Stony Brook University

A Multi-class Classification using Ensembles of Multinomial Logistic Regression Models

A Dissertation Presented

by

Kyewon Lee

to

The Graduate School

in Partial Fulfillment of the

Requirements

for the Degree of

Doctor of Philosophy

in

Applied Mathematics and Statistics

(Statistics)

Stony Brook University

August 2010

Stony Brook University

The Graduate School

Kyewon Lee

We, the dissertation committee for the above candidate for the

Doctor of Philosophy degree, hereby recommend

acceptance of this dissertation.

Dr.Hongshik Ahn - Dissertation Advisor
Professor, Department of Applied Mathematics and Statistics

Dr.Nancy Mendell - Chairperson of Defense
Professor, Department of Applied Mathematics and Statistics

Dr.Haipeng Xing - Member
Assistant Professor, Department of Applied Mathematics and Statistics

Dr.Sangjin Hong - Outside Member
Associate Professor, Department of Electrical and Computer Engineering

This dissertation is accepted by the Graduate School

Lawrence Martin
Dean of the Graduate School

Abstract of the Dissertation

A Multi-class Classification using Ensembles of Multinomial Logistic Regression Models

by

Kyewon Lee

Doctor of Philosophy

in

Applied Mathematics and Statistics

(Statistics)

Stony Brook University

2010

This research proposes a method for multi-way classification problems using ensembles of multinomial logistic regression models. A multinomial logit model is used as a base classifier in ensembles from random partitions of predictors. The multinomial logit model can be applied to each mutually exclusive subset of the feature space without variable selection. By combining multiple models the proposed method can handle a huge database without a parametric constraint needed for analyzing high-dimensional data, and the random partition can improve the prediction accuracy by reducing the correlation among base classifiers. The proposed method is implemented using R and the performance including overall prediction accuracy, sensitivity, and specificity for each category is evaluated on real data sets and simulation data sets. To investigate the quality of prediction in terms of sensitivity and specificity, area under the

ROC curve (AUC) is also examined. Performance of the proposed model is compared to a single multinomial logit model and another ensemble method combining multinomial logit models using the algorithm of Random Forest. The proposed model shows a substantial improvement in overall prediction accuracy over a multinomial logit model.

Contents

List of Figures

List of Tables

Acknowledgements

I would like to give a huge thanks to my advisor, Dr.Hongshik Ahn, for his guidance on the dissertation and also for teaching me so much about statistics both inside and outside of the classroom. I am grateful for the opportunity he has given me to have the statistical experience. I have learned many valuable lessons from him.

I would like to give a big thanks to Dr.Nancy Mendell and Dr.Haipeng Xing for serving on my preliminary exam committee and dissertation committee. I also wish to thank Dr.Sangjin Hong for being my dissertation committee.

I would like to especially thank my family and friends for all their support.

Chapter 1

Introduction

Classification problem is omnipresent. When we are checking our emails everyday, among numerous emails we want to classify spam mails from other important messages. When we are ill, we want a doctor to diagnose our disease from our symptoms. These are all classification problems. In statistics, classification is a procedure in which individual items are placed into groups based on quantitative information on one or more characteristics inherent in the items and based on a training set of previously labeled items. By using the usual traits of spam mails, spam filter can classify spam and non spam emails. Spam filter is a sort of classifier. Based on their knowledge, doctors diagnose patient's illness as a specific disease from the information obtained by several medical examinations. The decision made by doctors is also a type of classification. There are numerous classification algorithms and some of well-known algorithms are briefly summarized below.

One of the simplest classification algorithms is the k-nearest neighbor algo-

rithm (k-NN). The k-NN algorithm uses known samples of size k to determine the class of a given instance. The given object is assigned to the class of the most relevant sample of size k. In order to choose relevant k samples, k-NN uses some metric. A typical metric is Euclidean distance in a multi-dimensional vector space. If $k = 1$, then we classify the given object as the class of the most similar neighbor. The value of k is found by performing cross-validation and to break ties it is best to use an odd value of k. In 1951, Evelyn Fix and J.L. Hodges came up with the idea of nearest neighbors[14].

Artificial neural networks (ANN) simulate the structure of biological neural networks[27]. An ANN consists of artificial neurons or nodes connected together by different weights, where the connections representing the synapses of a brain. If there is no hidden layer in the network, ANN reduces to a linear regression model. If there are one or more hidden layers in the network, then ANN is a non-linear generalization of the linear regression model. The idea of neural networks started in the 1940s, and in the 1950s Frank Rosenblatt implemented the first practical ANN which was a simple feed-forward model known as perceptron[26]. ANN can be applied in several areas such as classification of data (medical diagnosis), pattern recognition (identification of faces or object recognition), and sequence recognition (handwritten text recognition).

Linear classifiers separate objects by the value of a linear combination of their features. The feature of an object is represented by a vector. There is another vector to be trained with known observations. This is called weight vector. We

classify the object with the value of dot product and some threshold. There are several algorithms in this category such as Linear Discriminant Analysis (LDA)[13], Support Vector Machines (SVM)[8], and logistic regression[4].

LDA is an algorithm to generate the linear combination of features which best separates two or more categories of objects. If there are only two features, the separator between object groups will become a line. If there are three features, the separator will be a plane and the number of features is more than three, the separator become a hyperplane. LDA was originally developed by R.A. Fisher in 1936[13].

SVM is a method to find a separating hyperplane in data space, which maximizes the margin between the two separated data sets. If the data are nonseparable in the original feature space, they are transformed to a higher dimensional space, where the data become linearly separable. SVM was first introduced by Vladimir Vapnik in 1995[8].

Logistic regression is a model that fits the log odds of the response to a linear combinations of the explanatory variables. It is used mainly for binary responses, although there are extensions for multinomial responses as well. Regression coefficients are determined by maximizing the likelihood function. Usually the coefficients are estimated by numerical methods such as the Newton-Raphson algorithm. Logistic regression is known as a robust model for classification and the model is presented clearly and succinctly, but on the flip side, it might not be able to produce complex models, leading to underfitting. Logistic regression is widely used in areas such as medical and social sciences.

Ensemble methods combine multiple models to improve the performance of a model in classification. In our daily lives we use such an approach before making a decision. We want to obtain opinions of a few doctors before agreeing to a medical procedure. We want to read many reviews before purchasing an item. After combining the opinions of several experts, we make a final decision. By doing so, we can reduce the chance of unnecessary medical procedures or getting a poor product, and then achieve a better result. Ensemble methods yield better results if there is diversity among the members of the ensemble system[21]. As an extreme example, if we have the same result from all the classifiers in an ensemble, there would not be an improvement over a single classifier. If we have errors in different places from individual classifiers of an ensemble, then by combining the individual classifiers, the total error can be reduced. A usual way to get diversity is to use different training data sets obtained by resampling technique such as bagging or bootstrap.

Bagging, which stands for bootstrap aggregation, is one of the earliest ensemble based algorithms[5]. Different training data sets are randomly drawn with replacement from the entire training data set. Each training data set is used to train a different classifier of the same type. Individual classifiers are then combined by taking a simple majority vote of their decisions. For any given instance, the class chosen by a majority of classifiers is the decision of the ensemble. In bootstrap, the same size of sample is drawn with replacement

from the original sample. If the size of original data set is large enough, then a bootstrap sample is expected to have about 63.2% of distinct instances chosen from the original sample due to duplication. The remaining instances are called the out-of-bag sample, which is used as a test set.

Boosting is another commonly used ensemble algorithm, which was introduced by Robert E. Schapire in 1990[28]. Boosting combines multiple weak classifiers to generate a single strong one[19, 29]. A weak classifier is slightly correlated with the true classification, while a strong classifier is arbitrarily well-correlated with the true classification. Boosting also uses resampling data sets to generate classifiers and then they are combined by majority voting. However, unlike bagging, boosting strategically creates resampling data sets to obtain the most informative training data. Each iteration of boosting creates three weak classifiers. The first classifier is trained on a random subset of the available training data. The second classifier is trained on a training data only half of which is correctly classified by the first classifier, and the other half is misclassified. The third classifier is trained with instances on which the first classifier and the second classifier disagree. These three classifiers are combined by majority voting. There are many boosting algorithms according to their training schemes such as AdaBoost, PBoost, TotalBoost, BrownBoost, MadaBoost, LogitBoost and so on. Among these, AdaBoost is a widely used algorithm. AdaBoost was formulated by Yoav Freund and Robert Schapire[15]. In each iteration of AdaBoost, a classifier is trained by giving more weight to

the case which made a wrong prediction on the previous iteration. The final decision is made by a weighted majority voting among all the classifiers.

Random Forest (RF) uses the result by combining multiple decision trees using the bagging algorithm[6]. Decision tree is a structure to classify an object into classes. From the root of a tree, the given object follows the relevant branches and arrives at a leaf. Branches are features and leaves are classes. If the number of cases in the original data set is N, a bootstrap sample of size N is generated as a training set to grow each tree. If there are M input variables, a number m which should be much less than M is specified, and m is held constant for the forest thereafter. At each node of a tree, m variables are randomly selected out of the M variables and the best split on these values is used to split the node. All trees are grown to their largest extent possible without pruning. Each tree gives a classification for a new object from an input vector, and we say the tree votes for that class. The forest chooses the class having the most votes over all the trees in the forest. Each tree is constructed using a different training set obtained from the original data set. When a training set for a tree is selected with replacement from the original data set, about one-third of the cases are left and not used in the construction of the tree as explained earlier. This out-of-bag data can be used to get the estimates of the classification error or variable importance. Hence in RF, there is no need for cross-validation or a separate test set. RF is known as efficient and applicable to large data sets like microarray data. Leo Breiman and Adele

Cutler developed the RF method in 2001[6].

CERP (Classification by Ensembles from Random Partitions)[3] is another tree-based classification method by ensembles of base classifiers. One of the most important characteristics of CERP is random partitioning of the feature space. For each ensemble, the feature space is randomly partitioned with roughly equal size, and a base classifier is constructed for each subspace. Through combining these multiple base classifiers, CERP is able to improve the prediction accuracy compared to a single classifier[6, 30]. All the base classifiers of an ensemble have the same probability of classification error because they are constructed on each of the randomly partitioned subspaces with nearly equal size. This property can enhance the prediction accuracy in an ensemble[3]. Since different combinations of predictors in a different ensemble can give more information, several ensembles are generated to achieve further improvement. CERP shows its usefulness clearly when it is applied to a high-dimensional data set. By partitioning the huge feature space into small spaces, the data set becomes easy to handle, and allows a variety of models.

LORENS (Logistic Regression Ensembles)[23] uses the logistic regression model as base classifier instead of tree in the CERP algorithm to classify binary responses. Although logistic regression is known to be a robust classification method for binary responses, it requires more observations than predictors. Thus, in order to apply logistic regression to a high-dimensional feature space, variable selection is unavoidable. However, in LORENS, each base classifier is

constructed from a different set of predictors determined by a random partition of the entire set of predictors, so that there are always more observations than predictors. Hence the logistic regression model can be used without variable selection.

Although LORENS is a useful classification method for high-dimensional data, it is designed for binary responses. Multiclass problems are common, thus it is necessary to develop a method comparable to LORENS for a multi-way classification. LORENS is expanded to multiclass problems (mLORENS) and the performance of the new method is evaluated in this study. The multinomial logistic regression (MLR) model can be easily implemented and it is less computer intensive than the tree-based CERP. It was shown that the prediction accuracy of LORENS is as good as that of RF or SVM using real data sets and a simulation study[23].

In this study, improvements of LORENS over the logistic regression model and mLORENS over the multinomial logistic regression model are investigated. To show the improvement of LORENS over the logistic regression model, data on detection of allelic expression of imprinted genes are used. To show the improvement of mLORENS over the multinomial logistic regression model, two real data sets as well as simulated data are used.

Besides the above comparison, mLORENS is compared to Random Multi-Nomial Logit (RMNL)[24] which is based on bagging in this research. RMNL builds an ensemble of multinomial logits instead of tress in the frame of RF

using bootstrap samples. The program for RMNL model was implemented in this study using R and the performance was compared to those of mLORENS and MLR using real data and simulated data. mLORENS showed better performance than RMNL in simulated data, but for real data, the two methods showed similar performance.

Chapter 2

Methods

2.1 LORENS (Logistic Regression Ensembles)

Based on the CERP algorithm, Lim et al.[23] developed LORENS by using logistic regression models as base classifiers. To minimize the correlation among classifiers in the ensemble, the feature space is randomly partitioned into K subspaces with roughly equal sizes. Since the subspaces are randomly chosen from the same distribution, we assume that there is no bias in selection of predictors in each subspace. In each of these subspaces, a full logistic regression model is fit without a variable selection when the number of subspaces is big enough. LORENS combines the results of these multiple logistic regression models by taking the average of the predicted probabilities within an ensemble. The predicted probabilities from all the base classifiers (logistic regression models) in an ensemble are averaged and the sample is classified as either 0 or 1 using a decision threshold on this average. Through finding the optimal

thresholds from cross validation, the balance of sensitivity and specificity on unbalanced data sets could be significantly improved compared to other classification methods without sacrificing the overall accuracy [23].

2.2 mLORENS for Multinomial Logistic Regression Model

2.2.1 Multinomial Logistic Regression Model

Suppose Y is a categorical response variable with J categories. Let $\{\pi_1, \ldots, \pi_J\}$ be the response probabilities satisfying $\sum_j \pi_j = 1$. When one takes n independent observations based on these probabilities, the probability distribution for the number of outcomes that occur as each of the J types is multinomial[2].

If a category is fixed as a baseline category, we have $J-1$ log odds paired with the baseline category. When the last category (J) is the baseline, the baseline-category logits are

$$\log\left(\frac{\pi_j}{\pi_J}\right), \qquad j = 1, \ldots, J-1.$$

The logit model using baseline-category logits with predictor x has the form

$$\log\left(\frac{\pi_j}{\pi_J}\right) = \alpha_j + \beta_j \mathbf{x}, \qquad j = 1, \ldots, J-1. \tag{2.1}$$

The model consists of $J-1$ logit equations, with separate parameters. By fitting these $J-1$ logit equations simultaneously, estimates of the model

parameters can be obtained, and the same parameter estimates occur for a pair of categories regardless of the baseline category[1]. The estimates of the response probabilities can be expressed as

$$\pi_j = \frac{\exp(\alpha_j + \beta_j x)}{\sum_h \exp(\alpha_h + \beta_h x)}, \qquad j = 1, \ldots, J-1. \tag{2.2}$$

Although MLR is a robust classification method for a multi-way classification, it is not suitable for high-dimensional data, and variable selection is inevitable. Current software packages do not seem to have variety of variable selection algorithms for MLR. Selecting an optimal set of variables can be computer intensive, and there is no guarantee that the best set can be chosen. MLR can be applied to CERP algorithm without variable selection. We develop a new method which possesses the nice properties of MLR and simultaneously inherits all the advantages of CERP handling high-dimensional data sets.

2.2.2 mLORENS

MLR is used as a base classifier of CERP to develop a classification method for multiclass problems. The procedure of mLORENS is described as follows.

Suppose Θ is the feature space of independent variables. This space is randomly divided into mutually exclusive K subspaces $(\theta_1, \theta_2, \ldots, \theta_K)$ with roughly equal size. The number of independent variables in a subspace should be small enough to fit a multinomial logit model. The partition size is determined and this constraint is satisfied. For each subspace, a multinomial logit model is fitted. That is, $J-1$ logit equations (2.1) with $\sum_j \pi_j = 1$ are simultaneously

fitted.

Using different sets of explanatory variables, base classifiers are generated respectively for each of the subspaces discussed above, and then we have a set of base classifiers, say, $\{h_1, \ldots, h_K\}$ corresponding to the K subspaces. The classification error is most reduced in an ensemble whose members make individual errors in a less correlated manner. Due to the randomness of the partitioning, we expect that the correlation among the base classifiers is small[22, 18]. Hence we expect improvement of the prediction accuracy in the whole ensemble[6]. In other words, the base classifiers are anticipated to have similar prediction errors, and by combining these weak classifiers we can have a better classifier using all explanatory variables.

To achieve a further improvement, several ensembles are constructed in the same manner. Different combinations of predictors are generated, and each ensemble consists of different base classifiers fitted using those different sets of predictors. In this study, eleven ensembles are used and hence we have eleven sets of base classifiers $\{h_{i1}, \ldots, h_{iK}\}$, where $i = 1, \ldots, 11$.

The predicted values are determined by the method of averaging. In the base classifier in each subspace of every ensemble, the estimate of predicted probability for each category is calculated as (2.2). The feature space is partitioned and the predictors in a subspace of the feature space are used to construct a base classifier. Hence, the same predictors which are used for fitting a base classifier are used to get the predicted values of an instnace for a subspace

of features. By averaging all predicted probabilities from every subspace in an ensemble (the sum of estimated probabilities divided by the number of partitions), we obtain the predicted probability for each ensemble, and again all ensemble probabilities are averaged (the sum of ensemble probabilities divided by the number of ensembles) to obtain the overall predicted probabilities for the whole model.

In a binary classification the majority voting method can be used with threshold to get the predicted values within or between ensembles. For a multicategory classification, application of threshold is not straightforward. Fortunately, averaging method works well for the multicategory cases in this study. It turned out that the averaging worked slightly better for the binary classification[23]. Through averaging we predict probability π_j for each category $j = 1, \ldots, J$, with $\sum_j \pi_j = 1$ and the category with the maximum predicted probability is chosen as the predicted class.

The number of subspaces in a partition is searched through a nested cross validation. Suppose the sample size of a training set is n. There are several candidates for the partition size such as $n/2, n/3, \ldots, n/10$ and $n/12$. In a learning set a 3-fold cross validation is performed. In each nested learning set of 3-fold cross validation, all candidates of the partition size are applied one by one. That is, in each learning set, a mLORENS model is built for each partition size. The prediction accuracies are evaluated and the partition sizes yielding the highest accuracy at the corresponding test set is chosen. If n/i is

chosen for some integer i, $i = 2, \ldots, 10$ or 12, then the next step is to find a number between $n/(i-1)$ and n/i, and between n/i and $n/(i+1)$ based on prediction accuracy through an adaptive dual binary search method[3]. After this step, two candidates are left, and the one with higher accuracy is chosen as the final partition size.

LORENS is less computer intensive than the tree-based CERP model. It is easy to implement and performs well[23]. The program for base classifiers in mLORENS was implemented in R using *multinom* function in *nnet* package.

2.3 Alternative Approaches to Multinomial Logistic Regression Model

A couple of alternative approaches can be tried instead of MLR. Multiple logistic regression analysis (one for each pair of outcomes) can be considered as an alternative. One problem of this approach is that each analysis is run on a different sample. The other problem is that without constraining the logistic models, we can end up with the probability of choosing all possible outcome categories greater than 1.

Another approach is nested binary models (NBM)[9]. The categories can be collapsed to two and then a logistic regression can be applied. For prediction, the class with the highest estimated response is chosen. This alternative model was compared to MLR in this research. In simulation experiment 1, NBM was tried as a base classifier in mLORENS. Along with the variable selection,

a single NBM was also tried, and the results were compared to MLR and mLORENS with both of NBM and MLR. Even though it is expected to have some information loss in this alternative case, the results of NBM were not significantly worse on these simulation data.

2.4 RMNL (Random MultiNomial Logit)

Prinzie et al.[24] proposed Random MultiNomial Logit model which fits multinomial logit models in different bootstrap samples. They borrowed the structure of RF, and used the idea of bagging. B bootstrap samples are drawn with replacement from N data instances, and in each bootstrap sample a MLR is fit with randomly selected m features from total of M features, so that we have B MLR models. To combine the results of MLR models, two method were used. One is Majority Voting (MV) which chooses the class with the greatest vote among B prediction results. The other method is adjusted Majority Voting (aMV) which is averaging the predicted probabilities from all B MLRs for each class, and the class with the highest predicted probability is selected as the predicted class.

In this study pre-fixed $B = 100$ was used. In RF, the number of variables randomly selected at each node of a tree is usually square root of the total number of variables. In this research, the number of selected variables m with the highest prediction accuracy was chosen in learning phase among several possible given candidates. The out-of-bag data were used to test the candidates. For each instance in out-of-bag data of each bootstrap sample, the predicted

class (MV) or the prediction probabilities for every class (aMV) are obtained for every candidate of m. The results from the B models are merged according to the method of combining (MV or aMV) for each candidate of m. The overall accuracy is calculated for every m, and the one with the highest accuracy is chosen. The models with the final number of selected variables are used in testing phase. RF generates diversity by randomly selecting variables in each node of a base tree. Thus various variables are involved in one tree model. However, in RMNL, one base classifier is built by only one random selection of variables.

2.5 Evaluation

In this research, efficiency of the classification methods was evaluated in terms of ACC (overall accuracy), SENS (sensitivity), SPEC (specificity), and AUC (areas under the receiver operating characteristic curves).

Overall accuracy was calculated by the total number of correct predictions divided by total number of predictions. Sensitivity is proportion of true positive (identified correctly as positive) among actual positives, and specificity is proportion of true negative (identified correctly as negative) among actual negatives. Sensitivity and specificity were calculated respectively for each category as follows. If the number of classes is three, for example, Table 2.1 shows the predicted classification and true classification. The sensitivity is calculated by $a/(a+b+c)$ for class 1, $e/(d+e+f)$ for class 2, and $i/(g+h+i)$

for class 3. The specificity is $(e+f+h+i)/(d+e+f+g+h+i)$ for class 1, $(a+c+g+i)/(a+b+c+g+h+i)$ for class 2, and $(a+b+d+e)/(a+b+c+d+e+f)$ for class 3.

Table 2.1: True class and predicted class

		True class		
		1	2	3
Predicted class	1	a	d	g
	2	b	e	h
	3	c	f	i

AUC was used to assess the quality of the methods regarding sensitivity and specificity. There is a trade-off between sensitivity and specificity, and the receiver operating characteristic (ROC) curve can represent the trade-off graphically. For a binary classification, AUC can be obtained from the Mann-Whitney statistic since this statistic is equivalent to the value of AUC[10]. For multiclass problems, volume should be considered instead of area. However, in this study, mean AUC was used as an estimate of the extension of AUC. For each category, AUC was obtained by grouping the data into the given category and the rest using the Mann-Whitney statistic. The mean AUC was obtained by averaging these AUC's[17]. This estimate was shown as one of the best estimates when the classifiers accompany each prediction with the estimated probabilities of each class[12].

The significance of the difference in each evaluation category (ACC, SENS, SPEC) was determined by McNemar's test. A two-way contingency table was built using actual and predicted classes of each subject for the classification methods to be compared. McNemar's test was performed with this contingency

table. For overall accuracy, the contingency table was obtained by checking if the predicted class is the same with the actual class for each subject. Accuracy for each class was also compared by collapsing the data into two classes (one and the rest). For sensitivity, a two-way contingency table was built using the subjects from the actual class. For specificity, the table was constructed from the subjects which are not from the actual class. If more than two classification methods are compared on a set of data, then the Bonferroni correction of significance level can be applied. If n methods need to be tested, $n-1$ pair-wise tests are required (the number of models required to test is different depending on the data), and thus significance level is changed from α to $\alpha/(n-1)$.

2.6 Variable Selection

Variable selection is required for logistic regression or MLR models for high-dimensional data. BW ratio was selected to use for the variable selection in this research. BW ratio can be obtained by computing the between-group sum of squares (BSS) and dividing it by the within-group sum of squares (WSS). For each variable, this ratio is calculated, and the variables with high BW ratios are selected. For a particular variable j, BW ratio is defined as

$$BWratio\,(j) = \frac{BSS\,(j)}{WSS\,(j)} = \frac{\sum_i \sum_k I\,(y_i = k)\,(\overline{x}_{kj} - \overline{x}_{.j})^2}{\sum_i \sum_k I\,(y_i = k)\,(\overline{x}_{ij} - \overline{x}_{kj})^2} \tag{2.3}$$

$$\text{where} \qquad \overline{x}_{\cdot j} = \frac{\sum_i x_{ij}}{N}, \qquad \overline{x}_{kj} = \frac{\sum_{I(y_i=k)} x_{ij}}{N_k} \qquad .$$

BW ratio was shown as a reliable method for variable selection for high-dimensional data[11].

Chapter 3

Applications to Real Data

3.1 Gene Imprinting Data

The improvement of LORENS over a single logistic regression model was investigated using Gene Imprinting Data[16]. Genomic imprinting is defined as gene expression dependent on the parent of origin, and it gives rise to numerous human diseases[31]. Greally[16] described the first characteristic sequence parameter that discriminates imprinted regions - a paucity of short interspersed transposable elements (SINEs). This finding has subsequently been confirmed by other groups. The genomic data collected to study imprinted genes were from the UCSC Genome Browser (http://genome.ucsc.edu/). Annotation data were downloaded for the human genome (hg 16, July 2003 freeze). The sequence features of interest were repetitive elements (chrN-rmsk files), CpG islands (cpgIsland file), transcription start sites of other genes and the exon count of each gene (refGene file). Each feature was examined for varying window sizes

around the transcription start and end sites.

These data have information of 131 samples with 1248 predictors to classify into imprinted or not imprinted genes. There are 43 imprinted genes and 88 non-imprinted genes. Twenty repetitions of 10-fold cross validation (CV) were conducted for LORENS and a single logistic regression model. The performance of the two methods were evaluated by accuracy, sensitivity, specificity, PPV (positive predictive value: rate of true positives among positive predictions) and NPV (negative predictive value: rate of true negatives among negative predictions), and AUC. The partition size in LORENS was determined in the learning phase according to accuracy. A partition size with the highest accuracy was selected among several trials using an adaptive dual bisection method[23]. For LORENS an optimal threshold with highest accuracy was searched in the training phase using nested cross validation[23]. A single logistic regression model does not search an optimal threshold. Thus, for a fair comparison, two fixed decision thresholds 0.5 and 0.33 (43/131, the proportion of imprinted genes) were tried for both methods. To run a single logistic regression model, variable selection is required. Among 1248 predictors 10, 30, and 50 predictors were selected in the training phase considering the sample size of 131. Variable selection was done by BW ratio.

Table 3.1 shows the results for this example, and Table 3.2 represents the results of McNemar's test to see the significance of difference between two methods in accuracy, sensitivity and specificity. LORENS showed better

performance than a single logistic model for both the fixed thresholds in accuracy. Among 3 trials with different number of selected variables for logistic regression model, the one with 30 variables showed the highest accuracy in both fixed thresholds. In LORENS the partition size was 68 (sd 32) for threshold of 0.5 and 69 (sd 31) for threshold of 0.33. So, the number of variables in each partition becomes approximately 18. When the logit models with 30 variables were compared to LORENS, the p-values showed that, in sensitivity, there was no significant difference between two models when the threshold was 0.5, and for the threshold of 0.33, specificities were not significantly different. However, when threshold search was done, LORENS showed better performance than a logistic model in all the measures. The p-values regarding these comparisons in accuracy, sensitivity, and specificity were all less than 0.0001. The accuracy of LORENS improved to 85% from 82% or 80% when threshold search was performed. In AUC, LORENS showed better performance than a single logistic model for both fixed thresholds. The AUC of LORENS was 0.74 for threshold of 0.5 and 0.80 for threshold of 0.33, and it improved to 0.84 when threshold search procedure was used. Figure 3.1 depicts the accuracies with 1-standard deviation bars for LORENS with searched and fixed thresholds and logit models with 30 selected variables, and Figure 3.2 depicts the AUCs for the models.

When the threshold was changed to 0.33 from 0.5, the balance of sensitivity and specificity improved without sacrificing the overall accuracy, while the positive and negative predictive values became less balanced. When the threshold search was conducted for LORENS, the balance of the positive and

negative predictive values improved from both of the fixed threshold cases, and the sensitivity and specificity showed better balance than when threshold of 0.5 was used. These results show that the threshold search is successful.

Table 3.1: Gene Imprinting Data: Performances (SD in parentheses) of LORENS and logistic regression models. Twenty repetitions of 10-fold CV were used for each method.

Threshold	Model	♯var.[a]	♯part.[b]	ACC	AUC	SENS	SPEC	PPV	NPV
.38(.02)[c]	LORENS	all	71.2	.85	.83	.77	.91	.81	.89
			(33.4)	(.03)	(.03)	(.04)	(.02)	(.03)	(.02)
.5[d]	LORENS	all	67.9	.82	.74	.49	.99	.96	.80
			(31.5)	(.02)	(.03)	(.06)	(.02)	(.05)	(.02)
	Logistic	10		.69	.59	.31	.87	.55	.72
				(.03)	(.03)	(.05)	(.04)	(.09)	(.02)
		30		.71	.66	.53	.80	.56	.78
				(.04)	(.05)	(.08)	(.04)	(.07)	(.03)
		50		.68	.64	.52	.75	.51	.76
				(.04)	(.04)	(.06)	(.05)	(.05)	(.03)
.33[d]	LORENS	all	68.5	.80	.80	.82	.79	.65	.90
			(30.9)	(.02)	(.02)	(.02)	(.04)	(.04)	(.01)
	Logistic	10		.66	.64	.60	.68	.48	.78
				(.04)	(.04)	(.07)	(.06)	(.04)	(.03)
		30		.69	.66	.58	.75	.53	.78
				(.05)	(.06)	(.09)	(.05)	(.07)	(.04)
		50		.68	.65	.56	.74	.51	.77
				(.04)	(.05)	(.09)	(.03)	(.06)	(.04)

[a] number of selected variables chosen in the training phase

[b] average number of mutually exclusive subsets of predictors in a partition, chosen in the trainiing phase

[c] threshold searched in the training phase

[d] fixed threshold

Table 3.2: McNemar's Test Results of Gene Imprinting Data

Models		p-value
LORENS[a]: Logistic w/0.5th. 30var.[b]	ACC:	<0.0001
	SENS:	<0.0001
	SPEC:	<0.0001
LORENS : Logistic w/0.33th. 30var.[c]	ACC:	<0.0001
	SENS:	<0.0001
	SPEC:	<0.0001
LORENS w/0.5th.[d]: Logistic w/0.5th. 30var.	ACC:	<0.0001
	SENS:	0.1028
	SPEC:	<0.0001
LORENS w/0.33th.[e]: Logistic w/0.33th. 30var.	ACC:	<0.0001
	SENS:	<0.0001
	SPEC:	0.2563

[a] LORENS with searched threshold
[b] Logistic regression with a fixed threshold of 0.5 and 30 variables
[c] Logistic regression with a fixed threshold of 0.33 and 30 variables
[d] LORENS with a fixed threshold of 0.5
[e] LORENS with a fixed threshold of 0.33

Figure 3.1: Accuracies for Gene Imprinting Data

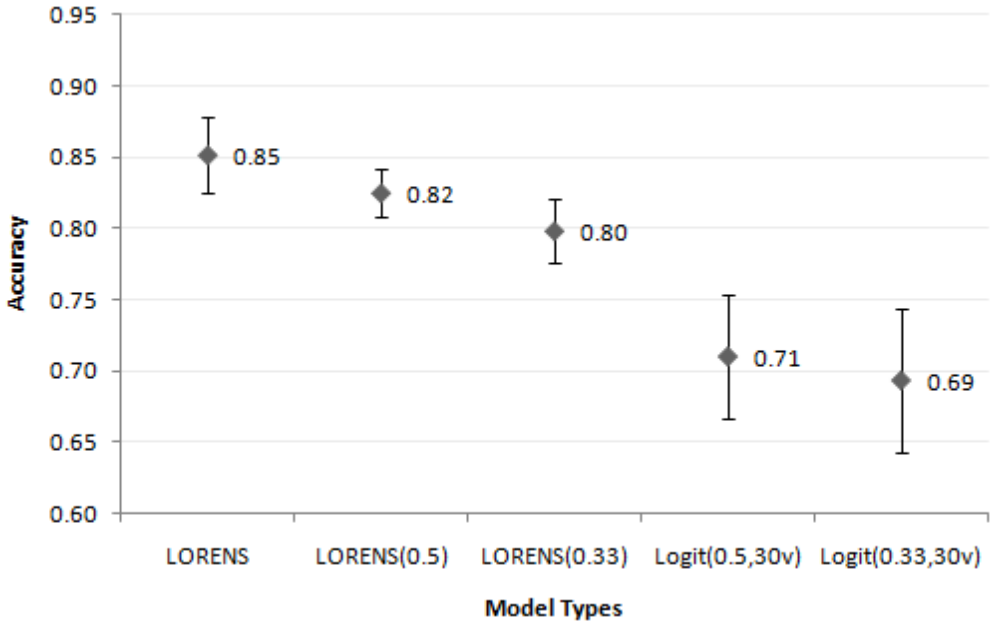
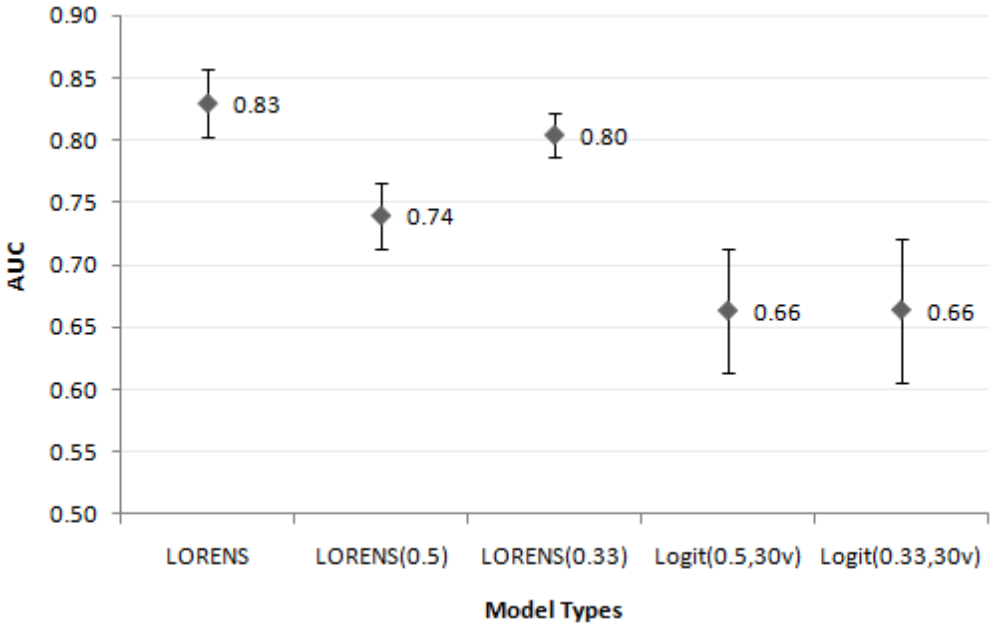


Figure 3.2: AUCs for Gene Imprinting Data



3.2 Gastrointestinal Bleeding Data

The first real data set used for comparison of the multi-way classification methods is data on Acute gastrointestinal bleeding (GIB). GIB is an increasing health care problem due to rising NSAID (non-steroidal anti-inflammatory drugs) uses in an aging population[25]. In the emergency room (ER), the ER physician can misdiagnose a GIB patient at least 50% of the time [20]. While it is best for a gastroenterologist to diagnose GIB patients, it is not feasible due to time and constraints. Classification models can be used to assist the ER physician to diagnose GIB patients efficiently and effectively, providing scarce health care resources to those who need it the most. Eight different classification models on a 121 patient GIB database were evaluated [7]. Using clinical and laboratory information available within a few hours of patient presentation, the models can be used to predict the source of bleeding, need for intervention, and disposition in patients with acute upper, mid, and lower GIB.

To reduce the mortality from acute GIB by an appropriate treatment, it is important to find the source of bleeding in its early stage. Hence it is required to predict the source of the bleed within a few hours of patient's presentation from the patient clinical data. The bleeding source is classified into three locations: Upper, Mid, and Lower intestine. The definitive source of bleeding was the irrefutable identification of a bleeding source at upper endoscopy, colonoscopy, small bowel enteroscopy, or capsule endoscopy. Twenty variables used to predict

the source of bleeding include prior history of GIB, hematochezia, hematemesis, melena, syncope/presyncope, risk for stress ulceration, cirrhosis, ASA/NSAID use, systolic and diastolic blood pressures, heart rate, orthostasis, NG (Naso Gastric) lavage, rectal exam, platelet count, creatinine, BUN (Blood Urea Nitrogen), and INR (International Normalized Ratio). The records of 121 GIB patients were used in this study. Among 121 subjects, 81 fall in upper, 29 in lower, and the remaining 11 fall in mid.

For mLORENS, fixed partition size of 2 or 3 was used without conducting the procedure of partition size searching since there are only 20 variables. About 10 or 7 variables were included in each subspace. For RMNL, 100 bootstrap samples were drawn, and in each sample MLR was fit. In RF, random variable selection is conducted at each node, but unlike RF, in RMNL variable selection should be done before fitting multinomial logit. Hence, the number of variables with the highest accuracy among pre-assigned numbers was chosen in learning phase. In this example, the number of variables was searched between 5 and 20. For each candidate number, average accuracy was calculated from the 100 MLR fits, and the number with the highest accuracy was chosen and used for the test sets. The average selected number of variables for RMNL was about 13 (sd 4). For the comparison of the three models, 20 repetitions of 10-fold CV were conducted for each model.

The performance of the methods for this example is provided in Table 3.3. In mLORENS, the results for partition sizes 2 and 3 were almost the same, but

the mean AUC of partition size 2 was little higher than that of partition size 3. In RMNL, there was no significant difference between two combining methods: MV and aMV in all measurements. Table 3.4 shows the results of McNemar's test comparing the three methods: mLORENS with partition size 2, MLR, and RMNL with aMV. In overall accuracy, mLORENS performed better than a single MLR. The accuracy of mLORENS was 93% and the accuracy of a MLR was 89%. The difference between the accuracies of mLORENS and MLR is significant yielding the p-value less than 0.0001. In sensitivity, mLORENS showed higher performance than a single MLR for the two large classes (Upper and Lower). In the smallest class (Mid), MLR showed higher sensitivity than mLORENS, but the difference was not significant since the p-value of the test was 0.0576. In specificity, mLORENS was better in Lower and Mid classes, but in the largest class (Upper), the specificity of MLR was significantly better than that of mLORENS. RMNL also performed better than MLR in accuracy. RMNL showed higher sensitivity in two large classes (Upper and Lower), and MLR showed better sensitivity in the smallest class (Mid). In the smallest class, both of the two ensemble methods using MLR models did not show higher sensitivity than a single MLR. This finding implies that the ensemble methods might have a problem of imbalance. When the accuracies of mLORENS and RMNL were compared, the p-value was 0.01495. In terms of AUC, all three methods showed high performance. In mLORENS, AUC for partition size of 2 was higher than that of partition size 3. The balance of sensitivity and specificity was also better in the case of partition size of 2 specifically in the

class of Mid. Figure 3.3 (accuracies) and Figure 3.4 (AUCs) are provided to help comparison among the models.

Table 3.3: GIB Data: Performance (SD in parentheses) of mLORENS , MLR and RMNL. Twenty repetitions of 10-fold CV were used for each method.

Model	♯Var.	♯Part.[a]	ACC	AUC [b]		Upper	Lower	Mid
mLORENS	all	2	.93	.90	SENS:	.98 (.01)	.89 (.03)	.68 (.09)
		(fixed)	(.01)	(.02)	SPEC:	.93 (.02)	.97 (.01)	.98 (.01)
					AUC:	.96 (.01)	.93 (.02)	.83 (.05)
	all	3	.93	.88	SENS:	.99 (.00)	.90 (.02)	.52 (.07)
		(fixed)	(.01)	(.01)	SPEC:	.89 (.02)	.96 (.01)	.99 (.00)
					AUC:	.94 (.01)	.93 (.01)	.75 (.03)
MLR	all		.89	.90	SENS:	.92 (.02)	.86 (.04)	.75 (.09)
			(.02)	(.02)	SPEC:	.97 (.03)	.95 (.01)	.94 (.01)
					AUC:	.95 (.01)	.90 (.02)	.85 (.05)
RMNL[c]	12.50[d]		.92	.90	SENS:	.96 (.01)	.91 (.02)	.66 (.13)
w/aMV	(4.04)		(.02)	(.03)	SPEC:	.96 (.03)	.95 (.02)	.97 (.01)
					AUC:	.96 (.01)	.93 (.02)	.81 (.07)
RMNL[e]	12.50[d]		.92	.89	SENS:	.96 (.01)	.91 (.04)	.61 (.15)
w/MV	(4.04)		(.02)	(.03)	SPEC:	.95 (.03)	.95 (.02)	.97 (.01)
					AUC:	.96 (.01)	.93 (.02)	.79 (.07)

[a] pre-determined number of subsets in a partition

[b] mean of the AUCs from the three classes

[c] RMNL with the combining method of averaging predictive probabilities

[d] average number of selected variables in a bootstrap sample to fit a multinomial logit model, chosen among the numbers from 5 to 20 in the learning phase

[e] RMNL with the combining method of majority voting

Table 3.4: McNemar's Test Results of GIB Data

Models		p-value			
		Overall	Upper	Lower	Mid
mLORENS w/2pt : MLR	ACC:	<0.0001	<0.0001	<0.0001	<0.0001
	SENS:		<0.0001	0.0152	0.0576
	SPEC:		<0.0001	<0.0001	<0.0001
mLORENS w/2pt : RMNL w/aMV	ACC:	0.0150	0.1334	0.0795	0.0252
	SENS:		<0.0001	0.0518	0.7119
	SPEC:		0.0008	0.0002	0.0081
MLR : RMNL w/aMV	ACC:	<0.0001	<0.0001	0.0004	0.0002
	SENS:		<0.0001	<0.0001	0.0117
	SPEC:		0.2031	0.2515	<0.0001

Figure 3.3: Accuracies for GIB Data

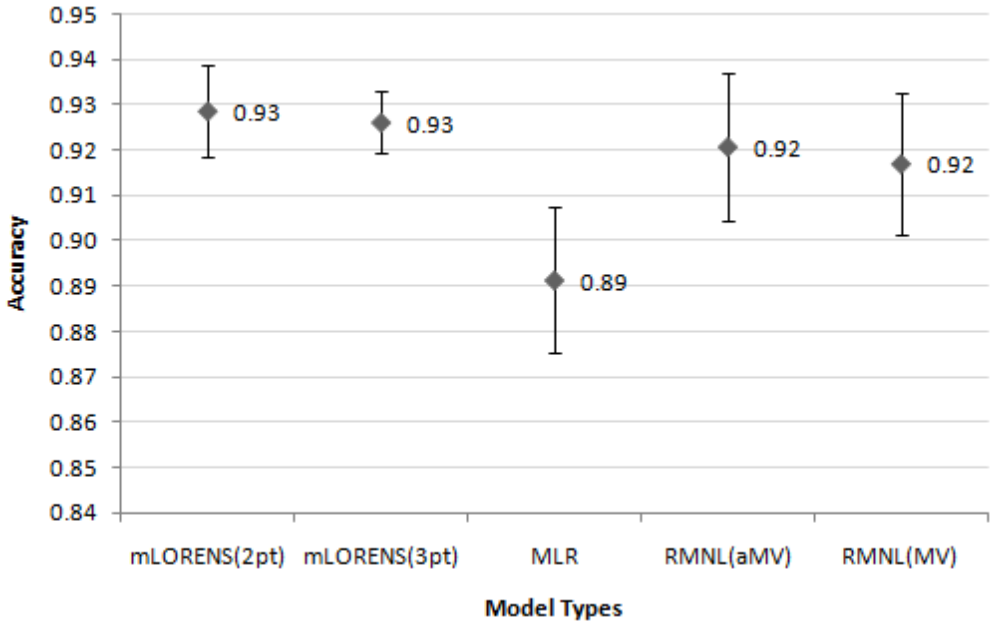
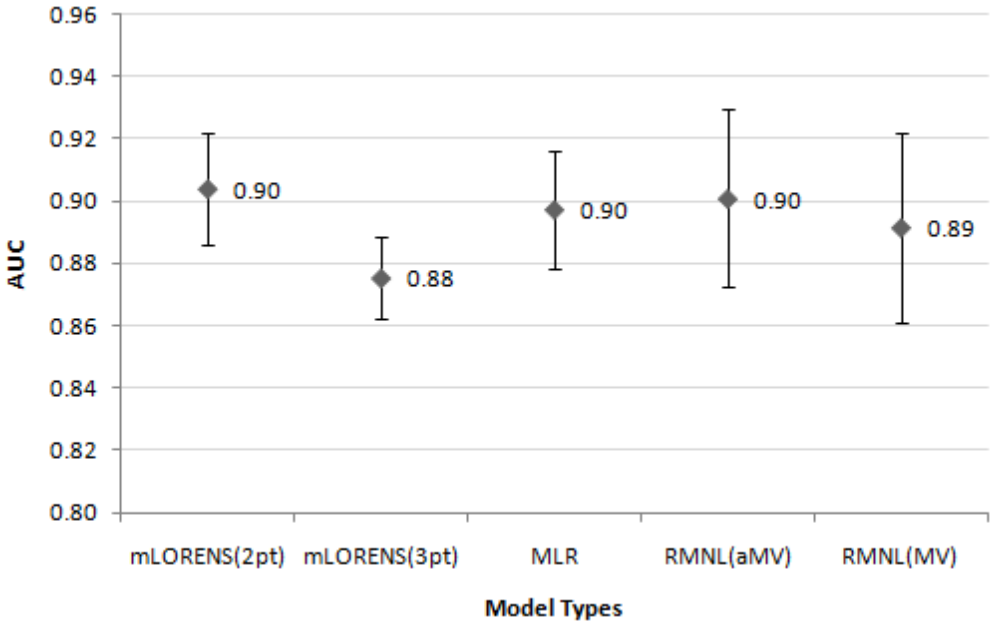


Figure 3.4: AUCs for GIB Data



3.3 Breast Cancer Data

Another example used for illustrating the multi-way classification is determination of stromal signatures in breast carcinoma. Two types of tumors with fibroblastic features, solitary fibrous tumor (SFT) and desmoid-type fibromatosis (DTF), were examined by DNA microarray analysis and the two tumor types were found to differ in their patterns of expression in carious functional categories of genes[32]. Their findings suggest that gene expression patterns characteristic of soft tissue tumors can be used to discover new markers for normal connective tissue cells. Compared to the GIB data the breast cancer data are high-dimensional. The data set contains 4148 variables on 57 tumor patients. A classification method can be applied to classify the data into DTF, SFT, and other types of tumors. Ten cases of DTF and 13 cases of benign SFT were compared to 34 other previously examined soft tissue tumors with expression profiling on 42,000 element cDNA microarray corresponding to approximately 36,000 unique gene sequences. The data were obtained from the web site http://smd.standford.edu/cgi-bin/publication/viewPublication.l?pub_no=436.

Performance of the three methods were evaluated by conducting 20 repetitions of 10-fold CV for each method, and the result is provided in Table 3.5. The results of McNemar's tests are in Table 3.6. Figures 3.5 and 3.6 represent the accuracies and the AUCs with 1-sd bars. For mLORENS, the partition size was searched in learning phase by selecting the one with the highest accuracy,

and the average was 207 (sd 79), which means that the number of variables in each partition is about 20. To apply the MLR model in this example, variable selection is necessary because the data are high-dimensional with 4148 predictors and only 57 subjects. By the BW ratio criterion, 10, 20 and 30 variables were selected in the training phase. For RMNL, 100 bootstrap samples were used to construct MLR models, and the number of variables having the highest accuracy was searched from 15 to 30 in the learning phase. The average number of selected variable was 26 (sd 4) for both of the MV and aMV approaches.

mLORENS outperformed a single MLR in all measurements regardless of the number of selected variables in MLR models. Among the 3 MLR models with different numbers of variables, the average accuracy and mean AUC of the models with 30 variables were higher than those with 10 or 20 variables. The MLR model with 30 variables was compared to mLORENS. The difference between the two models in terms of overall accuracy was significantly different (p-value < 0.0001). mLORENS showed higher sensitivity and specificity than MLR for all classes. MLR appeared to be more balanced in sensitivity and specificity in the classes of SFT and other tumors, but the numbers were lower than those of mLORENS. In RMNL, there was no effect of the method of combining MLR models. RMNL showed similar results as mLORENS in all measures with a better performance than MLR models. The p-value in accuracy between mLORENS and RMNL was 0.4098, and less than 0.0001 for the comparison of RMNL and MLR models. In sensitivity and specificity,

mLORENS and RMNL showed similar results, but the lowest number among the sensitivities of three classes for mLORENS was significantly lower than that of RMNL, and the same thing was found for specificities. From this result, we can conclude that the balance of sensitivity and specificity was better in RMNL than mLORENS for this example, and RMNL showed slightly higher AUC than mLORENS. Consequently, this example showed the advantage of the ensemble methods in classifying a high dimensional data set. mLORENS performs significantly better than MLR. The performance of RMNL with random variable selection was as good as that of mLORENS.

Table 3.5: Breast Cancer Data: Performance (SD in parentheses) of mLORENS, MLR, and RMNL. Twenty repetitions of 10-fold CV were used for each method.

Model	♯Var.[a]	♯Part.[b]	ACC	AUC[c]		DTF	SFT	other
mLORENS	all	207	.92	.92	SENS:	1.00 (.00)	.68 (.08)	.99 (.02)
		(79)	(.02)	(.02)	SPEC:	.99 (.01)	1.00 (.00)	.82 (.04)
					AUC:	1.00 (.01)	.84 (.04)	.91 (.03)
MLR	10		.66	.73	SENS:	.87 (.15)	.43 (.14)	.69 (.08)
	(fixed)		(.07)	(.07)	SPEC:	.93 (.04)	.83 (.05)	.64 (.12)
					AUC:	.90 (.07)	.63 (.07)	.67 (.08)
	20		.67	.76	SENS:	.84 (.13)	.58 (.14)	.66 (.09)
	(fixed)		(.07)	(.05)	SPEC:	.93 (.03)	.79 (.06)	.73 (.10)
					AUC:	.89 (.06)	.69 (.07)	.70 (.07)
	30		.69	.78	SENS:	.88 (.09)	.66 (.12)	.65 (.08)
	(fixed)		(.05)	(.04)	SPEC:	.92 (.04)	.79 (.06)	.79 (.09)
					AUC:	.90 (.04)	.73 (.05)	.72 (.05)
RMNL	25.86[d]		.93	.93	SENS:	1.00 (.02)	.75 (.07)	.98 (.02)
w/aMV	(3.92)		(.02)	(.02)	SPEC:	1.00 (.01)	.99 (.01)	.86 (.04)
					AUC:	1.00 (.01)	.87 (.03)	.92 (.02)
RMNL	25.86[d]		.93	.92	SENS:	1.00 (.02)	.74 (.08)	.98 (.03)
w/MV	(3.92)		(.03)	(.02)	SPEC:	1.00 (.01)	.99 (.01)	.85 (.05)
					AUC:	1.00 (.01)	.86 (.04)	.91 (.03)

[a] number of selected variables chosen in the training phase

[b] average number of mutually exclusive subsets of predictors in a partition, chosen in the trainiing phase

[c] mean of the AUCs from the three classes

[d] average number of selected variables in a bootstrap sample to fit a multinomial logit model, chosen among the numbers from 15 to 30 in the learning phase

Table 3.6: McNemar's Test Results of Breast Cancer Data

Models		p-value Overall	DTF	SFT	other
mLORENS : MLR w/30var	ACC:	<0.0001	<0.0001	<0.0001	<0.0001
	SENS:		<0.0001	0.5805	<0.0001
	SPEC:		<0.0001	<0.0001	0.1636
mLORENS : RMNL w/aMV	ACC:	0.4098	1.0000	0.4610	0.4098
	SENS:		1.0000	0.0068	0.0524
	SPEC:		0.6831	0.0026	0.0124
MLR w/30var : RMNL w/aMV	ACC:	<0.0001	<0.0001	<0.0001	<0.0001
	SENS:		<0.0001	0.0133	<0.0001
	SPEC:		<0.0001	<0.0001	0.0023

Figure 3.5: Accuracies for Breast Cancer Data

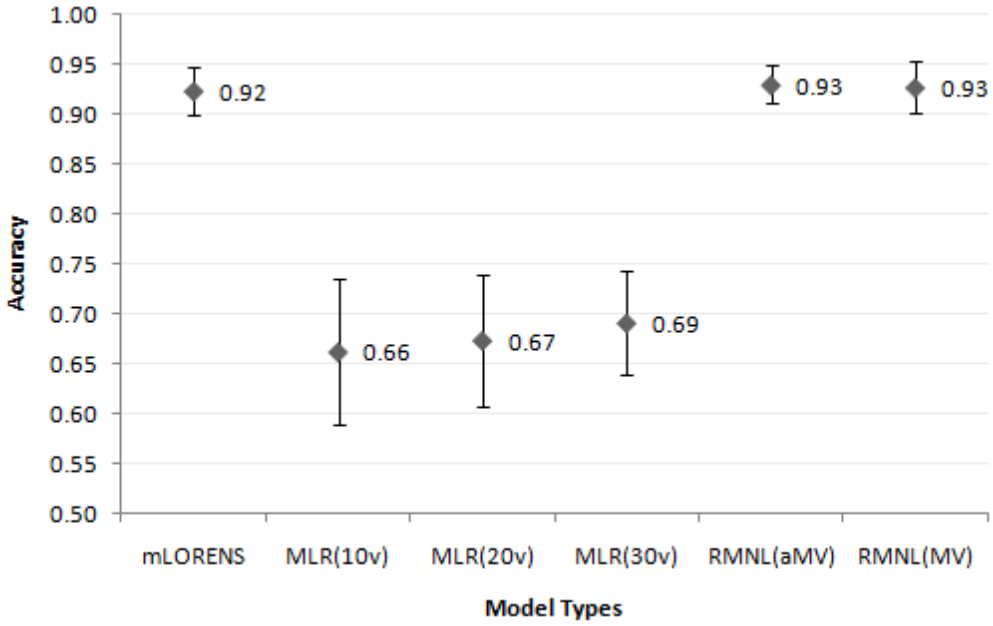
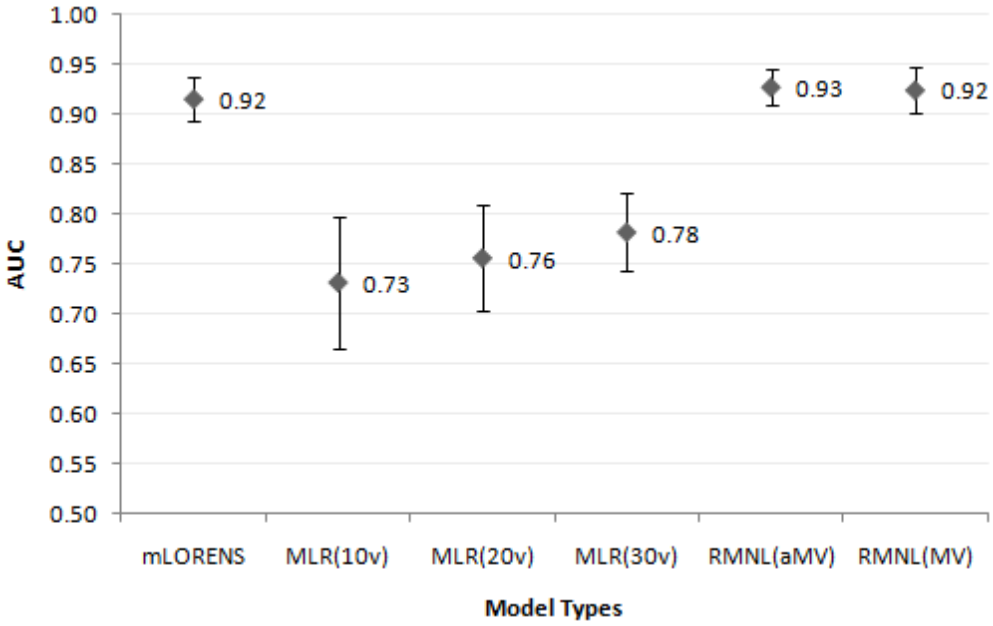


Figure 3.6: AUCs for Breast Cancer Data



Chapter 4

Simulation Study

4.1 Simulation Experiment 1

Simulation study was conducted on two different designs. First one is high-dimensional data, and the second one is relatively small data mimicking the GIB data. Details of the first design are given in this section. Two data sets, one for training and the other for testing, with 120 subjects and 500 predictors were generated. Average performance of 100 pairs of these data were calculated. The ratio of the classes was given as 40:40:40. The 500 predictors contain 50 significant variables constructed either independently or with correlation from normal distributions with different means with variance (σ^2) of 1 or 4, and the remaining 450 variables were generated from a normal distribution with the same mean to serve as noise.

Figure 4.1 displays the data design for the simulation experiment with independent predictors. Among the 50 significant variables, the first 10 variables

were generated from $N(0, \sigma^2)$ for class 1, $N(1, \sigma^2)$ for class 2, and $N(2, \sigma^2)$ for class 3. Next 10 variables were generated from $N(0, \sigma^2)$ for class 1, $N(1, \sigma^2)$ for classes 2 and 3. Next 10 variables were generated from $N(0, \sigma^2)$ for classes 1 and 2, and $N(1, \sigma^2)$ for class 3. Next 5 variables were generated from $N(0, \sigma^2)$ for 35 samples in class 1, $N(1, \sigma^2)$ for 5 samples in class 1 and 30 samples in class 2, and $N(2, \sigma^2)$ for 10 samples in class 2 and all samples in class 3. Next 5 variables were generated from $N(0, \sigma^2)$ for all samples in class 1 and 10 samples in class 2, $N(1, \sigma^2)$ for 30 samples in class 2 and 5 samples in class 3, and $N(2, \sigma^2)$ for the remaining 35 samples in class 3. Next 5 variables were generated from $N(0, \sigma^2)$ for 35 samples in class 1, $N(1, \sigma^2)$ for 5 samples each in classes 1 and 3, and all samples in class 2, and $N(2, \sigma^2)$ for the remaining 35 samples in class 3. Next 5 variables were generated from $N(0, \sigma^2)$ for all samples in class 1 and 5 samples in class 2, $N(1, \sigma^2)$ for 30 samples in class 2, and $N(2, \sigma^2)$ for 5 samples in class 2 and all samples in class 3. The remaining 450 variables were independently generated from $N(0, \sigma^2)$.

For the data with correlated predictors, a correlation matrix was created before generating the simulation data. The upper diagonal elements of positive definite correlation matrix were generated from Uniform$(0, 0.8)$. This correlation matrix was applied only to the 50 significant variables, and the remaining 450 noise variables were independently generated from $N(0, \sigma^2)$ like the data with independent predictors.

Figure 4.1: Data design for simulation experiment 1 with independent predictors

<table>
<tr><td>120 samples</td><td colspan="7">Significant variables : 50</td><td>Noise :450</td></tr>
<tr><td></td><td>10</td><td>10</td><td>10</td><td>5</td><td>5</td><td>5</td><td>5</td><td></td></tr>
<tr><td>Class 1 : 40</td><td>N(0,σ^2) : 40</td><td>N(0,σ^2) : 40</td><td rowspan="2">N(0,σ^2) : 80</td><td>N(0,σ^2) : 35</td><td rowspan="2">N(0,σ^2) : 50</td><td>N(0,σ^2) : 35</td><td>N(0,σ^2) : 45</td><td rowspan="3">N(0,σ^2) : 120</td></tr>
<tr><td>Class 2 : 40</td><td>N(1,σ^2) : 40</td><td rowspan="2">N(1,σ^2) : 80</td><td>N(1,σ^2) : 35</td><td rowspan="2">N(1,σ^2) : 35</td><td rowspan="2">N(1,σ^2) : 50</td><td rowspan="2">N(1,σ^2) : 30</td></tr>
<tr><td>Class 3 : 40</td><td>N(2,σ^2) : 40</td><td>N(1,σ^2) : 40</td><td>N(2,σ^2) : 50</td><td>N(2,σ^2) : 35</td><td>N(2,σ^2) : 35</td><td>N(2,σ^2) : 45</td></tr>
</table>

The performance of three classification models, mLORENS, MLR and RMNL, were evaluated using the test data sets to the models fitted from the learning sets. The average of the 100 test results is provided in Tables 4.1, 4.2, 4.4, 4.5, 4.7, 4.8, 4.10, and 4.11. Tables 4.1 and 4.2 are the results for the data with independent variables and standard deviation 1, and Tables 4.4 and 4.5 are the results for the data with correlated variables and standard deviation 1. Tables 4.7 and 4.8 are the results for the data with independent variables and standard deviation 2, and Tables 4.10 and 4.11 are the results for the data with correlated variables and standard deviation 2. Comparison among mLORENS, MLR with 10 variables selected in learning phase, and RMNL was conducted using McNemar's test for the data with standard deviation 1, and the results were provided in Table 4.3 for the data with independent variables, and Table 4.6 for the data with correlated variables. For the data with standard deviation 2, mLORENS, MLR with 10 variables selected in learning phase, MLR with 50 variables selected in learning phase, MLR with all 50 significant variables, and RMNL were compared, and the results are given in Table 4.9 for the data with independent variables and in Table 4.12 for the data with correlated variables. Figures 4.2, 4.4, 4.6, and 4.8 depict the accuracies and Figures 4.3, 4.5, 4.7, and 4.9 depict the AUCs for each data design.

In mLORENS, the partition size with the highest accuracy was searched in learning phase. Some fixed partition sizes were also tried, and as expected, the result with the search procedure was better than that with a fixed size. For MLR, variable selection was conducted by BW ratio. Since the number

of significant variables is unknown in practice, it was pre-assigned as 10, 30, 50 or 70 and the given number of variables were selected in learning phase. To see how many significant variables were actually selected, the number of selected variables from the 50 significant variables was counted. Unlike a real data analysis, the significant variables are known, thus the MLR model with these 50 significant variables was also tried as an ideal case. For RMNL, 100 bootstrap samples were used to construct a model. The number of predictors in the model yielding the highest accuracy was searched using a bootstrap sample in training phase among 10, 16, 22, 28, 34, 40, 46, 52, and 58.

mLORENS showed higher performance than MLR and RMNL in all performance measures. Among the 5 MLR models with different numbers of predictors, the model with all 50 significant predictors did not show the best performance, while the model with only 10 predictors was the best in overall accuracy and mean AUC. When 50 variables were used in MLR, the numbers of selected variables from the significant predictors were about 49 for the data with standard deviation 1 and about 39 for the data with standard deviation 2, but the accuracies were similar to the MLR model with the 50 significant variables. This result might be explained by the structure of the data. Only a few predictors possess most of the information and the rest of the significant predictors could be redundant, because the data contain variables from only 8 distinct sets of distributions except for errors. On the other hand, mLORENS seems to take advantage in dealing with redundancy. The searched partition size is about 30 to 38, with 13 to 17 variables in each partition. This implies

that only one or two significant variables may be included in each partition. Thus it would reduce the chance of redundancy.

Variable selection through BW ratio worked well on these data sets. In the data with standard deviation 1, almost all the significant variables were selected, while fewer significant variables were chosen in the data with standard deviation 2. In RMNL, the overall accuracy for the combining method of aMV was significantly higher than that of MV in all data designs (p-value was less than 0.0001) except for the data with independent predictors and standard deviation 2 (p-value was 0.1817), even though the numbers of selected variables of aMV were little less than those of MV. In AUC, MV and aMV showed similar results.

The accuracy of mLORENS was significantly higher than the highest accuracies of MLR and RMNL. The p-values were less than 0.0001 for all comparisons in all four different designs. As expected, the results with standard deviation 2 are poorer than those of standard deviation 1, and the data with correlated predictors showed poorer performance than the data with independent predictors. In the data with standard deviation 2, MLR models with 10, 30, 50 selected variables and all 50 significant variables showed similar accuracies when the predictors were independent. However, MLR models with 10 or 30 selected variables showed higher accuracies than MLR with 50 selected variables or all 50 significant variables when the predictors were correlated. Using more variables in correlated data negatively affected to the accuracy. Even though the 3 classes in the response were supposed to be balanced, the

50 significant variables did not fairly predict the 3 classes. We can check from the data design that the data have much less accurate information for class 2 than for classes 1 and 3. Hence, in all four different data types and all classification methods, sensitivity of class 2 was lower than that of the other 2 classes. As a result, the sensitivity and specificity became unbalanced in class 2. Hence, it is not easy from examining sensitivity and specificity to determine which method is better than the others in terms of balance of sensitivity and specificity. However, the mean AUC of mLORENS was the highest compared to those of the other methods in all four data types, and the numbers became lower as the variables of the data became correlated and the standard deviation became larger.

Table 4.1: Simulation Experiment 1: Performances (SD in parentheses) of mLORENS, MLR, and RMNL. Independent predictors from normal distribution with standard deviation 1.

Model	♯var.[a]	♯sig. var[b]	♯part.[c]	ACC	AUC[d]		class 1	class 2	class 3
mLORENS	all		38.0	.92	.94	SENS:	.99 (.01)	.76 (.07)	.99 (.01)
			(7.6)	(.02)	(.02)	SPEC:	.94 (.03)	.99 (.01)	.94 (.03)
						AUC:	.97 (.01)	.88 (.03)	.97 (.01)
	all		10	.89	.92	SENS:	.97 (.03)	.76 (.08)	.94 (.04)
			(fixed)	(.03)	(.02)	SPEC:	.92 (.03)	.96 (.03)	.95 (.02)
						AUC:	.95 (.02)	.86 (.04)	.95 (.02)
	all		20	.92	.94	SENS:	.99 (.01)	.79 (.07)	.99 (.02)
			(fixed)	(.02)	(.02)	SPEC:	.95 (.02)	.99 (.01)	.95 (.03)
						AUC:	.97 (.01)	.89 (.03)	.97 (.01)
	all		30	.92	.94	SENS:	.99 (.01)	.78 (.07)	.99 (.01)
			(fixed)	(.02)	(.02)	SPEC:	.95 (.03)	.99 (.01)	.94 (.03)
						AUC:	.97 (.01)	.88 (.03)	.97 (.01)
MLR	10	10		.83	.88	SENS:	.89 (.07)	.75 (.08)	.86 (.08)
with		(0)		(.04)	(.03)	SPEC:	.94 (.03)	.88 (.05)	.93 (.03)
variable						AUC:	.92 (.04)	.82 (.04)	.89 (.04)
selection	30	29.2		.82	.86	SENS:	.88 (.07)	.72 (.10)	.86 (.08)
		(.8)		(.05)	(.04)	SPEC:	.90 (.04)	.88 (.05)	.95 (.03)
						AUC:	.89 (.04)	.80 (.06)	.90 (.04)
	50	49.2		.74	.81	SENS:	.80 (.08)	.69 (.08)	.74 (.09)
		(.8)		(.04)	(.03)	SPEC:	.88 (.04)	.81 (.05)	.91 (.04)
						AUC:	.84 (.04)	.75 (.04)	.82 (.05)
	70	49.2		.67	.75	SENS:	.72 (.09)	.63 (.09)	.66 (.10)
		(.7)		(.06)	(.04)	SPEC:	.87 (.05)	.76 (.06)	.87 (.04)
						AUC:	.80 (.05)	.70 (.05)	.77 (.06)
MLR	50			.74	.81	SENS:	.80 (.09)	.68 (.10)	.73 (.10)
w/ signif.				(.05)	(.04)	SPEC:	.88 (.05)	.82 (.06)	.91 (.04)
variables						AUC:	.84 (.05)	.75 (.05)	.82 (.06)

[a] number of selected variables chosen in the training phase

[b] number of significant variables among the selected variables

[c] average number of mutually exclusive subsets of predictors in a partition, chosen in the training phase

[d] mean of the AUCs from the three classes

Table 4.2: Simulation Experiment 1: Performances (SD in parentheses) of mLORENS, MLR, and RMNL. Independent predictors from normal distribution with standard deviation 1 (continued).

Model	♯var.[a]	♯sig. var[b]	♯part.[c]	ACC	AUC[d]		class 1	class 2	class 3
RMNL	25.7[e]			.86	.89	SENS:	.96 (.05)	.66 (.10)	.95 (.05)
w/ aMV	(10.9)			(.04)	(.03)	SPEC:	.91 (.04)	.96 (.04)	.92 (.04)
						AUC:	.94 (.03)	.81 (.05)	.94 (.03)
RMNL	32.9[e]			.85	.89	SENS:	.95 (.06)	.66 (.09)	.93 (.05)
w/ MV	(10.1)			(.03)	(.03)	SPEC:	.91 (.04)	.94 (.05)	.92 (.03)
						AUC:	.93 (.03)	.80 (.04)	.93 (.03)

[a] number of selected variables chosen in the training phase

[b] number of significant variables among the selected variables

[c] average number of mutually exclusive subsets of predictors in a partition, chosen in the training phase

[d] mean of the AUCs from the three classes

[e] average number of selected variables in a bootstrap sample to fit a multinomial logit model, chosen among the numbers of 10, 16, 22, 28, 34, 40, 46, 52, 58 in the learning phase

Table 4.3: McNemar's Test Results of Simulation Experiment 1: Independent predictors with standard deviation 1

Models		p-value			
		Overall	Class 1	Class 2	Class 3
mLORENS : MLR w/10var	ACC:	<0.0001	<0.0001	<0.0001	<0.0001
	SENS:		<0.0001	0.1756	<0.0001
	SPEC:		0.5551	<0.0001	0.0349
mLORENS : RMNL w/aMV	ACC:	<0.0001	<0.0001	<0.0001	<0.0001
	SENS:		<0.0001	<0.0001	<0.0001
	SPEC:		<0.0001	<0.0001	<0.0001
MLR w/10var : RMNL w/aMV	ACC:	<0.0001	0.5813	<0.0001	<0.0001
	SENS:		<0.0001	<0.0001	<0.0001
	SPEC:		<0.0001	<0.0001	0.0003
RMNL w/aMV : RMNL w/MV	ACC:	<0.0001	0.0987	<0.0001	0.0004
	SENS:		<0.0001	0.5815	<0.0001
	SPEC:		0.4577	<0.0001	0.9600

Figure 4.2: Accuracies for Simulation Experiment 1: Independent predictors with standard deviation 1

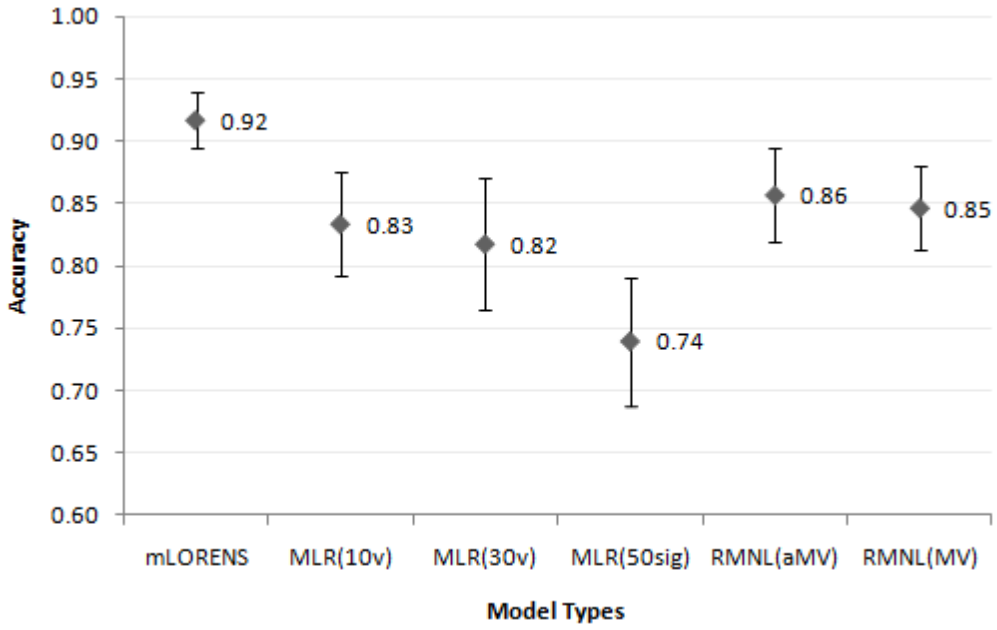


Figure 4.3: AUCs for Simulation Experiment 1: Independent predictors with standard deviation 1

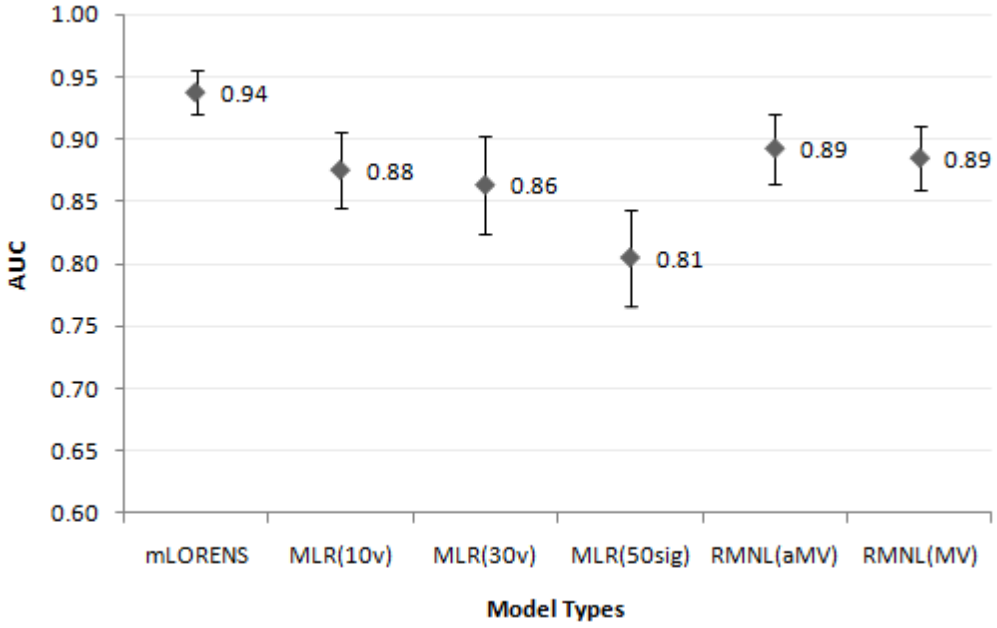


Table 4.4: Simulation Experiment 1: Performances (SD in parentheses) of mLORENS, MLR, and RMNL. Correlated significant predictors from multivariate normal distribution with standard deviation 1.

Model	♯var.[a]	♯sig. var[b]	♯part.[c]	ACC	AUC[d]		class 1	class 2	class 3
mLORENS	all		37.7	.88	.91	SENS:	.98 (.02)	.69 (.08)	.98 (.02)
			(7.9)	(.03)	(.02)	SPEC:	.92 (.03)	.98 (.01)	.92 (.03)
						AUC:	.95 (.02)	.84 (.04)	.95 (.02)
	all		10	.84	.88	SENS:	.95 (.04)	.68 (.08)	.91 (.05)
			(fixed)	(.03)	(.02)	SPEC:	.90 (.03)	.93 (.03)	.94 (.03)
						AUC:	.92 (.02)	.80 (.04)	.92 (.03)
	all		20	.88	.91	SENS:	.97 (.03)	.71 (.08)	.97 (.03)
			(fixed)	(.03)	(.02)	SPEC:	.93 (.03)	.97 (.02)	.93 (.03)
						AUC:	.95 (.02)	.84 (.04)	.95 (.02)
	all		30	.89	.92	SENS:	.98 (.02)	.71 (.07)	.98 (.02)
			(fixed)	(.02)	(.02)	SPEC:	.93 (.03)	.98 (.02)	.93 (.03)
						AUC:	.95 (.02)	.84 (.04)	.95 (.02)
MLR	10	10		.76	.82	SENS:	.83 (.08)	.66 (.10)	.80 (.09)
with		(0)		(.05)	(.03)	SPEC:	.92 (.04)	.83 (.05)	.90 (.04)
variable						AUC:	.87 (.04)	.74 (.05)	.85 (.05)
selection	30	29.2		.75	.81	SENS:	.82 (.07)	.65 (.09)	.79 (.08)
		(.8)		(.05)	(.04)	SPEC:	.90 (.04)	.81 (.06)	.92 (.04)
						AUC:	.86 (.04)	.73 (.05)	.85 (.05)
	50	49.2		.66	.74	SENS:	.71 (.09)	.58 (.09)	.67 (.10)
		(.8)		(.06)	(.04)	SPEC:	.87 (.05)	.74 (.06)	.87 (.04)
						AUC:	.79 (.05)	.66 (.05)	.77 (.06)
	70	49.2		.60	.70	SENS:	.63 (.08)	.55 (.08)	.61 (.10)
		(.8)		(.06)	(.04)	SPEC:	.85 (.05)	.70 (.06)	.84 (.05)
						AUC:	.74 (.05)	.62 (.05)	.72 (.06)
MLR	50			.66	.74	SENS:	.72 (.09)	.58 (.10)	.68 (.10)
w/signif.				(.06)	(.04)	SPEC:	.87 (.05)	.75 (.07)	.87 (.05)
variables						AUC:	.79 (.05)	.66 (.06)	.77 (.06)

[a] number of selected variables chosen in the training phase

[b] number of significant variables among the selected variables

[c] average number of mutually exclusive subsets of predictors in a partition, chosen in the training phase

[d] mean of the AUCs from the three classes

Table 4.5: Simulation Experiment 1: Performances (SD in parentheses) of mLORENS, MLR, and RMNL. Correlated significant predictors from multivariate normal distribution with standard deviation 1 (continued).

Model	♯var.[a]	♯sig. var[b]	♯part.[c]	ACC	AUC[d]		class 1	class 2	class 3
RMNL	24.5[e]			.83	.87	SENS:	.94 (.05)	.61 (.09)	.94 (.05)
w/aMV	(9.3)			(.03)	(.02)	SPEC:	.90 (.04)	.94 (.04)	.91 (.04)
						AUC:	.92 (.03)	.77 (.04)	.92 (.03)
RMNL	28.8[e]			.82	.86	SENS:	.94 (.05)	.59 (.09)	.92 (.05)
w/MV	(8.4)			(.03)	(.02)	SPEC:	.89 (.04)	.93 (.04)	.91 (.04)
						AUC:	.91 (.03)	.76 (.04)	.92 (.03)

[a] number of selected variables chosen in the training phase

[b] number of significant variables among the selected variables

[c] average number of mutually exclusive subsets of predictors in a partition, chosen in the training phase

[d] mean of the AUCs from the three classes

[e] average number of selected variables in a bootstrap sample to fit a multinomial logit model, chosen among the numbers of 10, 16, 22, 28, 34, 40, 46, 52, 58 in the learning phase

Table 4.6: McNemar's Test Results of Simulation Experiment 1: Correlated predictors with standard deviation 1

Models		p-value			
		Overall	Class 1	Class 2	Class 3
mLORENS : MLR w/10var	ACC:	<0.0001	<0.0001	<0.0001	<0.0001
	SENS:		<0.0001	0.0017	<0.0001
	SPEC:		0.1784	<0.0001	<0.0001
mLORENS : RMNL w/aMV	ACC:	<0.0001	<0.0001	<0.0001	<0.0001
	SENS:		<0.0001	<0.0001	<0.0001
	SPEC:		<0.0001	<0.0001	<0.0001
MLR w/10var : RMNL w/aMV	ACC:	<0.0001	<0.0001	<0.0001	<0.0001
	SENS:		<0.0001	<0.0001	<0.0001
	SPEC:		<0.0001	<0.0001	0.0519
RMNL w/aMV : RMNL w/MV	ACC:	<0.0001	0.0012	<0.0001	0.0119
	SENS:		0.2073	0.0444	<0.0001
	SPEC:		0.0028	<0.0001	0.7656

Figure 4.4: Accuracies for Simulation Experiment 1: Correlated predictors with standard deviation 1

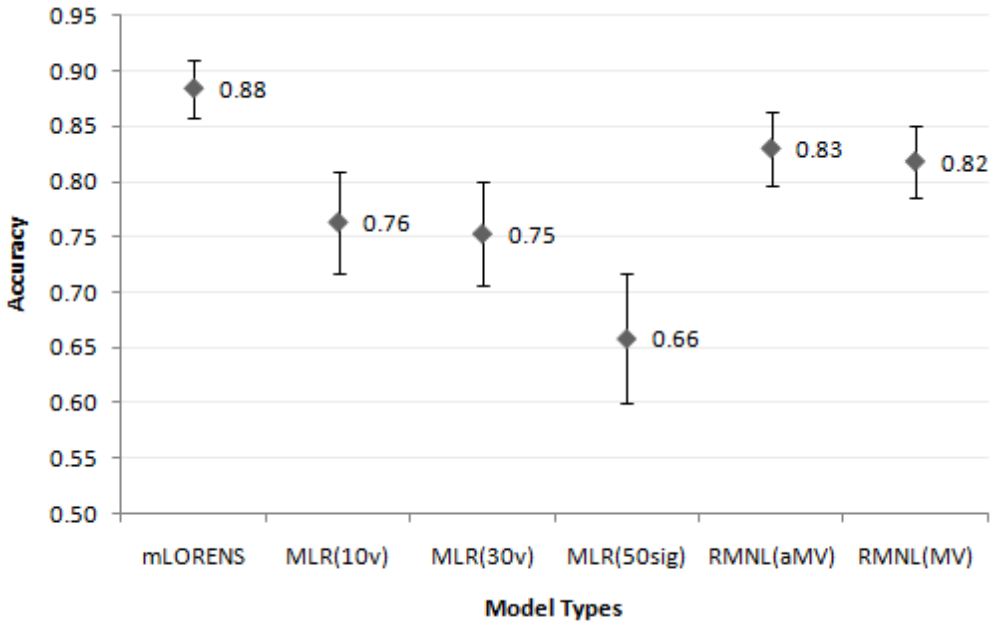


Figure 4.5: AUCs for Simulation Experiment 1: Correlated predictors with standard deviation 1

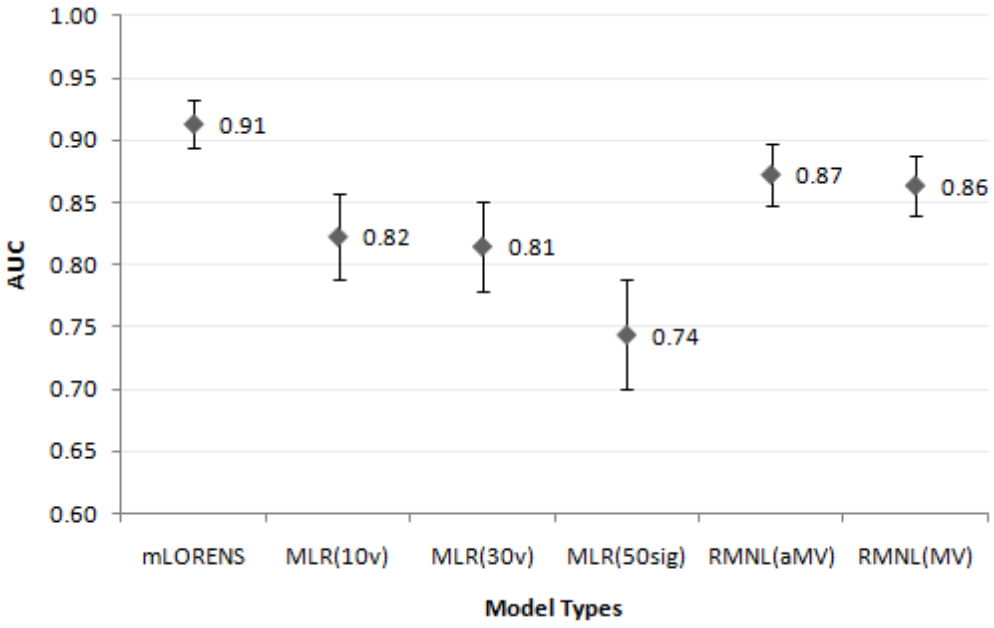


Table 4.7: Simulation Experiment 1: Performances (SD in parentheses) of mLORENS, MLR, and RMNL. Independent predictors from normal distribution with standard deviation 2.

Model	♯var.[a]	♯sig. var[b]	♯part.[c]	ACC	AUC[d]		class 1	class 2	class 3
mLORENS	all		29.6	.71	.78	SENS:	.86 (.06)	.40 (.08)	.86 (.06)
			(8.8)	(.04)	(.03)	SPEC:	.84 (.04)	.87 (.04)	.84 (.04)
						AUC:	.85 (.03)	.64 (.04)	.85 (.03)
	all		10	.67	.75	SENS:	.85 (.06)	.40 (.08)	.77 (.07)
			(fixed)	(.04)	(.03)	SPEC:	.80 (.05)	.83 (.04)	.87 (.04)
						AUC:	.82 (.04)	.62 (.04)	.82 (.04)
	all		20	.70	.78	SENS:	.85 (.06)	.41 (.07)	.85 (.06)
			(fixed)	(.04)	(.03)	SPEC:	.84 (.04)	.87 (.04)	.85 (.04)
						AUC:	.85 (.03)	.64 (.04)	.85 (.03)
	all		30	.71	.78	SENS:	.86 (.06)	.40 (.08)	.86 (.06)
			(fixed)	(.04)	(.03)	SPEC:	.84 (.04)	.87 (.03)	.84 (.04)
						AUC:	.85 (.03)	.64 (.04)	.85 (.03)
MLR	10	9.6		.61	.71	SENS:	.69 (.08)	.46 (.09)	.68 (.09)
with		(.6)		(.05)	(.04)	SPEC:	.83 (.05)	.76 (.06)	.83 (.05)
variable						AUC:	.76 (.05)	.61 (.05)	.75 (.06)
selection	30	20.2		.60	.70	SENS:	.68 (.09)	.46 (.11)	.67 (.09)
		(1.9)		(.06)	(.04)	SPEC:	.83 (.05)	.74 (.07)	.84 (.05)
						AUC:	.75 (.05)	.60 (.06)	.75 (.05)
	50	38.9		.60	.70	SENS:	.70 (.09)	.50 (.09)	.61 (.10)
		(2.0)		(.04)	(.03)	SPEC:	.82 (.05)	.73 (.06)	.86 (.05)
						AUC:	.76 (.05)	.62 (.05)	.73 (.05)
	70	39.5		.55	.66	SENS:	.63 (.09)	.45 (.09)	.56 (.10)
		(2.0)		(.05)	(.04)	SPEC:	.79 (.05)	.70 (.06)	.83 (.05)
						AUC:	.71 (.05)	.58 (.05)	.69 (.05)
MLR	50			.61	.71	SENS:	.70 (.11)	.50 (.10)	.63 (.10)
w/signif.				(.06)	(.04)	SPEC:	.82 (.06)	.74 (.07)	.86 (.05)
variables						AUC:	.76 (.06)	.62 (.05)	.75 (.05)

[a] number of selected variables chosen in the training phase

[b] number of significant variables among the selected variables

[c] average number of mutually exclusive subsets of predictors in a partition, chosen in the training phase

[d] mean of the AUCs from the three classes

Table 4.8: Simulation Experiment 1: Performances (SD in parentheses) of mLORENS, MLR, and RMNL. Independent predictors from normal distribution with standard deviation 2 (continued).

Model	♯var.[a]	♯sig. var[b]	♯part.[c]	ACC	AUC[d]		class 1	class 2	class 3
RMNL	27.9[e]			.64	.73	SENS:	.79 (.08)	.38 (.08)	.74 (.09)
w/aMV	(10.0)			(.04)	(.03)	SPEC:	.81 (.05)	.81 (.06)	.83 (.04)
						AUC:	.80 (.04)	.59 (.04)	.79 (.04)
RMNL	31.4[e]			.63	.72	SENS:	.77 (.10)	.39 (.09)	.74 (.09)
w/MV	(9.7)			(.05)	(.03)	SPEC:	.81 (.05)	.80 (.06)	.84 (.05)
						AUC:	.79 (.05)	.60 (.05)	.79 (.04)

[a] number of selected variables chosen in the training phase

[b] number of significant variables among the selected variables

[c] average number of mutually exclusive subsets of predictors in a partition, chosen in the training phase

[d] mean of the AUCs from the three classes

[e] average number of selected variables in a bootstrap sample to fit a multinomial logit model, chosen among the numbers of 10, 16, 22, 28, 34, 40, 46, 52, 58 in the learning phase

Table 4.9: McNemar's Test Results of Simulation Experiment 1: Independent predictors with standard deviation 2

		p-value			
Models		Overall	Class 1	Class 2	Class 3
mLORENS : MLR w/10var	ACC:	<0.0001	<0.0001	<0.0001	<0.0001
	SENS:		<0.0001	<0.0001	<0.0001
	SPEC:		0.0185	<0.0001	0.0126
mLORENS : RMNL w/aMV	ACC:	<0.0001	<0.0001	<0.0001	<0.0001
	SENS:		<0.0001	0.0291	<0.0001
	SPEC:		<0.0001	<0.0001	0.0820
MLR w/10var : RMNL w/aMV	ACC:	<0.0001	<0.0001	0.2915	<0.0001
	SENS:		<0.0001	<0.0001	<0.0001
	SPEC:		0.0025	<0.0001	0.2572
MLR w/10var : MLR w/sig.var	ACC:	1.0000	0.4259	0.9152	0.3667
	SENS:		0.3714	<0.0001	<0.0001
	SPEC:		0.0624	0.0005	<0.0001
MLR w/50var : MLR w/sig.var	ACC:	0.1282	0.4148	0.5179	0.0440
	SENS:		0.8136	0.5519	0.0857
	SPEC:		0.3996	0.7390	0.2652
RMNL w/aMV : RMNL w/MV	ACC:	0.1817	0.0079	0.5975	0.9310
	SENS:		0.0011	0.1116	0.3187
	SPEC:		0.4807	0.0376	0.2955

Figure 4.6: Accuracies for Simulation Experiment 1: Independent predictors with standard deviation 2

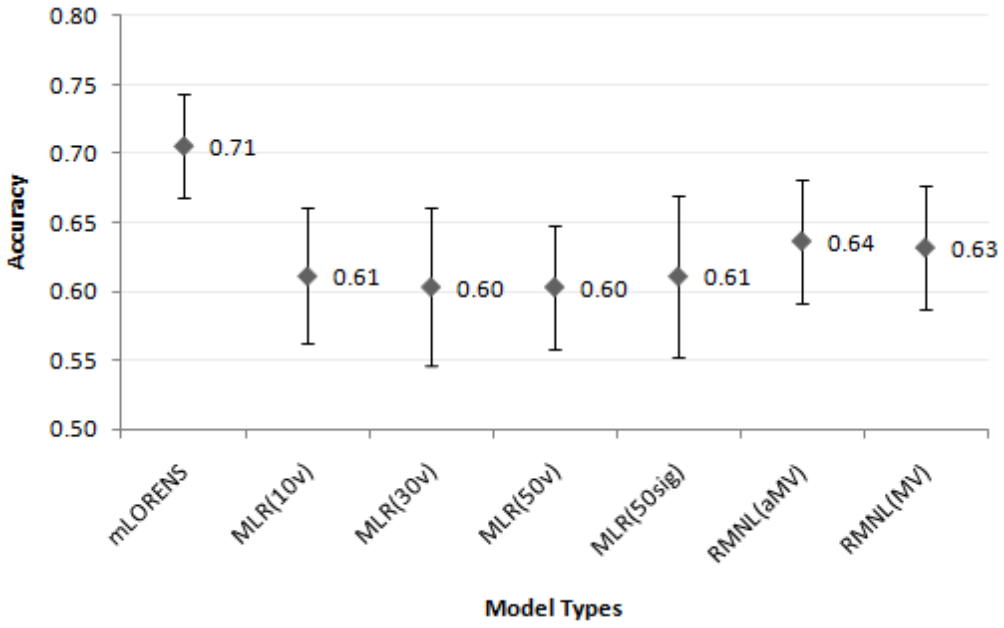


Figure 4.7: AUCs for Simulation Experiment 1: Independent predictors with standard deviation 2

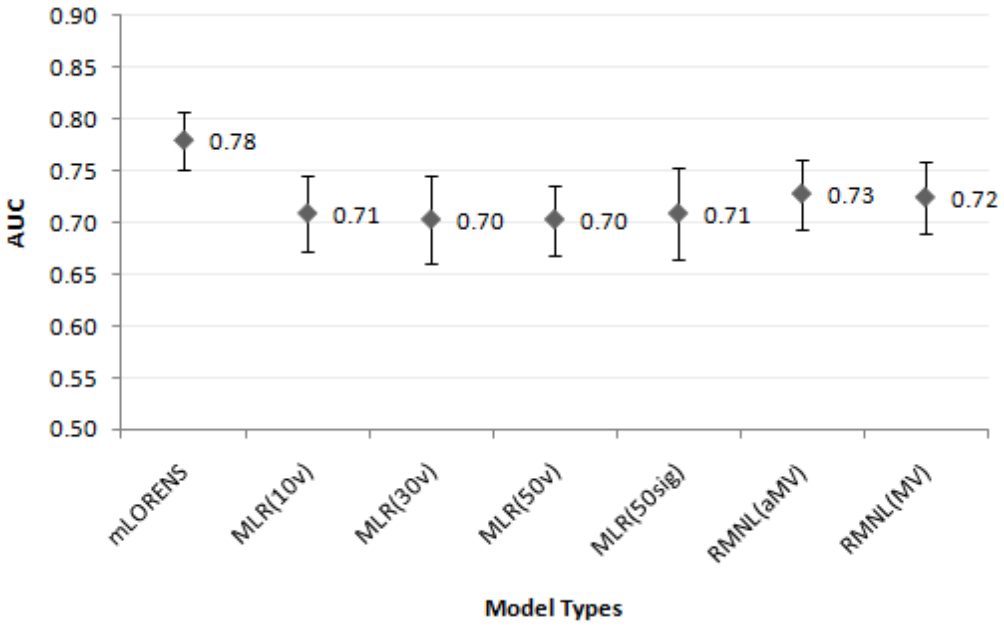


Table 4.10: Simulation Experiment 1: Performances (SD in parentheses) of mLORENS, MLR, and RMNL. Correlated significant predictors from multivariate normal distribution with standard deviation 2.

Model	♯var.[a]	♯sig. var[b]	♯part.[c]	ACC	AUC[d]		class 1	class 2	class 3
mLORENS	all		30.9	.68	.76	SENS:	.83 (.07)	.37 (.09)	.83 (.06)
			(8.4)	(.04)	(.03)	SPEC:	.82 (.04)	.86 (.04)	.84 (.04)
						AUC:	.83 (.03)	.62 (.05)	.83 (.04)
	all		10	.64	.73	SENS:	.82 (.07)	.37 (.09)	.74 (.08)
			(fixed)	(.04)	(.03)	SPEC:	.78 (.05)	.82 (.05)	.86 (.04)
						AUC:	.80 (.04)	.60 (.04)	.80 (.04)
	all		20	.68	.76	SENS:	.82 (.07)	.38 (.10)	.83 (.06)
			(fixed)	(.04)	(.03)	SPEC:	.83 (.04)	.85 (.05)	.84 (.04)
						AUC:	.82 (.03)	.62 (.05)	.83 (.03)
	all		30	.68	.76	SENS:	.83 (.07)	.37 (.09)	.84 (.06)
			(fixed)	(.03)	(.03)	SPEC:	.82 (.04)	.86 (.04)	.84 (.04)
						AUC:	.83 (.03)	.62 (.04)	.84 (.03)
MLR with variable selection	10	9.7		.58	.69	SENS:	.67 (.08)	.42 (.09)	.66 (.09)
		(.7)		(.04)	(.03)	SPEC:	.81 (.06)	.76 (.05)	.81 (.05)
						AUC:	.74 (.04)	.59 (.05)	.74 (.05)
	30	19.6		.58	.69	SENS:	.65 (.09)	.44 (.09)	.65 (.09)
		(2.4)		(.04)	(.03)	SPEC:	.82 (.05)	.72 (.06)	.83 (.06)
						AUC:	.74 (.05)	.58 (.05)	.74 (.05)
	50	38.4		.54	.66	SENS:	.62 (.10)	.44 (.09)	.56 (.08)
		(2.4)		(.05)	(.04)	SPEC:	.78 (.06)	.70 (.07)	.83 (.05)
						AUC:	.70 (.05)	.57 (.05)	.70 (.05)
	70	39.0		.49	.61	SENS:	.54 (.10)	.41 (.09)	.51 (.10)
		(2.4)		(.05)	(.04)	SPEC:	.76 (.05)	.67 (.06)	.80 (.06)
						AUC:	.65 (.05)	.54 (.05)	.65 (.06)
MLR w/signif. variables	50			.55	.66	SENS:	.64 (.10)	.43 (.07)	.59 (.09)
				(.05)	(.04)	SPEC:	.79 (.05)	.71 (.06)	.83 (.05)
						AUC:	.72 (.06)	.57 (.04)	.71 (.05)

[a] number of selected variables chosen in the training phase

[b] number of significant variables among the selected variables

[c] average number of mutually exclusive subsets of predictors in a partition, chosen in the training phase

[d] mean of the AUCs from the three classes

Table 4.11: Simulation Experiment 1: Performances (SD in parentheses) of mLORENS, MLR, and RMNL. Correlated significant predictors from multivariate normal distribution with standard deviation 2 (continued).

Model	♯var.[a]	♯sig. var[b]	♯part.[c]	ACC	AUC[d]		class 1	class 2	class 3
RMNL	25.5[e]			.62	.72	SENS:	.76 (.09)	.36 (.10)	.75 (.09)
w/aMV	(9.9)			(.05)	(.04)	SPEC:	.80 (.05)	.81 (.07)	.82 (.05)
						AUC:	.78 (.05)	.59 (.04)	.79 (.05)
RMNL	31.0[e]			.61	.71	SENS:	.73 (.10)	.38 (.09)	.72 (.09)
w/MV	(10.7)			(.05)	(.04)	SPEC:	.80 (.05)	.78 (.07)	.83 (.05)
						AUC:	.77 (.05)	.58 (.05)	.77 (.05)

[a] number of selected variables chosen in the training phase

[b] number of significant variables among the selected variables

[c] average number of mutually exclusive subsets of predictors in a partition, chosen in the training phase

[d] mean of the AUCs from the three classes

[e] average number of selected variables in a bootstrap sample to fit a multinomial logit model, chosen among the numbers of 10, 16, 22, 28, 34, 40, 46, 52, 58 in the learning phase

Table 4.12: McNemar's Test Results of Simulation Experiment 1: Correlated predictors with standard deviation 2

Models		p-value			
		Overall	Class 1	Class 2	Class 3
mLORENS : MLR w/10var	ACC:	<0.0001	<0.0001	<0.0001	<0.0001
	SENS:		<0.0001	<0.0001	<0.0001
	SPEC:		0.0158	<0.0001	<0.0001
mLORENS : RMNL w/aMV	ACC:	<0.0001	<0.0001	<0.0001	<0.0001
	SENS:		<0.0001	0.0654	<0.0001
	SPEC:		<0.0001	<0.0001	0.0004
MLR w/10var : RMNL w/aMV	ACC:	<0.0001	<0.0001	0.0004	<0.0001
	SENS:		<0.0001	<0.0001	<0.0001
	SPEC:		0.1005	<0.0001	0.0969
MLR w/10var : MLR w/sig.var	ACC:	<0.0001	<0.0001	<0.0001	0.0062
	SENS:		0.0002	0.3397	<0.0001
	SPEC:		0.0097	<0.0001	0.0153
MLR w/50var : MLR w/sig.var	ACC:	0.0232	<0.0001	0.7783	0.1062
	SENS:		0.0144	0.1059	0.0015
	SPEC:		0.0022	0.3779	0.4602
RMNL w/aMV : RMNL w/MV	ACC:	<0.0001	0.0056	0.0003	0.0251
	SENS:		<0.0001	0.0012	<0.0001
	SPEC:		0.6150	<0.0001	0.0501

Figure 4.8: Accuracies for Simulation Experiment 1: Correlated predictors with standard deviation 2

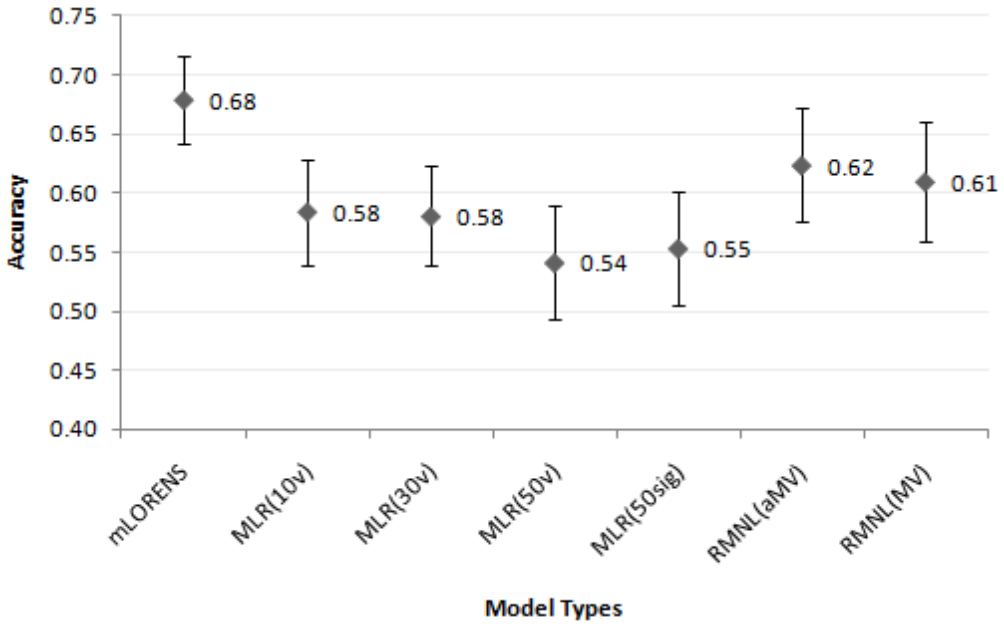
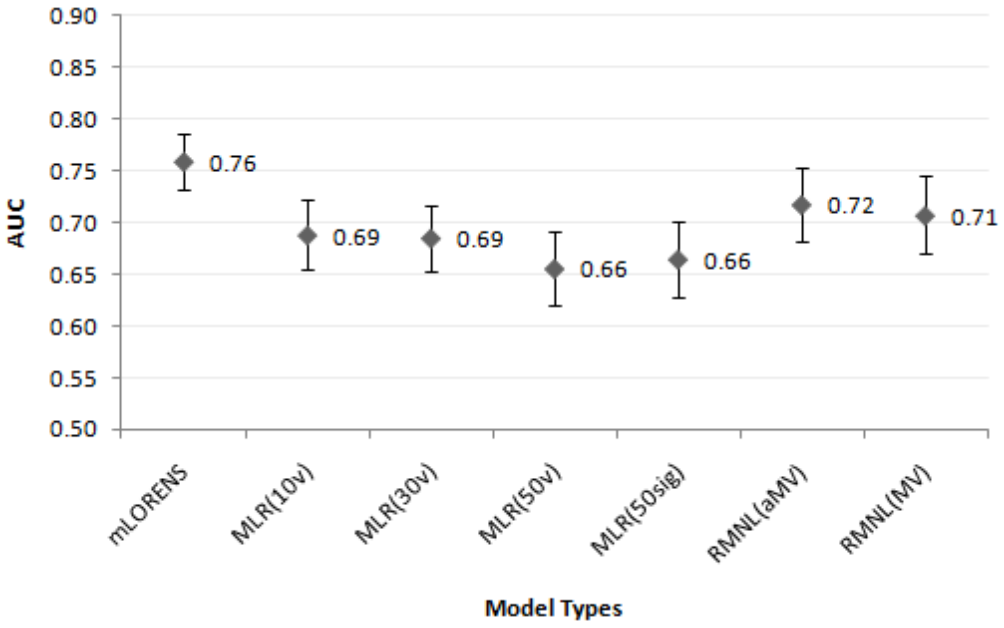


Figure 4.9: AUCs for Simulation Experiment 1: Correlated predictors with standard deviation 2



The improvement of mLORENS over a single base classifier with variable selection was shown again through the nested binary model (NBM). As an alternative model to MLR, NBM was applied in this simulation study. The program for NBM was coded and implemented in mLORENS as a base classifier using R. This method was only tested on the simulation data sets with independent variables. Tables 4.13 and 4.14 and Figures 4.10 and 4.11 show the results from the data with standard deviation 1, and Tables 4.15 and 4.16 and Figures 4.12 and 4.13 show the results from the data with standard deviation 2.

Variable selection was performed through BW ratio for a single NBM. In the data with standard deviation 1, among single NBMs with different numbers of selected variables, the one with 30 variables showed the highest accuracy and mean AUC. The accuracy was 84% and it was similar to the accuracy of NBM with 10 selected variables (p-value is 0.228), but significantly higher than that with 50 significant variables (p-value is less than 0.0001). When NBM was applied to mLORENS as a base classifier, the accuracy increased to 87%. This was significantly higher than that of NBM with 30 selected variables. The mean AUC also increased to 0.90 for mLORENS with NBM. As for the balance of sensitivity and specificity in both of the data sets, a single NBM was better than mLORENS with NBM, even though the sensitivities for classes 1 and 3 of a single NBM were usually lower than those of mLORENS with NBM. The imbalance was prominent in class 2 for both methods. However, it appeared

to be more severe in mLORENS with NBM since the sensitivity of class 2 for mLORENS with NBM was significantly lower than that for a single NBM (with 30 selected variables for the data with standard deviation 1, with the 50 significant variables for the data with standard deviation 2) and the specificity was significantly higher.

For the data with standard deviation 2, a single NBM with the 50 significant variables showed the highest accuracy among NBMs. When the variable selection was performed to select 50 variables, only about 39 significant variables were selected and the accuracy was 62%. This was significantly lower than the accuracy (64%) of NBM with all 50 significant variables without variable selection. This result is different from that of MLR. The accuracy of mLORENS with NBM was 70% and it was significantly higher than that of a single NBM. The highest mean AUC among single NBMs was 0.73 and it increased to 0.78 when NBM was applied to mLORENS.

Table 4.13: Simulation Experiment 1 (NBM): Performances (SD in parentheses) of mLORENS implemented NBM and NBM. Independent predictors from normal distribution with standard deviation 1.

Model	♯var.[a]	♯sig. var[b]	♯part.[c]	ACC	AUC[d]		class 1	class 2	class 3
mLORENS (NBM)	all		34.3 (8.6)	.87 (.03)	.90 (.02)	SENS:	1.00 (.01)	.61 (.08)	1.00 (.01)
						SPEC:	.91 (.03)	1.00 (.01)	.90 (.03)
						AUC:	.95 (.02)	.81 (.04)	.95 (.02)
	all		10 (fixed)	.87 (.03)	.90 (.02)	SENS:	.98 (.02)	.65 (.09)	.98 (.02)
						SPEC:	.91 (.03)	.98 (.02)	.91 (.03)
						AUC:	.95 (.02)	.81 (.05)	.94 (.02)
	all		20 (fixed)	.87 (.03)	.91 (.02)	SENS:	.99 (.02)	.64 (.08)	.99 (.01)
						SPEC:	.91 (.03)	.99 (.01)	.91 (.03)
						AUC:	.95 (.02)	.82 (.04)	.95 (.02)
NBM with variable selection	10	10 (0)		.83 (.04)	.87 (.03)	SENS:	.90 (.07)	.72 (.08)	.87 (.07)
						SPEC:	.93 (.03)	.89 (.05)	.92 (.04)
						AUC:	.92 (.04)	.81 (.04)	.90 (.04)
	30	29.2 (.8)		.84 (.05)	.88 (.04)	SENS:	.89 (.07)	.72 (.09)	.91 (.07)
						SPEC:	.93 (.03)	.90 (.05)	.93 (.04)
						AUC:	.91 (.04)	.81 (.05)	.92 (.04)
	50	49.2 (.8)		.81 (.04)	.86 (.03)	SENS:	.85 (.07)	.71 (.09)	.85 (.07)
						SPEC:	.92 (.04)	.86 (.05)	.93 (.03)
						AUC:	.89 (.04)	.79 (.05)	.89 (.04)
	70	49.2 (.7)		.76 (.04)	.82 (.03)	SENS:	.80 (.08)	.67 (.09)	.80 (.07)
						SPEC:	.91 (.04)	.81 (.09)	.91 (.03)
						AUC:	.85 (.05)	.74 (.05)	.86 (.04)
NBM w/ signif. variables	50			.81 (.04)	.85 (.03)	SENS:	.85 (.08)	.71 (.09)	.86 (.07)
						SPEC:	.93 (.04)	.86 (.05)	.93 (.03)
						AUC:	.89 (.04)	.78 (.05)	.89 (.04)

[a] number of selected variables chosen in the training phase

[b] number of significant variables among the selected variables

[c] average number of mutually exclusive subsets of predictors in a partition, chosen in the training phase

[d] mean of the AUCs from the three classes

Table 4.14: McNemar's Test Results of Simulation Experiment 1 (NBM): Independent predictors with standard deviation 1

Models		p-value			
		Overall	Class 1	Class 2	Class 3
mLORENS(NBM) : NBM w/30var	ACC:	<0.0001	<0.0001	<0.0001	0.0036
	SENS:		<0.0001	<0.0001	<0.0001
	SPEC:		<0.0001	<0.0001	<0.0001
NBM w/30var : NBM w/10var	ACC:	0.2280	0.0005	0.3117	<0.0001
	SENS:		0.0258	0.7998	<0.0001
	SPEC:		0.0086	0.0811	0.0118
NBM w/30var : NBM w/sig.var	ACC:	<0.0001	0.0001	<0.0001	<0.0001
	SENS:		<0.0001	0.3160	<0.0001
	SPEC:		0.9722	<0.0001	0.1251

Figure 4.10: Accuracies for Simulation Experiment 1 (NBM): Independent predictors with standard deviation 1

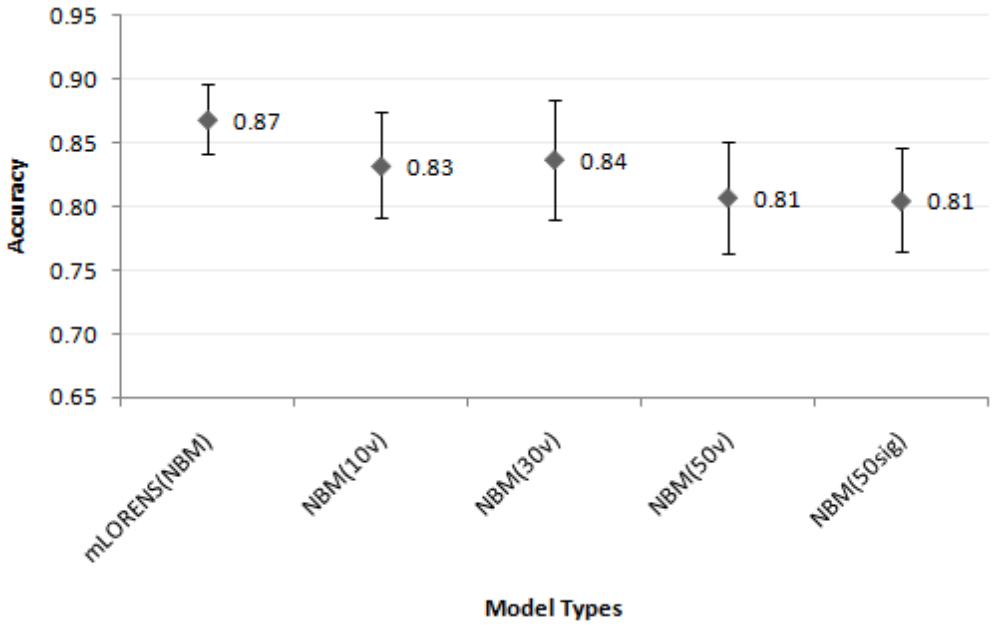


Figure 4.11: AUCs for Simulation Experiment 1 (NBM): Independent predictors with standard deviation 1

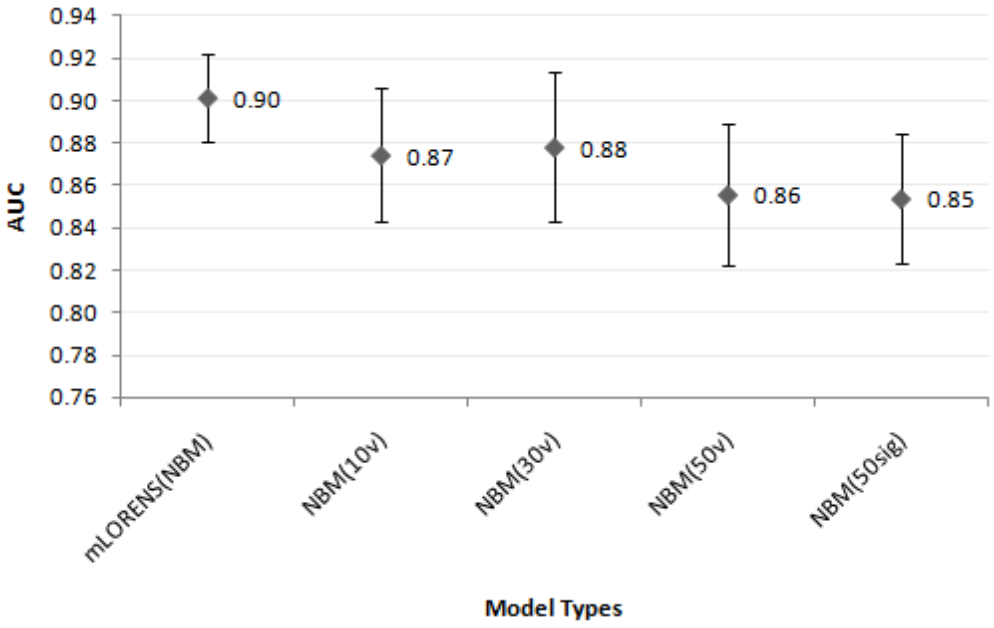


Table 4.15: Simulation Experiment 1 (NBM): Performances (SD in parentheses) of mLORENS implemented NBM and NBM. Independent predictors from normal distribution with standard deviation 2.

Model	♯var.[a]	♯sig. var[b]	♯part.[c]	ACC	AUC[d]		class 1	class 2	class 3
mLORENS (NBM)	all		29.2 (8.9)	.70 (.04)	.78 (.03)	SENS:	.87 (.06)	.38 (.08)	.86 (.06)
						SPEC:	.84 (.04)	.88 (.04)	.84 (.04)
						AUC:	.85 (.03)	.63 (.04)	.85 (.03)
	all		10 (fixed)	.67 (.04)	.75 (.03)	SENS:	.81 (.07)	.40 (.09)	.81 (.07)
						SPEC:	.84 (.04)	.83 (.05)	.84 (.05)
						AUC:	.83 (.04)	.62 (.04)	.82 (.04)
	all		20 (fixed)	.70 (.03)	.77 (.03)	SENS:	.86 (.07)	.38 (.08)	.85 (.06)
						SPEC:	.84 (.04)	.87 (.04)	.84 (.04)
						AUC:	.85 (.03)	.63 (.04)	.85 (.03)
NBM with variable selection	10	9.6 (.1)		.61 (.05)	.71 (.03)	SENS:	.72 (.08)	.40 (.09)	.71 (.09)
						SPEC:	.81 (.05)	.80 (.06)	.81 (.05)
						AUC:	.76 (.05)	.60 (.05)	.76 (.05)
	30	20.2 (1.9)		.62 (.05)	.71 (.04)	SENS:	.69 (.09)	.48 (.09)	.68 (.09)
						SPEC:	.84 (.05)	.75 (.06)	.84 (.05)
						AUC:	.76 (.05)	.61 (.05)	.76 (.05)
	50	38.9 (2.0)		.62 (.05)	.71 (.04)	SENS:	.70 (.09)	.47 (.10)	.68 (.09)
						SPEC:	.84 (.05)	.75 (.06)	.84 (.05)
						AUC:	.77 (.05)	.61 (.05)	.76 (.05)
	70	39.5 (2.0)		.57 (.05)	.68 (.04)	SENS:	.64 (.09)	.46 (.09)	.62 (.10)
						SPEC:	.82 (.05)	.71 (.07)	.83 (.05)
						AUC:	.73 (.05)	.59 (.05)	.72 (.05)
NBM w/ signif. variables	50			.64 (.05)	.73 (.04)	SENS:	.72 (.10)	.49 (.10)	.73 (.08)
						SPEC:	.85 (.05)	.76 (.06)	.85 (.04)
						AUC:	.79 (.05)	.62 (.05)	.79 (.04)

[a] number of selected variables chosen in the training phase

[b] number of significant variables among the selected variables

[c] average number of mutually exclusive subsets of predictors in a partition, chosen in the training phase

[d] mean of the AUCs from the three classes

Table 4.16: McNemar's Test Results of Simulation Experiment 1 (NBM): Independent predictors with standard deviation 2

Models		p-value			
		Overall	Class 1	Class 2	Class 3
mLORENS(NBM) : NBM w/sig.var	ACC:	<0.0001	<0.0001	<0.0001	<0.0001
	SENS:		<0.0001	<0.0001	<0.0001
	SPEC:		0.0097	<0.0001	0.0005
NBM w/sig.var : NBM w/30var	ACC:	<0.0001	<0.0001	0.0392	<0.0001
	SENS:		0.0005	0.6810	<0.0001
	SPEC:		0.0097	0.0211	0.0168
NBM w/sig.var : NBM w/50var	ACC:	<0.0001	<0.0001	0.0100	<0.0001
	SENS:		0.0134	0.1582	<0.0001
	SPEC:		0.0015	0.0320	0.0165

Figure 4.12: Accuracies for Simulation Experiment 1 (NBM): Independent predictors with standard deviation 2

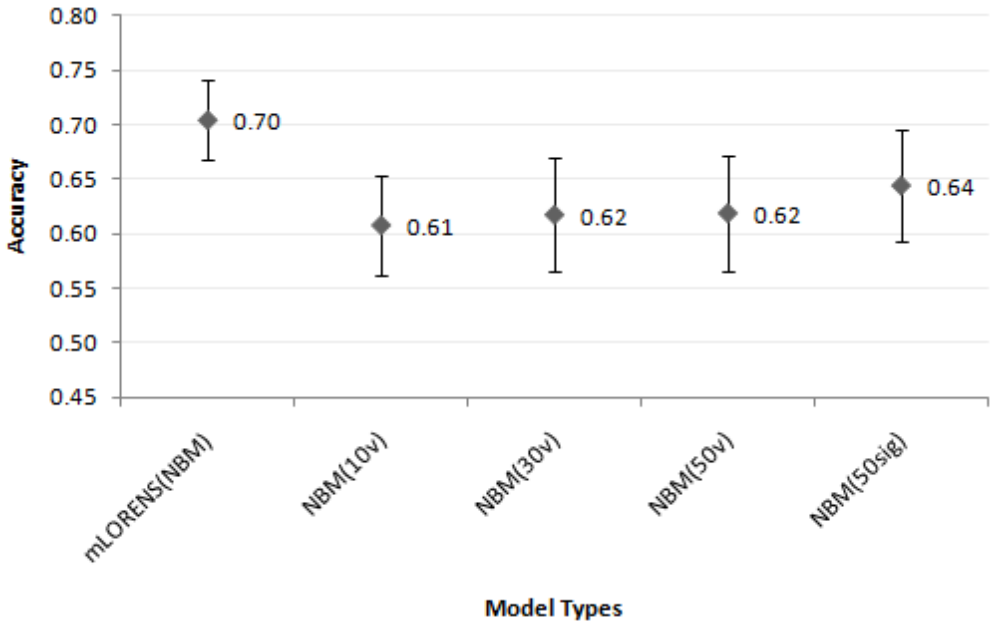
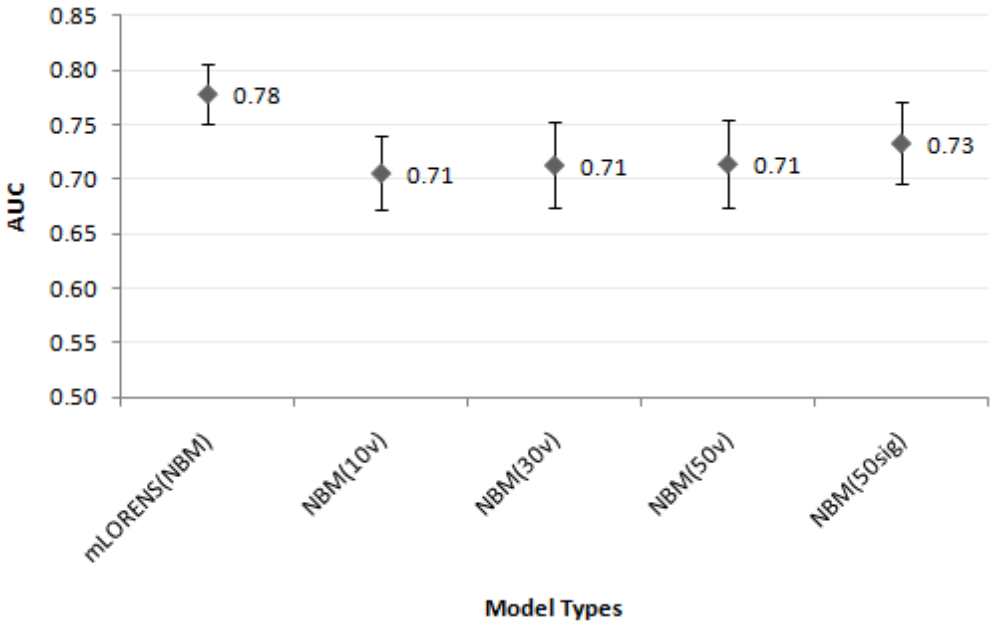


Figure 4.13: AUCs for Simulation Experiment 1 (NBM): Independent predictors with standard deviation 2



Tables 4.17 and 4.18 represent the results from McNemar's test to compare MLR and NBM. For the data with standard deviation 1, the highest accuracy among single MLRs was 0.83 and it was not significantly different from the highest accuracy (0.84) among single NBMs. The p-value for this comparison was 0.41. However, when these two models were applied to mLORENS as a base classifier, the accuracies became significantly different. The accuracy of mLORENS with MLR as a base classifier was 0.92 and it is significantly higher than that of mLORENS with NBM (0.87). For the data with standard deviation 2, mLORENS showed almost the same accuracy for both of the base classifiers (0.71 for mLORENS with MLR and 0.70 for mLORENS with NBM) even though NBM showed higher accuracy as a single classifier than MLR (0.61 for MLR and 0.64 for NBM). The same pattern was found for AUC. In conclusion, the improvement of mLORENS in terms of accuracy and AUC was more prominent when mLORENS adopted MLR as a base classifier. Figures 4.14 through 4.17 are provided to help understand the comparison between MLR and NBM.

Table 4.17: McNemar's Test Results of Simulation Experiment 1 (MLR and NBM): Independent predictors with standard deviation 1

Models		p-value			
		Overall	Class 1	Class 2	Class 3
mLORENS (MLR) : mLORENS (NBM)	ACC:	<0.0001	<0.0001	<0.0001	<0.0001
	SENS:		0.0005	<0.0001	0.3017
	SPEC:		<0.0001	0.0008	<0.0001
MLR w/10var : NBM w/30var	ACC:	0.4075	<0.0001	0.3935	<0.0001
	SENS:		0.1578	0.0008	<0.0001
	SPEC:		<0.0001	<0.0001	0.9710

Table 4.18: McNemar's Test Results of Simulation Experiment 1 (MLR and NBM): Independent predictors with standard deviation 2

Models		p-value			
		Overall	Class 1	Class 2	Class 3
mLORENS (MLR) : mLORENS (NBM)	ACC:	0.5729	0.7538	0.7024	0.2073
	SENS:		0.0156	0.0003	0.1306
	SPEC:		0.1817	0.0013	0.0060
MLR w/10var : NBM w/sig.var	ACC:	<0.0001	<0.0001	0.0468	<0.0001
	SENS:		0.0022	0.0179	<0.0001
	SPEC:		<0.0001	0.5483	<0.0001

Figure 4.14: Accuracies for Simulation Experiment 1 (MLR and NBM): Independent predictors with standard deviation 1

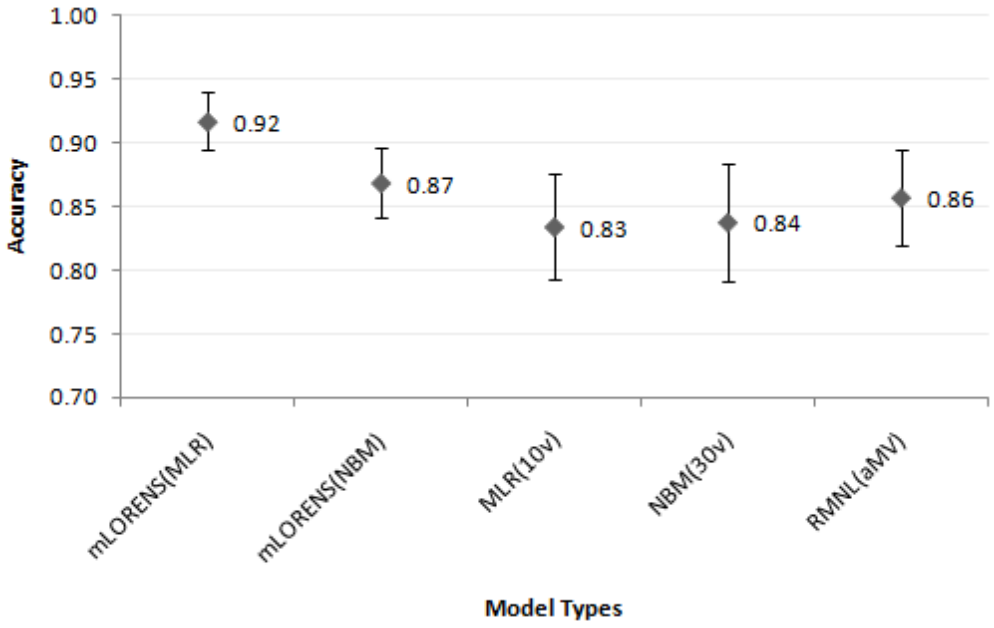


Figure 4.15: AUCs for Simulation Experiment 1 (MLR and NBM): Independent predictors with standard deviation 1

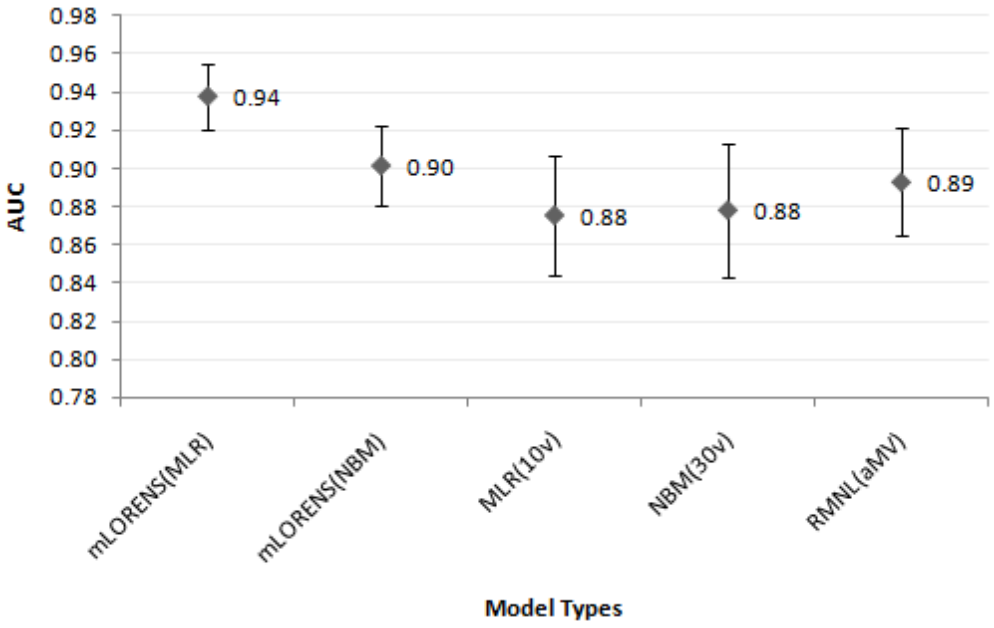


Figure 4.16: Accuracies for Simulation Experiment 1 (MLR and NBM): Independent predictors with standard deviation 2

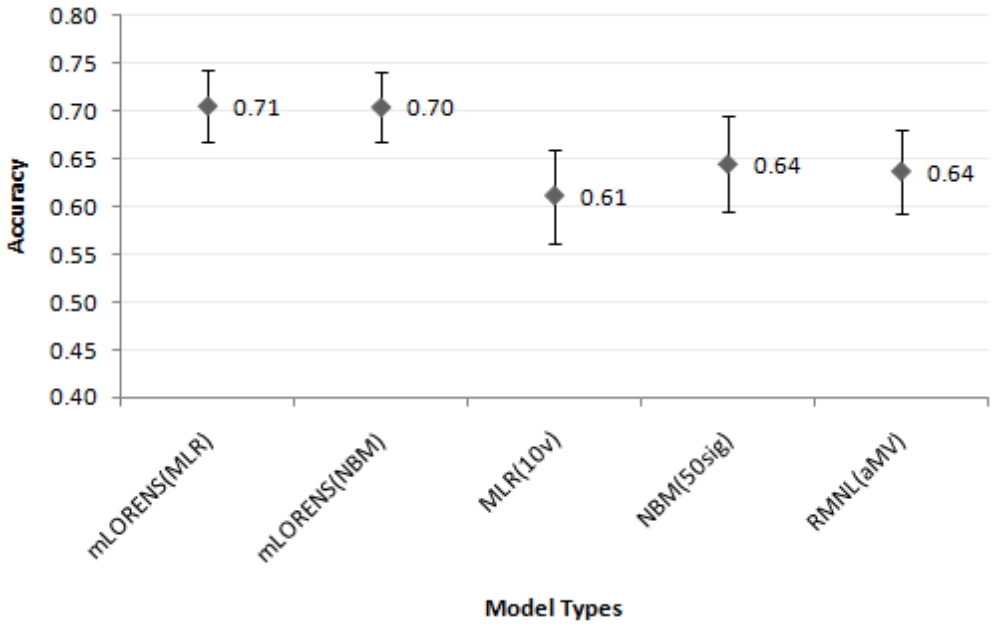
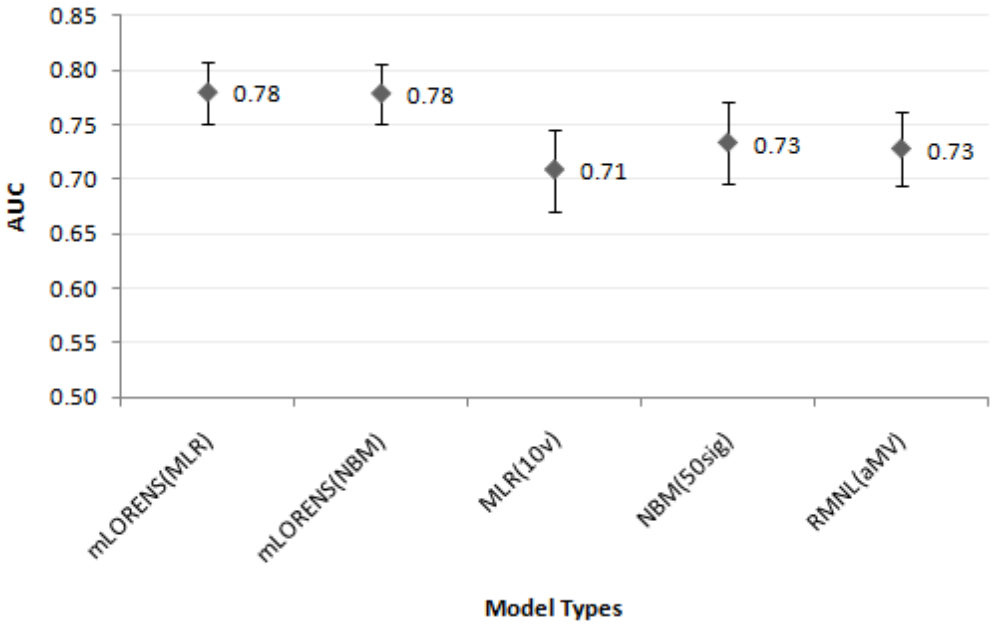


Figure 4.17: AUCs for Simulation Experiment 1 (MLR and NBM): Independent predictors with standard deviation 2



4.2 Simulation Experiment 2

In order to investigate the performance of methods with a low-dimensional data set in addition to high-dimensional data, one more simulation study was conducted. The data design was similar to that of the GIB data. Two data sets with 20 predictors and 120 subjects were generated: one for training and the other for testing. There were three classes, and the class sizes were given as 80, 29, and 11. Among the 20 predictors, 12 predictors were discrete, and the remaining 8 predictors were continuous. For each class, the means and variances were taken from the actual GIB data for each variable. For each variable and each class, data were generated from normal distributions with means close to the ones from the actual GIB data and two times the actual standard deviations. Higher standard deviation than that of the actual GIB data was used in this simulation study to increase uncertainty. To generate discrete variables, the initial values obtained from the normal distribution were changed to area under the normal curve to make the values between 0 and 1, and then these values were assigned into a class according to the proportions from the real GIB data using the percentile of the initial values. For correlated variables, correlation between two variables was generated by a random number from Uniform$(0, 0.3)$. Values higher than 0.3 caused problems in obtaining the covariance matrix and generating the multivariate normal distribution. One hundred learning sets and corresponding test sets were generated for evaluation. Like the GIB data, the search procedure for partition size was not carried out. The variables were partitioned into 2 or 3 subsets.

The results of this example are provided in Tables 4.19 through 4.22. The accuracy and mean AUC of this simulation experiment became lower than those of the actual GIB data, since the simulation data were constructed by doubling the standard deviations of the actual ones. In mLORENS, when the variables were partitioned into 2 subsets, the performance was better than when the partition size was 3. In RMNL, the number of selected variables was searched in learning phase among the numbers from 5 to 20 as the same way as in the actual GIB data. The average number of selected variables was about 12 or 13 and this result is similar to that of the actual GIB data. There was no difference between the two combining approaches for accuracy and mean AUC in RMNL.

mLORENS showed better performance than a single MLR in accuracy for both independent and correlated simulation designs. The p-values were less than 0.0001. The accuracy of RMNL was also significantly higher than that of a single MLR. The accuracies of mLORENS and RMNL were similar for the data with independent variables, but the accuracy of mLORENS was higher than that of RMNL for the corrected data. In actural GIB data, there was no significant difference between the accuracies of mLORENS and RMNL.

Regarding AUC, there was no significant difference among the three models for independent variables, but for correlated variables AUC of MLR was the highest. mLORENS and RMNL showed high sensitivity in the largest class (Upper), but for the smallest class (Mid), MLR showed higher sensitivity than

those of the other two models even though all of the numbers were less than 0.5. Figures 4.18 and 4.20 show accuracies, and Figures 4.19 and 4.21 show AUCs for this example.

Table 4.19: Simulation Experiment 2: Performances (SD in parentheses) of mLORENS, MLR, and RMNL. Independent predictors.

Model	♯Var.	♯Part.[a]	ACC	AUC[b]		Class 1	Class 2	Class 3
mLORENS	all	2	.85	.78	SENS:	.97 (.02)	.76 (.10)	.24 (.14)
		(fixed)	(.03)	(.04)	SPEC:	.82 (.08)	.91 (.03)	.98 (.01)
					AUC:	.90 (.04)	.83 (.05)	.61 (.07)
	all	3	.80	.68	SENS:	.99 (.01)	.57 (.12)	.05 (.08)
		(fixed)	(.03)	(.04)	SPEC:	.55 (.10)	.94 (.02)	1.0 (.01)
					AUC:	.77 (.05)	.76 (.06)	.52 (.04)
MLR	all		.78	.77	SENS:	.87 (.05)	.64 (.11)	.45 (.18)
			(.05)	(.05)	SPEC:	.88 (.07)	.89 (.04)	.88 (.04)
					AUC:	.88 (.04)	.77 (.05)	.67 (.09)
RMNL	12.4[c]		.85	.80	SENS:	.96 (.03)	.76 (.09)	.33 (.15)
w/aMV	(2.2)		(.03)	(.04)	SPEC:	.89 (.09)	.91 (.03)	.96 (.03)
					AUC:	.92 (.04)	.83 (.05)	.64 (.07)
RMNL	12.8[c]		.85	.79	SENS:	.96 (.03)	.74 (.10)	.34 (.16)
w/MV	(2.1)		(.03)	(.04)	SPEC:	.87 (.09)	.91 (.02)	.96 (.03)
					AUC:	.91 (.04)	.82 (.05)	.65 (.07)

[a] pre-determined number of subsets in a partition

[b] mean of the AUCs from the three classes

[c] average number of selected variables in a bootstrap sample to fit a multinomial logit model, chosen among the numbers from 5 to 20 in the learning phase

Table 4.20: McNemar's Test Results of Simulation Experiment 2: Independent predictors

Models		p-value			
		Overall	Class 1	Class 2	Class 3
mLORENS w/2pt : MLR	ACC:	<0.0001	<0.0001	<0.0001	<0.0001
	SENS:		<0.0001	<0.0001	<0.0001
	SPEC:		<0.0001	<0.0001	<0.0001
mLORENS w/2pt : RMNL w/aMV	ACC:	0.1518	<0.0001	0.0451	<0.0001
	SENS:		<0.0001	1.0000	<0.0001
	SPEC:		<0.0001	0.0024	<0.0001
MLR : RMNL w/aMV	ACC:	<0.0001	<0.0001	<0.0001	<0.0001
	SENS:		<0.0001	<0.0001	<0.0001
	SPEC:		0.6207	<0.0001	<0.0001

Figure 4.18: Accuracies for Simulation Experiment 2: Independent predictors

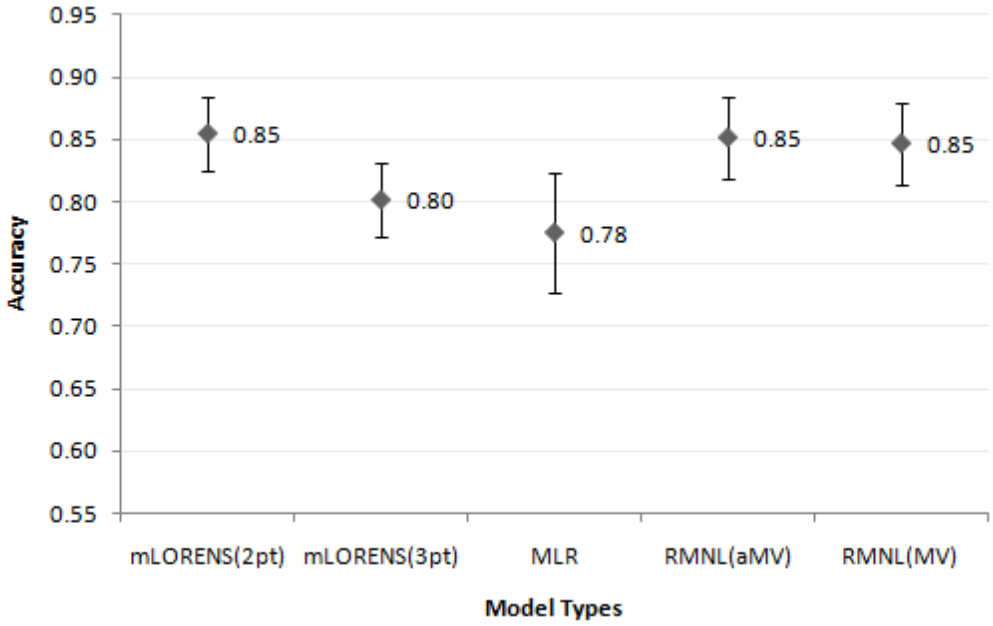


Figure 4.19: AUCs for Simulation Experiment 2: Independent predictors

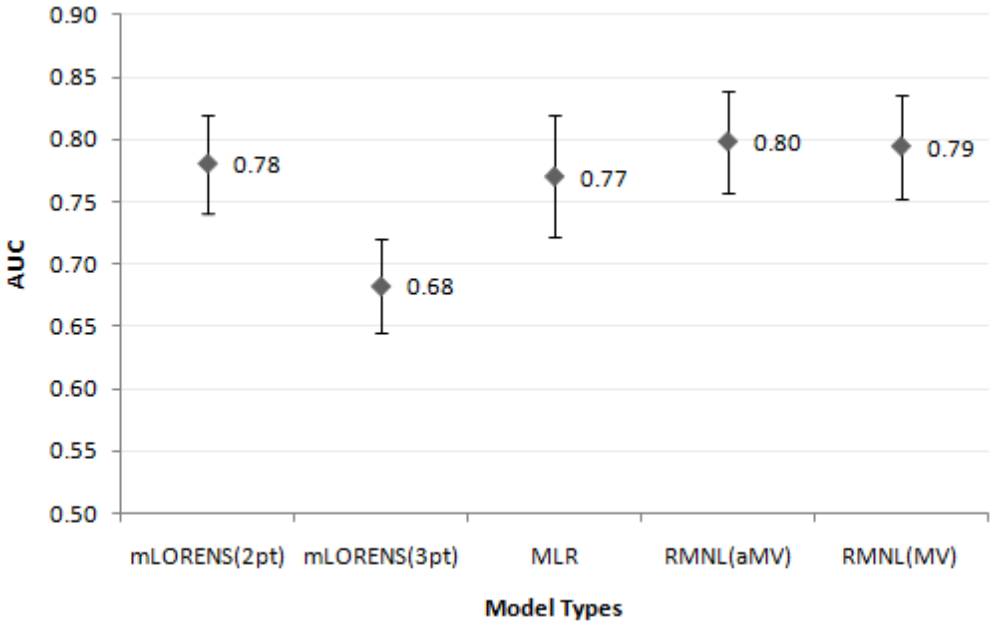


Table 4.21: Simulation Experiment 2: Performances (SD in parentheses) of mLORENS, MLR, and RMNL. Correlated predictors.

Model	♯Var.	♯Part.[a]	ACC	AUC[b]		Class 1	Class 2	Class 3
mLORENS	all	2	.70	.59	SENS:	.95 (.03)	.21 (.09)	.20 (.16)
		(fixed)	(.03)	(.04)	SPEC:	.25 (.08)	.96 (.03)	.99 (.01)
					AUC:	.60 (.04)	.58 (.04)	.59 (.08)
	all	3	.68	.53	SENS:	.99 (.01)	.08 (.06)	.05 (.07)
		(fixed)	(.02)	(.02)	SPEC:	.08 (.05)	.99 (.01)	1.0 (.00)
					AUC:	.54 (.02)	.53 (.03)	.52 (.04)
MLR	all		.64	.64	SENS:	.76 (.06)	.39 (.11)	.42 (.19)
			(.04)	(.05)	SPEC:	.53 (.09)	.85 (.05)	.90 (.05)
					AUC:	.65 (.05)	.62 (.06)	.66 (.09)
RMNL	11.6[c]		.70	.61	SENS:	.92 (.06)	.24 (.13)	.25 (.20)
w/aMV	(3.0)		(.04)	(.06)	SPEC:	.32 (.16)	.94 (.04)	.97 (.03)
					AUC:	.62 (.06)	.59 (.06)	.61 (.09)
RMNL	12.3[c]		.69	.61	SENS:	.91 (.06)	.26 (.13)	.26 (.19)
w/MV	(3.0)		(.03)	(.06)	SPEC:	.33 (.16)	.93 (.04)	.97 (.03)
					AUC:	.62 (.06)	.60 (.05)	.61 (.09)

[a] pre-determined number of subsets in a partition

[b] mean of the AUCs from the three classes

[c] average number of selected variables in a bootstrap sample to fit a multinomial logit model, chosen among the numbers from 5 to 20 in the learning phase

Table 4.22: McNemar's Test Results of Simulation Experiment 2: Correlated predictors

Models		p-value			
		Overall	Class 1	Class 2	Class 3
mLORENS w/2pt : MLR	ACC:	<0.0001	<0.0001	<0.0001	<0.0001
	SENS:		<0.0001	<0.0001	<0.0001
	SPEC:		<0.0001	<0.0001	<0.0001
mLORENS w/2pt : RMNL w/aMV	ACC:	<0.0001	0.9182	<0.0001	<0.0001
	SENS:		<0.0001	<0.0001	<0.0001
	SPEC:		<0.0001	<0.0001	<0.0001
MLR : RMNL w/aMV	ACC:	<0.0001	<0.0001	<0.0001	<0.0001
	SENS:		<0.0001	<0.0001	<0.0001
	SPEC:		<0.0001	<0.0001	<0.0001

Figure 4.20: Accuracies for Simulation Experiment 2: Correlated predictors

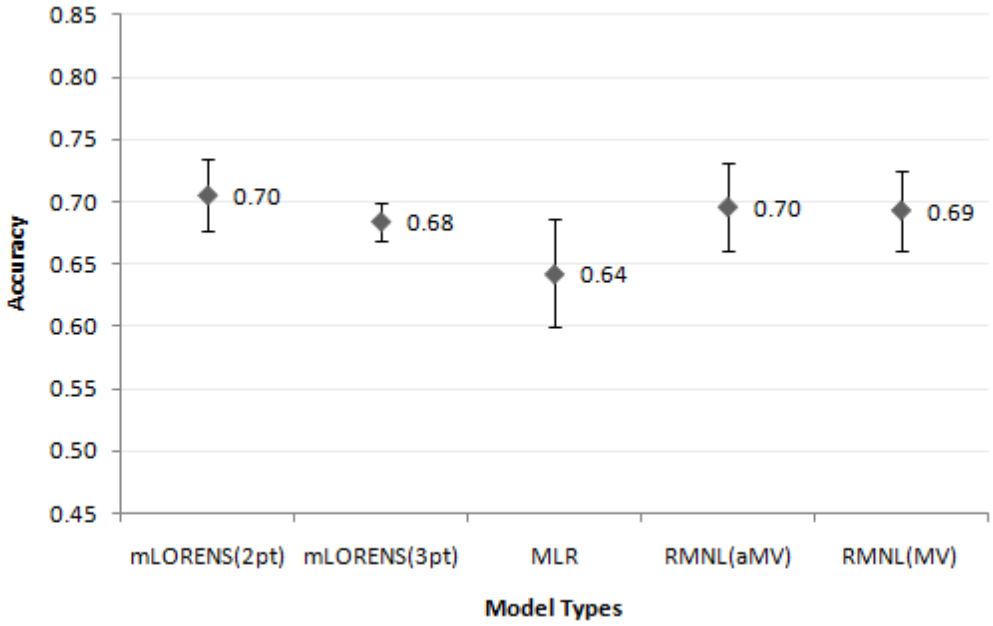
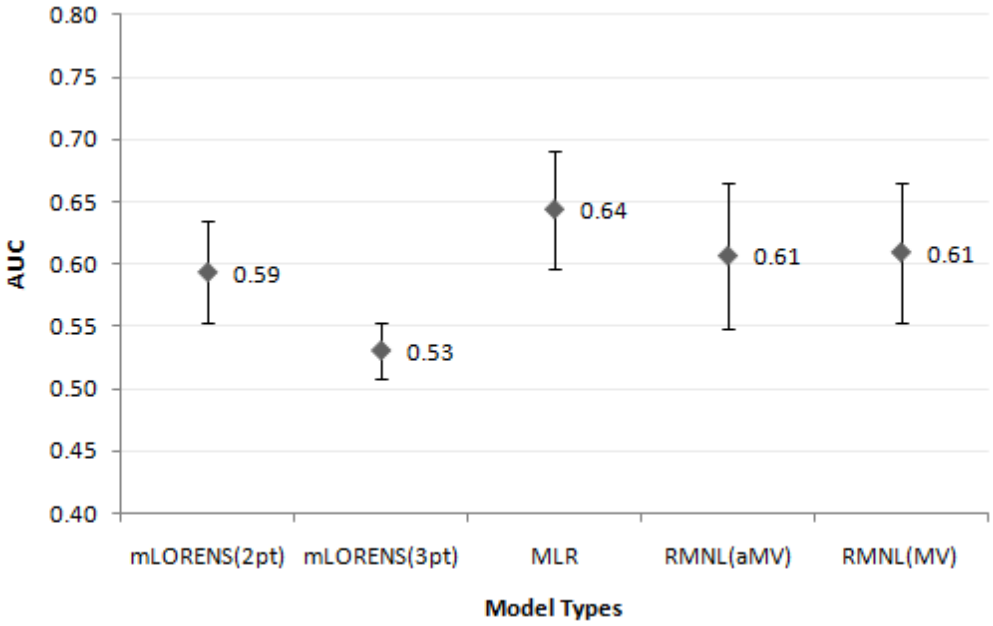


Figure 4.21: AUCs for Simulation Experiment 2: Correlated predictors



Chapter 5

Conclusion and Discussion

Multinomial logistic regression model was adopted into CERP to handle multi-class problems. mLORENS showed clear improvement in prediction accuracy over a single MLR for all data sets used in this study. For high-dimensional data sets, mLORENS showed higher AUC than MLR in general. In the simulation study, mLORENS showed better performance than RMNL as well as a single MLR in terms of prediction accuracy and mean AUC, while mLORENS and RMNL showed similar performance in all measures for the real data sets.

LORENS was designed for high-dimensional data. Due to the random partitioning in LORENS, MLR can be fit in each partition without variable selection. Hence, mLORENS can handle a huge feature space to which a single MLR cannot be applied without variable selection. Our examples show that even for low-dimensional data that a single MLR can be used without variable selection, mLORENS showed a higher prediction accuracy. In these

examples, the feature space was divided into a fixed number of subspaces, and in each subspace MLR was fit to smaller numbers of predictors. It means that combining several MLR models with small numbers of predictors may give better accuracy than a single fit with all predictors at once. The randomness of partitioning in mLORENS is one of the reasons for improving the prediction accuracy, since the correlation among base classifiers could be reduced. Since randomly selected mutually exclusive subsets of predictors are assigned to each of the randomly partitioned subspaces, redundancy of the data is reduced. By integrating these advantages, the accuracy of mLORENS is greatly improved.

While we had improvement in prediction accuracy, mLORENS encountered difficulties in improving sensitivity and specificity. For balanced data sets mLORENS performed very well in sensitivity and specificity. The rates were generally better than those of MLR or RMNL. However, for unbalanced data sets, the sensitivity of a small class tended to be low. By predicting to a large class, the overall accuracy increased, while the sensitivity becomes poor. In LORENS, this problem was solved through adjusting the decision threshold[23]. Decision threshold approach is expected to improve the balance between sensitivity and specificity when class sizes are unbalanced. In mLORENS it is not trivial to find an optimal decision threshold due to the high dimensionality of classes.

Chapter 6

Future Study

- Comparison with other ensemble methods: We plan to compare our method with other widely used classification methods including RF, SVM, Boosting, K-means clustering, and Linear Discriminant Analysis with high-dimensional data sets.

- Simulation study in unbalanced data and different data design: Only balanced data sets were studied in the simulation experiment 1 in this study. We are planning to apply the methods to unbalanced data sets. It is expected to help understand the trend of the balance of sensitivity and specificity specifically. In addition to this, different data designs would be constructed and applied. Results from simulation experiment 1 revealed that the performance of classification methods depends on the design of data. Several types of data design would help find the characteristic of mLORENS and other classification methods.

- Imbalance of Sensitivity and Specificity: Decision threshold technique might be complicated and computer-intensive to be applied to multiclass cases. However, if a method to determine the optimal threshold is developed, then we anticipate that the method improves the balance between sensitivity and specificity when class sizes are severely unbalanced.

- Generalized AUC for multi-dimensional cases: In this study, we used an estimate for AUC in 3 dimensional space, while there is an extension of ROC curve for multi-dimensional cases. If receiver operating characteristic surfaces instead of curve are considered, then the generalized AUC for multi-dimensional cases could be calculated. This is expected to provide more precise evaluation of the performance of models.

- Other Models for Multiclass Problems: The baseline logit models were used in this study. It is known as a robust model for multiclass problems. But to obtain a broad view of multicategory problems, we plan to study other models and apply these into CERP. It may be compared with mLORENS as well as other methods.

References

[1] A. Agresti. Multicategory logit models. In *An Introduction to Categorical Data Analysis*. Wiley-Interscience, 1996.

[2] A. Agresti. Logit models for multinomial responses. In *Categorical Data Analysis*. Wiley-IEEE, 2003.

[3] H. Ahn, H. Moon, M. J. Fazzari, N. Lim, J. J. Chen, and R. L. Kodell. Classification by ensembles from random partitions of high-dimensional data. *Comput. Stat. Data Anal.*, 51(12):6166–6179, 2007.

[4] E. Alpaydin. Introduction to machine learning. In *Introduction to Machine Learning*. MIT Press, 2004.

[5] L. Breiman. Bagging predictors. *Machine Learning*, 24:123–140, 1996.

[6] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[7] A. Chu, H. Ahn, B. Halwan, B. Kalmin, E. Artifon, A. Barkun, M. Lagoudakis, and A. Kumar. A decision support system to facilitate management of patients with acute gastrointestinal bleeding. *Artif. Intell. Med.*, 42(3):247–259, 2008.

[8] C. Cortes and V. Vapnik. Support vector networks. *Machine Learning*, 20:273–297, 1995.

[9] J. J. D. Applied multivariate data analysis. volume 1: Regression and experimental design. In *Applied multivariate data analysis. Volume 1: Regression and experimental design.*, page Chapter 8.3.6. Springer, 1992.

[10] E. R. DeLong, D. M. DeLong, and D. L. Clarke-Pearson. Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach. *Biometrics*, 44:837–845, 1988.

[11] S. Dudoit, J. Fridlyand, and T. P. Speed. Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American Statistical Association*, 97:77–87, 2002.

[12] C. Ferri, J. Hernandez-orallo, and M. Salido. Volume under the roc surface for multi-class problems. exact computation and evaluation of approximations. In *Proc. of 14th European Conference on Machine Learning*, pages 108–120, 2003.

[13] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(7):179–188, 1936.

[14] E. Fix and J. Hodges. Discriminatory analysis, nonparametric discrimination: Consistency properties. *Technical Report 4, USAF School of Aviation Medicine*, pages 261–279, 1951.

[15] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting, 1995.

[16] J. M. Greally. Short interspersed transposable elements (sines) are excluded from imprinted regions in the human genome. *Proceedings National Academy of Science*, 99(1):327–332, 2002.

[17] D. J. Hand and R. J. Till. A simple generalisation of the area under the curve for mutiple class classification problems. *Machine Learning*, 45:171–186, 2001.

[18] L. K. Hansen and P. Salamon. Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(10):993–1001, 1990.

[19] M. Kearns. Thoughts on hypothesis boosting. Unpublished manuscript, 1988.

[20] M. H. Kollef, J. D. O'Brien, G. R. Zuckerman, and W. Shannon. Bleed: A classification tool to predict outcomes in patients with acute upper and lower gastrointestinal hemorrhage. *Critical Care Medicine*, 25:1125–1132, 1997.

[21] L. Kuncheva and C. Whitaker. Measures of diversity in classifier ensembles. *Machine Learning*, 51:181–207, 2003.

[22] L. I. Kuncheva, C. J. Whitaker, and R. P. W. Duin. Limits on the majority vote accuracy in classifier fusion. *Pattern Analysis and Applications*, 6(1):22–31, April 2003.

[23] N. Lim, H. Ahn, H. Moon, and J. J. Chen. Classification of high-dimensional data with ensemble of logistic regression models. *Journal of Biopharmaceutical Statistics*, 20(1):160–171, 2010.

[24] A. Prinzie and D. V. den Poela. Random forests for multiclass classification: Random multinomial logit. *Expert Systems with Applications*, 34(3):1721–1732, 2008.

[25] T. A. Rockall, R. F. A. Logan, H. B. Devlin, and T. C. Northfield. Incidence of and mortality from acute upper gastrointestinal haemorrhage in the united kingdom. *British Medical Journal*, 311:222–226, 1995.

[26] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.

[27] S. Russell and P. Norvig. Artificial intelligence: A modern approach, 2003.

[28] R. E. Schapire. The strength of weak learnability. *Machine Learning*, 5:197–227, 1990.

[29] R. E. Schapire. The strength of weak learnability. *Machine Learning*, 5(2):197–227, 1990.

[30] J. H. F. Trevor Hastie, Robert Tibshirani. The elements of statistical learning: Data mining, inference, and prediction. In *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2001.

[31] R. W. and W. J. Genomic imprinting: Parental influence on the genome. *Nature reviews. Genetics*, 2(1):21–32, 2000.

[32] R. B. West, D. S. A. Nuyten, S. Subramanian, T. O. Nielsen, C. L. Corless, B. P. Rubin, K. Montgomery, S. Zhu, R. Patel, and T. Hernandez-Boussard. Determination of stromal signatures in breast carcinoma. *PLOS BIOLOGY*, 3(6):1101–1110, 2005.