國立交通大學

統計學研究所

碩士論文

用ＭＣＭＣ的方法來作基因選取與預測

Gene Selection and Prediction by MCMC Method

研究生：嚴翠英

指導教授：洪慧念　教授

　　　　　洪志真　教授

中華民國九十三年六月

用ＭＣＭＣ的方法來作基因選取與預測

Gene Selection and Prediction by MCMC Method

研究生：嚴翠英　　Student：Tsui-Ying Yen

指導教授：洪志真　　Advisor：Dr. Jyh-Jen Horng Shiau

洪慧念　　Dr. Hui-Nien Hung

國立交通大學

統計學研究所

碩士論文

A Thesis

Submitted to Institute of Statistics

College of Science

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Master

in

Statistics

June 2004

Hsinchu, Taiwan, Republic of China

中華民國九十三年六月

用ＭＣＭＣ的方法來作基因選取與預測

研究生：嚴翠英　　　　　　　　　　指導教授：洪志真 博士

洪慧念 博士

國立交通大學統計學研究所

摘 要

生物晶片的實驗能在短暫的時間內提供我們數以千計的基因資料；此時，如何從中找出重要的基因成為大家關心的問題。在 2003 年，Lee *et al.* 提出一個層級性的貝氏模型來選取基因，他們採用潛在變數來建立迴歸模型，然後用混合的貝氏先驗分配來執行基因選取的動作，MCMC 中的 Gibbs sampling 是他們模擬參數的方法。在此篇論文中，我們修正了他們在基因選取與作預測的演算法，並且，我們也成功的把它運用在俱有遺傳性的乳癌資料上面，主要是區別在 BRCA1 和 BRCA2 二種腫瘤上的突變基因。

關鍵字: 基因選取，MCMC，Gibbs sampling, 預測。

Gene Selection and Prediction by MCMC Method

Student：Tsui-Ying Yen　　　　Advisors：Dr. Jyh-Jen Horng Shiau
Dr. Hui-Nien Hung

Institute of Statistics
National Chiao Tung University

ABSTRACT

DNA micro-array experiments provide us thousands of genes data at once. How to identify the responsible genes is an important problem. Lee *et al.* (2003) propose a hierarchical Bayesian model for gene selection. They use latent variables to specialize the model as a regression setting, and then use a Bayesian mixture prior to perform the gene selection. The method they use to simulate parameters is Gibbs sampling, one kind of MCMC method. We modify their algorithm of gene selection and prediction in this paper. The method is applied successfully to hereditary breast cancer data to classify tumors with BRCA1 and BRCA2 mutations.

Key words : gene selection, MCMC, prediction.

誌　謝

這篇論文終於完成了，回首這些日子，心中有著無限的感動與感激，首先我要感謝的是自己，因為自己無比的毅力，才能堅持到最後，雖說這篇論文稱不上是一件頂尖的作品，但卻是我盡心盡力的傑作。除此之外，還要感謝曾經給我幫助與鼓勵的師長與同學們，我要將此篇論文獻給大家：

感謝我的指導老師 洪志真教授與 洪慧念教授，他們總是在我遇到瓶頸時推我一把，幫助我繼續前進；感謝口試委員 許文郁教授與 謝淑蓉教授細心的指導與建議，讓此篇論文更趨於完美；感謝我的研究伙伴 淑淨、寶文、巧慧，由於彼此的切磋琢磨與相互支持，才能解決所有難題，渡過這些感覺起來很漫長的日子；感謝我的同學 崢珮，由於她的熱心公益，讓我能夠順利地通過口試；此外，還要感謝班上其他同學，在我有困難時，給我適時的幫助與鼓勵，以及感謝時常陪我打球的學長 牛維方、陳泰賓與碩一的學弟妹們，讓我能夠舒解不少壓力與煩燥的情緒。

最後，我要感謝我的父母與家人，提供我一個溫暖又祥和的避風港；還有我的男朋友 彥霖，時常分享其寶貴的經驗，並且時時叮嚀我「人無遠慮，必有近憂」的道理，讓我獲益良多。

真的很感謝各位！

嚴翠英　　謹誌于
國立交通大學統計學研究所
中華民國九十三年六月

Contents

1. Introduction

DNA micro-array technology has made expression measurements for thousands of genes in a single experiment possible. One challenge is to find the genes most likely differentially expressed among different classes. Then, these informative genes can be used to classify a new observation.

Lee *et al.*(2003) propose a hierarchical Bayesian model for gene selection. They employ latent variables $\boldsymbol{Z}$ to specialize the model to a regression setting and use a Bayesian mixture prior to perform the gene selection, where the unknowns are $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$. By assigning a prior distribution π over the dimension of the model, they can control the size of model (number of significant genes). Owing to the posterior distributions of the parameters, $\boldsymbol{Z}$, $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ are not in explicit form, they use a combination of truncated sampling and Markov Chain Monte Carlo (MCMC) based computation techniques, specifically Gibbs sampling (Gelfand and Smith, 1990), to simulate the parameters from the posterior distribution. This model is very flexible to find significant genes as well as to perform future predictions.

As using the Gibbs sampler, we have to take samples from the stationary distribution. From the model of Lee *et al.* (2003), $\boldsymbol{\gamma}$, $\boldsymbol{\beta}$ and $\boldsymbol{Z}$, which are correlated with each other, are our unknowns. Typically, we first find their marginal conditional distribution, and then iterate $f(\boldsymbol{\gamma}^{t+1}|\boldsymbol{\beta}^t, \boldsymbol{Z}^t)$, $f(\boldsymbol{\beta}^{t+1}|\boldsymbol{\gamma}^{t+1}, \boldsymbol{Z}^t)$ and $f(\boldsymbol{Z}^{t+1}|\boldsymbol{\gamma}^{t+1}, \boldsymbol{\beta}^{t+1})$, until getting the samples we want.

$$\int_{Z^t}\int_{\beta^t}\int_{\gamma^t} f(\boldsymbol{\gamma}^t, \boldsymbol{\beta}^t, \boldsymbol{Z}^t) f(\boldsymbol{\gamma}^{t+1}|\boldsymbol{\beta}^t, \boldsymbol{Z}^t) f(\boldsymbol{\beta}^{t+1}|\boldsymbol{\gamma}^{t+1}, \boldsymbol{Z}^t) f(\boldsymbol{Z}^{t+1}|\boldsymbol{\gamma}^{t+1}, \boldsymbol{\beta}^{t+1}) d\boldsymbol{\gamma}^t d\boldsymbol{\beta}^t d\boldsymbol{Z}^t$$
$$= f(\boldsymbol{\gamma}^{t+1}, \boldsymbol{\beta}^{t+1}, \boldsymbol{Z}^{t+1}), \qquad (1)$$

where $(\boldsymbol{\gamma}^t, \boldsymbol{\beta}^t, \boldsymbol{Z}^t)$ is the sample obtained in the tth iteration. The left-hand side of (1) gives the marginal distribution of $(\boldsymbol{\gamma}^{t+1}, \boldsymbol{\beta}^{t+1}, \boldsymbol{Z}^{t+1})$ under assumption that $(\boldsymbol{\gamma}^t, \boldsymbol{\beta}^t, \boldsymbol{Z}^t)$ is from

$f(\gamma, \boldsymbol{\beta}, \boldsymbol{Z})$. Hence, (1) means that if $(\boldsymbol{\gamma}^t, \boldsymbol{\beta}^t, \boldsymbol{Z}^t)$ is from $f(\gamma, \boldsymbol{\beta}, \boldsymbol{Z})$, then $(\boldsymbol{\gamma}^{t+1}, \boldsymbol{\beta}^{t+1}, \boldsymbol{Z}^{t+1})$ is also from $f(\gamma, \boldsymbol{\beta}, \boldsymbol{Z})$.

The computation scheme is to iterate $f(\boldsymbol{\gamma}^{t+1}|\boldsymbol{Z}^t)$, $f(\boldsymbol{\beta}^{t+1}|\gamma^{t+1}, \boldsymbol{Z}^t)$ and $f(\boldsymbol{Z}^{t+1}|\gamma^{t+1}, \boldsymbol{\beta}^{t+1})$ in Lee *et el.* (2003). This process satisfies the stationary property. But they iterate $f(\boldsymbol{\gamma}^{t+1}|\boldsymbol{Z}^t)$, $f(\boldsymbol{Z}^{t+1}|\gamma^{t+1}, \boldsymbol{\beta}^t)$ and $f(\boldsymbol{\beta}^{t+1}|\gamma^{t+1}, \boldsymbol{Z}^{t+1})$ in the part of their algorithm, which does not satisfy the stationary property. Hence, there is a contradiction. Unlike Lee *et al.* (2003), we divide the model of gene selection into two parts: one is gene selection over all genes, where we integrate $\boldsymbol{\beta}$ out to get $f(\gamma|\boldsymbol{Z})$ and $f(\boldsymbol{Z}|\gamma)$; and the other is to check the model adequacy by leave-one-out cross validation for more significant genes from the first part by the model of Albert and Chib (1993), where our unknowns are just $\boldsymbol{\beta}$ and $\boldsymbol{Z}$. We simulate the samples from $f(\boldsymbol{Z}|\boldsymbol{\beta})$ and $f(\boldsymbol{\beta}|\boldsymbol{Z})$. It could also be used to make future prediction. Finally, we apply the model to hereditary breast cancer data (22 samples and 3226 genes). The results are also different from theirs.

In the next section we illustrate the Gibbs sampler. Section 3 draws the model for gene selection. The computation algorithm that we modify for gene selection and prediction is in Section 4. Section 5 is the application to Hereditary breast cancer data. Finally, we give a conclusion and some future work in Section 6.

2. Illustrating the Gibbs Sampler

The Gibbs sampler is one kind of Markov Chain Monte Carlo (MCMC) method. One can refer to Gilks *et al.* (1996) for more detail. Here is just an abstract from Casella and George (1992).

The Gibbs sampler is a technique for generating random variables from a (marginal) distribution indirectly without having to calculate the density. For example, if we are interested in obtaining mean or variance of the marginal density

$$f(x)=\int\cdots\int f(x{,}y_{1,\cdots,}y_p)dy_1\cdots dy_p. \tag{2}$$

Perhaps the most natural and straightforward approach would be calculating $f(x)$ and using it to obtain the mean or variance. However, it may be possible that the integration in (2) is extremely difficult to perform, either analytically or numerically. Another case is that if $(\gamma_{1,}\gamma_{2,\cdots}\gamma_p)$ are the unknowns that we are interested in. Unfortunately, the explicit form of the joint distribution of $(\gamma_{1,}\gamma_{2,\cdots}\gamma_p)$ is very difficult to get or even if we obtain it, it is still difficult to simulate samples directly. In such cases, the Gibbs sampling method provides an alternative method.

The Gibbs sampler allows us effectively to generate samples $X_1,\cdots,X_m$ from $f(x)$ or $(\gamma^1_{1,\cdots,}\gamma^1_p),\cdots,(\gamma^m_1{}_{,\cdots,}\gamma^m_p)$ from $f(\gamma_{1,\cdots,}\gamma_p)$ without the exact form of $f(x)$ or $f(\gamma_{1,\cdots,}\gamma_p)$. After suitable burn-in period, we can obtain the samples as we want. And the mean and the variance of $f(x)$ can be calculated to the desired degree of accuracy by simulating a large enough sample.

To understand the Gibbs sampler, we explore it as the following case. Starting with a set of random variables $(\gamma_{1,\cdots,}\gamma_p)$, the Gibbs sampler generates samples from $f(\gamma_{1,\cdots,}\gamma_p)$ by sampling instead from the conditional distributions $f(\,\gamma_1\,|\,\gamma_{2,\cdots,}\gamma_p)$, $f(\,\gamma_2\,|\,\gamma_{1,}\gamma_{3,\cdots,}\gamma_p),\cdots,f(\,\gamma_p\,|\,\gamma_1{}_{,\cdots,}\gamma_{p-1})$, which are often known in statistical models or easy to simulate. This is done by generating a "Gibbs sequence" of random variables

$$(\gamma_1^{(0)}{}_{,\cdots,}\gamma_p^{(0)}),(\gamma_1^{(1)}{}_{,\cdots,}\gamma_p^{(1)}),(\gamma_1^{(2)}{}_{,\cdots,}\gamma_p^{(2)}),\cdots,(\gamma_1^{(k)}{}_{,\cdots,}\gamma_p^{(k)}). \tag{3}$$

The initial value $(\gamma_1^{(0)}{}_{,\cdots,}\gamma_p^{(0)})$ is specified, and the rest of (3) is obtained iteratively by alternately generating values from

$$\gamma_1^{(j+1)}\sim f(\gamma_1|(\gamma_2^{(j)}{}_{,\cdots,}\gamma_p^{(j)}))$$

$$
\begin{aligned}
\gamma_2^{(j+1)} &\sim f(\gamma_2|(\gamma_1^{(j+1)}, \gamma_3^{(j)}, ..., \gamma_p^{(j)})) \\
&\vdots \\
\gamma_{p-1}^{(j+1)} &\sim f(\gamma_{p-1}|(\gamma_1^{(j+1)}, ..., \gamma_{p-2}^{(j+1)}, \gamma_p^{(j)})) \\
\gamma_p^{(j+1)} &\sim f(\gamma_p|(\gamma_1^{(j+1)}, ..., \gamma_{p-1}^{(j+1)})). \qquad (4)
\end{aligned}
$$

We refer to this generation method, (3), as Gibbs sampling. The distribution of $(\gamma_1^{(k)}{}_{,...,}\gamma_p^{(k)})$ converges to the true joint distribution $f(\gamma_{1,...,}\gamma_p)$. Thus, for k large enough, the final observation in (3), namely $(\gamma_1^{(k)}{}_{,...,}\gamma_p^{(k)})$, is effectively a sample point from $f(\gamma_{1,...,}\gamma_p)$.

The convergence (in distribution) of Gibbs sequence (3) can be exploited in a variety of ways to obtain an approximate sample from $f(\gamma_{1,...,}\gamma_p)$. For example, Gelfand and Smith (1990) suggest generating m independent Gibbs sequences of length k, and then using the final value of $(\gamma_1^{(k)}{}_{,...,}\gamma_p^{(k)})$ from each sequence. Another way is to generate one long Gibbs sequence and then extract every r observations, that is to take the set of $\left\{(\gamma_1^{(k)}{}_{,...,}\gamma_p^{(k)}), (\gamma_1^{(k+r)}{}_{,...,}\gamma_p^{(k+r)}), (\gamma_1^{(k+2r)}{}_{,...,}\gamma_p^{(k+2r)}), \cdots\right\}$(see Geyer, 1991). We can also take $(\gamma_1^{(j)}{}_{,...,}\gamma_p^{(j)})$ as $j \geq k$, a less wasteful approach. For k and r large enough, the samples which we take would yield approximate samples from $f(\gamma_{1,...,}\gamma_p)$ in all cases.

Gibbs sampling can be used to estimate the density itself by averaging the final conditional densities from m Gibbs sequences. For each sequence from (3), we take ($\gamma_1^{(k)}{}_{,...,}\gamma_p^{(k)}$) as a realization of $\gamma_{1,...,}\gamma_p$ from $f(\gamma_{1,...,}\gamma_p)$. Hence, we have totally m samples from $f(\gamma_{1,...,}\gamma_p)$. Moreover, the average of the conditional densities $f(\gamma_i|\gamma_{1,...,}\gamma_{i-1,}\gamma_{i+1,...,}\gamma_p)$ will closely approximate to $f(\gamma_i)$, and $f(\gamma_i)$ can be estimated as

$$\hat{f}(\gamma_i) = \frac{1}{m}\sum_{t=1}^{m} f(\gamma_i|\gamma^t_{1,...,}\gamma^t_{i-1,}\gamma^t_{i+1,...,}\gamma^t_p), \qquad (5)$$

where $(\gamma^t_1, \cdots, \gamma^t_{j-1}, \gamma^t_{j+1}, \cdots, \gamma^t_p)$, $t = 1, \cdots, m$, is the sequence of realized values taken from Gibbs sequences. The theory behind the calculation in (5) is that the expected value of the conditional density is

$$
\begin{aligned}
&E[f(\gamma_i|\gamma_{1,\cdots,}\gamma_{i-1,}\gamma_{i+1,\cdots,}\gamma_p)] \\
&= \int_{\gamma_1}\cdots\int_{\gamma_{i-1}}\int_{\gamma_{i+1}}\cdots\int_{\gamma_p} f(\gamma_i|\gamma_{1,\cdots,}\gamma_{i-1,}\gamma_{i+1,\cdots,}\gamma_p) \\
&\times f(\gamma_{1,\cdots,}\gamma_{i-1,}\gamma_{i+1,\cdots,}\gamma_p)\, d\gamma_1\cdots d\gamma_{i-1}d\gamma_{i+1}\cdots d\gamma_p = f(\gamma_i), \qquad (6)
\end{aligned}
$$

a calculation mimicked by (5), since $(\gamma^1_{1,\cdots,}\gamma^1_{i-1,}\gamma^1_{i+1,\cdots,}\gamma^1_p),\cdots,(\gamma^m_{1},_{\cdots,}\gamma^m_{i-1,}\gamma^m_{i+1,\cdots,}\gamma^m_p)$ approximate a sample from $f(\gamma_{1,\cdots,}\gamma_{i-1,}\gamma_{i+1,\cdots,}\gamma_p)$.

3. Model for Gene Selection

Suppose there are n independent sample. For each sample i, $\boldsymbol{x}'_i = (x_{i1,\cdots,}x_{ip})$ is the data of gene expression levels, and Y_i is a binary response (normal or tissue), distributed Bernoulli with probability of success p_i. Then, we define the binary regression model as $p_i = H(\boldsymbol{x}'_i\boldsymbol{\beta})$, $i = 1,\cdots,n.$, where $\boldsymbol{\beta}$ is a $p\times 1$ vector of regression parameters and H is a known cdf linking the probabilities p_i with the linear structure $\boldsymbol{x}'_i\boldsymbol{\beta}$.

1896

In order to compute the exact posterior distribution of $\boldsymbol{\beta}$, Albert and Chib (1993) introduce a simulation-based approach. Let the link function H be the standard Gaussian cdf, then we can write the model as $p_i = Pr(Y_i = 1|\boldsymbol{\beta}) = \Phi(\boldsymbol{x}'_i\boldsymbol{\beta})$, $i = 1,\cdots,n$. The key idea is to employ n independent latent variables $Z_1,\cdots,Z_n$, where Z_i is distributed $N(\boldsymbol{x}'_i\boldsymbol{\beta},1)$, and define $Y_i = 1$ if $Z_i > 0$ and $Y_i = 0$ otherwise. Also, the latent variables has a normal linear model $\boldsymbol{Z} = \boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{\varepsilon}$, where $\boldsymbol{X} = (\boldsymbol{x}'_{1,\cdots,}\boldsymbol{x}'_n)'$ and $\boldsymbol{\varepsilon}$ is distributed $N_n(\boldsymbol{0},\boldsymbol{I})$. If we choose a multivariate prior for $\boldsymbol{\beta}$, then we can find the posterior distribution of $\boldsymbol{\beta}$ conditional on $\boldsymbol{Z}$ and the distribution of $\boldsymbol{Z}$ conditional on $\boldsymbol{\beta}$. It is therefore easy to simulate from both the marginal posterior distributions by Gibbs sampling algorithm.

As performing gene selection, an indicator variable $\boldsymbol{\gamma}' = (\gamma_{1,\cdots,}\gamma_p)$ is needed. We select the ith gene if γ_i =1 ($\beta_i \neq 0$); otherwise, it is not selected ($\beta_i = 0$). Given $\boldsymbol{\gamma}$, $\boldsymbol{\beta}_\gamma$ is a $q \times 1$ vector, consisting of all nonzero elements of $\boldsymbol{\beta}$, and $\boldsymbol{X}_\gamma$ is a $n \times q$ matrix with the columns of $\boldsymbol{X}$ corresponding to those $\gamma_i = 1$, where $q = \sum_{i=1}^{p} \gamma_i$. Moreover, we make the following prior assumptions:

1. The ith gene has a prior probability π_i being selected, where $0 \leq \pi_i \leq 1$, $i = 1, \cdots, p$. We can control the number of genes in the model by choosing different values of π_i. Also, if we have known that some genes are more important than others, we can assign larger values of π to it. Here we only consider the case that all π_i, $i = 1, \cdots, p$, are equal and there is no correlation between γ_i, $i = 1, \cdots, p$, which means whether the ith gene is selected or not, it does not effect the jth gene being selected, $j \neq i$. Let m be the total number of genes and π be the prior probability, the number of selected genes will be $m \times \pi$ on average.

2. Given $\boldsymbol{\gamma}$, the prior for $\boldsymbol{\beta}_\gamma$ is $N_q(\mathbf{0}, c(\boldsymbol{X}'_\gamma \boldsymbol{X}_\gamma)^{-1})$, where c is a positive scale factor specified by the user. Smith and Kohn (1996) found that the choice of c works well and the results are insensitive to values of c in the range $10 \leq c \leq 100$. We want to choose a value of c such that the prior of $\boldsymbol{\beta}_\gamma$, given $\boldsymbol{\gamma}$, contains very little information about $\boldsymbol{\beta}_\gamma$ compared to the likelihood. Therefore, we can take $c = 100$.

Moreover, there are two things we need to note. One is that making $\boldsymbol{\beta}$ diffuse by taking c infinite is impossible. Since it will lead to $p(\gamma_i = 1|\boldsymbol{Z}, \gamma_{j \neq i}) = 0$ for all i; see equation (B.8) in appendix B. The other is that we have to normalize $\boldsymbol{X}$ with mean zero for each column (each gene). This procedure will lead to the covariance matrix of the prior for $\boldsymbol{\beta}_\gamma$ is proportional to the inverse of the covariance of the data $\boldsymbol{X}_\gamma$. Then, if any two genes are highly correlated, one of their regression coefficients would be larger, and the other would be smaller. Otherwise, their regression coefficients would be independent. We also normalize $\boldsymbol{X}$ with variance one for each column (each gene).

4. Computation

In the above model, $(\boldsymbol{\gamma}, \boldsymbol{\beta}, \boldsymbol{Z})$ are the unknowns. Since the posterior distribution is difficult to get, we use the Gibbs sampling to generate these parameters from the posterior distribution.

It is impossible to simulate $(\boldsymbol{\gamma}, \boldsymbol{\beta}, \boldsymbol{Z})$ directly from the complete posterior distribution (see appendix A). Therefore, we integrate $\boldsymbol{\beta}$ $(\boldsymbol{\beta}_\gamma)$ out (see appendix B.), and then we draw $\boldsymbol{\gamma}$ and $\boldsymbol{Z}$ from the marginal distribution. So, our computation process is to draw $(\boldsymbol{\gamma}^{(t+1)}, \boldsymbol{Z}^{(t+1)})$ from $\boldsymbol{\gamma}^{(t+1)}|\boldsymbol{Z}^{(t)}$ and $\boldsymbol{Z}^{(t+1)}|\boldsymbol{\gamma}^{(t+1)}$. We divide the model of gene section into two parts : gene selection over all genes and leave-one-out cross validation. The advantage of our model is that we can take c = 100, which can not be infinity (see section 3), in the former part; and we take c = ∞ in the latter part since we have change the model by getting rid of $\boldsymbol{\gamma}$. But Lee *et al.* (2003) draw $(\boldsymbol{\gamma}^{(t+1)}, \boldsymbol{\beta}^{(t+1)}, \boldsymbol{Z}^{(t+1)})$ from $\boldsymbol{\gamma}^{(t+1)}|\boldsymbol{Z}^{(t)}$, $\boldsymbol{\beta}^{(t+1)}|\boldsymbol{\gamma}^{(t+1)}, \boldsymbol{Z}^{(t)}$ and $\boldsymbol{Z}^{(t+1)}|\boldsymbol{\gamma}^{(t+1)}, \boldsymbol{\beta}^{(t+1)}$. Unlike our model, they take c = 100 always.

4.1 Gene Selection

After integrating the $\boldsymbol{\beta}_\gamma$ out, we get the marginal distribution of $(\boldsymbol{Z}, \boldsymbol{\gamma})$. The computation scheme is as follows:

1. Draw $\boldsymbol{Z}$ from its marginal conditional distribution given $\boldsymbol{\gamma}$.

$$\begin{aligned} p(\boldsymbol{Z}|\boldsymbol{\gamma}) &\propto p(\boldsymbol{Y}|\boldsymbol{Z}) \times \int p(\boldsymbol{Z}|\boldsymbol{\beta}_\gamma) p(\boldsymbol{\beta}_\gamma) d\boldsymbol{\beta}_\gamma \\ &\propto \exp\left\{-\frac{1}{2}\boldsymbol{Z}'(\boldsymbol{I} - \frac{c}{1+c}\boldsymbol{X}_\gamma(\boldsymbol{X}'_\gamma\boldsymbol{X}_\gamma)^{-1}\boldsymbol{X}_\gamma)\boldsymbol{Z}\right\} \times p(\boldsymbol{Y}|\boldsymbol{Z}), \end{aligned} \tag{7}$$

where $p(\boldsymbol{Y}|\boldsymbol{Z}) = 1$ if $Y_i = 1$ and $Z_i > 0$ or $Y_i = 0$ and $Z_i < 0$ for all $i = 1, \cdots, n$; otherwise, it is equal to zero. Hence, the distribution of $\boldsymbol{Z}$ given γ is a multivariate normal distribution $N_n(\boldsymbol{0}, (\boldsymbol{I} - \frac{c}{1+c}\boldsymbol{X}_\gamma(\boldsymbol{X}'_\gamma\boldsymbol{X}_\gamma)^{-1}\boldsymbol{X}'_\gamma)^{-1})$ restricted to a subset $\mathtt{R}$ of $\boldsymbol{R}^n$, where $\mathtt{R} = (\mathtt{R}_1 \times \mathtt{R}_2 \times \ldots \times \mathtt{R}_n)$ and $\mathtt{R}_i = (0, \infty)$ if $Y_i = 1$; otherwise, $\mathtt{R}_i = (-\infty, 0)$, for $i = 1, \cdots, n$.

To generate the truncated multivariate normal samples, there are two ways. One is following Robert's (1995) method: using Gibbs sampling to get converging multivariate samples and the optimal exponential accept-reject algorithm to get each Z_i conditional on Z_j, $i \neq j$ for $i, j = 1, \cdots, n$. The other is using Metropolis-Hastings algorithm (Chib and Greenberg 1995), which was developed by Metropolis, Rosenbluth, Rosenbluth, Teller, and Teller (1953). This method is also one kind of Markov chain Monte Carlo methods. Moreover, it gives rise to the Gibbs sampler as a special case. We use the Gibbs sampling method in Section 5, and the burn-in period is 500.

2. Draw $\boldsymbol{\gamma}$ from $\boldsymbol{Z}$.

$$\begin{aligned} p(\boldsymbol{\gamma}|\boldsymbol{Z}) &\propto p(\boldsymbol{Z}|\boldsymbol{\gamma})p(\boldsymbol{\gamma}) \\ &\propto (1+c)^{-q_\gamma/2} \exp\left\{-\frac{1}{2}S(\gamma)\right\} \prod_{i=1}^{p} \pi_i^{\gamma_i}(1-\pi_i)^{1-\gamma_i}, \end{aligned} \tag{8}$$

where $q_\gamma = \sum \gamma_i$ and $S(\gamma) = \boldsymbol{Z}'\boldsymbol{Z} - \frac{c}{1+c}\boldsymbol{Z}'\boldsymbol{X}_\gamma(\boldsymbol{X}'_\gamma\boldsymbol{X}_\gamma)^{-1}\boldsymbol{X}'_\gamma\boldsymbol{Z}$. We can draw $\boldsymbol{\gamma}$ componentwise from $p(\gamma_i|\boldsymbol{Z}, \gamma_{j\neq i})$. Then,

$$\begin{aligned} p(\gamma_i|\boldsymbol{Z}, \gamma_{j\neq i}) &\propto p(\boldsymbol{Z}|\boldsymbol{\gamma})p(\gamma_i) \\ &\propto \pi_i^{\gamma_i}(1-\pi_i)^{1-\gamma_i}(1+c)^{-q_\gamma/2} \exp\left\{-\frac{1}{2}S(\gamma)\right\}. \end{aligned} \tag{9}$$

After suitable burn-in period (10,000 or 100,000 in section 5.1), we obtain the samples at the tth iteration: $\left\{\boldsymbol{Z}^t, \boldsymbol{\gamma}^t, t = 1, \cdots, m\right\}$. Then, calculate the total number appeared in the sample for each gene. We can make prediction by those genes with higher frequency.

4.2 Prediction

After getting the posterior frequency of each gene, we can select q genes with higher frequency, where $q \leq p$. Let $\boldsymbol{X}_q = (\boldsymbol{x}'_{1,\cdots,}\boldsymbol{x}'_n)'$, a $n \times q$ matrix, be the columns of $\boldsymbol{X}$ corresponding to those q genes, where x_i is a $q \times 1$ vector, for $i = 1, \cdots, n$, and let $\boldsymbol{\beta}_q$ be the regression parameters. Then we use probit regression model with n latent variables $(Z_{1,\cdots,}Z_n)$ to make prediction (Albert and Chib 1993). The computation schemes are as follows:

1. Given $\boldsymbol{\beta}_q$ and $\boldsymbol{Y}$, draw Z_i, $i = 1, \cdots, n$, from the following distribution,

$$\begin{array}{llll} Z_i|Y_i, \boldsymbol{\beta}_q \sim N(\boldsymbol{x}_i{}'\boldsymbol{\beta}_q, 1) & \textit{truncated at the left by } 0 & \textit{if } Y_i & = 1 \\ Z_i|Y_i, \boldsymbol{\beta}_q \sim N(\boldsymbol{x}_i{}'\boldsymbol{\beta}_q, 1) & \textit{truncated at the right by } 0 & \textit{if } Y_i & = 0 \end{array}$$

where $\boldsymbol{\beta}_q$, the regression parameters of $\boldsymbol{X}_q$, is a $q \times 1$ vector. It is a truncated normal distribution, so we can use Robert's (1995) optimal exponential accept-reject algorithm to generate Z_i.

2. Draw $\boldsymbol{\beta}_q$ conditional on $\boldsymbol{Y}$ and $\boldsymbol{Z}$. The prior for $\boldsymbol{\beta}_q$ is $N_q(\boldsymbol{0}, c(\boldsymbol{X}_q'\boldsymbol{X}_q)^{-1})$, where c is a positive scale factor specified by the user. We obtain that $\boldsymbol{\beta}_q|\boldsymbol{Y}, \boldsymbol{Z}$ is distributed $N_q(\boldsymbol{V}\boldsymbol{X}_q'\boldsymbol{Z}, \boldsymbol{V})$, where $\boldsymbol{V} = \frac{c}{1+c}(\boldsymbol{X}_q'\boldsymbol{X}_q)^{-1}$. If the prior distribution of $\boldsymbol{\beta}_q$ is diffuse (taking $c = \infty$), then $\boldsymbol{\beta}_q|\boldsymbol{Y}, \boldsymbol{Z}$ is a multivariate normal distribution with mean $(\boldsymbol{X}_q'\boldsymbol{X}_q)^{-1}(\boldsymbol{X}_q'\boldsymbol{Z})$ and covariance matrix $(\boldsymbol{X}_q'\boldsymbol{X}_q)^{-1}$.

The starting value of $\boldsymbol{\beta}_q$, $\boldsymbol{\beta}_q^{(0)}$ may be taken to be the least squares (LS) estimate $(\boldsymbol{X}_q'\boldsymbol{X}_q)^{-1}\boldsymbol{X}_q'\boldsymbol{Y}$. After suitable burn-in period ($k = 200$ in Section 5.2), we obtain the samples : $\left\{\boldsymbol{Z}^t, \boldsymbol{\beta}_q^t, t = 1, \cdots, m\right\}$. Then , we can estimate the posterior mean of $\boldsymbol{\beta}_q$ with $\frac{1}{m}\sum_{t=1}^{m}\boldsymbol{\beta}_q^t$. As coming with a new observation Y_{new}, whose gene expression levels $\boldsymbol{x}$ is a $p \times 1$ vector, we can predict it based on the probit model. Let $\boldsymbol{x}_q$ be the elements of $\boldsymbol{x}$ corresponding to the q genes we selected, then the probability of $Y_{new} = 1$ conditional on $\boldsymbol{x}$ is

$$P(Y_{new} = 1|\boldsymbol{x}) = \Phi(\boldsymbol{x}_q'\boldsymbol{\beta}_q) \tag{10}$$

5. Application to Hereditary Breast Cancer Data

We apply the above model to a published data set (Hedenfalk *et al.*, 2001). There are totally 22 tumor samples (n = 22) from 21 breast cancer patients : 7 tumors with BRCA1 mutations, 8 tumors with BRCA2 mutations and 7 sporadic tumors. For each sample, the gene size is 3226 (p = 3226). Here we give each sample a number as the collum order of the original data, $1, \cdots, 22$, obtained in *http : //research.nhgri.nih.gov/microarray/selected_publications.html* .

Some pathological features help us to distinguish these tumors. For tumors with BRCA1 mutations, there are higher mitotic index, pushing tumor margins and lymphocytic infiltrate. Moreover, BRCA1 tumors are generally negative for both estrogen and progesterone receptors, but tumors with BRCA2 mutations are positive for these hormone receptors and heterogeneous with substantially less tubule formation. These features imply different but overlapping functions for BRCA1 and BRCA2 tumors.

Now, we want to use the scheme we propose to classify BRCA1 (Y = 1) versus the others (BRCA2 and sporadic : Y = 0). First, we have to select some significant genes, and then to make prediction.

5.1 Gene selection

We control the size of selected genes to be about 10 on average by fixing $\pi_i = 0.003$, for $i = 1, \cdots, 3226$, and take $c = 100$. Before we run the Gibbs sampler, the data has to be normalized with mean zero and variance one for each gene. To be sure that the result of gene selection would be convergent, we generate two different Gibbs sequences with two starting values of $\boldsymbol{\gamma}$. One is to select 10 genes, which are in the 253th, 555th, 556th, 585th, 806th, 1068th, 1443th, 1999th, 3009th and 3013th rows of the original data, by the weight of support vector machines (SVMs) (Hastie *et al.*, 2001), whose image cloneID number are 28469, 548957, 212198, 293104, 46182, 840702, 566887, 247818, 366647 and 375922 respectively. For another starting values, we select arbitrarily 10 genes in the 8th, 19th, 22th, 23th, 44th, 50th, 56th, 60th, 70th and 100th rows of the original data, which have small correlation coefficients with the above 10 genes. Their image cloneID number are 25584, 30272, 31169, 32875, 42059, 43231, 44180, 45233, 51293 and 36393 respectively. We describe only the genes as their row numbers of the original data in the following. Table 1 gives the correlation coefficients of these 20 genes.

After a 10,000 burn-in period, we collect 330,000 samples of $\boldsymbol{\gamma}^{(t)}$ for both starting values. Among these 330,000 samples, the sizes of selected genes at each iteration are 7.29 (the first starting value) and 7.31 (the second starting value) on average . Table3 lists 10 most significant genes with the highest frequencies. Besides the result of collecting 330,000 $\boldsymbol{\gamma}^{(t)}$, we also delete the first 90,000 of 330,000 samples to collect the latest 240,000 samples. The results of the first starting value are shown in Table3. Similarly, Table4 is the result of the second starting value. Among these 240,000 samples of $\gamma^{(t)}$, 7.28 and 7.32 are the averaged size of selected genes in each iteration for both starting values respectively. We found that the result of gene selection is almost the same regardless of the starting values and the size of samples being deleted. Table2 lists the description of these genes.

5.2 Leave-One-Out Cross Validation

After the above process, some significant genes which can differentiate the two classes are obtained. From Table3 and Table4, the frequency of gene 1068 is higher than others clearly. Therefore, we select it to the model. As for the other genes, the first four genes except gene 1068 are considered, which are gene 3009, 2734, 1999 and 2761.

Since there are 22 samples in total, we use the method of leave-one-out cross validation to check the adequacy. As leaving the ith sample out as test data, $i = 1, \cdots, 22$, we first find the mean and variance of the other 21 samples, and then normalize these 22 samples.

We can not iterate too many times since the round-off error would become larger as the number of iteration increases in our programs. But iterating too few times would not

1896

converge. Therefore, we take $k = 200$ (burn-in period) and $m = 500$ after monitoring the convergence of several different Gibbs sequences. Then, there are 500 samples in each Gibbs sequence. We simulate totally 20 Gibbs sequences to obtain 10,000 samples. To get the posterior mean of $\boldsymbol{\beta}$, we average these samples. Repeat this process until we collect 40 averaged $\boldsymbol{\beta}$s and get 40 probabilities of $Y_i = 1$, for all i. We list their mean and standard deviation in Table5 to Table8.

To make prediction, we consider the five selected genes and all combinations of 2 genes and 3 genes among them, where gene 1068 has to be contained. If the selected genes are 1068 and 2761, the leave-one-out error is 0. Table5 lists the detail for its mean and standard deviation of 40 probabilities that Y_i is equal to 1. All of the other combinations of 2 genes have 1 leave-one-out error. Selection of 1068 & 1999 classifies sample16 to the wrong class, and the combination of 1068 & 2734 classify sample5 wrong. Gene 1068 & 3009 classify sample 1 wrong, too. Moreover, these samples have also error when they are being training data. For 3 genes, we describe the result of all combinations in Table6. Although there are 3 kinds of combinations which have both 0 error in test and training data, only the combination, 1068, 2761 and 3009, does not have any problem in all samples. Thus, we think it as the best fit, and list its detail in Table7. Finally, using all of the 5 genes gives us that the leave-one-out error is also 0. Table8 lists the detailed result of 5 genes.

Of course, we obtain a better fit on training set as the size of selected genes increasing. But what we care about is to classify samples to the right class by fewer genes. We therefore select genes whose size are no more than five to make prediction.

6. Conclusion and Future Work

Lee *et el.* (2003) proposed a Bayesian model for gene selection with binary data, and then used a hierarchical probit model and MCMC based stochastic search techniques to obtain the posterior samples. We modify their algorithm in gene selection and prediction, and avoid the disadvantage of taking $c = \infty$ always. The results of gene selection are different from their result, but its adequacy is still good. One drawback of our method is that we have to iterate much more times than theirs since our method is to deal with a multivariate case.

Sha (2002) had extended two categories of events to multi-category data. Here we assumed that the probability of each genes being selected is independent with each other. As Lee *et el.* suggested, we can extend it to dependent case. For example, we know that the jth gene will be expressed if the ith gene is expressed. Then we can change the prior distribution of $\boldsymbol{\gamma}$ and use a Markov model whose transition matrices will be defined as $p(\gamma_j = 1|\gamma_i = 1)$ or so.

Lee *et al.* (2003) also suggested to extend the model with fixed π value by allowing π to be an unknown model parameter and assigning a conjugate beta prior to it. If we have prior knowledge that the ith gene is more important than others , it is possible to assign larger values of π_i in a scale of importance from 0 to 1.

We can also consider other kind of linking function in our model, for example, the logit linking function,

$$log(\frac{p_i}{1-p_i}) = \boldsymbol{x}_i'\boldsymbol{\beta}. \tag{11}$$

Finally, we can find some ways to avoid the computational round-off error in the algorithm of prediction.

Appendix

A The Reason to Integrate $\boldsymbol{\beta}_\gamma$ Out

From the model for gene selection, we obtain the joint distribution of $\boldsymbol{\beta}, \boldsymbol{\gamma}$ and $\boldsymbol{Z}$, which is

$$p(\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{Z}) \propto p(\boldsymbol{\gamma}) p(\boldsymbol{\beta}|\boldsymbol{\gamma}, \boldsymbol{X}) p(\boldsymbol{Z}|\boldsymbol{\beta}, \boldsymbol{X}) p(\boldsymbol{Y}|\boldsymbol{Z}), \tag{A.1}$$

where $p(\boldsymbol{Y}|\boldsymbol{Z}) = 1$ if $Y_i = 1$ and $Z_i > 0$ or $Y_i = 0$ and $Z_i < 0$ for all $i = 1, \cdots, n$; otherwise, it is equal to zero. In order to draw the samples $(\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{Z})$, we use Gibbs sampling. The computation scheme will be as follows:

(i) Draw $\boldsymbol{\gamma}|\boldsymbol{\beta}, \boldsymbol{Z}$: the conditional distribution is

$$\begin{aligned} p(\boldsymbol{\gamma}|\boldsymbol{\beta}, \boldsymbol{Z}) &\propto p(\boldsymbol{\gamma}) p(\boldsymbol{\beta}|\boldsymbol{\gamma}, \boldsymbol{X}) = p(\boldsymbol{\gamma}) p(\boldsymbol{\beta}_\gamma) \\ &\propto \prod_{i=1}^{p} \pi_i^{\gamma_i} (1 - \pi_i)^{1-\gamma_i} \frac{1}{c^{q_\gamma/2} \left| \boldsymbol{X}_\gamma' \boldsymbol{X}_\gamma \right|^{-1/2}} \exp\left\{ -\frac{1}{2c} \boldsymbol{\beta}_\gamma' (\boldsymbol{X}_\gamma' \boldsymbol{X}_\gamma) \boldsymbol{\beta}_\gamma \right\}. \end{aligned} \tag{A.2}$$

We can draw it componentwise from $p(\gamma_i|\boldsymbol{\beta}, \boldsymbol{Z}, \gamma_{j\neq i})$ which is

$$\begin{aligned} p(\gamma_i|\boldsymbol{\beta}, \boldsymbol{Z}, \gamma_{j\neq i}) &\propto p(\gamma_i) p(\boldsymbol{\beta}_\gamma) \\ &\propto \pi_i^{\gamma_i} (1 - \pi)^{1-\gamma_i} \frac{1}{c^{q_\gamma/2} \left| \boldsymbol{X}_\gamma' \boldsymbol{X}_\gamma \right|^{-1/2}} \exp\left\{ -\frac{1}{2c} \boldsymbol{\beta}_\gamma' (\boldsymbol{X}_\gamma' \boldsymbol{X}_\gamma) \boldsymbol{\beta}_\gamma \right\}. \end{aligned} \tag{A.3}$$

(ii) Draw $\boldsymbol{Z}|\boldsymbol{\beta}, \boldsymbol{\gamma}$: the conditional distribution is

$$p(\boldsymbol{Z}|\boldsymbol{\beta}, \boldsymbol{\gamma}) \propto p(\boldsymbol{Z}|\boldsymbol{\beta}, \boldsymbol{X}) p(\boldsymbol{Y}|\boldsymbol{Z}) = \prod_{i=1}^{n} p(Z_i|\boldsymbol{\beta}, \boldsymbol{x}_i) p(Y_i|Z_i)$$

Hence, the full conditional distribution of Z_i is as follows:

$$\begin{cases} Z_i|\boldsymbol{\beta}, Y_i = 1 \propto N(\boldsymbol{x}_i'\boldsymbol{\beta}, 1) & \textit{truncated at the left by } 0 \\ Z_i|\boldsymbol{\beta}, Y_i = 0 \propto N(\boldsymbol{x}_i'\boldsymbol{\beta}, 1) & \textit{truncated at the right by } 0 \end{cases} \tag{A.4}$$

(iii) Draw $\boldsymbol{\beta}|\boldsymbol{\gamma}, \boldsymbol{Z}$, which is equivalent to draw $\boldsymbol{\beta}_\gamma|\boldsymbol{Z}$: the conditional distribution is the same as (B.9).

Since the model we have is $\gamma_i = 0$ if $\beta_i = 0$ and $\gamma_i = 1$ if $\beta_i \neq 0$, we always get $\gamma_i^{(new)} = 0$ if $\beta_i^{(old)} = 0$ and $\gamma_i^{(new)} = 1$ if $\beta_i^{(old)} \neq 0$ in step (i). Then, we can not implement the gene selection scheme. Therefore, if we integrate $\boldsymbol{\beta}$ ($\boldsymbol{\beta}_\gamma$) out in step (i) and (ii), we can solve the problem.

B Derivation of the Marginal Conditional Distributions

Since $Z_i \sim N(\boldsymbol{x}_i'\boldsymbol{\beta}, 1), i = 1, 2, \cdots, n$, are independent, the distribution of $\boldsymbol{Z}$ given $\boldsymbol{\gamma}$, $\boldsymbol{\beta}_\gamma$ and $\boldsymbol{X}_\gamma$ is $N_n(\boldsymbol{X}_\gamma\boldsymbol{\beta}_\gamma, \boldsymbol{I})$. Also, the prior for $\boldsymbol{\beta}_\gamma$ is $N_q(\boldsymbol{0}, c(\boldsymbol{X}_\gamma'\boldsymbol{X}_\gamma)^{-1})$, where $q = \sum\limits_{i=1}^{p} \gamma_i$. By (A.1), the marginal conditional distribution of $\boldsymbol{\beta}$ and $\boldsymbol{Z}$ given $\boldsymbol{\gamma}$ is

$$p(\boldsymbol{\beta}, \boldsymbol{Z}|\boldsymbol{\gamma}) \propto p(\boldsymbol{\beta}|\boldsymbol{\gamma}, \boldsymbol{X})p(\boldsymbol{Z}|\boldsymbol{\beta}, \boldsymbol{X})p(\boldsymbol{Y}|\boldsymbol{Z}) = p(\boldsymbol{\beta}_\gamma)p(\boldsymbol{Z}|\boldsymbol{\beta}_\gamma)p(\boldsymbol{Y}|\boldsymbol{Z}),$$

where $p(\boldsymbol{Y}|\boldsymbol{Z}) = 1$ if $Y_i = 1$ and $Z_i > 0$ or $Y_i = 0$ and $Z_i < 0$ for all $i = 1, \cdots, n$; otherwise, it is equal to zero. Now, we derive $p(\boldsymbol{Z}|\boldsymbol{\beta}_\gamma)p(\boldsymbol{\beta}_\gamma)$.

$$\begin{aligned} p(\boldsymbol{Z}|\boldsymbol{\beta}_\gamma)p(\boldsymbol{\beta}_\gamma) &\propto \exp\left\{-\frac{1}{2}(\boldsymbol{Z} - \boldsymbol{X}_\gamma\boldsymbol{\beta}_\gamma)'(\boldsymbol{Z} - \boldsymbol{X}_\gamma\boldsymbol{\beta}_\gamma)\right\} \\ &\quad \times \frac{1}{c^{q_\gamma/2}\left|\boldsymbol{X}_\gamma'\boldsymbol{X}_\gamma\right|^{-1/2}} \exp\left\{-\frac{1}{2c}\boldsymbol{\beta}_\gamma'(\boldsymbol{X}_\gamma'\boldsymbol{X}_\gamma)\boldsymbol{\beta}_\gamma\right\} \\ &= \frac{1}{c^{q_\gamma/2}\left|\boldsymbol{X}_\gamma'\boldsymbol{X}_\gamma\right|^{-1/2}} \exp\left\{-\frac{1}{2}(1 + c^{-1})\boldsymbol{\beta}_\gamma'(\boldsymbol{X}_\gamma'\boldsymbol{X}_\gamma)\boldsymbol{\beta}_\gamma + \boldsymbol{Z}'\boldsymbol{X}_\gamma\boldsymbol{\beta}_\gamma\right\} \exp\left\{-\frac{1}{2}\boldsymbol{Z}'\boldsymbol{Z}\right\}, \end{aligned} \tag{B.1}$$

where $q_\gamma = \sum \gamma_i$. If we let $\boldsymbol{V}_\gamma = (1+c^{-1})^{-1}(\boldsymbol{X}'_\gamma\boldsymbol{X}_\gamma)^{-1}$ and $\boldsymbol{\beta}_0 = \boldsymbol{V}_\gamma\boldsymbol{X}'_\gamma\boldsymbol{Z} = (1+c^{-1})^{-1}(\boldsymbol{X}'_\gamma\boldsymbol{X}_\gamma)^{-1}\ \boldsymbol{X}'_\gamma\boldsymbol{Z}$. then (B.1) can be rewritten as

$$\begin{aligned}
p(\boldsymbol{Z}|\boldsymbol{\beta}_\gamma)p(\boldsymbol{\beta}_\gamma) &\propto \exp\left\{-\frac{1}{2}\boldsymbol{\beta}'_\gamma\boldsymbol{V}_\gamma^{-1}\boldsymbol{\beta}_\gamma + \boldsymbol{\beta}'_0\boldsymbol{V}_\gamma^{-1}\boldsymbol{\beta}_\gamma - \frac{1}{2}\boldsymbol{\beta}'_0\boldsymbol{V}_\gamma^{-1}\boldsymbol{\beta}_0\right\} \\
&\qquad \times\frac{1}{c^{q_\gamma/2}\left|\boldsymbol{X}'_\gamma\boldsymbol{X}_\gamma\right|^{-1/2}}\exp\left\{-\frac{1}{2}\boldsymbol{Z}'\boldsymbol{Z} + \frac{1}{2}\boldsymbol{\beta}'_0\boldsymbol{V}_\gamma^{-1}\boldsymbol{\beta}_0\right\} \\
&= \exp\left\{-\frac{1}{2}(\boldsymbol{\beta}_\gamma-\boldsymbol{\beta}_0)'\boldsymbol{V}_\gamma^{-1}(\boldsymbol{\beta}_\gamma-\boldsymbol{\beta}_0)\right\}\frac{1}{c^{q_\gamma/2}\left|\boldsymbol{X}'_\gamma\boldsymbol{X}_\gamma\right|^{-1/2}}\exp\left\{-\frac{1}{2}\boldsymbol{Z}'\boldsymbol{Z} + \frac{1}{2}\boldsymbol{\beta}'_0\boldsymbol{V}_\gamma^{-1}\boldsymbol{\beta}_0\right\} \\
&\propto \exp\left\{-\frac{1}{2}(\boldsymbol{\beta}_\gamma-\boldsymbol{\beta}_0)'\boldsymbol{V}_\gamma^{-1}(\boldsymbol{\beta}_\gamma-\boldsymbol{\beta}_0)\right\}\frac{|\boldsymbol{V}_\gamma|^{1/2}}{c^{q_\gamma/2}\left|\boldsymbol{X}'_\gamma\boldsymbol{X}_\gamma\right|^{-1/2}}\exp\left\{-\frac{1}{2}\boldsymbol{Z}'\boldsymbol{Z} + \frac{1}{2}\boldsymbol{\beta}'_0\boldsymbol{V}_\gamma^{-1}\boldsymbol{\beta}_0\right\} \\
&= \exp\left\{-\frac{1}{2}(\boldsymbol{\beta}_\gamma-\boldsymbol{\beta}_0)'\boldsymbol{V}_\gamma^{-1}(\boldsymbol{\beta}_\gamma-\boldsymbol{\beta}_0)\right\}(1+c)^{-q_\gamma/2}\exp\left\{-\frac{1}{2}\boldsymbol{Z}'\boldsymbol{Z} + \frac{1}{2}\boldsymbol{\beta}'_0\boldsymbol{V}_\gamma^{-1}\boldsymbol{\beta}_0\right\}, \qquad \text{(B.2)}
\end{aligned}$$

where $\boldsymbol{Z}'\boldsymbol{X}_\gamma\boldsymbol{\beta}_\gamma = \boldsymbol{\beta}'_0\boldsymbol{V}_\gamma^{-1}\boldsymbol{\beta}_\gamma$, and $\frac{|\boldsymbol{V}_\gamma|^{1/2}}{c^{q_\gamma/2}\left|\boldsymbol{X}'_\gamma\boldsymbol{X}_\gamma\right|^{-1/2}} = (1+c)^{-q_\gamma/2}$.

To integrate $\boldsymbol{\beta}$ ($\boldsymbol{\beta}_\gamma$) out, we obtain

$$\begin{aligned}
p(\boldsymbol{Z}|\boldsymbol{\gamma}) &\propto p(\boldsymbol{Y}|\boldsymbol{Z}) \times \int p(\boldsymbol{Z}|\boldsymbol{\beta}_\gamma)p(\boldsymbol{\beta}_\gamma)d\boldsymbol{\beta}_\gamma \\
&\propto (1+c)^{-q_\gamma/2}\exp\left\{-\frac{1}{2}S(\gamma)\right\} \times p(\boldsymbol{Y}|\boldsymbol{Z}) \\
&\propto \exp\left\{-\frac{1}{2}\boldsymbol{Z}'(\boldsymbol{I} - \frac{c}{1+c}\boldsymbol{X}_\gamma(\boldsymbol{X}'_\gamma\boldsymbol{X}_\gamma)^{-1}\boldsymbol{X}_\gamma)\boldsymbol{Z}\right\} \times p(\boldsymbol{Y}|\boldsymbol{Z}), \qquad \text{(B.3)}
\end{aligned}$$

where $S(\gamma) = \boldsymbol{Z}'\boldsymbol{Z} - \boldsymbol{\beta}'_0\boldsymbol{V}_\gamma^{-1}\boldsymbol{\beta}_0 = \boldsymbol{Z}'\boldsymbol{Z} - \frac{c}{1+c}\boldsymbol{Z}'\boldsymbol{X}_\gamma(\boldsymbol{X}'_\gamma\boldsymbol{X}_\gamma)^{-1}\boldsymbol{X}'_\gamma\boldsymbol{Z}$.

The distribution of $\boldsymbol{\gamma}|\boldsymbol{Z}$ is

$$p(\boldsymbol{\gamma}|\boldsymbol{Z}) \propto p(\boldsymbol{Z}|\boldsymbol{\gamma})p(\boldsymbol{\gamma})$$

$$\propto (1+c)^{-q_\gamma/2}\exp\left\{-\frac{1}{2}S(\gamma)\right\}\prod_{i=1}^{p}\pi_i^{\gamma_i}(1-\pi_i)^{1-\gamma_i}. \tag{B.4}$$

Then,

$$\begin{aligned} p(\gamma_i|\boldsymbol{Z},\gamma_{j\neq i}) &\propto p(\boldsymbol{Z}|\boldsymbol{\gamma})p(\gamma_i) \\ &\propto \pi_i^{\gamma_i}(1-\pi_i)^{1-\gamma_i}(1+c)^{-q_\gamma/2}\exp\left\{-\frac{1}{2}S(\gamma)\right\}. \end{aligned} \tag{B.5}$$

Since

$$p(\gamma_i=1|\boldsymbol{Z},\gamma_{j\neq i}) \propto \pi_i(1+c)^{-q_{\gamma^1}/2}\exp\left\{-\frac{1}{2}S(\gamma^1)\right\} \tag{B.6}$$

$$p(\gamma_i=0|\boldsymbol{Z},\gamma_{j\neq i}) \propto \pi_i(1+c)^{-q_{\gamma^0}/2}\exp\left\{-\frac{1}{2}S(\gamma^0)\right\}, \tag{B.7}$$

where $\gamma^1=(\gamma_1,\cdots,\gamma_i=1,\cdots,\gamma_p)$ and $\gamma^0=(\gamma_1,\cdots,\gamma_i=0,\cdots,\gamma_p)$,

$$p(\gamma_i=1|\boldsymbol{Z},\gamma_{j\neq i}) = \frac{1}{1+\frac{(B.7)}{(B.6)}} = \frac{1}{1+\frac{1-\pi_i}{\pi_i}(1+c)^{1/2}\exp\left\{-\frac{1}{2}[S(\gamma^0)-S(\gamma^1)]\right\}} \tag{B.8}$$

Since $p(\boldsymbol{\beta}_\gamma|\boldsymbol{Z}) \propto p(\boldsymbol{Z}|\boldsymbol{\beta}_\gamma)p(\boldsymbol{\beta}_\gamma)$, by (B.2) , we have

$$p(\boldsymbol{\beta}_\gamma|\boldsymbol{Z}) \propto \exp\left\{-\frac{1}{2}(\boldsymbol{\beta}_\gamma-\boldsymbol{\beta}_0)'\boldsymbol{V}_\gamma^{-1}(\boldsymbol{\beta}_\gamma-\boldsymbol{\beta}_0)\right\}. \tag{B.9}$$

Hence, the posterior distribution of $\boldsymbol{\beta}_\gamma$ is $\boldsymbol{\beta}_\gamma|\boldsymbol{Z} \sim N_q(\boldsymbol{\beta}_0,\boldsymbol{V}_\gamma)$.

References

Albert, J. and Chib, S. (1993). "Bayesian analysis of binary and polychotomous response data". *Journal of American Statistical Association* 88, pp. 669-679.

Casella, G. and George, E. I. (1992). "Explaining the Gibbs Sampler". *The American Statistician* 46, pp. 167-174.

Chib, S. and Greenberg, E. (1995). "Understanding the Metropolis-Hastings Algorithm". *The American Statistician* 49, pp. 327-335.

Gelfand, A. E. and Smith, A. F. M. (1990). "Sampling-Based Approaches to Calculating Marginal Densities". *Journal of the American Statistical Association* 85, pp. 398-409.

Geyer, C. J. (1991). "Markov Chain Monte Carlo Maximum Likelihood". *Computer Science and Statistics* 23. pp. 156-163

Gilks, W., Richardson, S., and Spiegelhalter,D. (1996).*Markov Chain Monte Carlo in practise.* Chapman and Hall, London.

Hedenfalk, I., Duggan, D., Chen, Y., Radmacher, M., Bittner, M., Simon, R., Meltzer, P., Gusterson, B., Esteller, M., Raffeld, M. *et al.*, (2001). "Gene expression profiles in hereditary breast cancer". *The New England Journal of Medicine* 344, pp. 539-548.

Hastie, T., Tibshirani, R. and Friedman, J. (2001). *The Elements of Statistical Learning.* Springer, London.

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

Kim, S., Dougherty, E. R., Barrera, J., Chen, Y., Bittner, M. and Trent, J. M. (2002). "Strong feature sets from small samples". *Journal of Computational Biology* 9. pp. 127-146

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. and Teller, E. (1953). "Equations of State Calculations by Fast Computing Machines". *Journal of Chemical Physics* 21, pp. 1087-1092.

Robert, C. (1995). "Simulation of truncated normal variables". *Statistics and Computing* 5, pp. 121-125.

Sha, N. (2002). *Bolstering CART and Bayesian variable selection methods for classification*, PH.D. thesis, Department of Statistics, Texas A&M University.

Smith, M., and Kohn, R. (1996). "Nonparametric regression using Bayesian variable selection". *Journal of Econometrics* 75, pp. 317-344.

Table1. The correlation coefficients of the genes between the two starting values of γ ($\rightarrow$: the first starting value ; $\downarrow$: the second starting value)

p	253	555	556	585	806	1068	1443	1999	3009	3013
8	0.0333	-0.2633	-0.1320	0.3481	-0.1951	-0.2192	0.0058	0.1036	-0.1150	-0.1580
19	-0.2782	0.1747	-0.0900	-0.3780	-0.1280	-0.0058	0.2356	0.0266	0.2147	0.2317
22	0.2481	-0.2492	0.2954	0.1030	-0.2415	-0.2333	-0.1464	-0.1928	-0.0687	0.0855
23	0.3492	-0.0368	0.3180	0.0491	-0.2392	-0.1310	-0.3713	-0.3935	0.0199	0.0089
44	-0.0451	-0.1533	-0.1760	-0.2278	0.1723	-0.4236	0.2843	-0.1925	-0.0488	0.1380
50	-0.1264	-0.6029	0.0471	0.0960	-0.1734	-0.0264	0.4177	0.0690	-0.1326	0.3199
56	-0.1000	-0.3118	-0.1860	-0.0529	0.2447	0.2735	0.2909	0.3084	0.3310	-0.1850
60	0.0355	-0.1657	-0.0040	0.0585	-0.1753	-0.0977	0.2585	-0.144	0.1034	0.0978
70	-0.0170	-0.2002	-0.0960	0.1952	-0.0245	-0.2882	0.2914	0.1384	0.1532	-0.3010
100	0.0487	0.3566	0.0799	-0.0827	-0.0669	0.0092	-0.2858	0.0040	0.3823	-0.1960

Table2. The description of the 10 most significant genes

# row	Clone ID	Gene descreption
963	897646	splicing factor, arginine/serine-rich 4
1068	840702	SELENOPHOSPHATE SYNTHETASE ; Human selenium donor protein
1277	73531	nitrogen fixation cluster-like
1620	137638	ESTs
1859	307843	ESTs; eukaryotic translation initiation factor 2C, 2*
1999	247818	ESTs; Homo sapiens cDNA FLJ13495 fis, clone PLACE1004425*
2423	26082	very low density lipoprotein receptor
2734	46019	minichromosome maintenance deficient (S. cerevisiae) 7
2761	47884	macrophage migration inhibitory factor (glycosylation-inhibiting factor)
3009	366647	butyrate response factor 1 (EGF-response factor 1)

* the descriptions in Kim *et al.* (2002)

Table3. 10 most significant genes with the first starting value of γ

delete 10,000; collect 330,000		delete 100,000; collect 240,000	
# row	Frequency* (%)	# row	Frequency* (%)
1068	6.487575758	1068	6.100000000
3009	4.249090909	3009	4.406666667
2734	3.983939394	2734	4.367500000
1999	3.872424242	1999	3.806250000
2761	3.175151515	2423	3.163333333
2423	2.887575758	2761	2.950000000
1620	2.517575758	1859	2.697083333
1859	2.498181818	963	2.582916667
963	2.474848485	1620	2.579583333
1277	1.993636364	1277	1.861250000

*Percentage of times the genes appeared in the samples

Table4. 10 most significant genes with the first starting value of γ

delete 10,000; collect 330,000		delete 100,000; collect 240,000	
# row	Frequency (%)	# row	Frequency (%)
1068	7.296666667	1068	7.695000000
3009	4.317878788	3009	4.322916667
2734	3.408181818	2734	3.793750000
1999	3.306969697	1999	3.183750000
2761	2.932727273	2761	2.957500000
2423	2.570909091	1859	2.420000000
1859	2.468787879	2423	2.392916667
1277	2.226666667	1277	2.194583333
963	2.084242424	963	2.128333333
1620	2.070000000	1620	1.939166667

Table5. The mean and standard deviation of 40 probabilities (Y_i = 1) for 2 genes, which are 1068 and 2761.

real Y	1	0	0	0
sample #	1	7	8	10
leave # out	mean (std.)	mean (std.)	mean (std.)	mean (std.)
1	***0.99993 (0.00015)***	0.000094 (0.00019)	0.0780 (0.0239)	0.00004 (0.000090)
2	0.99999 (0.00001)	0.000118 (0.00010)	0.1386 (0.0239)	0.00005 (0.000047)
3	1.00000 (0)	0.000868 (0.00064)	0.4873 (0.0260)	0.00041 (0.000376)
4	1.00000 (0.00001)	0.000120 (0.00013)	0.1617 (0.0298)	0.00006 (0.000073)
5	0.99999 (0.00002)	0.000155 (0.00016)	0.1143 (0.0270)	0.00006 (0.000064)
6	1.00000 (0)	0.000419 (0.00033)	0.3703 (0.0244)	0.00024 (0.000205)
7	0.99981 (0.00013)	***0.000012 (0.00001)***	0.0299 (0.0113)	0* (0.000005)
8	0.99984 (0.00022)	0.000044 (0.00008)	***0.0580 (0.0191)***	0.00002 (0.000046)
9	0.99904 (0.00147)	0.000016 (0.00008)	0.0112 (0.0086)	0.00001 (0.000032)
10	0.99977 (0.00023)	0.000013 (0.00002)	0.0239 (0.0104)	***0* (0.000010)***
11	0.99964 (0.00040)	0.000006 (0.00001)	0.0138 (0.0084)	0* (0.000004)
12	0.99977 (0.00017)	0.000021 (0.00002)	0.0333 (0.0102)	0* (0.000007)
13	0.99947 (0.00045)	0.000005 (0.00001)	0.0121 (0.0065)	0* (0.000003)
14	0.99941 (0.00055)	0.000006 (0.00001)	0.0115 (0.0056)	0* (0.000002)
15	0.99952 (0.00037)	0.000005 (0.00001)	0.0123 (0.0049)	0 (0)
16	0.99976 (0.00029)	0.000048 (0.00008)	0.0371 (0.0146)	0.00001 (0.000027)
17	0.99976 (0.00025)	0.000036 (0.00005)	0.0364 (0.0141)	0.00001 (0.000015)
18	0.99996 (0.00004)	0.000054 (0.00006)	0.1398 (0.0275)	0.00004 (0.000049)
19	0.99927 (0.00055)	0.000002 (0.00001)	0.0082 (0.0048)	0* (0.000002)
20	0.99978 (0.00015)	0.000019 (0.00002)	0.0256 (0.0093)	0* (0.000005)
21	0.99964 (0.00028)	0.000006 (0.00001)	0.0164 (0.0091)	0* (0.000003)
22	0.99972 (0.00036)	0.000010 (0.00002)	0.0197 (0.0095)	0* (0.000008)

0* means that its value is less than 10^{-5}.

If we leave 1 out as test data, the first element of its row is 1. The other samples are as the set of training data. Thus, all probabilities of this row are training probability except the probability which has a line under it.

All samples which we do not list here have perfect fit regardless of being training or test, whose probabilities are either 0 or 1 corresponding to their real responses. Moreover, their variance is zero.

Table5. (Continued) The mean and standard deviation of 40 probabilities (Y_i = 1) for 2 genes, which are 1068 and 2761.

real Y	0	0	0	1	0
sample #	12	16	17	18	20
Leave # out	mean (std.)	mean (std.)	mean (std.)	mean (std.)	mean (std.)
1	0.0008 (0.0011)	0.109 (0.023)	0.014 (0.008)	0.998 (0.0023)	0.002 (0.003)
2	0.0011 (0.0007)	0.144 (0.020)	0.019 (0.006)	0.998 (0.0012)	0.003 (0.002)
3	0.0078 (0.0034)	0.460 (0.014)	0.094 (0.014)	1.000 (0*)	0.013 (0.005)
4	0.0011 (0.0008)	0.140 (0.016)	0.016 (0.006)	0.998 (0.0016)	0.002 (0.001)
5	0.0015 (0.0011)	0.174 (0.020)	0.026 (0.009)	0.999 (0.0006)	0.005 (0.003)
6	0.0033 (0.0018)	0.265 (0.022)	0.039 (0.011)	1.000 (0.0002)	0.003 (0.002)
7	0.0002 (0.0001)	0.063 (0.012)	0.006 (0.002)	0.996 (0.0029)	0.001 (0.001)
8	0.0004 (0.0005)	0.072 (0.014)	0.008 (0.004)	0.993 (0.0054)	0.001 (0.001)
9	0.0002 (0.0004)	0.046 (0.015)	0.005 (0.004)	0.995 (0.0056)	0.001 (0.002)
10	0.0002 (0.0002)	0.067 (0.016)	0.007 (0.004)	0.997 (0.0029)	0.002 (0.001)
11	0.0001 (0.0002)	0.054 (0.015)	0.005 (0.003)	0.998 (0.0029)	0.001 (0.001)
12	*0.0003 (0.0002)*	0.073 (0.013)	0.008 (0.003)	0.997 (0.0021)	0.002 (0.001)
13	0.0001 (0.0001)	0.048 (0.014)	0.005 (0.003)	0.997 (0.0029)	0.001 (0.001)
14	0.0001 (0.0001)	0.049 (0.016)	0.005 (0.003)	0.997 (0.0025)	0.002 (0.001)
15	0.0001 (0.0001)	0.044 (0.013)	0.004 (0.002)	0.996 (0.0023)	0.001 (0.001)
16	0.0006 (0.0006)	*0.097 (0.021)*	0.013 (0.007)	0.998 (0.0022)	0.004 (0.003)
17	0.0004 (0.0004)	0.083 (0.020)	*0.010 (0.005)*	0.997 (0.0026)	0.003 (0.002)
18	0.0004 (0.0003)	0.059 (0.014)	0.005 (0.002)	*0.977 (0.0137)*	0.000 (0.000)
19	0.0001 (0.0001)	0.042 (0.012)	0.004 (0.002)	0.997 (0.0028)	0.001 (0.001)
20	0.0003 (0.0002)	0.092 (0.023)	0.012 (0.006)	0.999 (0.0014)	*0.004 (0.004)*
21	0.0001 (0.0001)	0.053 (0.013)	0.005 (0.003)	0.996 (0.0037)	0.001 (0.001)
22	0.0001 (0.0002)	0.054 (0.014)	0.005 (0.003)	0.997 (0.0031)	0.001 (0.001)

0* means that its value is less than 10^{-4}.

Table6. The result of all combinations for 3 genes.

3 genes	leave-one-out test error	train error
1068 2761 3009	***0*** ; all probabilities are almost 0 or 1, and their std. are almost 0. Except sample8,11, 14,16,17,20,21, the others have perfect fits.	***0*** ; all probabilities are almost 0 or 1, and their std. are almost 0. Except sample6,8, 11,14,16,17,20,21, the others have perfect fits.
1068 2734 3009	***1*** ; sample 5 is .0176(. 0166), which is classified wrong; sample1 is .7082 (.1084) and sample14 is .4537(.1896), which are both bad prediction. The others are almost perfect.	***0*** ; all probabilities are close to 0 or 1, and their std. are close to 0. Sample2~4,6,8,9, 11~13,18,19,22 have perfect fits.
1068 1999 3009	***0*** ; sample5 is .8341(.1636); sample16 is .3184(.0907).The others are perfect both in mean and std. except sample7 and 15, which are very close to 0.	***0*** ; All samples have perfect fits except sample7,12,14~17,20,21. Especially, the prob. of sample16 is between .0287 and .3282 with std. smaller than .0448.
1068 2734 2761	***0*** ; sample1 is .9381(.0388); sample18 is .986(.0124). Except that sample1, 7~10, 16~18, 20 are close to 0 or 1 with std. close to 0 , the others are perfect.	***0*** ; Except sample1,7,8,10,16~18,20 are close to 0 or 1 with std. close to 0, the others have perfect fits.
1068 1999 2761	***0*** ; sample16 is .3533(.0807), which is a bad prediction. Except that sample6, 7, 9, 12, 16 ~18 are close to 0 or 1 with std. close to 0, the others are almost perfect.	***1*** ; Except sample1,6, 7, 9, 12, 16, 17,20 close to 0 or 1 with std. close to 0, the others have perfect fits. The prob. of sample16 is between .0633 and .5653 with std. smaller than .0342.
1068 1999 2734	***1*** ; The error occurred in sample1 with prob. .4283 (.0634). Sample7 is .4049 (.01435), a bad training. As for the others, all are either 0 or 1 with std. zero except sample7,10,15~17 are close	***0*** ; Except sample1 and sample7, the others are very close to 0 or 1. The prob. of sample1 is between .5921 and .9229 with std. smaller than .03995. Sample7 is between .0129 and .0801 with std. smaller

to 0 in both prob. and std than .0276.

Table7. The mean and standard deviation of 40 probabilities (Y_i = 1) for gene 1068, 2761, and 3009.

real Y	1	0	0	0
sample #	6	8	11	14
leave # out	mean (std.)	mean (std.)	mean (std.) $(\times 10^{-7})$ $(\times 10^{-6})$	mean (std.) $(\times 10^{-7})$ $(\times 10^{-6})$
1	1 (0)	0.00069 (0.00094)	32.5 (12.90)	0 (0)
2	1 (0)	0.00066 (0.00068)	42.5 (11.50)	2.5 (1.60)
3	1 (0)	0.00100 (0.00103)	12.5 (5.200)	0 (0)
4	1 (0)	0.00121 (0.00149)	72.5 (23.90)	2.5 (1.60)
5	0.9781 (0.064)	0 (0)	0 (0)	35.0 (11.7)
6	***1 (0)***	0.00126 (0.00230)	70.0 (28.80)	2.5 (1.60)
7	1 (0)	0.00016 (0.00022)	7.5 (2.600)	0 (0)
8	1 (0)	***0.00046 (0.00061)***	35.0 (10.20)	0 (0)
9	1 (0)	0.00003 (0.00004)	12.5 (7.900)	0 (0)
10	1 (0)	0.00009 (0.00014)	52.5 (28.60)	0 (0)
11	1 (0)	0.00064 (0.00062)	***3190.0 (983.1)***	0 (0)
12	1 (0)	0.00006 (0.00006)	10.0 (3.800)	0 (0)
13	1 (0)	0.00005 (0.00007)	10.0 (4.400)	0 (0)
14	1 (0)	0.00011 (0.00018)	30.0 (12.20)	***2.5 (1.60)***
15	1 (0)	0.00016 (0.00026)	7.5 (3.500)	0 (0)
16	1 (0)	0.00024 (0.00028)	7.5 (2.700)	0 (0)
17	1 (0)	0.00018 (0.00022)	25.0 (7.800)	0 (0)
18	1 (0)	0.00183 (0.00188)	10.0 (3.800)	0 (0)
19	1 (0)	0.00004 (0.00006)	0 (0)	0 (0)
20	1 (0)	0.00015 (0.00019)	25.0 (9.300)	0 (0)
21	1 (0)	0.00011 (0.00015)	2.5 (1.600)	0 (0)
22	1 (0)	0.00009 (0.00009)	32.5 (12.30)	0 (0)

Table7. (Continued) The mean and standard deviation of 40 probabilities (Y_i = 1) for gene 1068, 2761, and 3009.

real Y	0	0	0	0
sample #	16	17	20	21
leave # out	mean (std.)	mean (std.) $(\times10^{-6})$ $(\times10^{-6})$	mean (std.) $(\times10^{-6})$ $(\times10^{-6})$	mean (std.) $(\times10^{-6})$ $(\times10^{-6})$
1	0.00236 (0.0018)	72.8 (90.80)	0 (0)	4.3 (11.30)
2	0.00198 (0.0014)	48.3 (61.00)	0 (0)	6.5 (18.60)
3	0.00912 (0.0074)	173.0 (309.0)	0.3 (1.58)	34.3 (41.00)
4	0.00241 (0.0021)	70.5 (94.10)	0 (0)	12.3 (22.80)
5	0.00237 (0.0016)	159.3 (161.0)	13.3 (33.4)	123.0 (147.6)
6	0.00359 (0.0017)	78.0 (61.30)	0 (0)	19.0 (25.40)
7	0.00024 (0.0003)	4.3 (10.70)	0 (0)	1.0 (3.700)
8	0.00044 (0.0004)	8.0 (14.90)	0 (0)	0.5 (2.200)
9	0.00007 (0.0001)	0.5 (2.200)	0 (0)	0 (0)
10	0.00014 (0.0002)	2.0 (7.200)	0 (0)	0.3 (1.600)
11	0.00008 (0.0001)	0.5 (2.200)	0 (0)	1.8 (4.500)
12	0.00008 (0.0001)	0 (0)	0 (0)	0.3 (1.600)
13	0.00004 (0.0000)	0 (0)	0 (0)	0 (0)
14	0.00033 (0.0006)	8.5 (26.40)	0 (0)	2.8 (8.500)
15	0.00022 (0.0004)	3.8 (13.70)	0 (0)	0.3 (1.600)
16	***0.00047 (0.0005)***	11.0 (20.00)	0 (0)	1.0 (3.000)
17	0.00027 (0.0003)	***4.0 (9.800)***	0 (0)	1.8 (4.500)
18	0.00272 (0.0021)	68.0 (76.50)	0 (0)	0 (0)
19	0.00008 (0.0001)	0.8 (2.700)	0 (0)	0 (0)
20	0.00018 (0.0003)	3.0 (14.40)	***0 (0)***	0.5 (2.200)
21	0.00031 (0.0004)	7.0 (17.40)	0 (0)	***0.3 (1.600)***
22	0.00011 (0.0001)	0.3 (1.600)	0 (0)	0.5 (2.200)

Table8. The mean and standard deviation of 40 probabilities (Y_i = 1) for 5 genes, which are 1068, 1999, 2734, 2761 and 3009.

real Y	0	0	0	0
sample #	7	14	15	16
leave # out	mean (std.)	mean (std.)	mean (std.)	mean (std.)
1	0.000510 (0.000921)	0 (0.000002)	0 (0)	0.000058 (0.000072)
2	0.000069 (0.000166)	0 (0)	0* (0.000002)	0.000287 (0.000614)
3	0.000017 (0.000028)	0.000001 (0.000003)	0 (0)	0.000361 (0.000549)
4	0.000161 (0.000294)	0.000001 (0.000003)	0 (0)	0.000115 (0.000177)
5	0.000002 (0.000005)	0.000007 (0.000023)	0* (0.000002)	0.000059 (0.000090)
6	0.000120 (0.000187)	0.000001 (0.000004)	0 (0)	0.000098 (0.000137)
7	***0.001257 (0.002861)***	0 (0)	0 (0)	0.000028 (0.000071)
8	0.000025 (0.000044)	0 (0)	0 (0)	0.000023 (0.000040)
9	0.000007 (0.000018)	0 (0)	0 (0)	0.000002 (0.000006)
10	0.000022 (0.000083)	0 (0)	0 (0)	0.000006 (0.000015)
11	0.000009 (0.000017)	0 (0)	0 (0)	0.000004 (0.000009)
12	0.000012 (0.000020)	0 (0)	0 (0)	0.000004 (0.000012)
13	0.000008 (0.000017)	0 (0)	0 (0)	0.000003 (0.000009)
14	0.000021 (0.000064)	***0.000006 (0.000029)***	0 (0)	0.000004 (0.000011)
15	0.000056 (0.000133)	0 (0)	***0* (0.000002)***	0.000016 (0.000026)
16	0.000080 (0.000113)	0 (0)	0 (0)	***0.000349 (0.000515)***
17	0.000023 (0.000039)	0 (0)	0 (0)	0.000001 (0.000003)
18	0.000023 (0.000036)	0 (0)	0 (0)	0.000046 (0.000052)
19	0.000013 (0.000024)	0 (0)	0 (0)	0.000013 (0.000050)
20	0.000013 (0.000023)	0 (0)	0 (0)	0.000001 (0.000002)
21	0.000043 (0.000082)	0 (0)	0 (0)	0.000021 (0.000045)
22	0.000020 (0.000060)	0 (0)	0 (0)	0.000004 (0.000013)

0* means that its value is less than 10^{-6}.