Multi-class cancer classification using multinomial probit regression with Bayesian gene selection

X. Zhou, X. Wang and E.R. Dougherty

Abstract: We consider the problems of multi-class cancer classification from gene expression data. After discussing the multinomial probit regression model with Bayesian gene selection, we propose two Bayesian gene selection schemes: one employs different strongest genes for different probit regressions; the other employs the same strongest genes for all regressions. Some fast implementation issues for Bayesian gene selection are discussed, including preselection of the strongest genes and recursive computation of the estimation errors using QR decomposition. The proposed gene selection techniques are applied to analyse real breast cancer data, small round blue-cell tumours, the national cancer institute's anti-cancer drug-screen data and acute leukaemia data. Compared with existing multi-class cancer classifications, our proposed methods can find which genes are the most important genes affecting which kind of cancer. Also, the strongest genes selected using our methods are consistent with the biological significance. The recognition accuracies are very high using our proposed methods.

1 Introduction

Through comparison of gene expression in normal and disease cells, microarrays can be used to identify disease genes and targets for therapeutic drugs. Therefore the huge volume of data provided by cDNA microarray measurements must be explored so that fundamental questions about gene functions and their inter-dependence can be answered and, hopefully, so that answers to questions such as which type of disease is affecting the cells or which genes have a very strong influence on this disease can be answered. Questions such as these lead to the study of gene classification problems.

For multi-class cancer classification and discovery, the performance of different discrimination methods, including nearest-neighbour classifiers, linear discriminant analysis, classification trees and bagging and boosting learning methods, are compared in [1]. Moreover, this problem has been studied using partial least squares [2] and using iterative classification trees [3]. However, in all of the above methods, gene selection and classification are treated as two separate steps, which may limit their performance.

Given the thousands of genes and the small number of data samples, gene selection becomes a very important issue. In the past decade, a number of variable (or gene) selection methods have been proposed: the support vector machine method [4], the perceptron method [5], Bayesian variable selection [6, 7], the minimum description length principle for model selection [8] and the vote technique [9]. Although [7] discussed gene selection for two-class classification using original expression data, to the best of our knowledge, there is no work treating multi-class cancer classification using multinomial probit regression with Bayesian variable selection from gene expression data.

In this paper, we will focus on Bayesian variable selection using gene expression data. We use a multinomial regression model (probit regressor) with data augmentation to turn the multinomial problem into a sequence of smoothing problems. To cope with gene selection for the multinomial probit model, we propose two methods: one is to select different important genes for different regression equations, and the other is to select the same strongest genes for all regression equations. The probit regressor is approximated as a linear combination of the genes. A Gibbs sampler is employed to find the strongest genes. As these methods have very high computational complexity, we also discuss some numerical techniques to speed up the computation. Furthermore, a gene preselection procedure is adopted to reduce the huge number of genes being considered for selection. After finding the strongest genes, we predict the test samples based on the strongest genes using the estimated probit regressors. Compared with the methods in [1–3], our proposed methods can find which genes are the most important genes affecting which cancer.

2 Multinomial probit regression with Bayesian variable selection

2.1 Problem formulation

Assume there are K classes of cancer. Let $\boldsymbol{w} = [w_1, \ldots, w_m]^T$ denote the class labels, where $w_i = k$ indicates the sample i being cancer k, where $k = 1, \ldots, K$. Assume $x_1, \ldots, x_n$ are the n genes. Let x_{ij} be the measurement of the expression level of the jth gene for the ith sample, where $j = 1, 2, \ldots, n$. Let $\boldsymbol{X} = (x_{ij})_{m,n}$ denote the

IEE Proceedings online no. 20050015

doi:10.1049/ip-syb:20050015

Paper first received 16th March and in revised form 27th August 2005

X. Zhou is with the Department of Electrical Engineering, Texas A&M University, College Station, TX 77843, USA

X. Wang is with the Department of Electrical Engineering, Columbia University, New York, NY 10027, USA

E.R. Dougherty is with the Department of Electrical Engineering, Texas A&M University, College Station, TX 77843, USA, and Department of Pathology, University of Texas M.D. Anderson Cancer Center, Houston, TX 77030, USA

E-mail: wangx@ee.columbia.edu

expression levels of all genes, i.e.

$$\boldsymbol{X} = \left[\begin{array}{cccc} \text{gene 1} & \text{gene 2} & \cdots & \text{gene } n \\ x_{11} & x_{12} & \cdots & x_{1n} \\ x_{11} & x_{12} & \cdots & x_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{array}\right] \tag{1}$$

Let $\boldsymbol{X}_i$ denote the ith row of matrix $\boldsymbol{X}$. In the binomial probit regression, i.e. $K = 2$, the relationship between w_i and the gene expression levels $\boldsymbol{X}_i$ is modelled using a probit regression model [10], which yields

$$P(w_i = 1|\boldsymbol{X}_i) = \Phi(\boldsymbol{X}_i\beta) \quad i = 1, \ldots, m \tag{2}$$

where $\beta = (\beta_1, \beta_2, \ldots, \beta_n)^T$ is the vector of regression parameters, and Φ is the standard normal cumulative distribution function. Introduce m independent latent variables $z_1, \ldots, z_m$, where $z_i \sim \mathcal{N}(\boldsymbol{X}_i\beta, 1)$, i.e.

$$z_i = \boldsymbol{X}_i\beta + e_i \quad i = 1, \ldots, m \tag{3}$$

and $e_i \sim \mathcal{N}(0, 1)$. Define γ as the $n \times 1$ indicator vector with the jth element γ_j, such that $\gamma_j = 0$ if $\beta_j = 0$ (the variable is not selected), and $\gamma_j = 1$ if $\beta_j \neq 0$ (the variable is selected). The Bayesian variable selection is to estimate γ from the posterior distribution $p(\gamma \mid z)$. See [11] for details.

However, when $K > 2$, the situation is different from the binomial case, because we have to construct $K - 1$ regression equations similar to (3). Introduce $K - 1$ latent variables $y_1, \ldots, y_{K-1}$ and $K - 1$ regression equations such that $y_k = \boldsymbol{X}\beta_k + e_k$, $k = 1, \ldots, K-1$, where $e_k \sim \mathcal{N}(0, 1)$. Let y_k take m values $\{y_{k,1}, \ldots, y_{k,m}\}$ for each equation. In matrix form

$$\left\{\begin{array}{ccc} y_{1,i} & = & \boldsymbol{X}_i\beta_1 + e_{1,i} \\ \vdots & \vdots & \vdots \\ y_{K-1,i} & = & \boldsymbol{X}_i\beta_{K-1} + e_{K-1,i} \end{array}\right. \quad i = 1, \ldots, m \tag{4}$$

Denote $\boldsymbol{y}_k \triangleq [y_{k,1}, \ldots, y_{k,m}]^T$ and $\boldsymbol{e}_k \triangleq [e_{k,1}, \ldots, e_{k,m}]^T$. Then (4) can be rewritten as

$$\boldsymbol{y}_k = \boldsymbol{X}\beta_k + \boldsymbol{e}_k \quad k = 1, \ldots, K-1 \tag{5}$$

This model is called the multinomial probit model. For background on multinomial probit models, see [12]. However, this multinomial model is a little different from the standard model in [12]: in a standard multinomial model, $\boldsymbol{e}_k$ follows Gaussian distribution with the mean being zeros and the covariance matrix being, say, Σ, which follows an inverse Wishart distribution, and, in our model, Σ is set as an identity matrix according to the assumption in the standard binary probit model [10]. Note that the multinomial probit model used here was also studied in another paper [13]. As a simple approximated calculation, the authors [13] assumed Σ is the identity matrix and had a successful application. Hence, in this paper, we adopted this approximated calculation. Note that, in this model, we do not have the observations of $\{\boldsymbol{y}_k\}_{k=1}^{K-1}$, which makes it difficult to estimate the parameters in (5).

We consider two schemes for gene selection: one selects different strongest genes for each equation in (5); the other selects the same strongest genes for all equations in (5). We will discuss the first case in the following section and the second case in Section 4. Given γ_k, let β_{γ_k} consist of all non-zero elements of β and let $\boldsymbol{X}_{\gamma_k}$ be the columns of $\boldsymbol{X}$ corresponding to those of γ that are equal to 1 for equation k. Then (5) is rewritten as

$$\boldsymbol{y}_k = \boldsymbol{X}_{\gamma_k}\beta_{\gamma_k} + \boldsymbol{e}_k \quad k = 1, \ldots, K-1 \tag{6}$$

Now the problem is how to estimate γ_k and the corresponding β_k and $\boldsymbol{y}_k$ for each equation in (6).

2.2 Bayesian variable selection

A Gibbs sampler is employed to estimate all the parameters. Given γ_k for equation k, the prior distribution of β_{γ_k} is $\beta_{\gamma_k} \sim \mathcal{N}(0, c(\boldsymbol{X}_{\gamma_k}^T\boldsymbol{X}_{\gamma_k})^{-1})$ [11], where c is a constant (we set $c = 100$ in this study). The detailed derivation of the posterior distributions of the parameters are the same as that in [7, 11]; here, we summarise the procedure for Bayesian variable selection. Denote

$$S(\gamma_k, \boldsymbol{y}_k) = \boldsymbol{y}_k^T\boldsymbol{y}_k - \frac{c}{c+1}\boldsymbol{y}_k^T\boldsymbol{X}_{\gamma_k}(\boldsymbol{X}_{\gamma_k}^T\boldsymbol{X}_{\gamma_k})^{-1}\boldsymbol{X}_{\gamma_k}^T\boldsymbol{y}_k \qquad k = 1, \ldots, K-1 \tag{7}$$

Then the Gibbs sampling algorithm for estimating $\{\gamma_k, \beta_k, \boldsymbol{y}_k\}$ is as follows:

Step 1: Draw γ_k from $p(\gamma_k \mid \boldsymbol{y}_k)$, where

$$p(\gamma_k|\boldsymbol{y}_k) \propto (1+c)^{(-n_{\gamma_k}/2)} \exp\left[-\frac{1}{2}S(\gamma_k, \boldsymbol{y}_k)\right] \times \prod_{j=1}^{n} \pi_j^{\gamma_{k,j}}(1-\pi_j)^{1-\gamma_{k,j}} \tag{8}$$

where $n_{\gamma_k} = \sum_{j=1}^{n} \gamma_{k,j}$ and $\pi_j = P(\gamma_{k,j} = 1)$ is the prior probability to select the jth gene. It is set as $\pi_j = 15/n$ according to the total number of samples $m = 22$. If π_j is chosen bigger, we find that often $(\boldsymbol{X}_{\gamma_k}^T\boldsymbol{X}_{\gamma_k})^{-1}$ does not exist. We sample each $\gamma_{k,j}$ independently from

$$p(\gamma_{k,j}|\boldsymbol{y}_k, \gamma_{k,i\neq j}) \propto (1+c)^{(-n_{\gamma_k}/2)} \exp\left[-\frac{1}{2}S(\gamma_k, \boldsymbol{y}_k)\right] \times \pi_j^{\gamma_{k,j}}(1-\pi_j)^{1-\gamma_{k,j}} \quad j = 1, \ldots, n \tag{9}$$

Step 2: Draw β_k from

$$p(\beta_k|\gamma_k, \boldsymbol{y}_k) \propto \mathcal{N}(\boldsymbol{V}_{\gamma_k}\boldsymbol{X}_{\gamma_k}^T\boldsymbol{y}_k, \boldsymbol{V}_{\gamma_k}) \tag{10}$$

where

$$\boldsymbol{V}_{\gamma_k} = \frac{c}{1+c}(\boldsymbol{X}_{\gamma_k}^T\boldsymbol{X}_{\gamma_k})^{-1}$$

Step 3: Draw $\boldsymbol{y}_k = [y_{k,1}, \ldots, y_{k,m}]$, $k = 1, \ldots, K$ from a truncated normal distribution as follows [14]:

For $i = 1, 2, \ldots, m$

- if $w_i = k$, then draw $y_{k,i}$ according to $y_{k,i} \sim \mathcal{N}(\boldsymbol{X}_{\gamma_k}\beta_k, 1)$ truncated left by $\max_{j\neq k} y_{j,i}$, i.e.

$$y_{k,i} \sim \mathcal{N}(\boldsymbol{X}_{\gamma_k}\beta_k, 1)1_{\{y_{k,i} > \max_{j\neq k} y_{j,i}\}} \tag{11}$$

- else $w_i \neq j$ and $j \neq k$, then draw $y_{j,i}$ according to $y_{j,i} \sim \mathcal{N}(\boldsymbol{X}_{\gamma_j}\beta_j, 1)$ truncated right by the newly generated $y_{k,i}$, i.e.

$$y_{j,i} \sim N(\boldsymbol{X}_{\gamma_j}\beta_j, 1)1_{\{y_{j,i} \leq y_{k,i}\}} \tag{12}$$

endfor

Here we set $y_{K,i} \sim \mathcal{N}(0, 1)$ when $w_i = K$. It is problematic if we only draw $\boldsymbol{y}_k = [y_{k,1}, \ldots, y_{k,m}]$ from $k = 1$ to $K - 1$. If we do so, that implies we never use the information in the Kth class. That actually means we introduce a new equation $y_{K,i} = X\beta_K + e_{K,i}$, $i = 1, \ldots, m$, with β_K being a zero vector and $e_{K,i} \sim \mathcal{N}(0, 1)$. Obviously, it is more complex than the binomial case because we only have m observations not Km.

In this study, 35 000 Gibbs iterations were implemented, with the first 5000 as the burn-in period. Then, we obtained the Monte Carlo samples as $\{\gamma_k^{(t)}, \beta_k^{(t)}, \boldsymbol{y}_k^{(t)}, t = 1, \ldots, T\}$, where $T = 30\,000$. Finally, we counted the number of times that each gene appears in $\{\gamma_k^{(t)}, t = 5001, \ldots, T\}$. The genes with the highest appearance frequencies play the strongest role in predicting the target gene. We will discuss some implementation issues of this algorithm in the following section.

2.3 Bayesian estimation using the strongest genes

Now assume the genes corresponding to non-zeros of γ_k are the strongest genes obtained by the above Bayesian variable selection algorithm. We still use $\boldsymbol{X}_{\gamma_k}$ to denote the profiles of these strongest genes. For fixed γ_k, we again use a Gibbs sampler to estimate the probit regression coefficients β_k as follows: First draw β_k according to (10), then draw y_k according to (11) and (12) and iterate the two steps. In this study, 1500 iterations are implemented, with the first 500 as the burn-in period. Thus we obtain the Monte Carlo samples $\{\beta_k^{(t)}, y_k^{(t)}, t = 501, \ldots, \tilde{T}\}$. The probability of a given sample under each class is given by

$$P(w = k|\boldsymbol{X}) = \frac{1}{\tilde{T}}\sum_{t=1}^{\tilde{T}} \prod_{j=1, j\neq k}^{K} \Phi\left(\boldsymbol{X}_{\gamma_k}\beta^{(t)}_{\gamma_k} - \boldsymbol{X}_{\gamma_j}\beta^{(t)}_{\gamma_j}\right) \qquad k = 1, \ldots, K-1 \qquad (13)$$

$$P(w = K|\boldsymbol{X}) = 1 - \sum_{k=1}^{K-1} P(w = k|\boldsymbol{X})$$

where $\beta^{(t)}_{\gamma_K}$ is a zero vector; and the classification of this sample is given by

$$\hat{w} = \arg\max_{1\le k\le K} P(w = k|\boldsymbol{X}) \qquad (14)$$

3 Some issues for fast implementation

The computational complexity of the Bayesian gene selection algorithm in the preceding section is very high. For example, if there are 3000 gene variables, then for each iteration, we have to compute the matrix inverse $(\boldsymbol{X}^T_{\gamma_k}\boldsymbol{X}_{\gamma_k})^{-1}$ 3000 times, because we need to compute (9) for each gene. As each iteration takes about 10 min on the Intel III 930 MHz machine, it will take about one month to run the program for 35 000 iterations. Hence, some fast algorithms must be developed to deal with the problem.

3.1 Preselection method

Here, we discuss the initialisation of the preselection method. In pattern recognition, we usually adopt the following criterion: the smaller the sum of squares within groups and the bigger the sum of squares between groups, the better the classification accuracy. Therefore we can define a score using the above two statistics to preselect genes, i.e. the ratio of the between-group to within-group sum of squares, i.e.

$$R(j) \triangleq \frac{\sum_{i=1}^{m}\sum_{k=1}^{K} 1_{(w_i=k)}(\bar{x}_{k,j} - \bar{x}_j)^2}{\sum_{i=1}^{m}\sum_{k=1}^{K} 1_{(w_i=k)}(x_{i,j} - \bar{x}_{k,j})^2} \qquad 1 \le j \le p \qquad (15)$$

where p is the total number of original genes (the number of genes n used in the Bayesian selection procedure is much smaller than p); $\bar{x}_j$ denotes the average expression level of gene j across all samples, and $\bar{x}_{k,j}$ denotes the average expression level of gene j across the samples belonging to class k, where class k corresponds to $\{w_i = k\}$; and the indicator function 1_Ω is equal to one if event Ω is true and zero otherwise. Next, we use the histogram of $R(j)$ to define a threshold. We use two parameters q and Γ to describe the histogram of (R), where q is a vector with entry being the numbers of elements in each bin when R is divided into ten equally spaced bins; and Γ is a vector that returns the positions of the bin centres in Γ. As $R(j)$ change little for many genes, the genes corresponding to $q(1)$ have little effect on the target gene prediction. Therefore, in this study, we select $\Gamma(3)$ as the threshold, i.e. we keep those genes j such that $R(j) \ge \Gamma(3)$. Notice that we cannot just use this score for selecting strongest genes, because the actual expression data are quite noisy. By using this preselection procedure, we finally obtain n genes such that $R(j) \ge \Gamma(3)$.

3.2 Computation of $p(\gamma_{k,j}|\boldsymbol{y}_k, \gamma_{k,i\neq j})$ in (9)

Because $\gamma_{k,j}$ only takes 0 or 1, we can consider $p(\gamma_{k,j} = 1|\boldsymbol{y}_k, i \neq j)$ and $p(\gamma_{k,j} \neq 0|\boldsymbol{y}_k, i \neq j)$. Let $\gamma_k^1 = (\gamma_{k,1}, \ldots, \gamma_{k,j-1}, \gamma_{k,j} = 1, \gamma_{k,j+1}, \ldots, \gamma_{k,n})$ and $\gamma_k^0 = (\gamma_{k,1}, \ldots, \gamma_{k,j-1}, \gamma_{k,j} = 0, \gamma_{k,j+1}, \ldots, \gamma_{k,n})$. After straightforward computation of (9), we have

$$p(\gamma_{k,j} = 1|\boldsymbol{y}_k, \gamma_{k,i\neq j}) \propto \frac{1}{1+h} \qquad (16)$$

with

$$h = \frac{1-\pi_j}{\pi_j}\exp\left\{\frac{S(\gamma_k^1, \boldsymbol{y}_k) - S(\gamma_k^0, \boldsymbol{y}_k)}{2}\right\}\sqrt{1+c} \qquad (17)$$

If $\gamma_k = \gamma_k^0$ before $\gamma_{k,j}$ is generated, that means we have obtained $S(\gamma_k^0, \boldsymbol{y}_k)$. Thus we only need to compute $S(\gamma_k^1, \boldsymbol{y}_k)$, and *vice versa*.

3.3 Fast computation of $S(\gamma_k)$ in (7)

It is of key importance to compute $S(\gamma_k, \boldsymbol{y}_k)$ fast where a gene variable is added or removed from γ_k. Denote

$$E(\gamma_k, \boldsymbol{y}_k) = \boldsymbol{y}_k^T\boldsymbol{y}_k - \boldsymbol{y}_k^T\boldsymbol{X}_{\gamma_k}(\boldsymbol{X}^T_{\gamma_k}\boldsymbol{X}_{\gamma_k})^{-1}\boldsymbol{X}^T_{\gamma_k}\boldsymbol{y}_k \qquad k = 1, \ldots, K-1 \qquad (18)$$

Then (18) can be computed using the fast QR decomposition, QR-delete and QR-insert algorithms when a variable is added or removed [15] (chap. 10.1.1b). Now we want to estimate $S(\gamma_k, \boldsymbol{y}_k)$ in (7). Comparing (18) and (7), we can obtain the following equation:

$$\boldsymbol{y}_k^T\boldsymbol{X}_{\gamma_k}(\boldsymbol{X}^T_{\gamma_k}\boldsymbol{X}_{\gamma_k})^{-1}\boldsymbol{X}^T_{\gamma_k}\boldsymbol{y}_k = (1+c)[S(\gamma_k, \boldsymbol{y}_k) - E(\gamma_k, \boldsymbol{y}_k)] \qquad (19)$$

Substituting (19) into (7), after straightforward computation, $S(\gamma_k, \boldsymbol{y}_k)$ is given by

$$S(\gamma_k, \boldsymbol{y}_k) = \frac{\boldsymbol{y}_k^T \boldsymbol{y}_k + cE(\gamma_k, \boldsymbol{y}_k)}{1+c} \quad k = 1, \ldots, K-1 \tag{20}$$

Thus, after computing $E(\gamma_k, \boldsymbol{y}_k)$ using QR decomposition, QR-delete and QR-insert algorithms, we obtain $S(\gamma_k, \boldsymbol{y}_k)$. The computational complexity is much smaller than that of the original algorithm [7] owing to our processing techniques. We summarise our fast Bayesian gene selection algorithm:

Algorithm 1 (fast Bayesian gene selection algorithm):

```
• Pre-select genes according to (15)
• Initialisation: Set initial parameters $\gamma_k^{(0)}$, $\beta_k^{(0)}$, $\boldsymbol{y}_k^{(0)}$
• For $t = 1, 2, \ldots, 35\,000$.
    – Draw $\gamma_k^{(t)}$. For $j = 1, \ldots, n$
        * Compute $S(\gamma_k^{(t)}, \boldsymbol{y}_k)$ using QR-delete or QR-insert
        * Compute $p(\gamma_{k,j} = 1 | \boldsymbol{y}_k, \gamma_{k,i \neq j}^{(t)})$ according to (16)
        * Draw $\gamma_{k,j}^{(t)}$ from $p(\gamma_{k,j} = 1 | \boldsymbol{y}_k^{(t-1)}, \gamma_{k,i \neq j}^{(t)})$
    – Draw $\beta_k^{(t)}$ according to (24)
    – Draw $\boldsymbol{y}_k^{(t)}$ according to (11) and (12)
• Endfor
• Count the frequency of each gene appeared in $\gamma_k^{(t)}$, $t = 5001, \ldots, 35\,000$.
```

4 Multinomial probit regression with approximate Bayesian gene selection

In Section 2, we discussed the Bayesian gene selection to obtain different strongest genes for different regression equations. Here, we discuss how to select the same strongest genes for different regression equations. The model is a little different from the model (6), i.e. the selected genes do not change with the different regression equations. This means we can drop index k from γ_k. However, the parameter β is still dependent on k and γ, denoted by $\beta_{k,\gamma}$. Then (6) is rewritten as

$$\boldsymbol{y}_k = \boldsymbol{X}_\gamma \beta_{k,\gamma} + e_k \quad k = 1, \ldots, K-1 \tag{21}$$

Now the problem is how to estimate γ and the corresponding $\beta_{k,\gamma}$ and $\boldsymbol{y}_k$ for each equation in (21). We will show in the Appendix (Section 9) that the posterior distribution $p(\gamma | \boldsymbol{y}_1, \ldots, \boldsymbol{y}_{K-1})$ is approximated by

$$\begin{aligned} p(\gamma | \boldsymbol{y}_1, \ldots, \boldsymbol{y}_{K-1}) &\\ \propto p(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{K-1} | \gamma) p(\gamma) &\\ \propto (1+c)^{-[(K-1)n_\gamma/2]} \exp\left\{ -\frac{1}{2} \sum_{k=1}^{K-1} S(\gamma, \boldsymbol{y}_k) \right\} &\\ \times \prod_{i=1}^{n} \pi_i^{\gamma_i} (1 - \pi_i)^{1-\gamma_i} & \end{aligned} \tag{22}$$

and the posterior distribution $p(\beta_{k,\gamma} | \boldsymbol{y})$ is given by $\beta_{k,\gamma} | \boldsymbol{y}_k, \boldsymbol{X}_\gamma \sim \mathcal{N}(\boldsymbol{V}_\gamma \boldsymbol{X}_\gamma^T \boldsymbol{y}_k, \boldsymbol{V}_\gamma)$. The Gibbs sampling algorithm for estimating γ, $\{\beta_{k,\gamma}\}$, $\{\boldsymbol{y}_k\}$ is as follows:

Step 1: Draw γ from $p(\gamma | \boldsymbol{y}_1, \ldots, \boldsymbol{y}_{K-1})$. We usually sample each γ_j independently from

$$\begin{aligned} p(\gamma_i | \boldsymbol{y}_1, \ldots, \boldsymbol{y}_{K-1}, \gamma_{j \neq i}) &\\ \propto p(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{K-1} | \gamma) p(\gamma_i) \propto (1+c)^{-[(K-1)n_\gamma/2]} &\\ \times \exp\left\{ -\frac{1}{2} \sum_{k=1}^{K-1} S(\gamma, \boldsymbol{y}_k) \right\} \pi_i^{\gamma_i} (1-\pi_i)^{1-\gamma_i} & \end{aligned} \tag{23}$$

where $n_\gamma = \sum_{j=1}^{n} \gamma_j$ and $c = 10$ in this study. See the Appendix (Section 9) for the derivation and fast implementation.

Step 2: Draw $\beta_{k,\gamma}$ from

$$p(\beta_{k,\gamma} | \gamma, \boldsymbol{y}_k) \propto \mathcal{N}(\boldsymbol{V}_\gamma \boldsymbol{X}_\gamma^T \boldsymbol{y}_k, \boldsymbol{V}_\gamma) \tag{24}$$

where

$$\boldsymbol{V}_\gamma = \frac{c}{1+c} (\boldsymbol{X}_\gamma^T \boldsymbol{X}_\gamma)^{-1}$$

Step 3: Draw $y_{k,i}$, $k = 1, \ldots, K$, $i = 1, \ldots, m$, according to (11) and (12). It is the same as the algorithm given in Section 2.

Again, 35 000 Gibbs iterations are implemented, with the first 5000 as burn-in period. Then, we obtain the Monte Carlo samples as $\{\gamma^{(t)}, \beta_{k,\gamma}^{(t)}, \boldsymbol{y}_k^{(t)}, t = 1, \ldots, T\}$, where $T = 35\,000$. Finally, we count the number of times that each gene appears in $\{\gamma^{(t)}, t = 5001, \ldots, T\}$. The genes with the highest appearance frequencies play the strongest role in predicting the target gene. The implementation issues of this algorithm are similar to the discussion in the preceding section, and the difference is shown in the Appendix (Section 9).

The classification step is almost the same as the corresponding step in Section 2. First draw $\beta_{k,\gamma}$ according to (24), then draw $\boldsymbol{y}_k$, and iterate the two steps. In this study, 1500 iterations are implemented, with the first 500 as the burn-in period. We obtain the Monte Carlo samples $\{\beta_{k,\gamma}^{(t)}, \boldsymbol{y}_k^{(t)}, t = 501, \ldots, \tilde{T}\}$. The probability of a given sample under each class is given by

$$P(w = k | \boldsymbol{X}) = \frac{1}{\tilde{T}} \sum_{t=1}^{\tilde{T}} \prod_{j=1,\, j \neq k}^{K} \Phi(\boldsymbol{X}_\gamma \beta_{k,\gamma}^{(t)} - \boldsymbol{X}_\gamma \beta_{j,\gamma}^{(t)}) \quad k = 1, \ldots, K-1 \tag{25}$$

$$P(w = K | \boldsymbol{X}) = 1 - \sum_{k=1}^{K-1} P(w = k | \boldsymbol{X})$$

where $\beta_{K,\gamma}^{(t)}$ is a zero vector. The difference between (25) and (14) is that we have the same strongest genes $\boldsymbol{X}_\gamma$ here. The classification of this sample is the same as (14).

Obviously, the proposed gene selection method in Section 2 employs more information than the method in this section, and we may think it is better to employ the method in Section 2 for classification because different important genes affect different cancers. However, it is also important to study how to select the same strongest genes for all classes; for example, it is better to employ the method developed in this section when we study gene regulatory networks [16], because we usually need to know which genes are the most important genes to affect the target gene.

5 Experimental results

In Section 2, we proposed a gene selection algorithm: different classes have different discriminant strongest genes. As we would usually like to see what the strongest genes are, here we also consider another method: after obtaining the frequencies for each gene in each equation, we rank the genes according to the appearance frequencies in all equations. We call the two methods MPB-1 and MPB-2, respectively. The algorithm proposed in Section 4 is called MPBA.

5.1 Breast cancer data

In our first experiment, we will focus on hereditary breast cancer data, which can be downloaded from the web page for the original paper [17]. In [17], cDNA microarrays are used in conjunction with classification algorithms to show the feasibility of using differences in global gene expression profiles to separate BRCA1 and BRCA2 mutation-positive breast cancers. Twenty-two breast tumour samples from 21 patients were examined: seven BRCA1, eight BRCA2 and seven sporadic. There are 3226 genes for each tumour sample. We use our methods to classify BRCA1, BRCA2 and sporadic. The ratio data are truncated from below at 0.1 and above at 20. The leave-one-out cross-validation method (LOOCV) is employed to compute all classification errors in this paper. At every instance of LOOCV, we used the proposed Gibbs sampling method to obtain the appearance frequency for each gene, then the average of the frequencies of each gene in LOOCV is regarded as the final appearance frequency, and the genes with the highest appearance frequencies are used as training genes.

Table 1 lists the strongest genes using the MPB-1, MPB-2 and MPBA methods (for reading ease, instead of CloneID, we use the gene index number in the database [17]). Gene 10 (phosphofructokinase, platelet) is the most important gene for all methods. It is also an important gene listed in [5, 7].

Using the top ten genes for classification, the recognition accuracy is zero error for MPB-1, one error for MPB-2 and zero error for MPBA. Looking at the probabilities for each sample using the MPB-1 method and MPBA method in Tables 2 and 3, respectively, it is seen that the estimated probabilities of some sporadic and BRCA2 are very close. It is also seen that the probabilities of the MPB-1 method are more distinct than those of the MPBA method. We obtained the same classification results as the MPBA's when we tested the logistic method [2] and support vector machine [18] in this data set.

Table 1: Index number of strongest genes for breast cancer data

Number	MPB-1		MPB-2	MPBA
	1	2		
1	10	10	10	10
2	2699	110	110	560
3	110	2119	2699	1008
4	157	838	1725	1209
5	1725	2456	2294	2636
6	2294	2294	2119	2280
7	1656	118	157	2977
8	1008	2984	838	110
9	2979	501	118	2734
10	750	1725	750	2715

The convergence of the MPB-1 method is monitored over the 5000 burn-in iterations: the parameters $1/T_1 \sum_{t=1}^{T_1} \gamma_k^{(t)}$ and $1/T_1 \sum_{t=1}^{T_1} \beta_k^{(t)}$ are almost the same when $T_1 > 4500$, respectively. For setting the prior of π_j, we also tested two other cases, i.e. $\pi_j = 10/n$ and $\pi_j = 5/n$. We found the gene index 10, 110, 2699, 157 and 750 always appeared in the top ten genes when we used MBP-2. This result is consistent with that in the binary probit model [7]. We

Table 2: Estimated probabilities for each sample using MPB-1

Number	w	$P(w=0\|X)$	$P(w=1\|X)$	$P(w=2\|X)$	Classification
1	0	0.9090	0.0672	0.0238	0
2	0	0.6732	0.1082	0.2186	0
3	0	0.3752	0.3198	0.3050	0
4	0	0.7110	0.2022	0.0868	0
5	0	0.6663	0.2947	0.0390	0
6	0	0.9429	0.0364	0.0207	0
7	1	0.4625	0.5304	0.0071	1
8	1	0.1172	0.7885	0.0943	1
9	1	0.0152	0.6316	0.35330	1
10	1	0.0646	0.4483	0.4871	1
11	2	0.0744	0.3321	0.5935	2
12	2	0.0366	0.4094	0.5540	2
13	2	0.2771	0.0689	0.6540	2
14	2	0.3213	0.2599	0.4188	2
15	2	0.3558	0.0175	0.6267	2
16	2	0.0050	0.0308	0.9642	2
17	2	0.3704	0.0235	0.5061	2
18	0	0.9867	0.0089	0.0044	0
19	1	0.0042	0.8583	0.1375	1
20	1	0.0091	0.9111	0.0798	1
21	1	0.1059	0.8881	0.0060	1
22	1	0.2062	0.7758	0.0180	1

Table 3: Estimated probabilities for each sample using MPBA

Number	w	$P(w=0\|X)$	$P(w=1\|X)$	$P(w=2\|X)$	Classification
1	0	0.5286	0.2714	0.2000	0
2	0	0.4638	0.1661	0.3701	0
3	0	0.4464	0.2152	0.3384	0
4	0	0.5555	0.1826	0.2619	0
5	0	0.4238	0.2933	0.2829	0
6	0	0.6411	0.0576	0.3013	0
7	1	0.0906	0.6154	0.2940	1
8	1	0.1198	0.4978	0.3824	1
9	1	0.2960	0.4061	0.2979	1
10	1	0.2370	0.41580	0.3470	1
11	2	0.1807	0.3785	0.4408	2
12	2	0.1828	0.3046	0.5126	2
13	2	0.3015	0.3016	0.3969	2
14	2	0.2095	0.2406	0.5499	2
15	2	0.2723	0.2617	0.4660	2
16	2	0.2348	0.2233	0.5419	2
17	2	0.3294	0.1946	0.4760	2
18	0	0.4998	0.1715	0.3287	0
19	1	0.1817	0.4957	0.3226	1
20	1	0.1188	0.5755	0.3057	1
21	1	0.2835	0.3931	0.3234	1
22	1	0.2089	0.5778	0.2133	1

then fixed $\pi_j = 10/n$, and changed the parameter c from 10 and 100 by. We again found that some genes, such as 10, 110 and 2699 always appeared in the top ten genes.

5.2 Small round blue-cell tumours

In this experiment, we consider the small, round blue cell tumours (SRBCTs) of childhood, which include neuroblastoma (NB), rhabdomyosarcoma (RMS), non-hodgkin lymphoma (NHL) and the Ewing family of tumours (EWS) in [19]. The data set of the four cancers is composed of 2308 genes and 63 samples, where the NB has 12 samples; the RMS has 23 samples; the NHL has eight samples and the EMS has 20 samples. We use our methods to classify the four cancers. The ratio data are truncated from below at 0.01.

Table 4 lists the strongest genes using the MPB-1, MPB-2 and MPBA methods. It is seen that gene 1 (clone ID 21652, (catenin, alpha 1)) is the strongest for all methods. It is also an important gene in [19]. Some other genes, such as gene 255 (clone ID 325182), gene 2050 (clone ID 295985), gene 1389 (clone ID 770394), gene 246 (clone ID 377461) and gene 107 (clone ID 365826), are also important genes listed in [19].

Using the top ten genes for classification, the recognition accuracy is no error for MPB-1, one error for MPB-2 and one error for MPBA. Selecting different strongest genes brings a better recognition result. approximate Bayesian selection method MPBA and the MPB-2 method have a slightly worse performance. We also tested the setting of π_j and the parameter c. We set $\pi_j = 20/n$, $\pi_j = 10/n$ and $\pi_j = 5/n$, as well as changing the parameter c from 10 to 100. We again found that some genes such as gene 255 and gene 2050 always appeared in the top ten genes.

Table 4: Index number of strongest genes for small round blue-cell tumours

Number	MPB-1			MPB-2	MPBA
	1	2	3		
1	1	1	1	1	1
2	2050	2	1955	2	255
3	1954	1916	255	1955	1389
4	1319	123	2	246	246
5	246	335	2144	107	107
6	1389	2046	1601	1954	2
7	1645	74	742	74	2050
8	2	2235	107	187	742
9	188	846	2046	2050	187
10	1497	107	800	255	1319

5.3 NCI60 data

The National Cancer Institute's anti-cancer drug-screen data (NCI60) of Ross *et al.* [20] consist of 61 samples from human cancer cell lines. Gene expression levels were measured for $n = 10\,000$ genes. For illustration of the proposed multi-class cancer classification methods, we consider the following four tumour classes: nine renal, eight melanoma, six leukaemia and seven colon. We use the ratio data directly, truncated from below at 0.01 and from above at 20.

Table 5 lists the strongest genes using the MPB-1, MPB-2 and MPBA methods. It is seen that gene 428 (human neuroendocrine-dlg (NE-dlg) mRNA) is the most important gene

Table 5: Index number of strongest genes for NCI60 data

Number	MPB-1			MPB-2	MPBA
	1	2	3		
1	428	6045	7357	428	2252
2	3822	428	428	6045	2967
3	9535	7335	2248	7357	428
4	6045	8864	701	8864	1116
5	2536	7183	9564	701	828
6	1715	3797	3073	7335	1222
7	9643	1371	7183	7183	1371
8	8864	701	865	2107	9535
9	6093	3800	1116	1371	7357
10	5268	5379	9770	9513	5131

Table 6: Index number of the strongest genes for acute leukaemia data

Number	MPB-1		MPB-2	MPBA
	1	2		
1	181	6345	6345	181
2	2853	4913	181	6345
3	5489	181	4913	4913
4	5300	5389	5389	5389
5	7101	4684	4684	4377
6	4913	4680	2853	1291
7	5352	5135	3189	5300
8	2445	1809	4535	6218
9	357	4535	4772	5932
10	2642	2639	5352	5191

for all methods. Some other genes, such as gene 7357 (human 54 kDa progesterone receptor-associated immunophilin FKBP54 mRNA), gene 1371 (ESTs), gene 9535 (ETV4 Ets variant gene 4) and so on are important genes. Using the top ten genes for classification, the recognition accuracy is one error for MPB-1, two errors for MPB-2 and one error for MPBA. We set $\pi_j = 20/n$, $\pi_j = 10/n$ and $\pi_j = 5/n$ and changed the parameter c from 10 to 100. We found that some genes, such as gene 7357 and gene 9535, always appeared in the top ten genes.

5.4 Acute leukaemia data

We have also applied the proposed methods to the leukaemia data of [9], which are publicly available at http://www.genome.wi.mit.edu/cgi-bin/cancer/publications/pub. The microarray data contain 7129 human genes, sampled from 72 cases of cancer, of which 38 are of type B-cell ALL, nine are of type T-cell ALL, and 25 are of type AML. The data are preprocessed as recommended in [1]: gene values are truncated from below at 100 and from above at 16 000; genes having a ratio of the maximum over the minimum less than 5 or a difference between the maximum and the minimum less than 500 are excluded; and, finally, the base-10 logarithm is applied to the 3571 remaining genes. Here, we study the 38 samples in the training set, which is composed of 19 B-cell ALL, eight T-cell ALL and 11 AML.

Table 6 lists the strongest genes using the MPB-1, MPB-2 and MPBA methods. It is seen that gene 181 (IL2RG interleukin 2 receptor gamma chain), gene 6345 (GLUL glutamate-ammonia ligase), gene 4913 (SMT3A protein) and gene 5389 (clone CIITA-8 MHC class II transactivator CIITA mRNA) are the most important genes for all methods. Using the top ten genes for classification, the recognition accuracy is one error for the three methods. We found that sample 2 (T-cell) is misclassified into AML. We set $\pi_j = 25/n$, $\pi_j = 10/n$ and $\pi_j = 5/n$ and changed the parameter c from 10 to 100. We found that some genes, such as gene 6345 and gene 5389, always appeared in the top ten genes.

6 Conclusions

In this paper, we have studied the problem of multi-class cancer classification from gene expression data. We discussed the multinomial probit regression with Bayesian gene selection. Two Bayesian gene selection schemes are proposed: one employs different strongest genes for different probit regression equations, and the other one employs the same strongest genes for all probit regression equations. Some fast implementation issues for this Bayesian gene selection are discussed, including preselection of the strongest genes and computation of the estimation errors using recursive QR decomposition. The proposed gene selection techniques were applied to the analysis of breast cancer data, small round blue-cell tumours, the NCI's anti-cancer drug-screen data and acute leukaemia data. The recognition accuracies are very high using our proposed methods.

7 Acknowledgment

This research was supported by the National Human Genome Research Institute.

8 References

1 Dudoit, S., Fridlyand, J., and Speed, T.P.: 'Comparison of discrimination methods for the classification of tumors using gene expression data', *J. Amer. Statist. Assoc.*, 2002, **97**, (457), pp. 77–87

2 Nguyen, D.V., and Rocke, D.M.: 'Multi-class cancer classification via partial least squares using gene expression profiles', *Bioinformatics*, 2002, **18**, pp. 1216–1226

3 Zhang, H.P., Yu, C.Y., Singer, B., and Xiong, M.: 'Recursive partitioning for tumor classification with gene expression microarray data', *Proc. Nat. Acad. Sci. USA*, 2001, **98**, pp. 6730–6735

4 Guyon, I., Weston, J., Barnhill, S., and Vapnik, V.: 'Gene selection for cancer classification using support vector machines', *Mach. Learning*, 2002, **46**, pp. 389–422

5 Kim, S., Dougherty, E.R., Barrea, J., Chen, Y., Bittner, M., and Trent, J.M.: 'Strong feature sets from small samples', *J. Comput. Biol.*, 2002, **9**, pp. 127–146

6 Chipman, H., George, E.I., and McCulloch, R.: 'The practical implementation of Bayesian model selection', *IMS Lect. Notes – Monogr. Ser.*, 2001, **38**, pp. 117–124

7 Lee, K.E., Sha, N., Dougherty, E.R., Vannucci, M., and Mallick, B.K.: 'Gene selection: a Bayesian variable selection approach', *Bioinformatics*, 2003, **19**, pp. 90–97

8 Jornsten, R., and Yu, B.: 'Simultaneous gene clustering and subset selection for classification via MDL', *Bioinformatics*, 2003

9 Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A., Bloomfield, C.D., and Lander, E.S.: 'Molecular classification of cancer: class discovery and class prediction by gene expression monitoring', *Science*, 1999, **286**, pp. 531–537

10 Albert, J., and Chib, S.: 'Bayesian analysis of binary and polychotomous response data', *J. Amer. Statist. Assoc.*, 1993, **88**, pp. 669–679

11 Smith, M., and Kohn, R.: 'Nonparametric regression using Bayesian variable selection', *J. Econometrics*, 1997, **75**, pp. 317–344

12 Imai, K., and van Dyk, D.A.: 'A Bayesian analysis of the multinomial probit model using marginal data augmentation', *J. Econometrics*, 2005, **124**, pp. 311–334

13 Yau, P., Kohn, R., and Wood, S.: 'Bayesian variable selection and model averaging in high-dimensional multinomial nonparametric regression', *J. Comput. Graph. Statist.*, 2003, **12**, pp. 23–54

14 Robert, C.: 'Simulation of truncated normal variables', *Statist. Comput.*, 1995, **5**, pp. 121–125

15 Seber, G.A.F.: 'Multivariate Observations' (Wiley, New York)

16 Zhou, X., Wang, X., and Dougherty, E.D.: 'Construction of genomic networks using mutual-information clustering and reversible-jump Markov-Chain-Monte-Carlo predictor design', *Signal Processing*, 2003, **84**, (4), pp. 745–761

17 Hedenfalk, I., Duggan, D., Chen, Y., Radmacher, M., Bittner, M., Simon, R., Meltzer, P., Gusterson, B., Esteller, M., Raffeld, M., Yakhini, Z., Ben-Dor, A., Dougherty, E., Kononen, J., Bubendorf, L., Fehrle, W., Pittaluga, S., Gruvberger, S., Loman, N., Johannsson, O., Olsson, H., Wilfond, B., Sauter, G., Kallioniemi, O.-P., Borg, A., and Trent, J.: 'Gene expression profiles in hereditary breast cancer', *New England J. Medic.*, 2001, **344**, pp. 539–548

18 Brown, M.P.S., Noble Grundy, W., Lin, D., Cristianini, N., Walsh Sugnet, C., Furey, T.S., Ares, M. Jr., and Haussler, D.: 'Knowledge-based analysis of micrarray gene expression data by using support vector machines', *Proc. Nat. Acad. Sci. USA*, 2000, **97**, pp. 262–267

19 Khan, J., Wei, J.S., Ringnr, M., Saal, L.H., Ladanyi, M., Westermann, F., Berthold, F., Schwab, M., Antonescu, C.R., Peterson3, C., and Meltzer, P.S.: 'Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks', *Nature Medicine*, 2001, **7**, pp. 673–679

20 Ross, D.T., Scherf, U., Eisen, M.B., Perou, C.M., Rees, C., Spellman, P., Iyer, V., Jeffrey, S.S., de Rijn, M.V., Waltham, M., Pergamenschikov, A., Lee, J.C.F., Lashkari, D., Shalon, D., Myers, T.G., Weinstein, J.N., Botstein, D., and Brown, P.O.: 'Systematic variation in gene expression patterns in human cancer cell lines', *Nature Genetics*, 2000, **24**, (3), pp. 227–235

21 Heyer, L.J., Kruglyak, S., and Yooseph, S.: 'Exploying expression data: identification and analysis of coexpressed genes', *Genome Res.*, 1999, **9**, pp. 1106–1115

9 Appendix

9.1 Derivation of (22) and (24)

First, according to the Bayesian theorem, we have

$$p(\boldsymbol{y}_k, \beta_{k,\gamma}|\boldsymbol{X}_\gamma) \propto p(\boldsymbol{y}_k|\beta_{k,\gamma}, \boldsymbol{X}_\gamma)p(\beta_{k,\gamma}) \quad k = 1, \ldots, K-1 \tag{26}$$

As

$$\boldsymbol{y}_k|\beta_{k,\gamma}, \boldsymbol{X}_\gamma \sim \mathcal{N}(\boldsymbol{X}_\gamma\beta_{k,\gamma}, \boldsymbol{I}_m)$$
$$\beta_{k,\gamma} \sim \mathcal{N}(0, c(\boldsymbol{X}_\gamma^T\boldsymbol{X}_\gamma)^{-1} \quad k = 1, \ldots, K-1$$

then, we have

$$p(\boldsymbol{y}_k|\beta_{k,\gamma}, \boldsymbol{X}_\gamma) \propto \exp\left\{-\frac{1}{2}(\boldsymbol{y}_k - \boldsymbol{X}_\gamma\beta_{k,\gamma})^T(\boldsymbol{y}_k - \boldsymbol{X}_\gamma\beta_{k,\gamma})\right\} \tag{27}$$

$$p(\beta_{k,\gamma}) \propto c^{-(n_\gamma/2)}|\boldsymbol{X}_\gamma^T\boldsymbol{X}_\gamma|^{1/2}\exp\left\{-\frac{1}{2}\beta_{k,\gamma}^T(\boldsymbol{X}_\gamma^T\boldsymbol{X}_\gamma)\beta_{k,\gamma}\right\} \tag{28}$$

Using (26)–(28), we have

$$\begin{aligned} p(\boldsymbol{y}_k, \beta_{k,\gamma}, |\boldsymbol{X}_\gamma) \propto\ & c^{-(n_\gamma/2)}|\boldsymbol{X}_\gamma^T\boldsymbol{X}_\gamma|^{1/2} \\ & \times \exp\left\{-\frac{1}{2}(1 + c^{-1})\beta_{k,\gamma}^T(\boldsymbol{X}_\gamma^T\boldsymbol{X}_\gamma)\beta_{k,\gamma}\right. \\ & \left. + \beta_{k,\gamma}^T\boldsymbol{X}_\gamma^T\boldsymbol{y}_k - \frac{1}{2}\boldsymbol{y}_k^T\boldsymbol{y}_k\right\} \end{aligned} \tag{29}$$

Define

$$\boldsymbol{V}_\gamma \triangleq (1 + c^{-1})^{-1}(\boldsymbol{X}_\gamma^T\boldsymbol{X}_\gamma)^{-1} \tag{30}$$

$$\beta_{k,0} \triangleq \boldsymbol{V}_\gamma\boldsymbol{X}_\gamma^T\boldsymbol{y}_k = (1 + c^{-1})^{-1}(\boldsymbol{X}_\gamma^T\boldsymbol{X}_\gamma)^{-1}\boldsymbol{X}_\gamma^T\boldsymbol{y}_k \tag{31}$$

Then, we have

$$|\boldsymbol{V}_\gamma|^{1/2} \cdot c^{-(n_\gamma/2)}|\boldsymbol{X}_\gamma^T\boldsymbol{X}_\gamma|^{1/2} = \left(\frac{1}{1+c}\right)^{n_\gamma/2} \tag{32}$$

Note that the following equality holds:

$$\begin{aligned} & -\frac{1}{2}(1 + c^{-1})\beta_{k,\gamma}^T(\boldsymbol{X}_\gamma^T\boldsymbol{X}_\gamma)\beta_{k,\gamma} + \beta_{k,\gamma}^T\boldsymbol{X}_\gamma^T\boldsymbol{y}_k - \frac{1}{2}\boldsymbol{y}_k^T\boldsymbol{y}_k \\ & = -\frac{1}{2}(\beta_{k,\gamma} - \beta_{k,0})^T\boldsymbol{V}_\gamma^{-1}(\beta_{k,\gamma} - \beta_{k,0}) \\ & \quad - \frac{1}{2}\underbrace{\left[\boldsymbol{y}_k^T\boldsymbol{y}_k - \frac{c}{c+1}\boldsymbol{y}_k^T\boldsymbol{X}_\gamma(\boldsymbol{X}_\gamma^T\boldsymbol{X}_\gamma)^{-1}\boldsymbol{X}_\gamma^T\boldsymbol{y}_k\right]}_{\triangleq S(\gamma, \boldsymbol{y}_k)} \\ & = -\frac{1}{2}(\beta_{k,\gamma} - \beta_{k,0})^T\boldsymbol{V}_\gamma^{-1}(\beta_{k,\gamma} - \beta_{k,0}) - \frac{1}{2}S(\gamma, \boldsymbol{y}_k) \end{aligned} \tag{33}$$

Then (29) becomes

$$\begin{aligned} p(\boldsymbol{y}_k, \beta_{k,\gamma}|\boldsymbol{X}_\gamma) \propto\ & c^{-(n_\gamma/2)}|\boldsymbol{X}_\gamma^T\boldsymbol{X}_\gamma|^{1/2} \\ & \times \exp\left\{-\frac{1}{2}(\beta_{k,\gamma} - \beta_{k,0})^T\boldsymbol{V}_\gamma^{-1}(\beta_{k,\gamma} - \beta_{k,0})\right\} \\ & \times \exp\left\{-\frac{1}{2}S(\gamma, \boldsymbol{y}_k)\right\} \end{aligned} \tag{34}$$

Then, we have

$$\begin{aligned} p(\boldsymbol{y}_k|\gamma) &= \int_{\beta_{k,\gamma}} p(\boldsymbol{y}_k|\beta_{k,\gamma}, \boldsymbol{X}_\gamma)p(\beta_{k,\gamma})d\beta_{k,\gamma} \\ & \propto (1 + c)^{-(n_\gamma/2)}\exp\left\{-\frac{1}{2}S(\gamma, \boldsymbol{y}_k)\right\} \quad k = 1, \ldots, K-1 \end{aligned} \tag{35}$$

We have

$$\begin{aligned} p(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{K-1}|\gamma) &\simeq p(\boldsymbol{y}_1|\gamma) \times \cdots \times p(\boldsymbol{y}_{K-1}|\gamma) \\ & \propto (1 + c)^{-[(K-1)n_\gamma/2]}\exp\left\{-\frac{1}{2}\sum_{k=1}^{K-1}S(\gamma, \boldsymbol{y}_k)\right\} \end{aligned} \tag{36}$$

Note that

$$p(\boldsymbol{y}_k, \beta_{k,\gamma}|\boldsymbol{X}_\gamma) = p(\beta_{k,\gamma}|\boldsymbol{X}_{k,\gamma}, \boldsymbol{y}_k)p(\boldsymbol{y}_k|\gamma) \tag{37}$$

Comparing (34), (35) and (37), we have

$$\begin{aligned} & p(\beta_{k,\gamma}|\boldsymbol{y}_k, \boldsymbol{X}_\gamma) \\ & \propto |\boldsymbol{V}_\gamma|^{-1/2}\exp\left\{-\frac{1}{2}(\beta_{k,\gamma} - \beta_{k,0})^T\boldsymbol{V}_\gamma^{-1}(\beta_{k,\gamma} - \beta_{k,0})\right\} \end{aligned} \tag{38}$$

That is, $\beta_{k,\gamma}|\boldsymbol{y}_k, \boldsymbol{X}_\gamma \sim \mathcal{N}(\boldsymbol{V}_\gamma\boldsymbol{X}_\gamma^T\boldsymbol{y}_k, \boldsymbol{V}_\gamma)$. Therefore (24) holds. Then,

$$p(\gamma|\boldsymbol{y}_1,\ldots,\boldsymbol{y}_{K-1}) \propto p(\boldsymbol{y}_1,\ldots,\boldsymbol{y}_{K-1}|\gamma)p(\gamma)$$
$$\propto (1+c)^{-[(K-1)n_\gamma/2]}\exp\left\{-\frac{1}{2}\sum_{k=1}^{K-1}S(\gamma,\boldsymbol{y}_k)\right\}$$
$$\times\prod_{i=1}^{n}\pi_i^{\gamma_i}(1-\pi_i)^{1-\gamma_i} \quad (39)$$

Moreover,

$$p(\gamma_i\,|\,\boldsymbol{y}_1,\ldots,\boldsymbol{y}_{K-1},\gamma_{j\neq i})$$
$$\propto p(\boldsymbol{y}_1,\ldots,\boldsymbol{y}_{K-1}|\gamma)p(\gamma_i)$$
$$\propto (1+c)^{-[(K-1)n_\gamma/2]}\exp\left\{-\frac{1}{2}\sum_{k=1}^{K-1}S(\gamma,\boldsymbol{y}_k)\right\}$$
$$\times\,\pi_i^{\gamma_i}(1-\pi_i)^{1-\gamma_i} \quad (40)$$

Denote

$$\phi_1 \overset{\Delta}{=} p(\gamma_i=1|\boldsymbol{y}_1,\ldots,\boldsymbol{y}_{K-1},\gamma_{j\neq i}) \propto \pi_i(1+c)^{-[(K-1)n_{\gamma^{(1)}}/2]}$$
$$\times\exp\left\{-\frac{1}{2}\sum_{k=1}^{K-1}S(\gamma^{(1)},\boldsymbol{y}_k)\right\} \quad (41)$$

After straightforward computation, we have

$$\phi_1=\frac{1}{1+h} \quad (42)$$

with

$$h \overset{\Delta}{=} \frac{1-\pi_i}{\pi_i}(1+c)^{(K-1)/2}$$
$$\times\exp\left[\sum_{k=1}^{K-1}\left(S(\gamma^{(1)},\boldsymbol{y}_k)-S(\gamma^{(0)},\boldsymbol{y}_k)\right)\right] \quad (43)$$