6-2-2021

Fully Bayesian Analysis of Relevance Vector Machine Classification With Probit Link Function for Imbalanced Data Problem

Wenyang Wang

Dongchu Sun

Peng Shao

Haibo Kuang

Cong Sui

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 7, 2020, accepted January 8, 2021, date of publication January 25, 2021, date of current version June 2, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3052935

Fully Bayesian Analysis of Relevance Vector Machine Classification With Probit Link Function for Imbalanced Data Problem

WENYANG WANG[1], DONGCHU SUN[2], PENG SHAO[3], HAIBO KUANG[1], AND CONG SUI[1,4]

[1]School of Maritime Economics and Management, Dalian Maritime University, Dalian 116026, China
[2]Department of Statistics, University of Nebraska–Lincoln, Lincoln, Nebraska 68588, USA
[3]Department of Statistics, University of Missouri, Columbia, MO 65203, USA
[4]Collaborative Innovation Center for Transport Studies, Dalian Maritime University, Dalian 116026, China

Corresponding author: Cong Sui (suicong@dlmu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 71971034, in part by the National Natural Science Foundation of China under Grant 71571034, and in part by the National Natural Science Foundation of China under Grant 71731003.

ABSTRACT The original RVM classification model uses the logistic link function to build the likelihood function making the model hard to be conducted since the posterior of the weight parameter has no closed-form solution. This article proposes the probit link function approach instead of the logistic one for the likelihood function in the RVM classification model, namely PRVM (RVM with the probit link function). We show that the posterior of the weight parameter in PRVM follows the Multivariate Normal distribution and achieves a closed-form solution. A latent variable is needed in our algorithms to simplify the Bayesian computation greatly, and its conditional posterior follows a truncated Normal distribution. Compared with the original RVM classification model, our proposed one is a Fully Bayesian approach, and it has a more efficient computation process. For the prior structure, we first consider the Normal-Gamma independent prior to propose a Generic Bayesian PRVM algorithm. Furthermore, the Fully Bayesian PRVM algorithm with a hierarchical hyperprior structure is proposed, which improves the classification performance, especially in the imbalanced data problem.

INDEX TERMS Bayesian analysis, imbalanced data problem, probit link function, RVM classification.

I. INTRODUCTION

In statistics, Relevance Vector Machine (RVM), initially proposed by [24], is an algorithm that uses the Bayesian model to obtain the parsimonious solutions for regression and probabilistic classification. RVM has obtained successful applications in text image recognition (e.g., [20], [23]), image classification (e.g., [6], [28]), time series analysis (e.g., [14]), mechanical fault diagnosis (e.g., [7], [12]), and electric demand forecasting (e.g., [21], [32]). As a generalized linear model, RVM has an identical functional form to the Support Vector Machine (SVM) but obtains a comparable performance with fewer kernel functions. Since the complex formation of the likelihood function, the original RVM has to use an Expectation Maximization (EM)-like learning method, and it is therefore not a fully Bayesian model. [29] proposed the Bayesian RVM classification model to address this issue, but it is not computationally efficient. This article proposed a concise RVM with the probit link function (PRVM) model to complete the RVM classification framework. PRVM classification model employs the probit link function to build the likelihood function and achieves the closed-form solution for the weight parameter's conditional posterior. Compared with the original RVM and the Bayesian RVM models, the PRVM model is efficient and straightforward.

The imbalanced data problem is the most challenging one in the classification field. Imbalanced datasets are common in real practice where the small number of samples is our research interest in a binary classification problem. In the medical field, cancer patients only account for a minority of the total samples. But if the minority samples are ignored or misclassified, the losses and negative impact are unacceptable. Most traditional classifiers are developed by maximizing the overall classification accuracy rate, and they cannot

The associate editor coordinating the review of this manuscript and approving it for publication was Miaohui Wang.

bestow the minority class in the imbalanced data a convincing classification result. A worse situation is that the minority class is ignored by the classifiers, and all the samples are classified as the majority class. A hierarchical prior structure proposed by [9] is considered in this article to modify the PRVM classification model for the imbalanced data problem. This prior reduces the dimensions of parameter space and builds the inner connection between hyperparameters. The present paper's numeric studies show that this hierarchical prior structure improves the classification results for the imbalanced data problem.

A. SUPPORT VECTOR MACHINE WITH KERNEL FUNCTIONS

In binary classification, we are given a set of input data $\boldsymbol{S}^{train} = \{(\boldsymbol{x}_1^{train}, y_1^{train}), (\boldsymbol{x}_2^{train}, y_2^{train}), \ldots, (\boldsymbol{x}_n^{train}, y_n^{train})\}$ along with the corresponding class label, $y_i^{train} \in \{-1, 1\}$. From the training data, we wish to learn a model of the dependency of the class label on the inputs to make accurate predictions for the unseen values of $\boldsymbol{x}$. A very successful approach to the classification is the Support Vector Machine (SVM). For the linearly separable data, SVM constructs an optimal separating hyperplane as the classification borderline obtaining the maximum distance between two classes for a binary dataset. When we do not have a linearly separable training dataset, the Kernel trick comes in handy. The idea is mapping the non-linear separable dataset into a higher-dimensional space where we find a hyperplane that can separate the samples. If we use a mapping function that projects the data into a higher-dimensional space, SVM's decision rule will depend on the dot products of the mapping function for different samples. The kernel function is employed here to reduces the complexity of finding the mapping function and defines the inner product in the transformed space. [27] and [18] proposed that the output of SVM for an arbitrary data point $\boldsymbol{x}_0$ can be expressed as a weighted summation of the form

$$f(\boldsymbol{x}_0; \boldsymbol{w}) = \sum_{i=1}^{n} w_i k(\boldsymbol{x}_0, \boldsymbol{x}_i^{train}) + w_0, \tag{1}$$

where $k(\cdot, \cdot)$ is the kernel function, w_0 and w_i are weight parameters, $i = 1, 2, \ldots, n$. Note that $\boldsymbol{x}_0$ can be a training data point or a test data point. The Radial Basis Function (RBF) Gaussian Kernel, namely

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp\Big(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\gamma^2}\Big),$$

is used throughout this article. RBF Gaussian Kernel is the most popular kernel function in Statistics and Machine Learning fields. The SVM output for the training data $\boldsymbol{S}^{train}$ can be expressed as the matrix form

$$f(\boldsymbol{x}^{train}; \boldsymbol{w}) = \boldsymbol{K}^{train}\boldsymbol{w}, \tag{2}$$

where $\boldsymbol{w} = (w_0, w_1, \ldots, w_n)^T$ is the weight parameter and

$$\boldsymbol{K}^{train} = (K_1^{train}, K_2^{train}, \ldots, K_n^{train})^T. \tag{3}$$

Note that given the training data $\boldsymbol{S}^{train}$, the kernel matrix $\boldsymbol{K}^{train}$ is fixed. The classification goal is to obtain $sign(f(\boldsymbol{x}^{train}; \hat{\boldsymbol{w}})) = \boldsymbol{y}^{train}$, where $\hat{\boldsymbol{w}}$ is the proper estimate of $\boldsymbol{w}$, $sign(z) = 1$ if $z \geq 0$, and $sign(z) = -1$ if $z < 0$. SVM's target function attempts to minimize the number of misclassified samples while simultaneously maximizing the two classes' margin distance. Plenty of convincing classification results in SVM have been reported, but it still has several significant disadvantages (see [24]):

(1) The classification output in SVM is not probabilistic;

(2) The number of the kernel functions and the parameters grows linearly with the size of the training data;

(3) The trade-off parameter in SVM, C, is necessary to be estimated by cross-validation procedure, which requires the large size of data and high computational quantities;

(4) The kernel functions in SVM must satisfy Mercer's condition.

[24], [25], and [26] introduced and developed the RVM model as a probabilistic sparse Kernel version of SVM to solve its above shortcomings. The next section reviews the original RVM classification model.

B. IMBALANCED DATA PROBLEM

We define $\boldsymbol{S}_+^{train} = \{(\boldsymbol{x}_i^{train}, y_i^{train}) \in \boldsymbol{S}^{train} : y_i = 1, i = 1, \ldots, n\}$ is the positive or minority training class and $\boldsymbol{S}_-^{train} = \{(\boldsymbol{x}_i^{train}, y_i^{train}) \in \boldsymbol{S}^{train} : y_i^{train} = -1, i = 1, \ldots, n\}$ is the negative or majority training class. The class type, {minority, majority} and {positive, negative} are used to describe $\{\boldsymbol{S}_+^{train}, \boldsymbol{S}_-^{train}\}$.

Definition 1: Let $|\boldsymbol{A}|$ denote the number of the elements in a set $\boldsymbol{A}$. Define the number of samples in the positive class and the negative class as $N_p^{train} = |\boldsymbol{S}_+^{train}|$ and $N_n^{train} = |\boldsymbol{S}_-^{train}|$, respectively. The imbalanced degree of data is defined as $b = N_n^{train}/N_p^{train}$.

The condition of $N_n^{train} > N_p^{train}$ is called the imbalanced data problem. Note that $b > 1$ in imbalanced data. b represents the imbalanced index of the sample dataset. The larger value b is, the more severely imbalanced situation of the sample dataset has.

The present article is organized as follows: the detailed introduction of the original RVM classification algorithm is stated in Section 2. Section 3 proposes two PRVM classification algorithms. The numeric studies are posted in Section 4, including the simulated and real data studies. The comparisons between the Bayesian RVM and PRVM are illustrated in Section 5. Section 6 concludes this article with some future research discussions.

II. RVM CLASSIFICATION

Relevance Vector Machine (RVM) is a Bayesian treatment for the output of the Support Vector Machine (SVM). The present article only focuses on the RVM classification, which applies the Bernoulli distribution and the logistic link function to SVM's output in (1) and constructs the probability density function $p(\boldsymbol{y}|\boldsymbol{x})$ for the classification problems.

The logistic link function is the cumulative distribution function of the logistic distribution, which is a continuous probability distribution. The logistic distribution is defined as

$$g_{logis}(t;\mu,s)=\frac{e^{-(t-\mu)/s}}{s\left(1+e^{-(t-\mu)/s}\right)^2}. \tag{4}$$

The logistic distribution receives its name from its cumulative distribution function, which is an instance of the family of logistic functions. The cumulative distribution function of the logistic distribution is also a scaled version of the hyperbolic tangent, which is the CDF of the standard logistic distribution:

$$G_{logis}(t;\mu=0,s=1)=\frac{1}{1+e^{-t}}. \tag{5}$$

The logistic sigmoid link function is used to map $f(\boldsymbol{x};\boldsymbol{w})$ into [0, 1]. The likelihood of the training dataset is

$$\begin{aligned} p(\boldsymbol{y}^{train}|\boldsymbol{w}) &= \prod_{i=1}^{n} G_{logis}\big(f(\boldsymbol{x}_i^{train};\boldsymbol{w})\big)^{\frac{1+y_i^{train}}{2}} \\ &\quad\cdot\left[1-G_{logis}\big(f(\boldsymbol{x}_i^{train};\boldsymbol{w})\big)\right]^{\frac{1-y_i^{train}}{2}} \\ &= \prod_{i=1}^{n}\left(\frac{1}{1+exp(-K_i^{train}\boldsymbol{w})}\right)^{\frac{1+y_i^{train}}{2}} \\ &\quad\cdot\left(\frac{exp(-K_i^{train}\boldsymbol{w})}{1+exp(-K_i^{train}\boldsymbol{w})}\right)^{\frac{1-y_i^{train}}{2}}, \end{aligned} \tag{6}$$

where K_i^{train} and $\boldsymbol{w}$ are defined in (2) and (3). The original RVM classification model proposed by [24], [25] introduced a zero-mean Gaussian prior distribution over $\boldsymbol{w}$, namely

$$p(\boldsymbol{w}|\boldsymbol{\alpha})=\prod_{s=0}^{n}\mathcal{N}(w_s|0,\alpha_s^{-1})=\mathcal{N}(\boldsymbol{w}|0,\boldsymbol{A}^{-1}), \tag{7}$$

where $\boldsymbol{\alpha}=(\alpha_0,\alpha_1,\ldots,\alpha_n)^T$, $\boldsymbol{A}=diag(\alpha_0,\alpha_1,\ldots,\alpha_n)$, α_s is the hyperparameter associated with weight w_s, and $s=0,1,2,\ldots,n$. This prior helps to obtain the sparsity constraint. Compared with SVM, RVM classification has fewer relevant vectors because of the sparsity prior. The Bayesian model provides a posterior distribution for $\boldsymbol{w}$ as

$$\begin{aligned} p(\boldsymbol{w}|\boldsymbol{y}^{train},\boldsymbol{\alpha}) &= \frac{p(\boldsymbol{y}^{train}|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\alpha})}{\int p(\boldsymbol{y}^{train}|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\alpha})d\boldsymbol{w}} \\ &= \frac{s(\boldsymbol{w})}{p(\boldsymbol{y}^{train}|\boldsymbol{\alpha})}, \end{aligned} \tag{8}$$

where $s(\boldsymbol{w})=p(\boldsymbol{y}^{train}|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\alpha})$, which implies that

$$p(\boldsymbol{w}|\boldsymbol{y}^{train},\boldsymbol{\alpha})\propto s(\boldsymbol{w}). \tag{9}$$

The original RVM classification model obtains $\hat{\boldsymbol{w}}$, which is the estimation of $\boldsymbol{w}$, by maximizing $s(\boldsymbol{w})$. The classification function for the training data $\boldsymbol{S}_{train}$ is

$$y_*^{train}=sign\left(\frac{1}{1+exp(-\boldsymbol{K}^{train}\hat{\boldsymbol{w}})}-\frac{1}{2}\right), \tag{10}$$

where $\boldsymbol{K}^{train}$ is defined in (3). A test data can be defined as $\boldsymbol{S}^{test}=\{(\boldsymbol{x}_1^{test},y_1^{test}),(\boldsymbol{x}_2^{test},y_2^{test}),\ldots,(\boldsymbol{x}_m^{test},y_m^{test})\}$, where $\boldsymbol{x}_j^{test}\in\boldsymbol{X}\subseteq\boldsymbol{R}^l$, $\boldsymbol{X}$ is in the same vector space as the training data. The response $y_j^{test}\in\{-1,1\}$ indicates two classes, $j=1,\ldots,m$. In the imbalanced data problem, we define $\boldsymbol{S}_+^{test}=\{(\boldsymbol{x}_j^{test},y_j^{test})\in\boldsymbol{S}^{test}:y_j^{test}=1,j=1,\ldots,m\}$ and $\boldsymbol{S}_-^{test}=\{(\boldsymbol{x}_j^{test},y_j^{test})\in\boldsymbol{S}^{test}:y_j^{test}=-1,j=1,\ldots,m\}$. The classification function for a test data $\boldsymbol{S}_{test}$ is

$$y_*^{test}=sign\left(\frac{1}{1+exp(-\boldsymbol{K}^{test}\hat{\boldsymbol{w}})}-\frac{1}{2}\right), \tag{11}$$

where $\boldsymbol{K}^{test}=(K_1^{test},K_2^{test},\ldots,K_m^{test})^T$, $K_j^{test}=(1,k(\boldsymbol{x}_j^{test},\boldsymbol{x}_1^{train}),k(\boldsymbol{x}_j^{test},\boldsymbol{x}_2^{train}),\cdots,k(\boldsymbol{x}_j^{test},\boldsymbol{x}_n^{train}))$, $j=1,2,\ldots,m$.

The original RVM classification algorithm is stated as follows:

Algorithm 1 The Original RVM Classification Algorithm

Input. The training data: $(\boldsymbol{x}_1,y_1),(\boldsymbol{x}_2,y_2),\ldots,(\boldsymbol{x}_n,y_n)$, $\boldsymbol{x}_i\in\boldsymbol{X}\subseteq\boldsymbol{R}^l$ and $y_i\in\{-1,1\}$, $i=1,\ldots,n$.

0. Let $t=1$ and initialize $\boldsymbol{w}$ and $\boldsymbol{\alpha}$ to obtain the started values $\boldsymbol{w}^1$ and $\boldsymbol{\alpha}^1$, calculate

$$\begin{aligned} h &= \nabla_{\boldsymbol{w}}\log g(\boldsymbol{w}) \\ &= \boldsymbol{\Phi}^\top\big(\boldsymbol{y}-[\sigma(\phi_1\boldsymbol{w}),\ldots,\sigma(\phi_n\boldsymbol{w})]^\top\big)-\boldsymbol{A}\boldsymbol{w}, \\ \boldsymbol{H} &= -\nabla\nabla_{\boldsymbol{w}}\log g(\boldsymbol{w})=\boldsymbol{\Phi}^\top\boldsymbol{B}\boldsymbol{\Phi}+\boldsymbol{A}, \end{aligned} \tag{12}$$

where $\boldsymbol{B}$ is a $(n+1)\times(n+1)$ diagonal matrix with diagonal elements $b_{ii}=\sigma(\phi_i\boldsymbol{w})\big[1-\sigma(\phi_i\boldsymbol{w})\big]$;

1. Fix $\boldsymbol{\alpha}$ and update $\boldsymbol{w}$ with

$$\boldsymbol{w}^{t+1}=\boldsymbol{w}^{t}+(\boldsymbol{H})^{-1}h|_{\boldsymbol{w}=\boldsymbol{w}^t}; \tag{13}$$

2. Fix $\boldsymbol{w}$ and update $\boldsymbol{\alpha}$ with

$$\alpha_s^{t+1}=\frac{\gamma_s^t}{\boldsymbol{w}_s^2}, \tag{14}$$

where $\gamma_s^t=1-\alpha_s^t\boldsymbol{H}_{ss}$, $s=0,1,2,\ldots,n$;

3. Repeat steps 1 and 2 until suitable convergence and obtain $\boldsymbol{w}_0$, the mode of $\boldsymbol{w}$;

Output. The final estimation of $\boldsymbol{w}$ is $\boldsymbol{w}_{\mathrm{MP}}=\boldsymbol{H}^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{B}\boldsymbol{y}|_{\boldsymbol{w}=\boldsymbol{w}_0}$.

Note that $\boldsymbol{w}_{\mathrm{MP}}$, the maximum posterior of $\boldsymbol{w}$, is obtained by Laplace's Method in [24], which approximates a Normal distribution with the mean value $\boldsymbol{w}_0$ to the posterior of $\boldsymbol{w}$. [4], [15], [30] concluded that RVM is better than SVM in the fields of classification and regression. They also showed that the conduction speed of RVM is faster than SVM. Nevertheless, [29] indicated several shortcomings of the original RVM, mainly caused by the complex likelihood function. [29] proposed a Bayesian RVM model, which ameliorated the original RVM classification model by directly doing the Gibbs sampling process from the posterior based on the log-concave property. Although the Bayesian RVM model works, it is non-efficient and high-computational.

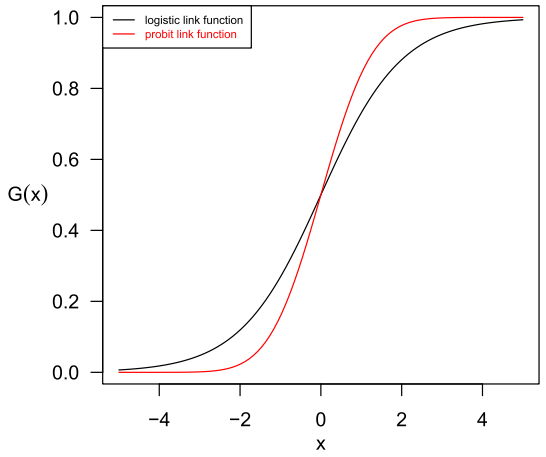


FIGURE 1. Logistic and Probit Link Functions.

This article proposes the probit link function to form a new likelihood function in the RVM classification model instead of the logistic one in the original algorithm. Benefiting from a latent variable, this new likelihood function can lead to a more concise posterior, which follows a Multivariate Normal distribution.

III. RVM CLASSIFICATION WITH PROBIT LINK FUNCTION

A. THE PROBIT LINK FUNCTIONS

A probit model is a type of regression where the dependent variable has two values, and the independent variable is on $(-\infty, +\infty)$. The probit model aims to estimate the probability that the observations with particular characteristics will fall into a specific category, so it is famous for the binary classification problem. The probit link function $G_{probit}(x)$ is used to map $f(\boldsymbol{x}; \boldsymbol{w})$ into [0, 1]. $G_{probit}(x)$ is defined as

$$G_{probit}(t) = \Phi(t) = \int_{-\infty}^{t} \frac{1}{\sqrt{2\pi}} exp(-\frac{1}{2}z^2)dz. \quad (15)$$

Figure 1 shows the logistic and probit link functions. The logistic one has slightly flatter tails. The probit curve approaches the axes more quickly than the logistic curve. In the binary classification problems, they are the same in the application.

B. GENERIC BAYESIAN PRVM CLASSIFICATION MODEL

The Bernoulli probability of every data point is

$$p_i = R_i^{\frac{1+y_i^{train}}{2}} (1 - R_i)^{\frac{1-y_i^{train}}{2}}, \quad (16)$$

where $R_i = G_{probit}(K_i^{train}\boldsymbol{w})$, K_i^{train} and $\boldsymbol{w}$ are defined in (2) and (3). The likelihood function of the training dataset is

$$\begin{aligned} P(\boldsymbol{y}^{train}|\boldsymbol{w}) &= \prod_{i=1}^{n} p_i \\ &= \prod_{i=1}^{n} R_i^{\frac{1+y_i^{train}}{2}} (1 - R_i)^{\frac{1-y_i^{train}}{2}} \\ &= \prod_{i=1}^{n} G_{probit}(K_i^{train}\boldsymbol{w})^{\frac{1+y_i^{train}}{2}} \cdot \\ &\quad (1 - G_{probit}(K_i^{train}\boldsymbol{w}))^{\frac{1-y_i^{train}}{2}}. \end{aligned} \quad (17)$$

Following [2], we bring in a latent variable $\boldsymbol{\mu}$ for the probit link function:

$$\begin{aligned} \boldsymbol{\mu} &= (\mu_1, \mu_2, \ldots, \mu_n)^T, \\ \mu_i &\overset{indep.}{\sim} \mathcal{N}(K_i\boldsymbol{w}, 1). \end{aligned} \quad (18)$$

We can show

$$\begin{aligned} R_i &= G_{probit}(K_i\boldsymbol{w}) \\ &= \int_{-\infty}^{K_i\boldsymbol{w}} \frac{1}{\sqrt{2\pi}} exp(-\frac{1}{2}z^2)dz \\ &= \int_{0}^{+\infty} \frac{1}{\sqrt{2\pi}} exp(-\frac{1}{2}(\mu_i - K_i\boldsymbol{w})^2)d\mu_i \\ &= P(\mu_i > 0). \end{aligned} \quad (19)$$

Note that $1-R_i = P(\mu_i \leq 0)$. Rewrite the likelihood function, including the latent variable

$$\begin{aligned} P(\boldsymbol{y}^{train}|\boldsymbol{w}, \mu) = \prod_{i=1}^{n} &\left(\mathbf{1}_{(\mu_i>0)}\right)^{\frac{1+y_i^{train}}{2}} \left(\mathbf{1}_{(\mu_i\leq 0)}\right)^{\frac{1-y_i^{train}}{2}} \\ &\cdot \phi(\mu_i - K_i\boldsymbol{w}). \end{aligned} \quad (20)$$

Follow the original RVM classification model to introduce a zero-mean Gaussian prior distribution over $\boldsymbol{w}$, namely

$$p(\boldsymbol{w}|\boldsymbol{\alpha}) = \prod_{s=0}^{n} \mathcal{N}(w_s|0, \alpha_s^{-1}) = \mathcal{N}(\boldsymbol{w}|0, \boldsymbol{A}^{-1}),$$

where $\boldsymbol{\alpha} = (\alpha_0, \alpha_1, \ldots, \alpha_n)^T$, $\boldsymbol{A} = diag(\alpha_0, \alpha_1, \ldots, \alpha_n)$, α_s is the hyperparameter associated with weight w_s, and $s = 0, 1, 2, \ldots, n$. A Gamma hyperprior is called for each α_s. The Gamma hyperprior is

$$(\alpha_s|a, b) \sim Gamma(\alpha_s|a, b).$$

The full posterior is

$$\begin{aligned} &p(\boldsymbol{w}, \boldsymbol{y}^{train}, \boldsymbol{\alpha}, \boldsymbol{\mu}) \\ &= (2\pi)^{-\frac{n+1}{2}} |\boldsymbol{A}|^{\frac{1}{2}} exp(-\frac{1}{2}\boldsymbol{w}^T\boldsymbol{A}\boldsymbol{w}) \\ &\quad \cdot \prod_{i=1}^{n} \left(\mathbf{1}_{(\mu_i>0)}\right)^{\frac{1+y_i^{train}}{2}} \left(\mathbf{1}_{(\mu_i\leq 0)}\right)^{\frac{1-y_i^{train}}{2}} \phi(\mu_i - K_i\boldsymbol{w}). \end{aligned}$$

The conditional posterior of $\boldsymbol{w}$ is

$$p(\boldsymbol{w}|\boldsymbol{y}^{train}, \boldsymbol{\alpha}, \boldsymbol{\mu}) \propto exp(-\frac{1}{2}\boldsymbol{w}^T\boldsymbol{A}\boldsymbol{w}) \prod_{i=1}^{n} \phi(\mu_i - K_i\boldsymbol{w}). \quad (21)$$

Theorem 1: The conditional posterior of $\boldsymbol{w}$ *follows a Multivariate Normal distribution*

$$p(\boldsymbol{w}|\boldsymbol{y}^{train}, \boldsymbol{\alpha}, \boldsymbol{\mu}) \propto \boldsymbol{N}(\hat{\boldsymbol{w}}, \boldsymbol{V}^{-1}), \quad (22)$$

where $\boldsymbol{V} = \boldsymbol{A} + \boldsymbol{K}^T\boldsymbol{K}$, $\hat{\boldsymbol{w}} = \boldsymbol{V}^{-1}\boldsymbol{K}^T\boldsymbol{\mu}$.

Proof: See Appendix A1. □

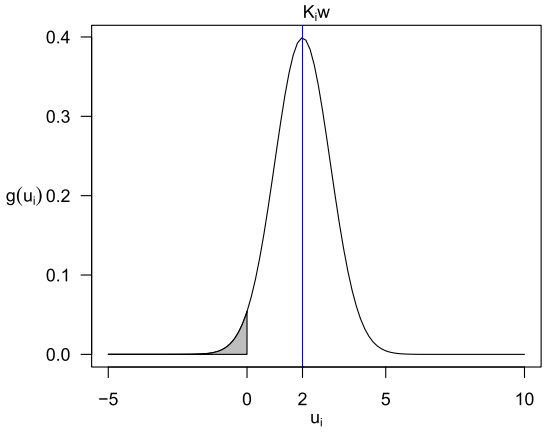


FIGURE 2. Sampling From a Truncated Normal Distribution.

The conditional posterior of α_s is

$$\begin{aligned} p(\alpha_s|w_s,a,b) &\propto \alpha_s^{\frac{1}{2}}\, exp\Big(-\frac{1}{2}\alpha_s w_s^2\Big)\cdot \alpha_s^{a-1}\, exp(-b\,\alpha_s) \\ &= \alpha_s^{a+\frac{1}{2}-1}\, exp\Big[-(b+\frac{1}{2}w_s^2)\alpha_s\Big] \\ &\propto Gamma(a+\frac{1}{2}, b+\frac{1}{2}w_s^2). \end{aligned} \tag{23}$$

The conditional posterior of μ_i is

$$\begin{aligned} &p(\mu_i|\boldsymbol{w},\boldsymbol{y}^{train},\boldsymbol{\alpha}) \\ &\propto \big(\mathbf{1}_{(\mu_i>0)}\big)^{\frac{1+y_i^{train}}{2}}\big(\mathbf{1}_{(\mu_i\le 0)}\big)^{\frac{1-y_i^{train}}{2}}\phi(\mu_i-K_i\boldsymbol{w}) \\ &= \begin{cases} g_{R_i}(u_i)\,\mathbf{1}_{(u_i>0)} & \text{if } y_i=1 \\ g_{R_i}(u_i)\,\mathbf{1}_{(u_i\le 0)} & \text{if } y_i=-1, \end{cases} \end{aligned} \tag{24}$$

where $g_{R_i}=\phi(\mu_i-K_i\boldsymbol{w})$, $i=1,2,\ldots,n$. This conditional posterior is a truncated Normal distribution and the sampling process may be inefficient. When $K_i\boldsymbol{w}$ is far away from 0, one sampling process of (24) for $y_i=1$ or $y_i=-1$ would have a low acceptable rate. Figure 2 shows a situation where we sample some negative values from a Normal distribution with a mean value of $K_i\boldsymbol{w}=2$. Only the shaded area can satisfy our requirement, and the acceptable sampling rate is low.

[16] proposed following Lemma 1 with a 100% acceptable rate sampling method for this conditional posterior.

Lemma 1: [16] Let u be a uniform random variable on $(0,1)$, the variable Z follows a normal distribution $Z\sim N(b,1)$.

(1) $D=b+\Phi^{-1}(u\Phi(-b))$ and $Z|Z\le 0$ have the same distribution.

(2) $D=b+\Phi^{-1}(1-u\Phi(b))$ and $Z|Z>0$ have the same distribution.

Proof: See Appendix A2. □

For $i=1,2,\ldots,n$, we can do the sampling of μ_i as follows based on Lemma 1:

(1) Sample

$$u\sim uniform(0,1); \tag{25}$$

(2) If $y_i=1$, calculate

$$\mu_i=K_i\boldsymbol{w}+\Phi^{-1}(1-u\Phi(K_i\boldsymbol{w})); \tag{26}$$

(3) If $y_i=-1$, calculate

$$\mu_i=K_i\boldsymbol{w}+\Phi^{-1}(u\Phi(-K_i\boldsymbol{w})). \tag{27}$$

$\Phi(\cdot)$ is the cumulative distribution function (CDF) of the standard Normal distribution. The following pseudo-code is implemented to perform this Generic PRVM classification model.

Algorithm 2 The Generic PRVM Classification Algorithm

Input. The training data: $(\boldsymbol{x}_1,y_1),(\boldsymbol{x}_2,y_2),\ldots,(\boldsymbol{x}_n,y_n)$, $\boldsymbol{x}_i\in\boldsymbol{X}\subseteq\boldsymbol{R}^l$ and $y_i\in\{-1,1\}$, $i=1,\ldots,n$.

0. Let $t=1$ and initialize $\boldsymbol{w}$, $\boldsymbol{\alpha}$, and $\boldsymbol{\mu}$ to obtain the started values $\boldsymbol{w}^t$, $\boldsymbol{\alpha}^t$ and $\boldsymbol{\mu}^t$. Choose (a,b), the number of burn-in B, and the number of iterations T;

1. Fix $\boldsymbol{\alpha}^t$ and $\boldsymbol{\mu}^t$, draw a new $\boldsymbol{w}^{t+1}$ according to (22);

2. Fix $\boldsymbol{w}^{t+1}$ and $\boldsymbol{\mu}^t$, draw a new $\boldsymbol{\alpha}^{t+1}$ according to (23);

3. Fix $\boldsymbol{\alpha}^{t+1}$ and $\boldsymbol{w}^{t+1}$, draw a new $\boldsymbol{\mu}^{t+1}$ according to (25, 26, 27);

3. Repeat steps 1, 2 and 3 until suitable convergence is obtained by T iterations;

Output. The final estimation of $\boldsymbol{w}$ is $\hat{\boldsymbol{w}}=(T-B)^{-1}\sum_{t=B+1}^{T}\boldsymbol{w}^t$.

Algorithm 2 is more succinct and efficient compared with the original RVM (see [24]) and the Bayesian RVM (see [29]). The conditional posteriors all have closed-form solutions, and the sampling process is simple. For the imbalanced data problem, we follow the Hierarchical Bayesian RVM classification model in [29] to apply the hierarchical prior structure in [9] to PRVM.

C. FULLY HIERARCHICAL BAYESIAN PRVM CLASSIFICATION MODEL

This subsection follows the hierarchical prior structure in [9] but is applied to the PRVM classification model instead of the regression problem. As one of the main contributions of [9], the hierarchical prior adds another layer random-coefficient structure for prior of $\boldsymbol{\alpha}$, which reduces the parameter dimensions. [9] shows that the estimation of parameters in the Generic RVM regression is not consistent, and the dimensions reduction by the hierarchical prior can solve this issue. This Fully Bayesian method could relate α_s's with the coefficient parameter and enhance the inner connection of parameters. Compared with [9], our model makes some improvements. Only n dimensions of the parameters were considered in [9]. The error term of the parameters, w_0 and α_0, were ignored. This present work considers all $n+1$ dimensions in the parameters. In the numeric study part, [9] specified all the hyperparameters and only sampled $\boldsymbol{w}$ and $\boldsymbol{\alpha}$ in the Gibbs sampling process. The numeric studies in this project run

the full Gibbs sampling iterations, including all the parameters. Compared with the Hierarchical Bayesian RVM model in [29], our Fully Hierarchical PRVM classification model is more concise in the sampling process and more efficient in the numeric studies.

Recall the prior for w_s is

$$p(w_s|\alpha_s) = \mathcal{N}(w_s|0, \alpha_s^{-1}). \tag{28}$$

Reparametrize $\boldsymbol{\alpha}$ as $\boldsymbol{\eta} = (\eta_0, \eta_1, \ldots, \eta_n)$, where $\eta_s = log(\alpha_s)$, and $s = 0, 1, 2, \ldots, n$. [9] defined the hyperprior for $\boldsymbol{\eta}$ is

$$\boldsymbol{\eta} \sim \mathcal{N}_{n+1}(\mu \mathbf{1}_{n+1}, \tau^2 \boldsymbol{\Sigma}_{n+1}), \tag{29}$$

where $\boldsymbol{\Sigma}_{n+1} = (1-\rho)\mathbf{I}_{n+1} + \rho \mathbf{1}_{n+1}\mathbf{1}_{n+1}^\top$, $\mathbf{I}_{n+1}$ is an identity matrix, and $\mathbf{1}_{n+1}$ is a vector with all values of 1. Note that ρ should remain in the interval of $(0, 1)$. The interpretation of ρ is to maintain the trade-off between absolute freedom of α_s's when ρ is close to 0 and the total tightness of α_s's when ρ is close to 1. τ^2 should be relatively large because sparsity is still an important goal in RVM classification. The value of ρ indicates the relative contribution of the joint effects between all the α_s's, the value of τ^2 controls the magnitude of information in $\boldsymbol{\alpha}$. Based on their expected effect, we propose the constant priors for ρ and μ, a conjugate prior for τ^2, namely

$$\begin{aligned} p(\rho) &= Uniform(0, 1), \\ p(\mu) &= Uniform(0, 1), \\ p(\tau^{-2}) &= Gamma(c, d). \end{aligned} \tag{30}$$

Since we only add a new layer to the prior, the Fully conditional posterior for $\boldsymbol{w}$ remains unchanged as (22). For the joint posterior of $\boldsymbol{\alpha}$, we can reach it through its reparametrized version $\boldsymbol{\eta}$, $\boldsymbol{\eta} = log(\boldsymbol{\alpha})$,

$$\begin{aligned} p(\boldsymbol{\eta}|\text{ others}) &\propto p(\boldsymbol{w}|\boldsymbol{\alpha}(\boldsymbol{\eta}))p\big(\boldsymbol{\eta}|\mu, \rho, \tau^2\big) \\ &\propto \left(\prod_{s=1}^{n+1} \frac{1}{\sqrt{2\pi}} e^{\eta_s/2}\right) \exp\left(-\frac{1}{2}\sum_{s=1}^{n+1} e^{\eta_s} w_s^2\right) \\ &\quad \cdot \exp\left\{-\frac{1}{2\tau^2(1-\rho)}\sum_{s=1}^{n+1}(\eta_s - \mu)^2 \right. \\ &\quad \left. + \frac{\rho}{2\tau^2(1-\rho)(1+n\rho)}\left[\sum_{s=1}^{n+1}(\eta_s - \mu)\right]^2\right\}. \end{aligned} \tag{31}$$

It seems hard to draw samples for the posterior of $\boldsymbol{\eta}$, but we can show the posterior's desired log-concave property.

Theorem 2: The conditional posterior of η_s, $p(\eta_s|others)$ is log-concave.

Proof: See Appendix A3. □

Based on the log-concavity, the Adaptive Rejection Sampling (ARS) Method employed by the Bayesian RVM models in [29] can be applied again. See Appendix B1 for more details about the ARS sampling method.

The prior for ρ allows us to write

$$\begin{aligned} p(\rho|\text{ others}) &\propto p(\rho)p(\boldsymbol{\eta}|\mu, \rho, \tau^2) \\ &\propto \frac{1}{(1-\rho)^{\frac{n}{2}}(1+n\rho)^{\frac{1}{2}}} \exp\left\{-\frac{1}{2\tau^2(1-\rho)} \cdot \right. \\ &\quad \sum_{s=1}^{n+1}(\eta_s - \mu)^2 + \frac{\rho}{2\tau^2(1-\rho)(1+n\rho)} \cdot \\ &\quad \left.\left[\sum_{s=1}^{n+1}(\eta_i - \mu)\right]^2\right\}. \end{aligned} \tag{32}$$

The method of Ratio of Uniforms is used to sample from this conditional posterior. See Appendix B2 for more details about the Ratio of Uniforms sampling method. The posterior of μ is

$$\begin{aligned} p(\mu|\text{ others }) &\propto p(\mu)p(\boldsymbol{\alpha}|\mu, \rho, \tau^2) \\ &\propto \exp\left\{-\frac{1}{2\tau^2(1-\rho)}\sum_{s=1}^{n+1}(\eta_s - \mu)^2 \right. \\ &\quad \left. + \frac{\rho}{2\tau^2(1-\rho)(1+n\rho)}\left[\sum_{s=1}^{n+1}(\eta_i - \mu)\right]^2\right\} \\ &\propto \exp\left\{-\frac{n+1}{2\tau^2(1+n\rho)}\left(\mu - \frac{\sum_{s=1}^{n+1}\eta_s}{n+1}\right)^2\right\} \\ &\propto \mathcal{N}\left(\frac{\sum_{s=1}^{n+1}\eta_s}{n+1}, \frac{\tau^2(1+n\rho)}{n+1}\right). \end{aligned} \tag{33}$$

For τ^2, we have

$$\begin{aligned} &p(\tau^{-2}|\text{ others }) \\ &\quad \propto p(\tau^{-2})p(\boldsymbol{\eta}|\mu, \rho, \tau^2) \\ &\quad \propto (\tau^{-2})^{c-1}\exp(-d\tau^{-2})(\tau^{-2})^{\frac{n+1}{2}} \\ &\qquad \cdot \exp\left(-\frac{1}{2\tau^2}(\boldsymbol{\eta} - \mu\mathbf{1}_{n+1})^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{\eta} - \mu\mathbf{1}_{n+1})\right) \\ &\quad \propto \text{Gam}\left(c + \frac{n+1}{2}, d + \frac{1}{2}\left\{\frac{1}{1-\rho}\sum_{s=1}^{n+1}(\eta_s - \mu)^2 \right.\right. \\ &\qquad \left.\left. - \frac{\rho}{(1-\rho)(1+n\rho)}\left[\sum_{s=1}^{n+1}(\eta_s - \mu)\right]^2\right\}\right). \end{aligned} \tag{34}$$

The samples of μ and τ^2 are easy to obtain from their special closed-forms. Based on the above derivations of full conditional posteriors, we have an alternative Algorithm 3.

IV. NUMERIC STUDIES

A. SIMULATION DATA STUDIES

To make the comparisons easier, we use the same simulated datasets as [29] in this subsection. Five two-dimensional simulated Gaussian datasets are chosen and they are distributed as:

$$\boldsymbol{X}_{ij} = \begin{pmatrix} X_{ij1} \\ X_{ij2} \end{pmatrix} \overset{iid}{\sim} Uniform_2(\boldsymbol{a}_i, \boldsymbol{b}_i), \tag{35}$$

Algorithm 3 The Fully Hierarchical Bayesian PRVM Classification Algorithm

Input. The training data: $(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_n, y_n)$, $\boldsymbol{x}_i \in \boldsymbol{X} \subseteq \boldsymbol{R}^l$ and $y_i \in \{-1, 1\}$, $i = 1, \ldots, n$.

0. Let $t = 1$ and initialize $\boldsymbol{w}$, $\boldsymbol{\alpha}$, μ, m, ρ and τ^2 to obtain the started values $\boldsymbol{w}_t$, $\boldsymbol{\alpha}_t$, μ_t, m_t, ρ_t, and τ_t^2. Choose (a, b), (c, d), the number of burn-in B, and the number of iterations T;

1. Fix other parameters and draw a new $\boldsymbol{w}_{t+1}$ according to (22);

2. Fix other parameters and draw a new $\boldsymbol{\alpha}_{t+1}$ according to (31);

3. Fix other parameters and draw a new μ_{t+1} according to (31);

4. Fix other parameters and draw a new m_{t+1} according to (33);

5. Fix other parameters and draw a new ρ_{t+1} according to (32);

6. Fix other parameters and draw a new τ_{t+1}^2 according to (34);

7. Repeat steps $1-5$ until suitable convergence is obtained by T iterations;

Output. The final estimation of $\boldsymbol{w}$ is $\hat{\boldsymbol{w}} = (T-B)^{-1} \sum_{t=B+1}^{T} \boldsymbol{w}_t$.

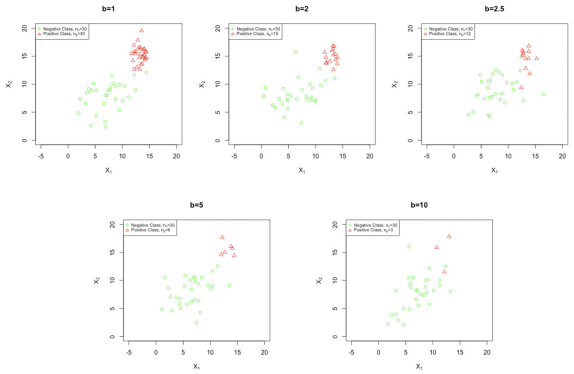


FIGURE 3. Simulated Gaussian Data for Numeric Studies.

where $i = -1, 1$ and $j = 1, \ldots, n_i$.

$$\boldsymbol{a}_{-1} = \begin{pmatrix} 1 \\ 5 \end{pmatrix}, \boldsymbol{b}_{-1} = \begin{pmatrix} 10 \\ 12 \end{pmatrix}, \boldsymbol{a}_1 = \begin{pmatrix} 4 \\ 12 \end{pmatrix}, \boldsymbol{b}_1 = \begin{pmatrix} 6 \\ 13 \end{pmatrix}. \quad (36)$$

All the variables in $\boldsymbol{X}_{-1,j}$ and $\boldsymbol{X}_{1,j}$ have the class labels -1 (Majority) and 1 (Minority), respectively. We set five kinds of sizes, $(n_p, n_n) = (30, 30), (15, 30), (12, 30), (6, 30)$, and $(3, 30)$ to illustrate the performance of different algorithms in different-sized data. $b = 1, 2, 2.5, 5, 10$ for these five cases and a larger b indicates a more severely imbalanced dataset. Following Figure 3 shows the training datasets.

Eight criteria listed in Table 1 are used to evaluate the performance of algorithms in this article. The calculations of r_g^{test} and r_p^{test} use the same-sized test data as the training data, $n^{train} = n^{test} = n$, $n_p^{train} = n_p^{test} = n_p$. Smaller-sized and larger-sized test data are used for the calculations of $(r_g^{stest}, r_p^{stest})$ and $(r_g^{ltest}, r_p^{ltest})$, which means $n^s < n < n^l$, $n_p^s < n_p < n_p^l$. The simulation data studies use all these eight criteria. The real data studies in next subsection only apply r_g^{train}, r_p^{train}, r_g^{test}, and r_p^{test} because it is hard to create more real test data. All the test datasets in this article keep the same imbalance index b as the training data, namely

$$\frac{|\boldsymbol{S}_-^{test}|}{|\boldsymbol{S}_+^{test}|} = \frac{|\boldsymbol{S}_-^{train}|}{|\boldsymbol{S}_+^{train}|} = b.$$

We run the **Algorithm 2** and **3** with $T = 5000$, $B = 500$, $(a, b) = (1, 1/999)$, and $(c, d) = (1, 1/999)$ on the simulated datasets, and run the **Algorithm 1** 30000 iterations. **Algorithm 2** and **3** recieve the significant convergence after 500 itrerations but **Algorithm 1** cannot obtain the parameters's convergence as we show before. The evaluation criteria come from Table 1. For all **Algorithm 1**, **2**, and **3**, we repeat the experiments 100 times for every case in Figure 3. Plus the simulation studies results in [29], Tables 2–6 display the mean values and standard deviation values (shown in the bracket) of 100 repeated results for all the algorithms in the RVM classification framework, including the original RVM classification model ([24]), the Generic Bayesian RVM classification model ([29]), the Fully Hierarchical Bayesian RVM classification model ([29]), the Generic PRVM classification model, and the Fully Hierarchical PRVM classification model. The larger accuracy rate is indicated by boldface.

These simulation studies show that PRVM has a similar performance as the original RVM and the Bayesian RVM models for the moderately imbalanced datasets. For the seriously imbalanced data as $b = 5, 10$. Two algorithms of PRVM outperform the others. Especially for the case of $b = 10$, the PRVM is significantly preferred.

TABLE 1. The Criteria for Classification Evaluation in RVM Studies.

Training Data Global Accuracy Rate	$r_g^{train} = \frac{\lvert y^{train} = y_*^{train} \rvert}{n}$
Training Data Positive Class Accuracy Rate	$r_p^{train} = \frac{\lvert y^{train} = y_*^{train} \,\&\, y^{train} = 1 \rvert}{n_p}$
Same Size Test Data Global Accuracy Rate	$r_g^{test} = \frac{\lvert y^{test} = y_*^{test} \rvert}{n}$
Same Size Test Data Positive Class Accuracy Rate	$r_p^{test} = \frac{\lvert y^{test} = y_*^{test} \,\&\, y^{test} = 1 \rvert}{n_p}$
Smaller Size Test Data Global Accuracy Rate	$r_g^{stest} = \frac{\lvert y^{stest} = y_*^{stest} \rvert}{n^s}$
Smaller Size Test Data Positive Class Accuracy Rate	$r_p^{stest} = \frac{\lvert y^{stest} = y_*^{stest} \,\&\, y^{stest} = 1 \rvert}{n_p^s}$
Larger Size Test Data Global Accuracy Rate	$r_g^{ltest} = \frac{\lvert y^{ltest} = y_*^{ltest} \rvert}{n^l}$
Larger Size Test Data Positive Class Accuracy Rate	$r_p^{ltest} = \frac{\lvert y^{ltest} = y_*^{ltest} \,\&\, y^{ltest} = 1 \rvert}{n_p^l}$

B. REAL DATA STUDIES

We choose four real imbalanced datasets, ''pima'', ''segment0'', ''vowel0'', and ''glass5'' from the KEEL-dataset repository (see [3]). Their imbalanced indexes are 1.87, 6.02, 9.98, and 22.78, respectively. ''pima'' dataset is originally from the Indian National Institute of Diabetes,

TABLE 2. The Results of Simulated Data Study ($b = 1$).

	$r_g^{train\,a}$	$r_g^{test\,b}$	$r_g^{stest\,c}$	$r_g^{ltest\,d}$	$r_p^{train\,a}$	$r_p^{test\,b}$	$r_p^{stest\,c}$	$r_p^{ltest\,d}$
Original RVM	0.9848	0.9703	**0.9848**	0.9683	0.9922	0.9886	0.9814	0.9617
	(0.0319)	(0.1777)	**(0.2066)**	(0.1623)	(0.0441)	(0.3882)	(0.2744)	(0.2371)
Generic Bayesian RVM	0.9823	0.9710	0.9780	0.9732	**0.9993**	0.9980	0.9980	0.9996
	(0.0148)	(0.0254)	(0.0306)	(0.0209)	**(0.0047)**	(0.0080)	(0.0141)	(0.0022)
Hierarchical Bayesian RVM	0.9770	0.9678	0.9705	0.9674	**0.9993**	**1.0000**	**0.9990**	**0.9998**
	(0.0170)	(0.0328)	(0.0390)	(0.0300)	**(0.0067)**	**(0.0000)**	**(0.0100)**	**(0.0016)**
Generic PRVM	**0.9990**	**0.9770**	0.9729	**0.9792**	0.9987	0.9907	0.9900	0.9987
	(0.0071)	**(0.0947)**	(0.0833)	**(0.0914)**	(0.0094)	(0.0802)	(0.0735)	(0.0593)
Hierarchical PRVM	0.9876	0.9717	0.9766	0.9778	**0.9993**	0.9967	0.9930	0.9911
	(0.1701)	(0.0760)	(0.0568)	(0.0750)	**(0.0067)**	(0.1350)	(0.1060)	(0.1254)

^{a} $(n_n^{train} = 30, n_p^{train} = 30)$, ^{b} $(n_n^{test} = 30, n_p^{test} = 30)$, ^{c} $(n_n^{stest} = 10, n_p^{stest} = 10)$, ^{d} $(n_n^{ltest} = 90, n_p^{ltest} = 90)$

TABLE 3. The Results of Simulated Data Study ($b = 2$).

	$r_g^{train\,a}$	$r_g^{test\,b}$	$r_g^{stest\,c}$	$r_g^{ltest\,d}$	$r_p^{train\,a}$	$r_p^{test\,b}$	$r_p^{stest\,c}$	$r_p^{ltest\,d}$
Original RVM	0.9709	0.9660	0.9699	0.9777	0.9522	0.9588	0.9683	0.9631
	(0.0411)	(0.0487)	(0.1821)	(0.1552)	(0.1041)	(0.1066)	(0.1188)	(0.2633)
Generic Bayesian RVM	0.9796	0.9791	0.9773	0.9757	0.9680	0.9693	**0.9760**	0.9698
	(0.0257)	(0.0235)	(0.0418)	(0.0147)	(0.0690)	(0.0542)	**(0.0870)**	(0.0378)
Hierarchical Bayesian RVM	0.9760	0.9822	0.9740	0.9808	0.9707	**0.9767**	0.9700	**0.9798**
	(0.0236)	(0.0214)	(0.0443)	(0.0138)	(0.0616)	**(0.0477)**	(0.0823)	**(0.0322)**
Generic PRVM	0.9802	0.9724	0.9707	0.9719	**0.9711**	0.9640	0.9640	0.9636
	(0.1507)	(0.0322)	(0.0408)	(0.0192)	**(0.2340)**	(0.0854)	(0.0875)	(0.0527)
Hierarchical PRVM	**0.9816**	**0.9874**	**0.9778**	**0.9843**	0.9698	0.9667	0.9667	0.9781
	(0.1251)	**(0.0348)**	**(0.0328)**	**(0.0412)**	(0.1456)	(0.0603)	(0.0778)	(0.0783)

^{a} $(n_n^{train} = 30, n_p^{train} = 15)$, ^{b} $(n_n^{test} = 30, n_p^{test} = 15)$, ^{c} $(n_n^{stest} = 10, n_p^{stest} = 5)$, ^{d} $(n_n^{ltest} = 90, n_p^{ltest} = 45)$

TABLE 4. The Results of Simulated Data Study ($b = 2.5$).

	$r_g^{train\,a}$	$r_g^{test\,b}$	$r_g^{stest\,c}$	$r_g^{ltest\,d}$	$r_p^{train\,a}$	$r_p^{test\,b}$	$r_p^{stest\,c}$	$r_p^{ltest\,d}$
Original RVM	0.9609	0.9687	0.9631	0.9688	0.9377	0.9283	0.9322	0.9400
	(0.0355)	(0.1330)	(0.1163)	(0.1383)	(0.1952)	(0.1542)	(0.1622)	(0.2266)
Generic Bayesian RVM	**0.9693**	**0.9745**	0.9729	0.9732	0.9375	0.9403	0.9550	0.9433
	(0.0296)	**(0.0294)**	(0.0416)	(0.0186)	(0.1015)	(0.0920)	(0.1088)	(0.0580)
Hierarchical Bayesian RVM	0.9679	0.9710	0.9650	0.9731	0.9383	0.9325	0.9275	0.9414
	(0.0376)	(0.0286)	(0.0482)	(0.0171)	(0.1212)	(0.0860)	(0.1390)	(0.0541)
Generic PRVM	0.9637	0.9700	**0.9786**	**0.9737**	0.9401	0.9400	0.9450	**0.9467**
	(0.0091)	(0.0350)	**(0.0542)**	**(0.0330)**	(0.0905)	(0.1038)	(0.1667)	**(0.0934)**
Hierarchical PRVM	0.9651	0.9719	0.9764	0.9619	**0.9411**	**0.9458**	**0.9575**	0.9431
	(0.1967)	(0.0526)	(0.0512)	(0.0302)	**(0.0975)**	**(0.0729)**	**(0.1111)**	(0.0833)

^{a} $(n_n^{train} = 30, n_p^{train} = 12)$, ^{b} $(n_n^{test} = 30, n_p^{test} = 12)$, ^{c} $(n_n^{stest} = 10, n_p^{stest} = 4)$, ^{d} $(n_n^{ltest} = 90, n_p^{ltest} = 36)$

TABLE 5. The Results of Simulated Data Study ($b = 5$).

	$r_g^{train\,a}$	$r_g^{test\,b}$	$r_g^{stest\,c}$	$r_g^{ltest\,d}$	$r_p^{train\,a}$	$r_p^{test\,b}$	$r_p^{stest\,c}$	$r_p^{ltest\,d}$
Original RVM	0.9233	0.9198	0.9140	0.8897	0.4332	0.3183	0.2884	0.3005
	(0.0262)	(0.0822)	(0.1522)	(0.1422)	(0.4111)	(0.2211)	(0.2995)	(0.3011)
Generic Bayesian RVM	0.8956	0.8694	0.8733	0.8703	0.3833	0.2250	0.2250	0.2306
	(0.0642)	(0.0465)	(0.0562)	(0.0421)	(0.3896)	(0.2807)	(0.3518)	(0.2600)
Hierarchical Bayesian RVM	0.9419	0.8964	0.8883	0.8893	0.7100	0.5300	0.5850	0.5489
	(0.0667)	(0.0674)	(0.0943)	(0.0711)	(0.4203)	(0.3846)	(0.4425)	(0.3689)
Generic PRVM	**0.9700**	0.9700	0.9167	0.9600	0.7805	0.8633	0.8500	0.8411
	(0.1139)	(0.0317)	(0.0512)	(0.0241)	(0.1140)	(0.1639)	(0.2901)	(0.1406)
Hierarchical PRVM	**0.9700**	**0.9781**	**0.9750**	**0.9606**	**0.7997**	**0.8837**	**0.9000**	**0.8500**
	(0.0213)	**(0.0316)**	**(0.0547)**	**(0.0223)**	**(0.1033)**	**(0.1631)**	**(0.2052)**	**(0.1362)**

^{a} $(n_n^{train} = 30, n_p^{train} = 6)$, ^{b} $(n_n^{test} = 30, n_p^{test} = 6)$, ^{c} $(n_n^{stest} = 10, n_p^{stest} = 2)$, ^{d} $(n_n^{ltest} = 90, n_p^{ltest} = 18)$

Digestive and Kidney Diseases. It consists of several medical predictor variables and one target variable. Predictor variables include the number of pregnancies the patient has had, their BMI, insulin level, age, and so on. It is modified by [3]) to rename the output as ''positive'' and ''negative'' to create an imbalanced dataset. The ''segment0'' data comes from the Image Segmentation DataSet (see [8]), whose instances were drawn randomly from a database of 7 outdoor images. The images were hand-segmented to create a classification for every pixel, and each instance is a 3×3 regions. The ''vowel0'' dataset is an imbalanced version of the Connectionist Bench (Vowel Recognition-Deterding Data) DataSet (see [8]), which consists of a three-dimensional array: speaker, vowel, and input. The speakers are indexed by integers $0 - 89$, the vowels are indexed by integers $0 - 10$. For each utterance, there are ten floating-point input values, with array indices $0 - 9$. The ''glass5'' also originally comes from the USA Forensic Science Service, including of 6 types of glasses, which are defined in terms of their oxide content (see [8]).

TABLE 6. The Results of Simulated Data Study ($b = 10$).

	$r_g^{train\,a}$	$r_g^{test\,b}$	$r_g^{stest\,c}$	$r_g^{ltest\,d}$	$r_p^{train\,a}$	$r_p^{test\,b}$	$r_p^{stest\,c}$	$r_p^{ltest\,d}$
Original RVM	0.9091	0.9091	0.9091	0.9091	0.0000	0.0000	0.0000	0.0000
	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)
Generic Bayesian RVM	0.9091	0.9091	0.9091	0.9091	0.0000	0.0000	0.0000	0.0000
	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)
Hierarchical Bayesian RVM	0.9503	0.9148	0.9118	0.9187	0.5233	0.1667	0.2200	0.1922
	(0.0357)	(0.0417)	(0.0758)	(0.0332)	(0.3914)	(0.2485)	(0.4163)	(0.2118)
Generic PRVM	0.9771	0.9654	0.9691	0.9631	0.7333	0.7267	0.7600	0.7467
	(0.1622)	(0.0274)	(0.0472)	(0.0000)	(0.0479)	(0.2988)	(0.4314)	(0.2480)
Hierarchical PRVM	**0.9802**	**0.9757**	**0.9818**	**0.9797**	**0.7434**	**0.8842**	**0.9113**	**0.8444**
	(0.0236)	**(0.0395)**	**(0.0407)**	**(0.0124)**	**(0.2214)**	**(0.1856)**	**(0.1431)**	**(0.1685)**

^{a} $(n_n^{train} = 30, n_p^{train} = 3)$, ^{b} $(n_n^{test} = 30, n_p^{test} = 3)$, ^{c} $(n_n^{stest} = 10, n_p^{stest} = 1)$, ^{d} $(n_n^{ltest} = 90, n_p^{ltest} = 9)$

TABLE 7. The Classification Results of Real Datasets.[a]

Dataset	b	Dimension	Total Data Size	r_g^{train}	r_g^{test}	r_p^{train}	r_p^{test}
pima	1.87	8	768	0.9331	0.9366	0.9132	0.9184
				0.9155	0.9271	0.9155	0.9183
				0.9140	0.9229	0.9773	**0.9588**
				0.9502	**0.9411**	0.9527	0.9344
				0.9689	0.9331	**0.9774**	0.9551
segment0	6.02	19	2308	0.9153	0.9201	0.9104	0.9006
				0.9221	0.9133	0.8935	0.8994
				0.9066	0.9161	**0.9122**	**0.9205**
				0.9182	0.9074	0.9072	0.9032
				0.9077	**0.9216**	0.9103	0.9200
vowel0	9.98	13	988	0.8662	0.8721	0.0000	0.0000
				0.9012	0.9195	0.3012	0.3558
				0.9363	0.9388	0.8331	0.8733
				0.9103	0.9131	0.3611	0.3210
				0.9411	**0.9416**	**0.8591**	**0.8825**
glass5	22.78	9	214	0.9579	0.9579	0.0000	0.0000
				0.9579	0.9579	0.0000	0.0000
				0.9378	0.9641	0.7783	0.8739
				0.9579	0.9579	0.0000	0.0000
				0.9688	**0.9772**	**0.8892**	**0.9103**

[a] Results in the table are listed by Original RVM, Generic Bayesian RVM, Hierarchical Bayesian RVM, Generic PRVM, and Hierarchical PRVM from top to bottom.

All the datasets are randomly divided into two parts: the training data (80%) and the test data (20%). Four criteria are chosen from Table 1, r_g^{tarin}, r_g^{test}, r_p^{tarin}, and r_p^{test}. The classification results are listed in Table 7. In the weakly imbalanced dataset, all five algorithms are similar. With the increase of the imbalanced index b, the Hierarchical models perform better.

TABLE 8. Epalsed Programming Time [a] of Bayesian RVM and PRVM Models.[b]

Data Size	30-30	30-15	30-12	30-6	30-3
Generic Bayesian RVM	87758.2994 6419.9621	40447.4915 3565.7886	38074.4325 (2127.4783)	14604.5066 (1049.5469)	16167.2713 (3893.8587)
Generic PRVM [c]	**236.5257** **(1.5314)**	**45.1316** **(1.5714)**	**42.4868** **(8.0555)**	**41.1134** **(3.1998)**	**40.6392** **(6.0307)**

[a] Time is measured in seconds.
[b] R Programmings are conducted on Intel(R) Xeon(R) Silver 4110 CPU @ 2.10GHz.
[c] AdapSamp::rARS is used for log-concave posterior sampling.

V. COMPARISON BETWEEN THE BAYESIAN RVM AND PRVM

We can conclude from the numeric studies that the Bayesian RVM and PRVM models are similar for classification accuracy results. The only theoretic difference between them is the link functions for the likelihood. The Bayesian RVM uses the logistic link function, but the PRVM employs the probit one. It is still worth discussing more comparisons between them.

A. ELAPSED PROGRAMMING TIME

The Bayesian RVM model needs the ARS method to conduct the sampling process. The model has to conduct one sampling iteration for every dimension of $\boldsymbol{w}$. Also, we do not have a strategy to determine the suitable support values for the ARS sampling process, so the ARS method could be inefficient. PRVM can sample the whole vector $\boldsymbol{w}$ directly from its posterior since it follows a Multivariate Normal distribution. Table 8 lists the elapsed programming time for these two models. We conduct every experiment on the simulated Gaussian datasets with 5000 iterations. Repeat every experiment 100 times and calculate the mean and standard deviation values listed in Table 8. The PRVM is significantly more efficient than the Bayesian RVM.

B. MODEL SELECTION

Many parameter estimation problems adopt the likelihood function as the objective function. When enough training data are available, the accuracy of models can be improved continuously. However, as the cost of model complexity increases, it also brings up a widespread problem in machine learning, namely overfitting. Therefore, the problem of model selection seeks an optimal balance between the complexity of the model and the model's ability to describe the dataset. Many information criteria have been proposed to avoid the overfitting problem by adding a penalty for model complexity. We introduce two commonly used model selection methods: Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC).

AIC is a standard to measure the goodness of model fitting (see [1]). It is based on entropy and provides a standard to balance the complexity of the model estimation and the goodness of model fitting. Generally, AIC is defined as:

$$AIC = 2\,k - 2\ln(\hat{L}), \tag{37}$$

TABLE 9. Maximum Likelihood Value of Bayesian RVM and PRVM Models.

Data Size	30-30	30-15	30-12	30-6	30-3
Bayesian RVM	0.9999928 (1.0120×10^{-6})	0.9999928 (1.0120×10^{-6})	0.9999928 (1.0120×10^{-6})	0.9999927 (1.8935×10^{-6})	0.9999927 (1.0593×10^{-6})
PRVM	**0.9999986** (4.9653×10^{-7})	**0.9999986** (5.3061×10^{-7})	**0.9999986** (5.3061×10^{-7})	**0.9999985** (4.1018×10^{-7})	**0.9999986** (5.1192×10^{-7})

where k is the number of model parameters, and $\hat{L}$ is the maximum value of the likelihood function. It is common to choose the model with minimum AIC.

BIC is similar to AIC, and it is also used for model selection (see [19]). The penalty term of BIC is larger than AIC since BIC also considers the number of samples. When the sample size is large, it can effectively prevent the model's complexity from being too high. The definition of BIC is:

$$BIC = k\ln(n) - 2\ln(\hat{L}), \tag{38}$$

where k is the number of model parameters, n is the number of samples, and $\hat{L}$ is the maximum value of the likelihood function. Given the same data, the two RVM models in this project, Bayesian RVM and PRVM, have the same k and n. So we only need to focus on $\hat{L}$ to compare them in the cases of AIC and BIC. Choose the Gaussian simulated datasets defined in (35) and (36) and repeat the process of seeking maximum-likelihood value 100 times for every simulated dataset in the Bayesian RVM and PRVM. Table 9 shows the mean and standard deviation results. In the case of $\hat{L}$, the Bayesian RVM and PRVM are similar. However, the PRVM seems a little preferred than the Bayesian RVM.

VI. CONCLUSION COMMENTS

Two RVM with the probit link function (PRVM) classification algorithms are proposed in this article. The posterior of the weight parameter in the original RVM has no closed-form solution, so it is hard to conduct. The intricate likelihood function is the reason for this. The original RVM uses the logistic link function to construct the likelihood function, which leads to all the difficulties in the algorithm. Benefiting from the probit link function, the posterior of the weight parameter in PRVM follows a Multivariate Normal distribution. PRVM is a more compact algorithm, and its programming speed is significantly faster than the Bayesia RVM, which is the algorithm proposed in [29]. The Fully Hierarchical PRVM follows the hyperprior structure in [9] to improve the classification performance in the imbalanced data problem.

A study of the comparison between Bayesian RVM and PRVM is conducted. The numeric studies show that these two models have similar classification accuracy results. For the severely imbalanced data, PRVM is significantly better than the Bayesian RVM. Also, PRVM is more efficient than the Bayesian RVM in the case of programming time. From the perspective of model selection, PRVM is a little preferred than the Bayesian RVM in AIC and BIC cases.

APPENDIX A

A1. PROOF OF THEOREM 1

$$
\begin{aligned}
&p(\boldsymbol{w}|\boldsymbol{y}^{train}, \boldsymbol{\alpha}, \boldsymbol{\mu}) \\
&\quad \propto exp(-\frac{1}{2}\boldsymbol{w}^T\boldsymbol{A}\boldsymbol{w})\prod_{i=1}^{n}\phi(\mu_i - K_i\boldsymbol{w}) \\
&\quad = exp\Big[-\frac{1}{2}\big(\boldsymbol{w}^T\boldsymbol{A}\boldsymbol{w} + \sum_{i=1}^{n}(\mu_i - K_i\boldsymbol{w})^2\big)\Big] \\
&\quad \propto exp\Big[-\frac{1}{2}\Big(\boldsymbol{w}^T\boldsymbol{A}\boldsymbol{w} + \sum_{i=1}^{n}\big((K_i\boldsymbol{w})^2 - 2\mu_i K_i\boldsymbol{w}\big)\Big)\Big] \\
&\quad = exp\Big[-\frac{1}{2}\big(\boldsymbol{w}^T\boldsymbol{A}\boldsymbol{w} + \boldsymbol{w}^T\boldsymbol{K}^T\boldsymbol{K}\boldsymbol{w} - 2\boldsymbol{w}^T\boldsymbol{K}^T\boldsymbol{\mu}\big)\Big] \\
&\quad = exp\Big[-\frac{1}{2}(\boldsymbol{w} - \hat{\boldsymbol{w}})\boldsymbol{M}(\boldsymbol{w} - \hat{\boldsymbol{w}})^T\Big] \\
&\quad \propto \mathcal{N}_{n+1}(\hat{\boldsymbol{w}}, \boldsymbol{M}^{-1}),
\end{aligned}
$$

where $\boldsymbol{M} = \boldsymbol{A} + \boldsymbol{K}^T\boldsymbol{K}$, $\hat{\boldsymbol{w}} = \boldsymbol{M}^{-1}\boldsymbol{K}^T\boldsymbol{\mu}$.

A2. PROOF OF LEMMA 1

It is obvious that

$$
\begin{aligned}
D &= b + \Phi^{-1}(u\Phi(-b)) \\
&\Rightarrow u(D) = \frac{\Phi(D-b)}{\Phi(-b)} \\
&\Rightarrow P(D) = \frac{\partial u(D)}{\partial D}\mathbf{1}_{(0\le u\le 1)} = \phi(D-b)\,\mathbf{1}_{(D\le 0)}.
\end{aligned}
$$

It is a similar process to prove another statement.

A3. PROOF OF THEOREM 2

The conditional posterior of any η_k in $\boldsymbol{\eta}$ is

$$
\begin{aligned}
&p(\eta_k \mid \text{others}) \\
&\propto \exp\Big[\frac{\eta_k}{2} - \frac{1}{2}\exp(\eta_k)w_k^2 - \frac{(\eta_k-\mu)^2}{2\tau^2(1-\rho)} \\
&\quad + \frac{\rho(\eta_k-\mu)^2}{2\tau^2(1-\rho)(1+n\rho)} + \frac{\rho(\eta_k-\mu)\sum_{s=0,s\neq k}^{n}(\eta_s-\mu)}{\tau^2(1-\rho)(1+n\rho)}\Big],
\end{aligned}
$$

where $k = 0, 1, 2, \ldots, n$. For a constant C, the log-posterior of η_k is

$$
\begin{aligned}
l_k^{\eta} &= \log p(\eta_k|\text{ others}) \\
&= C + \frac{\eta_k}{2} - \frac{1}{2}\exp(\eta_k)w_k^2 - \frac{1+(n-1)\rho}{2\tau^2(1-\rho)(1+n\rho)} \\
&\quad \cdot(\eta_k-\mu)^2 + \frac{\rho(\eta_k-\mu)\sum_{s=0,s\neq k}^{n}(\eta_s-\mu)}{\tau^2(1-\rho)(1+n\rho)}.
\end{aligned}
$$

The second divergence is

$$
\frac{\partial^2}{\partial\eta_k^2}l_k^{\eta} = -\frac{1}{2}\exp(\eta_k)w_k^2 - \frac{1+(n-1)\rho}{\tau^2(1-\rho)(1+n\rho)}.
$$

Because $1+(n-1)\rho > 0$ for any $\rho \in (-1, 1)$, $\frac{\partial^2}{\partial\eta_k^2}l_k^{\eta} < 0$ always holds.

APPENDIX B

B1. ADAPTIVE REJECTION SAMPLING METHOD

Sampling plays an important role in statistics. Sampling from the conventional distributions can be done directly by statistics software like *R*, but it is hard to do the sampling from the unconventional distributions. [5] proposed the Rejective Sampling method to conduct the sampling of unconventional distributions. It samples from a proposed conventional distribution and sets a ratio to decide the acceptance or rejection of this sampling value. But the Rejection Sampling method needs an upper boundary to restrict the proposed conventional distribution, and people do not have a specific approach to determine this boundary. The original idea of the Adaptive Rejection Sampling (ARS) method was proposed by [11]. It can determine the certain upper boundary of the unconventional distributions and has a high acceptance rate for the sampling process. For distributions whose probability density functions are log-concave, the Adaptive Rejection Sampling (ARS) method is powerful and efficient.

REJECTION SAMPLING METHOD

The Rejection Sampling method is a typical Monte Carlo Sampling method. When the aim distribution $X \sim p_X(x)$ is not suitable for direct sampling, the Rejection Sampling method employs a proposal distribution $Y \sim g_Y(y)$, which can produce the sampling values quickly. The basic idea is to sample a random value y' from the proposal distribution, then accept y' as the sample of aim distribution $p_X(x)$ with the probability of $p_X(y')/(M\,g_Y(y'))$, where $1 < M < \infty$ is a constant.

Algorithm 4 The Rejection Sampling Method

Input. The sample size N and the aim distribution $p_X(x)$.
0. Determine the proposal distribution $g_Y(y)$ and constant M, let $i = 1$;
while $i \le N$ **do:**
1. Sample $u \sim U(0, 1)$, $y_i \sim g_Y(y)$;
2. If $u < p_X(y_i)/(M\,g_Y(y_i))$ then $x_i = y_i$; else repeat Steps **1** and **2**;
3. $i = i + 1$;
Output. $\boldsymbol{x} = \{x_1, x_2, \ldots, x_N\}$ are the sample values.

Although the Rejection Sampling method works, it would produce inaccurate results and the process is inefficient sometimes. First, if the aim distribution $p_X(x)$ has peak value in some internals, the Rejection Sampling method may include samples that should not have been accepted. When the dimension of the aim distribution increases, the ratio of $p_X(y_i)/g_Y(y_i)$ convergence to 0 with N increasing. This would result in that a useful sample is rejected before it is produced. The most challenging thing is to find the proper proposal distribution $g_Y(y)$ and the bounded constant M.

ADAPTIVE REJECTION SAMPLING METHOD

For a better sampling performance in practice, we need a proposal distribution closer to the aim one. [11] proposed

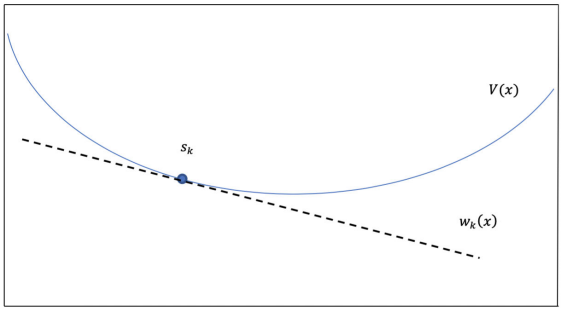


FIGURE 4. The Tangential Function $w_k(x)$ at s_k in ARS Method.

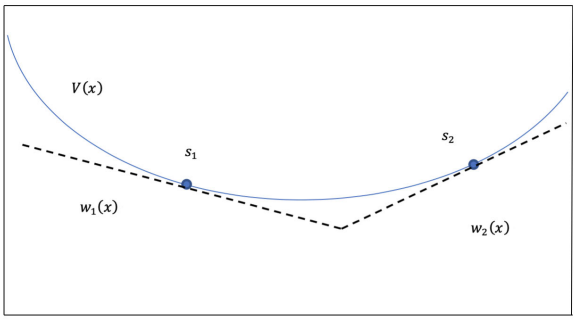


FIGURE 5. $w_n(x)$ Based on Two Support Points in ARS Method.

the Adaptive Rejection Sampling method idea. It exercises a series of envelope functions to do the sampling. If one sample value is rejected and it will be included to construct a more compact envelope function. First, we define the concavity and convexity of functions.

Definition 2: If $f(x)$ is continuous on $[a, b]$ and the second derivative exists.

(1) When $f''(x) > 0$ on (a, b), $f(x)$ is convex on $[a, b]$;

(2) When $f''(x) < 0$ on (a, b), $f(x)$ is concave on $[a, b]$.

A necessary assumption of the ARS method is that the aim distribution is log-concave. If the aim distribution function is $p_X(x)$ defined on $\mathbb{D} \subseteq \mathbb{R}$, based on Definition 2, let $V_X(x) = -log(p_X(x))$ and $V''_X((x) > 0$ always holds on $\mathbb{D}$. ARS method needs serial support points $s_1 < s_2 < \ldots < s_m$ to construct the envelope function. The more support points there are, the higher acceptable rate the sampling process will have at the efficiency cost. In Figure 4, let $w_k(x)$ be the tangential function of $V_X(x)$ at support point s_k:

$$w_k(x) = V'_X(s_k)(x - s_k) + V_X(s_k), \tag{39}$$

where $k = 1, 2, \ldots, m$. We obtain m tangential functions based on the support points.

$$W_n(x) = max\{w_1(x), w_2(x), \ldots, w_m(x)\}. \tag{40}$$

Because $V_X(x)$ is the convex function on $\mathbb{D}$, and $w_k(x)$ are the tangential functions at s_k, where $k = 1, 2, \ldots, m$. So $W_n(x) \leq V_X(x)$. Figure 5 shows the $W_n(x)$ based on two support points. After transformation, we have an envelope function

$$exp(-W_n(x)) \geq exp(-V_X(x)) = p_X(x). \tag{41}$$

Then a piecewise proposal function is obtained based on $exp(-W_n(x))$:

$$\pi_n(x) = c_n exp(-W_n(x)), \tag{42}$$

where $c_n = (\int_{\mathbb{D}} exp(-W_n(x))dx)^{-1}$ is the regularization constant. The basic idea of the ARS method is to first sample the random values u from $U(0, 1)$, x' from $\pi_n(x)$. If $u < \frac{p_X(x')}{exp(-W_n(x'))}$, we accept x' as the sample value from $p_X(x)$. Otherwise, we add x' into the support points set S_n to obtain $S_{n+1} = S_n \bigcup x'$, which will construct a more compact $W_{n+1}(x)$. Repeat this step until we have enough acceptable samples.

Algorithm 5 The Adaptive Rejection Sampling Method

Input. The sample size N, the aim distribution $p_X(x)$.

0. Let $i = 1$ and determine the support points set S_i;

while $i \leq N$ **do:**

1. $V_X(x) = -log(p_X(x))$ and construct the tangential functions of $V_X(x)$ based on the points in S_i;

2. Sample $u \sim U(0, 1)$, $x' \sim \pi_n(x) \propto exp(-W_n(x))$;

3. If $u < \frac{p_X(x')}{exp(-W_n(x'))}$, $x_i = x'$ and $S_{i+1} = S_i$; Else, $S_i = S_i \bigcup x'$ and return to Step **1**.

Output. $\boldsymbol{x} = \{x_1, x_2, \ldots, x_N\}$ are the sample values.

Thereafter, several improved ARS methods were proposed. [10] proposed the MABS method, which combines the Metropolis-Hastings and ARS methods. But this approach produces a Markov chain, which makes the samples are related to each other. [22] proposed a new ARS method that can also solve log-convex distribution sampling. It divides the distribution function into several sections based on the concavity and convexity, then sampling every section. [31] summarized all the existing ARS methods and published the *AdapSamp* package in *R*. In this project, we use *AdapSamp* :: *rARS* function in *R* to conduct the Adaptive Rejection Sampling method.

B2. ADAPTIVE REJECTION SAMPLING METHOD

This subsection introduces the ratio of uniforms method, which is a random number generation approach. This method was original proposed by [13]. Then [17] further improved this method. Suppose that a bivariate random variable (U_1, U_2) is uniformly distributed and satisfies the following inequality:

$$0 \leq U_1 \leq \sqrt{g\,(U_2/U_1)},$$

where $g(x)$ is any nonnegative function. So $X = U_2/U_1$ has a density function $f(x) = \frac{h(x)}{\int h(x)\mathrm{d}x}$. The joint density of U_1 and U_2, denoted by $f_{12}\,(u_1, u_2)$ is

$$f_{12}\,(u_1, u_2) = \begin{cases} k, & \text{if } 0 \leq u_1 \leq \sqrt{g\,(u_2/u_1)} \\ 0, & \text{otherwise}, \end{cases}$$

where k is a constant number. Conduct the following transformation from (u_1, u_2) to (x, y):

$$x = \frac{u_2}{u_1}, \quad y = u_1.$$

It is evident that $u_1 = y, u_2 = xy$. So the Jacobian for this simple transformation is:

$$J = \begin{vmatrix} \frac{\partial u_1}{\partial x} & \frac{\partial u_1}{\partial y} \\ \frac{\partial u_2}{\partial x} & \frac{\partial u_2}{\partial y} \end{vmatrix} = \begin{vmatrix} 0 & 1 \\ y & x \end{vmatrix} = -y.$$

Rewritten $f_{xy}(x, y)$ as:

$$f_{xy}(x, y) = |J| f_{12}(y, x\ y) = ky,$$

where $0 \leq y \leq \sqrt{g(x)}$. The marginal density of X, denoted by $f_x(x)$, is obtained as follows:

$$\begin{aligned} f_x(x) &= \int_0^{\sqrt{g(x)}} f_{xy}(x, y)\mathrm{d}y \\ &= \int_0^{\sqrt{g(x)}} ky\mathrm{d}y \\ &= k\left[\frac{y^2}{2}\right]_0^{\sqrt{g(x)}} \\ &= \frac{k}{2}g(x) \\ &= f(x), \end{aligned}$$

where k is taken as $k = \frac{2}{\int g(x)\mathrm{d}x}$. Thus, it is shown that $f_x(\cdot)$ is equivalent to $f(\cdot)$.

In practice, we need to choose the rectangle which encloses the area $0 \leq U_1 \leq \sqrt{g\,(U_2/U_1)}$ on the domain of (U_1, U_2). The basic idea is to generate a uniform point in the rectangle, and reject the point which does not satisfy $0 \leq u_1 \leq \sqrt{g\,(u_2/u_1)}$. So in this method, we generate two independent uniform random draws u_1 and u_2 from $U(0, b)$ and $U(c, d)$, respectively. The rectangle is given by:

$$0 \leq u_1 \leq b, \quad c \leq u_2 \leq d,$$

where b, c and d are given by:

$$b = \sup_x \sqrt{g(x)}, \quad c = -\sup_x x\sqrt{g(x)}, \quad d = \sup_x x\sqrt{g(x)}.$$

The sampling process is as follows (see [17]):

(1) Generate u_1 and u_2 independently from $U(0, b)$ and $U(c, d)$;

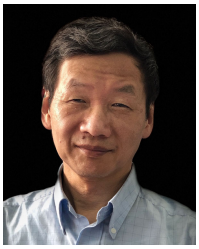
(2) If $u_1^2 \leq h\,(u_2/u_1)$, set $x = u_2/u_1$. Else, return to **(1)**.

REFERENCES

[1] H. Akaike, "Stochastic theory of minimal realization," *IEEE Trans. Autom. Control*, vol. 19, no. 6, pp. 667–674, Dec. 1974.
[2] J. Albert and S. Chib, "Bayesian residual analysis for binary response regression models," *Biometrika*, vol. 82, no. 4, pp. 747–769, 1995.
[3] J. Alcalá-Fdez, A. Fernández, J. Luengo, J. Derrac, S. García, L. Sánchez, and F. Herrera, "Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework," *J. Multiple-Valued Log. Soft Comput.*, vol. 17, pp. 255–287, Jun. 2011.
[4] A. C. Braun, U. Weidner, and S. Hinz, "Classification in high-dimensional feature spaces—Assessment using SVM, IVM and RVM with focus on simulated EnMAP data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 436–443, Apr. 2012.
[5] G. Casella, C. Robert, and M. Wells, "Generalized accept-reject sampling schemes," in *A Festschrift for Herman Rubin*. Beachwood, OH, USA: Institute of Mathematical Statistics, 2004, pp. 342–347.
[6] B. Demir and S. Erturk, "Hyperspectral image classification using relevance vector machines," *IEEE Geosci. Remote Sens. Lett.*, vol. 4, no. 4, pp. 586–590, Oct. 2007.
[7] W. Dinghai, Z. Peilin, and Z. Yingtang, "Study on diesel engine faults diagnosis based on time frequency singular value spectrum and RVM," *J. Mech. Strength*, vol. 33, no. 3, pp. 317–323, 2011.
[8] D. Dua and C. Graff. (2017). *UCI Machine Learning Repository*. [Online]. Available: http://archive.ics.uci.edu/ml
[9] E. Fokoué, D. Sun, and P. Goel, "Fully Bayesian analysis of the relevance vector machine with an extended hierarchical prior structure," *Stat. Methodol.*, vol. 8, no. 1, pp. 83–96, Jan. 2011.
[10] W. R. Gilks, N. G. Best, and K. Tan, "Adaptive rejection Metropolis sampling within Gibbs sampling," *J. Roy. Stat. Soc. Ser. C, Appl. Statist.*, vol. 44, no. 4, pp. 455–472, 1995.
[11] W. R. Gilks and P. Wild, "Adaptive rejection sampling for Gibbs sampling," *J. Roy. Stat. Soc. Ser. C, Appl. Statist.*, vol. 41, no. 2, pp. 337–348, Jun. 1992.
[12] Y. He, C. Li, T. Wang, T. Shi, L. Tao, and W. Yuan, "Incipient fault diagnosis method for IGBT drive circuit based on improved SAE," *IEEE Access*, vol. 7, pp. 92410–92418, 2019.
[13] A. J. Kinderman and J. F. Monahan, "Computer generation of random variables using the ratio of uniform deviates," *ACM Trans. Math. Softw.*, vol. 3, no. 3, pp. 257–260, Sep. 1977.
[14] N. Nikolaev and P. Tino, "Sequential relevance vector machine learning from time series," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, Jul. 2005, pp. 1308–1313.
[15] M. Pal, and G. M. Foody, "Evaluation of SVM, RVM and SMLR for accurate image classification with limited ground data," *IEEE J. Sel. Topics Appl. Ear*, vol. 5, no. 5, pp. 1344–1355, Oct. 2012.
[16] C. Ren, "Topics in Bayesian estimation: Frequentist risks and hierarchical models for time to pregnancy," Tech. Rep., 2002.
[17] B. D. Ripley and M. Thompson, "Regression techniques for the detection of analytical bias," *Analyst*, vol. 112, no. 4, pp. 377–383, 1987.
[18] B. Schölkopf, C. J. Burges, and A. J. Smola, *Advances in Kernel Methods: Support Vector Learning*. Cambridge, MA, USA: MIT Press, 1999.
[19] L. B. Schwarz, S. C. Graves, and W. H. Hausman, "Scheduling policies for automatic warehousing systems: Simulation results," *AIIE Trans.*, vol. 10, no. 3, pp. 260–270, Sep. 1978.
[20] C. Silva and B. Ribeiro, "Scaling text classification with relevance vector machines," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, Oct. 2006, pp. 4186–4191.
[21] X. Sun, X. Wang, D. Cai, Z. Li, Y. Gao, and X. Wang, "Multivariate seawater quality prediction based on PCA-RVM supported by edge computing towards smart ocean," *IEEE Access*, vol. 8, pp. 54506–54513, 2020.
[22] Y. W. Teh and D. Gorur, "Indian buffet processes with power-law behavior," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 1838–1846.
[23] A. Thakur, M. Goldbaum, and S. Yousefi, "Convex representations using deep archetypal analysis for predicting glaucoma," *IEEE J. Transl. Eng. Health Med.*, vol. 8, 2020, Art. no. 3800107.
[24] M. E. Tipping, "The relevance vector machine," in *Proc. Adv. Neural Inf. Process. Syst.*, 2000, pp. 652–658.
[25] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *J. Mach. Learn. Res.*, vol. 1, pp. 211–244, Sep. 2001.
[26] M. E. Tipping and A. C. Faul, "Fast marginal likelihood maximisation for sparse Bayesian models," in *Proc. AISTATS*, Jan. 2003, pp. 1–13.
[27] V. Vapnik, "Statistical learning theory," Tech. Rep., 1998.
[28] H. Wang, C. Wu, T. Li, Y. He, P. Chen, and A. Bezerianos, "Driving fatigue classification based on fusion entropy analysis combining EOG and EEG," *IEEE Access*, vol. 7, pp. 61975–61986, 2019.
[29] W. Wang, D. Sun, and Z. He, "Fully Bayesian analysis of the relevance vector machine classification for imbalanced data," 2020, *arXiv:2007.13140*. [Online]. Available: http://arxiv.org/abs/2007.13140
[30] X. Xiang-min, M. Yun-feng, X. Jia-ni, and Z. Feng-le, "Classification performance comparison between RVM and SVM," in *Proc. Int. Workshop Anti-Counterfeiting, Secur. Identificat. (ASID)*, Apr. 2007, pp. 208–211.

[31] D. Zhang, "Adaptive sampling algorithms and its r package development," M.S. thesis, East China Normal Univ., Shanghai, China, 2007.
[32] L. Zunxiong, Z. Deyun, S. Qindong, and X. Zheng, "Mid-term electric load prediction based on the relevant vector machine," *J. Xian Jiaotong Univ.*, vol. 38, no. 10, pp. 1005–1008, 2004.

WENYANG WANG was born in Dalian, Liaoning, China, in 1991. He received the B.Sc. degree in mathematics from the Dalian University of Technology in Summer 2014, the M.A. degree in statistics in Summer 2016, and the Ph.D. degree in statistics from the University of Missouri, in 2020. In Fall 2014, he joined a graduate program with the Department of Statistics, University of Missouri, Columbia, USA. He is currently an Assistant Professor with Dalian Maritime University, China. His research interests include Bayesian analysis, machine learning, and statistical finance.



DONGCHU SUN received the Ph.D. degree in statistics from Purdue University, West Lafayette, IN, USA, in 1991. He is currently a Research Professor with the Department of Statistics, University of Nebraska–Lincoln, and also an Emeritus Faculty with the Department of Statistics, University of Missouri. His current work and research include several distinct, but related, topics. Virtually all of his research is in the area of Bayesian statistics, involving almost all branches of statistics that has seen tremendous growth in the last decade. The goal of Bayesian methods is to improve statistical inference by incorporating information a researcher knows about a problem before collecting the data into the analysis. Incorporating such prior information can result in much improved analysis. Until recently, it was impossible to apply Bayesian methods to a large class of interesting problems because the computations were intractable. However, the advent of low-cost high-speed personal computers and the development of new techniques of Markov chain Monte Carlo simulation as Gibbs sampling have made Bayesian analysis feasible in numerous new areas. While his research before tenure was mostly on optimal properties of Bayesian analysis such as frequentist properties both point estimation and interval estimation, he has developed a systematic research agenda. During the last 25 years, he has not only continue doing research on theoretical properties of Bayesian methodology, he has done a great deal of interdisciplinary research by applying modern statistics to solve complicated practical problems such as wildlife expenditure in wildlife management, response time in psychology, vector autoregressive models in micro-economics, cancer mortality, and incidence rates in epidemiology study and time to pregnancy in reproductivity study. Meantime, he believes the natural connection between research and graduate teaching and advising graduate students. He has been elected as an Ordinary Member of the International Statistical Institute, the Fellow of the American Statistical Association, and the Fellow of the Institute of Mathematical Statistics.

PENG SHAO received the dual B.S. degree in information and computing science with the Information Management and Information System Department, Dalian University of Technology, Dalian, China, in 2014, and the M.A. degree in statistics from the University of Missouri, Columbia, USA, in 2016, where he is currently pursuing the Ph.D. degree.

From 2016 to 2018, he was a Research Assistant with the Conservation Research Center, Missouri Department of Conservation, Columbia, USA. His main work is statistical methods in habitat conservation and hunting management. His research interests include Bayesian method, categorical data analysis, discrete choice model, contingency table, and high dimensional data.

HAIBO KUANG received the Ph.D. degree in management from the Dalian University of Technology, China. He is currently a Professor with the School of Maritime Economics and Management, Dalian Maritime University (DMU), where he is also the President of the Collaborative Innovation Center for Transport Studies. He has served as the Team Leader for Port Synergistic Development and Green Growth. He has published more than 100 reviewed journal articles and earned more than 20 provincial and industrial awards of China. His research interests include maritime policy and management, green supply chain management, and maritime big data. He was awarded a grant by the Program of Innovative Research Team at the University of the Ministry of Education of China.

CONG SUI was born in Shenyang, Liaoning, China, in 1977. He received the B.S. and M.S. degrees in applied mathematics from Liaoning University, Shenyang, Liaoning, China, in 2002 and 2006, respectively, and the Ph.D. degree in financial engineering from the Dalian University of Technology, Dalian, Liaoning, China, in 2010.

He is currently a Finance Professor with the School of Maritime Economics and Management and also the Vice Director of the Collaborative Innovation Center for Transport Studies, Dalian Maritime University. He has authored more than 20 articles. His current research interests include mispricing, option price, shipping finance, risk contagion, and systemic risk.

Dr. Sui is the Managing Director and Deputy Secretary-General of the Financial Systems Engineering Committee, Chinese Society of Systems Engineering; the Managing Director of the Financial Measurement and Risk Management Committee, Chinese Society for Management Science and Engineering; a member of the council of Commission on Quantitative Finance and Insurance of China Superior Selection Law and Economic Mathematics Research Institute.