaches, respectively. In Chapter 4, we explore how these three proposed classifiers work on simulated multi-label data generated using two different schemes based on the logit-based and Gibbs sampling algorithms. The implementations of these proposed classifiers, the results, and discussions on their performance are also demonstrated. Following that, we examine the logit-based, naive logit-based and Markov

random field neutral classifiers on a real OFRG fungi data set in Chapter 5. Finally, we summarize essential conclusions about all findings and form some follow-up ideas on future research in Chapter 6. The appendix will present further technical details.

Chapter 2

Logit-based Neutral Zone Classifier

2.1: Model Definition

The logit-based classifier is developed to quantify a potential correlation between labels for the multi-label classification problem. The two-class classification problem is considered here. Denote C_{jk} and x_{jk} as a known class label of the training data set with the values of 0 or 1 and covariate measurement, respectively, for individual j at occasion k , where $j = 1, 2 \dots N$ and $k = 0, 1, \dots K$. Note that these K occasions occur under different experimental conditions. A generalized linear mixed model with a logit-link is used to fit a binary response conditioned on both observed data and unobserved random effect (Breslow and Clayton, 1993; Wolfinger and O'Connell, 1993; Agresti, 2002; Brown and Prescott, 2006) to model the conditional probability $p_{jk} = P(C_{jk} = 1 | \beta_k, x_{jk})$, where $\beta_k = (\beta_{0k}, \beta_{1k})'$ is represented as a fixed parameter of the model. A random subject-specific intercept and subject-specific slope model is given by

$$C_{jk} | \beta_k, x_{jk} \sim_{ind.} BERN(p_{jk})$$

$$\log \frac{p_{jk}}{1 - p_{jk}} = b_{0j} + \beta_{0k} + (b_{1j} + \beta_{1k})x_{jk} \quad (2.1)$$

where the subject-specific random effects are $b_j = (b_{0j}, b_{1j})' \sim MVN(0, \Sigma)$. The illustration example of the model for three subjects with the covariate information observed twice is presented as follows:

$$j = 1 \text{ \& } k = 1: \quad \log \frac{p_{11}}{1 - p_{11}} = b_{01} + \beta_{01} + (b_{11} + \beta_{11})x_{11}$$

$$j = 1 \text{ \& } k = 2: \quad \log \frac{p_{12}}{1 - p_{12}} = b_{01} + \beta_{02} + (b_{11} + \beta_{12})x_{12}$$

$$j = 2 \text{ \& } k = 1: \quad \log \frac{p_{21}}{1 - p_{21}} = b_{02} + \beta_{01} + (b_{12} + \beta_{11})x_{21}$$

$$j = 2 \text{ \& } k = 2: \quad \log \frac{p_{22}}{1 - p_{22}} = b_{02} + \beta_{02} + (b_{12} + \beta_{12})x_{22}$$

$$j = 3 \text{ \& } k = 1: \quad \log \frac{p_{31}}{1 - p_{31}} = b_{03} + \beta_{01} + (b_{13} + \beta_{11})x_{31}$$

$$j = 3 \text{ \& } k = 2: \quad \log \frac{p_{32}}{1 - p_{32}} = b_{03} + \beta_{02} + (b_{13} + \beta_{12})x_{32}$$

2.2: Parameter Estimation

To find model parameter estimates of the logit-based model, the integrated likelihood will be approximately evaluated using numerical techniques, such as Gauss-Hermite quadrature, adaptive quadrature, or Laplace expansions due to the difficulties of not having a closed form (McCulloch and Searle, 2001). Define $\beta = \{\beta_1, \beta_2, \dots, \beta_K\}$ as a set of fixed parameters β_k of all K occasions. The likelihood function used to find parameter estimates $\hat{\beta}_k = (\hat{\beta}_{0k}, \hat{\beta}_{1k})$ and the covariance parameter estimate of the random effect Σ is expressed below:

$$L(\beta, \Sigma) =$$

$$\begin{aligned} & \iint \cdots \int \left\{ \prod_{j=1}^N \prod_{k=1}^K [P(C_{jk} = 1 | \beta_k, x_{jk})]^{C_{jk}} [P(C_{jk} = 0 | \beta_k, x_{jk})]^{1-C_{jk}} f(b_j) \right\} db_j \\ &= \iint \cdots \int \left[\prod_{j=1}^N \prod_{k=1}^K \frac{\exp\{[b_{0j} + \beta_{0k} + (b_{1j} + \beta_{1k})x_{jk}]C_{jk}\}}{1 + \exp[b_{0j} + \beta_{0k} + (b_{1j} + \beta_{1k})x_{jk}]} f(b_j) \right] db_j \\ &= \prod_{j=1}^N \left\{ \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left[\prod_{k=1}^K \frac{\exp\{[b_{0j} + \beta_{0k} + (b_{1j} + \beta_{1k})x_{jk}]C_{jk}\}}{1 + \exp[b_{0j} + \beta_{0k} + (b_{1j} + \beta_{1k})x_{jk}]} f(b_j) db_j \right] \right\} \end{aligned}$$

Figure 2.1 illustrates schematic plots of the link function in equation (2.1) versus covariate measurements x varying from -5 to 5. Fixed parameters are chosen in three different types: (1) both intercept and slope equal 0; (2) positive slope; and (3) negative slope. In each plot, the link function is shown for a sample of 50 subjects scatter around a thick dark line, which represents the average value computed based on fixed parameters only. The subject-specific random effects for this sample are selected from a multivariate distribution, $MVN(0, \Sigma)$, where the components of Σ are $\sigma_0^2 = 5$, $\sigma_1^2 = 1.5$ and $\sigma_{01} = 2$.

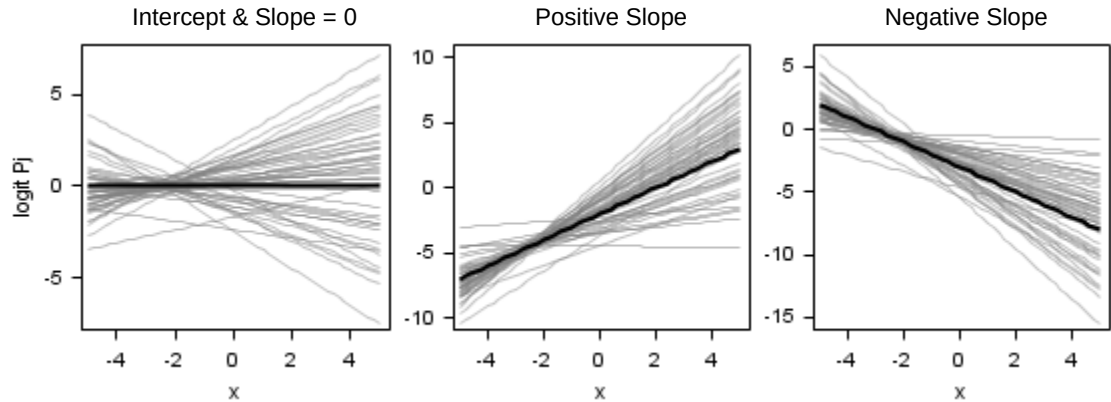


Figure 2.1: Schematic plot of link function (2.1) vs. covariate with different fixed parameters

2.3: Classifier Definition

For a new subject outside the training data set, the marginal expected value of the probability that $C_k = 1$ associated with the covariate measurement x_k is

$$E(p_k) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} P(C_k = 1 | \beta_k, x_k) f(b) db \quad (2.2)$$

where $f(b) = (2\pi)^{-1} |\Sigma|^{-1/2} \exp(-\frac{1}{2} b^T \Sigma^{-1} b)$ and

$$P(C_k = 1 | \beta_k, x_k) = \frac{\exp[b_0 + \beta_{0k} + (b_1 + \beta_{1k})x_k]}{1 + \exp[b_0 + \beta_{0k} + (b_1 + \beta_{1k})x_k]}$$

Similar to (1.1), the form of the multi-label logit-based neutral zone classifier (LNZC) is:

$$\hat{C}_k(x_k; L_k) = \begin{cases} 0 & \text{if } \hat{E}(p_k) < 0.5 - L_k/2 \\ 1 & \text{if } \hat{E}(p_k) > 0.5 + L_k/2 \\ N & \text{otherwise} \end{cases} \quad (2.3)$$

where $\hat{E}(p_k)$ is obtained from (2.2) after replacing β_k and Σ by their estimated values $\hat{\beta}_k$ and $\hat{\Sigma}$. Here, $L_k \in [0, 1]$. The algorithm for finding the neutral zone limit L_k for the covariate measured at k^{th} occasion is explained in detail in section 2.3.1.

Theorem:

The estimated expected probability $\hat{E}(p_k) = P(C_k = 1 | \hat{\beta}_k, \hat{\Sigma}, x_k)$ that is used to derive the logit-base neutral zone classifier can be interpreted as a mean of marginal Bayesian predictive distribution for C_k under a degenerate prior.

Proof: See Appendix A

2.4: Neutral Zone Boundary

Define the hard and soft error rates as follows:

$$P_H(L_k) = P(\hat{C}_k = 1|C_k = 0) + P(\hat{C}_k = 0|C_k = 1)$$

$$P_S(L_k) = P(\hat{C}_k = N|C_k = 0) + P(\hat{C}_k = N|C_k = 1)$$

A misclassification table for fixed neutral zone limits $L_k \in [0, 1]$ can be computed using M-fold cross-validation (Hastie, et al., 2001). **Table 2.1** shows the notation for the two misclassification rates.

Predicted Population	True Population	
	0	1
0	0	$P(\hat{C}_k = 0 C_k = 1)$
1	$P(\hat{C}_k = 1 C_k = 0)$	0
N	$P(\hat{C}_k = N C_k = 0)$	$P(\hat{C}_k = N C_k = 1)$

Table 2.1: A misclassification table for any fixed $L_k \in [0, 1]$

The total misclassification error rate for each class label k , which is the sum of $P_H(L_k)$ and $P_S(L_k)$, can be computed using the misclassification tables presented above. One should keep in mind that the neutral zone classifier is proposed to award a high penalty for hard errors and a lower penalty for soft errors. Hence, the algorithm to find the limit L_k to build LNZN considers two situations: (1) if the two class labels 0 and 1 are clearly separated (e.g., the hard error rate is small enough such that it is less than α), the neutral zone region does not exist, which means that the neutral zone limit is $L_k^* = 0$; (2) if the first condition is not satisfied, find the value c where $c = P_H(L_k) = \gamma P_S(L_k)$. The

constant γ is given based on historical data or determined subjectively, depending on the application. Note that the hard error rate $P_H(L_k)$ is a non-increasing function of L_k , whereas the soft error rate, $P_S(L_k)$, is a non-decreasing function of L_k . It is possible for $P_H(L_k)$ to stay constant while $P_S(L_k)$ is increasing. In this case, a higher value of L_k does not contribute to the accuracy of the logit-based neutral zone classifier since it increases the soft error rate without reducing the hard error rate. Define the set $S = \{L \in [0, 1]: P_H(L) = c \text{ \& } \gamma P_S(L_k) \leq c\}$. For case (2), the smallest value in S is chosen as the resulting neutral zone limit L_k^* . The full procedure is expressed as follows. Specify α and γ . Then,

Case 1: If $P_H(0) \leq \alpha$, $L_k^* = 0$.

Case 2: Define c to be the value when $c = P_H(L_k) = \gamma P_S(L_k)$.

Select $L_k^* = \min(S)$ where $S = \{L_k \in [0, 1]: P_H(L_k) = c \text{ \& } \gamma P_S(L_k) \leq c\}$.

For instance, see **Figure 2.2** for the illustration of how to implement the algorithm. This graph shows the hard and soft error rates versus the neutral zone limit L in the range from 0 to 1 with the step size of 0.001. We pick a threshold value of $\alpha = 5\%$. At $L = 0$, since the hard error rate is 13.08% and bigger than α , the neutral zone for this example does exist. This situation belongs to Case 2 of the proposed algorithm. The hard error rate decreases to 0%, whereas the soft error rates increases from 0% to 100% as L varies from 0 to 1.

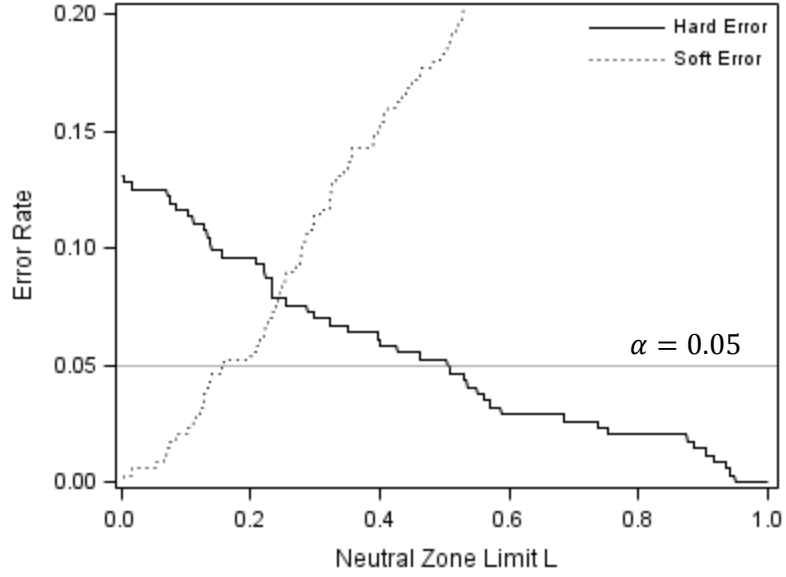


Figure 2.2: Graph of misclassification error rate vs. neutral zone limit L

Figure 2.3 zooms in on a portion of **Figure 2.2** near the intersection of the solid and broken lines, which represent the hard error rate, $P_H(L)$, and the soft error rate, $P_S(L)$. Assume $\gamma = 1$ for this study. The defined value c occurs at 7.8% when $P_H(L)$ meets $P_S(L)$. Therefore, the set S is $[0.232, 0.243]$, where $S = \{L \in [0, 1]: P_H(L) = 7.8\% \}$. Whereas the hard error rate, $P_H(L)$, stays constant from 0.232 to 0.243, the soft error rate, $P_S(L)$, increases from 7.27% to 7.85%. The chosen neutral zone limit L^* is 0.232. This sliding-back step helps to reduce the soft error rate when the hard error rate does not improve.

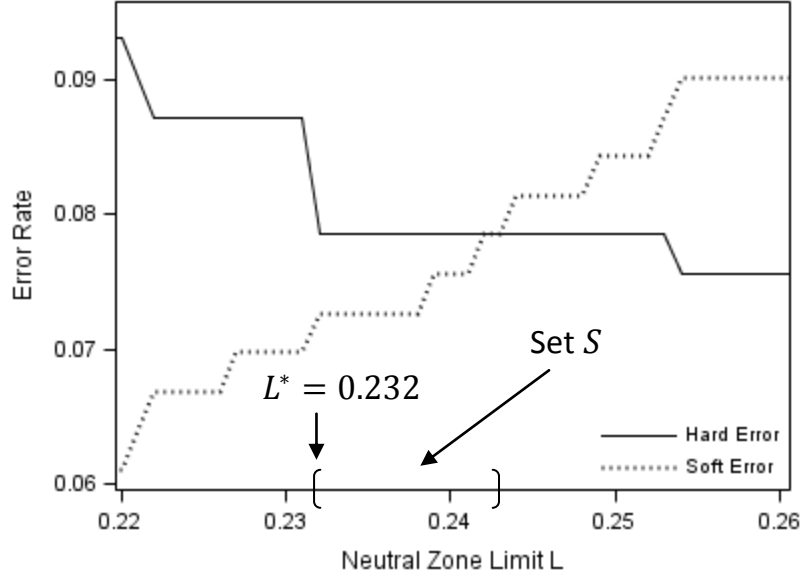


Figure 2.3: Magnified version of neighboring location of the neutral zone L^*

Together, the procedure for constructing the LNZC is illustrated in **Figure 2.4** using M-fold cross-validation ($M = 2$) to find the neutral zone limits L_k^* of K occasions.

2.5: Naive Logit-based Neutral Zone Classifier

The naive neutral zone logit-based classifier (nLNZC) is constructed using logistic regression (Hosmer & Lemeshow, 2000; Agresti, 2002; Hilbe, 2009) with the naive assumption of independent labels. Since logistic regression model is a part of the generalized linear mixed model, nLNZC is formed same as LNZC except the involvement of a possible random effect. The logistic regression model is reduced from (2.1) as follows:

$$C_{jk}|\beta_k, x_{jk} \sim_{ind.} BERN(p_{jk})$$

$$\log \frac{p_{jk}}{1 - p_{jk}} = \beta_{0k} + \beta_{1k}x_{jk} \quad (2.4)$$

The unknown parameters β_k are then estimated by maximum likelihood using this likelihood function:

$$L(\beta) = \prod_{j=1}^N \prod_{k=1}^K [P(C_{jk} = 1|\beta_k, x_{jk})]^{C_{jk}} [P(C_{jk} = 0|\beta_k, x_{jk})]^{1-C_{jk}} \quad (2.5)$$

The probability that $C_k = 1$ associated with the covariate measurement x_k at occasion k for any new subject is $P(C_k = 1|\beta_k, x_k) = \exp(\beta_{0k} + \beta_{1k}x_k)/[1 + \exp(\beta_{0k} + \beta_{1k}x_k)]$.

Denote $p_k = P(C_k = 1|\beta_k, x_k)$. The classification rule of the naive neutral zone classifier (nLNZC) is:

$$\hat{C}_k(x_k; L_k) = \begin{cases} 0 & \text{if } \hat{p}_k < 0.5 - L_k/2 \\ 1 & \text{if } \hat{p}_k > 0.5 + L_k/2 \\ N & \text{otherwise} \end{cases} \quad (2.6)$$

The neutral zone limit L_k in (2.6) will be found by following the algorithm described in section 2.4.

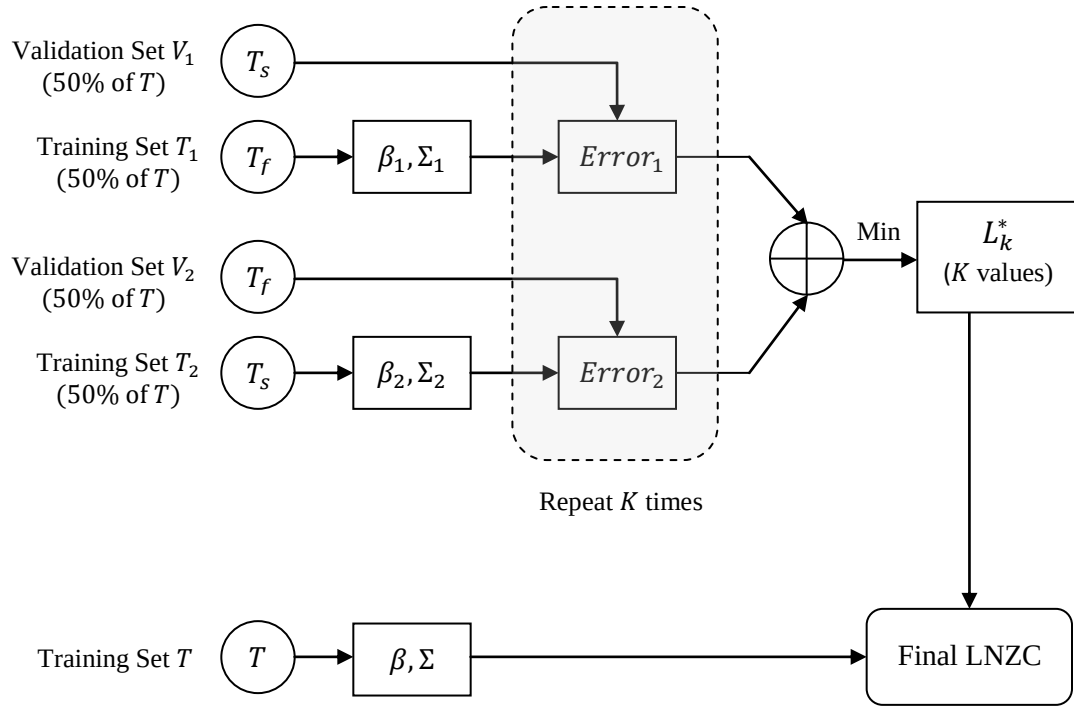


Figure 2.4: Flowchart of the procedure for constructing the logit-based neutral zone classifier using two-fold cross-validation to find L_k^*

Chapter 3

Markov Random Field Neutral Zone Classifier

A Markov random field or Markov network (Besag, 1974; Jordan, 1999; Kinderman and Snell, 1980; Li, 2009; Pearl, 1988), part of the family of probabilistic graphical models, is employed as an alternative way to model dependency for multi-label data like Ghamrawi and McCallum (2005). In this chapter, we will briefly review the Markov random field and propose a Markov random field neutral zone classifier.

3.1: Markov Random Field

Let $V = \{V_1, V_2, \dots, V_K\}$ represent a set of nodes associated with a set of binary random variables $C = \{C_1, C_2, \dots, C_K\}$. Let $x = \{x_1, x_2, \dots, x_K\}$ denote corresponding covariates associated with the nodes. The binary variables are introduced for simplicity, but the notation will generally be proper for discrete variables and also extended for continuous variables. The notation $P(C = l)$ is defined as the joint probability, where $l = \{l_1, l_2, \dots, l_K\}$ and each l_k takes a value of 0 or 1. Define $\mathbb{C}$ as the notation of a set that contains all possible configurations of l . The symbol E stands for a set of undirected edges that connect and depict influences between nodes. A set of nodes, N_k , is defined as

the neighboring nodes of $V_k \in V$ if every node in N_k ties to V_k by an edge in E . The set $N = \{N_k | \forall V_k \in V\}$ is called the neighborhood system of V . Define an undirected graph, $G = (V, E)$, which is constructed by a set of nodes, V , and edges, E .

Definition 1: Markov Random Fields (MRF)

A Markov random field is specified on V with respect to a neighborhood system, N , of an undirected graph, $G = (V, E)$, if, and only if, the following two conditions are satisfied

$$\text{Positivity:} \quad P(C = l) > 0 \quad \forall l \in \mathbb{C} \quad (3.1)$$

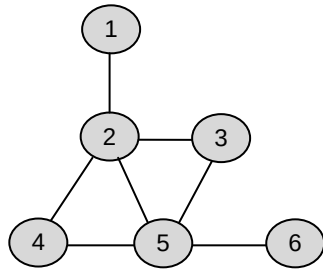
$$\text{Markovianity:} \quad P(C_k = l_k | C_{V-\{V_k\}}) = P(C_k = l_k | C_{N_k}) \quad (3.2)$$

where $C_{V-\{V_k\}}$ and C_{N_k} , respectively, denote the set random variables associated with the nodes in $V - \{V_k\}$ and at those nodes neighboring V_k . The Markovianity property illustrates that the random variable at each node is conditionally independent of the nodes outside its neighborhood.

Clique terminology is declared in the context of the Markov random field because it plays an important role when the joint probability $P(C = l)$ in the positivity property (3.1) is specified in terms of factorization of the distribution. Given an undirected graph, G , represent $c(G)$ as a set of cliques, where each clique $c \in c(G)$ is constructed by a subset of nodes, $V_c \in V$, such that each node is connected to every other nodes in V_c by the edges in E . A function, $\phi_{V_c}(l)$, is defined as a potential function or clique potential, over a clique c , where the notation l of a complete set of binary class labels implies l_c as being extracted based on a set of nodes in V_c . Each individual node also forms a clique

(called a single clique). Any clique that cannot absorb another node is defined as a maximal clique, and a maximum clique is a clique that contains the largest possible number of nodes. An MRF is said to be isotropic if the function $\phi_{V_c}(l)$ is independent of the orientation of V_c ; and an MRF is homogeneous when $P(C_k = l_k | C_{N_k}) = P(C_{k'} = l_{k'} | C_{N_{k'}})$ if $l_k = l_{k'}$ and $l_{N_k} = l_{N_{k'}}$.

An example of undirected graph and clique terminology is illustrated in **Figure 3.1**. In this figure, a graph G is constructed of a set of six nodes $V = \{V_1, \dots, V_6\}$ and seven edges. The six single-node cliques are $\{V_1\}$, $\{V_2\}$, $\{V_3\}$, $\{V_4\}$, $\{V_5\}$ and $\{V_6\}$. This graph forms seven pair-node cliques $\{V_1, V_2\}$, $\{V_2, V_3\}$, $\{V_2, V_4\}$, $\{V_2, V_5\}$, $\{V_3, V_5\}$, $\{V_4, V_5\}$ and $\{V_5, V_6\}$. The two triangle shapes form two triple-node cliques $\{V_2, V_4, V_5\}$ and $\{V_2, V_3, V_5\}$. By satisfying the condition of not being able to add any more nodes, both the triple-node cliques and two of the pair-node cliques $\{V_1, V_2\}$ and $\{V_5, V_6\}$ are called maximal cliques. The two triple-node cliques are maximum cliques in this example because they have a maximum number of nodes compare to the other cliques.



Single-node cliques:

$\{V_1\}, \{V_2\}, \{V_3\}, \{V_4\}, \{V_5\}, \{V_6\}$

Pair-node cliques: $\{V_1, V_2\}, \{V_2, V_3\}, \{V_2, V_4\},$
 $\{V_2, V_5\}, \{V_3, V_5\}, \{V_4, V_5\}$ and $\{V_5, V_6\}$

Triple-node cliques: $\{V_2, V_4, V_5\}, \{V_2, V_3, V_5\}$

Maximal cliques: $\{V_1, V_2\}, \{V_5, V_6\},$
 $\{V_2, V_4, V_5\}, \{V_2, V_3, V_5\}$

Figure 3.1: An undirected graph is constructed by six nodes and seven edges

Definition 2: Gibbs Random Field (GRF) or Gibbs Distribution

A set of random variables, C , follows a Gibbs distribution on V with respect to a neighborhood system, N , if $P(C = l)$ has the form

$$P(C = l) = \frac{1}{Z} \exp \left[-\frac{1}{T} U(l) \right] \quad (3.3)$$

where the normalization factor or partition function Z is written as

$$Z = \sum_{l \in \mathbb{C}} \exp \left[-\frac{1}{T} U(l) \right] \quad (3.4)$$

and the energy function is $U(l)$ of a configuration l is:

$$U(l) = \sum_{c \in \mathcal{C}(G)} \phi_{V_c}(l) \quad (3.5)$$

where the constant T is assumed to be 1 for simplicity, unless otherwise stated. The most frequent form of the energy function $U(l)$ is the case when the clique potentials are defined for up to two-node cliques (Besag, 1974; Li, 2009). In this context, with single-node and pair-node clique potentials denoted by $\phi_{\{V_k\}}(l_k)$ and $\phi_{\{V_k, V_{k'}\}}(l_k, l_{k'})$ respectively, the energy function is:

$$U(l) = \sum_{k=1}^K \phi_{\{V_k\}}(l_k) + \sum_{k=1}^K \sum_{V_{k'} \in N_k} \phi_{\{V_k, V_{k'}\}}(l_k, l_{k'}) \quad (3.6)$$

The Gibbs distribution is conceptually equivalent to an MRF based on the following theorem:

Markov-Gibbs Equivalence

Hammersley-Clifford Theorem (1971):

A set of random variables, C , is a Markov random field on V with respect to N if, and only if, C follows a Gibbs distribution on V , with respect to the same neighborhood system N .

Proof: See Appendix B

This theorem is very valuable because it provides a method for finding parameter estimates of an MRF via a GRF using the conditional probability $P(C_k = l_k | C_{N_k})$ to find the pseudo-likelihood of the Markov network. In image application with a large lattice limit, the pseudo-likelihood method was proven to provide consistent estimates of parameters with those of the true likelihood function from the probability function (Geman and Graffigne, 1987). The pseudo-likelihood of an MRF is written as:

$$P_{PL}(C = l) = \prod_{k=1}^K P(C_k = l_k | C_{N_k}) \quad (3.7)$$

Auto-model for Binary Random Variables

The MRF model for binary random variables is also called the auto-logistic model (Besag, 1974; Li, 2009). The energy function written in (3.6) becomes:

$$U(l) = \sum_{k=1}^K \alpha_k l_k + \sum_{k=1}^K \sum_{V_{k'} \in N_k} \beta_{kk'} l_k l_{k'} \quad (3.8)$$

where the coefficients $\beta_{kk'}$ are defined as interactions between variables. The conditional probability for the auto-logistic model at each node k is

$$\begin{aligned} P(C_k = l_k | C_{N_k}) &= \frac{\exp(\alpha_k l_k + \sum_{V_{k'} \in N_k} \beta_{kk'} l_k l_{k'})}{\sum_{l_k \in \{0,1\}} \exp(\alpha_k l_k + \sum_{V_{k'} \in N_k} \beta_{kk'} l_k l_{k'})} \\ &= \frac{\exp(\alpha_k l_k + \sum_{V_{k'} \in N_k} \beta_{kk'} l_k l_{k'})}{1 + \exp(\alpha_k + \sum_{V_{k'} \in N_k} \beta_{kk'} l_{k'})} \end{aligned} \quad (3.9)$$

This subclass of the MRF is broadly applied in many applications due to its simplicity.

3.2: Markov Random Field Neutral Zone Classifier

A Markov random field is utilized to develop a Markov random field neutral zone classifier (MNZC) for multi-label data using three steps: 1) graph construction; 2) parameter estimation; and 3) find a neutral zone boundary.

3.2.1: Graph Construction

Denote x to be a set of covariates associated with a set of binary random variables, V . Denote a Pearson correlation coefficient $r_{kk'}$ as a pair-wise relationship between x_k and $x_{k'}$, either provided from a historical information or estimated from available training data. Assume that larger positive correlation value implies a higher chance that these two nodes have the same value of C . Conversely, larger negative correlation implies a higher chance of opposite values of C . Define r^* as a cut off value for $|r_{kk'}|$. If $|r_{kk'}|$ is very small, say less than r^* , we assume the influence between these

two nodes can be neglected and an edge connecting them is not present in the graph. The undirected graph of **Figure 3.1** is used to demonstrate the contribution of pair-wise correlation coefficients when constructing the connections among nodes on that graph (see **Figure 3.2**).

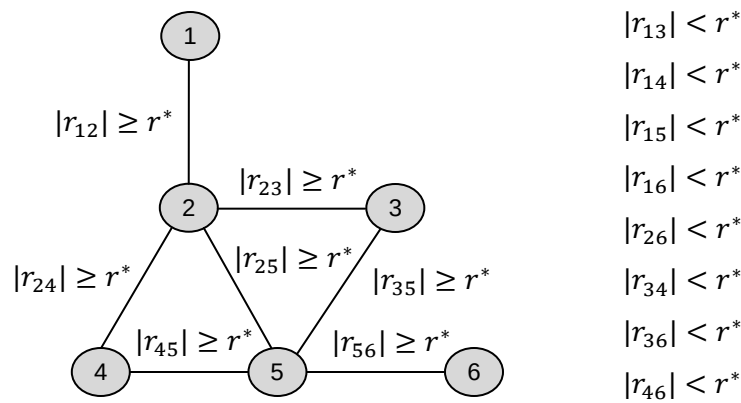


Figure 3.2: Connections between nodes of the graph are constructed depending on Pearson correlation coefficients

3.2.2: Parameter Estimation

The auto-logistic model is used when building the classifier. The correlation coefficients between variables are added to pair-node clique potentials to account for the weight of influence on each other.

Assume that neighboring nodes with approximate equal correlation coefficients will provide the same influence on the main node. Hence, pooled information is used for neighboring nodes who follow the same category of correlation coefficient, i.e., low, moderate and high correlation groups with $0.1 \leq |r_{kk'}| < 0.4$, $0.4 \leq |r_{kk'}| < 0.7$ and

$|r_{kk'}| \geq 0.7$, respectively. For this setting, the value of r^* is 0.1. The energy function in equation (3.8) with $\alpha_k = \beta_{0k} + \beta_{1k}x_k$ is of the form:

$$U(C = l|x) = \sum_{k=1}^K (\beta_{0k} + \beta_{1k}x_k)l_k + \sum_{k=1}^K f_{N_k}l_k \quad (3.10)$$

where

$$f_{N_k} = \beta_{k,1} \sum_{\substack{V_{k'} \in N_k \\ 0.1 \leq |r_{kk'}| < 0.4}} r_{kk'} C_{k'} + \beta_{k,2} \sum_{\substack{V_{k'} \in N_k \\ 0.4 \leq |r_{kk'}| < 0.7}} r_{kk'} C_{k'} + \beta_{k,3} \sum_{\substack{V_{k'} \in N_k \\ |r_{kk'}| \geq 0.7}} r_{kk'} C_{k'} \quad (3.11)$$

Alternatively, f_{N_k} could be defined as considering equal weights for all neighboring nodes as follows:

$$f_{N_k} = \beta_{k,1} \sum_{\substack{V_{k'} \in N_k \\ 0.1 \leq |r_{kk'}| < 0.4}} C_{k'} + \beta_{k,2} \sum_{\substack{V_{k'} \in N_k \\ 0.4 \leq |r_{kk'}| < 0.7}} C_{k'} + \beta_{k,3} \sum_{\substack{V_{k'} \in N_k \\ |r_{kk'}| \geq 0.7}} C_{k'}$$

The corresponding conditional probability is expressed as:

$$\begin{aligned} P(C_k = l_k | x_k, C_{N_k}) &= \frac{\exp[(\beta_{0k} + \beta_{1k}x_k + f_{N_k})l_k]}{\sum_{l_k \in \{0,1\}} \exp[(\beta_{0k} + \beta_{1k}x_k + f_{N_k})l_k]} \\ &= \frac{\exp[(\beta_{0k} + \beta_{1k}x_k + f_{N_k})l_k]}{1 + \exp(\beta_{0k} + \beta_{1k}x_k + f_{N_k})} \end{aligned} \quad (3.12)$$

Substitute the conditional probability $P(C_k = l_k | x_k, C_{N_k})$ in expression (3.12) into the pseudo-likelihood function $P(C_k = l_k | x_k, C_{N_k})$ in (3.7) to find parameter estimates

$\hat{\beta} = \{\hat{\beta}_{0k}, \hat{\beta}_{1k}, \hat{\beta}_{k,1}, \hat{\beta}_{k,2}, \hat{\beta}_{k,3}\}$ of this Markov random field model.

3.3: Neutral Zone Decision Algorithm

3.3.1: Algorithm

In the framework of MRF, the conditional probability $P(C_k = l_k | C_{N_k})$ is only obtained if its neighboring class labels are also available. In other words, the process of searching for neutral zone limits L_k requires the simultaneous participation of all nodes in the Markov network. The procedure for making multi-label neutral zone prediction for MNZC follow these steps:

Approach 1: Threshold Rule

1. Find the neutral zone limit L_k from training data set:
 - a) Compute the conditional probability $p_{jk} = P(C_{jk} = 1 | C_{N_k})$ using (3.12) and predict class labels for all subjects in the training data set at each node k using the following rule:
$$C_k(x_k; L_k) = \begin{cases} 0 & \text{if } p_k < 0.5 - L_k/2 \\ 1 & \text{if } p_k > 0.5 + L_k/2 \\ N & \text{otherwise} \end{cases} \quad (3.13)$$
 - b) Compute the hard and soft error rates $P_H(L_k)$ and $P_S(L_k)$ of each node k using M-fold cross-validation to find the neutral zone limit $L_k^* = \{L_k \in [0, 1]\}$ by following the algorithm described in section 2.4.
2. Assign class labels on a new subject:
 - a) Compute the pseudo-likelihoods

$$P = \left\{ f_n \in \mathbb{C}: P_{f_n}(C|x) = \prod_{k=1}^K P(C_k | C_{N_k}), n = 1 \dots 2^K \right\}$$

where the conditional probability $P(C_k|C_{N_k})$ is computed using (3.12).

- b) Select the optimal configuration $f_n^* \in \mathbb{C}$ that maximizes the pseudo-likelihood P .
- c) Compute the conditional probability $p_k = P(C_k = 1|C_{N_k})$ using labels of C_{N_k} from f_n^* , then follow the rule in (3.13) to assign class labels for the new subject.

Approach 2: Voting Scheme

1. Compute the pseudo-likelihood P for any new subject as step (2.a) of the threshold approach.
2. Denote S^* as a subset of $\mathbb{C}$, which contains the best selected configurations. The prediction of 0 or 1 will be made for each node k if there is an agreement among all of these best possible configurations. Otherwise, the neutral zone label N will be given. The voting rule will be expressed as below:

$$C_k = \begin{cases} 0 & \text{if } C_k = 0 \quad \forall C_k \in S^* \\ 1 & \text{if } C_k = 1 \quad \forall C_k \in S^* \\ N & \text{otherwise} \end{cases}$$

3.3.2: Implementation Examples

Example 1: Find the set of optimal configurations for the voting rule

The optimal set S^* can be determined by examining a Scree plot of pseudo-likelihood and looking for a big gap or a break in slope. Alternatively, for a large-scale sample, a simple and automatic procedure could be used to find S^* . Define $P_{max} =$

$\max(P)$. All configurations in S^* are extracted from $\mathbb{C}$, whose pseudo-likelihood is within a threshold δ below the maximum value, P_{max} , as in the following:

$$S^* = \{f_n \in \mathbb{C}: |P_{f_n}(C|x) - P_{max}| \leq \delta, n = 1 \dots 2^K\}$$

Figure 3.3 displays the size of the pseudo-likelihood of all $2^5 = 32$ possible configurations when $K = 5$. The procedure for finding the optimal set S^* manually by examining a Scree plot is illustrated from the left plot. The top four configurations are selected from the pool because they are clearly discriminated from the rest. Further, the plot on the right side demonstrates the automatic approach with a threshold of $\delta = 0.05$ for this study. Again, the top four configurations are also extracted because the second, third and fourth pseudo-likelihoods are 0.55, 0.51 and 0.49, respectively, are below the maximum one, which is 0.57, within a magnitude of 0.05.

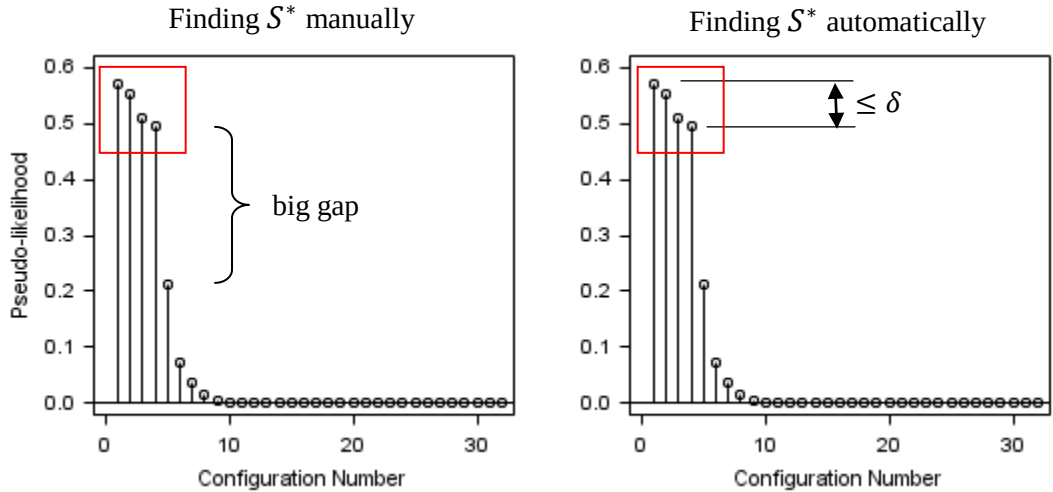


Figure 3.3: Scree plots of 32 configurations when $K = 5$

Example 2: Find neutral zone limits and assign class labels for a new subject using the threshold approach.

Figure 3.4 is a toy example of the Markov random field used to show how to find neutral zone limits, L_k , using the threshold rule.

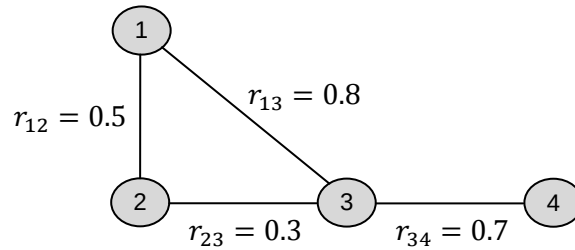


Figure 3.4: A Markov random field constructed by four nodes and four edges

Pearson pair-wise correlation coefficients among nodes are given as $r_{12} = 0.5$, $r_{13} = 0.8$, $r_{23} = 0.3$ and $r_{34} = 0.70$. The parameter estimates obtained from the training data are $\hat{\beta}_{01}$, $\hat{\beta}_{11}$, $\hat{\beta}_{1,2}$ and $\hat{\beta}_{1,3}$ for node 1; $\hat{\beta}_{02}$, $\hat{\beta}_{12}$, $\hat{\beta}_{2,1}$ and $\hat{\beta}_{2,2}$ for node 2; $\hat{\beta}_{03}$, $\hat{\beta}_{13}$, $\hat{\beta}_{3,1}$ and $\hat{\beta}_{3,3}$ for node 3 and; $\hat{\beta}_{04}$, $\hat{\beta}_{14}$ and $\hat{\beta}_{4,3}$ for node 4. In this situation, the model parameters $\beta_{1,1}$, $\beta_{2,3}$, $\beta_{3,2}$, $\beta_{4,1}$ and $\beta_{4,2}$ do not exist. See **Figure 3.5** for a graphical allocation of these parameter estimates. These estimates are used find the hard and soft error rates to find K neutral zone limits based on the conditional probability $p_k = P(C_k = 1 | C_{N_k})$ of known-label subjects in the training data set.

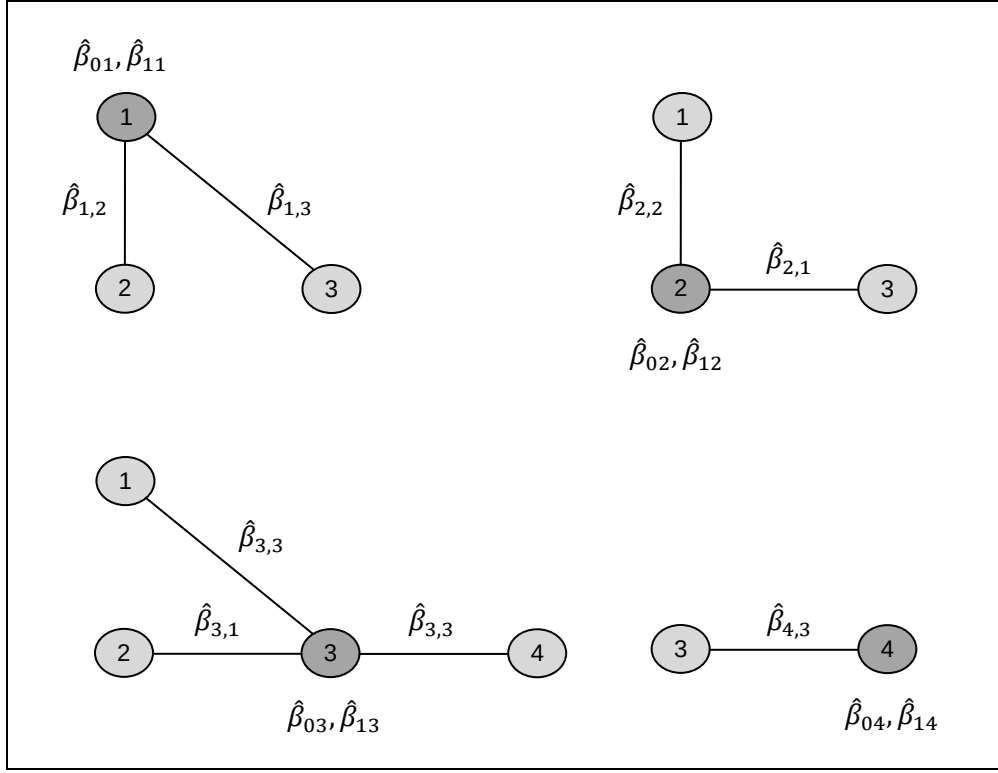


Figure 3.5: Graphical allocation illustration of parameter estimates of a four-node Markov random field

When $K = 4$, a vector of class labels of any new subject belongs to a set $\mathbb{C}$ that contains sixteen elements: $f_1 = 0000$, $f_2 = 0001$, $f_3 = 0010$, $f_4 = 0011$, $f_5 = 0100$, $f_6 = 0101$, $f_7 = 0110$, $f_8 = 0111$, $f_9 = 1000$, $f_{10} = 1001$, $f_{11} = 1010$, $f_{12} = 1011$, $f_{13} = 1100$, $f_{14} = 1101$, $f_{15} = 1110$ and $f_{16} = 1111$. To make label predictions on a new subject, firstly, we need to compute the conditional probabilities $P(C_k|C_{N_k})$ as below:

$$P(C_1 = 1|C_{N_1}) = \frac{\exp(\hat{\beta}_{01} + \hat{\beta}_{11}x_1 + \hat{\beta}_{1,2}r_{12}C_2 + \hat{\beta}_{1,3}r_{13}C_3)}{1 + \exp(\hat{\beta}_{01} + \hat{\beta}_{11}x_1 + \hat{\beta}_{1,2}r_{12}C_2 + \hat{\beta}_{1,3}r_{13}C_3)}$$

$$\begin{aligned}
P(C_1 = 0|C_{N_1}) &= \frac{1}{1 + \exp(\hat{\beta}_{01} + \hat{\beta}_{11}x_1 + \hat{\beta}_{1,2}r_{12}C_2 + \hat{\beta}_{1,3}r_{13}C_3)} \\
P(C_2 = 1|C_{N_2}) &= \frac{\exp(\hat{\beta}_{02} + \hat{\beta}_{12}x_2 + \hat{\beta}_{2,1}r_{23}C_3 + \hat{\beta}_{2,2}r_{12}C_1)}{1 + \exp(\hat{\beta}_{02} + \hat{\beta}_{12}x_2 + \hat{\beta}_{2,1}r_{23}C_3 + \hat{\beta}_{2,2}r_{12}C_1)} \\
P(C_2 = 0|C_{N_2}) &= \frac{1}{1 + \exp(\hat{\beta}_{02} + \hat{\beta}_{12}x_2 + \hat{\beta}_{2,1}r_{23}C_3 + \hat{\beta}_{2,2}r_{12}C_1)} \\
P(C_3 = 1|C_{N_3}) &= \frac{\exp[\hat{\beta}_{03} + \hat{\beta}_{13}x_3 + \hat{\beta}_{3,1}r_{23}C_2 + \hat{\beta}_{3,3}(r_{13}C_1 + r_{34}C_4)]}{1 + \exp[\hat{\beta}_{03} + \hat{\beta}_{13}x_3 + \hat{\beta}_{3,1}r_{23}C_2 + \hat{\beta}_{3,3}(r_{13}C_1 + r_{34}C_4)]} \\
P(C_3 = 0|C_{N_3}) &= \frac{1}{1 + \exp[\hat{\beta}_{03} + \hat{\beta}_{13}x_3 + \hat{\beta}_{3,1}r_{23}C_2 + \hat{\beta}_{3,3}(r_{13}C_1 + r_{34}C_4)]} \\
P(C_4 = 1|C_{N_4}) &= \frac{\exp(\hat{\beta}_{04} + \hat{\beta}_{14}x_4 + \hat{\beta}_{4,3}r_{34}C_3)}{1 + \exp(\hat{\beta}_{04} + \hat{\beta}_{14}x_4 + \hat{\beta}_{4,3}r_{34}C_3)} \\
P(C_4 = 0|C_{N_4}) &= \frac{1}{1 + \exp(\hat{\beta}_{04} + \hat{\beta}_{14}x_4 + \hat{\beta}_{4,3}r_{34}C_3)}
\end{aligned}$$

Next, the corresponding pseudo-likelihood $P = \{P_{f_1}, P_{f_2}, P_{f_3}, P_{f_4}, \dots, P_{f_{13}}, P_{f_{14}}, P_{f_{15}}, P_{f_{16}}\}$

is determined by these calculations:

$$\begin{aligned}
P_{f_1}(C|x) &= P(C_1 = 0|C_{N_1}) \times P(C_2 = 0|C_{N_2}) \times P(C_3 = 0|C_{N_3}) \times P(C_4 = 0|C_{N_4}) \\
P_{f_2}(C|x) &= P(C_1 = 0|C_{N_1}) \times P(C_2 = 0|C_{N_2}) \times P(C_3 = 0|C_{N_3}) \times P(C_4 = 1|C_{N_4}) \\
P_{f_3}(C|x) &= P(C_1 = 0|C_{N_1}) \times P(C_2 = 0|C_{N_2}) \times P(C_3 = 1|C_{N_3}) \times P(C_4 = 0|C_{N_4}) \\
P_{f_4}(C|x) &= P(C_1 = 0|C_{N_1}) \times P(C_2 = 0|C_{N_2}) \times P(C_3 = 1|C_{N_3}) \times P(C_4 = 1|C_{N_4}) \\
P_{f_5}(C|x) &= P(C_1 = 0|C_{N_1}) \times P(C_2 = 1|C_{N_2}) \times P(C_3 = 0|C_{N_3}) \times P(C_4 = 0|C_{N_4}) \\
P_{f_6}(C|x) &= P(C_1 = 0|C_{N_1}) \times P(C_2 = 1|C_{N_2}) \times P(C_3 = 0|C_{N_3}) \times P(C_4 = 1|C_{N_4}) \\
P_{f_7}(C|x) &= P(C_1 = 0|C_{N_1}) \times P(C_2 = 1|C_{N_2}) \times P(C_3 = 1|C_{N_3}) \times P(C_4 = 0|C_{N_4})
\end{aligned}$$

$$P_{f_8}(C|x) = P(C_1 = 0|C_{N_1}) \times P(C_2 = 1|C_{N_2}) \times P(C_3 = 1|C_{N_3}) \times P(C_4 = 1|C_{N_4})$$

$$P_{f_9}(C|x) = P(C_1 = 1|C_{N_1}) \times P(C_2 = 0|C_{N_2}) \times P(C_3 = 0|C_{N_3}) \times P(C_4 = 0|C_{N_4})$$

$$P_{f_{10}}(C|x) = P(C_1 = 1|C_{N_1}) \times P(C_2 = 0|C_{N_2}) \times P(C_3 = 0|C_{N_3}) \times P(C_4 = 1|C_{N_4})$$

$$P_{f_{11}}(C|x) = P(C_1 = 1|C_{N_1}) \times P(C_2 = 0|C_{N_2}) \times P(C_3 = 1|C_{N_3}) \times P(C_4 = 0|C_{N_4})$$

$$P_{f_{12}}(C|x) = P(C_1 = 1|C_{N_1}) \times P(C_2 = 0|C_{N_2}) \times P(C_3 = 1|C_{N_3}) \times P(C_4 = 1|C_{N_4})$$

$$P_{f_{13}}(C|x) = P(C_1 = 1|C_{N_1}) \times P(C_2 = 1|C_{N_2}) \times P(C_3 = 0|C_{N_3}) \times P(C_4 = 0|C_{N_4})$$

$$P_{f_{14}}(C|x) = P(C_1 = 1|C_{N_1}) \times P(C_2 = 1|C_{N_2}) \times P(C_3 = 0|C_{N_3}) \times P(C_4 = 1|C_{N_4})$$

$$P_{f_{15}}(C|x) = P(C_1 = 1|C_{N_1}) \times P(C_2 = 1|C_{N_2}) \times P(C_3 = 1|C_{N_3}) \times P(C_4 = 0|C_{N_4})$$

$$P_{f_{16}}(C|x) = P(C_1 = 1|C_{N_1}) \times P(C_2 = 1|C_{N_2}) \times P(C_3 = 1|C_{N_3}) \times P(C_4 = 1|C_{N_4})$$

Assume $P_{f_4}(C|x)$ yields the highest value, then the optimal configuration $f_n^* \in \mathbb{C}$ is $f_4 = 0011$. The conditional probability $p_k = P(C_k = 1|C_{N_k})$ is then evaluated using f_4 to assign labels C_{N_k} . Rule (3.13) is then used to assign labels to the new objects.

Chapter 4

Simulation Studies

The simulation in this section is conducted to investigate the performance of the three proposed neutral zone classifiers, with respect to the size of training samples, dependence structures of covariate data measurements, and the nature of subject-specific random effects.

4.1: Simulation Algorithm

The simulated data are generated from combinations of the following options:

- (1) Sample size N , 100, 200, 500 or 1000.
- (2) The number of covariates or labels, K , 5 or 10.
- (3) The variance-covariance matrix of random effects Σ_b , $2 \begin{bmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix}$, $2 \begin{bmatrix} 1.0 & 0.4 \\ 0.4 & 1.0 \end{bmatrix}$
or $2 \begin{bmatrix} 1.0 & 0.7 \\ 0.7 & 1.0 \end{bmatrix}$.
- (4) The variance-covariance structure of covariate measurements, Σ_x , associated with K random variables, diagonal, first-order autoregressive (AR1) or compound

symmetry (CS). Correlation coefficient ρ_x will be 0.4 and 0.7. For example, when $K = 5$ and $\sigma^2 = 2$, the structures of Σ_x are:

When Σ_x is diagonal:

$$2 \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

When Σ_x is AR1:

$$\rho_x = 0.4$$

$$2 \begin{bmatrix} 1.00 & 0.40 & 0.16 & 0.06 & 0.03 \\ 0.40 & 1.00 & 0.40 & 0.16 & 0.06 \\ 0.16 & 0.40 & 1.00 & 0.40 & 0.16 \\ 0.06 & 0.16 & 0.40 & 1.00 & 0.40 \\ 0.03 & 0.06 & 0.16 & 0.40 & 1.00 \end{bmatrix}$$

$$\rho_x = 0.7$$

$$2 \begin{bmatrix} 1.00 & 0.70 & 0.49 & 0.34 & 0.24 \\ 0.70 & 1.00 & 0.70 & 0.49 & 0.34 \\ 0.49 & 0.70 & 1.00 & 0.70 & 0.49 \\ 0.34 & 0.49 & 0.70 & 1.00 & 0.70 \\ 0.24 & 0.34 & 0.49 & 0.70 & 1.00 \end{bmatrix}$$

When Σ_x is CS:

$$\rho_x = 0.4$$

$$2 \begin{bmatrix} 1.00 & 0.40 & 0.40 & 0.40 & 0.40 \\ 0.40 & 1.00 & 0.40 & 0.40 & 0.40 \\ 0.40 & 0.40 & 1.00 & 0.40 & 0.40 \\ 0.40 & 0.40 & 0.40 & 1.00 & 0.40 \\ 0.40 & 0.40 & 0.40 & 0.40 & 1.00 \end{bmatrix}$$

$$\rho_x = 0.7$$

$$2 \begin{bmatrix} 1.00 & 0.70 & 0.70 & 0.70 & 0.70 \\ 0.70 & 1.00 & 0.70 & 0.70 & 0.70 \\ 0.70 & 0.70 & 1.00 & 0.70 & 0.70 \\ 0.70 & 0.70 & 0.70 & 1.00 & 0.70 \\ 0.70 & 0.70 & 0.70 & 0.70 & 1.00 \end{bmatrix}$$

For each combination among choices of sample size N , number of random variables K , subject-specific random effects matrix Σ_b and dependence structures of covariates Σ_x , we simulate a training set ($N = 100, 200, 500$ or 1000) using two different simulation approaches: the generalized linear mixed model (GLMM) and the Markov random field (MRF) algorithms. Three proposed neutral zone classifiers (nLNZC, LNZC and MNZC) are developed and tested on a set of new simulated objects ($N = 500$). The procedure for

simulating data, developing classifiers and obtaining error rates is demonstrated in **Figure 4.1**.

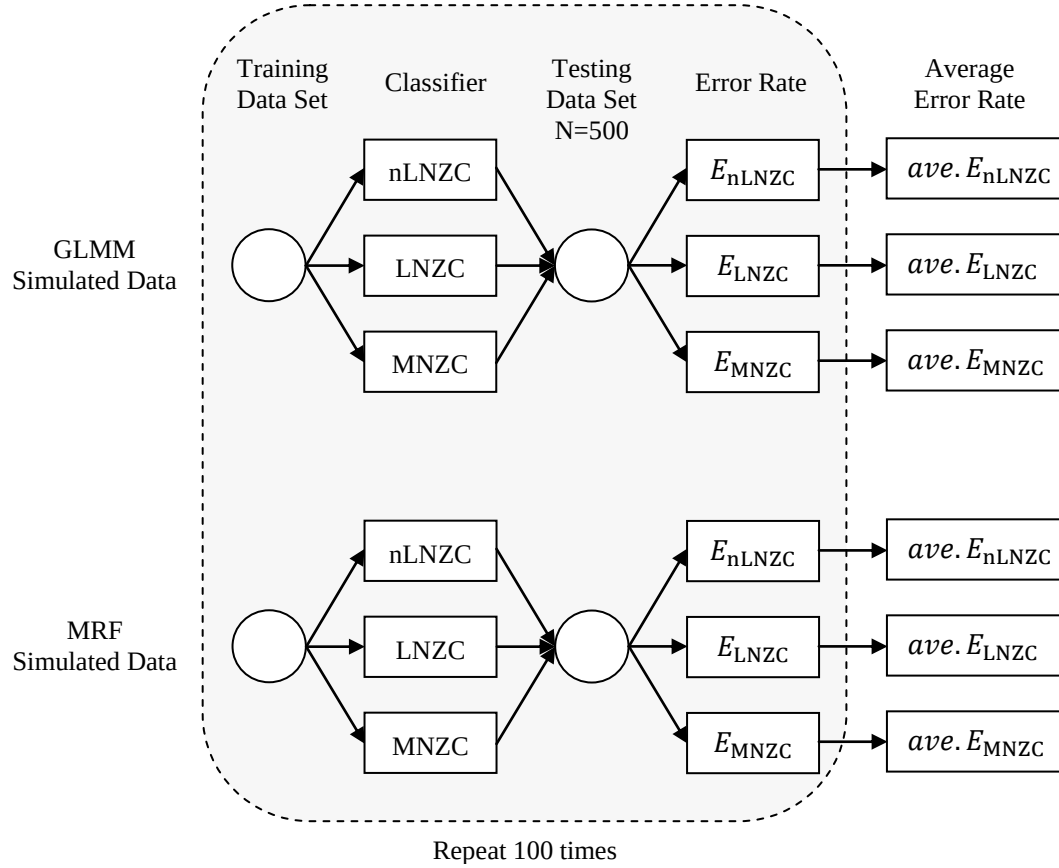


Figure 4.1: Flowchart of the procedure for simulating data, constructing proposed classifiers and obtaining misclassification error rates

4.1.1: Generalized Linear Mixed Model Algorithm

We set the fixed effect parameters for the GLMM model as:

$$\{\beta_{0k}\}_{k=1}^{K=5} = (3.0, 4.0, 2.0, 3.0, -5.0)$$

$$\{\beta_{1k}\}_{k=1}^{K=5} = (1.0, 2.0, 5.0, 3.0, 3.5)$$

$$\{\beta_{0k}\}_{k=1}^{K=10} = (3.0, 4.0, 2.0, 3.0, -5.0, -2.0, 2.0, 6.0, -1.0, 2.0)$$

$$\{\beta_{1k}\}_{k=1}^{K=10} = (1.0, 2.0, 5.0, 3.0, 3.5, 1.0, -8.0, 3.0, -5.0, 4.0)$$

The subject-specific random effects $b = (b_0, b_1)'$ and covariate measurements $x = (x_1, x_2, \dots, x_K)'$ are generated from $MVN(0, \Sigma_b)$ and $MVN(0, \Sigma_x)$, respectively. The corresponding class labels, $l = (l_1, l_2, \dots, l_K)'$, will be generated using a Bernoulli distribution with this form:

$$l_k \sim BERN \left\{ \frac{\exp[\beta_{0k} + b_{0k} + (\beta_{1k} + b_{1k})x_k]}{1 + \exp[\beta_{0k} + b_{0k} + (\beta_{1k} + b_{1k})x_k]} \right\}$$

In this simulation study, we use two-fold cross-validation to find neutral zone limits L_k^* as described in **Figure 2.4**.

4.1.2: Markov Random Field Algorithm

Gibbs sampling or Gibbs sampler is applied to simulate a random vector of binary class labels $C = \{C_1, C_2, \dots, C_K\}$ from the conditional distributions

$$P(C_k | x_k, C_{V-\{V_k\}}) = P(C_k = l_k | x_k, C_1 = l_1, \dots, C_{k-1} = l_{k-1}, C_{k+1} = l_{k+1}, \dots, C_K = l_K)$$

for a Markov random field. The simulated data constructed this way eventually converges to the joint distribution (Bishop, 2006; Koller and Friedman, 2009). Details of the implementation algorithm will be described subsequently.

Recall that:

$$P(C_k = 1 | x_k, C_{V-\{V_k\}}) = P(C_k = 1 | x_k, C_{N_k}) = \frac{\exp(\beta_{0k} + \beta_{1k}x_k + f_{N_k})}{1 + \exp(\beta_{0k} + \beta_{1k}x_k + f_{N_k})}$$

where f_{N_k} is defined in (4.11). The MRF simulation approach uses the same fixed-effect parameters of the GLMM model for the Markov random field, and the following parameters for the influences of neighboring nodes are added:

$$(\beta_{k,1})_{k=1}^{K=5} = (4.0, 2.5, 0.5, 2.0, 3.5)$$

$$(\beta_{k,2})_{k=1}^{K=5} = (1.5, 3.0, 4.0, 2.5, 5.0)$$

$$(\beta_{k,3})_{k=1}^{K=5} = (1.0, 2.5, 2.0, 3.0, 4.0)$$

$$(\beta_{k,1})_{k=1}^{K=10} = (4.5, 2.5, 3.0, 2.0, 5.5, 6.0, 1.5, 2.0, 3.0, 4.0)$$

$$(\beta_{k,2})_{k=1}^{K=10} = (5.0, 1.5, 2.0, 3.5, 1.0, 6.0, 4.0, 2.5, 7.5, 2.0)$$

$$(\beta_{k,3})_{k=1}^{K=10} = (2.0, 3.0, 2.0, 4.5, 3.5, 2.5, 5.0, 1.5, 2.5, 3.0)$$

These chosen parameters are used accordingly depend on the existing connections between nodes of K -node simulated graphs constructed based on five proposed structures of the covariate measurements matrix Σ_x (**Figure 4.2**). The algorithm for Gibbs sampling follows these steps:

1. At $t = 0$, assign randomly an initial state $l^0 = (l_1^0, l_2^0, \dots, l_K^0)$.
2. Repeat over iterations $t = 1, \dots, T$:

Draw $l_1^t \sim \text{BERN}[P(C_1 = 1|x_1, C_2 = l_2^{t-1}, C_3 = l_3^{t-1}, \dots, C_K = l_K^{t-1})]$

Draw $l_2^t \sim \text{BERN}[P(C_2 = 1|x_2, C_1 = l_1^t, C_3 = l_3^{t-1}, \dots, C_K = l_K^{t-1})]$

Draw $l_3^t \sim \text{BERN}[P(C_3 = 1|x_3, C_1 = l_1^t, C_2 = l_2^t, \dots, C_K = l_K^{t-1})]$

$\vdots$

Draw $l_{K-1}^t \sim \text{BERN}[P(C_{K-1} = 1|x_{K-1}, C_1 = l_1^t, C_2 = l_2^t, \dots, C_K = l_K^{t-1})]$

Draw $l_K^t \sim \text{BERN}[P(C_K = 1|x_K, C_1 = l_1^t, C_2 = l_2^t, \dots, C_{K-1} = l_{K-1}^t)]$

The number of iterations $T = 1000$ is used to generate simulated data.

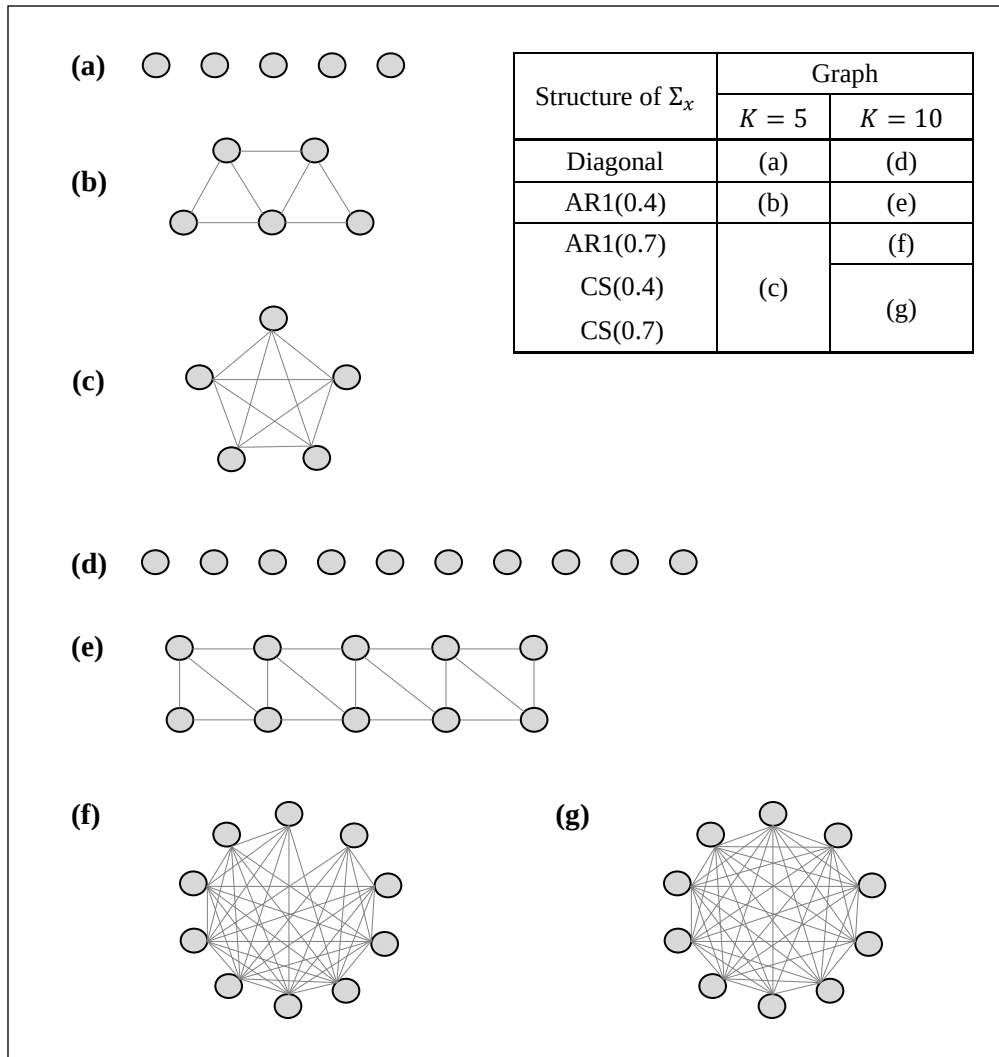


Figure 4.2: Simulated K -node graphs associated with different structures of Σ_x

4.1.3: Convergence Diagnostic for Markov Random Field Approach

To investigate convergence of the Gibbs sampler when using the number of iterations $T = 1,000$, we examined trace plots by looking for irreducible and aperiodic appearances (Givens and Hoeting, 2005; Meyn and Tweedie, 1993; Robert and Casella, 1999). As examples, we illustrate a Gibbs sampling sequence for each case $K = 5$ and $K = 10$. The examples correspond to generated labels for the case Σ_x was as AR1(0.4) structure. The values of x in each case were $x = (-4.0, -2.2, -0.65, -1.5, 1.2)$ and $x = (-4.0, -2.5, -0.6, -1.4, 1.2, -1.0, 0.55, -2.5, 0.4, -0.7)$, respectively. Two different initial states used were $(1, 0, 1, 0, 1)$, $(1, 1, 1, 1, 1)$ and $(0, 0, 0, 0, 0, 0, 0, 0, 0, 0)$, $(1, 1, 1, 1, 1, 1, 1, 1, 1, 1)$, respectively for $K = 5$ (**Figure 4.3**) and $K = 10$ (**Figure 4.4**). The trace plots show satisfactory convergence to a stationary distribution, that is independent of the starting values. Especially we can see the trace plots are statistically similar when looking at different intervals during the 1,000 iterations.

4.2: Results

4.2.1: Findings

The hard and soft misclassification error rates of all proposed classification methods throughout all simulation settings are represented in this section (see **Tables C.1** through **C.16** for more details). Define nLNZC, LNZC, MNZC1 and MNZC2 as the brief names of the naive logit-based neutral zone classifier, the logit-based neutral zone classifier, the Markov random field neutral zone classifier with the threshold approach, and the Markov random field neutral zone classifier with the voting rule using the

automatic voting procedure, respectively. **Figures 4.5 to 4.8** show comparison results across four different sample sizes for training data ($N = 100, 200, 500$ and 1000), five different covariate data structures AR1(0.4), AR1(0.7), CS(0.4), CS(0.7) and diagonal, and two different types of multi-label data ($K = 5$ and 10) using the GLMM and MRF simulation approaches.

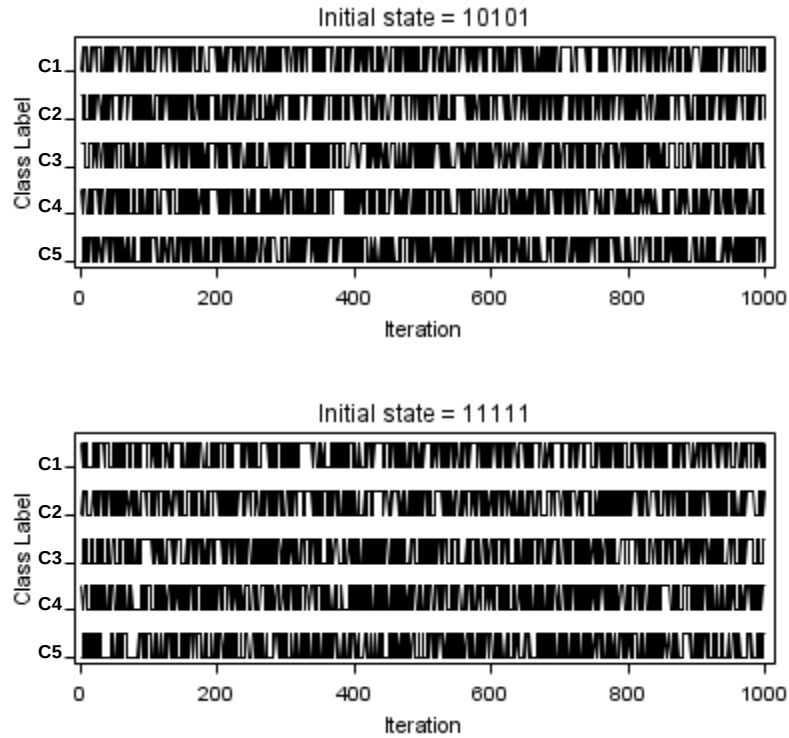


Figure 4.3: Trace plots of Gibbs sampler chains with number of labels $K = 5$ based on 1,000 iterations using two different initial states, $(1, 0, 1, 0, 1)$ and $(1, 1, 1, 1, 1)$

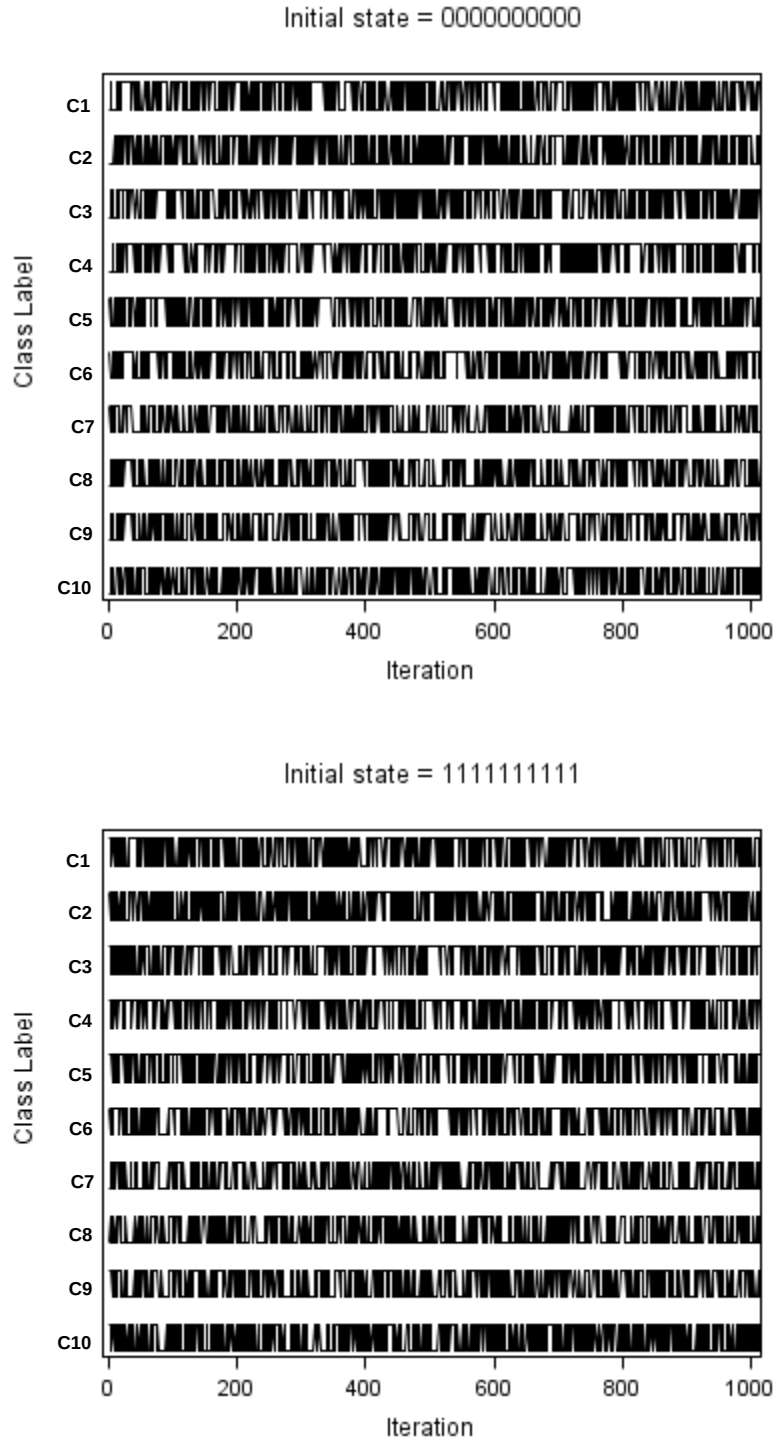


Figure 4.4: Trace plots of Gibbs sampler chains with number of labels $K = 10$ based on 1000 iterations using two different initial states, $(0,0,0,0,0,0,0,0,0,0)$ and $(1,1,1,1,1,1,1,1,1,1)$

For the GLMM simulated data (see **Figures 4.5** and **4.6**), LNZN does not work well when $N = 100$ (28% to 32% and 37% to 45% in the total error rate when $K = 5$ and 10, respectively, vs. 26% to 29% and 32% to 36% for nLNZN, MNZN1 and MNZN2). The performance of all four neutral zone classifiers illustrates a huge reduction in the total error rate as the sample size N increases from 100 to 1000, i.e., 11% to 18% ($K = 5$) and 22.5% to 30% ($K = 10$) for LNZN, 10% to 14% ($K = 5$) and 16% to 20% ($K = 10$) for the other three methods, nLNZN, MNZN1 and MNZN2. There is no improvement in prediction accuracy when the sample size N changes from 200 to 500; however, the total error rate is slightly small and up to 6% when N raises from 500 to 1000.

Figures 4.7 and **4.8** illustrate the findings for the MRF simulated data. Similarly, LNZN yields a very high total error rate when $N = 100$ for both $K = 5$ (38% to 43.6% vs. 27% to 35% for nLNZN, MNZN1 and MNZN2) and $K = 10$ (20% to 46% vs. 14% to 36.5% for the other three methods). All proposed classifiers have a little trouble with the CS(0.7) covariate data structure and perform very well with AR1(0.7) when $K = 10$ for the sample sizes of training data $N = 500$ and below. When $N = 1000$, nLNZN, LNZN, MNZN1 and MNZN2 perform very well across five different covariate data structures for both $K = 5$ and 10. However, nLNZN performs less well when $K = 10$ for CS(0.7) (32.5% vs. less than 15% for LNZN, MNZN1 and MNZN2). Together, LNZN and two MRF classifiers are superior to nLNZN in that they consistently have smaller total error rates when the size of training data is sufficiently large.

4.2.2: Discussion

Misclassification error rates of the proposed classification methods are apparently affected by the size of training data and the covariate data structure. They improve their ability to predict classification when the size of training data is suitably large using the GLMM simulation approach. When the MRF simulation algorithm is used, the proposed classifiers seem to perform less well for the compound symmetry covariate data structure ($\rho = 0.7$) with a relatively small size of training data; compared to the logit-based and MRF approaches, the naive logit-based method is the worse classifier for the MRF simulated data. The Markov random field classifier is robust in every case of the current simulation criteria as compared to the logit-based classifiers. These neutral zone classification methods for multi-label data each have their own advantages and disadvantages, which are compared in **Table 4.1**. The results of this simulation study show that these classifiers contribute useful alternatives when dealing with multi-label classification problems. If these labels are independent and there is a limited source of training data, the naive logit-based neutral zone classifier should be considered in terms of its relative good performance and simple, fast implementation compared to the logit-based and the Markov random field neutral zone classifiers.

Method	Advantages	Disadvantages
nLNZC	Simple and fast	Ignores label correlations Have some trouble with compound symmetry covariate structure
LNZC	Easy to implement Takes into account label correlations Correlation structure is relatively transparent to interpret.	Take more time to compute marginal expected probability
MNZC1 and MNZC2	Implementation aspects are easy. Takes into account label correlations	Computationally expensive during the prediction process for very large number of labels as creating a pool of 2^K configurations

Table 4.1. Comparison of the proposed neutral zone classifiers

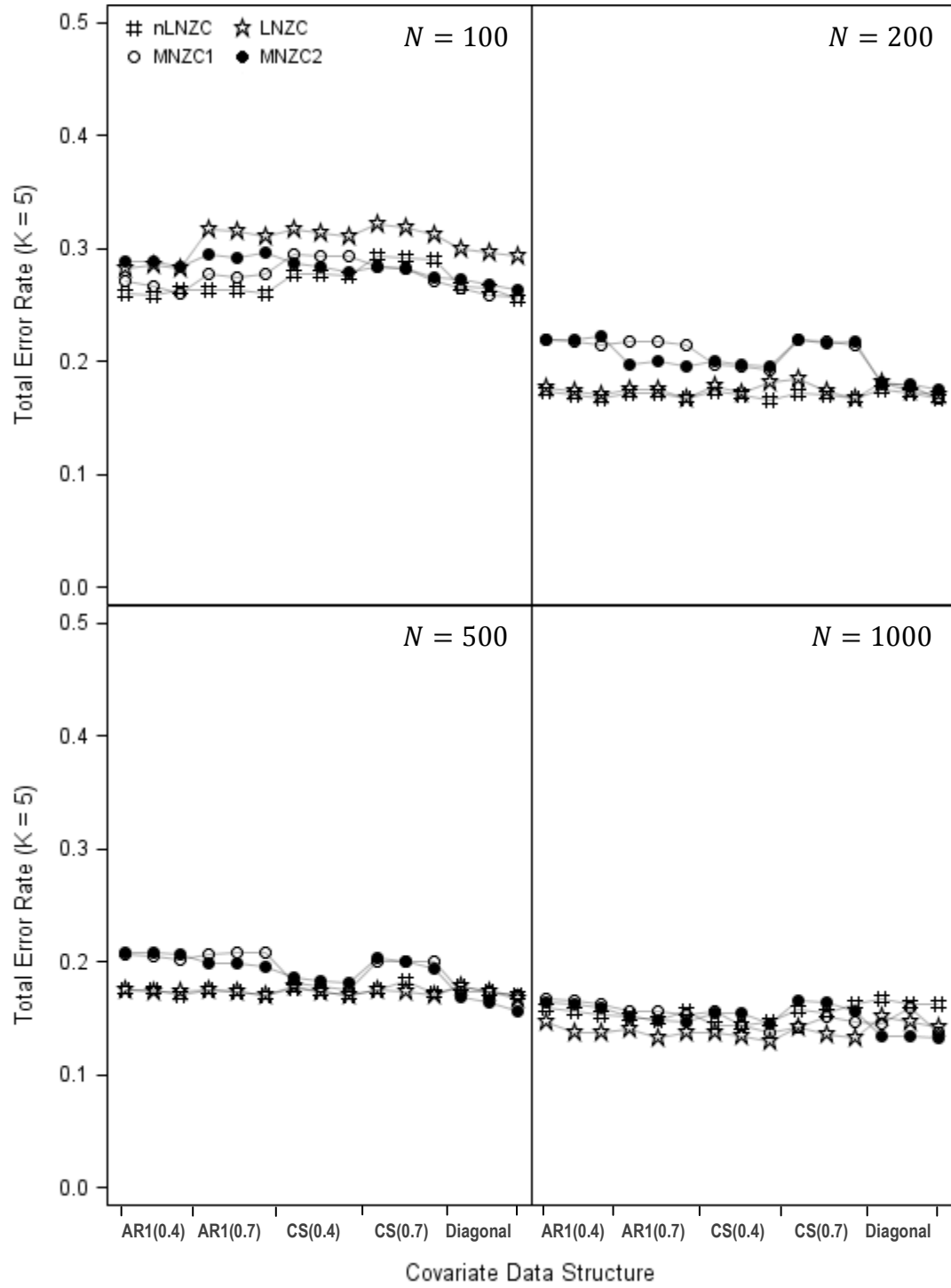


Figure 4.5: Misclassification error rates of proposed neutral zone classifiers vs. covariate data structure for the GLMM simulated data when $K = 5$

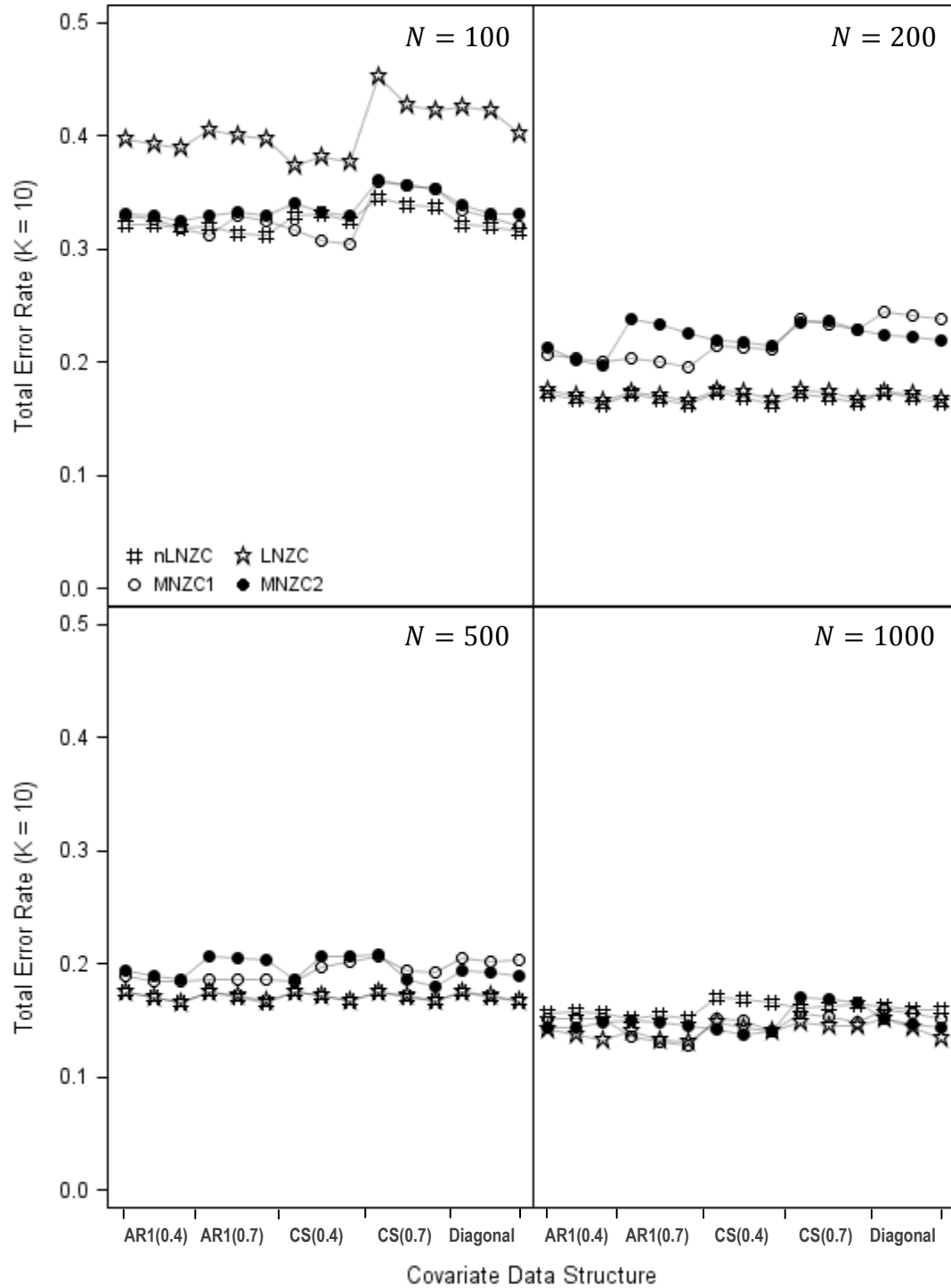


Figure 4.6: Misclassification error rates of proposed neutral zone classifiers vs. covariate data structure for the GLMM simulated data when $K = 10$

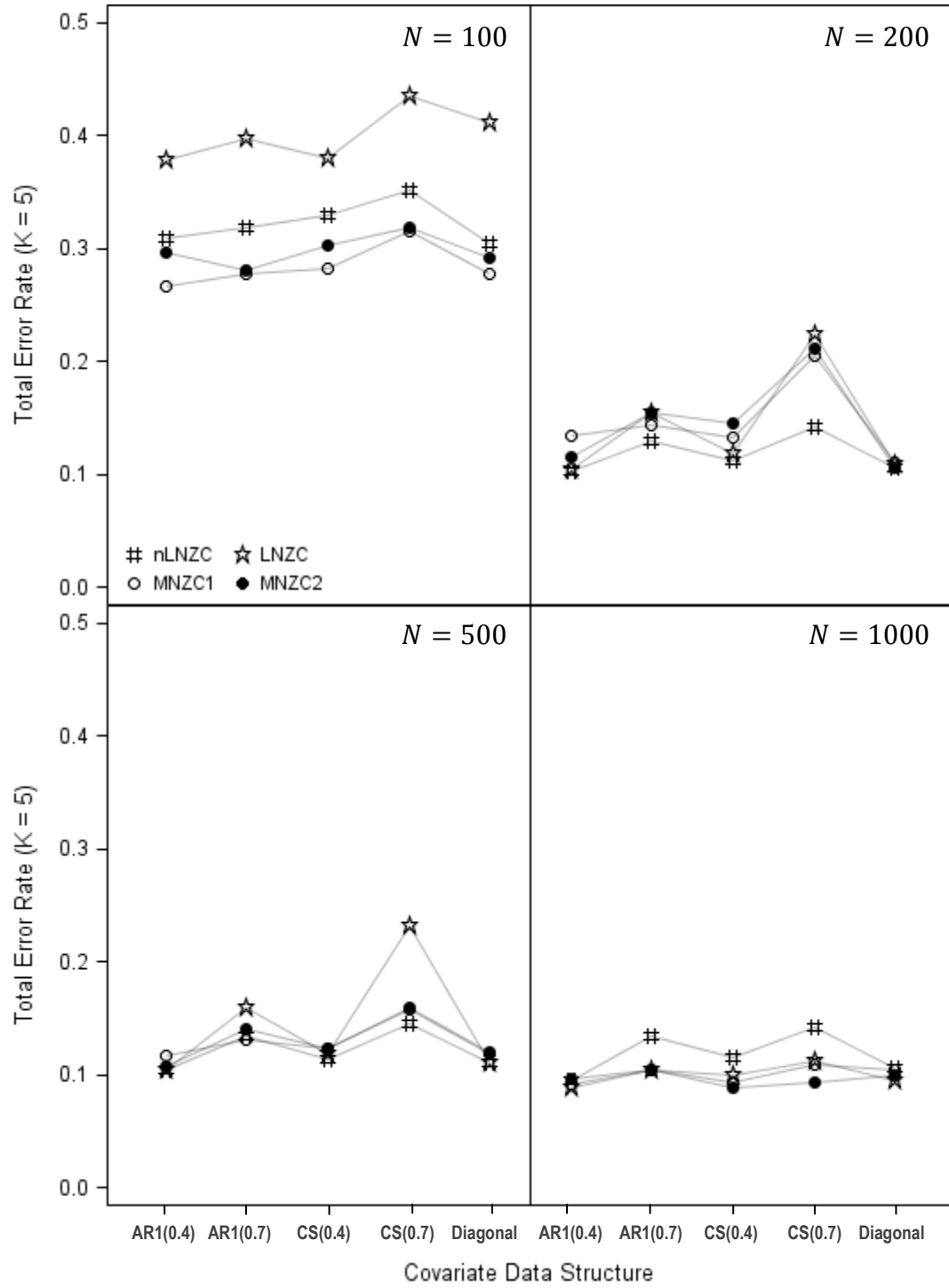


Figure 4.7: Misclassification error rates of proposed neutral zone classifiers vs. covariate data structure for the MRF simulated data when $K = 5$

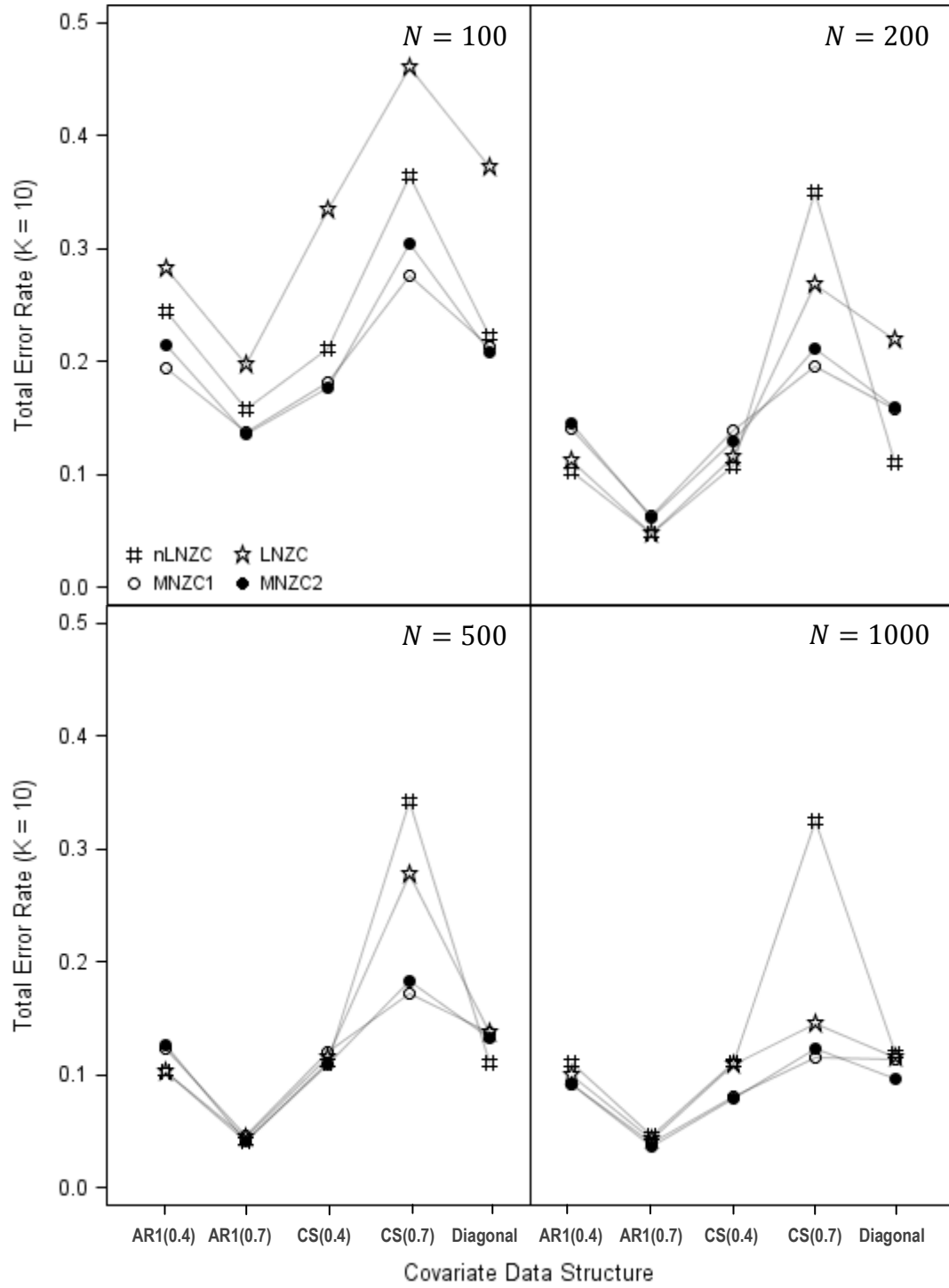


Figure 4.8: Misclassification error rates of proposed neutral zone classifiers vs. covariate data structure for the MRF simulated data when $K = 10$

Chapter 5

Biological Application

5.1: Data Introduction and Preparation

The proposed classification methods for repeated measures are investigated on a real data set of 344 fungi control clones hybridized with 26 optimally selected probes in the application of the microbial community profiling of the OFRG study that was introduced in section 1.3. See **Table 5.1** for an illustration of the data where C_{jk} denotes a known hybridization class label along with its natural log of observed intensity o_{jk} when clone j was hybridized with probe k .

	Probe 1	Probe 2	...	Probe k	...	Probe K
Clone 1	(C_{11}, x_{11})	(C_{12}, x_{12})	...	(C_{1k}, x_{1k})	...	(C_{1K}, x_{1K})
Clone 2	(C_{21}, x_{21})	(C_{22}, x_{22})	...	(C_{2k}, x_{2k})	...	(C_{2K}, x_{2K})
...	...	...	...	...	...	...
Clone j	(C_{j1}, x_{j1})	(C_{j2}, x_{j2})	...	(C_{jk}, x_{jk})	...	(C_{jK}, x_{jK})
...	...	...	...	...	...	...
Clone N	(C_{N1}, x_{N1})	(C_{N2}, x_{N2})	...	(C_{Nk}, x_{Nk})	...	(C_{NK}, x_{NK})

Table 5.1: Structure of a fungi training data set for N clones and K probes

The five-fold stratified split-sample method (Kohavi, 1995) is employed to partition this fungi data into five disjoint subsets for later uses such as finding neutral zone classification limits, L_k , and computing misclassification error rates to evaluate our proposed classifiers. Of these five subsamples of approximately equal size, each set will contain the same proportions of binding and non-binding class labels as the original as much as possible. The process of data preparation is illustrated in **Figure 5.1**. In general, define $D = \{(C_{jk}, x_{jk}), j = 1, 2 \dots N, k = 1, 2 \dots K\}$ as the original data set, where j and k are indexes of control clones and probes, respectively. The data set D is divided into five disjointed subsets $D_1, D_2, \dots, D_5$. At the first step, set D is split into two subgroups containing only class 0 or 1 for the first arbitrarily chosen probe, called sets $D_{1,C_1=0}$ and $D_{1,C_1=1}$, or briefly, $D_{1,0}$ and $D_{1,1}$. Next, each set D_{1,C_1} will be assigned into another two subsets, again holding only class 0 or 1 from the second selected probe to produce four subsets, namely $D_{2,00}$, $D_{2,01}$, $D_{2,10}$ and $D_{2,11}$. Note that at each split, each subset may yield another two subsets corresponding to two existing classes, or they may remain the same in case of having only class 0 or 1 across all clones. There is no restriction upon which probe to pick to divide earlier or later. One is to keep cutting this way for all K probes, one at the time. At the k^{th} step, we have 2^k subsets. One then randomly puts 20 percent of clones of each subset, after the final stage, into $D_1, D_2, \dots, D_5$. **Table D.1** shows details of five subsets of the original fungi data set for 26 working probes. Of these five subsamples of approximately equal size, each set will contain the same proportions of binding and non-binding class labels as the original as much as possible. Since seven

probes with the identification numbers (IDs) 9, 16, 22, 34, 36, 37 and 38 appear with a sparse portion of either two classes, they are excluded from the analysis.

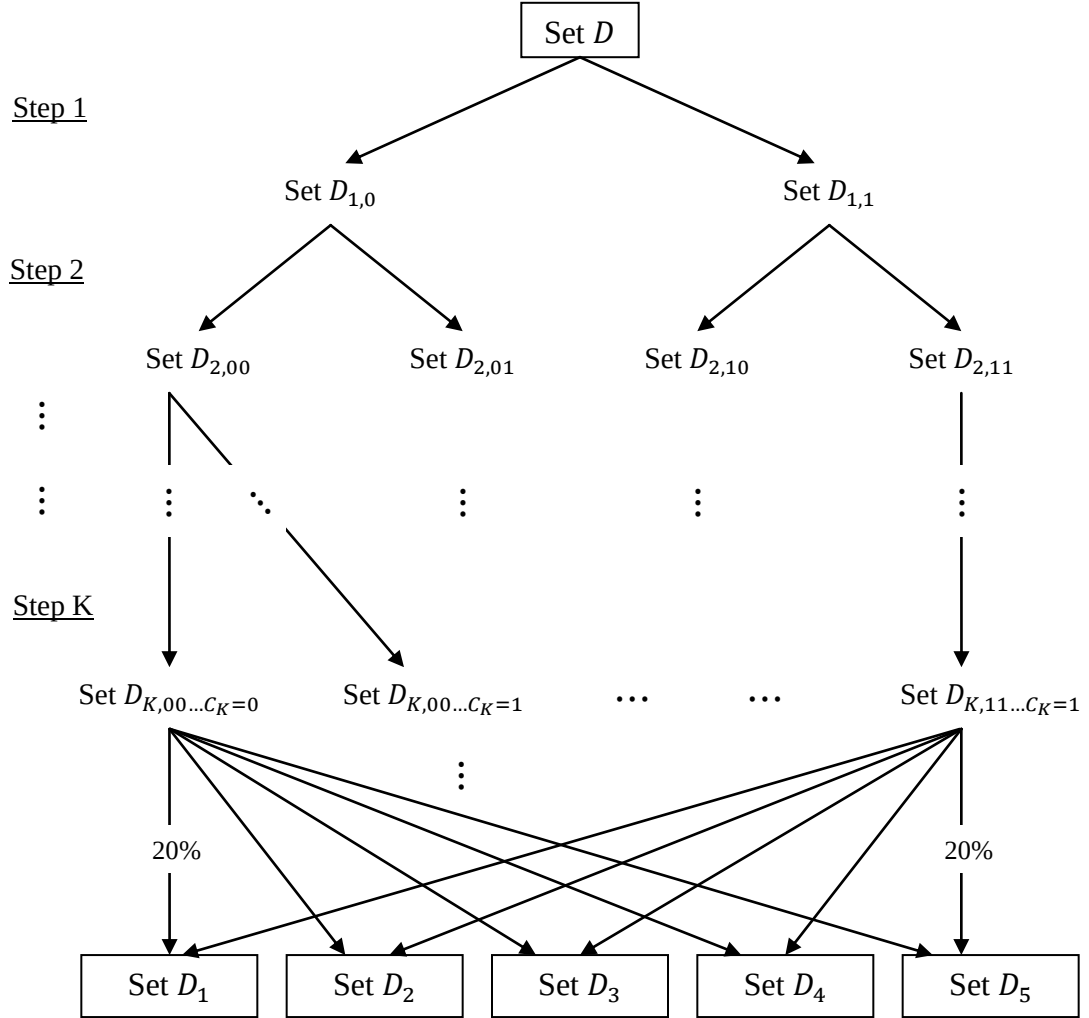


Figure 5.1: Layout of the data preparation process

5.2: Analysis of the Logit-based Classifier

5.2.1: Parameter Estimates

The implementation of the proposed logit-based neutral zone classifier (LNZC) is demonstrated in this section. **Table 5.1** shows fixed parameter estimates of 26 probes, $\hat{\beta}_{0k}$ and $\hat{\beta}_{1k}$, obtained from the 344-clone fungi training data. The 2×2 unstructured random effect covariance matrix Σ is highly significant ($p < 0.0001$). The estimates of covariance parameters of this fungi data set are $\sigma_0^2 = 5.2622$, $\sigma_1^2 = 1.4287$ and $\sigma_{01} = 2.2373$, which yield the high correlation coefficient $\rho = 0.82$ between random effect components. These appear to suggest that both the random intercept and slope are needed in the proposed logit-based model. **Table 5.2** also includes parameter estimates of the naive logit-based neutral zone classifier (nLNZC).

5.2.2: Neutral Zone Limit

We already demonstrated in detail the procedure of how to determine the neutral zone limit in the simulation studies. In this chapter, two chosen probes of the fungi data set, with IDs 23 and 13, will illustrate two cases of the algorithm: when the neutral zone does not exist ($L_k^* = 0$) and when it does. From **Figure 5.2b**, the hard error rate of probe 23 is 3.5% and less than $\alpha = 0.05$ at $L = 0$; hence, the neutral zone for this probe does not exist, or $L^* = 0$. **Figure 5.2a** supports this finding by showing a separate log of intensities for classes 0 and 1. On the other hand, **Figure 5.2c** suggests a neutral zone for probe 13 for displaying the overlapping distribution of the logarithm of intensity measurements of two classes. The neutral zone limit for this probe is shown in **Figure**

5.2d. Together, the proposed algorithm to find the neutral zone limit is simple and can determine whether the neutral zone classification region is needed for any particular probe.

Probe ID	LNZC		nLNZC	
	$\hat{\beta}_{0k}$	$\hat{\beta}_{1k}$	$\hat{\beta}_{0k}$	$\hat{\beta}_{1k}$
1	6.62	3.92	4.10	1.84
2	10.04	4.26	6.37	2.25
3	-0.61 *	5.63	-0.47 *	3.36
4	4.29	3.62	2.16	1.89
6	3.41	3.54	2.03	1.90
8	7.94	5.22	5.53	3.25
10	6.88	4.06	3.66	1.89
13	-0.05 ¶	2.66	-0.37 ¶¶	1.70
15	14.56	5.11	10.51	3.46
17	5.43	3.84	3.73	2.60
19	3.45	2.95	1.97	1.49
21	7.41	3.96	5.26	2.56
23	12.23	4.82	6.34	2.06
24	1.51	3.66	0.64 **	2.17
25	9.76	3.70	5.49	1.74
26	2.03	4.46	0.91	2.78
27	2.81	5.26	1.68	3.67
29	0.23 ¶	3.95	-0.42 ¶¶	2.36
30	14.69	5.44	8.55	2.88
31	27.47	9.24	18.92	6.30
32	2.05	3.90	0.88 ***	2.42
33	10.29	5.15	5.33	2.42
39	11.37	4.22	7.77	2.45
40	9.11	4.55	5.78	2.32
43	1.37	2.73	0.41 *	1.62
44	3.51	3.94	2.12	2.46

Table 5.2: Parameter estimates for 26 probes of fungi data of the logit-based and the naïve logit-based classifiers (LNZC and nLNZC)

¶: not significant; ¶¶: $p \leq 0.1$

*: $p \leq 0.05$; **: $p \leq 0.01$; ***: $p \leq 0.001$; No sign: $p < 0.0001$

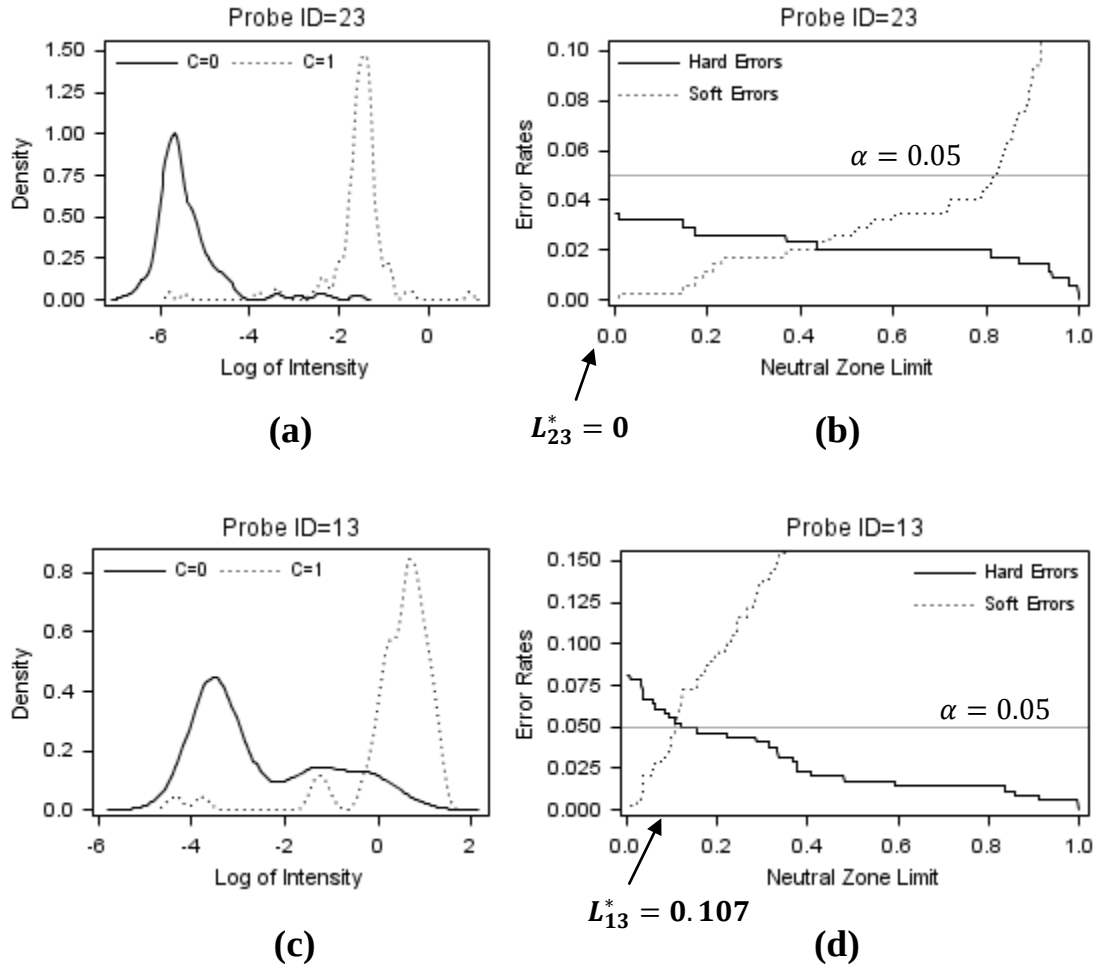


Figure 5.2: Graphical illustration of finding the neutral zone limits of two probes:

- (a) Log(intensity) distributions of probe ID = 23
- (b) Error rates vs. neutral zone limit of probe ID = 23
- (c) Log(intensity) distributions of probe ID = 13
- (d) Error rates vs. neutral zone limit of probe ID = 13

5.3: Implementation of the Markov Random Field Classifiers

5.3.1: Graph Construction and Parameter Estimation

The undirected graph represented the influence structure of twenty six probes on each other is constructed using a set of Pearson correlation coefficients $r_{kk'}$ obtained from the fungi data set because it is not assessable historically (**Figure 5.3**). All connections between probes are disregarded if their corresponding correlation coefficients are less than 0.1.

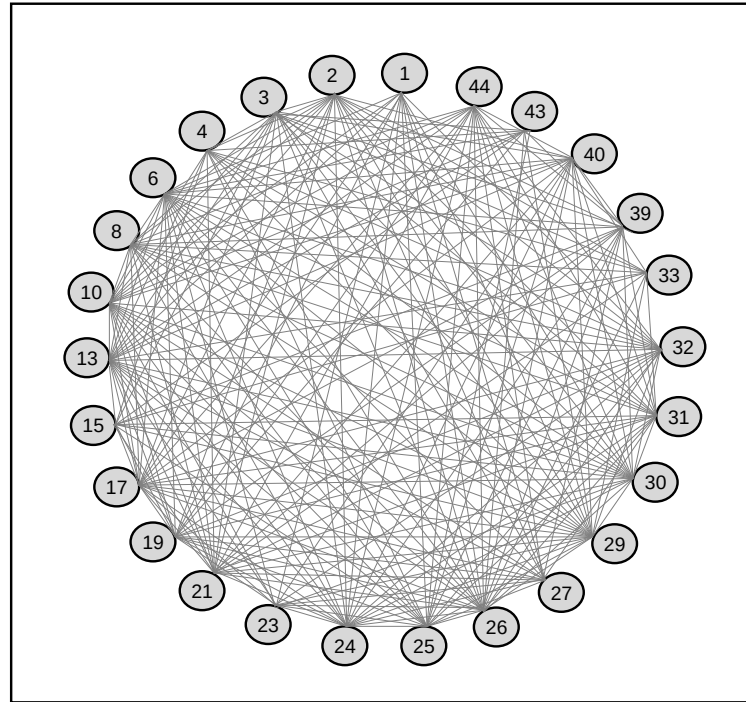


Figure 5.3: An undirected graph for 26 probes of fungi data

The parameter estimates of fungi data for the Markov random field model is given in **Table 5.3**. Parameter estimates associated with the high correlation coefficients (above

0.4), $\hat{\beta}_{k,2}$ and $\hat{\beta}_{k,3}$, appear to be likely more significant compared to those connected with the low bin of correlations.

Probe ID	$\hat{\beta}_{0k}$	$\hat{\beta}_{1k}$	$\hat{\beta}_{k,1}$	$\hat{\beta}_{k,2}$	$\hat{\beta}_{k,3}$
1	-0.47 ¶	1.46 **			8.40
2	-0.53 ¶	1.29		1.60 *	3.40
3	-2.44	3.34		1.55	
4	2.15	1.89			
6	2.03	1.90			
8	-1.33	8.94 ***		-12.34	24.67
10	2.63	1.82		1.27 **	
13	-3.48	1.46	-0.76 *	1.75	
15	0.59 ¶	0.60 *	1.85		
17	1.14 ¶	2.24		0.94 **	
19	-0.97 ¶	1.35	1.36 *	2.35	
21	3.37	2.69		1.45 ***	
23	-0.10 ¶	1.19 **			8.05
24	-0.64 ¶	2.19	0.91 ***		
25	3.82	1.60	0.91 *	1.37 *	
26	-6.17 *	3.59 ¶¶			6.95 **
27	1.68	3.67			
29	-0.42 ¶¶	2.36			
30	8.64	2.91			
31	15.21	5.57		-0.82 *	6.39
32	-2.72	1.27 ***			4.09
33	5.36	2.44			
39	2.91 *	2.23		1.59 *	6.59 **
40	0.20 ¶	1.48 ***			3.52
43	-1.64	1.64		5.57	
44	2.12	2.46			

Table 5.3: Parameter estimates of the Markov random field classifier for 26 probes of fungi data

¶: not significant; ¶¶: $p \leq 0.1$

*: $p \leq 0.05$; **: $p \leq 0.01$; ***: $p \leq 0.001$; No sign: $p < 0.0001$

5.3.2: Neutral Zone Limit

As proposed in section 3.3, the Markov random field neutral zone classifier gives neutral predictions following two methods: the threshold approach (MNZC1) and the voting rule (MNZC2). For the voting-rule approach, a vector of class labels for a new clone is voted from a set of optimal configurations that are selected carefully from a huge pool of $2^{26} = 67,108,864$ possible candidates using a Scree plot. **Figure 5.4** shows Scree plot of the top 25 pseudo-likelihoods for one selected clone (ID = 205) of fungi data. A big gap occurs between the top eight configurations of that clone, so they are appropriately extracted from the pool.

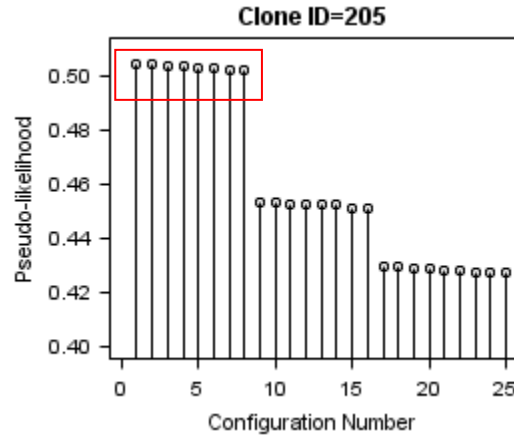


Figure 5.4: Scree plot of 25 configurations with highest pseudo-likelihoods for one selected clone of fungi data

See **Table 5.4** for neutral zone limits for 26 probes of the fungi data for three proposed classification methods (LNZC, nLNZC and MNZC1) and GNZC. All four classifiers assign neutral zone boundaries for probes with IDs 3, 13 and 43. LNZC and MNZC1

have neutral zone limits for 10 probes (vs. five and eight for nLNZC and GNZC, respectively), and six of them are in common. GNZC appears to produce large neutral zone boundaries compared to the other methods.

Probe ID	Method			
	LNZC	nLNZC	MNZC1	GNZC
1	0	0	0	0
2	0	0	0	0
3	0.207	0.08	0.244	0.223
4	0	0	0	0.321
6	0.264	0	0.328	0
8	0	0	0	0
10	0	0	0	0
13	0.107	0.151	0.205	0.464
15	0	0	0.199	0
17	0	0	0	0
19	0	0	0	0.414
21	0	0	0	0
23	0	0	0	0
24	0.215	0	0.183	0.449
25	0	0	0.463	0
26	0.248	0.149	0	0.28
27	0.197	0.115	0.176	0
29	0.301	0	0.316	0
30	0	0	0	0
31	0	0	0	0.417
32	0.457	0	0	0
33	0	0	0	0
39	0	0	0	0
40	0	0	0	0
43	0.309	0.082	0.208	0.530
44	0.169	0	0.140	0

Table 5.4: Neutral zone limits for 26 probes of the fungi data for the logit-based, naïve logit-based, threshold-based and Gaussian-mixture neutral zone classifiers (LNZC, nLNZC, MNZC1 and GNZC)

5.4: Results

5.4.1: Findings

The five-fold cross-validation misclassification error rates of five neutral zone classifiers LNZC, nLNZC, MNZC1, MNZC2 and GNZC are reported in **Tables D.2, D.3** and **D.4** for the fungi data set. See **Tables D.5** and **D.6** for the fitting results of GNZC. **Figure 5.5** shows that all five classification methods provide less than 10% in the total error rates, which indicate their good predictive ability. MNZC2 produces a relative high error rate on probe 25 (about 30%) compare to those of the other probes.

5.4.2: Discussion

Four proposed neutral zone classifiers (LNZC, nLNZC, MNZC1 and MNZC2) and GNZC perform very well, as they give more than 90% of the correct predictions in the fungi data set. **Figure 5.6** demonstrates the distributions of a log of measurement intensities and error rates of five classification methods for probes 2 and 3, which respectively produced good results and bad results (relative to the other probes) for all five methods (see **Figure D.1** for distributions of all 26 probes). For probe 2, it is not surprising to see a very good outcome (less than 3% in total error rates) for each approach, because there was a very clear distinction between two class labels 0 and 1. On the other hand, probe 3 indicates a large portion of overlapping data, and the poor predictions are expected for any classification methods (up to 20% total error rates). In this situation, GNZC tends to be worse, and MNZC2 performs best (20% and 14.5% total error rates, respectively). This phenomenon indicates that MNZC2 might be very good

for overlapping multi-label data. The performance of two methods that ignore the correlated labels nLNZC and GNZC is very much similar. Nonetheless, nLNZC is better in terms of its conceptually simple, fast procedure and relaxing underlying assumption of the Box-Cox Gaussian-mixture distribution of multi-label data. As showing a very clear separation between class label 0 and 1 for most of probes in the OFRG study, nLNZC is the recommended approach to reduce the computational cost.

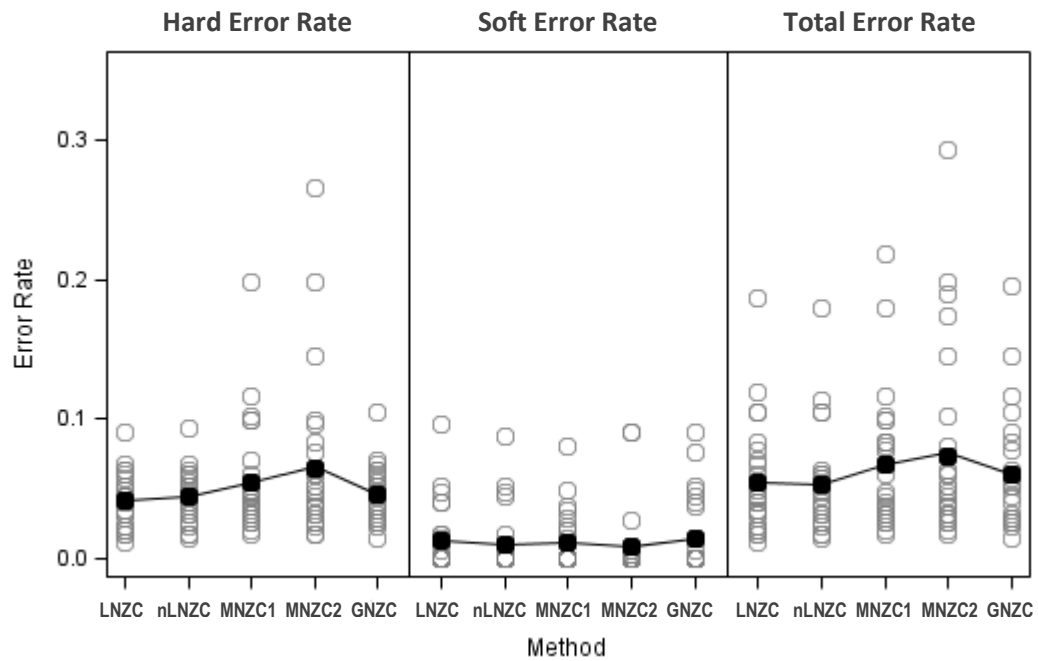
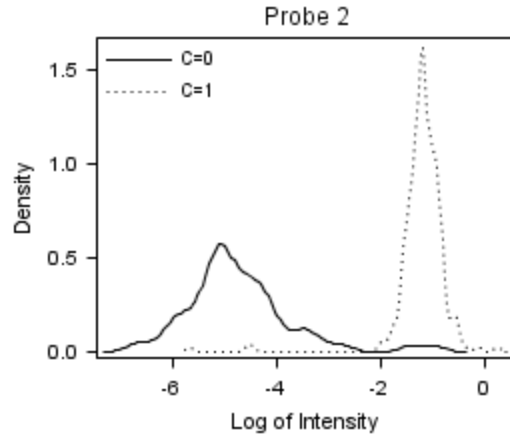
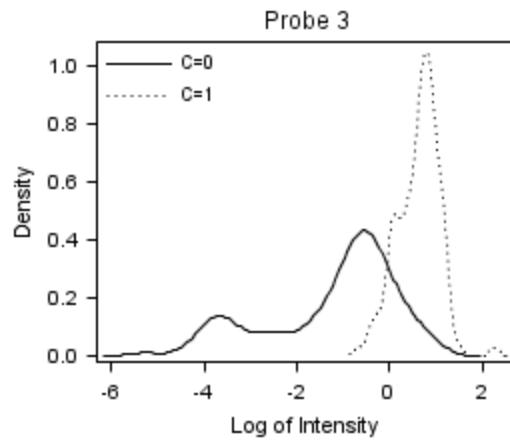


Figure 5.5: Five-fold cross-validation misclassification error rates of all neutral zone classification methods



Method	Hard Error	Soft Error	Total
LNZC	0.020	0.000	0.020
nLNZC	0.026	0.000	0.026
MNZC1	0.026	0.000	0.026
MNZC2	0.023	0.003	0.026
GNZC	0.029	0.000	0.029



Method	Hard Error	Soft Error	Total
LNZC	0.090	0.096	0.186
nLNZC	0.093	0.087	0.180
MNZC1	0.099	0.081	0.180
MNZC2	0.145	0.000	0.145
GNZC	0.105	0.090	0.195

Figure 5.6: Distributions of log (intensity) and five-fold cross-validation misclassification error rates of fungi data for probes 2 and 3

Chapter 6

Concluding Remarks and Future Work

6.1: Concluding Remarks

In this study, we have introduced alternative classification approaches using a generalized linear mixed model and Markov random field to deal with multi-label data and relax underlying assumptions of the previous neutral zone classification method. These proposed methods are reasonably easy to implement and perform very well for the simulation data sets and also work well on the biological data set. In the comparisons we made there was not an advantage to the multi-label approaches in terms of accuracy of the predictions compared to using a multiple single-label classification methodology. The naive logit-based neutral zone classifier is recommended for independent labels. In the framework of the generalized linear mixed model, we showed that the expected estimated probability used to derive the logit-based classifier can be interpreted as a Bayesian predictive marginal probability under the assumption of a degenerate prior.

6.2: Future Work

We believe that our proposed neutral zone classification approaches are useful for practical application when working with multi-label data. For further study, we can extend our proposed neutral zone classifiers for binary multi-label data to multi-class multi-label data. Future work also tries to characterize the equivalence of the marginal approach and the joint approach.

Bibliography

- Agresti, A. (2002) Categorical Data Analysis, 2nd ed., Hoboken, N.J.: Wiley.
- Aioli, F. & Sperduti, A. (2005). Multiclass classification with multi-prototype support vector machines. *Journal of Machine Learning Research*, 6: 817-850.
- Barutcuoglu, Z., Schapire, R.E., and Troyanskaya, O.G. (2006). Hierarchical multi-label prediction of gene function. *Bioinformatics*, 22: 830-836.
- Bent, E., Yin B., Figueroa, A., Ye, J., Fu, Q., Liu, Z., McDonald, V., Jeske, D., Jiang, T., Borneman, J.. (2006). Development of a 9600-clone procedure for oligonucleotide fingerprinting of rRNA genes: utilization to identify soil bacterial rRNA genes that correlate in abundance with the development of avocado root rot. *Journal of Microbiological Methods* 67: 171-180.
- Berg, R.D (1996). The indigenous gastrointestinal microflora. *Trends Microbiol* 11: 430-435.
- Besag, J. (1974). Spatial Interaction and the Statistical Analysis of Lattice Systems (with discussions). *Journal of the Royal Statistical Society. Series B36*, 192-236.
- Bishop, C. (2006). *Pattern Recognition and Machine Learning*. Springer Science & Business Media, LLC, New York.
- Blockeel, H., Schietgat, L., Struyf, J., Dzeroski, S. and Clare, A. (2006). Decision trees for hierarchical multilabel classification: A case study in functional genomics. *Proceedings of the 10th European Conference on Principles and Practice of Knowledge Discovery in Databases*, 4213: 18-29.
- Boutell, M., Shen, X., Luo, J. and Brown, C. (2003). Multi-label semantic scene classification. Technical Report, Department of Computer Science University of Rochester.
- Boutell, M.R., Luo, J., Shen, X., and Brown, C.M. (2004). Learning multi-label scene classification. *Pattern Recognition*, 37(9): 1757-1771.
- Breslow, N.E. and Clayton, D.G. (1993). Approximate inference in generalized linear mixed models. *Journal of the American Statistical Association* 88: 9-25.

- Brinker, K. & Hullermeier, E. (2007). Case-based multilabel ranking. Proceedings of the 20th International Conference on Artificial Intelligence, Hyderabad, India: 702-707.
- Brown, H. & Prescott, R. (2006). Applied Mixed Models in Medicine, 2nd edn., John Wiley and Sons.
- Derin, H. & Cole, W.S. (1986). Segmentation of Textured Images Using Gibbs Random Fields. Computer Vision, Graphics and Image Processing 35: 72-98.
- Derin, H. & Elliott, H. (1987). Modeling and Segmentation of Noisy and Textured Images Using Gibbs Random Fields. IEEE Transactions on Pattern Analysis and Machine Intelligence 9(1): 39-55.
- Dharmadhikari, S.C., Ingle, M. & Kulkarni, P. (2011). A comparative analysis of supervised multi-label text classification methods. International Journal of Engineering Research and Applications, 1(4): 1952-1961.
- Elisseeff, A. & Weston, J. (2002). Kernel methods for multi-labelled classification and categorical regression problems. Advances in Neural Information Processing Systems, 14: 681-687.
- Furnkranz, J., Hullermeier, E., Mencia, E.L. and Brinker, K. (2008). Multilabel classification via calibrated label ranking. Machine Learning, 73(2): 133-153.
- Geman, S. & Geman, D. (1984). Stochastic Relaxation, Gibbs Distribution and the Bayesian Restoration of Images. IEEE Transaction on Pattern Analysis and Machine Intelligence 6(6): 721-741.
- Geman, S. and Graffigne, C. (1986). Markov Random Field Image Models and Their Applications to Computer Vision. In Gleason, A.M. (Editor), Proceedings of the International Congress of Mathematicians: Berkeley, August 3-11: 1496-1517.
- Ghamrawi, N. & Mccallum, A. (2005). Collective multi-label classification. Proceedings of the 14th ACM International Conference on Information and Knowledge Management, 195-200.
- Givens, G.H. & Hoeting, J.A. (2005). Computational Statistics, John Wiley & Sons, Inc.
- Hammersley, J. M. and Clifford, P. (1971). Markov Field on Finite Graphs and lattices. Unpublished.
- Hastie, T., Tibshirani, R. and Friedman, J.H. (2001). The Elements of Statistical Learning, New York: Springer-Verlag.

- Hilbe, J.M. (2009). Logistic Regression Models. Chapman & Hall/CRD Press. ISBN 987-1-4200-7575-5.
- Hosmer, D.W & Lemeshow, S. (2000). Applied Logistic Regression, 2nd ed. New York; Chichester, Wiley. ISBN 0-471-35632-8.
- Hullermeier, E., Furnkranz, J., Cheng, W., and Brinker, K. (2008). Label ranking by learning pairwise preferences. *Artificial Intelligence*, 172: 1897-1916.
- Jeske, D.R., Liu, Z., Bent, E., and Borneman, J. (2007). Classification rules that include neutral zones and their application to microbial community profiling. *Communication in Statistics – Theory and Methods* 36(10): 1965-1980.
- Jin, B., Muller, B., Zhai, C., and Lu, X. (2008). Multi-label literature classification based on the Gene Ontology graph. *BMC Bioinformatics*, 9: 525.
- Jordan, M.I. (1999). *Learning in Graphical Models*. MIT Press, Cambridge.
- Kinderman, R., and Snell, J.L. (1980). *Markov Random Fields and Their Applications*. American Mathematical Society. ISBN 0-8218-5001-6. MR0620955.
- Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. *Proceedings of the International Joint Conference on Artificial Intelligence*, 1137-1143.
- Koller, D. and Friedman, N. (2009). *Probabilistic Graphical Models: Principles and Techniques*. The MIT Press, Cambridge, MA.
- Li, S.Z. (2009). *Markov Random Field Modeling in Image Analysis, Advances in Pattern Recognition*. Springer-Verlag London Limited, 3rd
- Li, T. & Ogihara, M. (2006). Toward intelligent music information retrieval. *IEEE Transactions on Multimedia*, 8(3): 564-574.
- Macpherson, A.J., Harris, N.L. (2004). Interactions between commensal intestinal bacteria and the immune system. *Nature Reviews Immunology* 4: 478-485.
- McCulloch, C.E. & Searle, S.R. (2001). *Generalized, Linear and Mixed Models*, Wiley, New York.
- Meyn, S.P. and Tweedie, R.L. (1993). *Markov Chains and Stochastic Stability*, New York: Springer-Verlag.

- Pearl, J. (1988): Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann.
- Read, J. (2008). A pruned problem transformation method for multi-label classification. Proceedings of New Zealand Computer Science Research Student Conference, 143-150.
- Read, J., Pfahringer, B. and Holmes, G. (2008). Multi-label classification using ensembles of pruned sets. Proceeding of the 8th IEEE International Conference on Data Mining, 995-1000.
- Rousu, J., Saunders, C., Szedmak, S., and Shawe-Taylor, J. (2006). Kernel-based learning of hierarchical multilabel classification models. The Journal of Machine Learning Research, 7: 1601-1626.
- Santos, A., Canuto, A. and Neto, A. (2011). A comparative analysis of classification methods to multi-label tasks in different application domains. International Journal of Computer Information Systems and Industrial Management Applications, 3: 218-227.
- Schapire, R.E & Singer, Y. (1999). Improved boosting algorithms using confidence-rated predictions. Machine Learning, 37(3): 1-40.
- Schietgat, L., Vens, C., Struyf, J., Blockeel, H., Kocev, D., and Dzeroski, S. (2010). Predicting gene function using hierarchical multi-label decision tree ensembles. BMC Bioinformatics 11(1):2.
- Strauss, D.J. (1977). Clustering on Colored Lattice. Journal of Applied Probability 14: 135-143.
- Trohidis, K., Tsoumakas, G., Kalliris, G., and Vlahavas, I. (2008). Multilabel classification of music into emotions. Proceeding 9th International Conference on Music Information Retrieval (ISMIR 2008): 325-330.
- Tsoumakas, G., Katakis, I., and Vlahavas, I. (2010). Mining multi-label data. Data Mining and Knowledge Discovery Handbook, O. Maimon, L. Rokach (Ed.). Springer, 2nd edition.
- Tsoumakas, G., Katakis, I. and Vlahavas, I.P (2011). Random k-Labelsets for multilabel classification. IEEE Transactions on Knowledge and Data Engineering, 23(7): 1079-1089.
- Ueda, N. & Saito, K. (2003). Parametric mixture models for multi-labeled text. Advances in Neural Information Processing Systems, 15: 721-728.

- Valinsky, L., Vedova, G. D., Jiang, T., and Borneman, J. (2002). Oligonucleotide fingerprinting of rRNA genes for analysis of fungal community composition. *Applied and Environmental Microbiology* 68 (12): 5999-6004.
- Valinsky, L., Vedova, G. D., Scumpham, A. J., Alvey, S., Figueroa, A., Yin, B., Hartin, J., Chrobak, M., Crowley, D. E., Jiang, T. and Borneman, J. (2002). Analysis of bacterial community composition by oligonucleotide fingerprinting of rRNA genes. *Applied and Environmental Microbiology* 68 (7): 3243-3250.
- Wieczorkowska, A., Synak, P., & Ras, Z.W. (2006). Multi-label classification of emotions in music. *Proceedings of the International Conference on Intelligent Information Processing and Web Mining (IIPWM 2006)*: 307-315.
- Wolfinger, R. and O'Connell, M. (1993). Generalized linear mixed models: a pseudo-likelihood approach. *Journal of Statistical Computation and Simulation* 48: 233-243.
- Yang, Y., Webb, G.I., Cerquides, J., Korb, K.B., Boughton, J. and Ting, K.M. (2007). To select or to weigh: A comparative study of linear combination schemes for superparent-one-dependence estimators. *IEEE Transactions on Knowledge and Data Engineering*, 19(12): 1652-1665.
- Yu, H., Jeske, D. R., Ruegger, P., Borneman, J. (2010). Neutral Zone Classifier Using a Decision-Theoretic Approach With Application to DNA Array Analyses in the following paginated issue of *Journal of Agricultural Biological, and Environmental Statistics* 15:474.
- Zhang, M.L. & Zhou, Z.H. (2007). ML-kNN: A lazy learning approach to multi-label learning. *Pattern Recognition*, 40: 2038-2048.

Appendices

A. Proof of the Theorem

Define $C = \{C_{jk}\}$, $x = \{x_{jk}\}$, where $j = 1 \dots N$ and $k = 1 \dots K$, as a set of class labels and its associated covariate measurements of the training data, respectively, at K different experimental occasions. Let $C^* = \{C_k\}$ represent a vector of class labels for a new subject. Then, given $\beta = \{\beta_1, \beta_2, \dots, \beta_K\}$ and Σ , let $\theta = (\beta, \Sigma)$. Then, the joint predictive distribution of the new subject C^* is:

$$P(C^*|\theta, x) = \iint \prod_{k=1}^K \frac{\exp\{[b_0 + \beta_{0k} + (b_1 + \beta_{1k})x_k]C_k^*\}}{1 + \exp[b_0 + \beta_{0k} + (b_1 + \beta_{1k})x_k]} f(b) db \quad (A.1)$$

when we have integrated out the unobservable subject-specific random effects $b = (b_0 + b_1)'$. We next sum for the components of C^* . For example,

$$\begin{aligned} P(C_1^* = 1|\theta, x_1) \\ &= \sum_{C_2^*=0}^1 \dots \sum_{C_K^*=0}^1 \iint \prod_{k=1}^K \frac{\exp\{[b_0 + \beta_{0k} + (b_1 + \beta_{1k})x_k]C_k^*\}}{1 + \exp[b_0 + \beta_{0k} + (b_1 + \beta_{1k})x_k]} f(b) db \\ &= \sum_{C_2^*=0}^1 \dots \sum_{C_K^*=0}^1 \iint \frac{\exp[b_0 + \beta_{01} + (b_1 + \beta_{11})x_1]}{1 + \exp[b_0 + \beta_{01} + (b_1 + \beta_{11})x_1]} \\ &\quad \times \prod_{k=2}^K \frac{\exp\{[b_0 + \beta_{0k} + (b_1 + \beta_{1k})x_k]C_k^*\}}{1 + \exp[b_0 + \beta_{0k} + (b_1 + \beta_{1k})x_k]} f(b) db \end{aligned}$$

$$\begin{aligned}
&= \iint \frac{\exp[b_0 + \beta_{01} + (b_1 + \beta_{11})x_1]}{1 + \exp[b_0 + \beta_{01} + (b_1 + \beta_{11})x_1]} \\
&\quad \times \left[\sum_{C_2^*=0}^1 \cdots \sum_{C_K^*=0}^1 \prod_{k=2}^K \frac{\exp\{[b_0 + \beta_{0k} + (b_1 + \beta_{1k})x_k]C_k^*\}}{1 + \exp[b_0 + \beta_{0k} + (b_1 + \beta_{1k})x_k]} \right] f(b) db \\
&= \iint \frac{\exp[b_0 + \beta_{01} + (b_1 + \beta_{11})x_1]}{1 + \exp[b_0 + \beta_{01} + (b_1 + \beta_{11})x_1]} f(b) db
\end{aligned}$$

Let $P(\theta|C, x)$ denote a posterior distribution for θ , given the training data. Then the marginal predictive distribution for C_1^* is:

$$P(C_1^* = 1|x_1) = \int P(C_1^* = 1|\theta, x_1)P(\theta|C, x) d\theta \quad (A.2)$$

If we pick a degenerate prior for θ that puts all its mass on $\hat{\theta}$, the integral (A.2) simplifies to

$$P(C_1^* = 1|x_1) = P(C_1^* = 1|\hat{\theta}, x_1) \quad (A.3)$$

and the right-hand-side of (A.3) is just the estimated expected probability that $C_1^* = 1$, as defined in section 2.3.

B. Proof of the Markov-Gibbs Equivalence

To show the equivalence between a MRF and a Gibbs distribution, the following statement should be proven:

$$P(l_k|l_{V-\{v_k\}}) = P(l_k|l_{N_k}) \text{ if and only if } P(l) = \frac{1}{Z} \exp\left[-\frac{1}{T} U(l)\right]$$

Refer to (Besag, J. 1974), (Kindermann and Snell 1980) and (Li 2009) for the complete proof of the theorem. Here, we only show that a Gibbs distribution is an MRF. Consider

$P(l)$ as a Gibbs distribution on V with respect to the neighborhood system N , then the conditional probability at a node k is defined as:

$$P(l_k | l_{V-\{V_k\}}) = \frac{P(l_k, l_{V-\{V_k\}})}{P(l_{V-\{V_k\}})} = \frac{P(l)}{\sum_{l'_k=0}^1 P(l')} \quad (B.1)$$

where $l' = \{l_1, \dots, l_{k-1}, l'_k, l_{k+1}, \dots, l_K\}$ is the same as any configuration of l except possibly node k . Using equations (3.3) and (3.5) yields

$$P(l) = \frac{1}{Z} \exp \left[- \sum_{c \in c(G)} \phi_{V_c}(l) \right] \quad (B.2)$$

and substituting (C.2) into (C.1) gives

$$P(l_k | l_{V-\{V_k\}}) = \frac{\exp[-\sum_{c \in c_k(G)} \phi_{V_c}(l)]}{\sum_{l'_k=0}^1 \exp[-\sum_{c \in c(G)} \phi_{V_c}(l')]} \quad (B.3)$$

Split the clique set $c(G)$ into two disjoint subsets defined as $c_k(G) = \{c \in c(G) | k \in c\}$ that includes node k and $c_{\bar{k}}(G) = \{c \in c(G) | k \notin c\}$, otherwise do not. Hence, the equation (B.3) can be written as

$$P(l_k | l_{V-\{V_k\}}) = \frac{\exp[-\sum_{c \in c_k(G)} \phi_{V_c}(l)] \exp[-\sum_{c \in c_{\bar{k}}(G)} \phi_{V_c}(l)]}{\sum_{l'_k=0}^1 \left\{ \exp[-\sum_{c \in c_k(G)} \phi_{V_c}(l')] \exp[-\sum_{c \in c_{\bar{k}}(G)} \phi_{V_c}(l')] \right\}}$$

Note that clique potentials $\phi_{V_c}(l)$ is equivalent to $\phi_{V_c}(l')$ when the node $k \notin c$, then the conditional probability $P(l_k | l_{V-\{V_k\}})$ becomes

$$P(l_k | l_{V-\{V_k\}}) = \frac{\exp[-\sum_{c \in c_k(G)} \phi_{V_c}(l)]}{\sum_{l'_k=0}^1 \exp[-\sum_{c \in c_k(G)} \phi_{V_c}(l')]} \quad (B.4)$$

The Equation (B.4) illustrates a conditional probability at node k that depends only on values at k and its neighbors. In addition, the exponential format of the joint probability

$P(l)$ from (B.2) guarantees that the positive property of an MRF is satisfied. Hence, a Gibbs Random Field determines an MRF.

C. Analysis of Simulated Data

Σ_x	Method	Sample Size N							
		Hard Error Rate				Soft Error Rate			
		100	200	500	1000	100	200	500	1000
AR1(0.4)	nLNZC	0.140	0.096	0.092	0.085	0.121	0.077	0.085	0.075
	LNZC	0.172	0.094	0.092	0.078	0.110	0.083	0.084	0.069
	MNZC1	0.151	0.124	0.116	0.091	0.120	0.096	0.091	0.077
	MNZC2	0.147	0.129	0.121	0.099	0.141	0.091	0.088	0.065
AR1(0.7)	nLNZC	0.132	0.096	0.092	0.080	0.132	0.077	0.084	0.071
	LNZC	0.176	0.093	0.092	0.067	0.141	0.083	0.084	0.073
	MNZC1	0.155	0.128	0.120	0.094	0.123	0.090	0.087	0.062
	MNZC2	0.163	0.130	0.129	0.110	0.132	0.068	0.070	0.041
CS(0.4)	nLNZC	0.138	0.095	0.091	0.078	0.140	0.077	0.086	0.065
	LNZC	0.165	0.092	0.091	0.075	0.152	0.086	0.087	0.063
	MNZC1	0.149	0.122	0.112	0.105	0.146	0.075	0.069	0.051
	MNZC2	0.158	0.128	0.125	0.111	0.129	0.073	0.061	0.044
CS(0.7)	nLNZC	0.151	0.094	0.090	0.084	0.143	0.078	0.086	0.074
	LNZC	0.173	0.091	0.090	0.077	0.149	0.093	0.086	0.065
	MNZC1	0.148	0.132	0.120	0.096	0.136	0.088	0.081	0.058
	MNZC2	0.152	0.139	0.131	0.113	0.132	0.081	0.073	0.053
Diagonal	nLNZC	0.147	0.095	0.092	0.091	0.120	0.081	0.086	0.076
	LNZC	0.169	0.092	0.092	0.087	0.131	0.090	0.086	0.065
	MNZC1	0.137	0.088	0.083	0.068	0.128	0.093	0.090	0.077
	MNZC2	0.138	0.110	0.108	0.096	0.135	0.070	0.061	0.038

Table C.1: Hard and soft error rates of proposed classification methods using the GLMM simulated data when $K = 5$ and $\rho_b = 0$

Σ_x	Method	Total Error Rate			
		$N = 100$	$N = 200$	$N = 500$	$N = 1000$
AR1(0.4)	nLNZC	0.261	0.173	0.176	0.160
	LNZC	0.282	0.177	0.176	0.147
	MNZC1	0.271	0.220	0.207	0.168
	MNZC2	0.288	0.220	0.209	0.164
AR1(0.7)	nLNZC	0.264	0.172	0.176	0.151
	LNZC	0.317	0.176	0.176	0.140
	MNZC1	0.278	0.218	0.207	0.156
	MNZC2	0.295	0.198	0.199	0.151
CS(0.4)	nLNZC	0.278	0.173	0.177	0.143
	LNZC	0.317	0.178	0.178	0.138
	MNZC1	0.295	0.197	0.181	0.156
	MNZC2	0.287	0.201	0.186	0.155
CS(0.7)	nLNZC	0.294	0.172	0.176	0.158
	LNZC	0.322	0.184	0.176	0.142
	MNZC1	0.284	0.220	0.201	0.154
	MNZC2	0.284	0.220	0.204	0.166
Diagonal	nLNZC	0.267	0.175	0.178	0.167
	LNZC	0.300	0.182	0.178	0.152
	MNZC1	0.265	0.181	0.173	0.145
	MNZC2	0.273	0.180	0.169	0.134

Table C.2: Total error rate of proposed classification methods using the GLMM simulated data when $K = 5$ and $\rho_b = 0$

Σ_x	Method	Sample Size N							
		Hard Error Rate				Soft Error Rate			
		100	200	500	1000	100	200	500	1000
AR1(0.4)	nLNZC	0.138	0.093	0.090	0.082	0.120	0.079	0.084	0.074
	LNZC	0.173	0.093	0.091	0.074	0.113	0.080	0.082	0.063
	MNZC1	0.148	0.122	0.115	0.090	0.119	0.095	0.090	0.076
	MNZC2	0.148	0.128	0.120	0.098	0.140	0.092	0.088	0.064
AR1(0.7)	nLNZC	0.131	0.093	0.090	0.081	0.132	0.078	0.084	0.067
	LNZC	0.173	0.092	0.091	0.063	0.142	0.083	0.082	0.070
	MNZC1	0.153	0.128	0.122	0.095	0.122	0.089	0.086	0.062
	MNZC2	0.162	0.129	0.128	0.108	0.130	0.071	0.071	0.040
CS(0.4)	nLNZC	0.137	0.094	0.090	0.079	0.141	0.077	0.083	0.064
	LNZC	0.164	0.093	0.090	0.072	0.150	0.080	0.083	0.062
	MNZC1	0.149	0.121	0.111	0.094	0.145	0.074	0.068	0.050
	MNZC2	0.157	0.127	0.123	0.112	0.127	0.071	0.060	0.043
CS(0.7)	nLNZC	0.150	0.093	0.089	0.083	0.142	0.078	0.085	0.072
	LNZC	0.172	0.092	0.089	0.072	0.147	0.081	0.084	0.064
	MNZC1	0.147	0.132	0.118	0.095	0.135	0.085	0.082	0.056
	MNZC2	0.152	0.136	0.130	0.112	0.131	0.080	0.070	0.052
Diagonal	nLNZC	0.146	0.094	0.090	0.089	0.119	0.078	0.084	0.074
	LNZC	0.167	0.092	0.089	0.083	0.130	0.081	0.084	0.064
	MNZC1	0.134	0.088	0.082	0.069	0.125	0.090	0.091	0.076
	MNZC2	0.136	0.109	0.106	0.094	0.132	0.071	0.058	0.041

Table C.3: Hard and soft error rates of proposed classification methods using the GLMM simulated data when $K = 5$ and $\rho_b = 0.4$

Σ_x	Method	Total Error Rate			
		$N = 100$	$N = 200$	$N = 500$	$N = 1000$
AR1(0.4)	nLNZC	0.258	0.171	0.175	0.156
	LNZC	0.286	0.173	0.173	0.137
	MNZC1	0.267	0.217	0.205	0.166
	MNZC2	0.288	0.220	0.208	0.162
AR1(0.7)	nLNZC	0.263	0.172	0.174	0.148
	LNZC	0.315	0.175	0.173	0.133
	MNZC1	0.275	0.217	0.208	0.157
	MNZC2	0.292	0.200	0.199	0.148
CS(0.4)	nLNZC	0.278	0.171	0.174	0.143
	LNZC	0.314	0.172	0.173	0.134
	MNZC1	0.294	0.195	0.179	0.144
	MNZC2	0.284	0.198	0.183	0.155
CS(0.7)	nLNZC	0.292	0.171	0.174	0.155
	LNZC	0.319	0.173	0.173	0.136
	MNZC1	0.282	0.217	0.200	0.151
	MNZC2	0.283	0.216	0.200	0.164
Diagonal	nLNZC	0.265	0.172	0.174	0.163
	LNZC	0.297	0.173	0.173	0.147
	MNZC1	0.259	0.178	0.172	0.160
	MNZC2	0.268	0.180	0.164	0.135

Table C.4: Total error rate of proposed classification methods using the GLMM simulated data when $K = 5$ and $\rho_b = 0.4$

Σ_x	Method	Sample Size N							
		Hard Error Rate				Soft Error Rate			
		100	200	500	1000	100	200	500	1000
AR1(0.4)	nLNZC	0.139	0.092	0.089	0.080	0.125	0.076	0.082	0.073
	LNZC	0.172	0.092	0.089	0.077	0.111	0.079	0.084	0.061
	MNZC1	0.143	0.120	0.113	0.088	0.118	0.094	0.089	0.075
	MNZC2	0.146	0.127	0.121	0.094	0.138	0.095	0.086	0.066
AR1(0.7)	nLNZC	0.130	0.092	0.089	0.080	0.130	0.076	0.083	0.077
	LNZC	0.170	0.092	0.089	0.064	0.141	0.074	0.081	0.073
	MNZC1	0.154	0.127	0.125	0.093	0.123	0.087	0.083	0.060
	MNZC2	0.167	0.126	0.126	0.104	0.129	0.070	0.070	0.043
CS(0.4)	nLNZC	0.136	0.092	0.090	0.078	0.140	0.074	0.080	0.069
	LNZC	0.164	0.087	0.089	0.071	0.147	0.093	0.081	0.058
	MNZC1	0.148	0.120	0.110	0.090	0.146	0.072	0.066	0.048
	MNZC2	0.159	0.125	0.124	0.105	0.120	0.070	0.058	0.040
CS(0.7)	nLNZC	0.151	0.092	0.088	0.082	0.139	0.076	0.084	0.081
	LNZC	0.170	0.092	0.088	0.069	0.143	0.075	0.083	0.063
	MNZC1	0.141	0.130	0.120	0.093	0.130	0.084	0.080	0.054
	MNZC2	0.150	0.138	0.127	0.106	0.125	0.080	0.067	0.050
Diagonal	nLNZC	0.140	0.093	0.089	0.087	0.116	0.076	0.083	0.076
	LNZC	0.162	0.092	0.090	0.079	0.131	0.077	0.078	0.063
	MNZC1	0.130	0.080	0.080	0.068	0.127	0.087	0.090	0.070
	MNZC2	0.134	0.105	0.100	0.090	0.130	0.070	0.056	0.043

Table C.5: Hard and soft error rates of proposed classification methods using the GLMM simulated data when $K = 5$ and $\rho_b = 0.7$

Σ_x	Method	Total Error Rate			
		$N = 100$	$N = 200$	$N = 500$	$N = 1000$
AR1(0.4)	nLNZC	0.264	0.168	0.171	0.153
	LNZC	0.283	0.171	0.173	0.138
	MNZC1	0.261	0.214	0.202	0.163
	MNZC2	0.284	0.222	0.207	0.160
AR1(0.7)	nLNZC	0.260	0.168	0.171	0.157
	LNZC	0.311	0.167	0.170	0.137
	MNZC1	0.277	0.214	0.208	0.153
	MNZC2	0.296	0.196	0.196	0.147
CS(0.4)	nLNZC	0.276	0.166	0.170	0.147
	LNZC	0.311	0.181	0.170	0.129
	MNZC1	0.294	0.192	0.176	0.138
	MNZC2	0.279	0.195	0.182	0.145
CS(0.7)	nLNZC	0.290	0.168	0.172	0.163
	LNZC	0.313	0.167	0.171	0.132
	MNZC1	0.271	0.214	0.200	0.147
	MNZC2	0.275	0.218	0.194	0.156
Diagonal	nLNZC	0.256	0.168	0.171	0.163
	LNZC	0.293	0.170	0.168	0.142
	MNZC1	0.257	0.167	0.170	0.138
	MNZC2	0.264	0.175	0.156	0.133

Table C.6: Total error rates of proposed classification methods using the GLMM simulated data when $K = 5$ and $\rho_b = 0.7$

Σ_x	Method	Sample Size N							
		Hard Error Rate				Soft Error Rate			
		100	200	500	1000	100	200	500	1000
AR1(0.4)	nLNZC	0.172	0.094	0.091	0.082	0.150	0.078	0.084	0.074
	LNZC	0.220	0.092	0.089	0.073	0.177	0.083	0.086	0.069
	MNZC1	0.183	0.096	0.093	0.079	0.147	0.110	0.096	0.072
	MNZC2	0.179	0.107	0.102	0.081	0.153	0.106	0.092	0.063
AR1(0.7)	nLNZC	0.168	0.093	0.090	0.085	0.153	0.078	0.085	0.067
	LNZC	0.196	0.091	0.089	0.071	0.210	0.083	0.087	0.069
	MNZC1	0.158	0.104	0.099	0.085	0.154	0.099	0.088	0.051
	MNZC2	0.167	0.121	0.110	0.091	0.162	0.117	0.097	0.059
CS(0.4)	nLNZC	0.180	0.094	0.090	0.095	0.149	0.079	0.085	0.075
	LNZC	0.175	0.091	0.089	0.077	0.199	0.085	0.086	0.072
	MNZC1	0.163	0.107	0.097	0.088	0.154	0.108	0.088	0.064
	MNZC2	0.170	0.110	0.095	0.079	0.171	0.110	0.092	0.063
CS(0.7)	nLNZC	0.193	0.094	0.091	0.088	0.152	0.078	0.084	0.073
	LNZC	0.231	0.091	0.090	0.074	0.221	0.085	0.087	0.074
	MNZC1	0.179	0.129	0.112	0.097	0.181	0.110	0.094	0.059
	MNZC2	0.182	0.128	0.119	0.102	0.179	0.107	0.089	0.068
Diagonal	nLNZC	0.158	0.094	0.091	0.087	0.164	0.080	0.084	0.076
	LNZC	0.240	0.092	0.090	0.086	0.186	0.084	0.086	0.065
	MNZC1	0.172	0.126	0.103	0.082	0.163	0.118	0.102	0.077
	MNZC2	0.168	0.104	0.096	0.078	0.171	0.120	0.098	0.073

Table C.7: Hard and soft error rates of proposed classification methods using the GLMM simulated data when $K = 10$ and $\rho_b = 0$

Σ_x	Method	Total Error Rate			
		$N = 100$	$N = 200$	$N = 500$	$N = 1000$
AR1(0.4)	nLNZC	0.322	0.172	0.175	0.156
	LNZC	0.397	0.175	0.176	0.142
	MNZC1	0.330	0.206	0.189	0.151
	MNZC2	0.332	0.213	0.194	0.144
AR1(0.7)	nLNZC	0.321	0.172	0.175	0.152
	LNZC	0.406	0.174	0.175	0.140
	MNZC1	0.312	0.203	0.187	0.136
	MNZC2	0.329	0.238	0.207	0.150
CS(0.4)	nLNZC	0.329	0.173	0.175	0.170
	LNZC	0.374	0.176	0.176	0.149
	MNZC1	0.317	0.215	0.185	0.152
	MNZC2	0.341	0.220	0.187	0.142
CS(0.7)	nLNZC	0.345	0.172	0.175	0.161
	LNZC	0.452	0.176	0.176	0.148
	MNZC1	0.360	0.239	0.206	0.156
	MNZC2	0.361	0.235	0.208	0.170
Diagonal	nLNZC	0.322	0.173	0.175	0.163
	LNZC	0.426	0.176	0.176	0.151
	MNZC1	0.335	0.244	0.205	0.159
	MNZC2	0.339	0.224	0.194	0.151

Table C.8: Total error rate of proposed classification methods using the GLMM simulated data when $K = 10$ and $\rho_b = 0$

Σ_x	Method	Sample Size N							
		Hard Error Rate				Soft Error Rate			
		100	200	500	1000	100	200	500	1000
AR1(0.4)	nLNZC	0.170	0.091	0.089	0.081	0.152	0.076	0.081	0.077
	LNZC	0.217	0.089	0.087	0.074	0.175	0.081	0.084	0.063
	MNZC1	0.177	0.095	0.090	0.081	0.150	0.109	0.095	0.071
	MNZC2	0.178	0.099	0.099	0.080	0.152	0.103	0.090	0.064
AR1(0.7)	nLNZC	0.166	0.092	0.087	0.087	0.148	0.076	0.084	0.068
	LNZC	0.195	0.089	0.087	0.070	0.206	0.081	0.085	0.063
	MNZC1	0.174	0.103	0.100	0.082	0.156	0.098	0.087	0.049
	MNZC2	0.172	0.118	0.109	0.090	0.161	0.115	0.096	0.058
CS(0.4)	nLNZC	0.177	0.092	0.089	0.093	0.155	0.077	0.083	0.076
	LNZC	0.181	0.089	0.088	0.072	0.200	0.085	0.084	0.071
	MNZC1	0.158	0.106	0.110	0.086	0.150	0.107	0.087	0.064
	MNZC2	0.167	0.106	0.116	0.078	0.166	0.111	0.090	0.060
CS(0.7)	nLNZC	0.190	0.092	0.088	0.089	0.149	0.077	0.083	0.074
	LNZC	0.228	0.089	0.087	0.075	0.199	0.084	0.085	0.071
	MNZC1	0.173	0.126	0.101	0.095	0.183	0.107	0.093	0.058
	MNZC2	0.181	0.129	0.098	0.103	0.176	0.108	0.088	0.066
Diagonal	nLNZC	0.155	0.092	0.089	0.083	0.165	0.077	0.082	0.077
	LNZC	0.237	0.090	0.088	0.082	0.185	0.082	0.084	0.061
	MNZC1	0.170	0.125	0.102	0.080	0.158	0.116	0.100	0.076
	MNZC2	0.166	0.104	0.097	0.077	0.166	0.119	0.095	0.070

Table C.9: Hard and soft error rates of proposed classification methods using the GLMM simulated data when $K = 10$ and $\rho_b = 0.4$

Σ_x	Method	Total Error Rate			
		$N = 100$	$N = 200$	$N = 500$	$N = 1000$
AR1(0.4)	nLNZC	0.322	0.168	0.170	0.158
	LNZC	0.392	0.170	0.171	0.137
	MNZC1	0.327	0.204	0.185	0.152
	MNZC2	0.330	0.202	0.189	0.144
AR1(0.7)	nLNZC	0.314	0.168	0.171	0.155
	LNZC	0.401	0.171	0.172	0.133
	MNZC1	0.330	0.201	0.187	0.131
	MNZC2	0.333	0.233	0.205	0.148
CS(0.4)	nLNZC	0.332	0.169	0.172	0.169
	LNZC	0.381	0.173	0.172	0.143
	MNZC1	0.308	0.213	0.197	0.150
	MNZC2	0.333	0.217	0.206	0.138
CS(0.7)	nLNZC	0.339	0.169	0.171	0.163
	LNZC	0.427	0.173	0.172	0.146
	MNZC1	0.356	0.233	0.194	0.153
	MNZC2	0.357	0.237	0.186	0.169
Diagonal	nLNZC	0.320	0.169	0.171	0.160
	LNZC	0.422	0.172	0.172	0.143
	MNZC1	0.328	0.241	0.202	0.156
	MNZC2	0.332	0.223	0.192	0.147

Table C.10: Total error rate of proposed classification methods using the GLMM simulated data when $K = 10$ and $\rho_b = 0.4$

Σ_x	Method	Sample Size N							
		Hard Error Rate				Soft Error Rate			
		100	200	500	1000	100	200	500	1000
AR1(0.4)	nLNZC	0.168	0.090	0.086	0.082	0.150	0.072	0.079	0.075
	LNZC	0.216	0.087	0.085	0.071	0.174	0.078	0.081	0.061
	MNZC1	0.170	0.094	0.091	0.082	0.148	0.106	0.094	0.070
	MNZC2	0.175	0.097	0.098	0.083	0.150	0.100	0.089	0.065
AR1(0.7)	nLNZC	0.166	0.090	0.085	0.085	0.146	0.073	0.080	0.066
	LNZC	0.194	0.087	0.084	0.071	0.204	0.079	0.082	0.060
	MNZC1	0.170	0.100	0.102	0.080	0.155	0.096	0.085	0.048
	MNZC2	0.170	0.116	0.109	0.089	0.160	0.110	0.095	0.056
CS(0.4)	nLNZC	0.175	0.090	0.086	0.091	0.150	0.073	0.081	0.074
	LNZC	0.180	0.087	0.085	0.070	0.197	0.081	0.083	0.070
	MNZC1	0.156	0.105	0.112	0.085	0.149	0.107	0.090	0.056
	MNZC2	0.165	0.104	0.115	0.077	0.164	0.110	0.091	0.063
CS(0.7)	nLNZC	0.190	0.090	0.086	0.086	0.147	0.074	0.081	0.078
	LNZC	0.225	0.088	0.085	0.074	0.197	0.080	0.083	0.072
	MNZC1	0.173	0.123	0.102	0.094	0.180	0.106	0.090	0.055
	MNZC2	0.180	0.126	0.096	0.102	0.174	0.103	0.084	0.064
Diagonal	nLNZC	0.153	0.090	0.086	0.085	0.163	0.074	0.080	0.074
	LNZC	0.220	0.087	0.085	0.078	0.182	0.079	0.082	0.057
	MNZC1	0.167	0.124	0.101	0.079	0.153	0.114	0.102	0.073
	MNZC2	0.165	0.100	0.095	0.076	0.167	0.120	0.094	0.068

Table C.11: Hard and soft error rates of proposed classification methods using the GLMM simulated data when $K = 10$ and $\rho_b = 0.7$

Σ_x	Method	Total Error Rate			
		$N = 100$	$N = 200$	$N = 500$	$N = 1000$
AR1(0.4)	nLNZC	0.318	0.162	0.166	0.157
	LNZC	0.390	0.165	0.166	0.132
	MNZC1	0.318	0.200	0.185	0.152
	MNZC2	0.325	0.197	0.187	0.148
AR1(0.7)	nLNZC	0.312	0.163	0.166	0.151
	LNZC	0.398	0.166	0.167	0.131
	MNZC1	0.325	0.196	0.187	0.128
	MNZC2	0.330	0.226	0.204	0.145
CS(0.4)	nLNZC	0.325	0.163	0.167	0.165
	LNZC	0.377	0.168	0.168	0.140
	MNZC1	0.305	0.212	0.202	0.141
	MNZC2	0.329	0.214	0.206	0.140
CS(0.7)	nLNZC	0.337	0.164	0.167	0.164
	LNZC	0.422	0.168	0.168	0.146
	MNZC1	0.353	0.229	0.192	0.149
	MNZC2	0.354	0.229	0.180	0.166
Diagonal	nLNZC	0.316	0.164	0.167	0.159
	LNZC	0.402	0.167	0.167	0.135
	MNZC1	0.320	0.238	0.203	0.152
	MNZC2	0.332	0.220	0.189	0.144

Table C.12: Total error rate of proposed classification methods using the GLMM simulated data when $K = 10$ and $\rho_b = 0.7$

Σ_x	Method	Sample Size N							
		Hard Error Rate				Soft Error Rate			
		100	200	500	1000	100	200	500	1000
AR1(0.4)	nLNZC	0.170	0.061	0.059	0.055	0.140	0.041	0.045	0.040
	LNZC	0.197	0.061	0.058	0.049	0.182	0.044	0.047	0.040
	MNZC1	0.146	0.077	0.063	0.052	0.121	0.057	0.054	0.039
	MNZC2	0.162	0.068	0.059	0.056	0.134	0.048	0.049	0.040
AR1(0.7)	nLNZC	0.179	0.076	0.072	0.068	0.139	0.055	0.062	0.066
	LNZC	0.230	0.088	0.084	0.052	0.167	0.066	0.075	0.052
	MNZC1	0.161	0.083	0.072	0.054	0.116	0.061	0.059	0.050
	MNZC2	0.153	0.086	0.077	0.057	0.128	0.068	0.064	0.047
CS(0.4)	nLNZC	0.167	0.066	0.062	0.061	0.163	0.046	0.052	0.055
	LNZC	0.210	0.067	0.063	0.053	0.170	0.052	0.056	0.047
	MNZC1	0.148	0.075	0.066	0.060	0.135	0.058	0.057	0.034
	MNZC2	0.161	0.079	0.061	0.059	0.142	0.066	0.062	0.029
CS(0.7)	nLNZC	0.180	0.081	0.077	0.080	0.172	0.060	0.068	0.062
	LNZC	0.248	0.119	0.119	0.053	0.188	0.106	0.113	0.059
	MNZC1	0.163	0.120	0.096	0.062	0.153	0.085	0.062	0.047
	MNZC2	0.171	0.119	0.088	0.056	0.148	0.093	0.072	0.038
Diagonal	nLNZC	0.169	0.064	0.059	0.061	0.136	0.042	0.052	0.045
	LNZC	0.220	0.062	0.058	0.052	0.192	0.047	0.053	0.043
	MNZC1	0.148	0.059	0.065	0.049	0.130	0.051	0.054	0.056
	MNZC2	0.154	0.053	0.064	0.058	0.138	0.053	0.056	0.041

Table C.13: Hard and soft error rates of proposed classification methods using the MRF simulated data when $K = 5$

Σ_x	Method	Total Error Rate			
		$N = 100$	$N = 200$	$N = 500$	$N = 1000$
AR1(0.4)	nLNZC	0.310	0.102	0.104	0.095
	LNZC	0.379	0.105	0.105	0.089
	MNZC1	0.267	0.134	0.117	0.091
	MNZC2	0.296	0.116	0.108	0.096
AR1(0.7)	nLNZC	0.318	0.130	0.134	0.134
	LNZC	0.397	0.154	0.159	0.104
	MNZC1	0.277	0.144	0.131	0.104
	MNZC2	0.281	0.154	0.141	0.104
CS(0.4)	nLNZC	0.330	0.112	0.114	0.116
	LNZC	0.380	0.119	0.119	0.100
	MNZC1	0.283	0.133	0.123	0.094
	MNZC2	0.303	0.145	0.123	0.088
CS(0.7)	nLNZC	0.352	0.142	0.145	0.142
	LNZC	0.436	0.224	0.232	0.112
	MNZC1	0.316	0.205	0.158	0.109
	MNZC2	0.319	0.212	0.160	0.094
Diagonal	nLNZC	0.305	0.106	0.110	0.106
	LNZC	0.412	0.109	0.111	0.095
	MNZC1	0.278	0.110	0.119	0.105
	MNZC2	0.292	0.106	0.120	0.099

Table C.14: Total error rate of proposed classification methods using the MRF simulated data when $K = 5$

Σ_x	Method	Sample Size N							
		Hard Error Rate				Soft Error Rate			
		100	200	500	1000	100	200	500	1000
AR1(0.4)	nLNZC	0.160	0.063	0.059	0.065	0.084	0.040	0.042	0.045
	LNZC	0.178	0.065	0.059	0.051	0.105	0.046	0.044	0.048
	MNZC1	0.110	0.071	0.068	0.049	0.084	0.069	0.055	0.043
	MNZC2	0.130	0.068	0.065	0.056	0.085	0.077	0.062	0.036
AR1(0.7)	nLNZC	0.097	0.039	0.035	0.041	0.061	0.009	0.006	0.005
	LNZC	0.115	0.037	0.037	0.033	0.082	0.011	0.009	0.010
	MNZC1	0.083	0.051	0.036	0.032	0.055	0.013	0.010	0.008
	MNZC2	0.073	0.047	0.031	0.029	0.063	0.015	0.011	0.007
CS(0.4)	nLNZC	0.136	0.065	0.061	0.060	0.076	0.042	0.050	0.051
	LNZC	0.194	0.068	0.066	0.060	0.140	0.048	0.051	0.049
	MNZC1	0.114	0.088	0.073	0.054	0.068	0.051	0.047	0.026
	MNZC2	0.105	0.073	0.057	0.048	0.072	0.056	0.052	0.031
CS(0.7)	nLNZC	0.193	0.184	0.181	0.185	0.172	0.166	0.163	0.140
	LNZC	0.220	0.149	0.143	0.073	0.240	0.119	0.135	0.072
	MNZC1	0.152	0.121	0.106	0.078	0.124	0.075	0.066	0.037
	MNZC2	0.161	0.130	0.110	0.081	0.143	0.082	0.073	0.042
Diagonal	nLNZC	0.146	0.065	0.062	0.066	0.077	0.045	0.049	0.053
	LNZC	0.242	0.182	0.090	0.072	0.130	0.037	0.048	0.044
	MNZC1	0.132	0.092	0.078	0.065	0.081	0.066	0.058	0.048
	MNZC2	0.130	0.088	0.083	0.061	0.079	0.071	0.049	0.036

Table C.15: Hard and soft error rates of proposed classification methods using the MRF simulated data when $K = 10$

Σ_x	Method	Total Error Rate			
		$N = 100$	$N = 200$	$N = 500$	$N = 1000$
AR1(0.4)	nLNZC	0.244	0.103	0.102	0.110
	LNZC	0.283	0.112	0.103	0.099
	MNZC1	0.194	0.140	0.123	0.092
	MNZC2	0.215	0.145	0.127	0.092
AR1(0.7)	nLNZC	0.158	0.048	0.041	0.046
	LNZC	0.197	0.048	0.045	0.043
	MNZC1	0.138	0.064	0.046	0.040
	MNZC2	0.136	0.062	0.042	0.036
CS(0.4)	nLNZC	0.212	0.108	0.111	0.111
	LNZC	0.334	0.116	0.116	0.109
	MNZC1	0.182	0.139	0.120	0.080
	MNZC2	0.177	0.129	0.109	0.079
CS(0.7)	nLNZC	0.365	0.350	0.343	0.325
	LNZC	0.460	0.269	0.278	0.145
	MNZC1	0.276	0.196	0.172	0.115
	MNZC2	0.304	0.212	0.183	0.123
Diagonal	nLNZC	0.223	0.110	0.111	0.119
	LNZC	0.372	0.219	0.138	0.116
	MNZC1	0.213	0.158	0.136	0.113
	MNZC2	0.209	0.159	0.132	0.097

Table C.16: Total error rate of proposed classification methods using the MRF simulated data when $K = 10$

D. Analysis of Fungi Data

Probe ID	Set D ₁ (%)		Set D ₂ (%)		Set D ₃ (%)		Set D ₄ (%)		Set D ₅ (%)	
	0	1	0	1	0	1	0	1	0	1
1	12.8	8.1	13.1	6.7	13.4	6.4	11.9	7.8	14.0	5.8
2	7.8	13.1	6.7	13.1	7.3	12.5	6.4	13.4	6.4	13.4
3	13.4	7.6	11.6	8.1	11.6	8.1	11.6	8.1	11.6	8.1
4	15.1	5.8	14.0	5.8	14.5	5.2	14.0	5.8	13.4	6.4
6	13.7	7.3	15.1	4.7	14.8	4.9	14.2	5.5	14.5	5.2
8	7.8	13.1	7.3	12.5	7.8	11.9	7.0	12.8	7.0	12.8
10	16.9	4.1	16.3	3.5	14.8	4.9	16.0	3.8	16.0	3.8
13	17.4	3.5	16.6	3.2	16.3	3.5	16.6	3.2	16.9	2.9
15	4.1	16.9	5.5	14.2	4.1	15.7	4.1	15.7	4.1	15.7
17	10.5	10.5	11.6	8.1	11.3	8.4	11.9	7.8	10.8	9.0
19	15.4	5.5	16.3	3.5	16.0	3.8	15.7	4.1	16.6	3.2
21	5.8	15.1	4.7	15.1	5.8	14.0	5.5	14.2	5.2	14.5
23	12.8	8.1	13.1	6.7	13.4	6.4	11.6	8.1	14.2	5.5
24	16.0	4.9	13.7	6.1	14.5	5.2	14.2	5.5	14.5	5.2
25	11.6	9.3	11.3	5.5	10.8	9.0	11.0	8.7	12.2	7.6
26	14.0	7.0	13.4	6.4	13.7	6.1	13.7	6.1	13.7	6.1
27	15.7	5.2	13.4	6.4	14.5	5.2	14.8	4.9	14.5	5.2
29	14.2	6.7	13.4	6.4	13.7	6.1	13.7	6.1	13.7	6.1
30	11.0	9.9	11.0	8.7	10.8	9.0	10.8	9.0	9.0	10.8
31	8.7	12.2	9.6	10.2	9.0	10.8	9.6	10.2	11.3	8.4
32	14.0	7.0	13.4	6.4	13.7	6.1	13.7	6.1	13.4	6.4
33	18.0	2.9	16.9	2.9	18.0	1.7	17.2	2.6	16.9	2.9
39	8.4	12.5	9.6	10.2	8.1	11.6	9.0	10.8	9.6	10.2
40	7.8	13.1	7.3	12.5	7.8	11.9	7.0	12.8	6.7	13.1
43	14.8	6.1	13.4	6.4	12.5	7.3	14.2	5.5	14.2	5.5
44	13.7	7.3	15.1	4.7	14.8	4.9	14.2	5.5	14.5	5.2
Total	20.9		19.8		19.8		19.8		19.8	

Table D.1: Data summary of five subsets categorized by hybridization classes

Probe ID	Method				
	LNZC	nLNZC	MNZC1	MNZC2	GNZC
1	0.035	0.038	0.038	0.038	0.041
2	0.020	0.026	0.026	0.023	0.029
3	0.090	0.093	0.099	0.145	0.105
4	0.041	0.047	0.047	0.047	0.047
6	0.061	0.061	0.049	0.061	0.061
8	0.032	0.032	0.032	0.032	0.029
10	0.041	0.041	0.041	0.041	0.052
13	0.058	0.061	0.070	0.076	0.070
15	0.012	0.015	0.198	0.198	0.015
17	0.032	0.032	0.032	0.032	0.023
19	0.044	0.044	0.044	0.044	0.035
21	0.020	0.023	0.032	0.026	0.026
23	0.035	0.038	0.038	0.038	0.044
24	0.044	0.061	0.052	0.061	0.047
25	0.044	0.058	0.044	0.265	0.055
26	0.064	0.052	0.116	0.099	0.064
27	0.044	0.041	0.041	0.052	0.058
29	0.052	0.052	0.047	0.052	0.052
30	0.023	0.026	0.026	0.026	0.026
31	0.049	0.055	0.035	0.038	0.038
32	0.052	0.052	0.102	0.084	0.052
33	0.017	0.017	0.017	0.017	0.032
39	0.017	0.017	0.020	0.017	0.029
40	0.023	0.026	0.029	0.026	0.026
43	0.067	0.067	0.099	0.096	0.067
44	0.064	0.064	0.061	0.064	0.064
Overall	0.042	0.044	0.055	0.065	0.046

Table D.2: Five-fold cross-validation hard error rates of five neutral zone classifiers for the fungi data set.

Probe ID	Method				
	LNZC	nLNZC	MNZC1	MNZC2	GNZC
1	0	0	0	0	0
2	0	0	0	0.003	0
3	0.096	0.087	0.081	0	0.090
4	0	0	0	0	0.006
6	0.017	0	0.049	0	0
8	0	0	0	0	0
10	0	0	0	0	0
13	0.047	0.044	0.015	0.006	0.076
15	0	0	0.020	0	0
17	0	0	0	0	0
19	0	0	0	0	0.044
21	0	0	0	0.006	0
23	0	0	0	0	0
24	0.041	0	0.029	0	0.038
25	0	0	0.035	0.028	0
26	0.041	0.052	0	0.090	0.041
27	0.017	0.017	0.020	0	0
29	0	0	0.038	0	0
30	0	0	0	0	0
31	0	0	0	0	0.052
32	0.015	0	0	0.090	0
33	0	0	0	0	0
39	0	0	0	0.003	0
40	0	0	0	0.003	0
43	0.052	0.047	0	0.006	0.049
44	0.006	0	0.023	0	0
Overall	0.013	0.010	0.012	0.009	0.015

Table D.3: Five-fold cross-validation soft error rates of five neutral zone classifiers for the fungi data set.

Probe ID	Method				
	LNZC	nLNZC	MNZC1	MNZC2	GNZC
1	0.035	0.038	0.038	0.038	0.041
2	0.020	0.026	0.026	0.026	0.029
3	0.186	0.180	0.180	0.145	0.195
4	0.041	0.047	0.047	0.047	0.052
6	0.078	0.061	0.099	0.061	0.061
8	0.032	0.032	0.032	0.032	0.029
10	0.041	0.041	0.041	0.041	0.052
13	0.105	0.105	0.084	0.081	0.145
15	0.012	0.015	0.218	0.198	0.015
17	0.032	0.032	0.032	0.032	0.023
19	0.044	0.044	0.044	0.044	0.078
21	0.020	0.023	0.032	0.032	0.026
23	0.035	0.038	0.038	0.038	0.044
24	0.084	0.061	0.081	0.061	0.084
25	0.044	0.058	0.078	0.293	0.055
26	0.105	0.105	0.116	0.189	0.105
27	0.061	0.058	0.061	0.052	0.058
29	0.052	0.052	0.084	0.052	0.052
30	0.023	0.026	0.026	0.026	0.026
31	0.049	0.055	0.035	0.038	0.090
32	0.067	0.052	0.102	0.174	0.052
33	0.017	0.017	0.017	0.017	0.032
39	0.017	0.017	0.020	0.020	0.029
40	0.023	0.026	0.029	0.029	0.026
43	0.119	0.113	0.099	0.102	0.116
44	0.070	0.064	0.084	0.064	0.064
Overall	0.054	0.053	0.067	0.074	0.061

Table D.4: Five-fold cross-validation total error rates of five neutral zone classifiers for the fungi data set.

Model Fitting of the Gaussian-mixture Neutral Zone Classifier

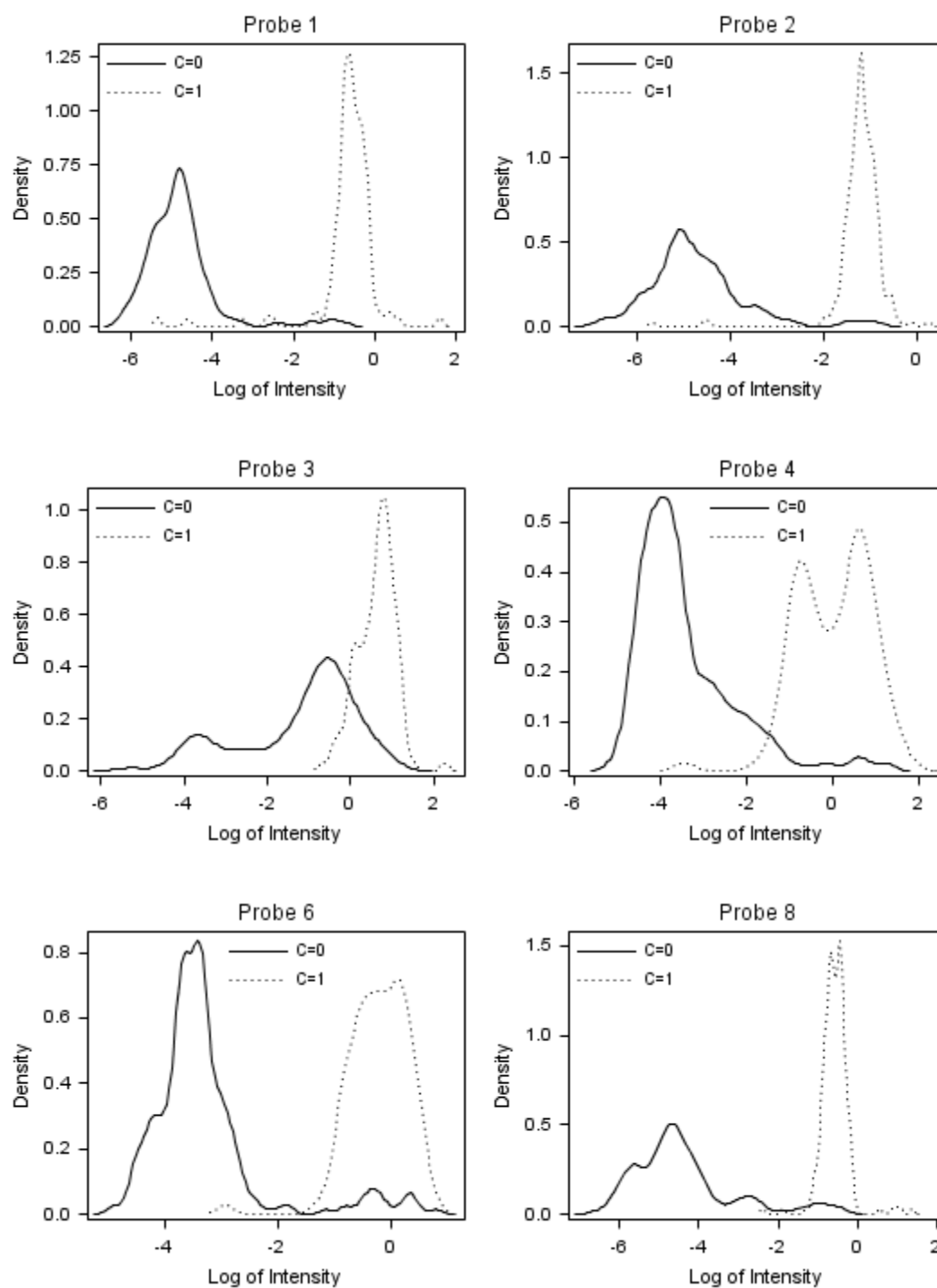
Probe ID	$\hat{\lambda}$	$\hat{q}_0$	$\hat{\mu}_{01}$	$\hat{\mu}_{02}$	$\hat{\sigma}_{01}$	$\hat{\sigma}_{02}$
1	-0.01	0.955	-5.085	-1.507	0.593	0.593
2	-0.19	0.968	-8.070	-1.339	2.049	0.326
3	-0.16	0.719	-0.529	-4.243	0.674	1.561
4	-0.08	0.361	-2.739	-4.732	1.515	0.596
6	0.00	0.923	-3.540	-0.185	0.515	0.515
8	-0.26	0.935	-9.355	-1.048	3.043	0.432
10	0.00	0.873	-4.994	-3.079	0.435	1.499
13	0.17	0.588	-2.655	-0.894	0.218	0.852
15	0.21	0.960	-2.896	-1.232	0.287	0.057
17	0.03	0.786	-4.076	-2.542	0.358	1.450
19	-0.27	0.057	-1.688	-7.950	1.539	1.539
21	-0.04	0.784	-4.173	-2.601	0.456	1.271
23	-0.11	0.951	-7.683	-3.067	0.846	0.846
24	0.00	0.660	-3.793	-1.129	0.550	0.838
25	-0.04	0.807	-6.774	-4.100	0.625	1.724
26	-0.13	0.903	-2.627	-6.039	1.523	0.323
27	0.08	0.641	-1.298	-3.459	0.478	0.478
29	-0.19	0.190	-4.984	-2.076	0.722	1.420
30	0.09	0.832	-4.175	-2.894	0.278	0.472
31	-0.27	0.306	-8.131	-5.994	1.086	1.086
32	-0.16	0.800	-2.628	-5.521	1.403	0.662
33	-0.46	0.521	-21.489	-15.273	8.507	1.831
39	-0.17	0.975	-8.747	-2.345	1.557	1.557
40	-0.23	0.047	-1.253	-10.997	0.201	3.117
43	0.20	0.656	-2.387	-0.922	0.296	1.061
44	0.26	0.925	-2.017	-0.100	0.379	0.379

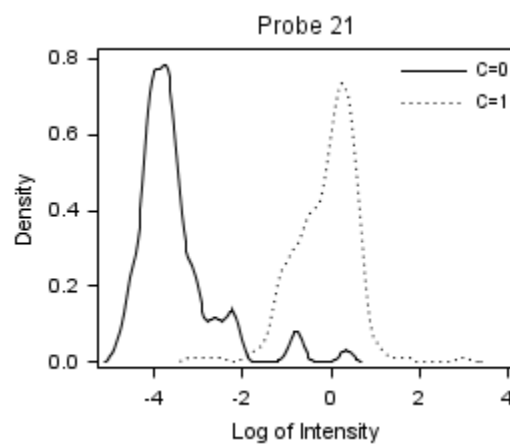
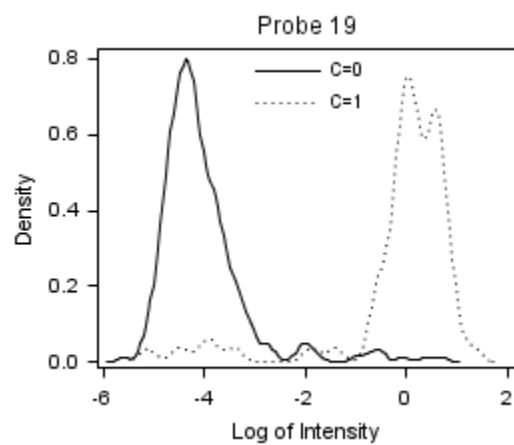
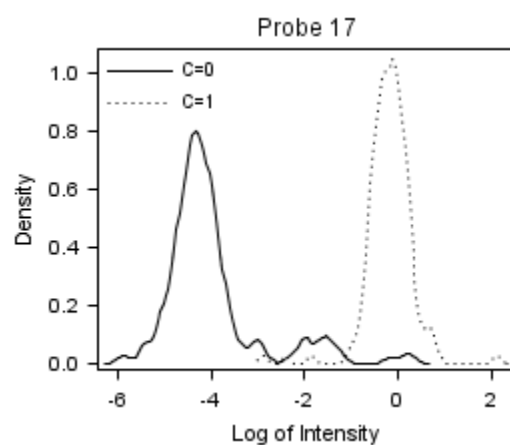
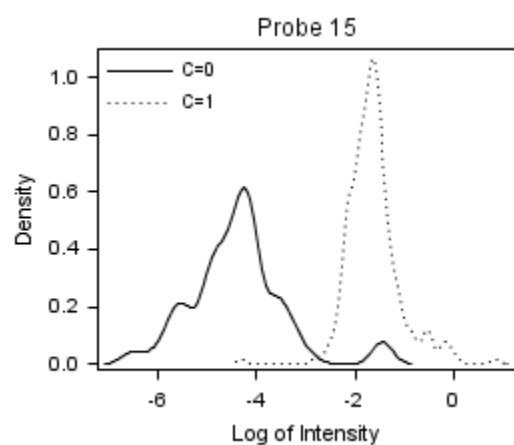
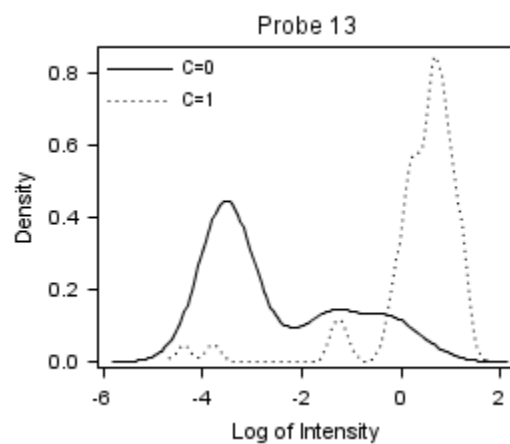
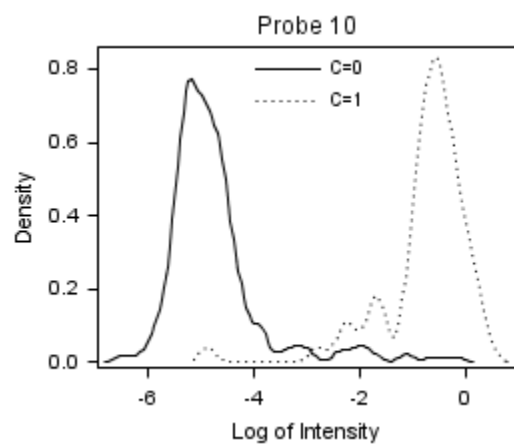
Table D.5: Distribution parameter estimates for non-binding class of fungi data

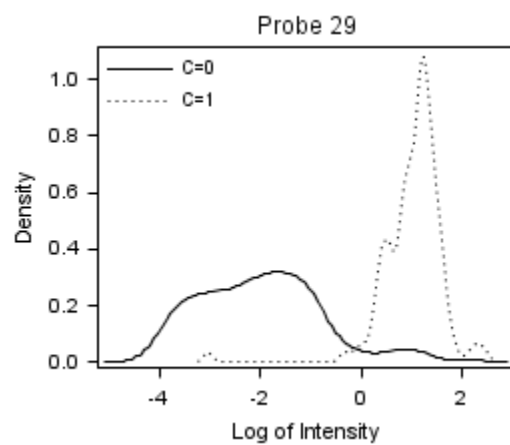
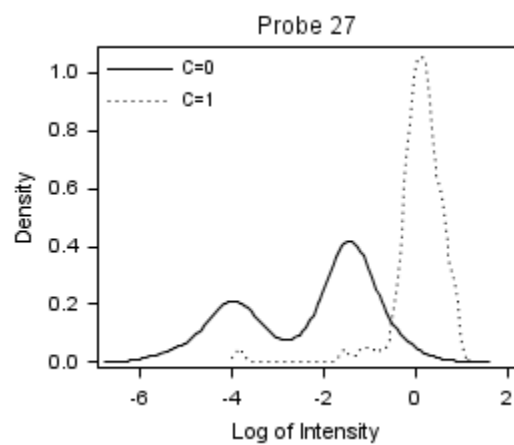
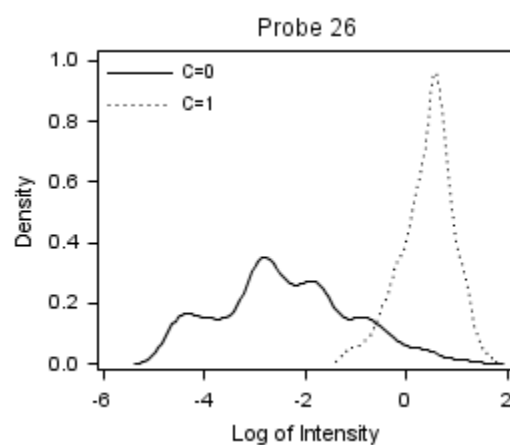
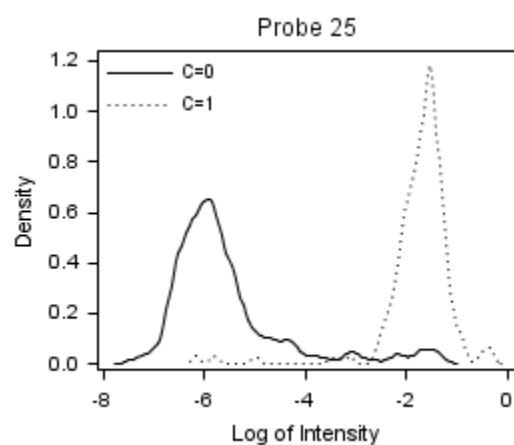
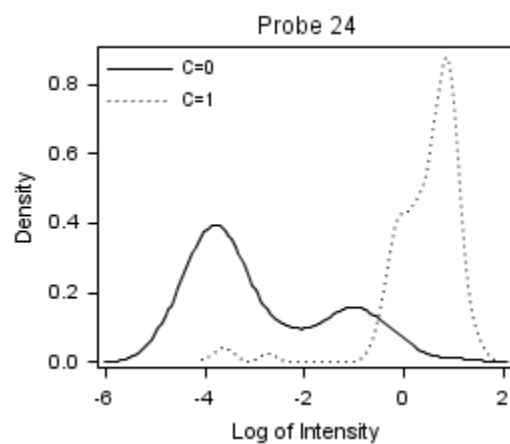
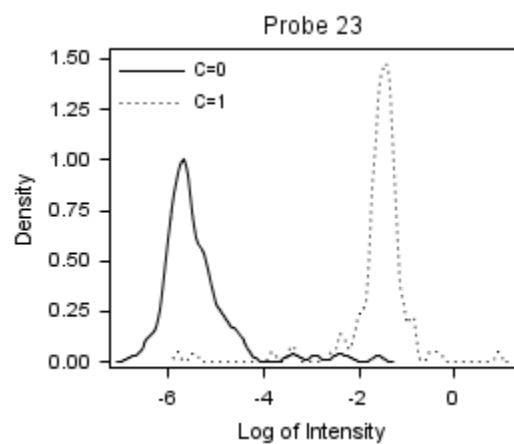
Probe ID	$\hat{\lambda}$	$\hat{q}_0$	$\hat{\mu}_{01}$	$\hat{\mu}_{02}$	$\hat{\sigma}_{01}$	$\hat{\sigma}_{02}$
1	-0.01	0.889	-0.554	-1.533	0.271	2.030
2	-0.19	0.987	-1.300	-8.176	0.365	1.426
3	-0.16	0.424	0.801	0.391	0.178	0.444
4	-0.08	0.295	0.700	-0.324	0.238	0.857
6	0.00	0.989	-0.196	-2.909	0.446	0.446
8	-0.26	0.937	-0.639	-0.522	0.263	1.308
10	0.00	0.770	-0.483	-1.729	0.363	1.115
13	0.17	0.894	0.651	-1.531	0.437	1.119
15	0.21	0.813	-1.455	-0.938	0.225	0.649
17	0.03	0.058	-0.555	-0.165	1.528	0.337
19	-0.27	0.893	0.186	-5.918	0.407	3.286
21	-0.04	0.388	0.281	-0.393	0.244	0.783
23	-0.11	0.856	-1.604	-2.848	0.283	2.268
24	0.00	0.968	0.513	-3.338	0.470	0.470
25	-0.04	0.936	-1.729	-3.377	0.383	2.309
26	-0.13	0.855	0.509	-0.496	0.365	0.365
27	0.08	0.942	0.168	-1.244	0.348	1.149
29	-0.19	0.991	0.948	-4.124	0.384	0.384
30	0.09	0.931	-1.618	-2.377	0.195	1.188
31	-0.27	0.826	-3.433	-3.808	0.407	1.305
32	-0.16	0.991	0.511	-5.855	0.431	0.431
33	-0.46	0.978	-1.406	-8.201	1.041	1.041
39	-0.17	0.950	-1.583	-2.987	0.389	2.486
40	-0.23	0.995	-1.043	-9.537	0.391	0.391
43	0.20	0.929	0.698	-1.597	0.418	1.009
44	0.26	0.985	-0.217	-1.599	0.312	0.312

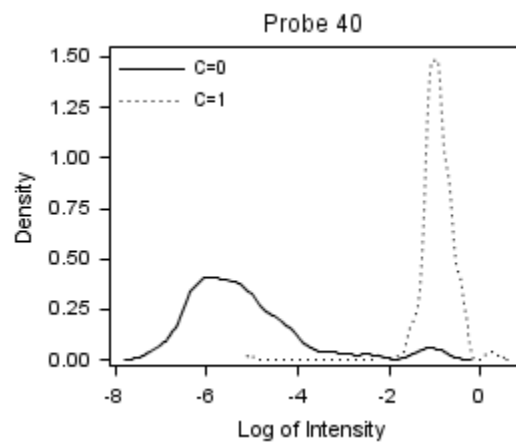
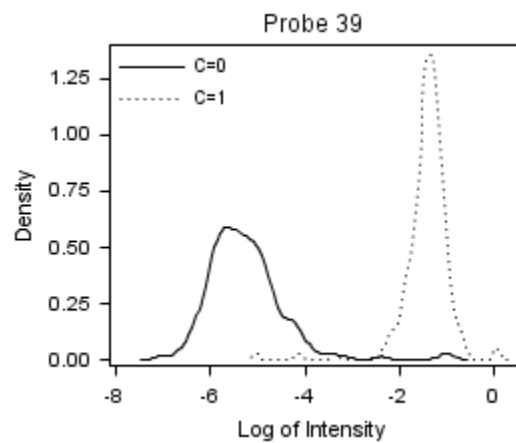
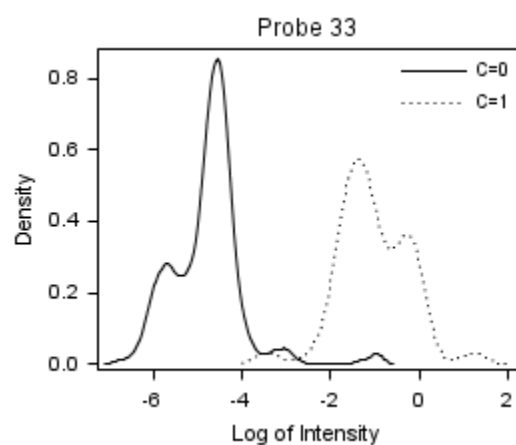
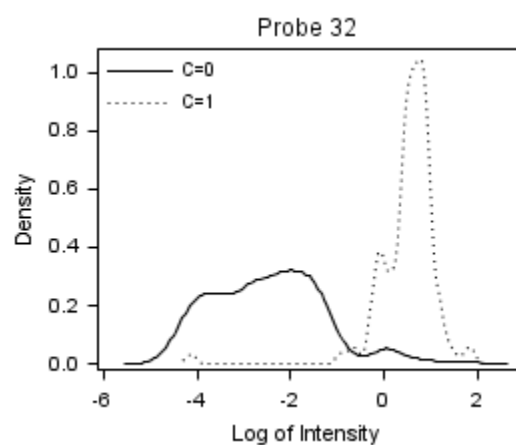
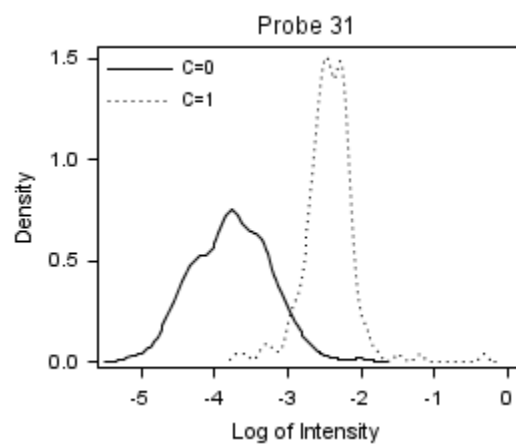
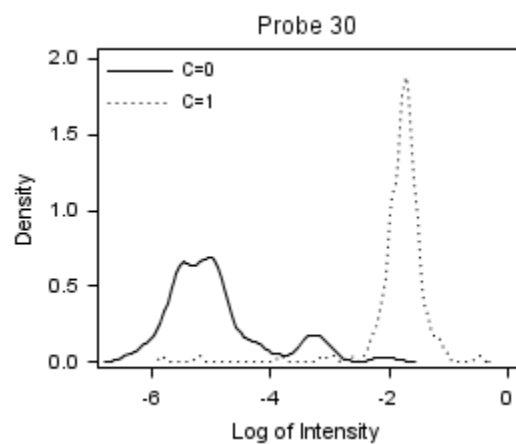
Table D.6: Distribution parameter estimates for binding class of fungi data

Distributions of Log of Intensity Measurements









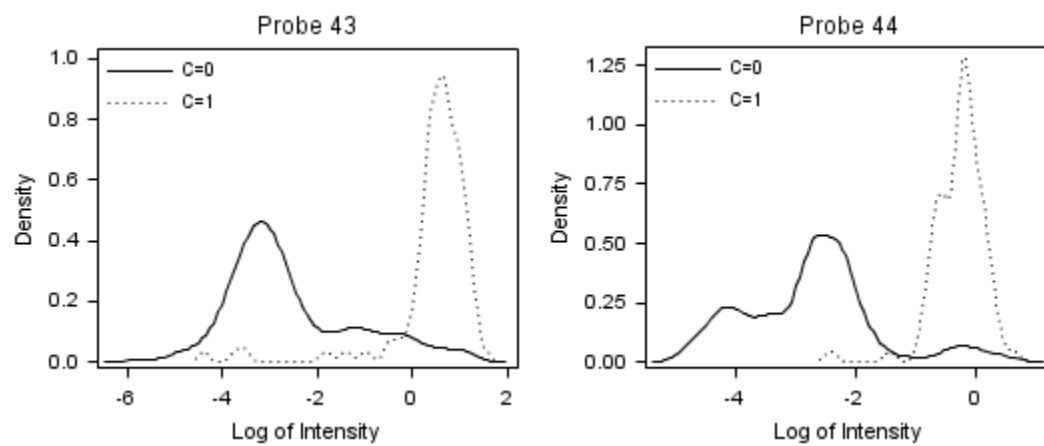


Figure D.1: Distributions of log(intensity) of fungi data for 26 probes