Journal of Statistics Applications & Probability
An International Journal

http://dx.doi.org/10.18576/jsap/090101

Extended Bayesian Framework for Multicategory Support Vector Machine

Yeqian Liu

Department of Mathematical Sciences, Middle Tennessee State University, Murfreesboro, USA

Received: 1 Jul. 2019, Revised: 20 Nov. 2019, Accepted: 22 Nov. 2019
Published online: 1 Mar. 2020

Abstract: The support vector machine (SVM) is widely used for machine learning and artificial intelligence. Traditional support vector machine has been extended to multicategory case for multicategory classification problem. However, it does not provide an established Bayesian Framework for Multicategory Support Vector Machine. Corresponding to this, we propose Bayesian methods for multi-class support vector machine. Extensive numerical studies were conducted to evaluate the performance of the proposed method. The numerical study suggests that the proposed Bayesian framework provides good results for practical situations. In addition, an illustrative example using MIT Genome data is presented.

Keywords: Multivariate classifcation, Support Vector Machine, EM algorithm, MCMC algorithm

1 Introduction

The support vector machine(SVM) is a very popular method within the machine learning literature. Recently, it has grabbed statisticians' attention as well. The traditional SVM, designed for the binary classification problem, has a nice geometrical interpretation of discriminating one class from another by a hyperplane with the maximum margin. In SVM, the separation is achieved by hyperplane which has the largest distance to the data of the two groups.

Bayesian approach, which has been rapidly developed throughout the past thirty years, plays a very important role in statistics. Nicholas G and Steven L [1] applied it to SVM classification problem. In their paper, they developed a latent variable representation of original SVM, which helps EM or MCMC algorithms to do parameter estimation. In their method,data augmentation methods can be formulated in terms of complete data sufficient statistics, which is a considerable advantage when working with large data sets. Most of the computational expense results from repeated iterating over the data. Methods based on complete data sufficient statistics need only compute those statistics once per iteration. (Nicholas G and Steven L [1]).

Recently, it has been shown that the support vector machine (SVM) [2] admits a Bayesian interpretation through the technique of data augmentation. However, existing inference methods for the Bayesian support vector machine [3] can only handle two-category classification problem under Bayesian framework. Based on stochastic variational inference [4] and inducing points [5], we develop a Bayesian support vector machine for multicategory classification problem in this paper. The proposed Bayesian multicategory SVM not only inherits the advantage of robustness against outliers, advanced accuracy [6], and guaranteed error rate [7] from the frequentist formulation of the SVM, but like all Bayesian methods, it also has the advantage of modeling with high flexibility, automatic parameter tuning, and providing estimates of uncertainty in predictions.

We propose to extend the SVM to the multicategory case under the Bayesian framework. We will first generalize the hinge loss function and show that the formulation of the generalized multicategory SVM encompasses that of the two-category SVM, as well as maintains the good properties of the binary SVM. We introduce the multicategory SVM for the standard case as well as some modifications for the nonstandard case. Finally, we derive the dual formulation which enables us to obtain the solution, and show how to tune the model-controlling parameters in MSVM.

The paper is organized as follows: Section Two addresses the loss function and Bayesian models for multi-SVM. Section Three handles the Point estimation by EM and other related algorithms. Section Four presents the MCMC for

* Corresponding author e-mail: yeqian.liu@mtsu.edu

SVM. Sections Five covers the simulations study. Section Six illustrates the proposed methods through applying it to MIT genome data. Sections Seven and Eight comprise discussion and concluding remarks.

2 Multicategory Support Vector Machines

Let's first consider a binary outcome $y_i \in \{-1,1\}$ based on a vector of predictors $\mathbf{x}_i = (1, x_1, \cdots, x_{k-1})$ for $i = 1, \cdots, n$. The objective of the L^α-norm regularized support vector classifier is to estimate the coefficients $\boldsymbol{\beta}$ through minimizing the following penalty likelihood function

$$d_\alpha(\boldsymbol{\beta}, \nu) = \sum_{i=1}^{n} (1 - y_i \mathbf{x}_i^T \boldsymbol{\beta})_+ + \nu^{-\alpha} \sum_{j=1}^{k} \left| \frac{\beta_j}{\sigma_j} \right|^\alpha \tag{1}$$

where σ_j is the standard deviation of the j'element of $\mathbf{x}$ and ν is a tuning parameter.

2.1 Model and Notations

Now we extend this model to the multicategory case assuming that all of the classification costs are equal and no sampling bias exists in the training dataset. In addition, the k-category classification problem has to be considered. To ensure the symmetry of class label representation, we define the following vector-valued class codes, denoted by $\mathbf{y}_i$. For notational convenience, we define $\mathbf{v}_j$ for j=1,...,k as a k-dimensional vector with 1 in the jth coordinate and -1/(k-1) elsewhere. Then $\mathbf{y}_i$ is coded as $\mathbf{v}_j$ if example i belongs to class j. For instance, if example i falls into class 1, $\mathbf{y}_i = \mathbf{v}_1 = (1, -1/(k-1), \ldots - 1/(k-1))$. Accordingly, we define a k-tuple of separating functions $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_k(\mathbf{x}))$ with constraint $\sum_{j=1}^{k} f_j(\mathbf{x}) = 0$, for any $\mathbf{x} \in \mathbb{R}^d$. We also define $p_j(\mathbf{x}), j = 1, \ldots, k$ to be the conditional probabilities of k classes and constrained by $\sum_{j=1}^{k} p_j(\mathbf{x}) = 1$. We justify the utility of the sum-to-0 constraint later as we illuminate properties of the proposed method. Analogous to the two-category case, we consider $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_k(\mathbf{x})) \in \prod_{j=1}^{k} (\{1\} + H_{K_j})$, the product space of k RKHS's H_{K_j} for j=1,...,k, we assume that they are the same RKHS denoted by H_K. Define $\mathbf{Q}$ as the k $\times$ k matrix with 0 on the diagonal and 1 elsewhere. This represents the cost matrix when all of the misclassification costs are equal. Let $\mathbf{L}(\cdot)$ be a function that maps a class label $\mathbf{y}_i$ to the jth row of the matrix $\mathbf{Q}$ if $\mathbf{y}_i$ indicates class j.

We propose that to find $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_k(\mathbf{x})) \in \prod_{1}^{k} (\{1\} + H_K)$ with the sum-to-0 constraint, minimizing the following quantity is a natrual extension of SVM methodology:

$$\frac{1}{n} \sum_{i=1}^{n} \mathbf{L}(\mathbf{y}_i) \cdot (\mathbf{f}(\mathbf{x}_i) - \mathbf{y}_i)_+ + \frac{1}{2} \lambda \sum_{j=1}^{k} \|h_j\|_{H_K}^2$$

, where $(\mathbf{f}(\mathbf{x}_i) - \mathbf{y}_i)_+$ is defined as $[(f_1(\mathbf{x}_i) - y_{i1})_+, \ldots, (f_k(\mathbf{x}_i) - y_{ik})_+]$ by taking the truncate function "$(\cdot)$"$_+$ componentwise.

As the binary case, the proposed loss function has an analogous relation to the misclassification loss. That is, if $\mathbf{f}(\mathbf{x}_i)$ itself is one of the class codes, $\mathbf{L}(\mathbf{y}_i) \cdot (\mathbf{f}(\mathbf{x}_i) - \mathbf{y}_i)_+$ is $k/(k-1)$ times the misclassification loss.

We show that the generalized hinge loss function reduces to the binary hinge loss function

$$\frac{1}{n} \sum_{i=1}^{n} (1 - y_i f(\mathbf{x}_i))_+ + \lambda \sum_{j=1}^{k} \|h\|_{H_K}^2$$

when k=2. Under the binary case, $\mathbf{y}_i = (1, -1)$(1 in the binary SVM notation), then $\mathbf{L}(\mathbf{y}_i) \cdot (\mathbf{f}(\mathbf{x}_i) - \mathbf{y}_i)_+ = (0, 1) \cdot [(f_1(\mathbf{x}_i) - 1)_+, (f_2(\mathbf{x}_i) + 1)_+] = (f_2(\mathbf{x}_i) + 1)_+ = (1 - f_1(\mathbf{x}_i))_+$. Similarly, if $\mathbf{y}_i = (-1, 1)$(-1 in the binary SVM notation), $\mathbf{L}(\mathbf{y}_i) \cdot (\mathbf{f}(\mathbf{x}_i) - \mathbf{y}_i)_+ = (1 - f_2(\mathbf{x}_i))_+$. Also, note that $(\lambda/2) \sum_{i=1} 2\|h_j\|_{H_K}^2 = (\lambda/2) \times (\|h_1\|_{H_K}^2 + \| - h_1\|_{H_K}^2) = \lambda \|h_1\|_{H_K}^2$, by the fact that $h_1(\mathbf{x}) + h_2(\mathbf{x}) = 0$ for any $\mathbf{x}$, discussed later. Therefore, the binary SVM formulation is a special case of Multicategory SVM formulation when k=2.

2.2 Conditional distribution

According to the above-mentioned computations, especially equations (7) and (9), the support vector machine pseudo-posterior distribution can be expressed as the marginal of the complete data pseudo-posterior distribution as follows

$$\begin{aligned} p(\beta,\lambda,\omega|y,\nu,\alpha) \propto & \prod_{i=1}^{n}\prod_{r=1}^{k}\left[\lambda_{ir}^{-0.5}\exp\left(-\frac{(\boldsymbol{x}_i^T\beta_r - y_{ir}+\lambda_{ir})^2}{2\lambda_{ir}}\right)\right]^{I(c_i\neq r)} \\ & \times\prod_{r=1}^{k}\prod_{j=1}^{p}\omega_{rj}^{-0.5}\exp\left(-\frac{\beta_{rj}^2}{2\nu^2\omega_{rj}\sigma_j^2}\right)p(\omega_{rj}|\alpha) \end{aligned} \tag{2}$$

Define $\theta_{-r} \triangleq \{i : c_i \neq r\}$ as the set of all subjects that do not fall in class r. We could rewrite the complete data pseudo-posterior distribution as

$$\begin{aligned} p(\beta,\lambda,\omega|y,\nu,\alpha) \propto & \prod_{r=1}^{k}\prod_{i\in\Theta_{-r}}\lambda_{ir}^{-0.5}\exp\left(-\frac{(\boldsymbol{x}_i^T\beta_r - y_{ir}+\lambda_{ir})^2}{2\lambda_{ir}}\right) \\ & \times\prod_{r=1}^{k}\prod_{j=1}^{p}\omega_{rj}^{-0.5}\exp\left(-\frac{\beta_{rj}^2}{2\nu^2\omega_{rj}\sigma_j^2}\right)p(\omega_{rj}|\alpha) \end{aligned} \tag{3}$$

The full conditional distribution of β given λ, ω, y

According to equation (10), we can get the full conditional distribution of β_r for any r=1,...,k

$$p(\beta_r|\nu,\lambda_r,\omega_r,y) \propto \prod_{i\in\Theta_{-r}}\prod_{j=1}^{p}\exp\left(-\frac{(\boldsymbol{x}_i^T\beta_r - y_{ir}+\lambda_{ir})^2}{2\lambda_{ir}}\right)\times\exp\left(-\frac{\beta_{rj}^2}{2\nu^2\omega_{rj}\sigma_j^2}\right) \tag{4}$$

Define the matrices $\Lambda_r = diag(\boldsymbol{\lambda}_r), \Omega_r = diag(\omega_r)$, where the diagram elements of Λ_r and Ω_r are the elements of $\boldsymbol{\lambda}_r$ and $\boldsymbol{\omega}_r$, respectively. And $\sigma = diag(\sigma_1^2,...,\sigma_p^2)$. Also let $\boldsymbol{X}_r$ denote a matrix with row i equal to, $\boldsymbol{x}_i^T$, the predictor vector of the i'th subject in Θ_{-r}.

We can write this model in hierarchical form [8]

$$\begin{aligned} \boldsymbol{y}_r - \boldsymbol{\lambda}_r &= \boldsymbol{X}_r\boldsymbol{\beta}_r + \Lambda_r^{\frac{1}{2}}\varepsilon^{\lambda_r} \\ \boldsymbol{\beta}_r &= \frac{1}{\nu}\Omega_r^{\frac{1}{2}}\Sigma^{\frac{1}{2}}\varepsilon^{\beta_r} \end{aligned}$$

where ε^{β_r} and ε^{λ_r} are vectors of iid standard normal deviates with dimensions matching $\boldsymbol{\beta}_r$ and $\boldsymbol{\lambda}_r$.

Thus, for $\boldsymbol{\beta}_r$ has a conditional normal posterior distribution given by

$$p(\beta_r|\nu,\lambda_r,\omega_r,y) \sim \mathcal{N}(b_r, B_r) \tag{5}$$

where

$$B_r^{-1} = \nu^{-2}\sigma^{-1}\omega_r^{-1} + \boldsymbol{X}_r^T\Lambda_r^{-1}\boldsymbol{X}_r \text{ and } b_r = B_r\boldsymbol{X}_r^T(\boldsymbol{y}_r \times \boldsymbol{\lambda}_r^{-1} - \mathbf{1}) \tag{6}$$

The full conditional distribution for λ_{ir} and ω_{rj} given β, ν, y We want the conditional distribution of λ_{ir} for r=1,..,k and $i \in \Theta_{-r}$. Note that from the complete pseudo-posterior distribution we can get

$$\begin{aligned} p(\lambda_{ir}|\beta_r,y_{ir}) &\propto \frac{1}{\sqrt{2\pi\lambda_{ir}}}\exp\left\{-\frac{1}{2}\left(\frac{(\boldsymbol{x}_i^T\beta_r - y_{ir})^2}{\lambda_{ir}} + \lambda_{ir}\right)\right\} \\ &\sim \mathcal{GIG}\left\{\frac{1}{2}, 1, (\boldsymbol{x}_i^T\beta_r - y_{ir})^2\right\} \end{aligned} \tag{12}$$

This implies that

$$(\lambda_{ir}^{-1}|\beta_r,y_{ir}) \sim \mathcal{IG}(|\boldsymbol{x}_i^T\beta_r - y_{ir}|^{-1}, 1) \tag{7}$$

For the full conditional distribution of ω_{rj}, we know that it is proportional to the integrand in equation (9). This is complicated because its prior density $p(\omega_{rj}|\alpha)$ is generally unavailable. However, for the two special cases of $\alpha = 1$ and

$\alpha = 2$, the closed-form solutions are available. When $\alpha = 2$, $p(\omega_{rj}|\beta_{rj})$ is a point mass at 1. For $\alpha = 1$, the full conditional distribution of ω_{rj} is

$$p(\omega_{rj}|\beta_{rj}, v) \propto \frac{1}{\sqrt{2\pi\omega_{rj}}} \exp\left\{-\frac{1}{2}\left(\frac{\beta_{rj}^2/v^2\sigma_j^2}{\omega_{rj}} + \omega_{rj}\right)\right\}$$
$$\sim \mathscr{GIG}\left(\frac{1}{2}, 1, \frac{\beta_{rj}^2}{v^2\sigma_j^2})\right)$$

Similarly, we know that

$$(\omega_{rj}^{-1}|\beta_{rj}, v) \sim \mathscr{IG}(v\sigma_j/|\beta_{rj}|, 1) \tag{8}$$

Later we will use these distributions to develop learning algorithms.

3 Point estimation by EM and other related algorithms

In this section, we use the distributions obtained in Section Two to construct EM-style algorithms to estimate the coefficients. First, we will develop an EM algorithm for learning β with a fixed value of the tuning parameter v. Then we develop an ECME algorithm to learn β and v simultaneously.

3.1 Learning β with fixed v

With the augmented data λ and ω, the EM algorithm is an iterative method for finding posterior modes or MLEs. From equation (12), we know that the posterior distributions of the β_r's for r=1,...,k are independent. Thus, we can estimate them separately using the EM algorithm. For β_r, the E-step and M-step are defined by

$$\begin{aligned} &\text{E-step } Q(\boldsymbol{\beta}_r|\boldsymbol{\beta}_r^{(g)}) = \int \log p(\boldsymbol{\beta}_r|v, \boldsymbol{\lambda}_r, \boldsymbol{\omega}_r, y) p(\boldsymbol{\lambda}_r, \boldsymbol{\omega}_r|\boldsymbol{\beta}_r^{(g)}, v, y) d\boldsymbol{\lambda}_r d\boldsymbol{\omega}_r \\ &\text{M-step} \qquad \boldsymbol{\beta}_r^{(g+1)} = \arg\max_{\boldsymbol{\beta}_r} Q(\boldsymbol{\beta}_r|\boldsymbol{\beta}_r^{(g)}) \end{aligned} \tag{9}$$

Note that any term in $\log p(\boldsymbol{\beta}_r|v, \boldsymbol{\lambda}_r, \boldsymbol{\omega}_r, y)$ that is free of $\boldsymbol{\beta}_r$ can be absorbed to the constant. This leaves us only the linear function of λ_{ir} and ω_{rj}. Thus, we only need to replace them with their conditional expectations $\hat{\lambda}_{ir}^{-1(g)}$ and $\hat{\omega}_{rj}^{-1(g)}$ for the calculation of function $Q(\boldsymbol{\beta}_r|\boldsymbol{\beta}_r^{(g)})$, given $\boldsymbol{\beta}_r$ and the observed data.

As discussed before, the result for ω_{rj} would depend on the value of α. Here, we still focus on the case where $\alpha = 1$. According to equation(16), we can obtain that

$$\omega_{rj}^{-1(g)} = v\sigma_j|\beta_{rj}|^{-1}$$

Recall that the conditional posterior of β_R follows a multivariate normal distribution. [9] Thus, the posterior mode will be the same as the posterior mean. Using equations (13) and (14), we can get the following algorithm:

Algorithm: EM-SVM

Repeat the following until convergence

E-Step: Given a current estimate $\boldsymbol{\beta}_r = \boldsymbol{\beta}_r^{(g)}$,compute

$$\hat{\lambda}_{ir}^{-1(g)} = |\boldsymbol{x}_i^T\boldsymbol{\beta}_r - y_{ir}|^{-1},$$
$$\hat{\Lambda}_r^{-1(g)} = diag\left(\hat{\boldsymbol{\lambda}}_r^{-1(g)}\right),$$
$$\hat{\Omega}_r^{-1(g)} = diag\left(\hat{\boldsymbol{\omega}}_r^{-1(g)}\right),$$

M-Step: Compute $\boldsymbol{\beta}_r^{(g+1)}$ as

$$\boldsymbol{\beta}_r^{(g+1)} = \left(v^{-2}\Sigma^{-1}\hat{\Omega}_r^{-1(g)} + \boldsymbol{X}_r^T\hat{\Lambda}_r^{-1(g)}\boldsymbol{X}_r\right)^{-1}\boldsymbol{X}_r^T(\boldsymbol{y}_r \times \hat{\boldsymbol{\lambda}}_r^{-1(g)} - \mathbf{1})$$

3.2 Stability

The EM algorithm gets unstable when some elements of $\boldsymbol{\lambda}^{-1}$ or $\boldsymbol{w}^{-1}$ equals ∞. However, they provide the following ways to restore the unstable problem.

When $w_j^{-1} = \infty$, then $\beta_j = 0$, we may simply omit column j from $\mathbf{X}$ and β_j from $\boldsymbol{\beta}$.

When $\lambda_i^{-1} = \infty$, then observation i is a support vector satisfying the constraint $y_i \boldsymbol{\beta}^T \mathbf{x}_i = 1$. Following Lee and Cui [10], the numerical instability can be solved by separating the support vectors from the rest of the data as follows:

–Let $\mathbf{X}_s$ denote the matrix obtained by stacking the linearly independent support vector row wise, i.e., each row of $\mathbf{X}_s$ is a support vector.
–Let $\mathbf{X}_{-s}$ denote $\mathbf{X}$ with the support vector rows deleted.
–Let $\boldsymbol{\lambda}_{-s}^{-1}$ denote the finite elements of $\boldsymbol{\lambda}$, and let $\boldsymbol{\Lambda}_{-s}^{-1} = diag(\boldsymbol{\lambda}_{-s}^{-1})$.
–Then a stable version of the M-step can be given by the following "restricted least squares",

$$\begin{aligned}\begin{pmatrix}\boldsymbol{\beta}\\ \boldsymbol{\psi}\end{pmatrix} &= \begin{pmatrix}\mathbf{B}_{-s} & \mathbf{X}_s^T\\ \mathbf{X}_s & \mathbf{0}\end{pmatrix}^{-1}\begin{pmatrix}\mathbf{X}_{-s}^T(\mathbf{1}+\boldsymbol{\lambda}_{-s}^{-1})\\ \mathbf{1}\end{pmatrix}\\ &= \begin{pmatrix}\mathbf{B}_{-s}(\mathbf{I}+\mathbf{X}_{-s}^T\mathbf{F}\mathbf{X}_s\mathbf{B}_{-s}) & -\mathbf{B}_{-s}\mathbf{X}_s^T\mathbf{F}\\ -\mathbf{F}\mathbf{X}_s\mathbf{B}_{-s} & \mathbf{F}\end{pmatrix}\begin{pmatrix}\mathbf{X}_{-s}^T(\mathbf{1}+\boldsymbol{\lambda}_{-s}^{-1})\\ \mathbf{1}\end{pmatrix}\end{aligned} \tag{10}$$

where $\mathbf{B}_{-s} = \nu^{-2}\boldsymbol{\Sigma}^{-1}\boldsymbol{\Omega}^{-1} + \mathbf{X}_{-s}^T\boldsymbol{\Lambda}_{-s}^{-1}\mathbf{X}_{-s}$ and $\mathbf{F} = -(\mathbf{X}_s\mathbf{B}_{-s}\mathbf{X}_s^T)^{-1}$. $\boldsymbol{\psi}$ is a vector of Lagrange multipliers.

3.3 Learning β and ν simultaneously

In order to learn β and γ together, we use the generalized expectation-conditional maximization algorithm(ECME), where the last"E" represents the conditional maximization of either function. To implement the ECME algorithm, we assume an inverse gamma prior distribution for ν^{α} [12]

$$p(\nu^{-\alpha}) \propto (\nu^{-\alpha})^{\alpha_\nu - 1}\exp(-b_\nu \nu^{-\alpha})$$

Combining this prior with equation (8), we can find the conditional posterior density of ν given β and α

$$p(\nu^{-\alpha}|\beta,\alpha) \propto (\nu^{-\alpha})^{\frac{pk}{\alpha}+\alpha_\nu-1}\exp\left\{-\nu^{-\alpha}\left[b_\nu + \sum_{r=1}^{k}\sum_{k=1}^{p}\left|\frac{\beta_{rj}}{\sigma_j}\right|^{\alpha}\right]\right\}$$

The following algorithm can be obtained with minor modification of the EM-SVM algorithm.

Algorithm:ECME-SVM

E-StepIdentical to the E-step of EM-SVM with $\nu = \nu^{(g)}$.
CM-stepIdentical to the M-step of EM-SVM with $\nu = \nu^{(g)}$.
CME-StepSet

$$\left(\nu^{\alpha}\right)^{(g+1)} = \frac{b_\nu + \sum_{r=1}^{k}\sum_{k=1}^{p}|\beta_{rj}^{(g)}/\sigma_j|^{\alpha}}{pk/\alpha + a_\nu - 1}$$

4 Fully Bayesian Multicategory Support Vector Machines

In the MSVM framework of Lee et al. [11], following Zhang and Jordan [12], we can find $\mathbf{f}(.)$ by minimizing the following penalized function when $\alpha = 1$,

$$d(\boldsymbol{\beta},\nu) = \sum_{i=1}^{n}\mathbf{L}(\mathbf{y}_i)\cdot(\mathbf{f}(\mathbf{x}_i)-\mathbf{y}_i)_+ + \nu^{-1}\sum_{r=1}^{k}\sum_{j=1}^{p}\left|\frac{\beta_{rj}}{\sigma_j}\right| \tag{11}$$

or equivalently,

$$d(\boldsymbol{\beta},\nu) = \sum_{r=1}^{k}\sum_{i\in\ominus_{-r}}(f_r(\mathbf{x}_i)+\frac{1}{k-1})_+ + \nu^{-1}\sum_{r=1}^{k}\sum_{j=1}^{p}\left|\frac{\beta_{rj}}{\sigma_j}\right| \tag{12}$$

with constraints $\sum_{r=1}^{k} f_r(\mathbf{x}_i) = 0$ for $i = 1,\cdots,n$.

where σ_j is the standard deviation of the j'element of $\mathbf{x}$, ν is a tuning parameter and $\ominus_{-r} = \{i : c_i \neq r\}$, c_i is the classification number of observation i.

4.1 Bayesian inference

The minimization problem (12) can be viewed to find the mode of pseudo-posterior distribution from the Bayesian perspective. That is

$$p(\boldsymbol{\beta}|\nu,\mathbf{y}) \propto \exp(-d(\boldsymbol{\beta},\nu)) \propto C(\nu)L(\mathbf{y}|\boldsymbol{\beta})p(\boldsymbol{\beta}|\nu) \tag{13}$$

where $C(\nu)$ is a normalization constant. According to the form of the objective function, we can adopt the following likelihood function for the data and assume an exponential power prior for $\boldsymbol{\beta}$ as follows:

$$L(\mathbf{y}|\boldsymbol{\beta}) = \prod_{i=1}^{n} L_i(y_i|\boldsymbol{\beta}) = \exp\left\{-2\sum_{i=1}^{n}\mathbf{L}(\mathbf{y}_i)\cdot(\mathbf{f}(\mathbf{x}_i)-\mathbf{y}_i)_+\right\} \tag{14}$$

$$p(\boldsymbol{\beta}|\nu) = \prod_{r=1}^{k}\prod_{j=1}^{p} p(\beta_{rj}|\nu) = \left(\prod_{r=1}^{k}\prod_{j=1}^{p}\frac{1}{2\nu\sigma_j}\right)\exp\left(-\sum_{r=1}^{k}\sum_{j=1}^{p}\frac{|\beta_{rj}|}{\nu\sigma_j}\right) \tag{15}$$

where $[\beta_{rj}|\nu]$ follows the Laplace distribution.

Now,following Polson & Scott [1], we assume a gamma prior on ν^{-1}, i.e.

$$p(\nu^{-1}) \propto (\nu^{-1})^{a_\nu-1}\exp(-b_\nu\nu^{-1}) \tag{16}$$

with hyper-parameters (a_ν, b_ν). Then we use the independent Jeffreys noninformative prior, i.e. the invariance prior, on σ_j,

$$p(\sigma_j) \propto \frac{1}{\sigma_j} \tag{17}$$

for $j = 1,\cdots,p$.

Theorem 1. *Under the penalized function (12) as well as the priors (16) and (17), following the data augmentation approach proposed by Polson and Scott [1], we have the following full conditional posterior distributions*

$$[\boldsymbol{\beta}_r|\nu,\boldsymbol{\lambda}_r,\boldsymbol{w}_r,\mathbf{y}] \sim \mathcal{N}(\mathbf{b}_r,\mathbf{B}_r) \tag{18}$$

$$[\lambda_{ir}^{-1}|\boldsymbol{\beta}_r,y_{ir}] \sim \mathcal{IG}(|\mathbf{x}_i^T\boldsymbol{\beta}_r - y_{ir}|^{-1},1), \tag{19}$$

$$[w_{rj}^{-1}|\beta_{rj},\nu,\sigma_j] \sim \mathcal{IG}(\nu\sigma_j/|\beta_{rj}|,1) \tag{20}$$

$$[\nu^{-1}|\boldsymbol{\beta},\sigma_j] \sim Gamma(pk+a_\nu-1, b_\nu+\sum_{r=1}^{k}\sum_{j=1}^{p}\frac{|\beta_{rj}|}{\sigma_j}) \tag{21}$$

$$[\sigma_j|\nu,\boldsymbol{\beta}] \sim Inv.\ Gamma(k,\frac{1}{\nu}\sum_{r=1}^{k}|\beta_{rj}|) \tag{22}$$

for $i \in \ominus_{-r}; r = 1,\cdots,k$ and $j = 1,\cdots,p$. Where $\mathbf{B}_r^{-1} = \nu^{-2}\boldsymbol{\Sigma}^{-1}\boldsymbol{\Omega}_r^{-1} + \mathbf{X}_r^T\boldsymbol{\Lambda}_r^{-1}\mathbf{X}_r$ and $\mathbf{b}_r = \mathbf{B}_r\mathbf{X}_r^T\boldsymbol{\Lambda}_r^{-1}(\mathbf{y}_r - \boldsymbol{\lambda}_r)$. And $\mathbf{y}_r = \{y_{ir}\}_{i\in\ominus_{-r}}$, $\boldsymbol{\lambda}_r = \{\lambda_{ir}\}_{i\in\ominus_{-r}}$, $\boldsymbol{\Lambda}_r = diag(\boldsymbol{\lambda}_r)$, $\boldsymbol{\Omega}_r = diag(\{w_{rj}\}_{j=1}^{p})$, $\boldsymbol{\Sigma} = diag(\{\sigma_j^2\}_{j=1}^{p})$, $\mathbf{1}$ is the vector of 1's. $\mathbf{X}_r$ is a matrix with row i is $x_i, i \in \ominus_{-r}$.

Then we can develop the MCMC algorithm from Theorem 1.

Algorithm: MCMC-MSVM

Step 1Draw $\boldsymbol{\beta}_r^{(g+1)}$ from $\mathcal{N}(\mathbf{b}_r^{(g)},\mathbf{B}_r^{(g)})$ for $r = 1,\cdots,k$;
Step 2Draw $\lambda_{ir}^{-1(g+1)}$ from $\mathcal{IG}(|\mathbf{x}_i^T\boldsymbol{\beta}_r^{(g+1)} - y_{ir}|^{-1},1)$ independently, for $r = 1,\cdots,k; i \in \ominus_{-r}$;
Step 3Draw $w_{rj}^{-1(g+1)}$ from $\mathcal{IG}(\nu^{(g)}\sigma_j^{(g)}/|\beta_{rj}^{(g+1)}|,1)$ independently, for $r = 1,\cdots,k$ and $j = 1,\cdots,p$;
Step 4Draw $\nu^{-1(g+1)}$ from Gamma$(pk+a_\nu-1, b_\nu+\sum_{r=1}^{k}\sum_{j=1}^{p}\frac{|\beta_{rj}^{(g+1)}|}{\sigma_j^{(g)}})$;
Step 5Draw $\sigma_j^{(g+1)}$ from Inv. Gamma$(k,\frac{1}{\nu^{(g+1)}}\sum_{r=1}^{k}|\beta_{rj}^{(g+1)}|)$ for $j = 1,\cdots,p$.

4.2 Simulation Example

We consider a simple three class example on the unit interval [0, 1] with $p_1(x) = 0.97exp(-3x)$, $p_3(x) = exp(-2.5(x-1.2)^2)$, and $p_2(x) = 1 - p_1(x) - p_3(x)$. Class 1 is most likely for small x, whereas class 3 is most likely for large x. The in-between interval is competing zone for three classes, although class 2 is slight dominant. In simulation, 60 data points are generated and considered as training data. Figure 1 shows the classification of training data, y-axis is another variable with uniform (0,1) distribution associated with X.

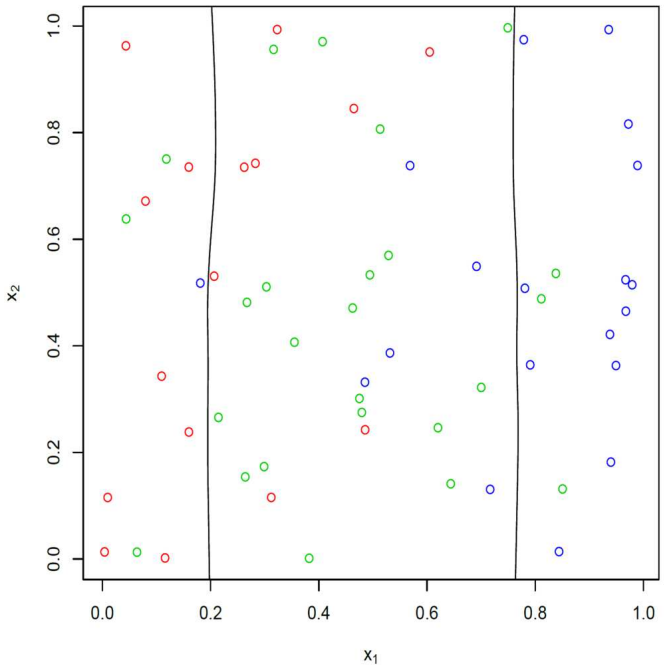


Fig. 1: Classification Result of Simulated Training Data Points with 3 Classes (Red, Green and Blue)

We generate 100 data points and classify through MSVM. In this example, error rate is 0.48, which is not relative large to our expectation. Figure 2 shows the predicted probability function for MSVM. All estimation and plots are obtained from "MSVMPATH" in R.

5 Application

We applied our multi-category classification method to MIT genome data. The data is available at http://www.genome.wi.mit.edu. The data is the gene expression levels of two different types of acute Leukemia: acute lymphoblastic leukemia (ALL) and acute myeloid leukemia (AML). The ALL can be further divided into two different classes: B-cell ALL and T-cell ALL. There are 56 cases of Leukemia in total, 32 cases of B-cell ALL, 12 cases of T-cell ALL and 28 cases of AML. In each case, the gene expression levels of p = 8,468 human genes are measured using Asymmetric high-density oligonucleotide arrays.

Two different types of genes are ruled out. The first type is the genes with too large or too small gene expression level. The second type is the genes varied too much over cases. After these two types of genes are ruled out, 4137 genes are left. Then, a base 10 logarithmic transformations were taken. But p = 4137 genes are too many to compute, since we have to calculate the inverse of a p by p matrix. Hence we select 40 genes with the largest BSS/WSS ratios. BSS and WSS denote between-group sum of square and within-group sum of square respectively. In our method, there is an intercept term in the parameter. Consequently, there are p = 59 dimensions in each of the 3 groups.

We first applied EM-SVM algorithm with fixed ν to the data. Figure 3 is the iteration histories of βs. It involves three plots, each of them shows the βs in a group and there are 41 βs in each group. We did 500 iterations. Moreover, it reveals that the iteration is stable before 400 iterations, but it is very unstable when the iteration is near 500 times. The 41 βs in

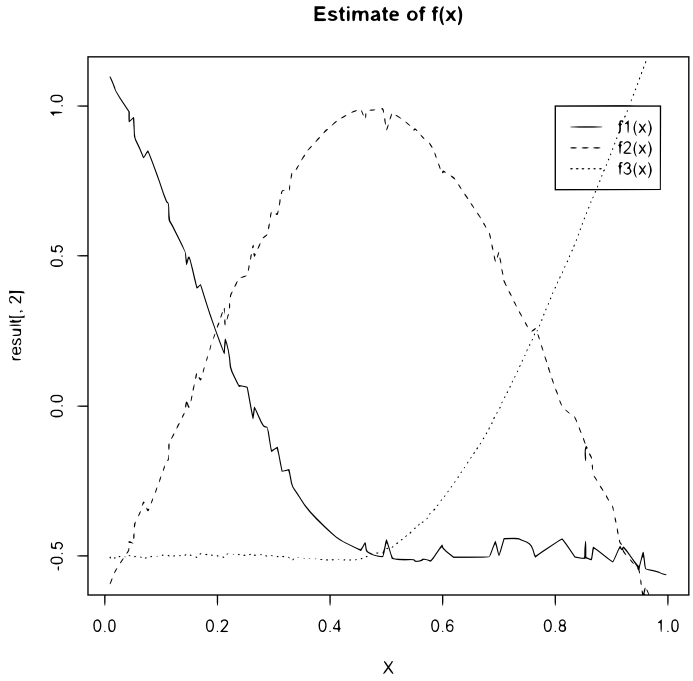


Fig. 2: Estimate of the predicted probability function f(x) for MSVM

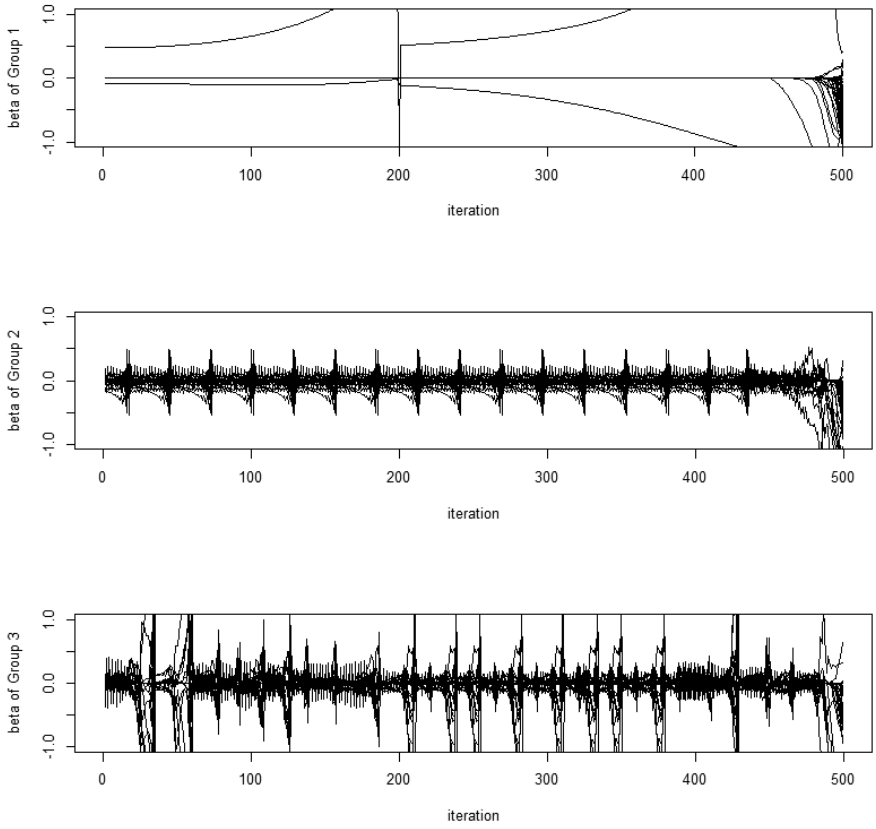


Fig. 3: EM-SVM Iteration Path of β's

group 1 are almost zero, except 3 of them. In group 2, the βs vary in a small range. But in group 3, the βs fluctuate a lot. Figure 4 is the sample variances of every β in each group. In group 1, only the variances of three βs are obviously larger than the other βs, which is consistent with the iteration plot in figure 1. The variances of βs in group 2 is relatively small compared with those in group 3.

We also applied ECME-SVM to the data. The results are presented in figure 3. It illustrates that after 300 iterations, the iteration becomes very unstable. The iteration ends after 350 times because of NA is produced. With v estimated simultaneously, the βs in group 2 and 3 fluctuate much less, but βs in group 1 fluctuate more.

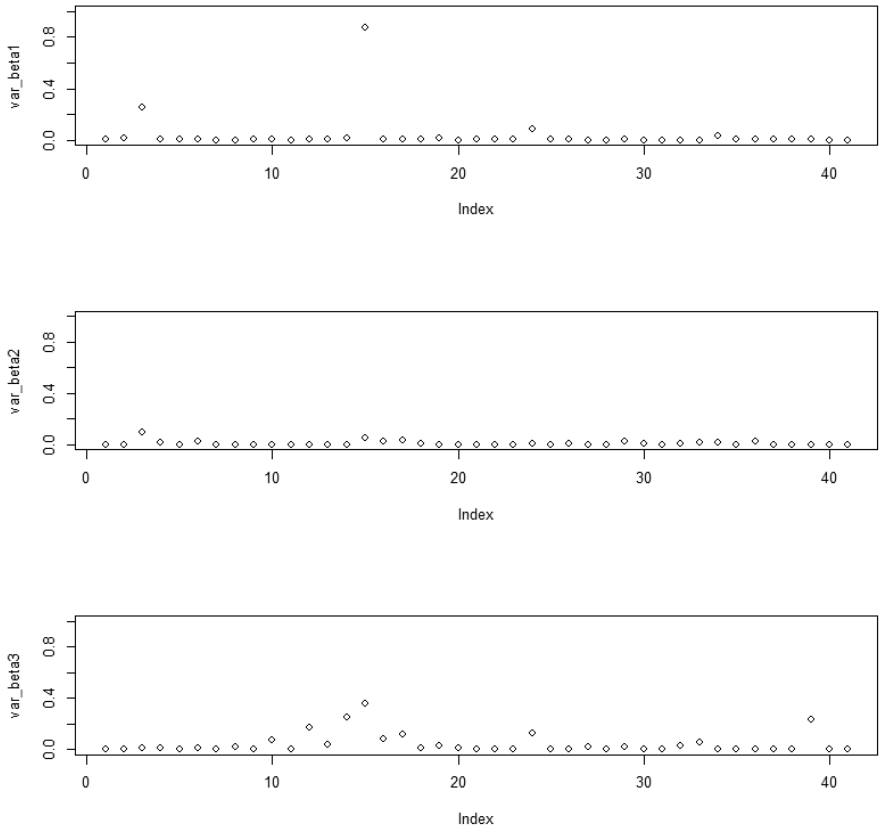


Fig. 4: Variances of β's in EM-SVM

6 Discussion

In the MSVM framework, we need to use the truncated multivariate normal distribution to satisfy the sum to zero constraint on $\mathbf{f}(.)$,i.e. $\sum_{r=1}^{k} f_r(\mathbf{x}_i) = 0$ for $i = 1, \cdots, n$, but it sacrifices the efficiency of the computation. In our ca suppose

$$f_r(\mathbf{x}_i) = \mathbf{x}_i \boldsymbol{\beta}_r \tag{23}$$

If we let

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \vdots \\ \mathbf{x}_n^T \end{pmatrix}, \mathbf{D} = \begin{pmatrix} \beta_{11} & \beta_{21} & \cdots & \beta_{k1} \\ \beta_{12} & \beta_{21} & \cdots & \beta_{k2} \\ \vdots & \vdots & \cdots & \vdots \\ \beta_{1p} & \beta_{21} & \cdots & \beta_{kp} \end{pmatrix} \tag{24}$$

,the sum-to-zero constraint is equivalent to $\mathbf{X}\sum_{r=1}^{k} \boldsymbol{\beta}_r = \mathbf{0}_n$. If the design matrix $\mathbf{X}$ is of full rank, $\sum_{r=1}^{k} \boldsymbol{\beta}_r = \mathbf{0}_p$ or $\mathbf{D}\mathbf{1}_k = \mathbf{0}_p$ can guarantee the constraint.

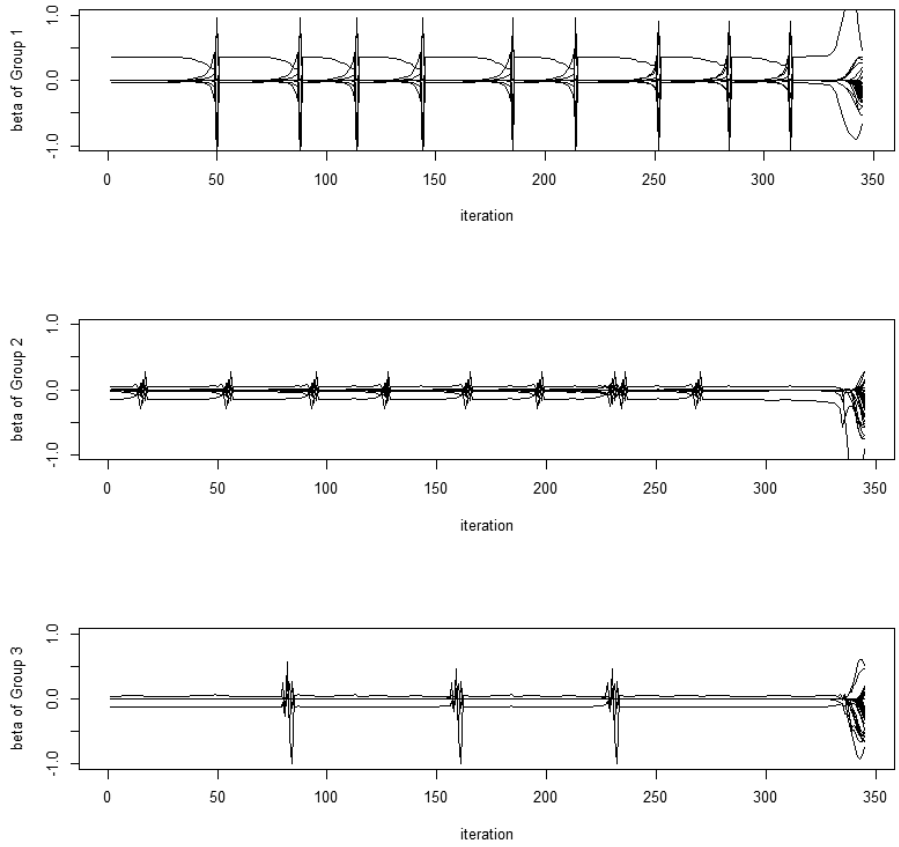


Fig. 5: ECME-SVM Iteration Path of β's

Following Zhang and Jordan [11], one possible solution could be taken the reparameterization procedure below

$$\mathbf{D} = \mathbf{BH} \tag{25}$$

where $\mathbf{H} = \mathbf{I}_k - \frac{1}{k}\mathbf{1}_k\mathbf{1}_k^T$. However, since the matrix $\mathbf{H}$ is singular, it is impossible to get the density distribution for the parameters $\mathbf{B}$ from the distribution of $\mathbf{D}$ using the density transformation formula. One possible solution for this issue could be solved by looking for a kernel function corresponding to the L_1-normal penalty in (11).

7 Conclusion

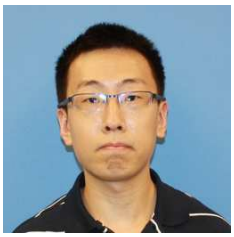
In this paper, we extended the multi-class support vector machine under the Bayesian framework, which can be used to solve multivariate classification problem. To minimize the loss function used here, we first developed the pseudo posterior density for the model coefficients. To maximize the the pseudo posterior density is the same as minimizing the loss function. We have developed an EM algorithm for locating point estimates of multivariate support vector machine and an MCMC algorithm for exploring the full pseudo-posterior distribution. Moreover, we developed the posterior predictive probabilities to classify for the observed data.

Acknowledgement

The authors are grateful to the anonymous referee for their beneficial and accurate comments that improved this paper.

References

[1] N. G. Polson and S. L. Scott, Data augmentation for support vector machines, *Bayesian Analysis*, **6**, 1-24 (2011).
[2] C. Cortes, V. Vapnik, Support-vector networks, *Machine Learning*, **20**, 273–297 (1995).
[3] R. Henao, X. Yuan and L. Carin, Bayesian nonlinear support vector machines and discriminative factor modeling. *Neural Information Processing Systems Conference* (2014).
[4] M. D. Hoffman, D. M. Blei, C. Wang and J. Paisley, Stochastic variational inference. *Journal of Machine Learning Research*, **14**, 1303-1347 (2013).
[5] J. Hensman, N. Fusi, N. D. Lawrence, Gaussian processes for big data. *Conference on Uncertainty in Artificial Intellegence*, 282-290 (2013).
[6] M. Fernández-Delgado, E. Cernadas, S. Barro, D. Amorim. Do we need hundreds of classifiers to solve real world classification problems? *Journal of Machine Learning Research*, **15**, 3133-3181 (2014).
[7] M. Mohri, A. Rostamizadeh, A. Talwalkar, *Support vector machines*, in Foundations of Machine Learning, 2nd ed., MIT press, 79-91 (2012).
[8] D. F. Andrews and C. L. Mallows, Scale mixtures of normal distributions, *Journal of the Royal Statistical Society. Series B*, **36**, 99-102 (1974).
[9] M. West, On scale mixtures of normal distributions, *Biometrika*, **74**, 646-648 (1987).
[10] Y. Lee and Z. Cui, Characterizing the solution path of multicategory support vector machines, *Statistica Sinica* , **16**, 391-409 (2006).
[11] Z. Zhang and M. I. Jordan, Bayesian multicategory support vector machines, *Twenty-Second Conference Annual Conference on Uncertainty in Artificial Intelligence*, 552-559 (2006).
[12] Y. Lee, Y. Lin and G. Wahba, Multicategory support vector machines theory and application to the classification of microarray data and satellite radiance data, *Journal of American Statistical Association*, **99**, 67-81 (2004).



Yeqian Liu received the PhD degree in Statistics at the University of Missouri. His research interest mainly lies in Biostatistics, in particular, survival analysis, interval-censored data analysis, semiparametric and nonparametric statistical methods. He has developed quantile and hazard based regression methods for survival data in the presence of dependent censoring, cured subgroups and measurement errors with applications to biomedical studies. He has published research articles in reputed international journals of statistics.