Avoiding Bias from Feature Selection with F-statistic for Classification Based on High-dimensional Features

Longhai Li*

3 March 2010

Abstract

There is a great demand for efficient, accurate and precise methods for predicting a discrete response from a great many of features, such as microarray gene expression data, and mass spectrometry data. For computational and other reasons, it is necessary to select a subset of features before tting a statistical model, by looking at how strongly the features are related to the response. However, such feature selection procedure will result in overcondent predictive probabilities for future cases if we model the retained data directly. In this paper, I show that this feature selection bias can be avoided in Gaussian classification models for continuous features and multi-class response. The classication method presented in this paper is tested using simulated data sets and a gene expression data that is related to small round blue cell tumors (SRBCT) of childhood.

Short title: Avoiding Bias from Feature Selection with F-statistic

Key words: Bayesian methods, feature selection bias, high-dimensional classication, Markov chain Monte Carlo

*Department of Mathematics and Statistics, University of Saskatchewan, Saskatoon, Saskatchewan, S7N5E6, CANADA. Email: longhai@math.usask.ca. Web: http://math.usask.ca/~longhai.

1 Introduction

Some new technologies can easily measure values of high-dimensional *features* (also known as covariates, explanatory variables, inputs, etc.) of objects/subjects (called *cases* generally). For example, microarray technology can simultaneously measure the expression levels of thousands of genes of a patient, and mass spectrometry technology can produce a high-dimensional proteomic profile. It is interesting to predict a certain *categorical* characteristic associated with a case, called *response*, given the values of its high-dimensional features. For example, a response may be an indicator of whether a kind of disease is present in a patient, or types of diseases (see eg. Khan et al., 2001; Tibshirani et al., 2002; Wu et al., 2003). Very often it is also interesting to find out a subset of features that are the most useful in predicting the response. For this purpose, data of features of a set of cases (called *training cases*) for which we know the values of response are collected, called *training data*. A link function between the response and features is then inferred by analyzing the training data, and may be used to predict the repose of future cases. The link function is often expressed as a conditional probability function for the response given the features, and the conditional probabilities given the values of features of a set of future cases are called *predictive probabilities*. When the true response values of future cases are also known, we can use them to test the performance of a prediction method, so often future cases are also called *test cases*. Typically we divide all the available cases into two subsets in some way, treating one as training cases and the other as test cases. In statistics, methods for such data analysis are called classification methods, or discriminant analysis. In contrast to the high dimensionality of features, the number of training cases is typically very small (no more than a hundred) in such data sets due to high cost in carrying out experiments, mostly from measuring the true response value. In statistics literature, usually the number of training cases is denoted by n, and the number of features is denoted by p. Therefore, such problems are often called "small n and large p" classification problems.

Feature selection is commonly applied to a "small n and large p" dataset before fitting a model to it because statistical inferences for such problems are challenging, both in theory and computation. Simple statistical methods, such as maximum likelihood estimation, will overfit the data, ie, model the noise in the data rather than the signal. For example, even a simple linear model will overfit a data set when the number of cases isn't much larger than the number of features. We may resort to sophisticated Bayesian methods, which do not suffer from the overfitting problem. However, we will likely need to use Markov chain

Monte Carlo methods to sample from a posterior distribution, which may be computationally infeasible when p is very large. Therefore, an almost ubiquitously used strategy is to select a small subset of features that appear fairly predictive for a response using training data before fitting the model, by looking at some simple score measuring the relevance between the response and the features, such as the absolute correlation and F-statistic. This practice was evidenced by the outcomes of 2003 NIPS feature selection competition (Guyon et al., 2006). There are other feature selection methods based on linear models, which can take into account the correlations between features, but they become infeasible when p is as large as thousands.

Unfortunately, feature selection will introduce optimistic bias into statistical inference. That is, the strength of relationship between the selected features and response is exaggerated by the feature selection procedure, and the estimate for the strength is therefore biased to a higher value (if there is a parameter associated with this strength). An extreme example is that all features are irrelevant to a response, but the selected features will appear fairly predictive to the response, which is, however, wholly made by chance (see Ambroise and McLachlan, 2002, for a formal demonstration). we will call this problem *feature selection bias*. The effect on predictions for future cases caused by feature selection bias is that the predictive probabilities will be overconfident. For example, for a group of future cases, the predictive probabilities of their responses being 1 are between 0.9 and 1, but the actual fraction of their responses being 1 is only 0.7.

Feature selection bias problem, especially in classification contexts, has so far drawn very few attention in statistics literature. However, to our knowledge, some authors have addressed this problem. In classification problems, feature selection bias in the cross-validation estimate of error rate has been noticed by researchers working on gene expression data. Ambroise and McLachlan (2002); Lecocke and Hess (2004); Raudys et al. (2005); Singhi and Liu (2006) have found that if a cross-validation evaluation of classification algorithms is applied on a data set containing only a subset of features selected beforehand based on the whole data set, the error rate could be misleadingly small, which could be 0 for easy data sets, such as leukemia data set (see eg. Xiong et al., 2001). It is therefore suggested that the feature selection should be "internal" to the cross-validation procedure, ie, the feature selection should be redone for each splitting of data set into training and test set. However, this is only a proper method for evaluating a possibly poorly-calibrated classification procedure (ie, the bias is considered in the evaluation), but not a method for giving well-

calibrated predictive probabilities with the selected features for future cases. In other words, this method eliminates the bias in estimating the error rate, rather than the bias in estimating predictive probabilities themselves. Feature selection bias has been long recognized in regression problems (in which the response is continuous). Hurvich and Tsai (1990) and Zhang (1992) respectively used Monte Carlo simulation and theoretical analysis to show that after feature selection the actual coverage rate of confidence regions for regression coefficients is smaller than the nominal probability. Be aware of the severe bias, some authors suggested that the feature selection and parameter estimation should be performed separately on different cases. In the context of high-dimensional data discussed here, this quick solution is undesirable because the number of training cases is usually very small. Using fewer cases in training stage will hurt the discriminative power, resulting in a well-calibrated but less accurate prediction for future cases. Shen et al. (2004) propose a methodology that uses optimal approximation and perturbation of response values to estimate the mean and variance of the least square estimator of coefficients after feature selection, and therefore finds approximately well-calibrated confidence region for the coefficients. A very recent work by Wang and Lagakos (2009) reaches the goal of Shen et al. (2004) by a method of permuting the response values. However, it seems impossible to apply these methods to classification problems.

Avoiding feature selection bias in the estimates of predictive probabilities for future cases is important. Interestingly, Singhi and Liu (2006) empirically shows that feature selection doesn't alter error rate of a classification method. For example if the response is binary, the predictive probabilities based on a selected subset of features for those cases on the classification boundary are still close to 1/2, therefore the predicted values of response variables for test cases by thresholding at 1/2 are the same after feature selection. This is also confirmed by Li et al. (2008). However, error rate is useful only when losses incurred from different types of prediction errors are the same. In practice, this is often not the case. For example, erroneously classifying a patient with a disease into non-disease may cause more loss than the opposite error. In such situations, overconfident predictive probabilities may result in higher loss on average (eg, error rate using a threshold other than 1/2). Even when the error rate is useful, an *expected error rate* for future cases is often required in addition to the predicted values of response. Here, the expected error rate means the estimate of error rate for future cases. For example, predictive probability for a future case is 0.7, and using threshold 1/2 we guess its response by 1, the expected error probability is 0.3. The expected error rate for a set of future (test) cases is the average of these expected error probabilities

of all future cases. Overconfident predictive probabilities will result in an over-optimistic expected error rate. Therefore solving the problem of feature selection bias in predictive probabilities is well demanded in practice.

The Bayesian methodology reported here uses the principle proposed by Li et al. (2008) to obtain well-calibrated predictions for future cases in classification problems. Based on modeling all features (including those omitted) with a hierarchical Bayesian model, we use both the values of retained features and the information that a certain number of features were omitted in the feature selection procedure to form a correct (unbiased) posterior distribution of model parameters. Intuitively speaking, the upward bias in estimating the strength of the relationship between the response and the features caused by the feature selection will be canceled by the fact that a certain number of features were omitted due to weak relevance with the response. The difficulty in applying this method is that the computation of an adjustment factor — the probability that a feature fails to pass a feature selection filter, is burdensome, for which we need to integrate (or sum) with respect to the distribution of the values of omitted features and the prior distribution of relevant parameters. Due to this computational difficulty, the model used in Li et al. (2008) for illustrating the principle is very simple, modeling only *binary* features and *binary* response. In this paper, we show that the principle can also be applied to a realistic classification model for *continuous* features and *multi-class* response. The resulting classification methodology may be practically useful, for example in disease diagnosis with gene expression and mass spectrometry data.

This paper will be structured as follows. In Section 2, we present the details of the methodology. In Section 3, we use a simulated dataset to illustrate the method, where we will show empirically that the predictive probabilities yielded by bias-corrected methods are indeed well-calibrated. In Section 4 we use a gene expression data to test the method, where we show that the bias-correction method does improve the predictions. The paper will conclude in Section 5.

2 The Methodology

2.1 A Hierarchical Bayesian Model for High-throughput Data

We are interested in predicting a categorical response variable y, which is sometimes called class label, given the information of a set of features $x_1, \ldots, x_p$. Here we assume that y can

take integer value from 1 to G, and all the x_i's are continuous. The observation for case i is denoted by $y^{(i)}$ and $x_1^{(i)}, \ldots, x_p^{(i)}$. Given parameter $\psi_1, \ldots, \psi_G$ (collectively written as $\boldsymbol{\psi}$), the response variable $y^{(i)}$ takes a value $g \in \{1, \ldots, G\}$ with a probability of ψ_g. Conditional on $y^{(i)} = g$, the predictor variables $x_1^{(i)}, \ldots, x_p^{(i)}$ are assumed to be independent, and $x_j^{(i)}$ is distributed with $N(\mu_j^{(g)}, w_j^x)$. Cases are also assumed to be independent given parameters. We will assume that the data have been collected on n cases. To ease elicitation of priors, we also assume that the values of x_j have been centralized, therefore having a mean close to 0.

The assumptions of independence between features and identical variances across different classes may not be realistic. However, because the number of observations in medical data (such as microarray data) is typically very small (no more than a hundred), methods with these assumptions often outperform sophisticated methods based on more flexible models (due to reduction of computational complexity). In particular, Dudoit et al. (2002) compared the diagonal linear discriminant analysis (DLDA) (which assumes the same data model), fisher's linear, linear and quadratic discriminant analysis (FLDA, LDA, QDA), decision tree (bagged and boosted), nearest neighbors (NN), as well as the weighted voting scheme of Golub et al. (1999). They found that DLDA performs the best in 3 out of 4 data sets, and makes only 1 missclassification in the easy lymphoma data set, very near the best, 0, by NN. Futhermore, due to the availability of a large number of features in high-throuput data, the loss of classification accuracy due to ignoring correlations between features may be well remedied by other differential features. However, we gain much computational simplicity by using such simple models. The method reported in this paper is very fast in computation since the posterior distribution depends only on some sufficient statistics — sample means and variances of features.

We will assign priors to $\mu_j^{(g)}$ and w_j^x with a hierarchical form. Before we give detailed explanation of the priors, we lay out the model for data and priors by the following equations, and display it graphically by Figure 1.

$$P(y^{(i)} = g \,|\, \boldsymbol{\psi}) = \psi_g, \quad \text{for } g = 1, \ldots, G\,, \tag{1}$$

$$\psi_1, \ldots, \psi_G \sim \text{Dirichlet}(c_1, \ldots c_G), \tag{2}$$

$$x_j^{(i)} \,|\, y^{(i)} = g, \mu_j^{(g)}, w_j^x \sim N(\mu_j^{(g)}, w_j^x), \quad \text{for } j = 1, \ldots, p\,, \tag{3}$$

$$\mu_j^{(1)}, \ldots, \mu_j^{(G)} \,|\, \nu_j, w_j^\mu \overset{\text{IID}}{\sim} N(\nu_j, w_j^\mu), \quad \text{for } j = 1, \ldots, p, \tag{4}$$

$$\nu_1, \ldots, \nu_p \,|\, w^\nu \overset{\text{IID}}{\sim} N(0, w^\nu), \tag{5}$$

$$w^\nu \sim \text{IG}\left(\frac{\alpha_0^\nu}{2}, \frac{\alpha_0^\nu w_0^\nu}{2}\right), \tag{6}$$

$$w_1^\mu, \ldots, w_p^\mu \,|\, w^\mu \overset{\text{IID}}{\sim} \text{IG}\left(\frac{\alpha_1^\mu}{2}, \frac{\alpha_1^\mu w^\mu}{2}\right), \tag{7}$$

$$w^\mu \sim \text{IG}\left(\frac{\alpha_0^\mu}{2}, \frac{\alpha_0^\mu w_0^\mu}{2}\right), \tag{8}$$

$$w_1^x, \ldots, w_p^x \,|\, w^x \overset{\text{IID}}{\sim} \text{IG}\left(\frac{\alpha_1^x}{2}, \frac{\alpha_1^x w^x}{2}\right), \tag{9}$$

$$w^x \sim \text{IG}\left(\frac{\alpha_0^x}{2}, \frac{\alpha_0^x w_0^x}{2}\right). \tag{10}$$

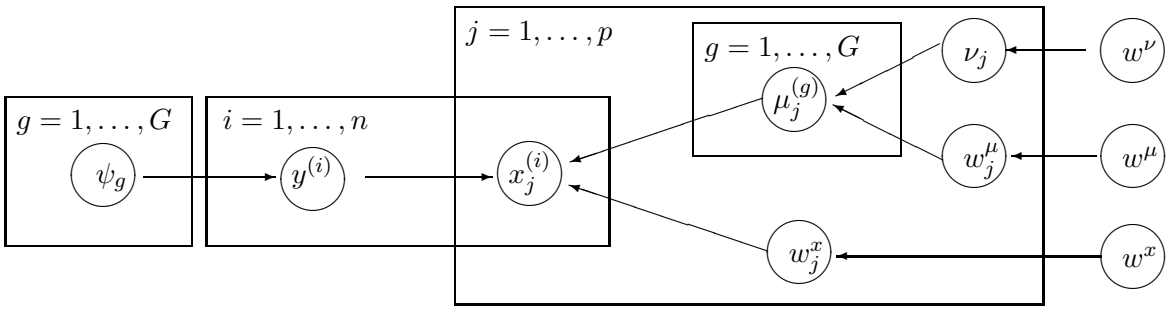

Figure 1: Graphical representation of the hierarchical Bayesian classification model.

For simplicity of presentation below, we will use $i{:}j$ to denote the vector of integers from i to j ($i \le j$), and use $\boldsymbol{A}_{i:j}$ to denote the collection of objects $A_i, \ldots, A_j$, similarly when we use a vector in superscript. In addition, we will use bold-faced letters to denote collections of items. By this convention, we will use the following notations frequently in this paper: $\boldsymbol{y}^{(1:n)} = (y^{(1)}, \ldots, y^{(n)})$, $\boldsymbol{x}_{1:p} = (x_1, \ldots, x_p)$, $\boldsymbol{x}_j^{(1:n)} = (x_j^{(1)}, \ldots, x_j^{(n)})$, and $\boldsymbol{x}_{1:p}^{(1:n)} = (\boldsymbol{x}_{1:p}^{(1)}, \ldots, \boldsymbol{x}_{1:p}^{(n)})$, $\boldsymbol{\mu}_{1:p}^{(g)} = (\mu_1^{(g)}, \ldots, \mu_p^{(g)})$, $\boldsymbol{\mu}_j^{(1:G)} = (\mu_j^{(1)}, \ldots, \mu_j^{(G)})$, $\boldsymbol{\mu}_{1:p}^{(1:G)} = (\boldsymbol{\mu}_1^{(1:G)}, \ldots, \boldsymbol{\mu}_p^{(1:G)})$, $\boldsymbol{w}_{1:p}^x = (w_1^x, \ldots, w_p^x)$, $\boldsymbol{w}_{1:p}^\mu = (w_1^\mu, \ldots, w_p^\mu)$, $\boldsymbol{\nu}_{1:p} = (\nu_1, \ldots, \nu_p)$. The last two groups of parameters will be introduced soon.

A natural choice of prior for $\boldsymbol{\psi}$ is a Dirichlet distribution (see eg Gelman et al., 2004) with parameters $c_1, \ldots, c_G$. The $\boldsymbol{\mu}_j^{(1:G)}$ are assigned independent normal distributions with mean ν_j and variance w_j^μ (see (4)). This reflects our prior belief that the mean parameters $\boldsymbol{\mu}_j^{(1:G)}$ are close to a common level ν_j. At a higher level, we assign $\boldsymbol{w}_{1:p}^\mu$ the conjugate Inverse-Gamma (IG) distribution with shape parameter $\alpha_1^\mu/2$ and rate parameter $\alpha_1^\mu w^\mu/2$

(see (7)), in which α_1^μ and w^μ control respectively the width and magnitude of $\boldsymbol{w}_{1:p}^\mu$. We assign a different variance w_j^μ for different features to have the effect of feature selection. For high-throughput data, we believe that most of features are indifferential across different classes, while a few of them may be much more differential than others. When fed with the data, the fitted model could have a few w_j^μ very large while others are very small if this is a scenario favored by the data. In this prior setting, if we integrate w_j^μ away, it will lead to a multivariate t distribution for $\boldsymbol{\mu}_j^{(1:G)}$ with degree freedom α_1^μ and scale parameter w^μ. As is well-known, t distribution has heavier tails than normal distribution, and therefore is more suitable to model a group of parameters that are mostly very small, but a few of which are extraordinarily large. This explains in another way the feature selection ability of this prior. We could let α_1^μ be a variable, and learn from the data. However, we do not have a choice that is computationally easy. We therefore leave α_1^μ to be fixed by an analyst. Setting it to 2 or 3 is expected to work well for most high-throughput data sets. But it could be chosen empirically from the data. A way to set α_1^μ is to look at the qq-plot of $w_{1:p}^\mu$ estimated based on the MLE of $\boldsymbol{\mu}_{1:p}^{(1:G)}$ with the quantiles of IG distribution with different α_1^μ. The hyperparameter w^μ is given an IG distribution with fixed parameters α_0^μ and w_0^μ (see (8)), allowing it to learn from the data (for example with information contained in w_j^μ in Gibbs sampling). The values of α_0^μ and w_0^μ is to be fixed by an analyst, usually given small positive values, defining a diffuse prior for w^μ. The result is not sensitive to the choice of them and a single data set doesn't provide much information for the shape of the distribution of w^μ (which can be learned only with multiple data sets of a problem.). It is similar to assign the prior for $\boldsymbol{w}_{1:p}^x$. If we have standardized the values of predictors $\boldsymbol{x}_j^{(1:n)}$ (which is necessary if one needs to look at the importance of features from the magnitudes of $\boldsymbol{w}_{1:p}^\mu$), we may set α_1^x to a larger value than α_1^μ, reflecting that the heterogeneousness among $\boldsymbol{w}_{1:p}^x$ is smaller. The common mean ν_j is given a normal distribution with mean 0, assuming that the $\boldsymbol{x}_j^{(1:n)}$ have been centralized and so the heterogeneousness among $\boldsymbol{\nu}_{1:p}$ is fairly small. We treat w^ν as a hyperparameter, allowing it to learn from the data.

The hyperparameters w^μ and w^x control the overall degree of relevance between the response and the features. When w^μ is larger, more features have large difference amongst $\boldsymbol{\mu}_j^{(1:G)}$, therefore more features are useful in predicting the response. The effect of w^x is opposite. Therefore, the value of w^μ/w^x indicates the information-noise ratio of a data set. As to be seen, our method for avoiding feature selection bias is just by modifying the posterior distribution of w^μ and w^x with the fact that a certain number of features were omitted before fitting the classification model.

2.2 Predictions with Complete Features

Suppose we want to predict the response y^* of a test case for which we know the values of features $\boldsymbol{x}^*_{1:p}$. We will first derive an expression of the predictive probability of $y^* = g$ given $\boldsymbol{x}^*_{1:p}$, $\boldsymbol{x}^{(1:n)}_{1:p}$ and $\boldsymbol{y}^{(1:n)}$, for $g = 1, \ldots, G$. Following Bayes rule, we obtain that:

$$P(y^* = g \,|\, \boldsymbol{x}^*_{1:p}, \boldsymbol{x}^{(1:n)}_{1:p}, \boldsymbol{y}^{(1:n)}) = \frac{P(y^* = g \,|\, \boldsymbol{y}^{(1:n)})\, P(\boldsymbol{x}^*_{1:p} \,|\, \boldsymbol{x}^{(1:n)}_{1:p}, y^* = g, \boldsymbol{y}^{(1:n)})}{\sum_{g=1}^{G} P(y^* = g \,|\, \boldsymbol{y}^{(1:n)})\, P(\boldsymbol{x}^*_{1:p} \,|\, \boldsymbol{x}^{(1:n)}_{1:p}, y^* = g, \boldsymbol{y}^{(1:n)})}. \quad (11)$$

To compute (11), we need to compute the numerator for all $g = 1, \ldots, G$, then divide them by their sum, which gives the denominator. The first factor in this numerator can be computed by Pòlya urn scheme:

$$P(y^* = g \,|\, \boldsymbol{y}^{(1:n)}) = \frac{n_g + c_g}{n + \sum_{g=1}^{G} c_g}\,, \quad (12)$$

where $n_g = \sum_{i=1}^{n} I(y^{(i)} = g)$ is the number of training cases in class g. The second factor can be written as:

$$\begin{aligned} &P(\boldsymbol{x}^*_{1:p} \,|\, \boldsymbol{x}^{(1:n)}_{1:p}, y^* = g, \boldsymbol{y}^{(1:n)}) \\ &\quad = \int\int P(\boldsymbol{x}^*_{1:p} \,|\, \boldsymbol{\mu}^{(g)}_{1:p}, \boldsymbol{w}^x_{1:p}, y^* = g)\, P(\boldsymbol{\mu}^{(g)}_{1:p}, \boldsymbol{w}^x_{1:p} \,|\, \boldsymbol{x}^{(1:n)}_{1:p}, \boldsymbol{y}^{(1:n)})\, d\boldsymbol{\mu}^{(g)}_{1:p}\, d\boldsymbol{w}^x_{1:p}, \end{aligned} \quad (13)$$

where,

$$P(\boldsymbol{x}^*_{1:p} \,|\, \boldsymbol{\mu}^{(g)}_{1:p}, \boldsymbol{w}^x_{1:p}, y^* = g) = (2\pi)^{p/2} \exp\left(-\frac{1}{2} \sum_{j=1}^{p} \left[\frac{(x^*_j - \mu^{(g)}_j)^2}{w^x_j} + \log(w^x_j) \right] \right). \quad (14)$$

We will use Markov chain Monte Carlo (MCMC) (see eg Neal, 1993, and references therein), to approximate the integral in (13), ie, averaging the quantity in (14) over a pool of samples of $\boldsymbol{\mu}^{(1:G)}_{1:p}$ and $\boldsymbol{w}^x_{1:p}$ drawn by simulating a Markov chain. To draw samples of $\boldsymbol{\mu}^{(1:G)}_{1:p}$ and $\boldsymbol{w}^x_{1:p}$, we can draw samples of all parameters ($\boldsymbol{\mu}^{(1:G)}_{1:p}$, $\boldsymbol{v}_{1:p}$, $\boldsymbol{w}^{\mu}_{1:p}$, $\boldsymbol{w}^x_{1:p}$ w_μ, w_ν) from their joint posterior distribution, which is proportional to the product of the distribution of $\boldsymbol{x}^{(1:n)}_{1:p}$ given $\boldsymbol{y}^{(1:n)}$ and the priors of parameters:

$$\prod_{j=1}^{p} \Big(P(\boldsymbol{x}^{(1:n)}_j \,|\, \boldsymbol{\mu}^{(1:G)}_j, w^x_j, \boldsymbol{y}^{(1:n)})\, P(\boldsymbol{\mu}^{(1:G)}_j \,|\, \nu_j, w^{\mu}_j)\, P(\nu_j \,|\, w^{\nu})\, P(w^{\mu}_j \,|\, w^{\mu})\, P(w^x_j \,|\, w^x) \Big) \times$$

$$P(w^{\mu})\, P(w^{\nu})\, P(w^{x}), \tag{15}$$

where the probability density functions are defined by the equations from (3) to (10), in particular, the first probability after product sign in (15) is given by:

$$P(\boldsymbol{x}_j^{(1:n)} \,|\, \boldsymbol{\mu}_j^{(1:G)}, w_j^x, \boldsymbol{y}^{(1:n)}) = (2\pi)^{n/2} \exp\left(-\frac{1}{2} \sum_{i=1}^{n} \left[\frac{(x_j^{(i)} - \mu_j^{(y^{(i)})})^2}{w_j^x} + \log(w_j^x) \right] \right). \tag{16}$$

In particular, we will use Gibbs sampling (Gelfand et al., 1990) to draw samples from this posterior. In Gibbs sampling, one partitions the whole set of parameters into subsets, mostly in a natural way, and then alternatively draw a sample from the conditional distribution of a subset of parameters given the parameter values of other subsets, in some fixed or random order. We may or may not be able to sample from these conditional distributions directly with standard methods. For the distribution (15), we can sample with standard methods from the following are the conditional distributions:

$$\mu_j^{(g)} \,|\, \boldsymbol{x}^{(1:n)}, w_j^{\mu}, v_j, w_j^x \;\sim\; N\left(\frac{\nu_j / w_j^{\mu} + \ddot{x}_j^{(g)} / w_j^x}{1/w_j^{\mu} + n_g / w_j^x}, \;\; \frac{1}{1/w_j^{\mu} + n_g / w_j^x} \right), \tag{17}$$

$$w_j^x \,|\, w^x, \boldsymbol{\mu}_j^{(1:G)}, \boldsymbol{y}^{(1:n)} \;\sim\; \mathrm{IG}\left(\frac{\alpha_1^x + n}{2}, \;\; \frac{\alpha_1^x \, w^x + \sum_{i=1}^{n} (x_j^{(i)} - \mu_j^{(y^{(i)})})^2}{2} \right), \tag{18}$$

$$\nu_j \,|\, \boldsymbol{\mu}_j^{(1:G)}, w_j^{\mu}, w_{\nu} \;\sim\; N\left(\frac{\ddot{\mu}_j / w_j^{\mu}}{1/w^{\nu} + G/w_j^{\mu}}, \;\; \frac{1}{1/w^{\nu} + G/w_j^{\mu}} \right), \tag{19}$$

$$w_j^{\mu} \,|\, w^{\mu}, v_j, \boldsymbol{\mu}_j^{(1:G)} \;\sim\; \mathrm{IG}\left(\frac{\alpha_1^{\mu} + G}{2}, \;\; \frac{\alpha_1^{\mu} \, w^{\mu} + \sum_{g=1}^{G} (\mu_j^{(g)} - \nu_j)^2}{2} \right), \tag{20}$$

$$w^{\nu} \,|\, \boldsymbol{\nu}_{1:p} \;\sim\; \mathrm{IG}\left(\frac{\alpha_0^{\nu} + p}{2}, \;\; \frac{\alpha_0^{\nu} \, w_0^{\nu} + \sum_{j=1}^{p} \nu_j^2}{2} \right), \tag{21}$$

where $\ddot{x}_j^{(g)} = \sum_{\{i:y^{(i)}=g\}} x_j^{(i)}$, ie, the sum of x_j in class g, and $\ddot{\mu}_j = \sum_{g=1}^{G} \mu_j^{(g)}$. The above conditional distributions are the standard results of the conditional posterior distributions of mean and variance of univariate normal distribution based on the semi-conjugate normal and inverse-gamma priors, which can be found from Gelfand et al. (1990). The conditional

distributions of w^μ and w^x are given by:

$$P(w^\mu \mid \boldsymbol{w}^\mu_{1:p}) \propto (w^\mu)^{p\,\alpha_1^\mu/2-\alpha_0^\mu/2-1} \exp\left\{-\sum_{j=1}^{p} \frac{\alpha_1^\mu}{2w_j^\mu}\, w^\mu\right\} \exp\left\{-\frac{\alpha_0^\mu\, w_0^\mu}{2w^\mu}\right\}, \tag{22}$$

$$P(w^x \mid \boldsymbol{w}^x_{1:p}) \propto (w^x)^{p\,\alpha_1^x/2-\alpha_0^x/2-1} \exp\left\{-\sum_{j=1}^{p} \frac{\alpha_1^x}{2w_j^x}\, w^x\right\} \exp\left\{-\frac{\alpha_0^x\, w_0^x}{2w^x}\right\}. \tag{23}$$

The above conditional distributions may not be sampled directly. We will apply Metropolis-Hasting method (see eg. Neal, 1993) with Gaussian proposal to draw samples of $\log(w^\mu)$ and $\log(w^x)$.

2.3 Bias-corrected Predictions with a Selected Subset of Features

2.3.1 A Principle for Correcting Feature Selection Bias

When p is very large (perhaps as large as tens or a hundred of thousands), training the model based on all the features with MCMC is inefficient, and accordingly (for pragmatic reasons) we intend to select a subset of features by some simple criterion measuring the relevance between the features and response, denoted by $R(\boldsymbol{x}_t^{(1:n)}, \boldsymbol{y}^{(1:n)})$, for $t = 1, \ldots, p$. The examples of such criteria include the sample correlation of the response with features and the F-statistic given by expression (24), and many others. There are also some Bayesian and non-Bayesian methods based on linear models for selecting features. However, these methods become infeasible in computation when p is larger than one thousand. Therefore such simple univariate screening methods are almost always applied before fitting a model (if one doesn't use other dimension reduction methods such as principle component analysis.) This practice was evidenced by the outcomes of 2003 NIPS feature selection competition (Guyon et al., 2006).

A widely used criterion to select features is F-statistic:

$$R_F(\boldsymbol{x}^{(1:n)}, \boldsymbol{y}^{(1:n)}) = \frac{\sum_{g=1}^{G} n_g(\bar{x}^{(g)} - \bar{x})^2/(G-1)}{\sum_{g=1}^{G}\sum_{i\in N_g}(x^{(i)} - \bar{x}^{(g)})^2/(n-G)}, \tag{24}$$

where $\bar{x}^{(g)}$ is the average of the $x^{(i)}$'s with $y^{(i)} = g$, ie, $\sum_{i\in N_g} x^{(i)}/n_g$, and $\bar{x}$ is the overall average $\sum_{i=1}^{n} x^{(i)}/n$. Here we have dropped the index for distinguishing different features, since this is a selection criterion applied to all features. Despite and because of the simplicity, F-statistic is used very often in practice, see for example Dudoit et al. (2002); Chen and

Lin (2006); Lal et al. (2006); Wu and Li (2006); Zhou et al. (2006), and many others. One way to select features is by fixing a threshold, γ, for the relevance measure R_F of a selected feature with the response. We then omit feature j from the feature subset if $R_F(\boldsymbol{x}_j^{(1:n)}, \boldsymbol{y}^{(1:n)}) \leq \gamma$, retaining those features with a greater degree of relevance to $\boldsymbol{y}^{(1:n)}$. We will assume that the features are renumbered so that the subset of retained features is $x_1, \ldots, x_k$. The determination of γ may be complicated, for example by looking at the p-value. However, no matter how the γ was chosen, once it was determined, we know that $R_F(\boldsymbol{x}_j^{(1:n)}, \boldsymbol{y}^{(1:n)}) \leq \gamma$ for $j = k+1, \ldots, p$. The other way to select features is by fixing the number of features we intend to retain, k, by considering computational burden. We then retain the top k features with highest relevance measures, and omit the others. After renumbering the features so that the subset of retained features is $x_1, \ldots, x_k$, we will set γ to $R_F(\boldsymbol{x}_k^{(1:n)}, y^{(1:k)})$. Again, here we know that $R_F(\boldsymbol{x}_j^{(1:n)}, \boldsymbol{y}^{(1:n)}) \leq \gamma$ for $j = k+1, \ldots, p$.

Omitting a large number, $p - k$, of features to model can drastically reduce the computation time, however, more care must be taken to the model for the retained features. One can fit the values of the k features we retained simply with a model, for example the one defined in Section 2.1, pretending that no feature selection has occurred. However, such *uncorrected methods*, are invalid from a theoretical point-view. Specifically, the overall degree of relevance between the response and the features has been exaggerated by the feature selection. *A correct way* is that when forming the posterior distribution for parameters of the model using a selected subset of features, we should condition not only on the values in the training set of the response and of the k features we retained, but also on the fact that the other $p-k$ features were omitted because their relevances with the response measured by R_F are less than γ. That is, the posterior distribution should be conditional on the following information:

$$\boldsymbol{y}^{(1:n)}, \quad \boldsymbol{x}_{1:k}^{(1:n)}, \quad \mathcal{S}_j : R_F(\boldsymbol{x}_j^{(1:n)}, \boldsymbol{y}^{(1:n)}) \leq \gamma \text{ for } j = k+1, \ldots, p. \tag{25}$$

An intuitive explanation of the above principle is that, the upward bias in estimating the strength of the relationship between the response and the features caused by the feature selection will be canceled by $P(\mathcal{S}_{(k+1):p})$, as it is larger when the strength is weaker. The formal justification is that, the posterior formed this way is based on *all the available information to a modeler*. Therefore, predictions for future cases with such a posterior distribution will be well-calibrated. As proved explicitly by Dawid (1982) and Li et al. (2008), if a model describes the actual data generation mechanism, and the actual values of the model parameters

are indeed randomly chosen according to an assumed prior, Bayesian inference of parameters conditional on all the available information *always* leads to well-calibrated predictions for future cases, on average with respect to the data and model parameters generated from the Bayesian model. In practice, of course we can never guarantee that our model for the data and our priors for the parameters are exactly correct for our problem. The corrected method is, however, expected to produce at least better calibrated predictions than the uncorrected method, as shown by real data examples in Section 4.

2.3.2 Bias-corrected Posterior Distribution

We will derive the bias-corrected posterior distribution of the model parameters when additionally conditioning on the information $\mathcal{S}_{(k+1):p}$.

The joint distribution of $x_j^{(1:n)}$ and $\mathcal{S}_{(k+1):p}$ given $y^{(1:n)}$ and all the relevant parameters is:

$$\prod_{j=1}^{k}\Big[P(\boldsymbol{x}_j^{(1:n)} \,|\, \boldsymbol{\mu}_j^{(1:G)}, w_j^x, \boldsymbol{y}^{(1:n)})\, P(\boldsymbol{\mu}_j^{(1:G)} \,|\, \nu_j, w_j^\mu)\, P(\nu_j \,|\, w^\nu)\, P(w_j^\mu \,|\, w^\mu)\, P(w_j^x \,|\, w^x)\Big] \times$$
$$\prod_{j=k+1}^{p}\Big[P(\mathcal{S}_j \,|\, \boldsymbol{\mu}_j^{(1:G)}, w_j^x, \boldsymbol{y}^{(1:n)})\, P(\boldsymbol{\mu}_j^{(1:G)} \,|\, \nu_j, w_j^\mu)\, P(\nu_j \,|\, w^\nu)\, P(w_j^\mu \,|\, w^\mu)\, P(w_j^x \,|\, w^x)\Big] \times$$
$$P(w^\mu)\, P(w^\nu)\, P(w^x). \tag{26}$$

From (26), we can see that the conditional distributions of the parameters, $\boldsymbol{\mu}_j^{(1:G)}$, ν_j, and w_j^μ for $j = 1, \ldots, k$ are the same as given by equations (17), (19), and (20). If we integrate away this set of parameters associated with the omitted features, and the second line of (26) becomes $p - k$ multiples of a value that is the same for all $j = k+1, \ldots, p$. In fact, as to be shown in Section 2.3.3, this value is unrelated to w^ν. We will denote this value by $C(w^\mu, w^x)$, and call it *adjustment factor*. With this notation, the second line of (26) becomes $C(w^\mu, w^x)^{\,p-k}$ with

$$C(w^\mu, w^x) = P(\mathcal{S}_j \,|\, w^\mu, w^x, \boldsymbol{y}^{(1:n)}). \tag{27}$$

The conditional distribution of w^ν is therefore not affected by $\mathcal{S}_{(k+1):p}$, still having a form similar to (21), but with only $\boldsymbol{\nu}_{1:k}$ used and p set to k. What's different from without considering $\mathcal{S}_{(k+1):p}$ is only a *bias-corrected* conditional distribution for (w^μ, w^x), which is

written explicitly as:

$$P(w^\mu, w^x \,|\, \boldsymbol{w}^\mu_{1:k}, \boldsymbol{w}^x_{1:k}, \mathcal{S}_{(k+1):p}) \;\; \propto \;\; P(w^\mu \,|\, \boldsymbol{w}^\mu_{1:k})\, P(w^x \,|\, \boldsymbol{w}^x_{1:k})\, C(w^\mu, w^x)^{\,p-k}, \tag{28}$$

where $P(w^\mu \,|\, \boldsymbol{w}^\mu_{1:k})$ is similar to (22), with only $\boldsymbol{w}^\mu_{1:k}$ used and p set to k, and $P(w^x \,|\, \boldsymbol{w}^x_{1:k})$ is similar.

It is crucial to note that we need to compute $C(w^\mu, w^x)$ only once no matter how many features were omitted. If the computation of $C(w^\mu, w^x)$ is efficient, MCMC training for the bias-corrected method can still enjoy the drastically reduction of time by omitting a large of number of features. In next section we will describe an efficient algorithm for this computation.

2.3.3 A Monte Carlo Method for Approximating the Adjustment Factor

A method for approximating $C(w^\mu, w^x)$ is based on precursors' work on computing the power function of one-way ANOVA.

Given $\boldsymbol{\mu}_j^{(1:G)}$, and $\boldsymbol{y}^{(1:n)}$, F-statistic in (24) has a non-central F distribution (Knight, 2000, pg. 411-416), with $G-1$ and $n-G$ degrees of freedom, and a non-centrality parameter $2\,\Lambda(\boldsymbol{\mu}_j^{(1:G)}, w_j^x)$, where

$$\Lambda(\boldsymbol{\mu}_j^{(1:G)}, w_j^x) = \frac{D(\boldsymbol{\mu}^{(1:G)})}{2\, w_j^x}, \quad D(\boldsymbol{\mu}^{(1:G)}) = \sum_{g=1}^{G} n_g(\mu_j^{(g)} - \tilde{\mu}_j)^2, \quad \tilde{\mu}_j = \frac{\sum_{g=1}^{G} n_g \mu_j^{(g)}}{n}. \tag{29}$$

We therefore have

$$P(\mathcal{S}_j \,|\, \boldsymbol{y}^{(1:n)}, \boldsymbol{\mu}_j^{(1:G)}, w_j^x) = P\left(F_{(G-1,\, n-G,\, 2\,\Lambda(\boldsymbol{\mu}_j^{(1:G)}, w_j^x))} \le \gamma \right) \equiv c(\Lambda(\boldsymbol{\mu}_j^{(1:G)}, w_j^x)), \tag{30}$$

where $F_{(G-1,\, n-G,\, 2\,\Lambda(\boldsymbol{\mu}_j^{(1:G)}, w_j^x))}$ denotes a random variable with a non-central F distribution.

The function $c(\Lambda)$ can be computed easily using the fact that a non-central χ^2 distribution can be expressed as an infinite mixture of central χ^2 distributions with Poisson weights (Knight, 2000). With χ^2_ν denoting a random variable having central χ^2 distribution with degree freedom ν, we can now express $c(\Lambda)$ as:

$$c(\Lambda) \;\; = \;\; \sum_{\ell=0}^{+\infty} f_\ell \, \frac{\exp(-\Lambda)\, \Lambda^\ell}{\ell\,!}, \tag{31}$$

where

$$f_\ell = P\left(\frac{\chi^2_{G-1+2\ell}/(G-1)}{\chi^2_{n-G}/(n-G)} \leq \gamma\right) = P\left(\frac{\chi^2_{G-1+2\ell}/(G-1+2\ell)}{\chi^2_{n-G}/(n-G)} \leq \frac{\gamma\,(G-1)}{G-1+2\ell}\right). \tag{32}$$

Here, f_ℓ can be computed with the CDF of central F distribution. From (32), we can see that f_ℓ decreases to 0 as ℓ tends to $+\infty$. Therefore, $c(\Lambda)$ is a decreasing function of Λ. In addition, since Poisson weights are always between 0 and 1, we can truncate the above infinite summation by setting a threshold for f_ℓ, while controlling a same tolerable error for all Λ.

To obtain the adjustment factor — $C(w^\mu, w^x)$, we need to integrate $c(\Lambda(\boldsymbol{\mu}_j^{(1:G)}, w_j^x))$ with respect to the prior distribution of $\Lambda(\boldsymbol{\mu}_j^{(1:G)}, w_j^x)$, which is induced by the priors for $\boldsymbol{\mu}_j^{(1:G)}$ and w_j^x, conditional on w^μ, w^x, and w^ν. It is useful to note that the prior distribution of $\Lambda(\boldsymbol{\mu}_j^{(1:G)}, w_j^x)$ is unrelated to ν_j, and so neither to w^ν. To show this, we will re-parameterize $\boldsymbol{\mu}_j^{(1:G)}$ and w_j^x as follows:

$$\boldsymbol{\mu}_j^{(1:G)} = \boldsymbol{m}_j^{(1:G)} \sqrt{s_j^\mu}\sqrt{w^\mu} + \nu_j, \qquad w_j^x = s_j^x\, w^x, \tag{33}$$

where,

$$\boldsymbol{m}_j^{(1:G)} \overset{\text{IID}}{\sim} N(0,1), \quad s_j^\mu \sim \text{IG}(\alpha_1^\mu/2, \alpha_1^\mu/2), \quad s_j^x \sim \text{IG}(\alpha_1^x/2, \alpha_1^x/2). \tag{34}$$

Then it can be shown that

$$\Lambda(\boldsymbol{\mu}_j^{(1:G)}, w_j^x) = \frac{1}{2}\, D\left(\boldsymbol{m}_j^{(1:G)}\right)\ \frac{s_j^\mu}{s_j^x}\ \frac{w^\mu}{w^x}. \tag{35}$$

From the above expression, we can see readily that the prior distribution of $\Lambda(\boldsymbol{\mu}_j^{(1:G)}, w_j^x)$ is not related to w^ν. Therefore we denote the adjustment factor by a function of only w^μ and w^x — $C(w^\mu, w^x)$. It is explicitly written as:

$$C(w^\mu, w^x) = E_{\boldsymbol{m}_j^{(1:G)},\, s_j^\mu,\, s_j^x}\left(c\left(\frac{1}{2}\, D\left(\boldsymbol{m}_j^{(1:G)}\right)\ \frac{s_j^\mu}{s_j^x}\ \frac{w^\mu}{w^x}\right)\right). \tag{36}$$

As shown above, $c(\Lambda)$ is a decreasing function of Λ. Therefore, when w^μ/w^x is larger, $C(w^\mu, w^x)$ is smaller, resulting in fewer features to be omitted using γ as threshold. This explains more precisely that the ratio w^μ/w^x controls the overall strength of the relationship between the features and response. The method presented in this paper corrects for the

upward bias in the posterior of w^μ/w^x based only on the retained features with $C(w^\mu, w^x)$, as it is a decreasing function of w^μ/w^x.

The expression of $C(w^\mu, w^x)$ in (36) also indicates that we can use Monte Carlo method to estimate it at different values of w^μ and w^x, with a *common* pool of i.i.d. random samples of $\boldsymbol{m}_j^{(1:G)}$, s_j^μ and s_j^x. There are two advantages of doing this. First, it saves computation time. We need to draw samples of $\boldsymbol{m}_j^{(1:G)}$, s_j^μ, and s_j^x and compute $D\left(\boldsymbol{m}_j^{(1:G)}\right) s_j^\mu/s_j^x$ only once, regardless of how many iterations of Markov chain sampling are to be run. More importantly, it improves the accuracy (measured by mean square error) of estimating the ratios of $C(w^\mu, w^x)$ at different values, which are needed in simulating Markov chain for updating w^μ and w^x, since two random variables with two different sets of values of w^μ and w^x whose expectations are computed with (36) are positively correlated. One can show this explicitly by approximating the mean square error of a ratio of two random variables with a Taylor expansion of the ratio at their expected values.

The following demonstrations with simulated data and SRBCT data are based on a previous simpler model in which w_j^μ are all the same. I am still working on revising them based on the model presented here. The details of conclusions may be different, but the main idea is expected to the same, ie, our method can avoid feature selection.

3 Demonstrations with Simulated Data

Fixing $\tau^\nu = 100, \tau = 100, \alpha^x = 4, w^x = 1, G = 6$, and $p = 4000$, we generated a data set of 5200 cases from the model described in Section 2.1, with the values of response drawn from uniform distribution over the set $\{1, \ldots, G\}$. We randomly selected 200 cases as training set, and left the remaining 5000 as test cases to evaluate the predictive performance. With the training cases, we then selected four subsets of features, containing 10, 50, 200, and 1000 features, by comparing their values of F-statistic, giving thresholds 6.20, 4.41, 3.15, and 1.79 respectively for these four subsets. These are the values of γ used by the bias-correction method when computing the adjustment factor of equation (36).

In training the models with MCMC, we set the prior as follows: for $\boldsymbol{\psi}$, $c_1 = 1, \ldots, c_G = 1$, for , $\alpha^x = 4$ and $w^x = 1$, and for τ and τ^ν, $\alpha = \alpha^\nu = 1.5$, and $w = w^\nu = 0.01$. When bias-correction method is applied, for sampling $\log(\tau)$, we used Metropolis method with Gaussian proposal centering at previous state and with standard deviation 0.5. For each iteration of Gibbs sampling, this Metropolis update was applied 5 times. In computing the adjustment

factor with approximation (36), the upper bound of l was set to include f_l larger than e^{-10}, and number of samples of $\mathcal{Z}$ and $\mathcal{T}^x$ was set as 1000. These settings are adequate for this problem, different settings may be necessary in other problems.

We ran 6000 iterations of Gibbs sampling with settings as above to draw samples from the posterior distribution of $\boldsymbol{\mu}_{1:k}, \boldsymbol{\nu}_{1:k}, \boldsymbol{\tau}^x_{1:k}, \tau, \tau^\nu$, with and without correction for selection bias. The first 750 iterations were omitted, and every 15th iteration afterwards was used to make Monte Carlo estimations for test cases. We obtained the predictive probabilities of y equal to each of $g = 1, \ldots, 6$ for 5000 test cases. We examined these predictive probabilities to demonstrate the bias-correction method. All of the computation in this paper were carried out with an R add-on package written by myself.

We first directly plotted the predictive probabilities of $y = 1$ for only 500 test cases. Figure 2 plots the predictive probabilities given by the methods with correction for selection bias against without correction. From it, we see clearly that the predictive probabilities without bias-correction tends to be closer to 0 and 1, ie, are more confident than those with bias-correction. This confidence is however incorrect. Let's look at how well calibrated the predictive probabilities of $y = 1$ are in Table 1. We grouped the 5000 test cases into 10 categories according to the first decimals of predictive probabilities, ie, the predictive probabilities of $y = 1$ in category C are between $C/10$ and $(C + 1)/10$, for $C = 0, \ldots, 9$. For those test cases in category C, We calculated the average of predictive probabilities, namely "Pred" in Table 1, and the actual fraction of $y = 1$, namely "actual" in Table 1. If the predictive probabilities are well-calibrated, the "Pred" and "Actual" in each category should be close if the number of cases in this category isn't very small (Dawid, 1982). From Table 1, it is clear that without correction for selection bias, the values in "Pred" are not close to those in "Actual". For example the values of "Pred" are incorrectly close to 0 in category $C = 0$, and to 1 in category $C = 9$. After correcting for selection bias, the "Pred" and "Actual" are fairly close, indicating that the predictive probabilities are well-calibrated.

We now examine the effect when the poorly-calibrated predictive probabilities are used in practice to make single-valued guesses, given certain choice of loss function. We will first introduce the concept of *expected loss* for a predictive method, as it is not seen very often in the literature. Suppose we have defined a loss function for our problem, denoted by $L(y \to y')$, which means the loss incurred when we guess the true value y with a possibly wrong value y'. From a predictive distribution $\hat{P}(y \,|\, \mathcal{X})$ ($\mathcal{X}$ represents the covariates and other information such as $\mathcal{S}$), one can obtain a prediction for Y, denoted by $\hat{Y}(\mathcal{X})$. The

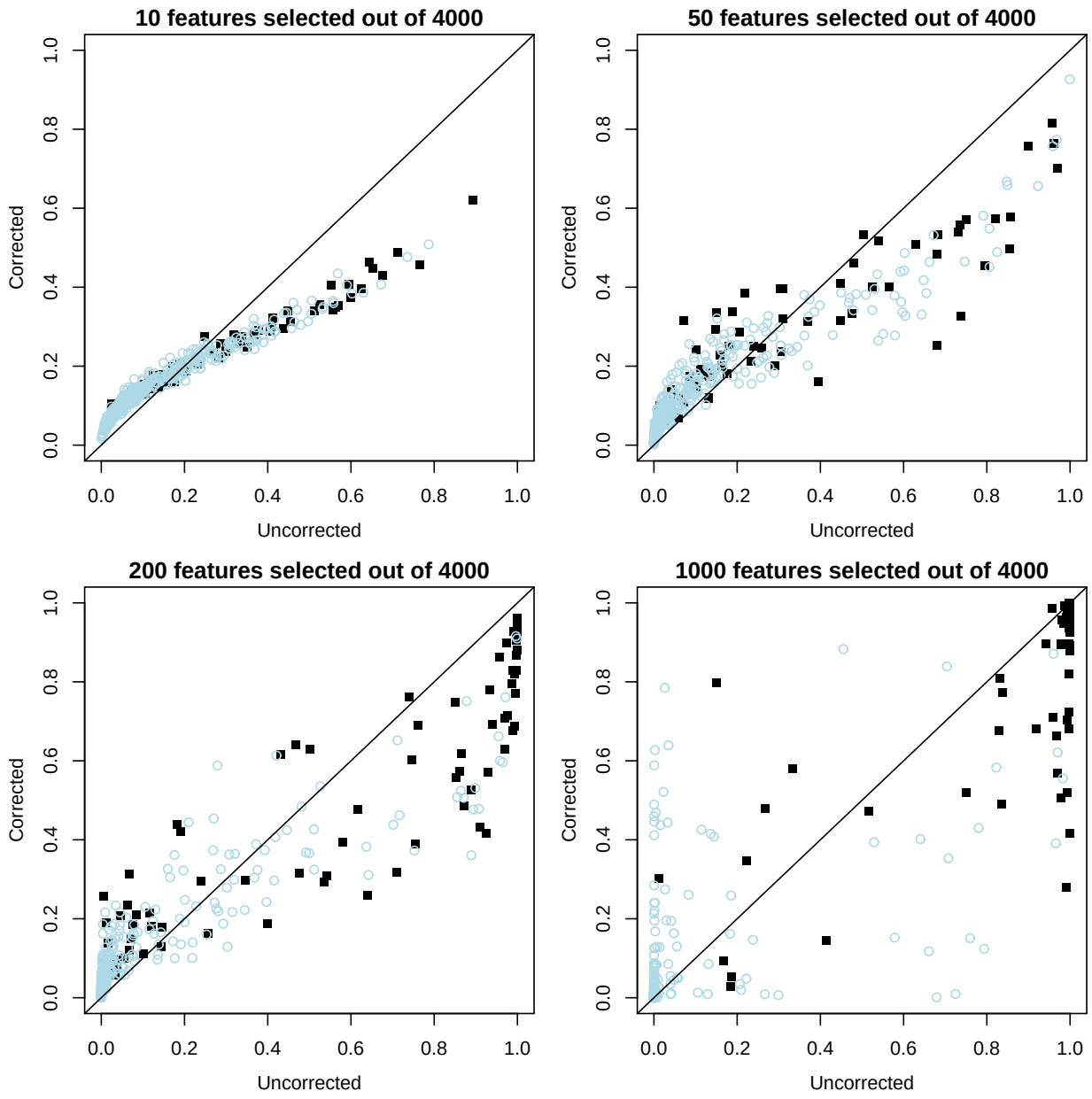

Figure 2: Scatter plots of predictive probabilities of $y = 1$ for 500 test cases simulated from a Bayesian Gaussian classification model, computed by methods with and without correction for selection bias. The solid squares indicate cases with $y = 1$ and the hollow circles indicate cases with $y \neq 1$. Without correction for selection bias, the predictive probabilities tend to be closer to 0 and 1.

	10 features selected out of 4000						50 features selected out of 4000					
	Corrected			Uncorrected			Corrected			Uncorrected		
C	#	Pred	Actual	#	Pred	Actual	#	Pred	Actual	#	Pred	Actual
0	1000	0.072	0.059	1888	0.048	0.090	1968	0.048	0.058	2621	0.023	0.087
1	1827	0.145	0.150	968	0.143	0.175	962	0.144	0.167	436	0.143	0.167
2	839	0.243	0.257	484	0.243	0.240	442	0.243	0.253	237	0.247	0.308
3	237	0.342	0.350	300	0.344	0.310	266	0.347	0.380	158	0.347	0.316
4	80	0.438	0.525	162	0.443	0.290	162	0.443	0.432	123	0.451	0.398
5	15	0.524	0.600	90	0.550	0.411	97	0.544	0.557	107	0.552	0.336
6	2	0.618	1.000	63	0.643	0.476	50	0.646	0.580	83	0.650	0.422
7	0	–	–	36	0.740	0.389	28	0.757	0.750	81	0.747	0.543
8	0	–	–	9	0.846	1.000	19	0.844	0.895	68	0.846	0.544
9	0	–	–	0	–	–	6	0.926	0.833	86	0.954	0.698

	200 features selected out of 4000						1000 features selected out of 4000					
	Corrected			Uncorrected			Corrected			Uncorrected		
C	#	Pred	Actual	#	Pred	Actual	#	Pred	Actual	#	Pred	Actual
0	2490	0.029	0.033	2941	0.010	0.061	3025	0.008	0.011	3126	0.003	0.011
1	501	0.144	0.180	187	0.145	0.257	131	0.143	0.130	54	0.148	0.222
2	263	0.248	0.274	104	0.246	0.288	81	0.246	0.259	41	0.244	0.244
3	197	0.351	0.391	94	0.347	0.287	49	0.349	0.286	32	0.351	0.281
4	143	0.444	0.455	79	0.453	0.342	53	0.447	0.264	38	0.450	0.342
5	120	0.549	0.608	74	0.550	0.405	51	0.553	0.549	20	0.552	0.450
6	91	0.648	0.670	86	0.647	0.547	53	0.652	0.698	22	0.646	0.364
7	79	0.754	0.759	70	0.751	0.414	50	0.748	0.700	27	0.758	0.444
8	66	0.844	0.864	92	0.860	0.598	78	0.851	0.821	53	0.857	0.679
9	50	0.943	0.940	273	0.969	0.777	429	0.979	0.981	587	0.991	0.925

	Complete data		
C	#	Pred	Actual
0	3169	0.003	0.009
1	52	0.147	0.077
2	31	0.241	0.355
3	24	0.348	0.417
4	22	0.443	0.364
5	24	0.546	0.375
6	23	0.644	0.565
7	26	0.751	0.577
8	42	0.850	0.667
9	587	0.990	0.952

Table 1: Comparison of calibration for predictions found with and without correction for selection bias, on data simulated from a Bayesian naive Bayes Gaussian classification model. Results are shown with four subsets of features and with the complete data (for which no correction is necessary). The test cases were divided into 10 categories by the first decimal of the predictive probability of class 1, which is indicated by the 1st column "C". The table shows the number of test cases in each category for each method ("#"), the average predictive probability of class 1 for cases in that category ("Pred"), and the actual fraction of these cases that were in class 1 ("Actual").

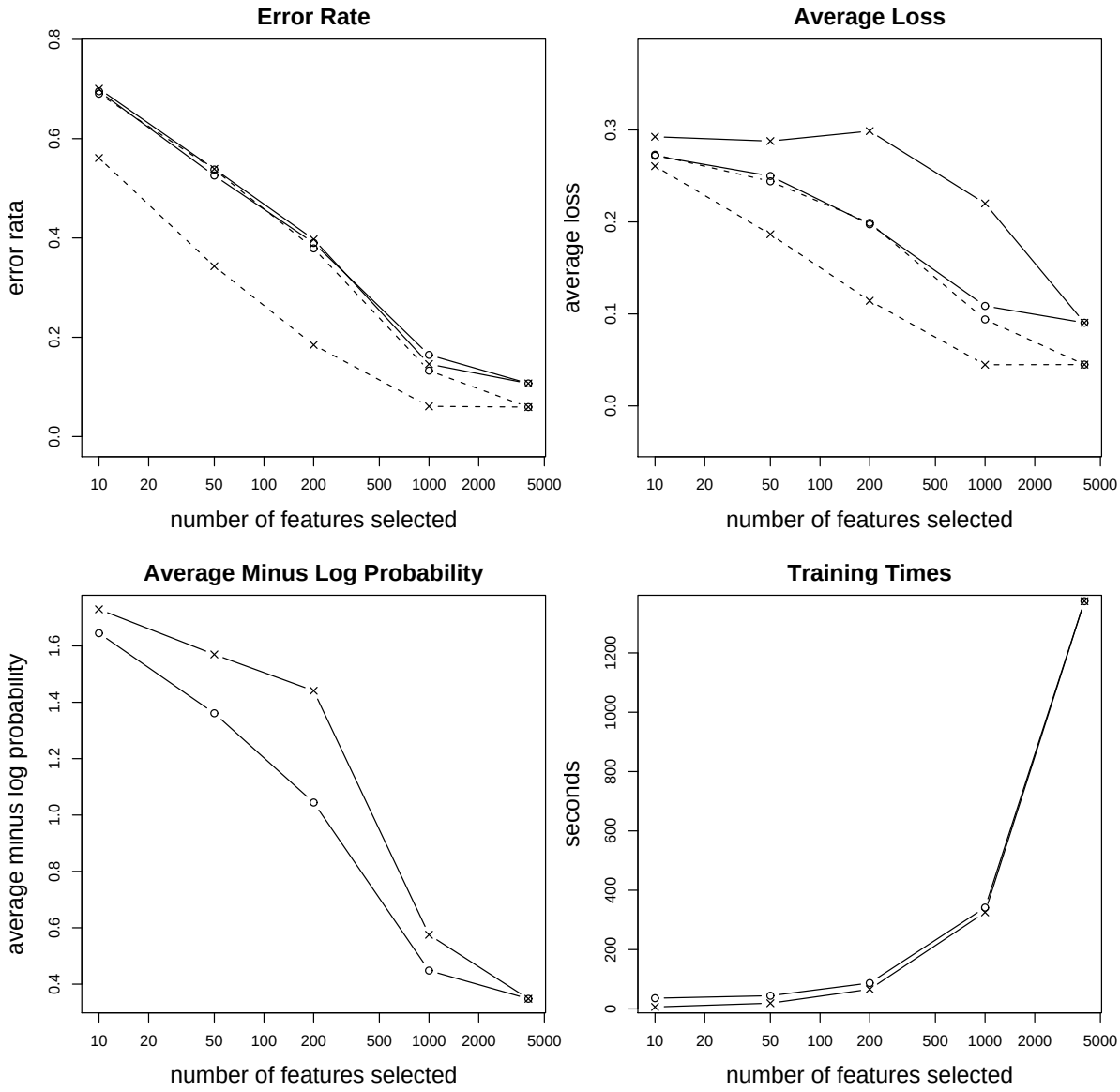

Figure 3: These plots compare the prediction methods with and without correction for selection bias in terms of, error rate (top-left), average of loss based on a loss function whose values are drawn from $\exp(N(0, 2^2))$ (top-right), average of the minus log probabilities of observing the actual symbols (bottom-left), and the times for training the models with Gibbs sampling (bottom-right), on data simulated from a Bayesian naive Bayes Gaussian classification model. The lines with $\times$ show the methods without correction for selection bias, the lines with $\circ$ show the methods with correction. On the top two plots, the dashed lines show the expected error rates (left), or the average of expected loss (right). (Note that the curves of actual error rates and training times overlap for methods with and without correction for selection bias.)

expected loss of $\hat{Y}(\mathcal{X})$ is:

$$\widehat{\mathrm{EL}}(\hat{Y}(\mathcal{X})) = \hat{E}(L(Y \to \hat{Y}(\mathcal{X})) = \sum_{y=1}^{G} \hat{P}(y|\mathcal{X})\, L(y \to \hat{Y}(\mathcal{X})) \tag{37}$$

Rationally, given $\hat{P}(y \,|\, \mathcal{X})$, the choice of $\hat{Y}(\mathcal{X})$ is the y minimizing (37) over all possible y. The average of expected loss $\widehat{\mathrm{EL}}(\hat{Y}(\mathcal{X}))$ across n test cases is:

$$\frac{1}{n} \sum_{i=1}^{n} \widehat{\mathrm{EL}}(\hat{Y}(\mathcal{X}^{(i)})) \tag{38}$$

This quantity is used to measure the uncertainty of predictions based on $\hat{P}$.

If a predictive distribution is equal to the true conditional distribution $P(y \,|\, \mathcal{X})$ (in this case we say $\hat{P}$ is well-calibrated), one can easily see from the iterative expectation formula that the quantity in (38) is a good estimate of the actual loss: $E(L(Y \to \hat{Y}(\mathcal{X}))$ (provided that n is fairly large). We can therefore compare the average of expected loss and the average of actual losses across test cases to see whether a predictive distribution is well-calibrated or not. We will compare the average of expected loss with the average of actual loss for each method, and also compare the actual loss between uncorrected and corrected methods.

Let's first consider the 0-1 loss $L(y \to y') = I(y \neq y')$. For this loss function, the average of loss is the familiar error rate, the fraction of cases for which we make wrong guess. We will call the average of expected loss as *expected error rate*. We compared the actual error rates of the corrected and uncorrected methods, when different subset of features are used, shown by the solid lines on the upper-left plot in Figure 3. Surprising to some readers, the actual error rates are very close for corrected and uncorrected methods. This fact was confirmed also by Singhi and Liu (2006), and by Li et al. (2008). This is good news for practitioners if the error rate is the only issue in the problem they consider. However, this is often not the case. In particular, people in practice usually need the uncertainty estimate accompanying single-valued guesses, ie, the expected error rate. We plotted the expected error rates in Figure 3 with dashed lines. If the predictive probabilities are well-calibrated, the expected error rates are close to the actual error rates. From Figure 3, for the corrected methods, they are fairly close, but they are not for uncorrected methods, with the expected error rates consistently smaller than the actual error rates, indicating the predictive probabilities are overconfident, giving smaller estimate of uncertainty.

In situations where loss incurred by different errors are not the same (eg, missing detecting cancer in a patient incurs more loss than erroneously detecting cancer in a normal patient), poorly-calibrated predictive probabilities could result in more loss on average. We experimented with this by generating the values of $L(y \to y')$ from distribution $\exp(N(0, 2^2))$ when $y \neq y'$, and set $L(y \to y) = 0$. The average of actual loss are shown for corrected and uncorrected methods on the top-right plot in Figure 3, from which we see that the average of actual loss by uncorrected methods are consistently larger than corrected methods. The average of expected loss are also shown with dashed lines on the same plot in Figure 3. For uncorrected methods, the average of expected loss are again consistently smaller than the average of actual loss, whereas for corrected methods, they stay fairly close.

From Figure 3, as well as Table 1, we see that when all of 4000 features were used, some overconfidence in predictive probabilities also occurs, with the averages of expected loss a little smaller than the averages of actual loss, and "pred" more extreme than "actual". In some other experiments (which are not shown here) we have also observed that using all features might even result in predictions with higher error rate and average loss. We suspected that this was because of that Markov chains got trapped in some local modes and therefore failed to explore parameter space thoroughly, as the number of parameters is so large. We have experimented with smaller number of total features, such as 1000, and/or with larger number of training cases, such as 1000, and haven't seen such bias or worse predictions. Since such data sets do not look like real data sets, we do not use them as illustrative examples here.

Prediction methods can be evaluated also by the average of minus log probabilities (AMLP) of observing the actual values of responses: $(1/n) \sum_{i=1}^{n} \big[-\log(\hat{P}(y^{(i)} \mid \mathcal{X}^{(i)})) \big]$, where $y^{(i)}$ denotes the true value of response for ith test case, and $\mathcal{X}^{(i)}$ is the information used to predict $y^{(i)}$ by $\hat{P}$. This criterion penalizes heavily those probabilities of actual values close to 0. The plots of AMLPs for uncorrected and corrected methods are shown in Figure 3, indicating that corrected methods perform better than uncorrected methods.

Direct explanation of feature selection bias is that the posterior distribution of τ favors incorrectly smaller values than the true value generating the data set. We plotted the Markov chain traces of $\log(\tau)$ in Figure 4. With correction for selection bias, the posterior distribution of $\log(\tau)$ favors values around the true value $\log(100)$, whereas without correction, the posterior distribution concentrates around smaller values than the true value, which was displayed by that the Markov chain traces never touch the true value.

From Figure 3 we can see that the extra time for computing adjustment factor is very little, nearly negligible. Compared to using all 4000 features, training times with a selected subset of features are much less. Though we do not plot prediction times here, one can believe that prediction times will increase rapidly with number of features selected, due to computation of difference between feature value and mean parameter μ. Combining with previous observation that Markov chains for training models with a large number of features more easily get trapped in local modes, which may yield poorly-calibrated, and even less accurate predictions, we can see clearly the benefits of selecting a smaller number of features, provided that the feature selection bias is corrected effectively, as we do here.

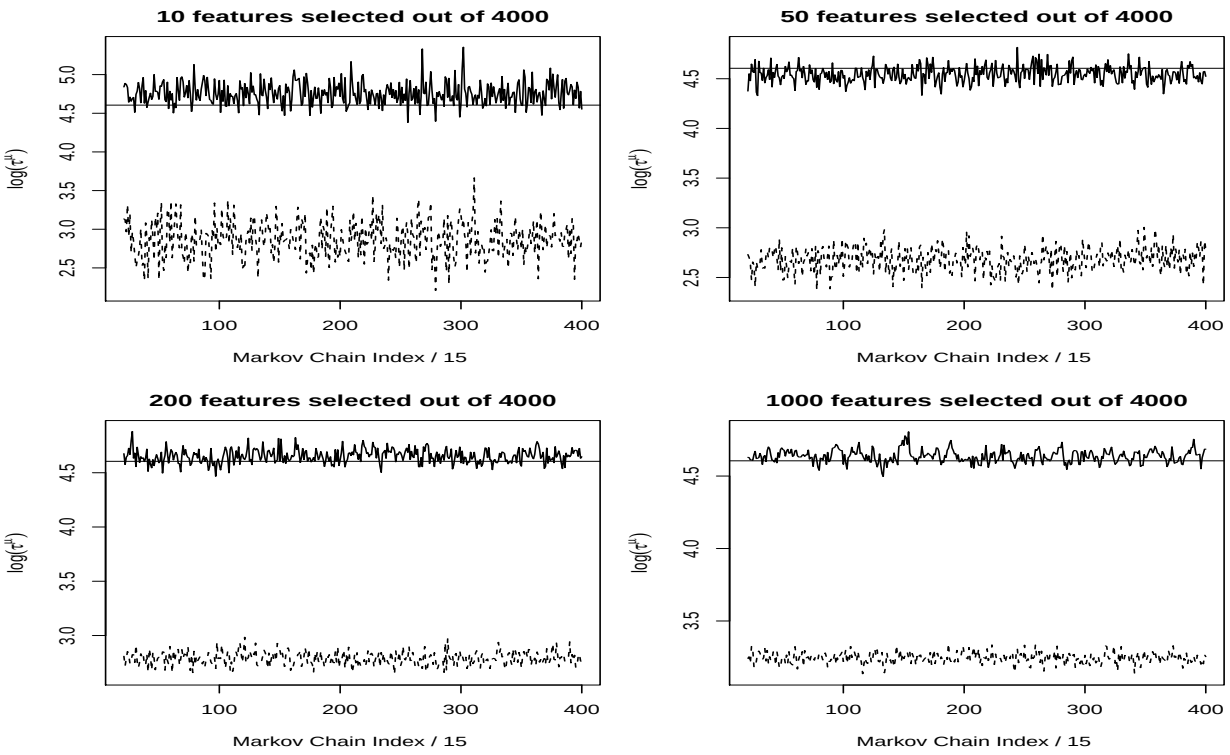

Figure 4: Markov chain traces of $\log(\tau)$, on data simulated from a Bayesian naive Bayes Gaussian classification model. The dashed lines show methods without correction for selection bias, and the solid lines show methods with correction. The horizontal straight lines indicate the true values of $\log(\tau)$ generating the data, which is $\log(100)$. Without correction for selection bias, the Markov chains of $\log(\tau)$ move around some values smaller than the true value.

4 Applications to Gene Expression Data

We also tested the bias-correction method on a gene expression data related to small round blue cell tumors (SRBCT) of childhood, which was first released along with Khan et al. (2001). SRBCTs include neuroblastoma (NB), rhabdomyosarcoma (RMS), non-Hodgkin lymphoma (NHL) and the Ewing family of tumors (EWS), are so named because of their similar appearance on routine histology. However, accurate diagnosis of SRBCTs is essential because the treatment options, responses to therapy and prognoses vary widely depending on the diagnosis. As their name implies, these cancers are difficult to distinguish by light microscopy, and currently no single test can precisely distinguish these cancers. Khan et al. (2001) proposed approaches to diagnosing these four types of tumors from gene expression data. The released data set (available from `http://research.nhgri.nih.gov/microarray/Supplement/`) contains expression data of 2308 genes on 83 patients who have been categorized into one of these tumors with conventional diagnosis.

The assumption that the expression levels for different genes are independent given the response may not be realistic. When the predictor variables are correlated given the response, and the covariance matrices of the original variables are believed to the same for different classes, we can apply a linear transformation to the original variables such that the resulting variables are nearly independent. Suppose that given $y^{(i)} = g$, the original variables $z_1^{(i)}, \ldots, z_p^{(i)}$ have a multivariate distribution $N(\tilde{\boldsymbol{\mu}}^{(g)}, \Sigma)$. Using Choleski decomposition, we can factor Σ as LL'. We can then construct new variables $x_1^{(i)}, \ldots, x_p^{(i)}$ with L by letting $(x_1^{(i)}, \ldots, x_p^{(i)})' = L^{-1}\,(z_1^{(i)}, \ldots, z_p^{(i)})'$. Given $y^{(i)} = g$, the transformed variables $x_1^{(i)}, \ldots, x_p^{(i)}$ have a multivariate distribution $N(L^{-1}\tilde{\boldsymbol{\mu}}^{(g)}, I_p)$, where I_p is the p-dimensional identity matrix. We can estimate this unknown Σ using some non-Bayesian methods. In the real data example presented here, We shrink the usual pooled estimation S towards the diagonal matrix with a reasonable choice of λ: $\hat{\Sigma}(\lambda) = \frac{S+\lambda\,\mathrm{diag}(S)}{1+\lambda}$, where $\mathrm{diag}(S)$ is a matrix with the diagonal elements equal to those of S and off-diagonal elements equal to 0. This estimator keeps the usual unbiased estimates of variances unchanged, but shrinks the correlations towards 0 by a factor of $1/(1+\lambda)$.

We divided these 2038 genes randomly into 10 almost equal groups, producing 10 small data sets, each with about 204 features, as well as the same tumor indicators. We applied the corrected and uncorrected methods separately to each of these 10 data sets, allowing some

assessment of variability when comparing performance. For each of these 10 data sets, we used 20-fold cross-validation to obtain predictive probabilities for the class in the 83 cases. In this cross-validation procedure, the cases are divided into 20 almost equal subsets, in turn we left out one of the 20 subsets of cases as test cases, treated the remaining 19 subsets as training cases. For each of such splitting of 83 cases, we estimated pooled variance with training cases, then used it to transform the whole data set with Choleski decomposition, setting $\lambda = 10$, next selected the 10 features with the largest F-statistic in training cases, and finally ran MCMC to find the predictive probabilities for the left-out test cases, with and without bias correction. Note that here we have made data transformation and feature selection "internal" to cross-validation (Lecocke and Hess, 2004), ie, without using any information from test cases in performing data transformation and feature selection. In running each Markov chain, we used the same prior distributions except setting $\alpha^x = 2$ and $w^x = 0.3$, and the same computational methods, as in Section 3.

Using the predictive probabilities produced as above, we compared corrected and uncorrected methods. The results are shown by Figure 5. The top two plots show that our bias correction method reduces optimistic bias in the predictions. For each of the 10 data sets, this plot shows the actual error rate against the error rate expected from the predictive probabilities. For all 10 data sets, the expected error rate with the uncorrected method is substantially less than the actual error rate. This optimistic bias is reduced by our bias-correction method, though it is not eliminated entirely. The remaining bias presumably results from the failure in this data set of the models we assume here, especially the assumption that features are independent given class. The bottom-left plot in Figure 5 shows the averages of actual loss based on a random loss functions drawn from $\exp(N(0, 2^2))$, and we can see that corrected methods made much less or no much more loss on average than uncorrected methods. The bottom-right plot shows the average of minus log probabilities of observing true responses, and again we can see that with correction of selection bias, the predictions are much improved.

We have also applied the classification algorithms with and without correction for selection bias on the whole data set with certain numbers of genes selected with F-statistic. Since using all features this data set makes the classification very simple, when more than 50 genes are included, both corrected and uncorrected methods performed almost identically, giving an error rate of 0.024, ie, 2 out of 83 cases were misclassified, and giving similar expected error rates. This error rate is the same as the result reported by Tibshirani et al. (2002).

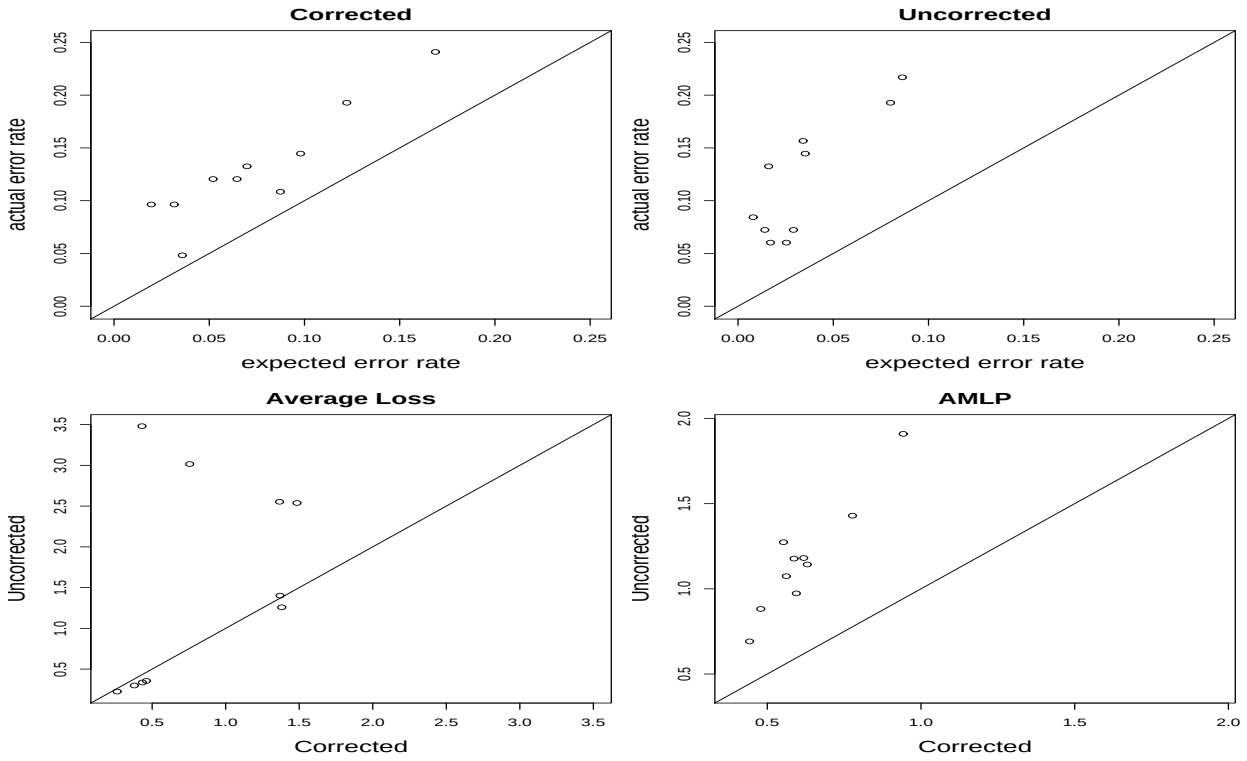

Figure 5: Comparison of corrected and uncorrected methods on small round blue cell tumors (SRBCT) gene expression data. The top two plots compare the expected error rates and actual error rates separately for corrected and uncorrected methods. The bottom two plots compare corrected and uncorrected methods in terms of averages of actual loss based on a random loss function drawn from $\exp(N(0, 2^2))$ and averages of minus log probabilities of observing true responses.

5 Conclusions and Future work

In this paper, we show how we can calibrate predictions based on a selected subset of features modeled by Bayesian Gaussian classification models. Specifically we have shown that we can efficiently compute the adjustment factor required to avoid feature selection bias — the probability of a feature being omitted by F-statistic, based on the precursors' work on computing the power function of ANOVA. we have used a simulated data set to show that after correcting for selection bias the predictive probabilities for future cases are well-calibrated. We have also tested the bias-correction method with a gene expression data and found that it does reduce the feature selection bias and yield better predictions.

The method presented here can be applied to a wide variety of practical problems,

however it can be further improved. First, a more reasonable prior for $\mu_j^{(g)}$ given ν_j in high-dimensional problems may be some distribution with heavier tail than Gaussian distribution, since we believe in such problems most of the features are useless in predicting the response, while a few of them may be very useful. Student's t distributions with small degree of freedom or Cauchy distributions are attractive. The difficulty of using these distributions is that we cannot directly draw samples of $\mu^{(g)}$ given others as we do here, we therefore may have to use some Markov chain sampling methods, which may hurt the sampling efficiency, such as getting trapped in local modes more easily. However, it is noticed that using these distributions doesn't increase the difficulty of computing the adjustment factor, for which one simply turns to draw samples of $\mathcal{Z}$ from t or Cauchy distributions in approximating (36) with Monte Carlo method. Second, adjusting feature selection bias by modeling features explicitly may be unnecessarily expensive. Another approach is to directly modify the posterior distribution of some hyperparameters controlling the overall relationship between features and response, such as τ here, in light of number of features omitted and threshold used. This modification may need to use some unknown parameters which could be determined by minimizing the difference between the expected error rate and actual error rate.

References

Ambroise, C. and McLachlan, G. J. (2002), "Selection Bias in Gene Extraction on the Basis of Microarray Gene-expression Data," *PNAS*, 99, 6562–6566. 3

Chen, Y. W. and Lin, C. J. (2006), "Combining SVMs with various feature selection strategies," *Feature Extraction: Foundations and Applications, vol 207 of Studies in Fuzziness and Soft Computing*, 315–324. 11

Dawid, A. P. (1982), "The well-calibrated Bayesian," *Journal of the American Statistical Association*, 77, 605–610. 12, 17

Dudoit, S., Fridlyand, J., and Speed, T. P. (2002), "Comparison of discrimination methods for the classification of tumors using gene expression data," *Journal of the American Statistical Association*, 97, 77–87. 6, 11

Gelfand, A. E., Hills, S. E., Racine-Poon, A., and Smith, A. F. M. (1990), "Illustration of Bayesian inference in normal data models using Gibbs sampling," *Journal of the American Statistical Association*, 972–985. 10

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (2004), *Bayesian Data Analysis*, Texts in Statistical Science, Chapman and Hall/CRC. 7

Golub, T., Slonim, D., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J., Coller, H., Loh, M., and Downing, J. (1999), "Molecular classification of cancer:class discovery and class prediction by gene expression monitoring," *Science*, 286, 531–537. 6

Guyon, I., Gunn, S., Nikravesh, M., and Zadeh, L. A. (2006), *Feature Extraction: Foundations and Applications*, vol. 207 of *Studies in Fuzziness and Soft Computing*, Springer. 3, 11

Hurvich, C. M. and Tsai, C.-L. (1990), "The Impact of Model Selection on Inference in Linear Regression," *The American Statistician*, 44, 214–217. 4

Khan, J., Wei, J. S., Ringnr, M., Saal, L. H., Ladanyi, M., Westermann, F., Berthold, F., Schwab, M., Antonescu, C. R., and Peterson, C. (2001), "Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks," *Nature Medicine*, 7, 673–679. 2, 24

Knight, K. (2000), *Mathematical Statistics*, Texts in Statistical Science, Chapman and Hall/CRC. 14

Lal, T. N., Chapelle, O., and Scholkopf, B. (2006), "Combining a filter method with SVMs," *Feature Extraction: Foundations and Applications, vol 207 of STUDIES IN FUZZINESS AND SOFT COMPUTING*, 439–445. 12

Lecocke, M. L. and Hess, K. (2004), "An Empirical Study of Optimism and Selection Bias in Binary Classification with Microarray Data," UT MD Anderson Cancer Center Department of Biostatistics Working Paper Series. 3, 25

Li, L., Zhang, J., and Neal, R. M. (2008), "A Method for Avoiding Bias from Feature Selection with Application to Naive Bayes Classification Models," *Bayesian Analysis*, 3, 171–196. 4, 5, 12, 21

Neal, R. M. (1993), "Probabilistic Inference using Markov Chain Monte Carlo Methods," Tech. rep., Dept. of Computer Science, University of Toronto. 9, 11

Raudys, S., Baumgartner, R., and Somorjai, R. (2005), "On Understanding and Assessing Feature Selection Bias," *Artificial Intelligence in Medicine*, 468–472. 3

Shen, X., Huang, H.-C., and Ye, J. (2004), "Inference After Model Selection," *Journal of the American Statistical Association*, 99, 751–762. 4

Singhi, S. K. and Liu, H. (2006), "Feature Subset Selection Bias for Classification Learning," *Proceedings of the 23rd International Conference on Machine Learning*, 849–856. 3, 4, 21

Tibshirani, R., Hastie, T., Narasimhan, B., and Chu, G. (2002), "Diagnosis of multiple cancer types by shrunken centroids of gene expression," *Proceedings of the National Academy of Sciences*, 99, 6567. 2, 25

Wang, R. and Lagakos, S. W. (2009), "Inference after variable selection using restricted permutation methods," *Canadian Journal of Statistics*, 37, 625–644. 4

Wu, B., Abbott, T., Fishman, D., McMurray, W., Mor, G., Stone, K., Ward, D., Williams, K., and Zhao, H. (2003), "Comparison of statistical methods for classification of ovarian cancer using mass spectrometry data," *Bioinformatics*, 19, 1636–1643. 2

Wu, Z. and Li, C. (2006), "Feature Selection with Transductive Support Vector Machines," *STUDIES IN FUZZINESS AND SOFT COMPUTING*, 207, 325–341. 12

Xiong, M., Li, W., Zhao, J., Jin, L., and Boerwinkle, E. (2001), "Feature (gene) selection in gene expression-based tumor classification," *Mol Genet Metab*, 73, 239–247. 3

Zhang, P. (1992), "Inference After Variable Selection in Linear Regression Models," *Biometrika*, 79, 741–746. 4

Zhou, X., Wang, X., and Dougherty, E. (2006), "Multi-class cancer classification using multinomial probit regression with Bayesian gene selection," *IEEE Proceedings - Systems Biology*, 153, 70. 12

Acknowledgement

This work was supported by Natural Sciences and Engineering Research Council of Canada, and a start-up research grant from the University of Saskatchewan.