New Probabilistic Techniques for Classification Problems and an Application

Yi Tan

B.S. Statistics, Wuhan University, 2011

M.S. Business Analytics, University of Cincinnati 2013

Submitted to the graduate degree program in Business and the Graduate Faculty of the University of Kansas in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Committee members

Prakash P. Shenoy, Chairperson

Ben Sherwood

Steve Hillmer

Mazhar Arikan

Zongwu Cai

Date defended: May 08, 2020

The Dissertation Committee for Yi Tan certifies
that this is the approved version of the following dissertation :

New Probabilistic Techniques for Classification Problems and an Application

Prakash P. Shenoy, Chairperson

Date approved: _____

Abstract

The main focus of this dissertation is to develop new machine learning and statistical methodologies for classification problems, with a real–life application in healthcare. The dissertation has three chapters. In the first chapter, we examine the construction of hybrid logistic regression–naïve Bayes model, a restricted Bayesian network classifier that combines two probabilistic models in a graphical way, with the aim of combining the strengths of both models. We follow the strategy of balancing the tradeoff between model bias and variance with the objective of minimizing the sum of these two errors. Specifically, we use training set size as a proxy for model variance and conditional dependence among features as a proxy for model bias. Experimental results show that, the resulting hybrid logistic regression–naïve Bayes model is a competitive alternative to a variety of state-of-the-art classifiers.

In the second chapter, we focus on a regularization method, which is a technique of adding information to the learning algorithm to improve the estimation of the model. Most of the existing regularization methods (e.g., lasso) rely on sparsity assumption, which reduces a model's variance by shrinking its coefficients towards zero. One limitation of lasso is that, in practice, sparsity assumption is often violated. Shrinking the coefficients of influential predictors towards zero introduces bias, and make the regression estimates suboptimal. As a consequence, lasso may not perform well when the training set size is relatively large as compared to the number of parameters to be estimated. We argue that for such a situation, shrinking the coefficients towards a low-variance data driven estimate could be a better strategy. For classification purposes, we propose a naïve Bayes regularized logistic regression, which shrinks its coefficients towards naïve Bayes estimates, a well-known low variance estimator, instead of zero. This method is driven by the fact that naïve Bayes and logistic regression converge toward identical classifiers if the naïve Bayes' conditional independence assumptions hold. Simulation and experimental results suggest

that this method is highly competitive with a variety of state-of-the-art classifiers.

In the third chapter, we are collaborating with the U.S. Veterans Affairs' (VA) Eastern Kansas Health Care System, to help them construct a clinical model that can assist doctors in predicting and diagnosing the post-traumatic stress disorder (PTSD). This study is motivated by the need to provide more efficient service process of VA hospitals and reduce veterans' waiting time. Specifically, we propose a sparsity-enforcing l_1 penalized Bayesian network-based model by addressing three clinical challenges presented in veteran PTSD prediction problem: 1. probabilistic classification, 2. large amount of missing data, and 3. high dimensional search space. The proposed model provides better prediction in veterans' likelihood of suffering from PTSD as compared with a variety of state-of-art probabilistic classifiers. In addition, our model identifies eight variables which provide the most directly predictive power.

Acknowledgements

Firstly, I would like to express my deepest gratitude to my advisor, Prof. Prakash Shenoy, for his unreserved support and mentoring during my 6 years of PhD study at KU. Prof. Shenoy is an authority in the area of Decision Sciences and Analytics that I gained valuable knowledge and received exceptional training from him through taking his doctoral seminar and collaborating with him on several research projects. He provides more than sufficient time for my PhD supervision, and always put me first when I need his help. Meanwhile, Prof. Shenoy gives me a good amount of independence in our research and be willing to listen to my ideas. I could not have imagined having a better advisor and mentor for my Ph.D study. Besides, I would like to thank Prof. Ben Sherwood for his insightful comments and considerate guidance on my dissertation. He is such a talented researcher that it is my great honor to work with him. My sincere thanks also goes to Prof. Steve Hillmer, Prof. Shaobo Li, Prof. Mazhar Arikan, Prof. Wei Chen, and Prof. Suman Mallik for the great help on my coursework, research projects, and job seeking. Finally, I would like to thank my family for all their love and encouragement. I could not have done it without them.

Contents

List of Figures

List of Tables

Chapter 1

A Bias-Variance Based Heuristic for Constructing a Hybrid Logistic Regression-Naïve Bayes Model for Classification

Abstract

Discriminative classifiers tend to have lower asymptotic classification errors, while generative classifiers can be more accurate when the training set size is small. In this paper, we examine the construction of hybrid models from categorical data, where we use logistic regression (LR) as a discriminative component, and naïve Bayes (NB) as a generative component. We adopt a bias-variance tradeoff based strategy, with the objective of minimizing the sum of these two errors. Specifically, the proposed heuristic consists of functions of training sample size and conditional dependence among features. These functions serve as proxies for model variance and model bias. We implement our method on 25 different classification datasets, and find that the hybrid model does better than pure LR and pure NB. Our proposed method is competitive with random forest. Although the hybrid model fails to beat LASSO in predictive performance, as suggested by the experimental results, the difference appears to be insignificant when the number of features is small. Also, the hybrid model requires less training time than LASSO, which makes it more attractive when the training time is a big concern.

1.1 Introduction

For classification problems, people are often faced with a choice between a generative and a discriminative classifier. Generative classifiers learn the joint probability distribution $P(F_1, \ldots, F_n, C)$ of the features $F_1, \ldots, F_n$, and the class C, make their predictions by using Bayes rule to compute $P(C \mid F_1, \ldots, F_n)$, and then predict a label with the highest posterior probability. In contrast, discriminative classifiers directly learn the conditional probability $P(C \mid F_1, \ldots, F_n)$, without assuming anything about the feature distribution, $P(F_1, \ldots, F_n)$. When training data are large, discriminative classifiers often achieve better prediction performance than generative classifiers, and hence are widely preferred. However, generative classifiers often have better performance when the size of training data is small [39]. Also, generative classifiers are more tolerant of missing values than discriminative classifiers.

To take advantage of both worlds, this paper investigates the construction of hybrid models from categorical data where we use logistic regression (LR) as a discriminative component, and naïve Bayes (NB) as a generative component, for datasets in the general domain. Both LR and NB belong to the family of probabilistic classifiers, and form a well known discriminative-generative pair [49]. Because LR and NB models have few parameters, they scale well to high dimensions, and can be trained very efficiently. It has been shown that LR can be modeled as a Bayesian network [47]. The hybrid LR-NB model, first proposed by Kang and Tian [33], is recognized as a restricted class of Bayesian network classifier that combines LR and NB in a graphical way. The task we are concerned with is *learning Bayesian network structures*, i.e., deciding to which part of the hybrid model a given feature should be assigned. Kang and Tian [33] use a greedy method based on in-sample classification accuracy. We argue that this method may yield a local optimal solution. Kang and Tian provide only one-round cross validation result for each dataset. We conjecture that this resulting hybrid model may not perform well in terms of average out-of-sample classification accuracy (with a suitably large number of trials). Also, their method is time intensive.

In this paper, we propose a more efficient model construction heuristic that balances the trade-

off between model bias and model variance. Specifically, LR produces the lowest prediction error among all linear classifiers by achieving the lowest bias if there are sufficient training data. However, this is not the case when training set size is limited. LR estimates may overfit the data, which makes the prediction less accurate due to high variance. On the other hand, NB overcomes the overfitting issue by making a strong assumption of conditional independence, and thus learns each parameter using the entire training sample. This makes NB work surprisingly well for small datasets. However, as the conditional independence assumptions rarely hold in practice, NB estimates are often suboptimal due to the introduction of bias in comparison to the situation where the conditional independence assumptions are satisfied. Our heuristic consists of functions of training sample size and conditional dependence among features. These functions serve as proxies for model variance and model bias. For each given feature, we estimate the bias and the variance introduced if the feature is assigned to the LR or to the NB part of the hybrid model, respectively. By minimizing the sum of bias and variance errors, we assign the feature to the corresponding part. If all features are assigned to the LR (NB) part, the resulting model is pure LR (NB), otherwise the resulting model is strictly hybrid. Our heuristic can be regarded as a selection mechanism that helps make the choice between pure LR, pure NB and strictly hybrid models.

We conduct experiments on 25 different machine learning datasets from UCI Machine Learning Repository. We select these datasets such that we have a diversity of sample sizes, number of features, and number of classes. We compare the 0-1 loss and root mean square error (RMSE) of hybrid model with pure LR, pure NB, random forest (RF), and LASSO, which are widely recognized as state-of-the-art classifiers, using paired t-tests with 0.05 significance level. Experimental results show that the hybrid model constructed using our heuristic achieves a more accurate classification performance than both pure LR and pure NB models. Specifically, the Win/Draw/Loss (W/D/L) of the hybrid model versus pure LR (pure NB) in terms of 0-1 loss and RMSE are 6/19/0 (18/2/5) and 10/14/1 (18/1/6), respectively. Also, the hybrid model is competitive with RF. It has a higher 0-1 loss (W/D/L=5/7/13), but lower RMSE (W/D/L=14/2/9). LASSO is difficult to beat, in terms of both 0-1 loss (5/10/10) and RMSE (1/14/10), in general. However, experimental results

suggest that the difference appears to be insignificant when the number of features is small. For example, the W/D/L for 0-1 loss is 3/7/4, and for RMSE is 0/12/2 on datasets with fewer than 10 features. If we have a large number of features, estimating the parameters of a pure LR model by maximizing the conditional log-likelihood can be computationally intensive. We also compare the training time, showing that the hybrid model is more efficient than pure LR by assigning some of the features to the NB part. Also, the training time for hybrid models is much lower on an average than the corresponding RF and LASSO models.

An outline of the remainder of the chapter is as follows. In Section 1.2, we describe related work on hybrid discriminative/generative classifiers, and state the contributions of our paper. Section 1.3 sketches and compares the LR and the NB models. In Section 1.4, we describe the hybrid model. Section 1.5 describes our method for constructing a hybrid model. Section 1.6 shows the empirical results from our experiments using 25 datasets from the UCI Machine Learning Repository. Finally, in Section 1.7, we summarize and conclude.

1.2 Related Literature

Our paper is related to the literature on the comparison of discriminative and generative classifiers, which has a long history. Efron [10] presents theoretical and simulation studies showing that linear discriminant analysis (LDA), a generative classifier, is asymptotically (between one third and one half) more efficient than LR if the model is correctly specified. On the other hand, Ng and Jordan [39] do an empirical and theoretical study of LR and NB models for classification. They find that there are two distinct regimes of prediction performance with respect to training set size. Particularly, an LR model has a lower asymptotic error compared to NB, but an NB model approaches its asymptotic error much faster than an LR model. In other words, for large training datasets, LR classifiers have higher accuracies, whereas for small training datasets, NB classifiers may have higher accuracies than LR. In our construction of a hybrid model, we make use of results discussed by Ng and Jordan [39] for penalizing the model complexity. Xue and Titterington [57] conduct empirical and simulation studies, as a complement to Ng and Jordan's work. However,

they find no convincing evidence to support Ng and Jordan's claim.

In the last two decades, several researchers have been exploring hybrid models that combine discriminative and generative models into one model [2, 21, 33, 45, 49, 56]. These are discussed in the following paragraphs.

Rubinstein and Hastie [49] are among the earliest to suggest combining discriminative and generative models. They suggest that features that satisfy the assumption of a generative model be retained in the generative part, with the remaining moved to the discriminative part. They compare linear discriminant analysis (LDA), a generative model, with LR, a discriminative model, for three different simulated datasets, and discover that LDA does better than LR when the class densities are Gaussian, and vice-versa. They also compare NB, a generative model, with generalized additive model (GAM), a discriminative model, for a simulated dataset with log-spline density that satisfies the assumptions of the GAM model. Asymptotically, the GAM model achieves a lower error rate than the NB model. However, when the training set is a small subset (25 observations) of the entire dataset, NB model has a lower average error than GAM. While they propose combining the two approaches, they do not describe any experimental results.

Raina *et al*. [45] investigate a hybrid model with LR as the discriminative component, and NB as the generative component, in the context of text classification problems. They run experiments using pairs of newsgroups from a dataset of USENET new postings, where the documents have two disjoint regions—a subject region and a message body region. An NB model treats the two regions in exactly the same way due to the strong conditional independence assumptions of an NB model. A hybrid model treats the two regions differently using different weight parameters for each. As the subject region has fewer words than the message body region, the words in the subject region are weighted higher than the words in the message body region. Depending on how the weight parameters are estimated from a dataset, the hybrid model reduces to an LR model. Experimental results show that hybrid models can provide lower test error than either pure LR or pure NB. Fujino *et al*. [21] investigate hybrid discriminative-generative classifiers similar to [45] for text classification having multiple components (such as titles, hyperlinks, anchor text, images, etc.).

They use a generative classifier for each component that are then combined using weights learnt using the maximum entropy principle. They do an empirical evaluation on four text-classification datasets, and find that hybrid classifiers outperform pure NB and pure LR models.

In an attempt to benefit from the advantages of both generative and discriminative approaches, Bishop and Lasserre [2] propose a heuristic that interpolate between these two extremes by taking a convex combination of the generative and discriminative log-likelihood functions. They apply their approach to object recognition in static images. Each image has two sets of features—observable features, and latent patch labels—in addition to class. They compare the performance of hybrid models with different combination weights and find that the best performance is obtained with a blend of generative and discriminative extremes.

The studies described above all focus on one specific domain. Our paper differs from these prior works in that we examine the construction of the hybrid model that is applicable in general for any domain.

Zaidi *et al*. [59, 60] discuss a weighted variant of NB with the goal of alleviating the feature conditional independence assumption of NB, or using the maximum likelihood parametrization of NB to pre-condition the discriminative search of LR. By exploiting the direct equivalence between a weighted NB and LR, they introduce a hybrid discriminative-generative estimation approach, i.e., minimization of either the negative conditional log-likelihood, or the mean squared error objective functions. As a result, the weighted NB learns models that are exactly equivalent to LR, but computationally much more efficient. Experimental results suggest that the resulting weighted NB is a competitive alternative to state-of-the-art classifiers, such as random forest, LR, and A1DE (Average One-Dependence Estimators). Following the same intuition, Zaidi *et al.* [61] introduce a hybrid discriminative-generative approach, called *accelerated logistic regression* (ALR), to train LR with high-order features. They show that ALR significantly improves the efficiency of LR. Moreover, by incorporating higher order features to reduce the bias, ALR predicts well as compared to state-of-the-art classifiers including random forest and average n-dependence estimators, especially on large datasets. All of these methods are driven by the fact that weighted NB has an

equivalent functional form compared to LR. The model approach is hybrid in the sense of estimating a generative model discriminatively. In contrast, our focus is on a restricted Bayesian network classifier, which combines two probabilistic models in a graphical way.

One of the closest to our work is that by Xue and Titterington [56]. They study hybrid discriminative-generative classifiers where the discriminative component is LR, and the generative component is Fisher's linear discriminant analysis (LDA). They test all features for univariate normality, and those that fail the test are assigned to the LR portion of the hybrid model. Xue and Titterington test their algorithm for 6 datasets that have only numeric features. They find that for smaller sizes of the training set, the hybrid model does better than the pure LR and pure LDA models. Our focus is on classification tasks for categorical data. Instead of using LDA, which is applicable only for numeric features, we use NB, which is well suited for categorical features, as the generative component. Although NB can also be used for numeric features, it involves making distributional assumptions for the conditional distribution of a feature given the class, and this makes a formal comparison messy.

Our study contributes to the literature by providing a new method on construction of such a hybrid discriminative-generative classifier where the discriminative component is LR, and the generative component is NB, first introduced by Kang and Tian [33]. Kang and Tian learn a hybrid model by starting with an empty discriminative component, and then greedily add one feature at a time (which results in the maximum in-sample classification accuracy gain) to the discriminative component until the in-sample classification accuracy does not improve. They test their algorithm for 20 different datasets, which are pre-processed so that there are no missing values, and all features are categorical. They measure classification accuracy using either 10-fold cross-validation (for small datasets) or 3-fold cross-validation (for large datasets). This is done just once, so they get a point estimate of the classification accuracy. The average point estimate of the classification errors for all 20 datasets is lowest for the hybrid LR-NB model. However, the average classification accuracy (with a suitably large number of trials) is not captured, in terms of which the resulting hybrid model may not outperform the benchmark models because it yields a local optimal solution.

Also this method is computationally intensive.

The strategy of a recent work by Tan *et al.* [53] for constructing such a hybrid model is based on reducing the conditional dependence of features in the NB part of the hybrid model. They estimate the normalized conditional mutual information (*norMI*) given the class variable for all pairs of features, and find a pair of features with highest *norMI* (using 0.05 as a cutoff point). Then for each of these two features, the authors compare their second highest *norMI* and move the one with higher value to the LR part. This strategy deals with the model bias, but ignores the impact of model variance. Consequently, the resulting hybrid model outperforms the pure NB model, but does worse than the pure LR model in the pairwise comparison.

In this paper, we construct the hybrid model by balancing the tradeoff between model bias and model variance. The proposed heuristic consists of functions of training sample size and conditional dependence among features. These functions serve as proxies for model variance and model bias,and our objective is to minimize the sum of these two errors.

The contributions of this work are as follows:

- We investigate the strengths of hybrid LR-NB models by reviewing the literature on comparison between LR and NB.
- We propose an efficient heuristic for constructing a hybrid LR-NB model. Experimental results show that the heuristic is effective in improving the classification performance of hybrid model compared to pure LR and pure NB. Also, the resulting hybrid model is comparable in classification accuracy to random forest. Although it fails to beat LASSO in general, experimental results suggest that the difference in predictive performance between hybrid model and LASSO appears to be insignificant when the number of features is small. Also, the training time for hybrid models is less on an average than corresponding LR, RF, and LASSO models.
- We propose a novel idea for balancing the bias-variance tradeoff. Compared to traditional bias-variance techniques, which add a regularizer to a loss function, our method proposes proxies for both bias and variance, and then minimizes the sum of these two errors.

1.3 LR versus NB

1.3.1 Logistic Regression

In this subsection, we discuss LR as a classification method for categorical data. Suppose we seek to assign a class label $c \in \Omega_C$ of the class variable C to instances described by a set of n categorical features $\mathbf{F} = (F_1, \ldots F_n)$ defined on the probabilistic space χ. For simplicity of exposition, we assume that $F_1, \ldots, F_n$ are all Boolean. If feature F_j is not Boolean, i.e., has k_j states with $k_j > 2$, we can represent F_j by $k_j - 1$ Boolean features $F_{j2}, \ldots, F_{jk_j}$ where $F_{jt} = 1$ if variable F_j takes state t and $F_{jt} = 0$ otherwise, $t = 2, \ldots, k_j$.

The LR model is a discriminative classifier that directly learns the conditional probability $P(C \mid \mathbf{F})$ by assuming that the log odds for a class c_j is a linear function of the features:

$$\ln\left(\frac{P(C = c_j \mid \mathbf{f})}{1 - P(C = c_j \mid \mathbf{f})}\right) = \beta_{0\,j} + \sum_{i=1}^{n} \beta_{i\,j} f_i, \tag{1.1}$$

where $\mathbf{f} = (f_1, \ldots f_n)$.

We can derive the parametric form for the distribution $P(C \mid \mathbf{F})$ by rewriting Eq. (1.1) as:

$$P(C = c_j|\mathbf{f}) = \frac{\exp(\beta_{\mathbf{0j}} + \sum_{\mathbf{i=1}}^{\mathbf{n}} \beta_{\mathbf{ij}} \mathbf{f_i})}{\sum_{\mathbf{k=1}}^{\mathbf{c}} \exp(\beta_{\mathbf{0k}} + \sum_{\mathbf{i=1}}^{\mathbf{n}} \beta_{\mathbf{ik}} \mathbf{f_i})}. \tag{1.2}$$

Notice that for a dataset that has a class variable with q classes, we have $(q-1) \cdot (n+1)$ parameters. The small number of parameters is one reason for the simplicity and robustness of the LR classifier. Using Eq. (1.2), we can compute the probability distribution of classes in C. The predicted class is the one with the highest probability.

LR parameters are usually estimated by maximizing the conditional likelihood, i.e., choosing parameters $\mathbf{B}$ that satisfy $\mathbf{B} \leftarrow \arg\max_{\beta} \prod P(C = c_j \mid \mathbf{f}, \beta)$, where $\beta = (\beta_0, \ldots, \beta_n)$. As there is no closed form solution with respect to $\mathbf{B}$, one common approach is to use gradient-based methods. Ng and Jordan [39] show that the prediction performance of LR converges to the best performance of all linear classifiers as the training sample size goes to infinity. However, with a small training

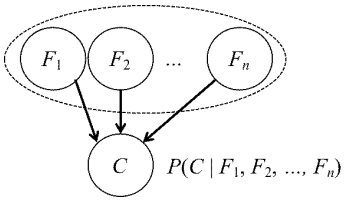

Figure 1.1: An LR Model as a Bayesian Network

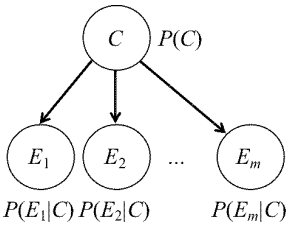

Figure 1.2: An NB Model as a Bayesian Network

sample size, LR estimates may overfit the data, which makes the prediction less accurate. Also, the gradient-based method can be computationally intensive especially for a large number of parameters of interest. For datasets with large number of features, any speed-up to the parameter learning process may be highly desired.

Rijmen [47] models an LR model as a Bayesian network, where Eq. (1.2) constitutes the conditional probability distribution for C given $\mathbf{F} = (F_1, \ldots F_n)$. LR assumes a parametric form for the distribution $P(C|\mathbf{F})$, and has its model structure as shown in Fig. (1.1). In this figure, the dotted oval around the features denotes that the Bayesian network structure of the feature variables is not represented, as it is irrelevant to C, assuming that we have observed values of all features.

1.3.2 Naïve Bayes

In this subsection, we discuss NB model as a classification method for categorical data. Suppose C is the class variable, whose value we wish to predict based on observation of a set of m categorical features $\mathbf{E} = (E_1, \ldots E_m)$.

The NB model is a generative classifier that learns the joint probability distribution $P(C, \mathbf{E})$ by making an assumption that features $\mathbf{E} = (E_1, \ldots E_m)$ are mutually conditionally independent given the class variable C. Fig. 1.2 is a Bayesian network depiction of an NB classifier. Using Bayes

rule, it can be shown that

$$odds(C = c_j \mid \mathbf{e}) = odds(C = c_j) \prod_{i=1}^{m} lr(e_i, C = c_j), \tag{1.3}$$

where $\mathbf{e} = (e_1, \ldots e_n)$ and $lr(e_i, C = c_j) = \frac{P(e_i|C=c_j)}{P(e_i|C\neq c_j)}$ is the likelihood ratio for $C = c_j$ based on the observed value $E_i = e_i$. In words, the posterior odds of $C = c_j$ is equal to prior odds of $C = c_j$ times the likelihood ratios of observed features for $C = c_j$. If a feature is not observed, we can regard its likelihood ratio as equal to 1.

We can also derive the parametric form of the posterior probability $P(C \mid \mathbf{E})$ from Eq. (1.3) as

$$P(C = c_j \mid \mathbf{e}) = \frac{P(C = c_j) \prod_{i=1}^{m} P(e_i \mid c_j)}{\sum_{k=1}^{c} P(C = c_k) \prod_{i=1}^{m} P(e_i \mid c_k)}. \tag{1.4}$$

As we are interested in the most probable value of C, we have the classification rule for NB as:

$$C \leftarrow \arg\max_{c_j} \frac{P(C = c_j) \prod_{i=1}^{m} P(e_i \mid c_j)}{\sum_{k=1}^{c} P(C = c_k) \prod_{i=1}^{m} P(e_i \mid c_k)}. \tag{1.5}$$

The conditional independence assumption reduces the complexity of the NB model (number of parameters), which makes it a robust model. Specifically, if the class variable C has q classes, and features $E_1, \ldots, E_m$ are all binary, the number of parameters to be estimated is $qm + q - 1$. Also, the conditional independence assumption helps overcome the overfitting issue by making NB estimate its parameters using the entire training sample. In spite of its simplicity and strong conditional independence assumption, NB performs surprisingly well, even against other more sophisticated classifiers, especially when the training set size is small.

However, conditional independence assumption rarely holds in practice, and as a consequence an NB model may not predict well. A great amount of literature addresses approaches to improving the prediction performance of NB by either relaxing the conditional independence assumptions between features given the class label [13, 18, 36, 42, 53, 63] or by weighting the features [17, 25, 60]. In the next section, we will describe a hybrid LR-NB classifier that relaxes the conditional

independence assumptions in NB by assigning mutually conditionally dependent features to the LR part of the hybrid model.

1.3.3 Comparison of LR and NB

In supervised learning, the prediction error for a given machine learning algorithm can always be broken down into three parts: bias, variance, and irreducible error. Irreducible error is associated with the inherent variability in the data, and there is nothing one can do about it. Thus, an effective learning algorithm should minimize the sum of bias and variance errors.

The *bias* error is the result of erroneous assumptions in the given learning algorithm. A high bias learner is usually less flexible, has a simpler functional form, and can be trained more efficiently (than low bias learners). On the other hand, the *variance* reflects the sensitivity of learning algorithm to the changes in the training dataset. A high variance learner is usually more flexible, has a more complex functional form, and is more likely to overfit the training data (than low variance learners). Reducing the bias usually results in increasing the variance, and vice-versa. As there is no way out of this inverse relationship between bias and variance, we need to balance the trade-off between these two errors. Such bias-variance tradeoff forms the conceptual basis for many regularization methods such as LASSO and ridge regression.

NB is a learning algorithm with lower variance, but higher bias, in comparison to LR. First, we can show that the conditional independence assumption of NB implies the same parameteric form of $P(C \mid \mathbf{E})$ as $P(C \mid \mathbf{F})$ in LR. To maximize the conditional likelihood of LR and NB using Eqs. (1.2) and (1.5) respectively, we get a direct equivalence between LR and NB as $e^{\beta_{0,j}} \propto P(C = c_j)$ and $e^{\beta_{i,j}} \propto P(e_i \mid c_j)$. This equivalence suggests that LR and NB have the same hypothesis space, and asymptotically converge toward identical classifiers assuming that the conditional independence assumptions of NB holds. As a result, they tend to produce the same classification error as the number of training examples approach infinity. However, when the conditional independence assumption does not hold, it introduces bias and consequently NB parameter estimates are suboptimal, i.e., the asymptotic classification error for LR is often lower than that of NB.

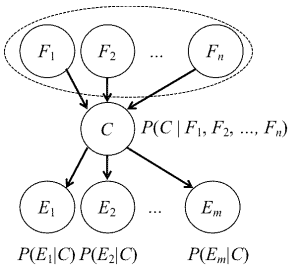

Figure 1.3: A Hybrid LR-NB Model as a Bayesian Network

On the other hand, the conditional independence assumption makes NB estimate the parameters over the entire sample, thus exhibits low variance by preventing it from overfitting. Ng and Jordan [39] also show that, the generative NB converges to its asymptotic error more quickly than the discriminative LR. Accordingly, the low-variance learner NB is expected to achieve lower error compared to LR, when the training set size is small.

1.4 A Hybrid LR-NB Model

In this section, we discuss a hybrid LR-NB model (hybrid, in short) as a classification method for categorical data. The graphical structure of a hybrid model represented as Bayesian network is shown in Fig. 1.3. Node C is the class variable, whose value we need to predict based on observation of two sets of features: $\mathbf{F} = (F_1, \ldots F_n)$, the parents of C in Fig. 1.3, called the LR part, and $\mathbf{E} = (E_1, \ldots E_m)$, the children of C, called the NB part. As in Section 1.3, we assume that $F_1, \ldots F_n$ are all Boolean.

The conditional independence assumptions of a hybrid model are as follows. First, the features in the LR part of the model are conditionally independent of the features in the NB part given the class variable C. Second, the features in the NB part of the model are mutually conditionally independent given the class variable C.

One implication of the first conditional independence assumption is that to learn the parameters of the conditional distribution of C given the features in the LR part, the features in the NB part are irrelevant for this task. Thus, one can use standard LR parameter estimation methods to learn these parameters. Similarly, to learn the parameters of the NB part of the hybrid model, the features in

the LR part are irrelevant for this task, and thus, we can use standard NB parameter estimation methods for learning these parameters.

Making inferences in a hybrid model is easy. Using variable elimination [62], after we eliminate the observed features in the LR part, the posterior distribution of the class variable C is given by the LR model:

$$\ln(odds(C = c_j|\mathbf{f})) = \beta_{0,j} + \sum_{i=1}^{n} \beta_{i,j} f_i,$$

where $\mathbf{f} = (f_1, \ldots, f_n)$, f_i is the observed state of feature F_i in the LR part, and $\beta_{i,j}$ are the parameters of the LR model.

This gives the posterior odds of $C = c_j$ given $\mathbf{f} = (f_1, \ldots, f_n)$ as:

$$odds(C = c_j|\mathbf{f}) = \exp(\beta_{0,j} + \sum_{i=1}^{n} \beta_{i,j} f_i). \tag{1.6}$$

After elimination of the features $\mathbf{F}$ in the LR part, what is left is an NB model such that the prior distribution of C (defined as the posterior distribution of C given $\mathbf{F} = \mathbf{f}$) is as given in Eq. (1.6). Thus, we can now compute the posterior distribution of C given $\mathbf{F} = \mathbf{f}$ and $\mathbf{E} = \mathbf{e}$ using the NB model as follows:

$$odds(C = c_j \mid \mathbf{e}, \mathbf{f}) = \exp\left(\beta_{0,j} + \sum_{i=1}^{n} \beta_{i,j} f_i\right) \prod_{k=1}^{m} lr(e_k, C = c_j), \tag{1.7}$$

where $\mathbf{e} = (e_1, \ldots, e_m)$, and $lr(e_k, C = c_j) = \frac{P(e_k|C=c_j)}{P(e_k|C\neq c_j)}$ is the likelihood ratio for $C = c_j$ based on the observed state $E_k = e_k$. Similar to NB model, if a feature in the NB part is not observed, we can regard its likelihood ratio as equal to 1.

Eq. (1.7) is used for making inferences from a hybrid classifier, which estimates the probability that the class variable C will take the value of $c_1, \ldots, c_q \in \Omega_c$ given the observed values of all features.

A hybrid model has an advantage of reducing the prediction error by relaxing the conditional independence assumptions of an NB model. From Fig. 1.3, the LR part of hybrid model does not

require a full model structure, and hence makes no conditional independence assumptions among its features. Given a set of features that are highly conditionally dependent on each other given the class variable, we can assign them to the LR part, as the conditional likelihood maximization algorithm for LR part can easily adjust its parameters to maximize the conditional likelihood of the data, hence reduce the bias of the hybrid model.

Apart from relaxing the conditional independence assumption of an NB model, hybrid model retains the simplicity of LR and NB models. Assuming all the features in both LR and NB parts of the model are binary-valued, then the number of parameters in the hybrid model is $(q-1)(n+1)+qm$, where n is the number of features in the LR part of the model, m is the number of features in the NB part of the model, and q is the number of classes of the class variable.

As a combination of LR model and NB model, the hybrid model is considered to be more flexible in terms of training set size. When the training sample is large, we tend to assign features to the LR part to reach the lowest bias among all linear classifiers. However, when the training sample is small, we may sacrifice some bias to achieve a lower variance by assigning features to the NB part. When the training sample is neither too large nor too small, we may assign some features to the LR part, and some features to the NB part by balancing the bias-variance tradeoff with the consideration of their conditional dependence with other features.

Finally, pure LR model can be computationally intensive, as its training time grows exponentially with the number of parameters to be estimated. The hybrid model is capable of reducing the LR's computational complexity by assigning some of the features to the NB part when we have a large number of features. Consequently, there are fewer parameters in the LR part to be estimated. Note that learning NB parameters does not require any optimization, thus offers a much faster training step as compared to LR. On the other hand, because it is just as easy to make inferences from a hybrid model as from pure LR and pure NB models, the hybrid model is computationally efficient at classification time.

Constructing a hybrid model is strictly more general than making the selection between a pure LR and a pure NB model. A hybrid model is identical to LR if all its features are assigned to the LR

part, and is identical to NB if all its features are assigned to the NB part. However, other feature assignments can result in classifiers that differ from both pure LR and pure NB. Later, we will show that by implementing our hybrid model construction heuristic, the hybrid model is capable of outperforming both pure LR and pure NB in many cases.

1.5 A Method for Constructing a Hybrid LR-NB Model

The main focus of this section is on construction of a hybrid model, with LR as the discriminative component, and NB as the generative component, that predicts well. As there are 2^{m+n} possible hybrid model structures, where $m+n$ is the total number of features in the hybrid model, searching the space of all possible hybrid models is computationally intractable for large values of $m+n$.

To the best of our knowledge, not much work has been done to address this issue. Kang and Tian [33] use a greedy method by starting with all features in the NB part, and they move one feature at a time to the LR part until the in-sample classification accuracy does not increase. One problem with this approach is that it may yield a local optimal solution. Also, this method is time intensive. Tan *et al*. [53] select the NB part based on the conditional independence relations of pairs of features. Their strategy tries to control for the model bias, however, it ignores the impact of model variance. As a result, their heuristic fails to outperform pure LR.

In this section, we propose a new heuristic to select the LR and NB parts for a hybrid model. Our heuristic only decides to which part a variable should be assigned. It does not serve as a feature selection process, i.e., we use all features in the datasets to train a hybrid model.

1.5.1 Feature Evaluation

In this section, we adopt a bias-variance tradeoff based strategy to decide whether we should assign a given feature to the LR part, or to the NB part, of a hybrid model. Traditional bias-variance techniques, such as LASSO and ridge regression, balance the tradeoff by adding a regularizer to a loss function. Our method differs from them in that we use a proxy for relative bias and

relative variance of assigning features to either the LR part or to the NB part. Thus, we construct a hybrid model by evaluating each feature to see if it favors high-bias, low-variance NB part or high-variance, low-bias LR part.

At the heart of our model construction method are two proxies for relative bias and relative variance of assigning features to either the LR or the NB part of a hybrid model, respectively, based on the following observations:

1. *Exponential Decrease*: As the size of the training set increases, the prediction error for both LR and NB model decreases exponentially.
2. *Convergence Rate*: NB's strong assumption of conditional independence allows NB to estimate the parameters using the entire training sample. This prevents NB from overfitting to the training data, and makes NB converge to its asymptotic error more quickly than LR [39]. As a result, NB tends to produce lower variance.
3. *Identical Classifier*: Assuming that the NB model's conditional independence assumption holds, LR and NB converge toward identical classifiers as the number of training instances tends to infinity.
4. *Conditional Independence*: A hybrid model assumes that features in the LR part are conditionally independent of the features in the NB part given the class variable, and that all features in the NB part are mutually conditionally independent given the class variable. In other words, any feature in the NB part is assumed to be conditionally independent with all other features in the hybrid model, and thus the violation of such conditional independence assumption will result in a bias.

First, we define $\bar{r}_i$ as the average conditional dependence given class variable C of feature F_i with all other features. We use it as a proxy for relative bias error induced by assigning feature F_i to the NB part, as compared with assigning the feature to the LR part, of a hybrid model: $B(F_i) = \bar{r}_i$. The prediction performance of LR converges to the best performance of all linear classifiers as the training sample size goes to infinity, i.e., LR produce the lowest bias among all

linear classifiers; while, as per observations *Identical Classifier* and *Conditional Independence*, if we assign a feature to the NB part of a hybrid model, it induces some additional bias by violating the conditional independence assumptions of the hybrid model. In this paper, we use normalized conditional mutual information as a measure of conditional dependence.

Next, we use the training set size as a proxy for relative variance error induced by assigning feature F_i to the LR part, as compared with assigning the feature to the NB part, of a hybrid model. Specifically, we adopt an exponential decay function relating the total number of instances in the training set, N, to the relative variance of the learning algorithm: $V(F_i) = e^{-\lambda N}$, following *Exponential Decrease*. It helps aviod the high influence of the sample size for large datasets. This function form is also consistent with the observation *Identical Classifier* as it converges to zero when N goes to infinite, and with the observation *Convergence Rate* as we set $\lambda > 0$.

Notice that both LR and NB models can handle numeric features. However, in the case of an NB model, we need to make distributional assumptions for the conditional distributions of numeric features given the class variable. Thus any violation of such distributional assumptions will result in additional bias. Similarly, regarding missing data, in the case of an LR model, we need to either ignore the corresponding instances, or impute the data, for example, with expectation-maximization algorithm or Markov chain Monte Carlo approaches. Both listwise deletion and data imputation result in additional variance. These add complexity to our heuristic. Therefore, we only focus on categorical data with no missing values in this paper. In addition, we do not apply any regularization in LR part of the hybrid model, because it increases the bias and reduces the variance of the LR part by shrinking its coefficients towards zero. Our approach controls the model variance by assigning features towards the NB part of the hybrid model.

1.5.2 Model Construction Algorithm

The previous paragraph describes the basic idea of the proposed heuristic in an intuitive manner. Following the bias-variance tradeoff strategy, we define $e(F_i)$ as a proxy for the relative model reducible error if F_i is assigned to the LR part, as compared with assigning the feature to the NB

part, of a hybrid model:

$$\begin{aligned} e(F_i) &= V(F_i) - B(F_i) \\ &= e^{-\lambda N} - \bar{r}_i \end{aligned} \tag{1.8}$$

Our heuristic criterion is described in Algorithm 1. First, we calculate the index of relative reducible errors, $e(F_i)$, for each of the features $F_1, \ldots, F_n$ using Eq. (1.8). Next, for any given feature F_i, if $e(F_i) \leq 0$, then F_i is assigned to the LR part. Otherwise, F_i is assigned to the NB part. We determine the value of the tuning parameter $\lambda \geq 0$ using cross-validation, which is described in Section 1.6.1.

Algorithm 1 Find structure of a hybrid model

Input: A set of labelled instances.
Output: A hybrid network structure with identified *LR-part* and *NB-part*.

1: Set *NB-part* $= \emptyset$ and *LR-part* $= \emptyset$.
2: For each $F_i \in \{F_1, \ldots, F_n\}$:
3: Calculate the index of relative errors $e(F_i)$ using Eq. (1.8).
4: If $e(F_i) \leq 0$, then *LR-part* $=$ *LR-part* $\cup \{F_i\}$
5: If $e(F_i) > 0$, then *NB-part* $=$ *NB-part* $\cup \{F_i\}$
6: end for;
7: end algorithm

1.6 Experimental Analysis

To evaluate the performance of hybrid models constructed using the heuristic introduced in Section 1.5, we conduct experiments on 25 different machine learning datasets from the UCI Machine Learning Repository. A summary of these datasets is given in Table 1.1. The datasets are selected such that we have a diversity of sample sizes, number of features, and binary/non-binary class variables.

In Section 1.6.1, we start with the description of our experimental setup. In Section 1.6.2, we compare the performance of hybrid models constructed using our heuristic with that of pure

Table 1.1: A Summary of 25 Bench-Mark Datasets

Dataset	*# Features*	*# Instances*	*# Classes*	*Dataset*	*# Features*	*# Instances*	*# Classes*
Abalone	8	4,177	3	Iris	4	150	3
Balance Scale	4	625	3	Liver Disorders	6	345	2
Banknote Authentication	4	1,372	2	Magic Gamma Telescope	10	19,020	2
Qualitative Bankruptcy	6	250	2	Mammographic Mass	5	961	2
Blogger	5	100	2	Mushroom	21	8,124	2
Blood Transfusion Service Center	4	748	2	New Thyroid	5	215	3
Car Evaluation	6	1,728	4	Pima Indians Diabetes	8	768	2
Connectionist Bench	60	208	2	Statlog Vehicle Silhouettes	18	846	4
Credit Approval	15	690	2	Vertebral Column	6	310	2
Hepatitis	19	155	2	Congressional Voting Records	16	435	2
Heart Disease (Hungarian)	13	294	5	Wilt	5	4,839	2
Hypothyroid	17	3,163	2	Wine	13	178	3
ILPD (Indian Liver Patient Dataset)	10	583	2				

LR and pure NB models. The training time of hybrid models is compared with that of pure LR and pure NB models in Section 1.6.3. The performance of our proposed method is compared with state-of-the-art classifiers, Random Forest (RF) and regularized LR (LASSO) in Sections 1.6.4 and 1.6.5 respectively.

1.6.1 Experimental Setup

The experiments are conducted using R. Numeric features in the original datasets are discretized using an entropy-based method (minimum description length (MDL) method), proposed by Fayyad and Irani [16]. We carry out the discretization procedure using a filter in WEKA. Besides, missing values of any features are imputed with the conditional probability given the response variable, i.e., $P(E_i \mid C)$.

First, we randomly divide each dataset into two parts, a training set with about 90% of the instances, and a test set with the remaining 10% of the instances. Using the training set, we implement the Algorithm 1 described in Section 1.5 to identify the hybrid model structure, i.e., to partition the feature set into an LR part and an NB part. Specifically, we select the value of tuning parameter λ using 10-fold cross-validation in the training set by minimizing the root mean square error (RMSE), which is calculated as follows:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{T}\sum_{s=1}^{S}(\hat{P}(y_{ts}) - y_{ts})^2}{T \cdot S}}$$

where y_{ts} is an indicator, which takes the value of 1 if the observed category of class variable y for instance t is s, and 0 otherwise. $\hat{P}(y_{ts})$ is the predicted probability of the class variable y for instance t to take the category s, T is the total number of instances, and S is the total number of categories for the class variable. Note that, the value of λ can also be chosen based on hybrid models' 0-1 loss, posterior likelihood, or AUC (for binary responses).

Next, we learn the parameters of the corresponding hybrid model with the entire training data. The estimation of parameters of the NB part of hybrid model and also those of pure NB model are conducted using the Laplace correction [40] to prevent the high influence of zero probabilities. Specifically, we assume the training set is large enough that adding one to each count would not make a significant difference in estimated probabilities.

Finally, we predict the class for instances in the test set. The prediction performance is measured using 0-1 loss, and RMSE.

We repeat the entire procedure (division of dataset, identification of model structure, estimation of model parameters using the training set, and computation of prediction accuracies using the test set) 1,000 times. For computational reasons, experiments for datasets with more than 8,000 instances, or 20 features (*Connectionist*, *Magic*, *Mashroon*) are conducted only 200 times. In the remainder of this section, we report Win/Draw/Loss (W/D/L) results when comparing the 0-1 loss and RMSE of two models. A two-tail paired t-test is used and we consider the results to be significant if its p-value is less than 0.05. The detailed results in terms of average structure of the hybrid model, prediction accuracies measured by 0-1 loss (in units of %) and RMSE with their standard errors, and training time are presented in the appendix.

Table 1.2: Win/Draw/Loss: Hybrid versus LR, Hybrid versus NB

W/D/L	*Hybrid vs. LR*	*Hybrid vs. NB*
0-1 Loss	6/19/0	18/2/5
RMSE	10/14/1	18/1/6

1.6.2 Hybrid versus LR and NB

We start by examining the average structure of the hybrid model found using our heuristic before making the comparison between hybrid model versus pure LR and pure NB. On an average for the 25 datasets, the hybrid model consists of 77.33% of the features in the LR part, with the remaining 22.67% in the NB part. Notice that if no feature is assigned to the LR (NB) part, then the hybrid model is exactly the same as pure NB (LR) model. The resulting hybrid model is identical to pure LR in 5 datasets, and possibly strictly hybrid in the remaining 20 datasets.

The Win/Draw/Loss (W/D/L) results of hybrid model against pure LR and pure NB are given in Table 1.2. It can be seen that hybrid model has significantly better 0-1 loss and RMSE than both pure LR and pure NB, indicating that our heuristic decribed in Section 1.5 is effective in improving the classification performance of hybrid model compared to pure LR and pure NB.

It is also worth noting that for those 20 datasets where the resulting model is possibly strictly hybrid, the hybrid model significantly outperforms both LR and NB in terms of 0-1 loss in 2 of them, and in terms of RMSE in 4 of them. This suggests that the hybrid model can be more powerful than making the selection between a pure LR and a pure NB model by providing the additional model structure fexibility.

1.6.3 Training Time Comparison

LR is computationally expensive at training time. Given a large number of features, estimating the parameters of an LR model by maximizing the conditional log-likelihood is a computationally intensive task. In this section, we examine the effectiveness of hybrid model on reducing the LR's training time by assigning some of the features to the NB part, and consequently achieving dimension reduction in the LR part.

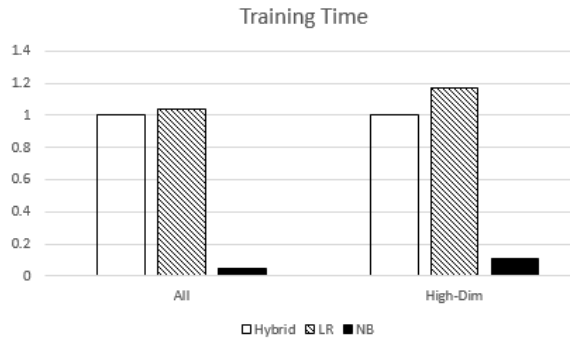

Figure 1.4: Average training time of hybrid model, LR and NB on all 25 or high-dimensional datasets. Results are normalized with respect to hybrid model.

Table 1.3: Win/Draw/Loss: Hybrid versus RF

W/D/L	0-1 Loss	RMSE
Hybrid vs. RF	5/7/13	14/2/9

A comparison of the training time of hybrid model versus LR and NB is given in Fig. 1.4. The results shown are consistent with our expectation that hybrid model requires less training time than pure LR. While over the entire 25 datasets, the hybrid model exhibits a slightly faster training speed than pure LR, the difference is more significant over the high-dimensional datasets (7 datasets with at least 15 features). Notice that we ignore the issue of classification time in this paper as hybrid model, LR and NB are all quite efficient in terms of classification time.

1.6.4 Hybrid versus Random Forest

Random forest (RF) [4] is recognized as a state-of-the-art classification technique. It basically consists of many classification trees, where each tree is trained using randomly selected (with replacement) instances from the training set. The RF makes a classification by choosing the most frequently selected category over all trees in the forest. We use 100 decision trees in this work.

The hybrid model is compared with RF in terms of W/D/L of 0-1 loss and RMSE in Table 1.3. It can be seen that hybrid model has higher 0-1 loss, but lower RMSE than RF, indicating that our heuristic results in a hybrid model that is competitive with the well known random forest. Also, Figure 1.5 suggests that RF requires much more training time than the hybrid model.

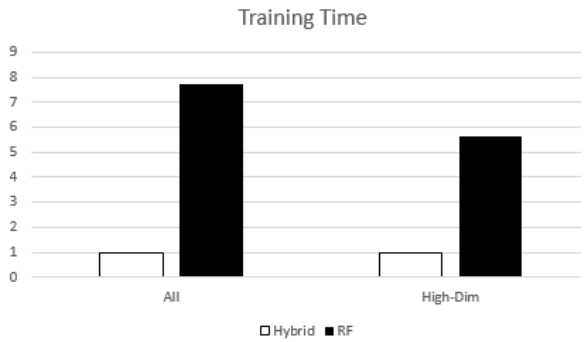

Figure 1.5: Average training time of hybrid model and RF on all 25 or high-dimensional datasets. Results are normalized with respect to hybrid model.

Table 1.4: Win/Draw/Loss: Hybrid versus LASSO

W/D/L	0-1 Loss	RMSE
Hybrid vs. LASSO	5/10/10	1/14/10

1.6.5 Hybrid versus LASSO

L_1 regularized logistic regression, also known as LASSO [55], is a bias-variance technique that performs both feature selection and shrinkage in order to improve the algorithm prediction power. It reduces the variance at the expense of increasing its bias, by shrinking the coefficients towards zero. By assuming sparsity, which reduces the model complexity and ensures the identifiability of the true underlying sparse model given limited sample size, LASSO is particularly useful in high-dimensional cases. We select the value of penalty parameter using 10-fold cross validation in the training set by minimizing the MSE in this work.

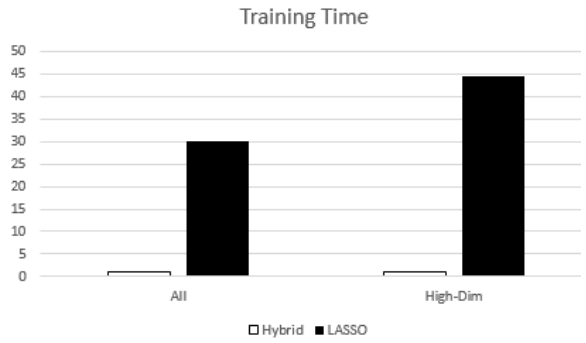

Figure 1.6: Average training time of hybrid model and LASSO on all 25 or high-dimensional datasets. Results are normalized with respect to hybrid model.

The hybrid model is compared with LASSO in terms of W/D/L of 0-1 loss and RMSE in Table 1.4. Our method performs worse than LASSO over the entire 25 datasets in terms of both 0-1 loss and RMSE. However, when the number of predictors is small, LASSO's sparsity assumption is more likely to be violated. As a result, the difference between hybrid model and LASSO appears to be insignificant. For example, the W/D/L for 0-1 loss is 3/7/4, and for RMSE is 0/12/2 on datasets with fewer than 10 features. Also, Figure 1.6 suggests that LASSO requires much more training time than the hybrid model.

1.7 Summary and Conclusions

In this paper, we describe a new hybrid LR-NB model construction method that follows the strategy of balancing the tradeoff between model bias and model variance, with the objective of minimizing the sum of these two errors. Our approach is primarily motivated by the intuition of taking advantage of the strengths of both LR and NB, and takes into account the training sample size and the conditional dependence among features. Experimental results are presented showing that the hybrid model constructed using our heuristic can generally outperform both pure LR and pure NB models. Hybrid model offers an improvement over pure LR in terms of training time by optimizing fewer parameters in the LR part. Also its prediction performance is comparable to random forest. Although the hybrid model fails to beat LASSO, the difference appears to be insignificant when the number of features is small. Also, the hybrid model requires much less training time, which makes it attractive when the training time is a big concern.

Our work adds to the literature that investigates the properties of LR and NB [38, 47], and makes the comparison between them [39, 57]. Particularly, our heuristic posits functions that try to link the training sample size and the conditional dependence among features, to the bias and the variance of assigning a given feature to the LR and the NB part of a hybrid model respectively. By balancing the tradeoff between bias and variance, we provide a selection mechanism that helps in making the choice between pure LR, pure NB, and strictly hybrid models.

Chapter 2

The Naïve Bayes Penalized Logistic Regression Model for Classification

Abstract

Attempting to reduce the variance of the estimator and to prevent overfitting, regularization techniques have attracted great interests from the statistics and machine learning community. Most of the existing regularized methods rely on the sparsity assumption, thus work particularly well in high-dimensional problems. However, the sparsity assumption may not be necessary when the number of predictors is relatively small compared to the number of training instances. In this paper, we argue that for such situations, shrinking the coefficients towards a low-variance data-driven estimate could be a better regularization strategy. For the classification problems, we propose a naïve Bayes regularized logistic regression (NBRLR), that shrinks the logistic regression coefficients toward the naïve Bayes estimate to provide a reduction in variance. Our approach is primarily motivated by the fact that naïve Bayes has an equivalent functional form compared to logistic regression given that naïve Bayes' conditional independence assumption holds. We also present the consistency result for the NBRLR estimator. Extensive simulation and empirical experimental results show that, NBRLR is a competitive alternative to a variety of state-of-the-art classifiers.

2.1 Introduction

Logistic regression (LR) is widely used in machine learning for classification problems. It is a discriminative classifier, which directly learns the conditional probability of the class variable given the predictors without assuming anything about the distribution of the predictors. As per Ng and Jordan [39], LR converges to the best linear classifier when the training sample size, n, goes to infinity by producing the smallest bias, and therefore is highly preferred amongst linear classifiers when the training sample size is large. However, when the training sample is limited, or there is a large number of parameters, p, to be estimated, regularization is required to avoid overfitting. Many regularized methods have been proposed to improve prediction error in regression frameworks, including lasso [55], SCAD [14], elastic net [64], and LARS [11]. These estimators rely largely on the sparsity assumption, i.e., only a small proportion of predictors are likely to be informative. Thus, they work particularly well in high-dimensional problems, i.e., p is relatively large compared to n.

A good regularization strategy should be shrinking the regression coefficients towards the values which are close to the truth. One limitation of these approaches is that, in practice, sparsity assumption is often violated. Especially, when p is relatively small compared to n, predictors are less likely to be irrelevant with the class variable, and thus tend to be influential. Shrinking the coefficients of influential predictors towards zero introduces bias, and causes the regression estimates to be suboptimal. As a result, traditional sparsity-enforced approaches may not perform well. Also in this setting, there also tends to be less multicollinearity among predictors. This limits the benefit of ridge regression [30], which is motivated by dealing with multicollinearity, not sparsity. We argue that, when p is relatively small compared to n, a better strategy of regularization is to shrink the coefficients towards a low–variance data–driven estimate.

It has been shown that naïve Bayes (NB), a probabilistic classifier with equivalent functional form compared to LR, tends to have lower variance than LR [59, 60]. NB is a generative classifier which learns the joint probability distribution of the predictors and the class variable. It infers the posterior probability of a class label given data by using Bayes rules with an assumption about

the distribution of the predictors and that the predictors are mutually conditionally independent of each other given the class variable. This assumption is mostly motivated by the need to learn parameters from high-dimensional data, and to overcome overfitting. Consequently, NB performs surprisingly well, even against other more sophisticated classifiers, especially when the training set size is small [9, 26].

In this paper, we propose a naïve Bayes regularized logistic regression model (NBRLR) for classification problems, which uses regularization to shrink the estimates of a LR model towards the NB estimate. As LR and NB form a well known discriminative-generative pair, our work adds to the literature that explore hybrid models which take advantage of both approaches. Such models can be placed into two categories. The first category comprises two-stage approaches, which train the model generatively with the NB model in one stage, while training the model discriminatively with the LR model in the other stage. Raina *et al*. [45] and Fujino *et al*. [21] investigate a class of hybrid model for supervised learning in the context of text classification problems, that are partly generative and partly discriminative. Specifically, they allow different partitions of the predictors into subgroups, each of which is modeled under the NB assumption, based on domain knowledge. Then these subgenerative models are combined with weight parameters that are determined discriminatively. Our study differs because we do not require any prior domain knowledge to fit a model.

Kang and Tian [33] introduce a restricted class of Bayesian network classifier where they use LR as the discriminative component, and NB as the generative component. Tan and Shenoy [52] examine the construction of such hybrid models, i.e., to decide whether a given predictor should be assigned to the LR part, or to the NB part. Specifically, they develop a metric to compare models, which uses conditional independence as a proxy for model bias and training sample size as a proxy for variance. The weakness of this method is that it serves as a selection mechanism, a predictor is either classified as a NB or LR predictor with no middle ground. Our proposed method is a shrinkage approach, which is more stable to small perturbations of data changes, and may improve the prediction accuracy.

Our work belongs to the second category, which uses the maximum likelihood parameterization of NB to pre-condition the discriminative search of LR. Zaidi *et al*. [59,60] discuss a weighted variant of NB with predictor weights selected by minimizing either the negative conditional log likelihood or the mean squared error, rather than based on measures of predictiveness. Their strategy can also be viewed as using weights to alleviate the predictor independence assumption of NB. The resulting weighted NB model is exactly equivalent to LR, but computationally much more efficient. Zaidi *et al.* [61] introduces *accelerated logistic regression* for training LR with high-order predictors. The proposed method significantly improves the efficiency and reduces the bias of LR, which makes it particularly useful for large datasets. In these papers, authors search for the optimal feature weights of the weighted NB by maximizing discriminative scores. Our work differs in that, we estimate the LR coefficients by a penalized likelihood with coefficients being shrunk towards the NB estimates.

An outline of the remainder of the chapter is as follows. In Section 2, we compare the LR and the NB models, and describe our method for both the categorical and numerical predictors. We also provide theoretical results, including consistency of our estimator. Section 3 presents the coordinate descent algorithm we use. Section 4 includes simulations results to show how our estimator performance is affected by the number of predictors and the dependence among predictors, under three simulation settings. In Section 5, we provide empirical results from experiments using ten datasets from the UCI Machine Learning Repository. Finally, in Section 6, we summarize and conclude.

2.2 Naïve Bayes Penalized Logistic Regression

2.2.1 Logistic Regression

In this study, we consider the independent and identically distributed samples $\{y_i, \mathbf{x}_i\}_{i=1}^n$, with $\mathbf{x}_i = (1, \mathbf{x}_{i1}^T, \ldots, \mathbf{x}_{ip}^T)^T$, $\mathbf{x}_{ij} = [x_{ij1}, \ldots, x_{ij(df_j)}]^T \in \mathbb{R}^{df_j}$ and $y_i \in \{0,1\}$. We consider two cases, either the predictors are all continuous or all categorical. Where, df_j is the degrees of freedom of

the j^{th} predictor. For example, the main effect of a categorical predictor with 4 levels has $df = 3$, whereas a continuous predictor has $df = 1$. Define $d = \sum_{j=1}^{p} df_j$, then $\mathbf{x}_i \in \mathbb{R}^{d+1}$.

LR is a discriminative classifier which directly learns the conditional probability $P(y_i = 1 \mid \mathbf{x}_i)$, by assuming the form

$$P(y_i = 1 \mid \mathbf{x}_i) = \frac{\exp(\mathbf{x}_i^T \beta^*)}{1 + \exp(\mathbf{x}_i^T \beta^*)}. \tag{2.1}$$

where $\beta^* = (\beta_0^*, \beta_1^{*T}, \ldots, \beta_p^{*T})^T \in \mathbb{R}^{d+1}$ with $\beta_0 \in R$ being the intercept, and $\beta_j \in \mathbb{R}^{df_j}$ being the coefficient corresponding the the j^{th} predictor. In LR, β^* is estimated by maximizing the conditional likelihood as:

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^{d+1}}{\arg\max} \prod_{i=1}^{n} P(y_i = 1 \mid \mathbf{x}_i, \beta).$$

LR is a well-known low-bias high-variance estimator. In fact, as shown by Ng and Jordan [39] LR is asymptotically the best linear classifier. However, when the sample size is relatively small compared to the number of predictors, LR estimates can have very large variances. In the cases of perfect fits, they can be infinitely large. One advantage of regularization techniques, for example lasso [55], is they increase the stability of the estimates.

For a vector $\mathbf{a} = (a_1, \ldots, a_q)^T \in \mathbb{R}^q$, define the L_p-norm as $||\mathbf{a}||_p = (\sum_{i=1}^{q} |a_i|^p)^{1/p}$. The lasso estimator for logistic regression is defined as

$$\hat{\beta}_\lambda = \underset{\beta \in \mathbb{R}^{d+1}}{\arg\min} \left[-\frac{1}{n} l(\beta) + \frac{\lambda}{n} \sum_{j=1}^{p} ||\beta_j||_1 \right],$$

where $l(\beta) = \sum_{i=1}^{n} \{y_i \mathbf{x}_i' \beta - \log[1 + \exp(\mathbf{x}_i' \beta)]\}$ is the log-likelihood function. By assuming sparsity of the true β^*, lasso is particularly useful in the high-dimensional situation. This assumption is primarily driven by the "*bet on sparsity*" principle ([28]):

"*Use a procedure that does well in sparse problems, since no procedure does well in dense problems.*"

Our intuition is that a good regularization strategy is to shrink the regression coefficient towards the values which are close to the truth. In the cases where p is relatively small compared to n, the sparsity assumption may not be necessary. Specifically, the predictors are less likely to be irrelevant with the class variable, and tend to be influential. Shrinking the coefficients of influential predictors towards zero introduces bias, and makes the regression estimates suboptimal. Ridge regression is another common regularization method, but unlike lasso it does not assume sparsity. Ridge estimators reduce the variance caused by correlated predictors, but a the cost of introducing bias to the estimator. However, such sacrifice may not be worthy when p is relatively small compared to n, as there tends to be less multicollinearity among predictors. In this paper, we propose a model based approach for balancing the bias–variance tradeoff by shrinking β^* towards the NB estimate, instead of zero. In some setting NB can be preferred to LR because of the low variance in NB estimates. In the following we present the equivalent function forms of LR and NB, which, along with the small variance in NB estimates, motivates our decision to shrink LR coefficients towards the NB estimates.

2.2.2 Logistic Regression versus Naïve Bayes

Naïve Bayes (NB) is a simple and effective supervised classification model based on applying Bayes' rule with the strong assumption of conditional independence, i.e., predictors are conditionally independent of each other given the class variable. Define $\tilde{P}_j(\mathbf{x}_{ij}|y_i=\tilde{y}) = \prod_{k=1}^{df_j} P(x_{ijk} = 1|y_i=\tilde{y})^{x_{ijk}}$. When the class variable is binary, NB can be expressed as:

$$P(y_i = 1 \mid \mathbf{x}_i) = \frac{P(y_i=1)\prod_{j=1}^{p}\tilde{P}_j(\mathbf{x}_{ij}|y_i=1)}{P(y_i=0)\prod_{j=1}^{p}\tilde{P}_j(\mathbf{x}_{ij}|y_i=0) + P(y_i=1)\prod_{j=1}^{p}\tilde{P}_j(\mathbf{x}_{ij}|y_i=1)}. \tag{2.2}$$

The conditional independence assumption reduces the complexity of the naïve Bayes model, and therefore naïve Bayes exhibits low variance, and performs surprisingly well when the training set size is small [9, 26]. However, conditional independence assumption rarely holds in practice. Any violation of the assumption will result in a bias, and make naïve Bayes estimates suboptimal.

Accordingly, naïve Bayes is a low-variance high-bias classifier, in comparison to LR [**?**].

To take advantage of both approaches, many papers have explored hybrid models that combine LR and NB into one model. One approach is to fit the model in two stages, a generative stage where we fit a NB model and a discriminative stage where we fit an LR model. Methods in the second category use the maximum likelihood parameterization of NB to pre-condition the discriminative search of LR [59–61]. Our method belongs to the second category, but instead of searching for the optimal feature weights of the weighted NB by maximizing discriminative scores, we estimate the LR coefficients by a penalized likelihood with coefficients being shrunk towards the NB estimates.

2.2.3 Naïve Bayes Regularized Logistic Regression: Categorical Predictors

Our approach is primarily motivated by the fact that LR and NB converge toward the identical classifier assuming that NB's conditional independence holds. Specifically, we rewrite the parametric form of $P(y_i = 1 \mid x_i)$ of NB as:

$$\begin{aligned} P(y_i = 1 \mid \mathbf{x}_i) &= \frac{P(y_i=1)\prod_{j=1}^{p}\tilde{P}_j(\mathbf{x}_{ij}|y_i=1)}{P(y_i=0)\prod_{j=1}^{p}\tilde{P}_j(\mathbf{x}_{ij}|y_i=0)+P(y_i=1)\prod_{j=1}^{p}\tilde{P}_j(\mathbf{x}_{ij}|y_i=1)} \\ &= \frac{\exp\left[\log\frac{P(y_i=1)}{P(y_i=0)}+\sum_{j=1}^{p}\log\frac{\tilde{P}_j(x_{ij}|y_i=1)}{\tilde{P}_j(x_{ij}|y_i=0)}\right]}{1+\exp\left[\log\frac{P(y_i=1)}{P(y_i=0)}+\sum_{j=1}^{p}\log\frac{\tilde{P}_j(x_{ij}|y_i=1)}{\tilde{P}_j(x_{ij}|y_i=0)}\right]}. \end{aligned} \tag{2.3}$$

Define $G_{ij}(a) = 1 - \sum_{k=1}^{df_j} P(x_{ijk} = 1|y_i = a)$. Then (2.3) is equivalent to

$$P(y_i = 1 \mid \mathbf{x}_i) = \frac{\exp\left[\log\frac{P(y_i=1)}{P(y_i=0)}+\sum_{j=1}^{p}\log\frac{G_{ij}(1)}{G_{ij}(0)}+\sum_{j=1}^{p}\log\frac{\tilde{P}_j(x_{ij}|y_i=1)/G_{ij}(1)}{\tilde{P}_j(x_{ij}|y_i=0)/G_{ij}(0)}\right]}{1+\exp\left[\log\frac{P(y_i=1)}{P(y_i=0)}+\sum_{j=1}^{p}\log\frac{G_{ij}(1)}{G_{ij}(0)}+\sum_{j=1}^{p}\log\frac{\tilde{P}_j(x_{ij}|y_i=1)/G_{ij}(1)}{\tilde{P}_j(x_{ij}|y_i=0)/G_{ij}(0)}\right]}. \tag{2.4}$$

This is precisely the form of $P(y_i = 1 \mid \mathbf{x}_i)$ of LR, where the intercept is $\beta_0 = \log\frac{P(y_i=1)}{P(y_i=0)} + \sum_{j=1}^{p}\log\frac{G_{ij}(1)}{G_{ij}(0)} \in \mathbb{R}$ and the remaining coefficients are $\beta_{jk} = \log\frac{P(x_{ijk}=1|y_i=1)/G_{ij}(1)}{P(x_{ijk}=1|y_i=0)/G_{ij}(0)} \in \mathbb{R}$. Then (2.1) and (2.4) are equivalent.

Next, we define the NBRLR model as a classification method for categorical data. Following the conventional regularization methods set-up, we assume that $\{x_{ijk}\}_{i=1}^{n}$, the values for the kth class of the jth predictor, are standardized so that $\sum_{i=1}^{n} x_{ijk} = 0$ and $\frac{1}{n}\sum_i x_{ijk}^2 = 1$ for all j and k. Let's denote by $\{\hat{\eta}_0, \hat{\eta}_j\}$ is the naïve Bayes estimate of the model defined as:

$$\begin{aligned} \hat{\eta}_0 &= \log\frac{P(y_i=1)}{P(y_i=0)} + \sum_{j=1}^{p}\log\frac{G_{ij}(1)}{G_{ij}(0)}, \\ \hat{\eta}_{jk} &= \log\frac{P(x_{ijk}=1 \mid y_i=1)/G_{ij}(1)}{P(x_{ijk}=1 \mid y_i=0)/G_{ij}(0)}. \end{aligned}$$

The NBRLR estimator $\hat{\beta}_{\lambda,\hat{\eta}}$ is defined by

$$\hat{\beta}_{\lambda,\hat{\eta}} = \underset{\beta\in\mathbb{R}^{d+1}}{\arg\min} -\frac{1}{n}l(\beta) + \frac{\lambda}{n}\sum_{j=0}^{p}||\beta_j - \hat{\eta}_j||_1, \tag{2.5}$$

where $\lambda \geq 0$ is the tuning parameter that controls the amount of regularization. Note that for $\lambda = 0$, then $\hat{\beta}_{\lambda,\hat{\eta}}$ is equivalent to the LR estimate. In addition for a sufficiently large value of λ, $\hat{\beta}_{\lambda,\hat{\eta}}$ will provide predicted probabilities that are the same as NB. As the NB probabilities depend on the value of the intercept, we need to shrink the intercept towards the NB estimate. That is why we penalize the intercept, which is uncommon in traditional regularization techniques, such as lasso and ridge regression.

In practice, when p is relative small as compared with n, the sparsity assumption is more likely to be violated. However, this is not the case for NB's assumption of conditional independence. The smaller number of predictors raises the chance of satisfying the conditional independence assumption among features, which makes the NB estimates more reliable. In these settings, we believe that shrinking coefficients towards the NB estimates, instead of zero, will produce less bias while still providing a reduction in the variance compared to LR. Although, the proposed NBRLR estimator will have larger variance than Lasso because no coefficients will be set to zero and $\hat{\eta}_j$ is an estimate for all $j \in \{0,\ldots,p\}$. This issue will be more problematic for larger p. However, the proposed method can outperform lasso when the predictors are informative and the number of

predictors is small relative to the sample size.

2.2.4 Naïve Bayes Regularized Logistic Regression: Continuous Predictors

In this subsection, we extend the NBRLR model to cases where we have continuous predictors. One common assumption for each continuous predictor x_{ij} of a NB model is that, $x_{ij} \mid y_i = 1 \sim N(u_{j1}, \sigma_j^2)$ and $x_{ij} \mid y_i = 0 \sim N(u_{j0}, \sigma_j^2)$ and let $f_j(x_j|y)$ be the corresponding conditional distribution of x_j. Note that the standard deviations σ_j varies from predictor to predictor, but does not depend on the value of y_i. Thus, we can expand the summation term in the numerator of the Eq. (2.3) as:

$$\sum_{j=1}^{p} \log\left[\frac{f_j(x_{ij}|y_i=1)}{f_j(x_{ij}|y_i=0)}\right] = \sum_{j=1}^{p} \log\left[\frac{\frac{1}{\sqrt{2\pi}\sigma_j}\exp(-\frac{(x_{ij}-u_{j1})^2}{2\sigma_j^2})}{\frac{1}{\sqrt{2\pi}\sigma_j}\exp(-\frac{(x_{ij}-u_{j0})^2}{2\sigma_j^2})}\right].$$

Define $\rho = P(y_i = 1)$. We get the direct equivalence between LR and NB by substituting this expression back into equation 2.3:

$$P(y_i = 1 \mid \mathbf{x}_i) = \frac{\exp(\log\frac{\rho}{1-\rho} + \sum_{j=1}^{p}\frac{u_{j0}^2-u_{j1}^2}{2\sigma_j^2} + \sum_{j=1}^{p}\frac{u_{j1}-u_{j0}}{\sigma_j^2}x_{ij})}{1+\exp(\log\frac{\rho}{1-\rho} + \sum_{j=1}^{p}\frac{u_{j0}^2-u_{j1}^2}{2\sigma_j^2} + \sum_{j=1}^{p}\frac{u_{j1}-u_{j0}}{\sigma_j^2}x_{ij})}.$$

We define the NB penalty $\{\hat{\theta}_0, \hat{\theta}_j\}$ for our NBRLR estimator as:

$$\begin{aligned}\hat{\theta}_0 &= \log\frac{\rho}{1-\rho} + \sum_{j=1}^{p}\frac{u_{j0}^2-u_{j1}^2}{2\sigma_j^2},\\ \hat{\theta}_j &= \frac{u_{j1}-u_{j0}}{\sigma_j^2}.\end{aligned}$$

The NBRLR estimator $\hat{\beta}_\lambda$ for continuous predictors is

$$\hat{\beta}_{\lambda,\hat{\eta}} = \underset{\beta\in\mathbb{R}^{p+1}}{\arg\min} -\frac{1}{n}l(\beta) + \frac{\lambda}{n}\sum_{j=0}^{p}|\beta_j - \hat{\theta}_j|, \tag{2.6}$$

where $\lambda \geq 0$ is the tuning parameter that controls the amount of regularization.

2.2.5 Asymptotic Results

In this section we will provide a consistency result for the NBRLR estimator. In fact, the result is a general result that will apply to shrinking towards any value. The asymptotic result does depend on some conditions. In this setting, let $\mathbf{x}_i = (1, x_{i1}, \ldots, x_{iq})^\top \in \mathbb{R}^{q+1}$, so in Section 2.3 we have $q = d$ and for Section 2.4 we have $q = p$. For the theoretical results we do not assume that all predictors are categorical or all predictors are continuous. Let $\hat{\eta}_j$ represent the, potentially estimated, value the jth coefficient is being shrunk towards and $\eta = (\eta_0, \ldots, \eta_q)^T \in \mathbb{R}^q$. Define,

$$Z_n(\beta, \hat{\eta}) = -\frac{1}{n}\sum_{i=1}^{n}(y_i x_i'\beta - \log\left[1 + \exp(x_i'\beta)\right]) + \frac{\lambda}{n}\sum_{j=0}^{q}|\beta_j - \hat{\eta}_j| \tag{2.7}$$

and

$$\hat{\beta}_{\lambda,\hat{\eta}} = \underset{\beta\in\mathbb{R}^{q+1}}{\arg\min} Z_n(\beta, \hat{\eta}). \tag{2.8}$$

Define $\psi_{jkm}(\beta) = \frac{\partial^3}{\partial\beta_j\partial\beta_k\partial\beta_m}l(\beta)$. To prove $\hat{\beta}_{\lambda,\hat{\eta}}$ is a consistent estimator of β^*, as defined in (2.1), we require the following conditions.

Condition 1. *Let* Σ *and* Σ_{β^*} *be positive definite matrices where*

$$\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^T \to \Sigma$$

and

$$\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^T\frac{\exp(\mathbf{x}_i'\beta^*)}{\left[1+\exp(\mathbf{x}_i'\beta^*)\right]^2} \to \Sigma_{\beta^*}.$$

Condition 2. *For some positive constant C_1, define $B_n = \{\beta | ||\beta - \beta^*||_2 \leq C_1 n^{-1/2}\}$. There exist positive constants C_2 and C_3 such that for all $\beta \in B_n$, $j \in \{0,\ldots,q\}$, $k \in \{0,\ldots,q\}$ and $m \in \{0,\ldots,q\}$ that*

$$C_2 < \left|\psi_{jkm}(\beta)\right| < C_3.$$

For Condition 1 Knight and Fu [35] make the same assumption about the design matrix in their proof of consistency for a general class of regularized estimators, including lasso, with the least squares loss function. The assumption regarding Σ_{β^*} provides that the asymptotic variance of the LR estimator is well behaved. Condition 2 ensures that when β is close to β^* that $l(\beta)$ can be well approximated by a second order Taylor expansion. Similar conditions have been made on the third partial partial derivative of a likelihood, when analyzing the asymptotics of a penalized likelihood method [15, 37].

Theorem 1. *Assume that* (2.1) *and Conditions 1 and 2 hold and that $\lambda = o(n)$ then $||\hat{\beta}_{\lambda,\hat{\eta}} - \beta^*|| = O_P(n^{-1/2})$.*

Proof. By the properties of convex functions, for more details see the proof of Theorem 2.1 in He and Shi [29] and Corollary 25, p.47, of Eggleston [12], it is sufficient to show that there exists L such that

$$P\left[\inf_{\left|\left|\beta-\beta^*\right|\right|_2 = Ln^{-1/2}} Z_n(\beta,\hat{\eta}) - Z_n(\beta^*,\hat{\eta}) > 0\right] \to 1. \tag{2.9}$$

By Taylor's approximation, for any β there is $\tilde{\beta}$ between β and β^* such that

$$\begin{aligned} l(\beta) - l(\beta^*) &= (\beta-\beta^*)^T\left[\frac{\partial}{\partial\beta} l(\beta^*)\right] + \frac{1}{2}(\beta-\beta^*)^T\left[\frac{\partial^2}{\partial\beta^2} l(\beta^*)\right](\beta-\beta^*) \\ &+ \sum_{j=0}^{q}\sum_{k=0}^{q}\sum_{m=0}^{q}(\beta_j-\beta_j^*)(\beta_k-\beta_k^*)(\beta_m-\beta_m^*)\psi_{jkm}(\tilde{\beta}). \end{aligned}$$

Notice,

$$\begin{aligned}\frac{\partial}{\partial\beta}l(\beta) &= -\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i\left[y_i-\frac{\exp(\mathbf{x}_i'\beta)}{1+\exp(\mathbf{x}_i'\beta)}\right],\\ \frac{\partial^2}{\partial\beta^2}l(\beta) &= \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^T\frac{\exp(\mathbf{x}_i'\beta)}{\left[1+\exp(\mathbf{x}_i'\beta)\right]^2}.\end{aligned}$$

Therefore by Condition 1

$$\begin{aligned}\sup_{\left|\left|\beta-\beta^*\right|\right|_2=Ln^{-1/2}}\left|(\beta-\beta^*)^T\left[\frac{\partial}{\partial\beta}l(\beta^*)\right]\right| &= \sup_{\left|\left|\beta-\beta^*\right|\right|_2=Ln^{-1/2}}\left|(\beta-\beta^*)^T\frac{1}{\sqrt{n}}\frac{1}{\sqrt{n}}\sum_{i=1}^{n}\mathbf{x}_i\left[y_i-\frac{\exp(\mathbf{x}_i'\beta^*)}{1+\exp(x_i'\beta^*)}\right]\right|\\ &= O_p(n^{-1/2}\left|\left|\beta-\beta^*\right|\right|_2)=O_P(n^{-1}L).\end{aligned}$$

In addition by Condition 1 there exists a positive constant $\tilde{C}$ such that

$$\inf_{\left|\left|\beta-\beta^*\right|\right|_2=Ln^{-1/2}}\frac{1}{2}(\beta-\beta^*)^T\left[\frac{\partial^2}{\partial\beta^2}l(\beta^*)\right](\beta-\beta^*)\geq\tilde{C}||\beta-\beta^*||_2^2=\tilde{C}L^2n^{-1}.$$

By Condition 2

$$\begin{aligned}\sum_{j=0}^{p}\sum_{k=0}^{q}\sum_{m=0}^{q}(\beta_j-\beta_j^*)(\beta_k-\beta_k^*)(\beta_m-\beta_m^*)\psi_{jkm}(\tilde{\beta}) &\leq C_3\sum_{j=0}^{q}|\beta_j-\beta_j^*|\sum_{k=0}^{q}|\beta_k-\beta_k^*|\sum_{m=0}^{q}|\beta_m-\beta_m^*|\\ &\leq C_3q^{3/2}||\beta-\beta^*||_2^3,\end{aligned}$$

and

$$\begin{aligned}\sum_{j=0}^{q}\sum_{k=0}^{q}\sum_{m=0}^{q}(\beta_j-\beta_j^*)(\beta_k-\beta_k^*)(\beta_m-\beta_m^*)\psi_{jkm}(\tilde{\beta}) &\geq -C_2\sum_{j=0}^{q}|\beta_j-\beta_j^*|\sum_{k=0}^{q}|\beta_k-\beta_k^*|\sum_{m=0}^{q}|\beta_m-\beta_m^*|\\ &\geq -C_2q^{3/2}||\beta-\beta^*||_2^3.\end{aligned}$$

Therefore

$$\sup_{\left|\left|\beta-\beta^*\right|\right|_2=Ln^{-1/2}} \left| \sum_{j=0}^{q}\sum_{k=0}^{q}\sum_{m=0}^{q} (\beta_j-\beta_j^*)(\beta_k-\beta_k^*)(\beta_m-\beta_m^*)\psi_{jkm}(\tilde{\beta}) \right| = O_P(L^3 n^{-3/2}).$$

Under the assumption that $\lambda = o(n)$

$$\frac{\lambda}{n}\sum_{j=0}^{q} |\beta_j-\hat{\eta}_j| - |\beta_j^*-\hat{\eta}_j| \le \frac{\lambda}{n}\sum_{j=0}^{q}|\beta_j-\beta_j^*| \le \frac{\lambda}{n}\sqrt{q}||\beta-\beta^*||_2 = o(\sqrt{q}Ln^{-1/2}),$$

and

$$\frac{\lambda}{n}\sum_{j=0}^{q} |\beta_j-\hat{\eta}_j| - |\beta_j^*-\hat{\eta}_j| \ge -\frac{\lambda}{n}\sum_{j=0}^{q}|\beta_j-\beta_j^*| \ge -\frac{\lambda}{n}\sqrt{q}||\beta-\beta^*||_2 = o(\sqrt{q}Ln^{-1/2}).$$

Therefore, for sufficiently large L the lower bound of the quadratic term will dominate the other terms and (2.9) holds.

□

2.3 Algorithm

We consider a coordinate descent step for solving (2.8), which is a generalization of (3.2) and (3.3), following Friedman *et al*. [27]. The unpenalized log-likelihood $l(\beta)$ is maximized by implementing Newton's method with iteratively reweighted least square algorithm. Specifically, given the current estimates of the parameters $\beta^{old} = (\beta_0^{old}, \beta_1^{old}, \ldots, \beta_p^{old})^T$ with corresponding probability $p^{old}(\mathbf{x}_i) = P(y_i = 1 \mid \mathbf{x}_i, \beta^{old})$ for observation i, we obtain a quadratic approximation to the $l(\beta)$ as:

$$l_Q(\beta) = -\frac{1}{2}\sum_{i=1}^{n} p^{old}(\mathbf{x}_i)\left[1-p^{old}(\mathbf{x}_i)\right]\left\{\mathbf{x}_i^T\beta^{old} + \frac{y_i - p^{old}(\mathbf{x}_i)}{p^{old}(\mathbf{x}_i)\left[1-p^{old}(\mathbf{x}_i)\right]} - \mathbf{x}_i^T\beta\right\}^2 + C, \quad (2.10)$$

where C is a constant term. Then our task becomes minimizing the following penalized weighted least-squares problem

$$-\frac{1}{n} l_Q(\beta) + \lambda \sum_{j=0}^{q} |\beta_j - \hat{\eta}_j|. \tag{2.11}$$

Define soft-thresholding operator $S(a,b) = sign(a)(|a|-b)_+$, the update of coordinate descent is performed by

$$\beta_j^{new} \leftarrow \hat{\eta}_j + \frac{S(A - \hat{\eta}_j B, \lambda)}{B} \tag{2.12}$$

where

$$\begin{aligned} A &= \frac{1}{n}\sum_{i=1}^{n} p^{old}(\mathbf{x}_i)\left[1-p^{old}(\mathbf{x}_i)\right] x_{ij}\left[x_{ij}\beta_j^{old} + \frac{y_i - p^{old}(\mathbf{x}_i)}{p^{old}(\mathbf{x}_i)(1-p^{old}(\mathbf{x}_i))}\right], \\ B &= \frac{1}{n}\sum_{i=1}^{n} p^{old}(\mathbf{x}_i)\left[1-p^{old}(\mathbf{x}_i)\right] x_{ij}^2. \end{aligned}$$

Thus, equation (2.11) is minimized by iterating through $j \in \{0,1,\ldots,q\}$ until its difference between two iterations is less than 10^{-7}.

Given a fixed value of λ, we propose the following algorithm.

1. Begin with initial estimates of $\hat{\beta}^0 = \{\hat{\beta}_0^0, \hat{\beta}_1^0, \ldots, \hat{\beta}_q^0\}$.
2. For the t^{th} step, where $t \geq 1$, repeat the steps below until the difference of the penalized log-likelihood (3.2) between $(t-1)^{th}$ and t^{th} step is less than 10^{-7}.
   (a) Update the quadratic approximation ł_Q with the current parameters $\hat{\beta}^{t-1}$.
   (b) Given current ł_Q, the t^{th} iterative estimate of β is:

$$\hat{\beta}^t = \arg\min_{\beta} -\frac{1}{n} l_Q(\beta) + \lambda \sum_{j=0}^{q} |\beta_j - \hat{\eta}_j|,$$

   where it can be solved following the coordinate descent solution from (2.12) using $\hat{\beta}^{t-1}$

as the current estimate β^{old}.

An R package implementing the described algorithm will be made publicly available upon acceptance of publication of this work.

2.4 Simulations

In this section, we compare the naïve Bayes regularized logistic estimator with pure LR, pure NB, and regularized LR (lasso). Lasso is fit using the glmnet package [27] in R. We use 10-folds cross validation with the objective of minimizing the out-of-sample prediction error to determine the turning parameter in each case. When the predictors are categorical, the estimation of parameters of pure NB model is conducted using the Laplace correction [40] to prevent the high influence of zero probabilities. Specifically, we add one of each class to the data. The Laplace corrected values are used for the NBLR estimator.

Under each simulation setting, we generate 100 training sample to fit the models, and 1000 testing samples to assess their prediction performance. Let N be the testing sample size, y_i be the observed class for the ith testing observation, $\hat{y}_i$ be the estimated class for testing observation i and $\hat{P}(y_i = 0)$ be the estimated probability that y_i is 0. If the predicted probability of an observation is below .5 than we predict that sample belongs to class 0, otherwise we predict it belongs to class 1. We compare the prediction performance of the models using the average prediction 0-1 loss ($L_{0\text{-}1}$) and root squared prediction error ($RSPE$), which are defined as

$$L_{0\text{-}1} = 1 - \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}(y_i = \hat{y}_i),$$

and

$$RSPE = \sqrt{\frac{1}{2 \cdot N}\sum_{i=1}^{N}\{(\hat{P}(y_i = 0) - \mathbf{1}(y_i = 0))^2 + (\hat{P}(y_i = 1) - \mathbf{1}(y_i = 1))^2\}}.$$

We also report the mean squared error of the estimator $\hat{\beta}$, $MSE(\hat{\beta})$, which for the truth β^* is defined as

$$MSE(\hat{\beta}) = \frac{1}{p+1}||\hat{\beta} - \beta^*||_2^2.$$

We repeat the entire procedure 100 times. When comparing the $L_{0\text{-}1}$, *RSPE* and $MSE(\hat{\beta})$ of the model, we present boxplots of the value of the three metrics in all simulations. Further, we compare the results of NBRLR to the other three methods by reporting the averages of the three metrics, performing two-tailed, paired t-tests and report the corresponding p-values.

Three simulation settings are considered in this section. First, we consider generating data with categorical predictors from a discriminative LR model. Second, we consider generating data with categorical predictors from a generative NB model. Third, we generate data with continuous predictors from a discriminative LR model. In addition, we vary the number of predictors and the dependence among predictors (conditional dependence for the second simulation setting) in each simulation, to see how these two factors affect the models' performance.

2.4.1 Simulation Setting 1: Binomial Predictor 1

In the first simulation, p categorical variables $\mathbf{x}_i = \{x_{i1}, \ldots x_{ip}\}$ are simulated by first generating $\tilde{\mathbf{x}}_i \sim N(0_p, \Sigma_{p\times p})$, where $\Sigma_{jj} = 1$ and $\Sigma_{jk} = r$ for $j \neq k$ and then x_{ij} is dichotomized as 0 if $\tilde{x}_{ij}$ is smaller than 0, and 1 otherwise. The class variable y_i is then simulated from

$$P(y_i = 1 \mid \mathbf{x}_i) = \frac{\exp(\beta_0 + \sum_{j=1}^{p} \beta_j x_{ij})}{1 + \exp(\beta_0 + \sum_{j=1}^{p} \beta_j x_{ij})}$$

where $\beta = (0, -1_{p/2}^T, 1_{p/2}^T)^T$. We consider four situations in which $p \in \{10, 50\}$ and $r \in \{0.1, 0.6\}$.

Comparisons of the four estimators in terms of $MSE(\hat{\beta})$, $L_{0\text{-}1}$ and *RSPE* across all the combinations of p and r for this setting are reported in Figure 2.1 - 2.3, respectively. Table 2.1 provides the averages of the three metrics across the four different combinations of p and r and includes p-values from a two-sided, paired t-test comparing the performance of NBRLR to the other three methods. Notice that in this section, the boxplots of $MSE(\hat{\beta})$, reported in Figure 2.1, use a log-10 scale for the Y axis due to LR's excessively poor performance in parameter estimation, especially

Table 2.1: Summary of results from simulation setting 1 comparing NBRLR with pure LR, pure NB and Lasso, at different numbers of predictors p and different levels of conditional dependence among predictors r. The Esti. columns present the averages across the 100 simulations and the p-values are from a two-sided, paired t-test comparing the performance of NBRLR with the corresponding estimator.

Simulation1			NBRLR	LR		NB		Lasso	
	p	r	Esti.	Esti.	p-value	Esti.	p-value	Esti.	p-value
MSE	10	0.1	0.325	0.450	<0.001	0.292	0.083	0.575	<0.001
		0.6	0.437	0.936	0.085	0.623	<0.001	0.693	<0.001
	50	0.1	0.744	4107.500	0.053	0.648	0.257	0.860	0.190
		0.6	0.822	25369.000	<0.001	1.421	<0.001	0.899	0.322
$L_{0\text{-}1}$	10	0.1	0.301	0.295	<0.001	0.308	<0.001	0.304	0.042
		0.6	0.361	0.353	<0.001	0.402	<0.001	0.371	0.001
	50	0.1	0.285	0.301	<0.001	0.298	<0.001	0.339	<0.001
		0.6	0.337	0.356	<0.001	0.427	<0.001	0.397	<0.001
RSPE	10	0.1	0.445	0.446	0.442	0.445	0.571	0.447	0.019
		0.6	0.470	0.472	0.044	0.500	<0.001	0.473	0.002
	50	0.1	0.446	0.544	<0.001	0.448	0.447	0.472	<0.001
		0.6	0.475	0.581	<0.001	0.574	<0.001	0.492	<0.001

when p is large. The results show that our proposed NBRLR estimator generally performs the best, especially with respsect to *RSPE*. One exception is that when we compare $L_{0\text{-}1}$ on low-dimensional datasets, $p = 10$, NBRLR does worse than LR. This might be because low-dimensional simulation setting favors the low bias estimator. Note, NB is competitive with or outperforms NBRLR in terms of $MSE(\hat{\beta})$ and *RSPE* when $r = 0.1$. In this situation, the conditional correlation among predictors given the class variable is low, which favors NB. Specifically, the conditional correlation, which is measured by standardized conditional mutual information, is on an average 0.005 for $p = 10$, and 0.004 for $p = 50$. However, they will increase to 0.134 and 0.129, respectively, when $r = 0.6$.

2.4.2 Simulation Setting 2: Binomial Predictor 2

In the second simulation, we generate the class variable y_i from $y_i \sim Bern(0.5)$. Then, p categorical variables $\mathbf{x}_i = \{x_{i1}, \ldots x_{ip}\}$ are simulated in a two step process. First, $\tilde{\mathbf{x}}_i \mid y_i = 1 \sim N(\mathbf{u}_1, \Sigma_{p\times p})$

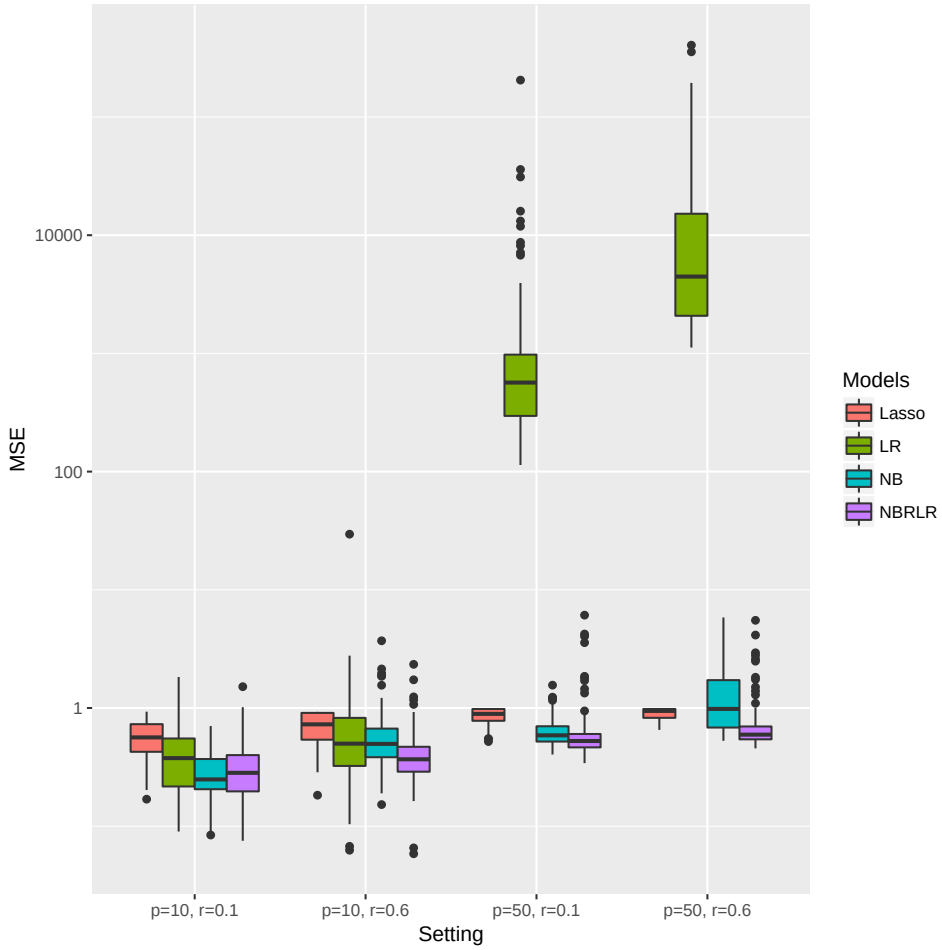

Figure 2.1: MSE results for 100 simulations in setting 1 for the four different combinations of p and r.

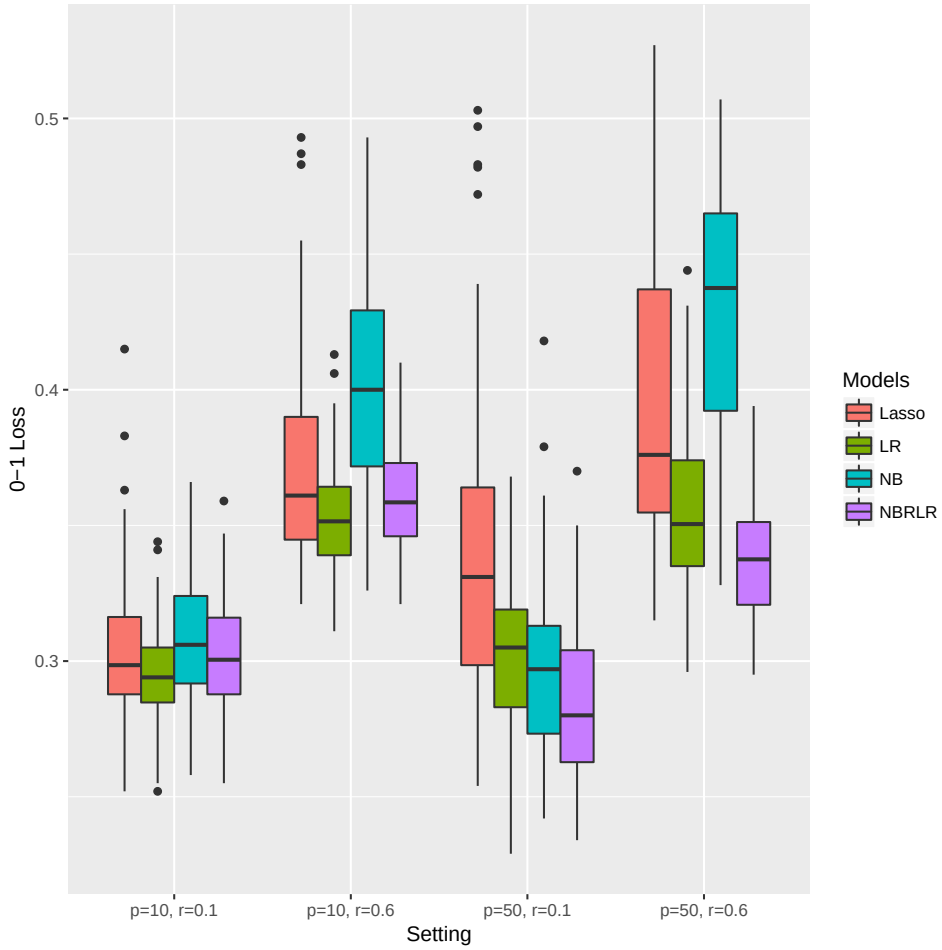

Figure 2.2: $L_{0\text{-}1}$ results for 100 simulations in setting 1 for the four different combinations of p and r.

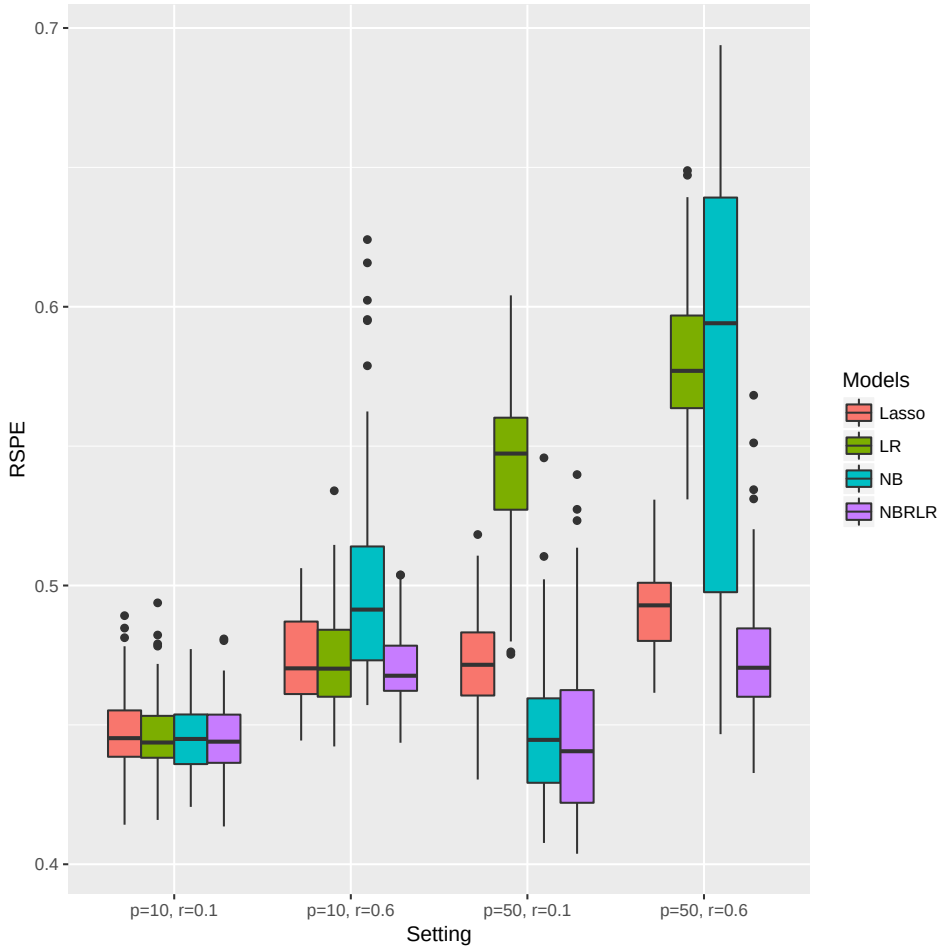

Figure 2.3: *RSPE* results for 100 simulations in setting 1 for the four different combinations of p and r.

Table 2.2: Summary results from simulation setting 2 comparing NBRLR with pure LR, pure NB and Lasso, at different numbers of predictors p and different levels of conditional dependence among predictors r. The Esti. columns present the averages across the 100 simulations and the p-values are from a two-sided, paired t-test comparing the performance of NBRLR with the corresponding estimator.

Simulation2			NBRLR	LR		NB		Lasso	
	p	r	Esti.	Esti.	p-value	Esti.	p-value	Esti.	p-value
MSE	10	0.1	0.217	0.342	<0.001	0.191	0.016	0.331	<0.001
		0.6	0.372	1.610	0.058	0.489	<0.001	0.690	<0.001
	50	0.1	0.337	2846.410	0.150	0.275	0.032	0.357	0.493
		0.6	0.701	9498.500	<0.001	1.028	<0.001	1.029	<0.001
$L_{0\text{-}1}$	10	0.1	0.330	0.334	<0.001	0.331	0.179	0.350	<0.001
		0.6	0.306	0.309	0.020	0.334	<0.001	0.316	<0.001
	50	0.1	0.175	0.278	<0.001	0.171	0.003	0.244	<0.001
		0.6	0.149	0.243	<0.001	0.247	<0.001	0.205	<0.001
RSPE	10	0.1	0.462	0.469	<0.001	0.461	0.022	0.469	<0.001
		0.6	0.441	0.448	<0.001	0.464	<0.001	0.447	<0.001
	50	0.1	0.356	0.522	<0.001	0.350	<0.001	0.416	<0.001
		0.6	0.321	0.476	<0.001	0.439	<0.001	0.376	<0.001

where $\mathbf{u}_1 = \{0.2_{p/2}, -0.2_{p/2}\}$, and $\tilde{\mathbf{x}}_i \mid y_i = 0 \sim N(\mathbf{u}_0, \Sigma_{p\times p})$ where $\mathbf{u}_0 = \{-0.2_{p/2}, 0.2_{p/2}\}$. For both distributions $\Sigma_{jj} = 1$ and $\Sigma_{jk} = r$ if $j \neq k$. Define $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})^T \in \mathbb{R}^p$ as a vector of ones and zeros, where x_{ij} is zero if $\tilde{x}_{ij}$ is smaller than 0 and x_{ij} is one otherwise. To get the value of β^* we generate 500,000 training samples, fit an LR model and treat the corresponding coefficients as β^*.

Comparisons of the four estimators in terms of $MSE(\hat{\beta})$, $L_{0\text{-}1}$ and $RSPE$ for different simulation models are reported in Figure 2.4 - 2.6, respectively. Table 2.2 is the equivalent of Table 2.1, but for simulation setting 2. When r = 0.1, NB is competitive with or outperforms NBRLR because the conditional independence assumption is only weakly violated. However, NBRLR performs the best in the rest of the settings.

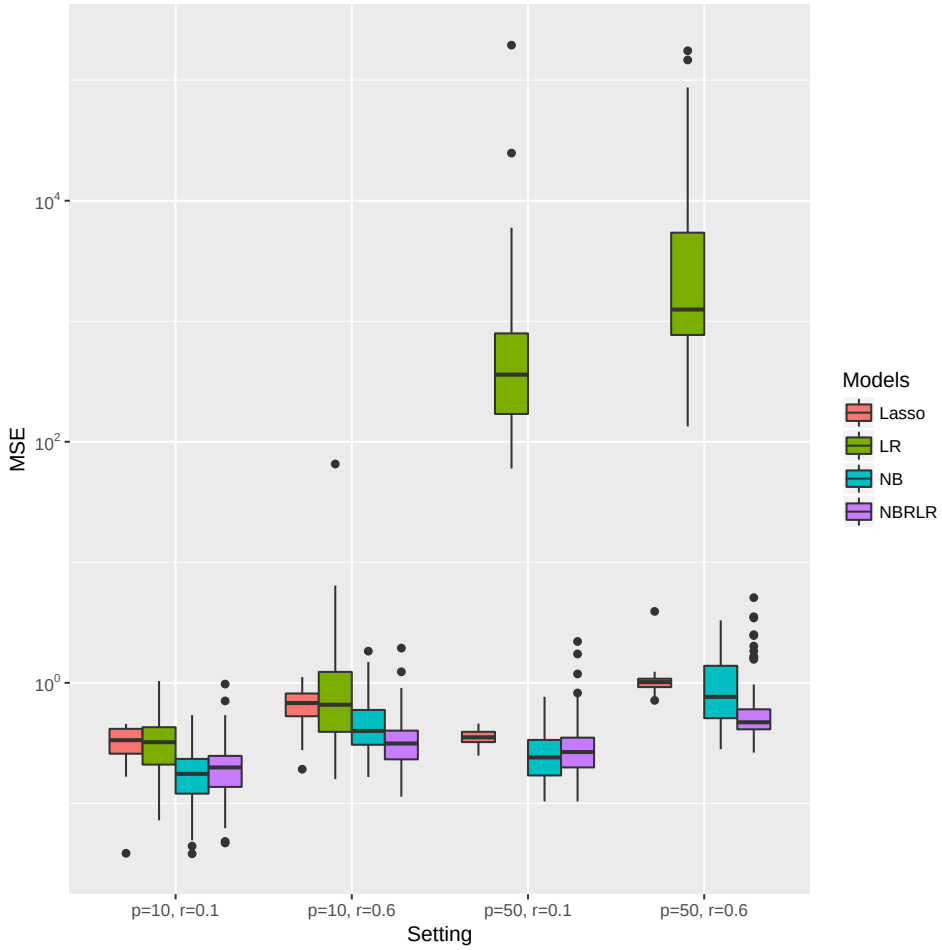

Figure 2.4: MSE results for simulation setting 2. The x-axis includes the four different combinations of p and r.

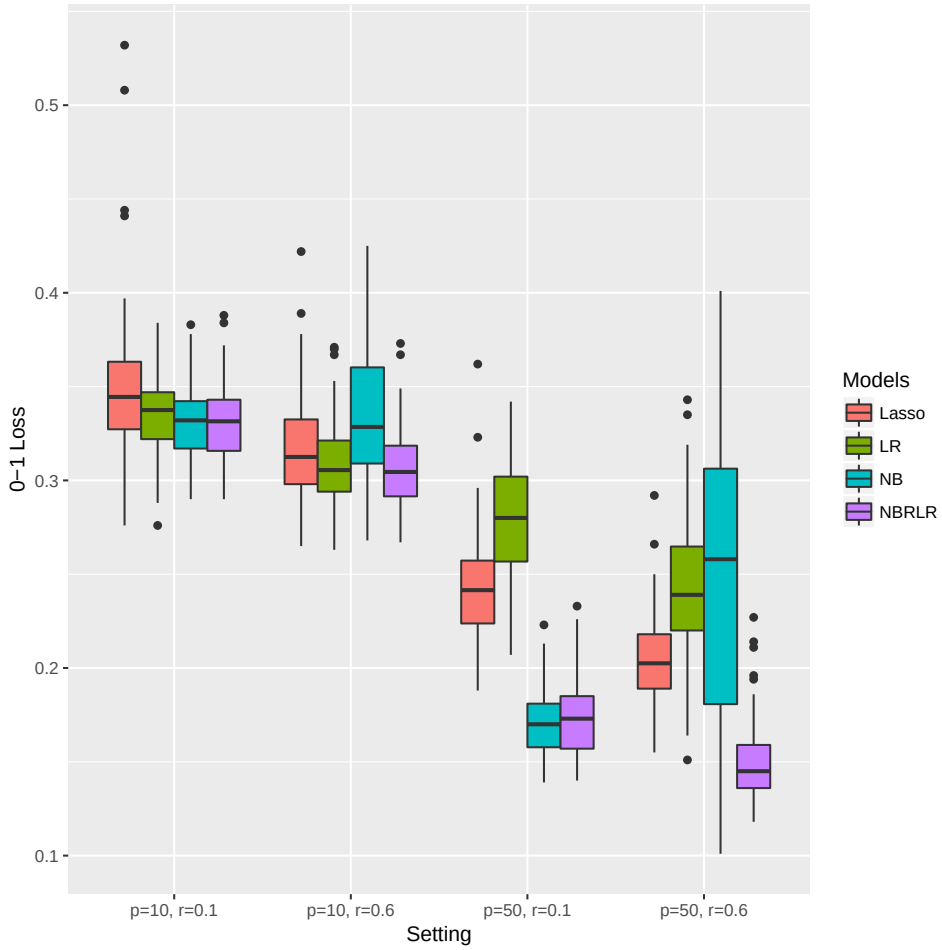

Figure 2.5: $L_{0\text{-}1}$ results for simulation setting 2. The x-axis includes the four different combinations of p and r.

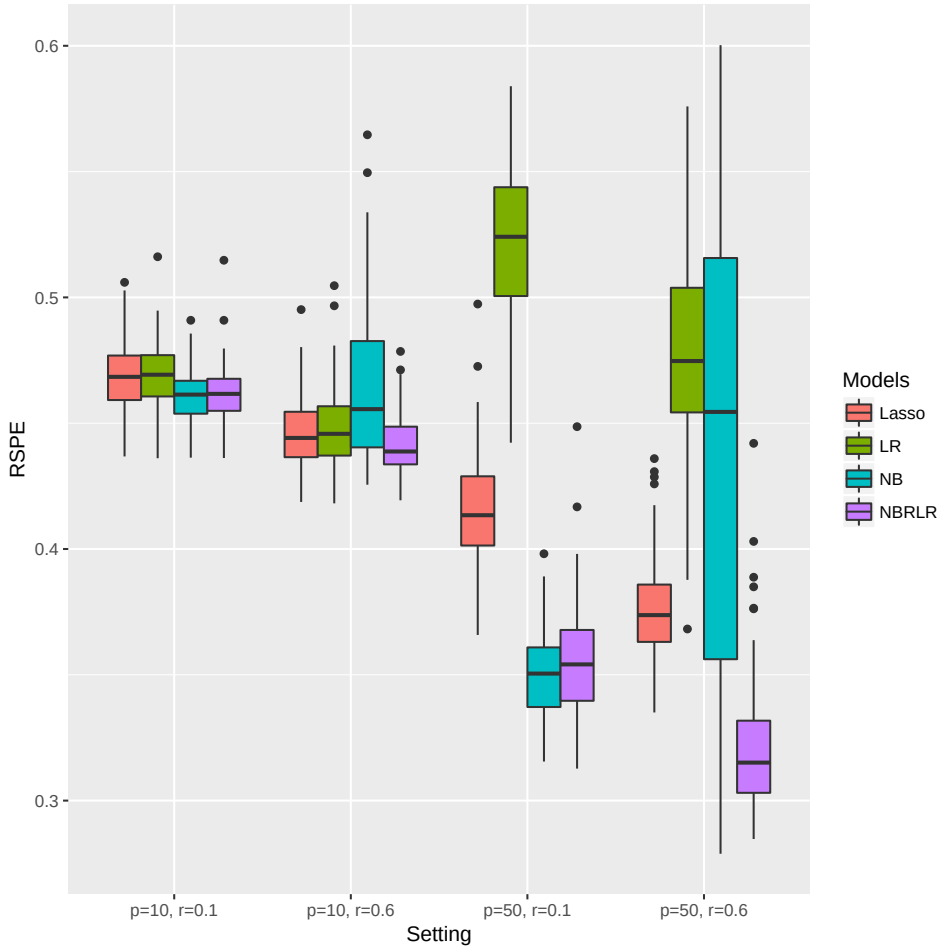

Figure 2.6: *RSPE* results for simulation setting 2. includes the four different combinations of p and r.

Table 2.3: Summary of results from simulation setting 3 comparing NBRLR with pure LR, pure NB and Lasso, at different numbers of predictors p and different levels of conditional dependence among predictors r. The Esti. columns present the averages across the 100 simulations and the p-values are from a two-sided, paired t-test comparing the performance of NBRLR with the corresponding estimator.

Simulation3			NBRLR	LR		NB		Lasso	
	p	r	Esti.	Esti.	p-value	Esti.	p-value	Esti.	p-value
MSE	10	0.1	0.207	0.463	<0.001	0.330	<0.001	0.387	<0.001
		0.6	0.282	0.428	<0.001	0.549	<0.001	0.429	<0.001
	50	0.1	0.542	129.925	<0.001	0.650	<0.001	0.860	<0.001
		0.6	0.620	782.520	<0.001	0.785	<0.001	0.844	<0.001
$L_{0\text{-}1}$	10	0.1	0.199	0.195	<0.001	0.230	<0.001	0.200	0.255
		0.6	0.260	0.256	<0.001	0.378	<0.001	0.262	0.217
	50	0.1	0.248	0.259	0.002	0.289	<0.001	0.298	<0.001
		0.6	0.282	0.289	0.038	0.434	<0.001	0.320	<0.001
RSPE	10	0.1	0.372	0.375	0.003	0.396	<0.001	0.372	0.830
		0.6	0.421	0.422	0.042	0.498	<0.001	0.421	0.698
	50	0.1	0.418	0.501	<0.001	0.446	<0.001	0.451	<0.001
		0.6	0.444	0.532	<0.001	0.590	<0.001	0.464	<0.001

2.4.3 Simulation Setting 3: Continuous Predictor

The third simulation setting is the same as the first one, but with all predictors being generated directly from the multivariate normal distribution and without dichotomization. NB estimates are made by, correctly, assuming the conditional distribution of the predictors given the response class is normal as outlined in Section 2.4. Figures 2.7 - 2.9 show the $MSE(\hat{\beta})$, $L_{0\text{-}1}$ and $RSPE$, respectively, for each simulation in setting 3. Table 2.3 is the equivalent of Tables 2.1 and 2.2, but for simulation setting 3. The results in Figures 2.7 - 2.9 and Table 2.3 demonstrate that generally NBRLR outperforms the other three methods, but there are some exceptions. When $p = 10$, LR outperforms NBRLR with respect to $L_{0\text{-}1}$, while differences with Lasso in terms of $L_{0\text{-}1}$ and $RSPE$ are not significant.

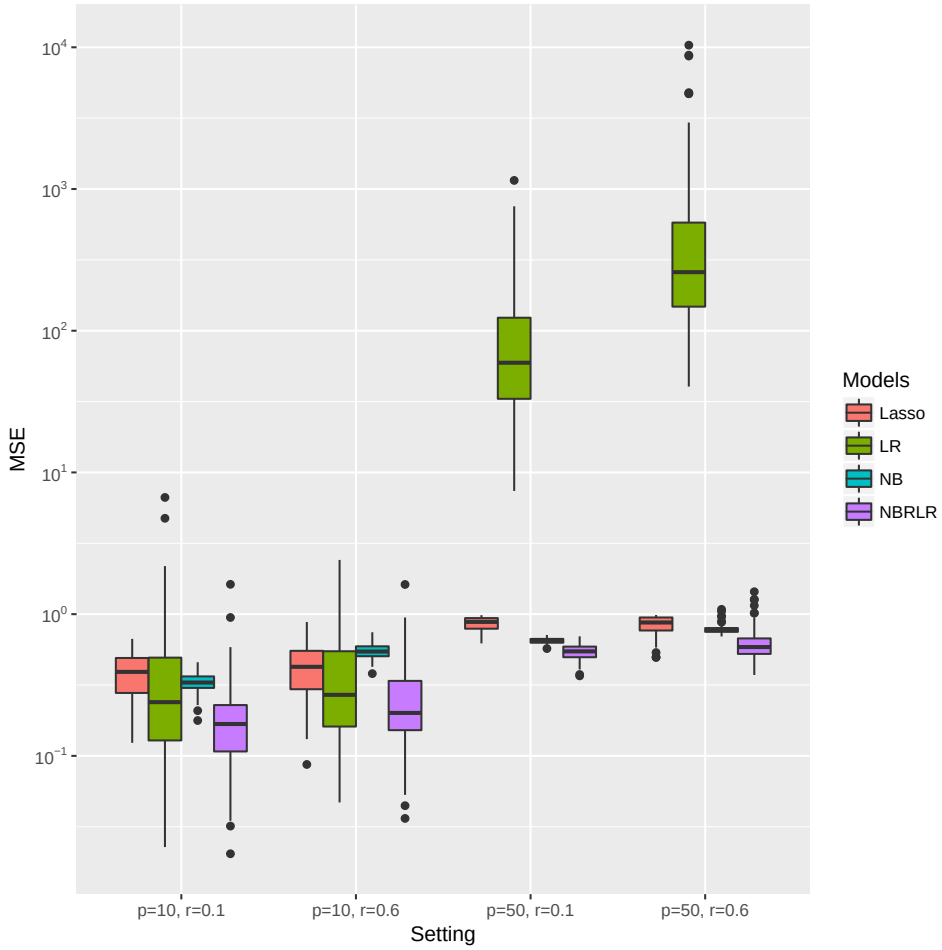

Figure 2.7: MSE results for simulation setting 3. The x-axis includes the four different combinations of p and r.

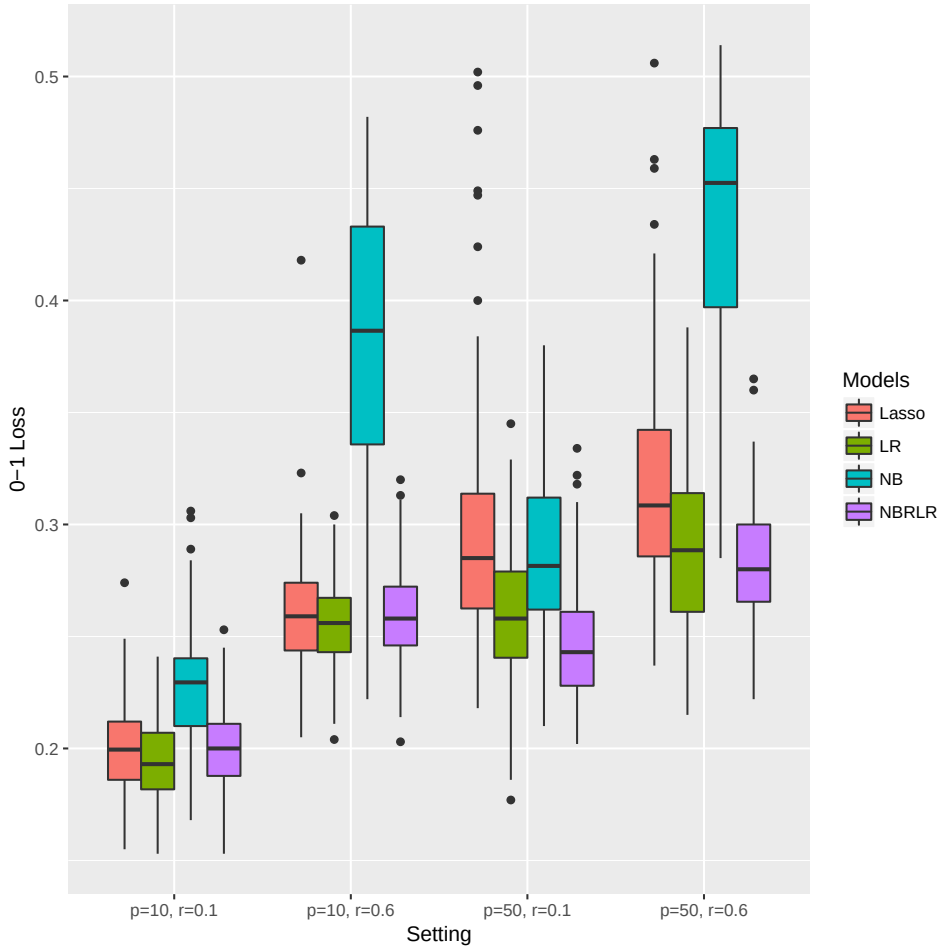

Figure 2.8: $L_{0\text{-}1}$ results for simulation setting 3. The x-axis includes the four different combinations of p and r.

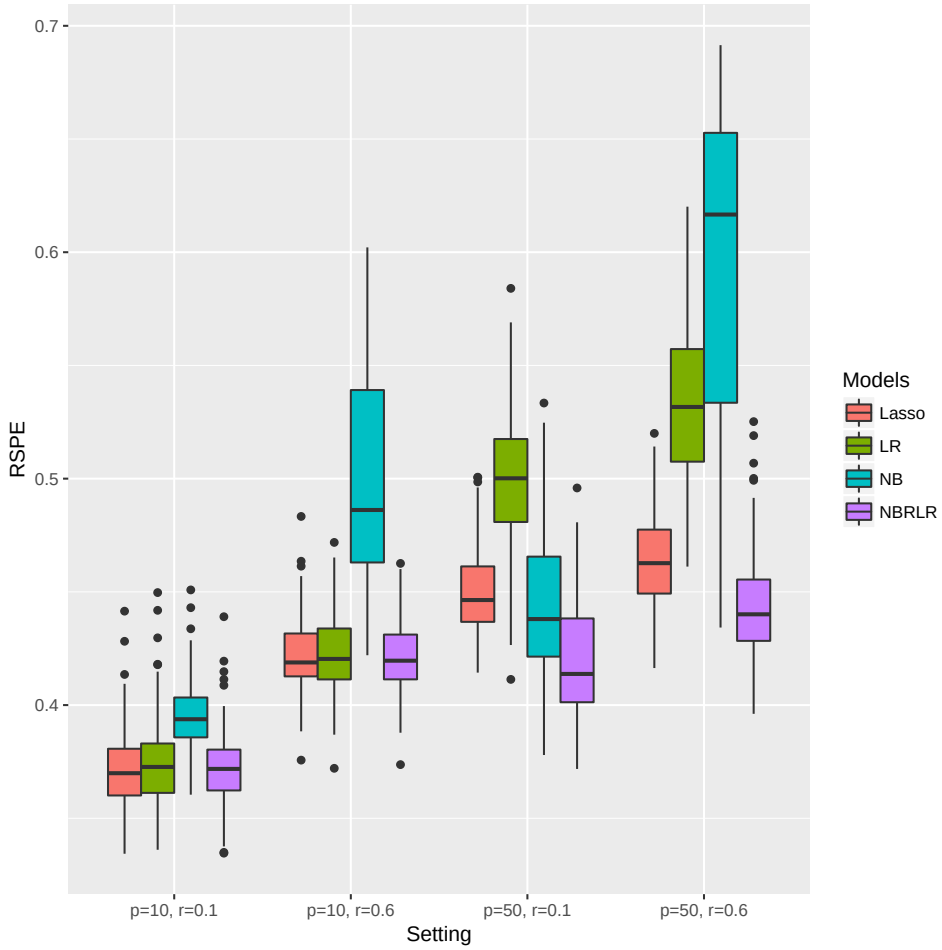

Figure 2.9: *RSPE* results for simulation setting 3. The x-axis includes the four different combinations of p and r.

2.4.4 Bias and Variance Analysis

Our proposed method follows traditional bias-variance tradeoff strategy. To provide valuable insight into the components of the error of the classifiers, we discuss the squared bias and variance of the four methods we compared. Let $\hat{\beta}_k = (\hat{\beta}_{0,n}, \ldots, \hat{\beta}_{p,n}) \in \mathbb{R}^{p+1}$ represent an estimator from the kth simulation and $\hat{\mathbf{B}} = (\hat{\beta}_1, \ldots, \hat{\beta}_{100}) \in \mathbb{R}^{p+1\times 100}$ represent the 100 estimators for a given method. The squared bias and variance of an estimator for a given simulation setting, with a true coefficient vector of β^*, is

$$Bias^2(\hat{\mathbf{B}}) = \frac{1}{p+1}\sum_{j=0}^{p}\left(\beta_j^* - \frac{1}{100}\sum_{n=1}^{100}\hat{\beta}_{j,n}\right)^2,$$

and

$$Var(\hat{\mathbf{B}}) = \frac{1}{p+1}\cdot\frac{1}{99}\sum_{j=0}^{p}\sum_{n=1}^{100}\left(\hat{\beta}_{j,n} - \frac{1}{100}\sum_{n=1}^{100}\hat{\beta}_{j,n}\right)^2.$$

Table 2.4 and 2.5 presents the results $Bias^2(\hat{\mathbf{B}})$ and the $Var(\hat{\mathbf{B}})$ of the 4 estimators given different simulation models, respectively. The results are mostly consistent with our intuition. The proposed NBRLR estimator has in general higher variance, but lower bias than NB and Lasso. In addition, NBRLR does better than LR in terms of both bias and variance, which may due to the convergence failures in LR, especially when p is large and n is relatively small.

2.5 Empirical Results

In this section, we evaluate the performance of our proposed NBRLR estimator on 12 different machine learning datasets from the UCI Machine Learning Repository. A summary of these datasets is given in Table 2.6, including the number of predictors, instances and the predictor type. The datasets are selected such that we have six datasets with categorical predictors, and six datasets with continuous predictors. For datasets with missing values, the missing values of categorical predictors are imputed with the conditional probability given the response variable, i.e. $P(x_{ij} \mid y_i)$. For continuous predictors with missing values, we assume they are conditionally normally dis-

Table 2.4: Squared bias of the four compared estimators at different numbers of predictors, p, and different levels of dependence among predictors, r, for the three different simulation settings.

Simulation Setting	p	r	NBRLR	LR	NB	Lasso
1	10	0.1	0.011	0.033	0.101	0.458
		0.6	0.020	0.067	0.300	0.596
	50	0.1	0.196	602.373	0.379	0.804
		0.6	0.205	4008.615	0.534	0.847
2	10	0.1	0.001	0.013	0.005	0.263
		0.6	0.032	0.120	0.225	0.535
	50	0.1	0.003	193.509	0.005	0.265
		0.6	0.116	1190.092	0.315	0.808
3	10	0.1	0.001	0.104	0.274	0.324
		0.6	0.002	0.065	0.500	0.336
	50	0.1	0.400	40.987	0.603	0.840
		0.6	0.332	195.665	0.727	0.800

tributed given the response variable, $x_{ij} \mid y_i = a \sim N(u_{ja}, \sigma^2_{ja})$. The missing values of continuous predictors are imputed with the corresponding conditional distribution given the response variable, i.e. $f(x_{ij} \mid y_i)$. The imputation procedure is conducted before we analyze the data.

We randomly divide each dataset into two parts, a training set with about 90% of the instances, and a test set with the remaining 10% of the instances. In the training set, the same methods used in Section 2.4 are used to fit the data. In the test set, we only compare the $L_{0\text{-}1}$ and *RSPE* of different estimators, as we do not assume to know the true β^* of the predictors. We repeat the experiments 100 times for each dataset. Similar to the simulations section we compare $L_{0\text{-}1}$, and *RSPE* of the different estimators using boxplots to show the results for each experiment. Again, we perform two-tailed, paired t-tests to compare NBRLR to the other three methods.

Table 2.5: Variance of the four compared estimators at different numbers of predictors, p, and different levels of dependence among predictors, r, for the three different simulation settings.

Simulation Setting	p	r	NBRLR	LR	NB	Lasso
1	10	0.1	0.317	0.421	0.193	0.118
		0.6	0.421	0.877	0.326	0.097
	50	0.1	0.554	3540.557	0.272	0.057
		0.6	0.623	21575.680	0.896	0.052
2	10	0.1	0.218	0.332	0.188	0.068
		0.6	0.343	1.505	0.267	0.157
	50	0.1	0.337	2679.698	0.272	0.093
		0.6	0.591	8392.322	0.720	0.223
3	10	0.1	0.208	0.363	0.057	0.064
		0.6	0.283	0.364	0.049	0.094
	50	0.1	0.143	89.836	0.047	0.021
		0.6	0.291	592.785	0.058	0.045

2.5.1 Categorical Datasets

Models are fit to the training data using NBRLR, as outlined in Section 2.3, and the other three methods used in Section 4. Figures 2.10–2.11 report the $L_{0\text{-}1}$, and *RSPE*, respectively, for the different methods on the different data sets. The y-axis of Figure 2.11 is on a log-10 scale to better present the difference between estimators. In terms of $L_{0\text{-}1}$, NBRLR performs the best on Qualitative Bankruptcy, Blogger, and Tic-Tac-Toe Endgame. However, it does worse than both LR and Lasso on SPECT Heart and Congressional Voting Records. All four estimators perform the same with perfect classification on Balloons. In terms of *RSPE*, NBRLR performs the best on Blogger. It does worse than LR on Balloons and Qualitative Bankruptcy, and worse than Lasso on SPECT Heart, Tic-Tac-Toe Endgame and Congressional Voting Records.

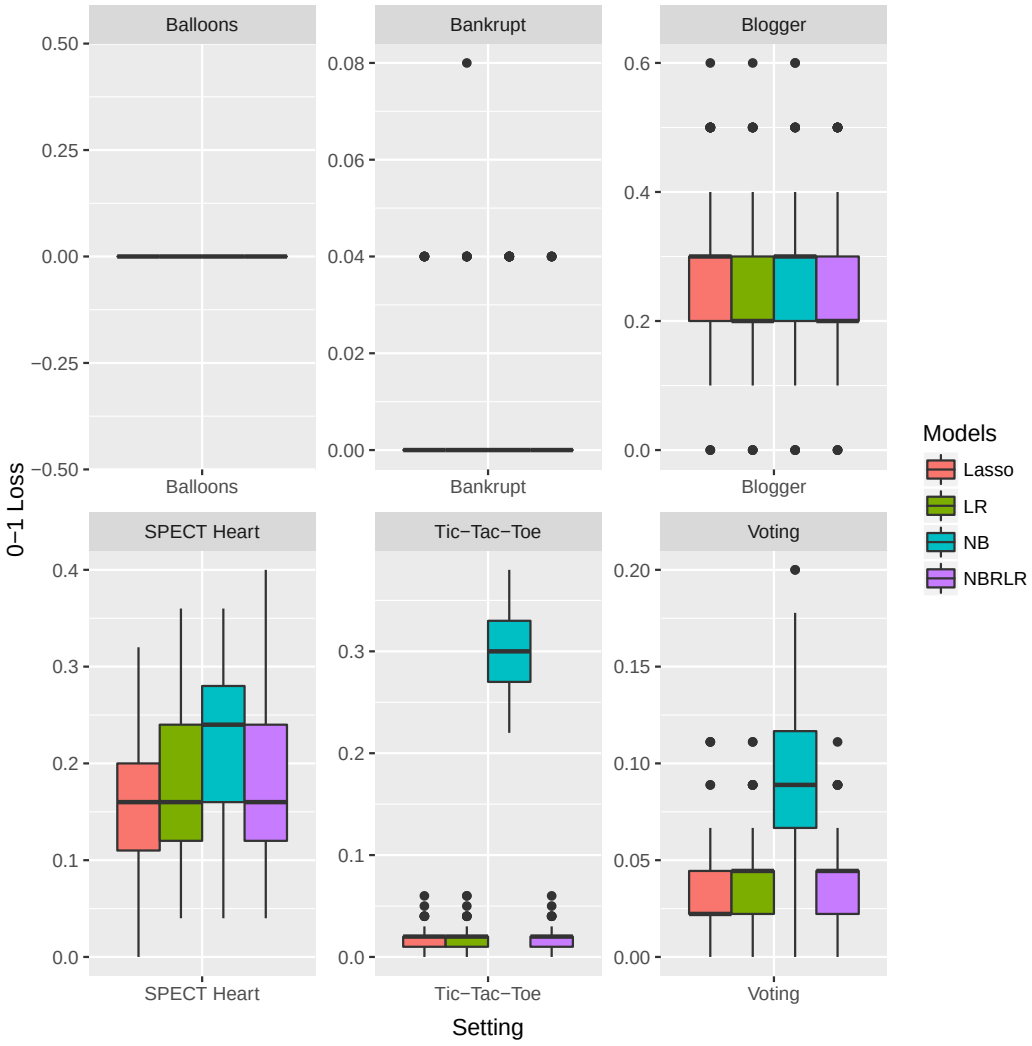

Figure 2.10: $L_{0\text{-}1}$ from the 100 experiments for the six categorical datasets.

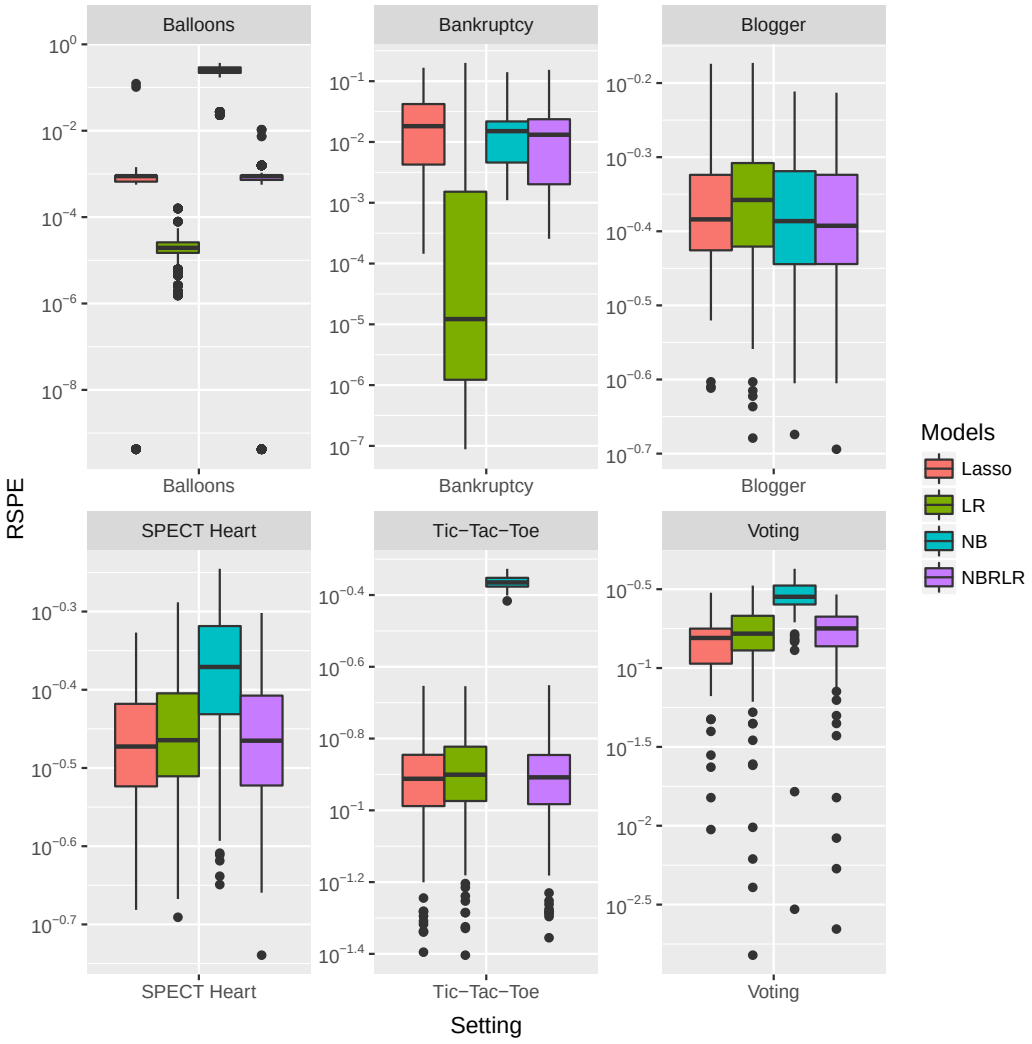

Figure 2.11: *RSPE* from the 100 experiments for the six categorical datasets.

Table 2.6: A Summary of the 12 datasets used in the empirical results. The Type column indicates if the predictors are categorical or continuous. Instances is the number of observations in the data set.

Dataset	*# Predictors*	*# Instances*	*# Type*
Balloons	4	20	Categorical
Qualitative Bankruptcy	6	250	Categorical
Blogger	5	100	Categorical
SPECT Heart	22	267	Categorical
Tic-Tac-Toe Endgame	9	958	Categorical
Congressional Voting Records	16	435	Categorical
Blood Transfusion Service Center	4	748	Continuous
Connectionist Bench	60	208	Continuous
Haberman's Survival	3	306	Continuous
Liver Disorders	6	345	Continuous
Pima Indians Diabetes	8	768	Continuous
Vertebral Column	6	310	Continuous

2.5.2 Continuous Datasets

For the continuous datasets the NBRLR estimator is fit using the approach outlined in Section 2.4, including assuming the conditional distribution of the predictors given the response class is normal. The four methods are compared using $L_{0\text{-}1}$ and *RSPE*, with results of these from the 100 experiments for each continuous dataset reported in Figures 2.12–2.13, respectively. Table 2.8 includes the averages of *RSPE* and $L_{0\text{-}1}$ for all six continuous datasets. In addition the table includes p-values from a two-sided paired t-test comparing NBRLR to the other three methods. The boxplot of *RSPE* is in a log-10 scale for the Y axis to better present the difference between estimators.

With respect to $L_{0\text{-}1}$, NBRLR achieves the best performance or comparable with the best performance on Blood Transfusion Service Center, Connectionist Bench, Haberman's Survival, and Pima Indians Diabetes. However, it does worse than both LR and Lasso on Vertebral Column. It also performs slightly worse than Lasso on Liver Disorders. In terms of *RSPE*, NBRLR performs

Table 2.7: Summary of empirical results for the six datasets with categorical predictors comparing NBRLR with pure LR, pure NB and lasso. The Esti. columns present the averages across the one hundred experiments. The p-values come from a two-sided, paired t-test comparing the performance of NBRLR with the corresponding estimator.

Categorical Datasets		NBRLR	LR		NB		Lasso	
		Esti.	Esti.	p-value	Esti.	p-value	Esti.	p-value
$L_{0\text{-}1}$	Balloons	0.000	0.000	1.000	0.000	1.000	0.000	1.000
	Qualitative Bankruptcy	0.002	0.004	0.158	0.006	0.001	0.003	0.320
	Blogger	0.254	0.254	1.000	0.262	0.011	0.265	0.016
	SPECT Heart	0.178	0.174	0.436	0.213	<0.001	0.152	<0.001
	Tic-Tac-Toe Endgame	0.017	0.018	0.049	0.302	<0.001	0.017	0.320
	Congressional Voting Records	0.040	0.038	0.356	0.094	<0.001	0.034	0.003
RSPE	Balloons	0.001	0.000	<0.001	0.250	<0.001	0.001	0.337
	Qualitative Bankruptcy	0.024	0.018	0.045	0.029	<0.001	0.033	<0.001
	Blogger	0.415	0.434	<0.001	0.416	0.570	0.420	0.082
	SPECT Heart	0.345	0.350	0.008	0.418	<0.001	0.340	0.018
	Tic-Tac-Toe Endgame	0.122	0.125	<0.001	0.432	<0.001	0.121	<0.001
	Congressional Voting Records	0.168	0.168	0.994	0.285	<0.001	0.006	<0.001

the best or comparable with the best on Connectionist Bench, Pima Indians Diabetes, and Vertebral Column. It does worse than LR on Blood Transfusion Service Center and Liver Disorders, worse than NB on Haberman's Survival, and worse than Lasso on Blood Transfusion Service Center, and Liver Disorders.

2.6 Conclusion

In this paper, we present a naïve Bayes regularized logistic regression model for classification problems. As LR is a low–bias, high–variance classifier, many regularized methods have been proposed to overcome LR's overfitting issue, which may leads to the poor prediction performance when the training sample is limited, or there is a large number of parameters to be estimated. Most of these methods assume that the true coefficients of LR are sparse, however, this sparsity assumption is often violated when p is relatively small compared to n, which makes the regression estimates suboptimal. Meanwhile, there also tends to be less multicollinearity among predictors. This limits the benefits of ridge regression, which is not motivated by the sparsity. We argue

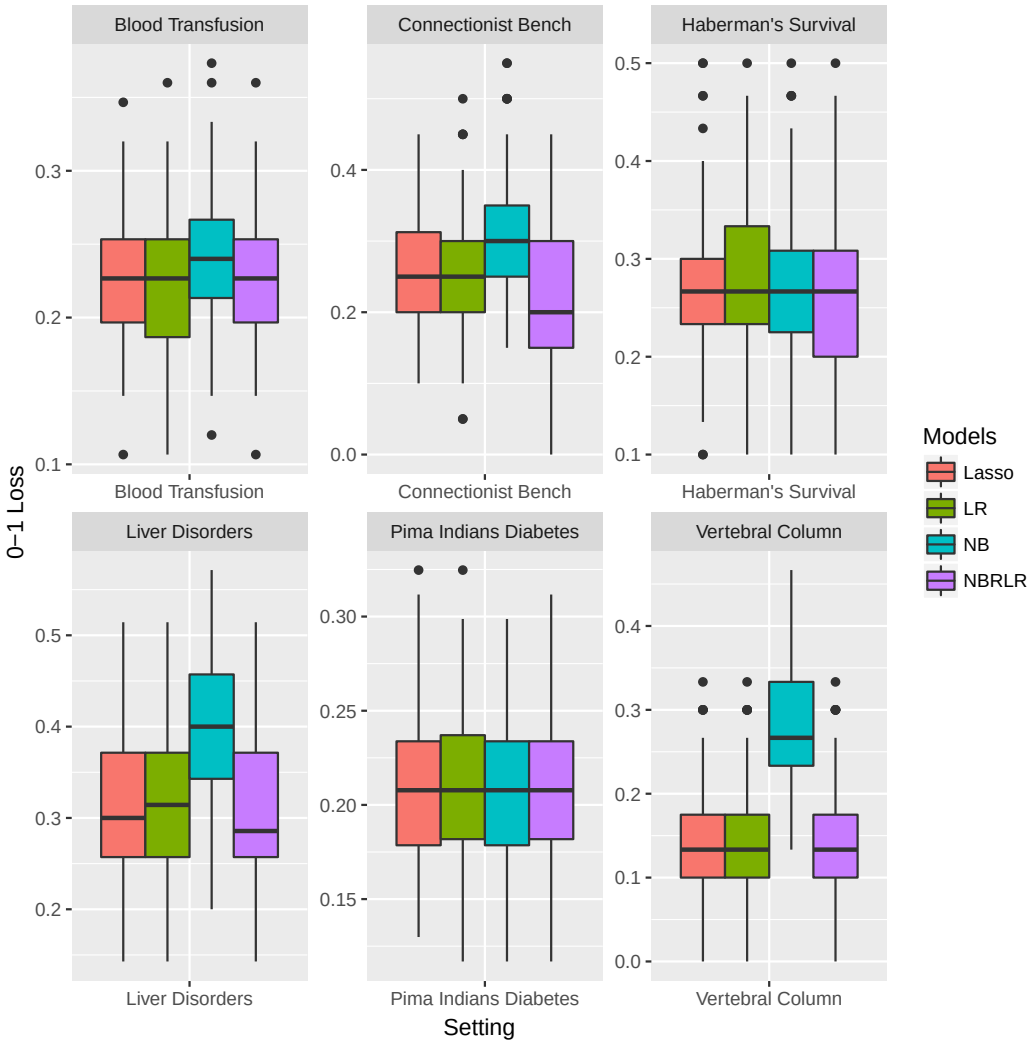

Figure 2.12: $L_{0\text{-}1}$ from the 100 experiments for the six continuous datasets.

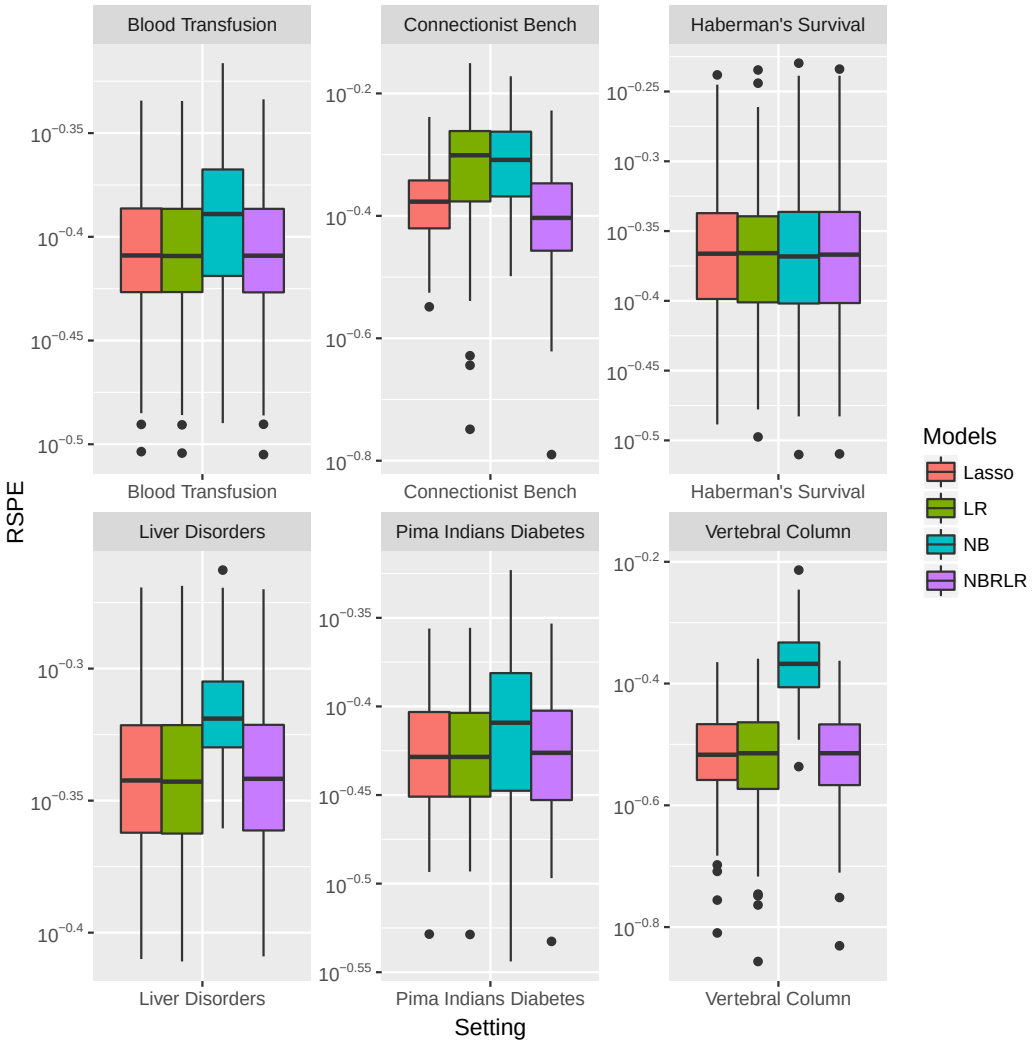

Figure 2.13: *RSPE* from the 100 experiments for the six continuous datasets.

Table 2.8: Summary of empirical results for the six datasets with continuous predictors comparing NBRLR with pure LR, pure NB and lasso. The Esti. columns present the averages across the one hundred experiments. The p-values come from a two-sided, paired t-test comparing the performance of NBRLR with the corresponding estimator.

Continuous Datasets		NBRLR	LR		NB		Lasso	
		Esti.	Esti.	p-value	Esti.	p-value	Esti.	p-value
$L_{0\text{-}1}$	Blood Transfusion Service Center	0.228	0.228	0.657	0.238	<0.001	0.228	0.470
	Connectionist Bench	0.227	0.250	0.010	0.301	<0.001	0.262	<0.001
	Haberman's Survival	0.268	0.272	0.007	0.268	1.000	0.271	0.118
	Liver Disorders	0.306	0.305	0.765	0.394	<0.001	0.305	0.482
	Pima Indians Diabetes	0.208	0.209	0.140	0.209	0.733	0.209	0.664
	Vertebral Column	0.146	0.145	0.181	0.274	<0.001	0.145	0.482
RSPE	Blood Transfusion Service Center	0.391	0.391	0.007	0.406	<0.001	0.391	0.232
	Connectionist Bench	0.396	0.484	<0.001	0.494	<0.001	0.418	<0.001
	Haberman's Survival	0.432	0.432	0.545	0.432	0.356	0.432	0.725
	Liver Disorders	0.457	0.456	<0.001	0.481	<0.001	0.457	<0.001
	Pima Indians Diabetes	0.374	0.373	0.712	0.387	<0.001	0.374	0.547
	Vertebral Column	0.306	0.306	0.384	0.432	<0.001	0.307	0.012

that when p is relatively small compared to n, shrinking the coefficients towards a low–variance data–driven estimate could be a better regularization strategy.

Our approach is primarily motivated by the fact that NB has an equivalent functional form compared to LR given NB's conditional independence assumption holds. The resulting classifier tends to have higher variance but lower bias as compared to Lasso, when p is relatively small compared to n. Simulation and empirical experimental results suggest that, NBRLR can generally outperform both LR and NB, and is highly competitive with Lasso on low and moderate dimension datasets. We present how to use the method for the cases of categorical predictors and continuous predictors.

Chapter 3

A New Bayesian Network-Based Approach for PTSD Detection

Abstract

Prediction of post-traumatic stress disorder (PTSD) has gained great interest in clinical studies. It can not only provide key guidance for making personal mental healthcare decisions, but also help identify high-risk PTSD population. This paper aims to address the challenge of providing veterans with timely healthcare access by improving VA PTSD diagnostic process with a diversion strategy. Specifically, we propose a sparsity-enforcing l_1 penalized Bayesian network-based model to measure the veterans' risk of suffering from PTSD based on easily available information. This will allow VA to send high-risk patients to the mental health provider directly. Experimental results show that our proposed model exhibits better out-of-sample prediction power as compared with a variety of state-of-art probabilistic classifiers. We also identify eight variables which provide the most directly predictive power.

3.1 Introduction

Post-traumatic stress disorder (PTSD) is a prevalent and seriously impairing disorder, especially for veterans. Prediction of PTSD is a research domain which has attracted great attention in the last two decades. Many studies focused on seeking for the risk factors of PTSD [3, 5, 24], which can not only provide key guidance for making personal mental healthcare decisions, but also be a great help for government or other healthcare organizations to identify high-risk population of PTSD. In recent years, machine learning techniques have been applied on PTSD prediction to fill in the gap between scientific discovery of risk factors for PTSD, and practical application in

making accurate prediction of PTSD in individuals. Commonly used methods include support vector machine (SVM) [22, 23, 34], random forest [50], logistic regression [31], and naive Bayes [41].

While the experience of combat and military sexual trauma are significant risk factors, veterans are associated with high risk for the development of PTSD. As there is a dramatic increase in the number of veterans seeking help for PTSD in last few decades, the Department of Veterans Affairs (VA) has been facing with the significant challenges in providing veterans with timely access due to the lack of availability of mental health providers. Patients diversion is one of the most widely adopted strategies to improve healthcare efficiency, with the purpose of making high-risk patients exposed to healthcare resources earlier. Then the goal of our study is to accurately measure the veterans' risk of suffering from PTSD based on easily available information.

In this paper, we introduce a novel, generic, scalable network based method for veterans' PTSD prediction. We identify three challenges that occur during our study: that is, the proposed model should be a probabilistic classifier, tolerant of a large amount of missing data, and the network construction algorithm should be efficient under the context of high dimensional search space. Specifically, we propose a Bayesian network model with the conditional probability distribution of each node identified with multivariate logistic regression. l_1 group penalization is applied in order to learn a sparser network structure which makes the model estimation more stable under the context of large amount of missing data. We test the validity of our approach on a real data set obtained from VA Informatics and Computing Infrastructure (VINCI), which is a Health Services Research & Development (HSR&D) Resource Center that provides researchers with a nationwide view of detailed VA patients data. Out-of-sample prediction results indicate that the proposed model provides a better measure of the risk of suffering from PTSD for veterans, as compared with a variety baseline probabilistic classifiers. Our study also contribute to the literature by identifying a group of features which are in the Markov boundary of *PTSD* for the proposed Bayesian network model. Such features tend to provide the most directly predictive power of *PTSD* given the population of veterans.

This chapter is organized as follows: Section 2 motivates our study by introducing the background of PTSD among veterans, current VA diagnostic process, and how diversion strategy may help to improve the process efficiency and to reduce waiting time. We also identify three challenges we face in this study while providing an accurate measure of the veterans' risk of suffering from PTSD. Section 3 describes our data and variables. In Section 4, we describe our method and present the algorithm we use to construct the model. Section 5 includes empirical results based on a real data set obtained from VINCI. Finally, in Section 6, we summarize and conclude.

3.2 Veteran PTSD Diagnostic Process

3.2.1 Background

Post-traumatic stress disorder (PTSD), occurs in persons who have experienced or witnessed a traumatic event. It has been recognized as one of the most disabling psychopathological conditions affecting the U.S. veteran population. Veterans are more exposed to life-threatening events, including combat or military exposure, terrorist attacks, and military sexual trauma, thus have a much higher prevalence of PTSD than non-veterans. As per the National Center for PTSD, the diagnosed number of PTSD in veterans varies by service era: it is between 11–20% for Operation Iraqi Freedom and Enduring Freedom, about 12% for Gulf War, and about 15% for Vietnam War. These numbers are all significantly higher than that of U.S. civilians (about 7–8%). A critical review of prevalence estimates of combat-related PTSD among veterans also suggests that the lifetime PTSD prevalence for veterans is 10–30% [46].

The U.S. government provides a wide range of benefits, including cash payments and VA-sponsored services, for veterans with disabilities that are the result of a disease or injury incurred or aggravated during active military service. The VA PTSD claims and benefits paid-out have been increasing rapidly over the recent decades. During 1999–2004, the number of veterans receiving VA disability payments for PTSD increased by 79.5%, as compared to 12.2% for other disabilities [20]. By 2016, PTSD has grown to be the third most compensated disability from all wars [8].

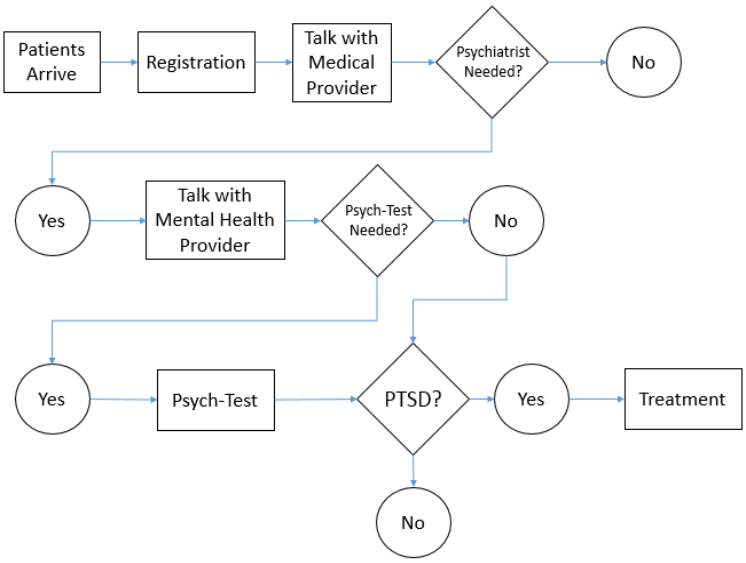

Figure 3.1: Traditional process for PTSD diagnosis

A current diagnosis from a VA hospital is one of the requirements that make up a claim for PTSD. Thus the increase of VA PTSD claims indicates a growing population of veterans seeking PTSD diagnosis and treatment, which combined with an acute shortage of mental health provider (MHP), has led to months-long waiting times. A VA audit (2014) show that the average waiting time for 51% of PTSD veterans is 50 days. The lack of provider appointment availability has become the largest barrier of providing timely and effective PTSD treatment for VA hospital.

3.2.2 PTSD Diagnostic Process

Fig. 3.1 illustrates the common PTSD diagnostic process for VA hospital. The process starts with the registration through which VA staffs capture patients' demographic information, military history information, and previous healthcare records. It follows with the appointment of medical providers. Medical providers work as general practice physicians. When they recognize the symptoms of PTSD of a patient, they make a referral to mental health providers. Mental health providers are professionals who diagnose PTSD and provide treatment. Besides, some of the patients are offered the psychological tests to confirm the diagnosis and help develop the treatment plan.

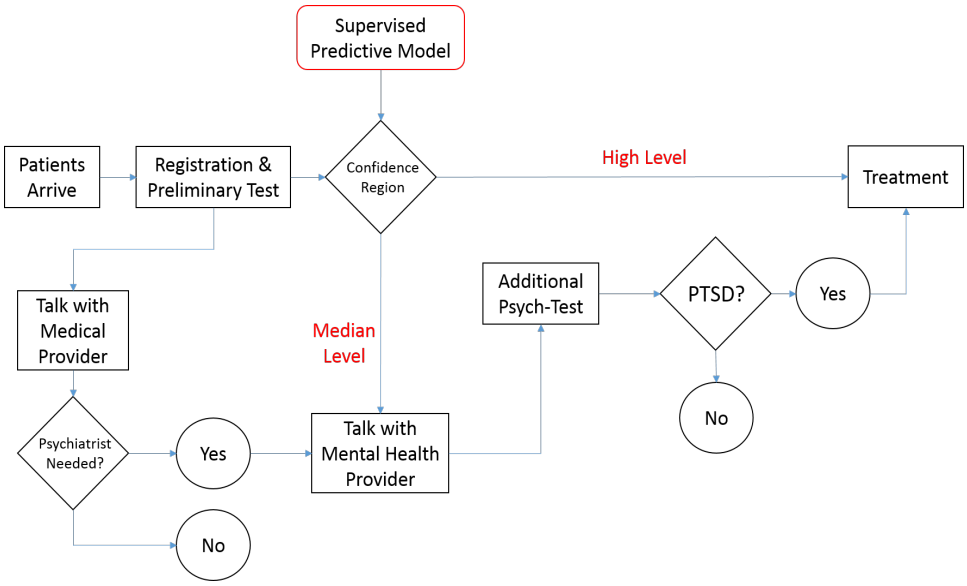

Figure 3.2: Proposed process for PTSD diagnosis

Different diversion strategies have been widely adopted to improve process efficiency and reduce waiting time in healthcare settings. The general purpose of diversion is to make patients exposed to healthcare resources earlier. In this work, we propose a proactive protocol which utilizes patients' personal information to identify those with high-risk of suffering PTSD. Same as the common process, patients need to make an appointment with medical providers after the registration. In the meantime however, high-risk patients are sent to the mental health provider directly or even skip to the treatment stage (in this situation, mental health provider will try to confirm the PTSD diagnosis during the treatment), instead of waiting for the referrals from medical providers. In addition, we suggest having a preliminary psychological test ordered at the registration stage to improve both the accuracy of our veterans' PTSD risk measurement and the efficiency of diagnostic process. The details of proposed PTSD diagnostic process is shown in Fig. 3.2.

3.2.3 PC-PTSD-5

The Primary Care PTSD Screen for Diagnostic and Statistical Manual of Mental Disorders 5 (PC-PTSD-5) [44] was designed to identify individuals with probable PTSD. The PC-PTSD-5 was used by VA starting from 2015, and has demonstrated strong results for PTSD diagnostic accu-

racy. The PC-PTSD-5 is a required screening instrument for *all* veterans. Thus, it exhibit less sample selection bias that those who have taken the PC-PTSD-5 are more likely to have PTSD. More importantly, the PC-PTSD-5 is a 5-item screen and very easy to understand, such that the participants would feel comfortable completing it. This makes the PC-PTSD-5 more acceptable as a preliminary test at the registration stage in our proposed PTSD diagnostic process.

3.2.4 Challenges

At the heart of this proposed PTSD diagnostic process is a supervised predictive model that tries to bridge the gap between the existing academic/clinical knowledge about PTSD, and veteran personal level PTSD diagnostics. We identify three challenges presented in veteran PTSD detection problem and discuss how each challenge can be addressed.

Challenge 1: Probabilistic classification. The proposed model should be a probabilistic classifier that is able to predict the probability distribution over a set of classes. In this study, such probability can interpreted as a measure of veterans' risk of suffering from PTSD given existing information, which is the basis of how we divert patients. Support vector machine (SVM) [22,23,34] has been the most popular machine learning technique to improve the prediction of PTSD. However, it is a deterministic approach which has no notion of probability involved, but simply returns the class (PTSD & non-PTSD) of the patients. Treating such discrete classes as the measure of risk level is too arbitrary. Other commonly used techniques include random forest [50], logistic regression [31], and naïve Bayes [41].

In this paper, we explore the use of Bayesian network, a method of probabilistic graphical model to predict the likelihood of visiting patients suffering from PTSD.

Challenge 2: Large amount of missing data. Our proposed model need to deal with extremely large amount of missing data, especially in terms of historical military information. For example, as shown in Table 3.4, the proportion of missing data is 94.88% for both southwest asian flag and combat flag. This makes statistical estimates unstable and reduces the predictive power, especially of complex models.

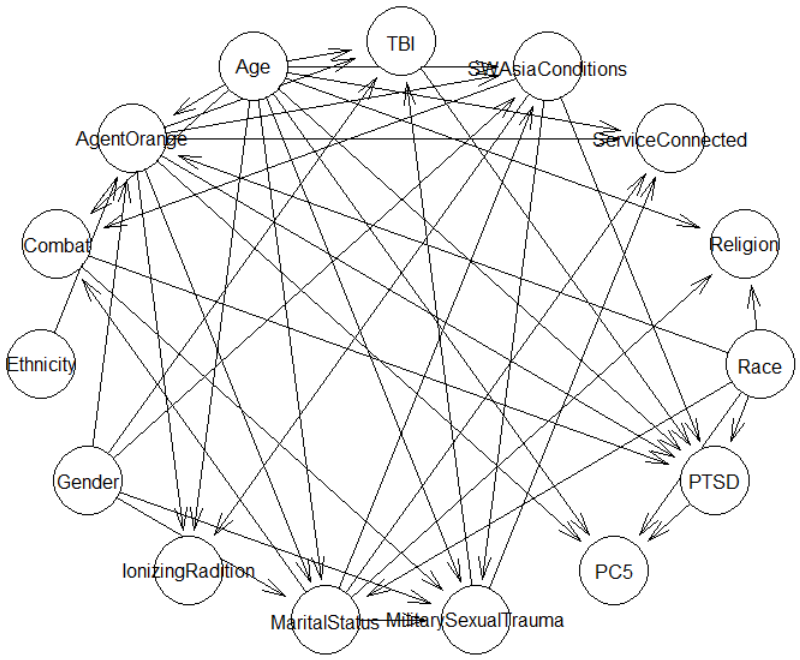

Figure 3.3: The structure for Bayesian network model constructed using score-based technique (BIC)

Figure 3.3 represents the regular Bayesian network structure constructed using a score-based technique (Bayes information criterion, BIC). Missing data are imputed with the conditional probability given veterans' PTSD diagnostic status. Notice that, this Bayesian network model has a very dense structure, which leads to a large number of parameters to be estimated. For the common situations, this should not be a big concern because our data set is quite rich. However, the estimation of conditional probability may not be reliable when the corresponding observed training instances is limited. For example, the number of observations with Southwest Asian Conditions flag, Agent Orange flag, and TBI diagnosis is 0 for young and old veterans, and is 9 for middle aged veterans.

In this study, we adopt a sparsity-enforcing l_1-regularized Bayesian network learning algorithm to reduce the model complexity. Missing data are addressed using multiple imputation (MI).

Challenge 3: Large search space of network structure. Learning a Bayesian network structure from data has been known to be an NP-hard problem, because that the network structure has to be a directed acyclic graph (DAG). As our task is to construct a Bayesian network with 15 nodes from more than 1 million instances, many of the commonly used methods are computationally expensive, and thus hard to implement. In this study, we adopt an ordering-based search strategy, and integrate it with significant domain knowledge, to improve the efficiency of our structure learning algorithm.

3.3 Data

We obtain data from VA Informatics and Computing Infrastructure (VINCI), which is a Health Services Research & Development (HSR&D) Resource Center that provides researchers with a nationwide view of detailed VA patients data. We start by searching for veterans who have ever taken the PC-PTSD-5 test by the end of 2019. Veterans' personal level information is obtained by aggregating across their lifetime visits to the VA hospital. We exclude veterans who reached the age of over 120 in the database, which is very unlikely and is probably misrecorded. Finally, our search identifies 1,113,676 distinct veterans.

3.3.1 Variable Definition and Miscellaneous Issues

This section introduces the variables that we use for constructing the Bayesian network model, and explains the issues related to how we collect the data.

Multiple descriptive epidemiologic studies have been conducted to examine the patterns of PTSD in response to a range of demographic factors [1, 19, 24, 43, 51]. Following the literature, we start by collecting the veterans' information of *Age*, *Gender*, *Marital Status*, *Ethnicity*, and *Race*. We also include the information of veterans' military experience to explain the variation

in veterans' risk of suffering PTSD. As we discussed in Section 3.2 that the prevalence of veterans with PTSD varies by service era, such military experience information can be an indicator of veterans' combat situation. Specifically, we include whether a veteran has ever attended a combat (*CombatFlag*), which combat he/she has attended (*AgentOrangeFlag*, *IonizingRadiationFlag*, *SWAsiaConditionsFlag*), and the trauma types that veterans have experienced (*MilitarySexualTraumaFlag*, *ServiceConnectedFlag*). These indicator variables will take the value of one if the veteran has ever answered "yes" in response of corresponding questions during his/her life time visits to the VA, and zero if he/she always say "no". Next, we collect the lifetime PC-PTSD-5 test results. For veterans who have taken the test multiple times, we take the average of all the scores. Finally, we investigate veterans' historical diagnosis of PTSD which is our variable of interest, and TBI which has been widely recognized as a cause of PTSD. The variables will take the value of one if the veteran has ever been diagnosed with PTSD or TBI, respectively, and zero otherwise.

Other data collecting issues are discussed below.

- Inconsistent Records - For demographic information, veterans may provide inconsistent responses during different visit to the VA. In such situation, we treat the corresponding variables as missing.

- Numeric Variables - To handle numeric variables, i.e., *Age* and *PC-PTSD-5*, we discretized the variables using supervised discretization with decision tree model. Specifically, we train a decision tree using the age/PC-PTSD-5 to predict PTSD. As a result, *Age* is discretized into three categories of young ($<$52), middle (52-76), old ($>$76), and *PC-PTSD-5* is discretized into three categories of low ($<$0.04545), median (0.04545-1.0625), high ($>$1.0625).

3.3.2 Summary Statistics

Table 3.1 provides the summary statistics for our key variables. Given the sample of 1,113,676 veterans, 23.09% of them have ever been diagnosed with PTSD and only 2.73% of them have ever been diagnosed with traumatic brain injury (TBI). Most of the veterans are male (91.82%), white

Table 3.1: Summary statistics for our key variables. We report the category values with corresponding frequencies and proportions. The category of 'NA' stands for missing data.

Variables	Values	Frequencies	Proportions (in %)
Age	Young	227641	20.44
	Middle	682393	61.27
	Old	203642	18.29
Gender	F	91098	8.18
	M	1022578	91.82
MaritalStatus	Divorced	350143	31.44
	Married	605050	54.33
	Single	152772	13.72
	NA	5711	0.51
Ethnicity	Hispanic/Latino	72362	6.50
	Non-Hispanic/Latino	1002472	90.01
	NA	38842	3.49
Race	Black	209196	18.78
	Other	85398	7.67
	White	794939	71.38
	NA	24143	2.17
Religion	Christian	801801	72.00
	Other	51382	4.61
	NA	260493	23.39
CombatFlag	N	15582	1.40
	Y	126059	11.32
	NA	972035	87.28

Variables	Values	Frequencies	Proportions (in %)
AgentOrange	N	290080	26.05
	Y	120519	10.82
	NA	703077	63.13
IonizingRadiation	N	308323	27.69
	Y	2229	0.20
	NA	803124	72.11
SWAsiaConditions	N	29496	2.65
	Y	27552	2.47
	NA	1056628	94.88
MilitarySexualTrauma	N	17461	1.57
	Y	39517	3.55
	NA	1056698	94.88
ServiceConnected	N	32027	2.88
	Y	644294	57.85
	NA	437355	39.27
PC-PTSD-5	Low	819633	73.60
	Median	63945	5.74
	High	230098	20.66
TBI	N	1083282	97.27
	Y	30394	2.73
PTSD	N	856552	76.91
	Y	257124	23.09

(71.38%), married (54.33%), not hispanic or latino (90.01%), and at the age of 52-76 (61.27%). It is worth noting that there is large amount of missing data for military experience related variables. For example, the proportion of missing data is 87.28% for combat flag, 94.88% for southwest asian flag, and 94.88% for military sexual trauma flag.

We present the prevalence of PTSD for our sample with respect to different categories of other variables in Figure 3.4. The bar-plots show that young, female veterans are more likely to suffer from PTSD. Also, attending a combat, experiencing military sexual trauma and service connected trauma will increase the risk of PTSD. Finally, high score of PC-PTSD-5 test is a strong indicator of PTSD.

3.4 Model

Our proposed model is based on sparsity-enforcing l_1-regularized Bayesian network, incorporating Challenge 1-3 described in Section 3.2. A Bayesian network is a directed acyclic graphical model with a set of m nodes $\{X_1, \ldots, X_m\}$. We define $Pa(X_j)$ as the vector of parents of a

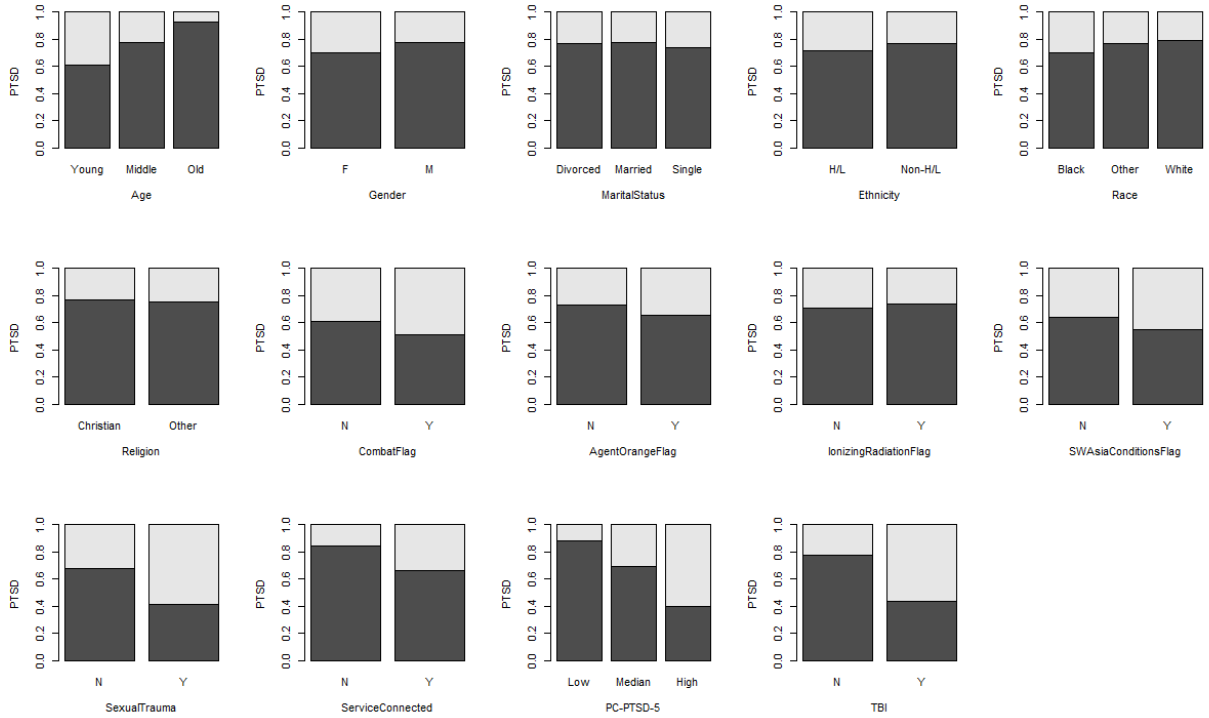

Figure 3.4: Prevalence of PTSD with respect to different categories of other variables.

node X_j, i.e., there is a directed arc from X_i to X_j if $X_i \in Pa(X_j)$. Given categorical data, we use a multivariate logistic regression for the conditional probability distribution of each node $P(X_j = x_{j,k} \mid Pa(X_j), \beta_{j,k}) = \frac{\exp(Pa(X_j)'\beta_{j,k})}{\sum_k \exp(Pa(X_j)'\beta_{j,k})}$, where $\beta_{j,k} = (\beta_{j,k,0}, \beta_{j,k,1}^T, \ldots, \beta_{j,k,m}^T)^T$ is the vector of unknown parameters to be estimated from data. Our task becomes obtaining a sparse estimate of $\beta_{j,k,i}$'s where $X_i \in Pa(X_j)$, under the constraint that the estimated Bayesian network structure G must be a directed acyclic graph (DAG). The nonzero values of $\beta_{j,k,i}$'s indicate the presence of edges in the structure G.

To tackle the challenge of structural learning of Bayesian network from high-dimensional data, Huang *et al.* [32] proposed a Sparse Bayesian Network (SBN) structure learning algorithm. Given fully observed data $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_m]$, where $\mathbf{x}_j$ is a vector of N observations for node X_j, the estimate of $\beta_{j,k,i}$'s is obtained by minimizing the negative log-likelihood of data with the sparsity enforcing l_1 penalty as:

$$\min_{\beta_j} \frac{1}{N} \sum_{j=1}^{m} NLL(\mathbf{x}_j, \mathbf{x}_{-j}, \beta_j) + \lambda \sum_{j=1}^{m} ||\beta_j||_1 \quad s.t. \quad G \in DAG, \tag{3.1}$$

where X_{-j} is the set of all nodes excluding X_j by assuming all of these nodes are candidate parents of node X_j, and $NLL(\mathbf{x}_j, \mathbf{x}_{-j}, \beta_j)$ is the negative log-likelihood for node X_j. Given the estimate of β_j's, the set of parents for node X_j can be found as $Pa(X_j) = \{X_i \mid \beta_{j,.,i} \neq 0\}$. Tuning parameter λ determines the strength of regularization, and can be determined by out-of-sample prediction performance.

3.4.1 Multiple Imputation

As we discussed in Section 3.2, the statistical analysis of the veterans PTSD likelihood and its influencing factors is hindered by the missing data. According to the VA psychologists, this was due largely to the item nonresponse in registration questionnaire as most of the questions are optional for veterans. The nonresponse rate is incredibly high especially for military experience information. Thus, ignoring incomplete cases may lead to significant information loss, and our statistical inference can be biased if the data is not missing completely at random. EM algorithm [7] is another commonly used method for handling missing data. However, it becomes overwhelmed and increasingly impractical to use with millions of data points.

In this subsection, we adopt multiple imputation (MI) to address the problem of missing data. MI was first proposed by Rubin [48], and has been widely used in large-scale healthcare/medical studies. It has practical advantages of preserving sample size and statistical power, providing unbiased parameter estimates, and allowing standard complete-data methods of analysis to be used. Essentially, MI is an iterative form of stochastic imputation which tries to the variability of missing data. MI has three basic steps:

1. Imputation: impute the missing entries D times. This step results in D complete datasets.
2. Analysis: analyze each of the D completed data sets.
3. Pooling: integrate the D analysis results into a final result.

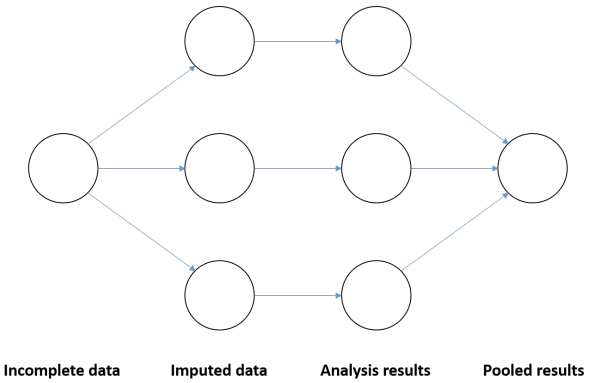

Figure 3.5: Multiple Imputation

When multi-level predictors and responses are present, lasso may not be satisfactory as it only selects individual dummy variables instead of whole factors, and the lasso solution also depend on how the dummies are encoded. Yuan and Lin [58] proposed grouped lasso to overcome these issues, while Chen and Wang [6] extended this idea to multiply-imputed data in order to select or remove the estimated regression coefficients associated with the same variable together across different imputed datasets. In this paper, we denote $\hat{\beta}_{j,1},\ldots,\hat{\beta}_{j,D}$ be the vectors of estimated regression coefficients for child node X_j on the D imputed datasets, where $\hat{\beta}_{j,d} = (\hat{\beta}^T_{j,0,d},\hat{\beta}^T_{j,1,d},\ldots,\hat{\beta}^T_{j,j-1,d},\hat{\beta}^T_{j,j+1,d},\ldots,\hat{\beta}^T_{j,m,d})^T$. Here, $\hat{\beta}_{j,0,d} \in \mathbb{R}^{df_{j0}}$ is the estimated vector of intercept with degree of freedom df_{j0}, and $\hat{\beta}_{j,i,d} \in \mathbb{R}^{df_{ji}}$ is the estimated vector of regression coefficient of parent node X_i with degree of freedom df_{ji}. Let's also define $\hat{\beta}_{j,i} = (\hat{\beta}^T_{j,i,1},\ldots,\hat{\beta}^T_{j,i,D})^T \in \mathbb{R}^{df_{ji}\cdot D}$. If X_i is important for predicting X_j, $\hat{\beta}_{j,i,d}$ should be all nonzero, and if X_i is not important for predicting X_j, $\hat{\beta}_{j,i,d}$ should be all zero for any given imputed dataset d. Thus, we obtain the estimate of $\beta_{j,i,d}$ by minimizing the following objective function:

$$\min_{\beta_{j,i,d}} \frac{1}{N\cdot m\cdot D}\sum_{j=1}^{m}\sum_{d=1}^{D} NLL(\mathbf{x}_j,\mathbf{x}_{-j},\beta_{j,d}) + \lambda\sum_{j=1}^{m}\sum_{i\neq j}\sqrt{p_{j,i}}||\beta_{j,i}||_2 \quad s.t. \quad G\in DAG. \tag{3.2}$$

Here, $||\beta_{j,i}||_2 = \sqrt{\sum_{d=1}^{D}\sum_{k=1}^{df_{ji}}\beta^2_{j,i,k,d}}$ is called the group lasso penalty where $\beta_{j,i,k,d}\in\beta_{j,i,d}$, and $p_{j,i} = df_{ji}\cdot D$ is the varying group size. The penalty function is adjusted by $\sqrt{p_{j,i}}$ to ensure that

the same degree of penalization is applied to large and small groups. The group LASSO penalty gaurantees the consistency of edges selection with respect to all different predictor levels, response levels, and imputed datasets.

3.4.2 Ordering-Based Search

Solving the optimization Eqs. (3.2) can be challenging given the constraint that the estimated Bayesian network structure G must be a DAG because of the huge search space of network structures. Much work has been done to address this problem, but only a few outperform the baseline of greedy hill-climbing with tabu lists. In this paper, we adopt ordering-based search strategy [54], and use greedy hill-climbing search, with a tabu list. Determining an appropriate ordering is a difficult problem, however, our strong causality based clinical knowledge helps significantly reduce our search space.

We conduct ordering-based search by seeking for the best ordering $\prec$ over $X_1, \ldots, X_m$, such that if X_i is a potential candidate for $Pa(X_j)$, then $X_i \prec X_j$. Once the ordering $\prec$ is determined, finding the optimal Bayesian network that consistent with $\prec$ is no longer NP-hard because we can easily implement regular grouped LASSO on each node separately. We use hill-climbing to find $\prec$, i.e., only consider swapping a pair of adjacent nodes in the ordering for each move until the value of objective function (3.2) does not decrease:

$$(\ldots, X_{i-1}, X_i, X_{i+1}, X_{i+2}, \ldots) \rightarrow (\ldots, X_{i-1}, X_{i+1}, X_i, X_{i+2}, \ldots)$$

As there are only two new neighborhood are generated: (X_{i-1}, X_{i+1}) and (X_i, X_{i+2}) for each move, we use tabu list to prevent the algorithm from reversing a swap that was executed recently in the search.

We use domain knowledge to reduce the search space of possible ordering $\prec$. Specifically, we divide the nodes into five layers, which is illustrated in Fig. (3.6), based on causality. For example, personal characteristics at the first layer are what people born with, thus can never be affected by the other nodes; while as suggested by VA psychologists, TBI is usually a cause of PTSD. If a

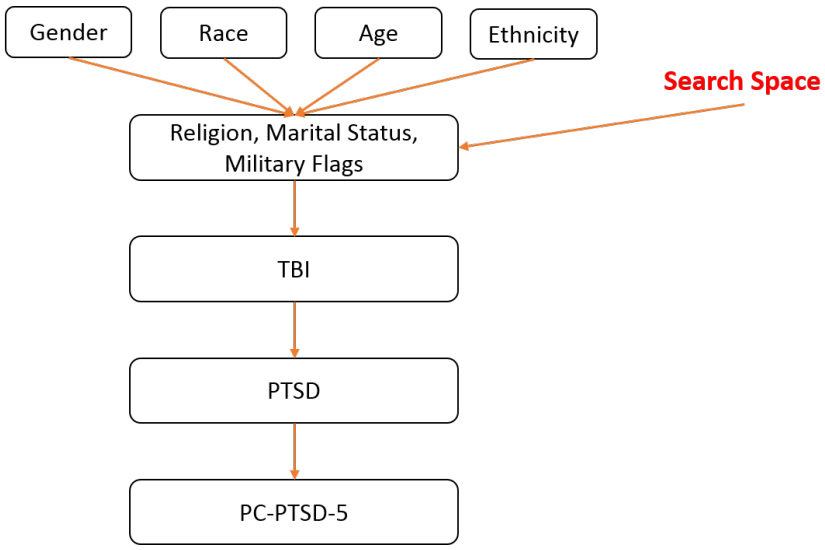

Figure 3.6: Ordering-based search

node X_i is at the upper layer of node X_j, then X_i should always precede X_j in $\prec$. In this way, we restrict our ordering search space only within the second layer.

With a predetermined ordering $\prec$, Eq. (3.2) can be transformed as:

$$\min_{\beta_{j,i,d}} \frac{1}{m} \sum_{j=1}^{m} [\frac{1}{N \cdot D} \sum_{d=1}^{D} NLL(\mathbf{x}_j, \mathbf{x}_{i \prec j}, \beta_{j,i,d}) + \lambda \sum_{i \prec j} \sqrt{p_{j,i}} ||\beta_{j,i}||_2] \tag{3.3}$$

that our task becomes solving m optimization problems independently. The grouped LASSO penalty function is singular at the origin point. We apply the local quadratic approximation proposed by Li and Fan [14] to overcome this issue. For each given node X_j, we minimize the objective function iteratively. Suppose the estimation at the t^{th} iteration is $\beta_{j,i,d}^{(t)}$, we can have the following approximation:

$$\sqrt{\Sigma_{d=1}^{D} \Sigma_{k=1}^{df_{ji}} \beta_{j,i,k,d}^2} \approx \frac{\Sigma_{d=1}^{D} \Sigma_{k=1}^{df_{ji}} \beta_{j,i,k,d}^2}{\sqrt{\Sigma_{d=1}^{D} \Sigma_{k=1}^{df_{ji}} \beta_{j,i,k,d}^{(t)2}}}.$$

Thus, the objective function 3.3 can be approximated by:

$$\min_{\beta_{j,i,d}} \frac{1}{m \cdot D} \sum_{j=1}^{m} \sum_{d=1}^{D} [\frac{1}{N} NLL(\mathbf{x}_j, \mathbf{x}_{i \prec j}, \beta_{j,i,d}) + \lambda \sum_{i \prec j} \sqrt{p_{j,i}} \frac{||\beta_{j,i}||_2^2}{||\beta_{j,i}^{(t)}||_2}], \tag{3.4}$$

such that the estimated coefficient $\hat{\beta}_{j,i,d}^{(t+1)}$ can be obtained by solving $m \cdot D$ separate weighted ridge

regression. We iterate the above step until the convergence is reached.

3.4.3 Model Construction

The previous paragraphs sketched out the basic idea of how we address challenge 1-3. In general, we learn a Bayesian network based model by minimizing the negative log-likelihood of data with sparsity enforcing l_1 penalty across multiple imputed dataset with respect to a pre-specified level of regularization. Given a fixed value of regularization parameter λ, a summary of our model training procedure is presented in Alg. 2.

Algorithm 2 Construction of Sparsity-enforcing Bayesian Network

Input: A set of veterans with a panel of unique predictive personal characteristics.
Output: A sparsity-enforcing Bayesian network model for PTSD prediction.

1: Impute the missing values of training set D times with a pre-specified model.
2: Start with pre-specified order $\prec_0 = (X_{0_1}, \ldots, X_{0_m})$
3: Obtain the initial estimate of $\hat{\beta}^{(0)}_{j,i,d}$ by implementing regular LASSO on each imputed dataset. Set $t = 0$.
4: Calculate $w_{j,i} = \sqrt{p_{j,i}}/||\beta^{(t)}_{j,i}||_2$ for each j, i.
5: Let $t = t + 1$. Solving Eq. 3.4 by conducting $m \cdot D$ separate ridge regression with weight $w_{j,i}$.
6: Iterate between Row 4 and Row 5 until the estimates converges.
7: Considering all successors (the search space is restricted as Fig. 3.6) of current $\prec$ by performing adjacent swap, and pick $\prec'$ by minimizing Eq. 3.3. Stop until Eq. 3.3 does not improve.
8: Integrate the estimation of $\hat{\beta}_{j,i,d}$ into a final result based on Rubin's rule: $\hat{\beta}_{j,i} = \frac{1}{D}\sum_{d=1}^{D}\hat{\beta}_{j,i,d}$.
9: end algorithm

3.5 Results

In this section, we discuss the results of our proposed sparsity-enforcing l_1 penalized Bayesian network-based model compared to the baselines. All models are evaluated in terms of the 0-1 loss ($L_{0\text{-}1}$) and mean squared error (MSE), which are defined as

$$L_{0\text{-}1} = 1 - \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}(PTSD_i = \hat{PTSD}_i),$$

and

$$MSE = \frac{1}{N}\sum_{i=1}^{N} \{\hat{P}(PTSD_i = 1) - \mathbf{1}(PTSD_i = 1)\}^2.$$

Here N is the testing sample size, $PTSD_i$ is the observed PTSD status for the ith testing observation, $\hat{PTSD}_i$ is the predicted PTSD status for testing observation i, $\hat{P}(PTSD_i = 1)$ is the predicted probability that the i^{th} testing observation is suffering from PTSD, and $\mathbf{1}()$ is an indicator function for the condition in the parenthesis.

We start by randomly dividing our sample into three parts, a training set with 913,676 observations, a test set with 100,000 observations, and a validation set with the remaining 100,000 observations. In the training set, the method described in Section 3.4 with pre-specified a sequence of lambda values is used to train the model. For variables with missing values, we impute them 5 times with the traditional Bayesian network model presented in Figure 3.3. As our goal is to measure veterans' risk of suffering PTSD, we compare the *MSE* of models trained with different values of lambda in the test set to determine λ value. Specifically, we follow the "one-standard-error" rule by selecting the largest λ value with its *MSE* within one standard error of λ_{min}. The main idea is to choose the simplest model whose accuracy is comparable with the best one. Finally, we compare the resulting model with regular Bayesian network, regularized logistic regression (lasso), and naïve Bayes in the validation set.

Figure 3.7 reports the *MSE* score for the proposed sparsity-enforcing l_1 penalized Bayesian network-based model with a pre-specified sequence values of penalty parameter λ. As we see, the model with $\lambda_{min} = 0.00005$ achieves the lowest *MSE* (0.1358659). The red dash line is one standard error above this lowest *MSE* value. Following the "one-standard-error" rule, we choose $\lambda_{1se} = 0.002$.

The structure of the resulting Bayesian network model with $\lambda = 0.002$ is presented in Figure 3.8. As compared with the regular Bayesian network model in Figure 3.3, our proposed model ex-

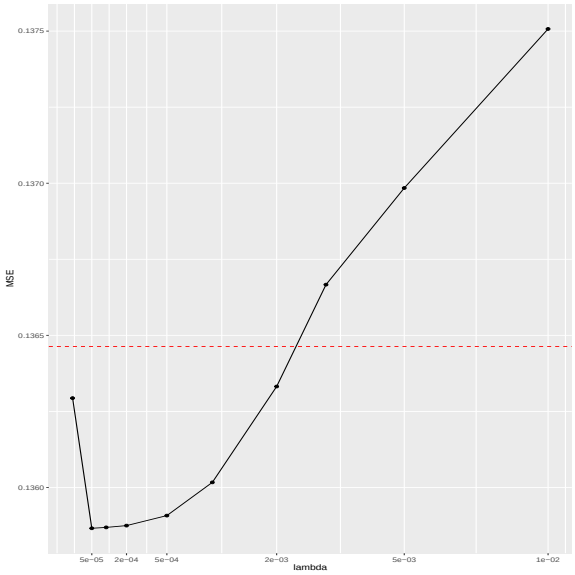

Figure 3.7: Searching for λ value

hibit sparser structure which is more tolerant with missing values. 8 variables (*Age*, *AgentOrange*, *MaritalStatus*, *MilitarySexualTrauma*, *Race*, *ServiceConnected*, *TBI*, and *PC5*) are in the Markov Boundary of *PTSD*. We recommend VA to mark the corresponding 8 questions as mandatory, as they provide the most direct predictive power. This will make our model more reliable, and the proposed patients diversion process more efficient.

Comparison of our model with regular Bayesian network, regularized logistic regression (lasso), and naïve Bayes in terms of $L_{0\text{-}1}$ and *MSE* are reported in Table 3.2. For regularized logistic regression, we do multiple imputation with group penalty to address the problem of missing data. The penalty parameter is determined by conducting training-testing validation following the "one-standard-error" rule with respect to out-of-sample *MSE*, same as what we did for sparsity-enforcing l_1 penalized Bayesian network-based model. The estimation of parameters of regular Bayesian network and naïve Bayes model is conducted using the Laplace correction [40] to prevent the high influence of zero probabilities. Specifically, we add one of each class to the data. The results show that our proposed model outperforms regular Bayesian network, regularized logistic regression and naïve Bayes model for both $L_{0\text{-}1}$ and *MSE*.

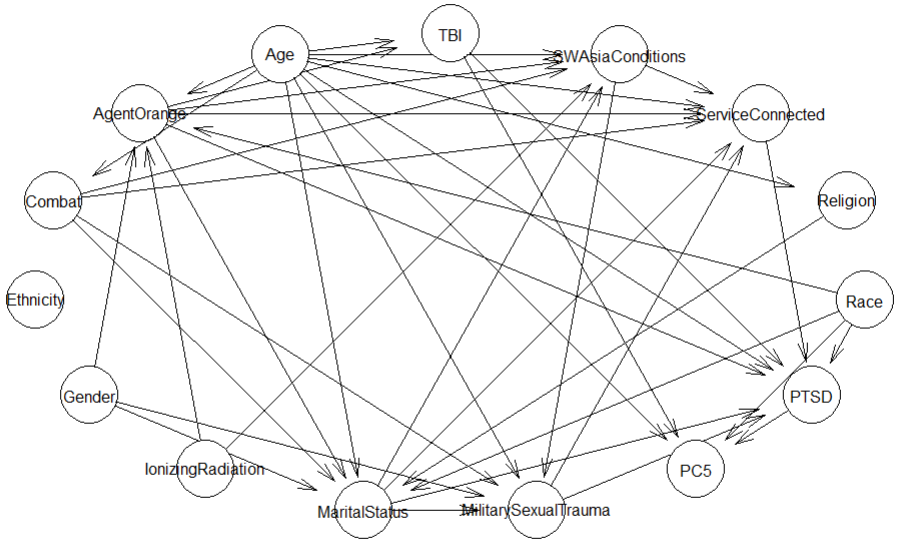

Figure 3.8: The structure for sparsity-enforcing l_1 penalized Bayesian network-based model at $\lambda = 0.002$.

Table 3.2: Summary of results comparing sparsity-enforcing l_1 penalized Bayesian network-based model with regular Bayesian network, regularized logistic regression (lasso), and naïve Bayes in terms of $L_{0\text{-}1}$ and MSE.

	l_1-BN	BN	Lasso	NB
$L_{0\text{-}1}$	0.1868	0.2128	0.1870	0.1881
MSE	0.1358	0.1804	0.1361	0.1386

3.6 Conclusion

In this paper, we improve the performance of traditional probabilistic classifier for PTSD detection by introducing a new Bayesian network based approach. We start by identifying three challenges presented in veteran PTSD detection problem: probabilistic classification, large amount of missing data, large search space of network structure, and thus our proposed model addresses these challenges accordingly. Particularly, we represent a Bayesian network model with the conditional probability distribution of each node defined with multivariate logistic regression. We add an l_1 penalty, which yields a sparser model, to make the model estimation more stable under the context of large amount of missing data. A ordering-based search algorithm with strong causality based clinical knowledge is adopted to search for the network structure. As a result, our proposed sparsity-enforcing l_1 penalized Bayesian network-based model provides better prediction in veterans' likelihood of suffering from PTSD as compared with a variety of state-of-art probabilistic classifiers.

Our study contributes to the Department of Veterans Affairs in two ways. First, the proposed model measures veterans' risk of suffering from PTSD only based on some basic information, which are easy to obtain. VA can apply a diversion strategy by assigning the high-risk patient directly to the mental health provider to make them access to the healthcare resource earlier. Such strategy will improve the efficiency of VA's PTSD diagnostic process, and reduce veterans' waiting time. Second, we identify 8 variables which provide the most directly predictive power by looking at the Markov boundary of *PTSD* in the Bayesian network model. This helps VA to identify the

high-risk veteran population for PTSD, and provides further guidance for the psychologists at the clinical treatment.

References

[1] ADAMS, R. E., AND BOSCARINO, J. A. Differences in mental health outcomes among Whites, African Americans, and Hispanics following a community disaster. *Psychiatry: Interpersonal and Biological Processes 68*, 3 (2005), 250–265.

[2] BISHOP, C. M., AND LASSERRE, J. Generative or discriminative? Getting the best of both worlds. In *Bayesian Statistics 8*. Oxford Univ. Press, 2007, pp. 3–24.

[3] BOSCARINO, J. A., ERLICH, P. M., HOFFMAN, S. N., AND ZHANG, X. Higher FKBP5, COMT, CHRNA5, and CRHR1 allele burdens are associated with PTSD and interact with trauma exposure: implications for neuropsychiatric research and treatment. *Neuropsychiatric Disease and Treatment 8* (2012), 131–139.

[4] BREIMAN, L. Random forests. *Machine Learning 45*, 1 (2001), 5–32.

[5] BREWIN, C. R., ANDREWS, B., AND VALENTINE, J. D. Meta-analysis of risk factors for posttraumatic stress disorder in trauma-exposed adults. *Journal of Consulting and Clinical Psychology 68*, 5 (2000), 748–766.

[6] CHEN, Q., AND WANG, S. Variable selection for multiply-imputed data with application to dioxin exposure study. *Statistics in Medicine 32*, 21 (2013), 3646–3659.

[7] DEMPSTER, A. P., LAIRD, N. M., AND RUBIN, D. B. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological) 39*, 1 (1977), 1–38.

[8] DEPARTMENT OF VETERANS AFFAIRS. Veterans benefits administration annual benefits report fiscal year 2018. `https://www.benefits.va.gov/REPORTS/abr/docs/2018-abr.pdf`.

[9] DOMINGOS, P., AND PAZZANI, M. Beyond independence: Conditions for the optimality of the simple Bayesian classifier. In *Proc. 13th Intl. Conf. Machine Learning* (1996), pp. 105–112.

[10] EFRON, B. The efficiency of logistic regression compared to normal discriminant analysis. *Journal of the American Statistical Association 70*, 352 (1975), 892–898.

[11] EFRON, B., HASTIE, T., JOHNSTONE, I., AND TIBSHIRANI, R. Least angle regression. *The Annals of Statistics 32*, 2 (2004), 407–499.

[12] EGGLESTON, H. G. *Convexity*. Cambridge University Press, 1958.

[13] EZAWA, K. J., AND SCHUERMANN, T. Fraud/uncollectible debt detection using a Bayesian network based learning system: A rare binary outcome with mixed data structures. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence* (1995), Morgan Kaufmann Publishers Inc., pp. 157–166.

[14] FAN, J., AND LI, R. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association 96*, 456 (2001), 1348–1360.

[15] FAN, J., AND PENG, H. Nonconcave penalized likelihood with a diverging number of parameters. *The Annals of Statistics 32*, 3 (2004), 928–961.

[16] FAYYAD, U. M., AND IRANI, K. B. Multi-interval discretization of continuous-valued attributes for classification learning. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence* (1993), Morgan Kaufmann Publishers Inc., pp. 1022–1027.

[17] FERREIRA, J., DENISON, D., AND HAND, D. Weighted naïve Bayes modelling for data mining, 2001.

[18] FRIEDMAN, N., GEIGER, D., AND GOLDSZMIDT, M. Bayesian network classifiers. *Machine Learning 29*, 2–3 (1997), 131–163.

[19] FRUEH, B. C., GRUBAUGH, A. L., ACIERNO, R., ELHAI, J. D., CAIN, G., AND MAGRUDER, K. M. Age differences in posttraumatic stress disorder, psychiatric disorders, and healthcare service use among veterans in Veterans Affairs primary care clinics. *The American Journal of Geriatric Psychiatry 15*, 8 (2007), 660–672.

[20] FRUEH, B. C., GRUBAUGH, A. L., ELHAI, J. D., AND BUCKLEY, T. C. US department of veterans affairs disability policies for posttraumatic stress disorder: Administrative trends and implications for treatment, rehabilitation, and research. *American Journal of Public Health 97*, 12 (2007), 2143–2145.

[21] FUJINO, A., UEDA, N., AND SAITO, K. A hybrid generative/discriminative approach to text classification with additional information. *Information Processing and Management 43*, 2 (2007), 379–392.

[22] GALATZER-LEVY, I. R., KARSTOFT, K.-I., STATNIKOV, A., AND SHALEV, A. Y. Quantitative forecasting of PTSD from early trauma responses: A machine learning application. *Journal of Psychiatric Research 59* (2014), 68–76.

[23] GALATZER-LEVY, I. R., MA, S., STATNIKOV, A., YEHUDA, R., AND SHALEV, A. Y. Utilization of machine learning for prediction of post-traumatic stress: a re-examination of cortisol in the prediction and pathways to non-remitting PTSD. *Translational Psychiatry 7*, 3 (2017), 1070.

[24] GAVIRIA, S. L., ALARCÓN, R. D., ESPINOLA, M., RESTREPO, D., LOTERO, J., BERBESI, D. Y., SIERRA, G. M., CHASKEL, R., ESPINEL, Z., AND SHULTZ, J. M. Sociodemographic patterns of posttraumatic stress disorder in Medellin, Colombia and the context of lifetime trauma exposure. *Disaster Health 3*, 4 (2016), 139–150.

[25] HALL, M. A decision tree-based attribute weighting filter for naïve Bayes. *Knowledge-Based Systems 20*, 2 (2007), 120–126.

[26] HAND, D. J., AND YU, K. Idiot's Bayes—not so stupid after all? *International statistical review 69*, 3 (2001), 385–398.

[27] HASTIE, T., TIBSHIRANI, R., AND FRIEDMAN, J. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software 33*, 1 (2010), 1–22.

[28] HASTIE, T., TIBSHIRANI, R., FRIEDMAN, J., AND FRANKLIN, J. The elements of statistical learning: data mining, inference and prediction. *The Mathematical Intelligencer 27*, 2 (2005), 83–85.

[29] HE, X., AND SHI, P. Convergence rate of B-spline estimators of nonparametric conditional quantile functions. *Journal of Nonparametric Statistics 3*, 3-4 (1994), 299–308.

[30] HOERL, A. E., AND KENNARD, R. W. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics 12*, 1 (1970), 55–67.

[31] HOLEVA, V., AND TARRIER, N. Personality and peritraumatic dissociation in the prediction of PTSD in victims of road traffic accidents. *Journal of Psychosomatic Research 51*, 5 (2001), 687–692.

[32] HUANG, S., LI, J., YE, J., FLEISHER, A., CHEN, K., WU, T., REIMAN, E., INITIATIVE, A. D. N., ET AL. A sparse structure learning algorithm for Gaussian Bayesian network identification from high-dimensional data. *IEEE Transactions on Pattern Analysis and Machine Intelligence 35*, 6 (2012), 1328–1342.

[33] KANG, C., AND TIAN, J. A hybrid generative/discriminative Bayesian classifier. In *Proceedings of the 19th International Florida Artificial Intelligence Research Society Conference* (2006), AAAI Press, pp. 562–567.

[34] KARSTOFT, K.-I., GALATZER-LEVY, I. R., STATNIKOV, A., LI, Z., AND SHALEV, A. Y. Bridging a translational gap: using machine learning to improve the prediction of PTSD. *BMC Psychiatry 15*, 1 (2015), 30.

[35] KNIGHT, K., AND FU, W. Asymptotics for lasso-type estimators. *The Annals of Statistics 28*, 5 (2000), 1356–1378.

[36] KONONENKO, I. Semi-naïve Bayesian classifier. In *Proceedings of the Sixth European Working Session on Learning* (1991), Springer, pp. 206–219.

[37] KWON, S., AND KIM, Y. Large sample properties of the SCAD-penalized maximum likelihood estimation on high dimensions. *Statistica Sinica 22*, 2 (2012), 629–653.

[38] MITCHELL, T. M. *Machine Learning*. WCB/McGraw-Hill, Boston, MA, 1997.

[39] NG, A. Y., AND JORDAN, M. I. On discriminative vs. generative classifiers: A comparison of logistic regression and naïve Bayes. In *Advances in Neural Information Processing Systems* (2002), pp. 841–848.

[40] NIBLETT, T. Constructing decision trees in noisy domains. In *Proceedings of the Second European Working Session on Learning* (Bled, Yugoslavia, 1987), Sigma, pp. 67–78.

[41] OMURCA, S. İ., AND EKINCI, E. An alternative evaluation of post traumatic stress disorder with machine learning methods. In *2015 International Symposium on Innovations in Intelligent SysTems and Applications (INISTA)* (2015), IEEE, pp. 1–7.

[42] PAZZANI, M. J. Searching for dependencies in Bayesian classifiers. In *Learning from Data*. Springer, 1996, pp. 239–248.

[43] POLE, N., BEST, S. R., METZLER, T., AND MARMAR, C. R. Why are hispanics at greater risk for PTSD? *Cultural Diversity and Ethnic Minority Psychology 11*, 2 (2005), 144.

[44] PRINS, A., BOVIN, M. J., SMOLENSKI, D. J., MARX, B. P., KIMERLING, R., JENKINS-GUARNIERI, M. A., KALOUPEK, D. G., SCHNURR, P. P., KAISER, A. P., LEYVA, Y. E., AND TIET, Q. Q. The primary care PTSD screen for DSM-5 (PC-PTSD-5): Development and evaluation within a veteran primary care sample. *Journal of General Internal Medicine 31*, 10 (2016), 1206–1211.

[45] RAINA, R., SHEN, Y., NG, A. Y., AND MCCALLUM, A. Classification with hybrid generative/discriminative models. In *Advances in Neural Information Processing Systems* (2003), pp. 545–552.

[46] RICHARDSON, L. K., FRUEH, B. C., AND ACIERNO, R. Prevalence estimates of combat-related post-traumatic stress disorder: critical review. *Australian and New Zealand Journal of Psychiatry 44*, 1 (2010), 4–19.

[47] RIJMEN, F. Bayesian networks with a logistic regression model for the conditional probabilities. *International Journal of Approximate Reasoning 48*, 2 (2008), 659–666.

[48] RUBIN, D. B. *Multiple Imputation for Nonresponse in Surveys*. John Wiley & Sons, 1987.

[49] RUBINSTEIN, Y., AND HASTIE, T. Discriminative vs. informative learning. In *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining* (1997), AAAI Press, pp. 49–53.

[50] SCHALINSKI, I., TEICHER, M. H., NISCHK, D., HINDERER, E., MÜLLER, O., AND ROCKSTROH, B. Type and timing of adverse childhood experiences differentially affect severity of PTSD, dissociative and depressive symptoms in adult inpatients. *BMC Psychiatry 16*, 1 (2016), 295.

[51] SCHUMM, J. A., BRIGGS-PHILLIPS, M., AND HOBFOLL, S. E. Cumulative interpersonal traumas and social support as risk and resiliency factors in predicting PTSD and depression among inner-city women. *Journal of Traumatic Stress 19*, 6 (2006), 825–836.

[52] TAN, Y., AND SHENOY, P. P. A bias-variance based heuristic for constructing a hybrid logistic regression-naïve Bayes model for classification. *International Journal of Approximate Reasoning 117* (2020), 15–28.

[53] TAN, Y., SHENOY, P. P., CHAN, M. W., AND ROMBERG, P. M. On construction of hybrid logistic regression-naïve Bayes model for classification. In *Proceedings of Eighth Interna-*

tional Conference on Probabilistic Graphical Models (Lugano, Switzerland, 2016), PMLR, pp. 523–534.

[54] TEYSSIER, M., AND KOLLER, D. Ordering-based search: A simple and effective algorithm for learning bayesian networks. *arXiv preprint arXiv:1207.1429* (2012).

[55] TIBSHIRANI, R. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological) 58*, 1 (1996), 267–288.

[56] XUE, J.-E., AND TITTERINGTON, D. M. Joint discriminative-generative modelling based on statistical tests for classification. *Pattern Recognition Letters 31*, 9 (2010), 1048–1055.

[57] XUE, J.-H., AND TITTERINGTON, D. M. Comment on "on discriminative vs. generative classifiers: A comparison of logistic regression and naïve Bayes". *Neural Processing Letters 28*, 3 (2008), 169.

[58] YUAN, M., AND LIN, Y. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 68*, 1 (2006), 49–67.

[59] ZAIDI, N. A., CARMAN, M. J., CERQUIDES, J., AND WEBB, G. I. Naïve-Bayes inspired effective pre-conditioner for speeding-up logistic regression. In *2014 IEEE International Conference on Data Mining* (2014), IEEE, pp. 1097–1102.

[60] ZAIDI, N. A., CERQUIDES, J., CARMAN, M. J., AND WEBB, G. I. Alleviating naïve Bayes attribute independence assumption by attribute weighting. *Journal of Machine Learning Research 14*, 1 (2013), 1947–1988.

[61] ZAIDI, N. A., WEBB, G. I., CARMAN, M. J., PETITJEAN, F., AND CERQUIDES, J. *ALR^n*: accelerated higher-order logistic regression. *Machine Learning 104*, 2-3 (2016), 151–194.

[62] ZHANG, N. L., AND POOLE, D. A simple approach to Bayesian network computations. In *Proceedings of the Tenth Biennial Conference-Canadian Society for Computational Studies of Intelligence* (1994), pp. 171–178.

[63] ZHENG, F., WEBB, G. I., SURAWEERA, P., AND ZHU, L. Subsumption resolution: An efficient and effective technique for semi-naïve Bayesian learning. *Machine Learning 87*, 1 (2012), 93–125.

[64] ZOU, H., AND HASTIE, T. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 67*, 2 (2005), 301–320.

Appendix A

Appendix

A.1 Detailed Results for Chapter 1

This appendix presents the detailed results for average structure of the hybrid model, prediction accuracies measured by 0-1 loss (in units of %) and RMSE, and training time.

Table A.1: Average Structure of the Hybrid Model

Dataset	*# Features*	*# LR-part*	*# NB-part*	*Dataset*	*# Features*	*# LR-part*	*# NB-part*
Abalone	8	8	0	Iris	4	1.133	2.867
Balance Scale	4	4	0	Liver Disorders	6	5.858	0.142
Banknote Authentication	4	4	0	Magic Gamma Telescope	10	10	0
Qualitative Bankruptcy	6	4.872	1.128	Mammographic Mass	5	4.74	0.26
Blogger	5	1.402	3.598	Mushroom	21	15.376	5.624
Blood Transfusion Service Center	4	3.89	0.11	New Thyroid	5	2.302	2.698
Car Evaluation	6	6	0	Pima Indians Diabetes	8	7.313	0.687
Connectionist Bench	60	47.505	12.495	Statlog Vehicle Silhouettes	18	15.585	2.415
Credit Approval	15	7.109	7.891	Vertebral Column	6	4.656	1.344
Hepatitis	19	13.747	5.253	Congressional Voting Records	16	15.4	0.6
Heart Disease (Hungarian)	13	5.555	7.445	Wilt	5	4.489	0.511
Hypothyroid	17	16.019	0.981	Wine	13	1.314	11.686
ILPD (Indian Liver Patient Dataset)	10	9.926	0.074				

Table A.2: Summary of Average 0-1 Loss of Hybrid Model, LR, NB, RF, and LASSO in units of % (SE in parentheses).

Dataset	*Hybrid*	*LR*	*NB*	*RF*	*LASSO*
Abalone	36.34 (0.14)	36.34 (0.14)	41.26 (0.16)	36.11 (0.14)	36.26 (0.14)
Balance Scale	1.84 (0.06)	1.84 (0.06)	8.59 (0.11)	16.21 (0.14)	1.47 (0.04)
Banknote Authentication	5.39 (0.06)	5.39 (0.06)	7.18 (0.06)	5.41 (0.06)	5.38 (0.06)
Qualitative Bankruptcy	0.86 (0.06)	0.77 (0.06)	0.76 (0.05)	0.00 (0.00)	0.58 (0.05)
Blogger	27.68 (0.41)	27.00 (0.42)	28.54 (0.41)	15.71 (0.37)	28.66 (0.43)
Blood Transfusion Service Center	22.30 (0.14)	22.30 (0.14)	24.70 (0.15)	22.61 (0.15)	22.34 (0.15)
Car Evaluation	6.59 (0.06)	6.59 (0.06)	14.73 (0.09)	2.78 (0.04)	6.71 (0.06)
Connectionist Bench	18.16 (0.29)	20.48 (0.28)	19.77 (0.30)	10.31 (0.24)	12.81 (1.00)
Credit Approval	13.54 (0.12)	14.37 (0.13)	13.25 (0.12)	13.71 (0.12)	13.93 (0.13)
Hepatitis	11.38 (0.26)	13.20 (0.28)	11.72 (0.25)	9.07 (0.22)	9.81 (0.23)
Heart Disease (Hungarian)	33.13 (0.26)	33.02 (0.26)	34.21 (0.26)	33.29 (0.25)	34.29 (0.25)
Hypothyroid	0.81 (0.02)	0.79 (0.02)	1.29 (0.02)	0.84 (0.02)	0.76 (0.02)
ILPD (Indian Liver Patient Dataset)	27.53 (0.16)	27.52 (0.16)	31.29 (0.18)	29.51 (0.17)	28.51 (0.17)
Iris	5.97 (0.18)	5.82 (0.18)	5.53 (0.17)	5.03 (0.17)	6.13 (0.18)
Liver Disorders	33.33 (0.24)	33.27 (0.24)	35.59 (0.24)	31.23 (0.23)	33.06 (0.24)
Magic Gamma Telescope	15.12 (0.07)	15.12 (0.07)	21.63 (0.08)	14.29 (0.06)	15.13 (3.16)
Mammographic Mass	17.21 (0.11)	17.19 (0.11)	16.20 (0.11)	17.10 (0.11)	16.88 (0.11)
Mushroom	0.00 (0.00)	0.00 (0.00)	5.51 (0.10)	0.00 (0.00)	0.00 (0.00)
New Thyroid	5.07 (0.15)	5.60 (0.16)	3.81 (0.12)	3.35 (0.12)	4.11 (0.13)
Pima Indians Diabetes	18.74 (0.13)	18.74 (0.13)	19.82 (0.14)	20.29 (0.14)	18.71 (0.13)
Statlog Vehicle Silhouettes	25.48 (0.15)	25.31 (0.16)	34.92 (0.16)	22.92 (0.14)	24.65 (0.15)
Vertebral Column	15.27 (0.19)	15.84 (0.20)	21.14 (0.22)	15.21 (0.19)	15.84 (0.20)
Congressional Voting Records	3.80 (0.09)	3.77 (0.09)	8.90 (0.13)	3.05 (0.08)	3.25 (0.08)
Wilt	2.78 (0.02)	2.78 (0.02)	5.38 (0.03)	2.91 (0.02)	2.78 (0.02)
Wine	2.34 (0.11)	3.09 (0.14)	1.16 (0.08)	1.18 (0.08)	1.64 (0.10)

Table A.3: Summary of Average RMSE of Hybrid Model, LR, NB, RF, and LASSO (SE in parentheses).

Dataset	*Hybrid*	*LR*	*NB*	*RF*	*LASSO*
Abalone	0.3899 (0.0005)	0.3899 (0.0005)	0.4513 (0.0009)	0.4100 (0.0007)	0.3895 (0.0005)
Balance Scale	0.0841 (0.0021)	0.0841 (0.0021)	0.2837 (0.0007)	0.2807 (0.0009)	0.0651 (0.0012)
Banknote Authentication	0.2018 (0.0009)	0.2018 (0.0009)	0.2264 (0.0007)	0.2169 (0.0012)	0.2016 (0.0009)
Qualitative Bankruptcy	0.0411 (0.0024)	0.0359 (0.0025)	0.0361 (0.0015)	0.0448 (0.0010)	0.0419 (0.0017)
Blogger	0.4331 (0.0029)	0.4440 (0.0030)	0.4272 (0.0027)	0.3315 (0.0034)	0.4322 (0.0023)
Blood Transfusion Service Center	0.3962 (0.0009)	0.3962 (0.0009)	0.4103 (0.0010)	0.4406 (0.0015)	0.3963 (0.0009)
Car Evaluation	0.1498 (0.0006)	0.1498 (0.0006)	0.2272 (0.0004)	0.1326 (0.0003)	0.1500 (0.0006)
Connectionist Bench	0.4070 (0.0038)	0.4369 (0.0034)	0.3866 (0.0032)	0.3218 (0.0013)	0.2896 (0.0022)
Credit Approval	0.3179 (0.0014)	0.3188 (0.0013)	0.3237 (0.0015)	0.3173 (0.0012)	0.3159 (0.0012)
Hepatitis	0.2792 (0.0039)	0.3157 (0.0044)	0.2831 (0.0040)	0.2677 (0.0023)	0.2703 (0.0028)
Heart Disease (Hungarian)	0.2959 (0.0010)	0.2985 (0.0010)	0.3022 (0.0011)	0.2999 (0.0011)	0.2921 (0.0009)
Hypothyroid	0.0840 (0.0008)	0.0832 (0.0008)	0.1017 (0.0009)	0.0812 (0.0007)	0.0809 (0.0008)
ILPD (Indian Liver Patient Dataset)	0.4162 (0.0009)	0.4161 (0.0009)	0.4601 (0.0013)	0.4348 (0.0011)	0.4137 (0.0008)
Iris	0.1351 (0.0032)	0.1457 (0.0031)	0.1258 (0.0031)	0.1271 (0.0029)	0.1373 (0.0025)
Liver Disorders	0.4537 (0.0010)	0.4534 (0.0010)	0.4655 (0.0008)	0.4723 (0.0016)	0.4538 (0.0010)
Magic Gamma Telescope	0.3347 (0.0005)	0.3347 (0.0005)	0.3903 (0.0007)	0.3309 (0.0006)	0.3348 (0.0006)
Mammographic Mass	0.3424 (0.0010)	0.3422 (0.0010)	0.3565 (0.0012)	0.3592 (0.0012)	0.3421 (0.0009)
Mushroom	0.0000 (00000)	0.0002 (0.0001)	0.2065 (0.0062)	0.0061 (0.0002)	0.0035 (0.0001)
New Thyroid	0.1308 (0.0027)	0.1416 (0.0033)	0.1130 (0.0022)	0.1264 (0.0018)	0.1214 (0.0025)
Pima Indians Diabetes	0.3689 (0.0010)	0.3691 (0.0010)	0.3784 (0.0011)	0.3771 (0.0011)	0.3694 (0.0010)
Statlog Vehicle Silhouettes	0.2910 (0.0008)	0.2942 (0.0009)	0.3695 (0.0009)	0.2750 (0.0007)	0.2803 (0.0007)
Vertebral Column	0.3435 (0.0017)	0.3461 (0.0016)	0.3879 (0.0021)	0.3481 (0.0023)	0.3448 (0.0014)
Congressional Voting Records	0.1640 (0.0024)	0.1635 (0.0025)	0.2761 (0.0022)	0.1572 (0.0015)	0.1481 (0.0018)
Wilt	0.1584 (0.0006)	0.1584 (0.0006)	0.1911 (0.0006)	0.1630 (0.0006)	0.1584 (0.0006)
Wine	0.0751 (0.0029)	0.0909 (0.0033)	0.0431 (0.0022)	0.0994 (0.0011)	0.0690 (0.0022)

Table A.4: Summary of Average Train Time of Hybrid Model, LR, NB and RF in seconds.

Dataset	*Hybrid*	*LR*	*NB*	*RF*	*LASSO*
Abalone	0.5224	0.5204	0.0090	2.5621	27.9373
Balance Scale	0.0206	0.0211	0.0023	0.1575	5.1914
Banknote Authentication	0.0339	0.0330	0.0030	0.4136	2.1155
Qualitative Bankruptcy	0.0050	0.0049	0.0026	0.0229	0.3455
Blogger	0.0056	0.0055	0.0021	0.0240	0.2145
Blood Transfusion Service Center	0.0084	0.0083	0.0024	0.1022	0.4382
Car Evaluation	0.1188	0.1185	0.0038	0.5156	10.5964
Connectionist Bench	0.0222	0.0226	0.0169	0.0781	0.6644
Credit Approval	0.0183	0.0204	0.0056	0.1418	1.2105
Hepatitis	0.0134	0.0132	0.0070	0.0434	1.1598
Heart Disease (Hungarian)	0.0130	0.0200	0.0043	0.0750	1.2388
Hypothyroid	0.0715	0.0722	0.0116	0.9972	3.7391
ILPD (Indian Liver Patient Dataset)	0.0087	0.0088	0.0038	0.0956	0.4695
Iris	0.0045	0.0074	0.0020	0.0178	0.7708
Liver Disorders	0.0091	0.0092	0.0034	0.0789	0.3703
Magic Gamma Telescope	2.0696	2.0794	0.0291	18.5753	21.2618
Mammographic Mass	0.0118	0.0114	0.0031	0.1633	0.6603
Mushroom	0.4388	0.5057	0.0239	2.1637	12.6604
New Thyroid	0.0065	0.0079	0.0022	0.0281	0.8481
Pima Indians Diabetes	0.0127	0.0132	0.0037	0.1518	0.5654
Statlog Vehicle Silhouettes	0.1260	0.1734	0.0065	0.4541	11.1105
Vertebral Column	0.0054	0.0047	0.0025	0.0346	0.3363
Congressional Voting Records	0.0178	0.0187	0.0065	0.0932	0.8938
Wilt	0.0519	0.0463	0.0062	0.9906	3.1263
Wine	0.0082	0.0134	0.0042	0.0276	0.6744