Bayesian Analysis (2009) 4, Number 3, pp. 413–436

A Stochastic Partitioning Method to Associate High-dimensional Responses and Covariates

Stefano Monni* and Mahlet G. Tadesse†

Abstract. We consider the problem of variable selection in data sets with many response variables and many covariates. A method is proposed that allows some covariates to affect some response variables and not others, and that clusters responses which have similar dependence on the same set of covariates. A Markov chain Monte Carlo procedure is employed to sample from the space of pairwise partitions of covariates and outcomes, where a pair consists of a subset of all outcomes and their associated covariates. We assess the performance of the method on simulated data and apply it to genomic data.

Keywords: multivariate model selection, mixture model, Markov chain Monte Carlo, parallel tempering, CGH analysis

1 Introduction

Large data sets with thousands of high-dimensional variables collected on few experimental units have become common in many applications. Several procedures have been proposed to relate these data to univariate outcomes and identify relevant predictors. Methods that are particularly useful when the number of regressors is substantially larger than the sample size include the Bayesian stochastic search variable selection (SSVS) technique (George and McCulloch 1997) and the elastic net procedure (Zou and Hastie 2005). However, with $q > 1$ outcomes, these methods can only be applied by fitting each outcome independently, which gives rise to q separate models. The correlation structure of the outcomes is thus necessarily ignored. In addition, even when doing separate univariate variable selection appears to be an appropriate scheme, it can quickly become impracticable for large q. In the context of variable selection, one of the few methods of which we are aware that deals with few outcomes simultaneously is that of Brown et al. (1998), which extends the SSVS algorithm to a multivariate setting. In the latter paper, the same regressors are selected for all outcomes. This is reasonable only if the number of outcomes is small, which is the case in the examples therein considered. In the presence of many outcomes, one should however expect some covariates to affect some response variables and not others. One can try to overcome this limitation by first clustering the outcomes and then applying Brown et al. (1998)'s multivariate regression method on the clustered outcomes, one cluster at a time. This approach is justifiable but it has some shortcomings as we will demonstrate in Section 4.1. On more general grounds, this two-stage procedure would treat the clusters

*Department of Public Health, Weill Cornell Medical College, New York, NY, mailto:stm2013@med.cornell.edu

†Department of Mathematics, Georgetown University, Washington, DC, mailto:mgt26@georgetown.edu

 DOI:10.1214/09-BA416

as known, would ignore the uncertainty in estimating the cluster memberships when searching for relevant predictors, and thus would inevitably introduce some bias into the analysis (Bryant and Williamson 1978).

In this paper we are interested in cases where not only is the number of regressors much larger than the sample size but so too is the number of responses. This important problem has not received the attention it deserves, especially in light of the multivariate nature of the data with which one is usually confronted. The method we propose allows for each response to be determined by its own covariates, and at the same time identifies outcomes that are acted upon by the same covariates. Namely, we present a stochastic algorithm that searches for sets of covariates associated with sets of correlated outcomes. This is achieved by constructing a Markov chain in the space of pairwise partitions of the set of regressors into (possibly) non-disjoint subsets and of the set of responses into disjoint subsets. Each element of a partition is a pair of subsets, one composed of covariates and the other composed of their correlated outcomes; with the additional requirement that each outcome should belong to one and only one pair. To improve mixing and reach convergence more rapidly, we also implement parallel tempering. We apply the procedure to a genomic data set, to examine the relationship between comparative genomic hybridization (CGH) profiles from 261 clones and mRNA expression levels from 3291 probe sets, and locate DNA copy number variations associated with changes in mRNA transcript levels.

Various partition methods have previously appeared in the literature to relate covariates with a univariate response. For example, in tree models (Breiman et al. 1984; Chipman et al. 1998), the basic idea is to partition the coordinate space by hyperplanes (see also Holmes et al. (2005) for similar partition methods in Bayesian decision theory). Here instead we partition the index space of the variables and consider multivariate responses. A recent model along these lines was proposed by Lau and Green (2007) who considered the problem of clustering multivariate outcomes in the presence of a small number of covariates. Their method focuses on determining the optimal partition of the response variables using all regressors and is not concerned with identifying cluster specific covariates, which is instead our goal.

Our proposed model combines the ideas of mixture models, regression models, and variable selection to identify group structures and key relationships in high-dimensional data sets. This model was motivated by genomic applications, where there is a growing effort to relate gene expression data with other genome-wide technologies to better understand regulatory mechanisms. We rely on the common premise that genes with similar expression profiles share similar regulatory mechanisms. Thus, co-expressed genes would be co-regulated and share the same regression relationship, whereas genes in different clusters would have different regression models. Although this formulation may be too simple to capture adequately the complex underlying biological processes, which is the case for most models, it can provide some insights. As all methods for high-dimensional data analysis, this is an exploratory technique that can help investigate and discover important features in the data.

The paper is organized as follows. Section 2 describes the model and presents details

on the prior specification. In Section 3 the Markov chain is defined and strategies for its implementation are discussed. In Section 4 the method is tested on some simulated data and its robustness to deviations from the assumptions of normality and linearity is examined. Further, the method is compared with other methods, and applied to genomic data. Section 5 concludes the paper with a brief discussion.

2 Proposed method

2.1 Model formulation

Our data consist of N independent samples with p covariates, $\mathcal{X} = (X_1, \ldots, X_p)$, and q outcomes $\mathcal{Y} = (Y_1, \ldots, Y_q)$. As we have outlined in the previous section, in order to identify sets of outcomes that have the same dependence on a set of predictors, we consider partitions of the variables into sets of pairs $\mathcal{S} = (X_I, Y_J)$, with $I \subset \{1, \ldots p\}$ and $J \subset \{1, \ldots q\}$. A partition of the data will be henceforth referred to as a configuration, its pairs as the components of the configuration and the number of the latter as the length of the configuration. Each response Y_j is assigned to one and only one component, whereas a predictor X_r may belong to many components. The rationale for this choice, which is inherent in the asymmetric role of predictors and responses, is the possibility that a given predictor may affect different subsets of $\mathcal{Y}$ differently. If one views the model as a form of clustering, this is tantamount to considering non-intersecting clusters of responses. As a simplification, when we do not need to make the variables composing each component explicit, we label the latter by the count of its variables and write the partition accordingly. For example, the following configuration of length K

$$\mathcal{S}_1 \oplus \ldots \oplus \mathcal{S}_K = (X_{I_1}, Y_{J_1}) \oplus \ldots \oplus (X_{I_K}, Y_{J_K})$$

will be also written as

$$(|I_1|, |J_1|) \oplus \ldots \oplus (|I_K|, |J_K|),$$

where $0 \leq |I_k| \leq p$, $0 \leq |J_k| \leq q$, and $\sum_{k=1}^{K} |J_k| = q$. The $\oplus$ symbol is our way of indicating that the union of variables is disjoint for the Y and not necessarily disjoint for the X variables.

On the space of partitions we associate a probability function to the response variables. The probability of each configuration will be defined as the product of the probabilities of its components, because we assume the outcomes in distinct components of a given partition to be independent.

The model we consider is a multivariate Gaussian mixture model with an unknown number of components, in which the mean and the scale of each component are determined by a regression model on a subset of predictors. Namely, the distribution of the outcomes $Y_{t_1}, \ldots, Y_{t_{n_k}}$ of a component $\mathcal{S}_k = (m_k, n_k)$ is assumed to be:

$$Y_{ji} | \mathcal{S}_k \overset{iid}{\sim} \mathcal{N}(\alpha_j + \mu_k, \sigma_k^2), \qquad j = t_1, \ldots, t_{n_k}, \qquad i = 1, \ldots, N.$$

We write the location of the distribution as the sum of two terms α_j and $\mu_k = g_k(X_{s_1}, \ldots, X_{s_{m_k}})$ to emphasize that possible dispersions in the baselines α_j are ir-

relevant and that responses are clustered together if they are similarly affected by the same predictors. Responses associated with the same set of predictors can however belong to different components if their dependence on these predictors is dissimilar. We are therefore fitting a mixture of regression models, where the effects of the regressors on the response variables are the same within a component, but vary from one component to another (Jiang and Tanner 1999). The component means, μ_k, instead of being estimated based on the Y variables alone, as is commonly done in Gaussian mixture models, are estimated based on a linear regression model that captures the association between the Ys and the covariates X. We are also simultaneously performing variable selection to identify the relevant regressors for each component.

If we write the regression model as:

$$Y_{ji} = \alpha_j + \sum_{r=1}^{m_k} \beta_{ks_r} \cdot X_{s_r i} + \epsilon_{ji}, \qquad \epsilon_{ji} \sim \mathcal{N}(0, \sigma_k^2), \tag{1}$$

the likelihood function for $\mathcal{S}_k = (m_k, n_k)$ is given by:

$$\phi(m_k, n_k) = C \cdot \exp\left\{ -\frac{1}{2\sigma_k^2} \sum_{i=1}^{N} \sum_{j=1}^{n_k} \left[Y_{t_j i} - \alpha_{t_j} - \sum_{r=1}^{m_k} \beta_{ks_r} \cdot X_{s_r i} \right]^2 \right\},$$

with the normalizing constant being $C = (2\pi\sigma_k^2)^{-n_k N/2}$.

In a component of type $(0, f)$, which corresponds to having no regressor associated with f response variables, the Y_{ji} are distributed as $\mathcal{N}(\alpha_j, \sigma_k^2)$. In standard model-based clustering, the clusters are what in our model would be $(0, n)$ components. However, contrary to standard model-based clustering where variables are clustered if they are centered around a common mean and exhibit similar variability, we group variables only on the basis of the scale of their distributions, irrespectively of the location. This is similar to fitting a standard model-based clustering on centered data, where all components would have mean 0 and be distinguished by their variances only. The likelihood function of such a component is therefore given by:

$$\phi(0, f) = (2\pi\sigma_k^2)^{-fN/2} \exp\left\{ -\frac{1}{2\sigma_k^2} \sum_{i=1}^{N} \sum_{j=1}^{f} (Y_{t_j i} - \alpha_{t_j})^2 \right\}.$$

Notice that in the above model components of type $(v, 0)$ are equivalent to components of type $(1, 0)$ since X is viewed as a fixed covariate matrix and the corresponding functions are $\phi(1, 0) = 1 = \phi(v, 0)$. For this reason we only allow components $(1, 0)$ to be present in a configuration, and focus on $(0, n)$ and (m, n) components with $n > 0$ and $m > 0$. Put it differently, one can view the variable space as q copies of each of the p X variables, and a copy of the q Y variables and consider partitions with components $(1, 0)$, (m, n), with $m \geq 0$ and $n > 0$ with the only requirement that no variable be present more than once in each component.

2.2 Prior specification

Let us now specify the prior probability distributions for the model parameters. We take conjugate priors and exploit the conjugacy for computational efficiency by integrating out the component parameters. Let $\theta_k^T = (\alpha_{t_1}, \ldots, \alpha_{t_{n_k}}, \beta_{s_1}, \ldots, \beta_{s_{m_k}})$ be the (n_k+m_k)-vector of regression coefficients. We choose:

$$\begin{aligned} \theta_k &\sim \mathcal{N}(\theta_{0k}, H_0\sigma_k^2) \\ \sigma_k^2 &\sim \mathcal{IG}(\sigma_0^2, \nu) \end{aligned}$$

where $\theta_{0k}^T = (\alpha_{0t_1}, \ldots, \alpha_{0t_{n_k}}, \beta_{0s_1}, \ldots, \beta_{0s_{m_k}})$, $H_0 = \text{diag}(h_0\mathbf{1}_{n_k}, h\mathbf{1}_{m_k})$ with $\mathbf{1}_n$ an n-vector with all components equal to one, and $\mathcal{IG}$ the inverse-gamma distribution. H_0 controls the strength of the prior information on the regression coefficients with larger values of h_0 and h corresponding to a wider spread around θ_{0k}. After integrating out the model parameters, the marginalized likelihood for a component (m_k, n_k) reduces to an n_kN-dimensional multivariate t-distribution with $2\sigma_0^2$ degrees of freedom, mean $W\theta_{0k}$, and scale $\left(WH_0W^T + I_{n_kN\times n_kN}\right)\nu/\sigma_0^2$, where $W = (\mathbf{1}_N \otimes I_{n_k\times n_k} \qquad X^T \otimes \mathbf{1}_{n_k})$ is an $n_kN \times (n_k + m_k)$ matrix of covariates:[1]

$$\begin{aligned} f(m_k, n_k) &= \frac{\Gamma(\frac{2\sigma_0^2+n_kN}{2})}{\Gamma(\sigma_0^2)}(2\pi\sigma_0^2)^{-n_kN/2}\left\{\det\left(\frac{\nu}{\sigma_0^2}(WH_0W^T + I)\right)\right\}^{-\frac{1}{2}} \\ &\times \left\{1 + \frac{1}{2\sigma_0^2}(Y - W\theta_0)^T\left[\frac{\nu}{\sigma_0^2}(WH_0W^T + I)\right]^{-1}(Y - W\theta_0)\right\}^{-\frac{n_kN+2\sigma_0^2}{2}} \end{aligned} \tag{2}$$

This can also be written (see the Appendix) as:

$$\begin{aligned} f(m_k, n_k) &= f(Y|X) = \int \phi(Y|X, \alpha, \beta, \sigma^2)p(\alpha|\sigma^2)p(\beta|\sigma^2)p(\sigma^2)d\alpha d\beta d\sigma^2 \\ &= \frac{\nu^{\sigma_0^2}}{\left(\nu + \frac{1}{2}\Omega\right)^{\sigma_0^2+Nn_k/2}}\frac{\Gamma(\sigma_0^2 + Nn_k/2)}{\Gamma(\sigma_0^2)}(2\pi)^{-n_kN/2}h^{-m_k/2}(Nh_0 + 1)^{-n_k/2} \\ &\times (\det A)^{-1/2}, \end{aligned} \tag{3}$$

where the $m_k \times m_k$ matrix A is

$$A_{rs} = \frac{\delta_{rs}}{h} + n_k \cdot (X \cdot X^T)_{rs} - \frac{n_kh_0}{Nh_0 + 1}\sum_i X_{ri}\sum_i X_{si}, \tag{4}$$

[1] In this subsection, X and Y are the submatrices of the covariate and outcome matrices corresponding to the covariates and outcomes that are present in the component.

with δ_{rs} being the Kronecker delta, the scalar

$$\Omega = \sum_{r=1}^{m_k} \frac{\beta_{0s_r}^2}{h} + \sum_{j=1}^{n_k} \frac{\alpha_{0t_j}^2}{h_0} + \sum_{ij} Y_{ji}^2 - \frac{h_0}{Nh_0+1} \sum_{j=1}^{n_k} \left(\frac{\alpha_{0t_j}}{h_0} + \sum_{i=1}^{N} Y_{ji} \right)^2 - V^T A^{-1} V,$$

and the m_k-vector V is

$$V_r = \frac{\beta_{0r}}{h} + \sum_{ij} Y_{ji} X_{ri} - \frac{h_0}{(Nh_0+1)} \left(\sum_j \frac{\alpha_{0j}}{h_0} + \sum_{ij} Y_{ji} \right) \sum_i X_{ri}.$$

From the formulae above, one can thus see that the correlation among the outcomes in a component is accounted for.

Owing to the independence of the components in our model, the marginalized likelihood of a configuration is the product of the marginalized likelihoods of its components.

Finally, we assign a prior to each configuration

$$p((m_1, n_1) \oplus \ldots \oplus (m_K, n_K)) \propto \prod_{k=1}^{K} \rho^{m_k \cdot n_k} \tag{5}$$

with $0 < \rho \leq 1$. Thus, *a priori*, large components are penalized, with stronger penalty as ρ decreases. We try to favor smaller components because larger ones tend to fit the noise. We have experimented with different choices of configuration priors and found this to be the best at imposing some penalty without being too restrictive. As we show with explicit simulations in Section 4, where we also give a criterion to select ρ, this prior allows us to identify large components if there is a true signal for them.

3 Model fitting and posterior inference

3.1 MCMC implementation

An exhaustive evaluation of the posterior probabilities of all possible configurations is unfeasible. The total number of possibilities for components that are of type (m, n), with $n > 0$, is $\sum_{k=1}^{q} S_2(q, k) 2^{p \cdot k}$, where S_2 are the Stirling numbers of the second kind (see *e.g.* Abramowitz and Stegun (1972)). To sample from the probability measure, we thus construct a Markov chain whose unique stationary distribution is the measure of interest: starting at a random point in the configuration space, we move with probability to an adjacent point. The transition between two adjacent configurations is implemented by merging two components of the configuration or by splitting one component into two. In order to ensure better mixing among both the regressors and the response variables, we implement the Markov chain as a two-step process. At each step either a merge or a split move is randomly chosen. The types of components involved in the moves differ in the two steps: in the first step, only moves that allow creation (merging) of

$(1,0)$ components are considered, while in the second step the components to be split (merged) are sampled from the pool of (m,n) components with $n>1$ ($n\geq 1$).

Moves of the first type

Among all components (m,n) in the configuration, with $m>0, n>0$, whose number we denote by $|(+,+)|$, we randomly select one component and remove one of its X variables. In our notation,

$$(m,n)\rightarrow(m-1,n)\oplus(1,0).$$

In the reverse move we choose one component of type (m,n) with $0\leq m<p$ and $n>0$ and add to it a covariate uniformly selected among the $p-m$ covariates not present in the component

$$(1,0)\oplus(m,n)\rightarrow(m+1,n).$$

We use the Metropolis acceptance function (Metropolis et al. 1953) and thus the split move is accepted with probability

$$P_{split}=\min\left\{1,\quad \lambda_s^{(1)}\cdot\frac{f(m-1,n)}{f(m,n)}\cdot\frac{1}{\rho^n}\right\},$$

where

$$\lambda_s^{(1)}=\frac{m\cdot(|(+,+)|)}{(K-|(1,0)|-|(p,+)|')\cdot(p-m+1)},$$

and the merge move is accepted with probability

$$P_{merge}=\min\left\{1,\quad \lambda_m^{(1)}\cdot\frac{f(m+1,n)}{f(m,n)}\cdot\rho^n\right\},$$

where

$$\lambda_m^{(1)}=\frac{(p-m)\cdot(K-|(1,0)|-|(p,+)|)}{(m+1)\cdot(|(+,+)|')}.$$

The λs are the ratios of the kernels that describe the probability of going from one configuration to the other; K is the length of the initial configuration; $|(p,+)|$ ($|(p,+)|'$) is the number of components with p Xs and at least one Y before (after) the move; $|(1,0)|$ ($|(1,0)|'$) is the number of components with one X and no Ys before (after) the move.

Moves of the second type

In this case a random component (m,n) having $n\geq 2$ is split into two components:

$$(m,n)\rightarrow(m_1,n_1)\oplus(m_2,n_2),$$

with $n_1 > 0$ and $n_2 > 0$, and $m_1 + m_2 = m + c$, where c is chosen uniformly in the interval $[0, m/2]$ when m is even and in the interval $[0, (m-1)/2]$ when m is odd. The X variables are assigned to the new components by assigning c of the m X variables to both components to account for the intersection of the subsets of the new components, and the remaining $m - c$ variables to one or the other component with probability $1/2$. As for Y, n_1 $(1 \leq n_1 \leq n-1)$ of the n variables are randomly selected and placed in one component and the remaining $n_2 = n - n_1$ are allocated to the second component. The split move is accepted with probability

$$P_{split} = \min\left\{1, \quad \lambda_s^{(2)} \cdot \frac{f(m_1, n_1) \cdot f(m_2, n_2)}{f(m, n)} \cdot \rho^{m_1 \cdot n_1 + m_2 \cdot n_2 - m \cdot n}\right\},$$

where

$$\lambda_s^{(2)} = \frac{(n-1) \cdot (m/2+1) \cdot \binom{n}{n_1} \cdot \binom{m}{c} \cdot 2^{m-c+1} \cdot (K - |(1,0)| - |(+,1)| - |(0,1)|)}{(K+1-|(1,0)|) \cdot (K - |(1,0)|)}.$$

The probability for the move that merges two components containing Y variables

$$(m_1, n_1) \oplus (m_2, n_2) \to (m_1 + m_2 - c, n_1 + n_2) = (m, n),$$

where c is the number of X variables shared by both components, is then

$$P_{merge} = \min\left\{1, \quad \lambda_m^{(2)} \cdot \frac{f(m_1 + m_2 - c, n_1 + n_2)}{f(m_1, n_1) \cdot f(m_2, n_2)} \cdot \rho^{m \cdot n - m_2 \cdot n_2 - m_1 \cdot n_1}\right\},$$

where

$$\lambda_m^{(2)} = \frac{(K - |(1,0)|) \cdot (K - |(1,0)| - 1)}{(n-1) \cdot (m/2+1) \cdot \binom{n}{n_1} \cdot \binom{m}{c} \cdot 2^{m-c+1} \cdot (K - 1 - |(1,0)| - |(0,1)|' - |(+,1)|')},$$

The notations are as above.

3.2 A Tempering Extension

Simplicity of the moves is the most apparent feature enjoyed by the Markov chain defined in the previous subsection. In addition, the marginalization over the model parameters substantially accelerates the MCMC implementation by eliminating the need of defining appropriate re-allocations of the parameters at every step and the demand of updating them from their posterior distributions. One may wonder if those choices hinder the efficiency of the chain to explore the large configuration space, especially those regions associated with the dominant contributions of the measure. In fact, the trade-off between simplicity and efficiency is typical of all methods and is not a limitation of Markov chain methods only. Fortunately, there exist suitable Monte Carlo techniques that improve considerably the efficiency of the chain by increasing its mixing and limiting the possibility for it to be trapped in restricted regions corresponding to local modes of the probability density. One such method is parallel tempering (in the statistics

literature see *e.g.* Geyer (1991)), which we have implemented. A sequence of R Markov chains is run in the same configuration space with different stationary distributions $\psi_i(x) = \psi(x)^{1/T_i}$, where $1 = 1/T_0 > \ldots > 1/T_{R-1} > 0$ and $\psi(x) = \psi_0(x)$ is the posterior distribution from which we want to sample, which is $\psi(C) = f(C) \cdot p(C)$ for a configuration C. The higher the index of a chain (i.e. the larger its parameter T) the easier it is to jump from one configuration to the next in that chain. After a fixed number of updates in the respective Markov chains, a swap move is proposed that exchanges neighboring T values. The swap that involves the configurations at T_i and T_{i+1} is accepted with probability

$$P(C(T_{i+1}) \leftrightarrow C(T_i)) = \min\left\{1, \left(\frac{f(C(T_{i+1}))}{f(C(T_i))}\frac{p(C(T_{i+1}))}{p(C(T_i)))}\right)^{1/T_i - 1/T_{i+1}}\right\},$$

so that the move will always be accepted if the posterior of the configuration at T_{i+1} is larger than that of the configuration at T_i. Therefore configurations with largest posterior probabilities tend to move toward the distribution from which we shall eventually sample but the higher acceptance rates in the peripheral chains allow a better and faster mixing. The number and spacing of the $1/T_i$ are chosen so as to ensure good acceptance rates for the swaps. It is also important to make sure that exchanges do not only occur locally between neighboring T_i's but also between low and high values. In our applications, we verified this by checking that each chain spent (roughly) the same amount of time at each temperature. As a consequence, we can be reasonably confident that the sampler is able to escape local modes of the probability density, and therefore the diagnostic tools we used, *viz.* trace-plots of the log-posterior probabilities of visited models, similarity of marginal and pairwise posterior probability values estimated from sets of non-correlated samples, were supporting convergence rather than localization around some local mode.

4 Applications

In this section we apply the method to several data sets. First, we consider some simulated data, evaluate the performance of the method under various settings, assess its sensitivity to the choice of hyperparameters, test its robustness to the normality and linearity assumptions, and compare it with other methods. Then, we apply the method to genomic data. In our MCMC runs, we have recorded the maximum *a posteriori* (MAP) configuration. However, as there may be some degeneracy, *viz.* there may be different configurations with the same (or very similar) posterior probabilities, we have also considered, the $p \times q$ matrix of posterior probabilities of association between a covariate X_i and an outcome Y_j, the $p \times p$ matrix of posterior probabilities that any pair (X_i, X_j) is assigned to the same component (m, n) and the two $q \times q$ matrices of posterior probabilities that any pair (Y_i, Y_j) is allocated to the same $(0, n)$ or (m, n) component. The contributions of different configurations are thus averaged over, and different configurations with similar high posterior probabilities are weighted similarly.

4.1 Simulated data

Sensitivity analysis

A data set with $p = 200$ covariates, $\mathcal{X}$, and $q = 100$ responses, $\mathcal{Y}$, was generated for $N = 50$ samples. A configuration with $K = 10$ components was constructed in the following way. The Y variables were randomly assigned to these 10 components, with the vector $(n_1, \ldots, n_{10})$ that counts how many outcomes are in each component sampled from a multinomial distribution. Subsets of X with varying sizes between 1 and 25 were randomly assigned to 7 components, allowing for overlap between the components. The resulting configuration is listed in Table 1. The 200×50 entries of the matrix X were sampled from a multivariate normal distribution with covariance $\mathrm{cov}(X_i, X_j) = 0.5^{|i-j|}$. The n_k outcomes in each component (m_k, n_k) with $m_k > 0$ were then generated using its m_k covariates according to the model

$$Y_{ji}|\mathcal{S}_k \sim \mathcal{N}(\alpha_j + \sum_{r=1}^{m_k} \beta_{ks_r} X_{s_r i}, 1), \tag{6}$$

with $j = t_1, \ldots, t_{n_k}$, $i = 1, \ldots, N$, α_j drawn from a normal distribution and the regression coefficients β_{ks_r} sampled in the range $[-5, -2] \cup [2, 5]$. For the 3 $(0, n)$ components, the Ys were generated with the same model but using as Xs a random set of 15 additional covariates which were not made available to the algorithm.

The parameter ρ in the priors (5) must be chosen to ensure that the Markov chain performs correctly, which is to say that moves of both type 1 and type 2 should be accepted with a similar ratio. This is a critical fact and indeed the Markov chain was constructed with two types of moves to guarantee a good mixing. It is instructive to observe what happens when one type of move is hugely favored over the other. If much fewer type 2 moves than type 1 moves are accepted, there is a tendency to overfit the data: in components (m, n) with large n, the number of m tends to increase and many covariates are pulled into the configuration leading to false positives, even if the acceptance rate is the same for both breaking and merging moves. Notice that the limiting case in which only moves of type 1 are present is equivalent to a stochastic search on the subspace of partitions with fixed clusters of Ys. This gives some support to our claim that clustering first the outcomes and then fitting regression models cluster by cluster may not be ideal. In the reverse situation, it is mainly moves of type 2 that are responsible for the assignment of covariates to the components, but some relevant predictors tend to be dismissed as noise. We verified these observations in several simulations. In particular, Table 2 summarizes the results of a sensitivity analysis to the parameter ρ in one set of simulations. It lists the components of the MAP configurations identified for various values of ρ keeping the other hyperparameters fixed ($\alpha_0 = \beta_0 = 0$, $h_0 = 10$, $h = 1$, $\nu = \sigma_0^2 = 0.1$). For $\rho = 0.001$ the ratio of the acceptance rate of type 1 moves over the acceptance rate of type 2 moves was 0.12 and some predictive X's were missed in a few components. For $\rho = 0.5$, the ratio of acceptance rate was 121 in the initial stages of the run and in this case some non-informative X variables were assigned to several components. $\rho = 0.01$ gave a good balance between moves of the two types. The MAP configuration recorded during the $\rho = 0.01$ run contained

S_1	$[X_{23}, X_{36}, X_{67}, X_{77}, X_{85}, X_{107}, X_{112}, X_{122}, X_{130}, X_{145}]$ $[Y_1, Y_{14}, Y_{15}, Y_{16}, Y_{17}, Y_{19}, Y_{25}, Y_{33}, Y_{38}, Y_{46}, Y_{47}, Y_{51}, Y_{64}, Y_{70}, Y_{71}, Y_{82}, Y_{85}, Y_{90}, Y_{98}]$
S_2	$[X_1, X_{23}, X_{77}, X_{102}, X_{135}, X_{145}, X_{175}, X_{198}]$ $[Y_2, Y_{23}, Y_{37}, Y_{43}, Y_{57}, Y_{79}, Y_{80}, Y_{81}, Y_{88}]$
S_3	$[X_{37}, X_{58}, X_{73}, X_{83}, X_{91}, X_{100}, X_{168}, X_{173}, X_{174}]$ $[Y_3, Y_4, Y_6, Y_9, Y_{12}, Y_{13}, Y_{18}, Y_{22}, Y_{24}, Y_{26}, Y_{29}, Y_{36}, Y_{42}, Y_{44}, Y_{45}, Y_{54}, Y_{56}, Y_{58}, Y_{65}, Y_{66}, Y_{68}, Y_{69}, Y_{75}, Y_{77}, Y_{86}, Y_{87}]$
S_4	$[X_{22}, X_{52}, X_{82}, X_{83}, X_{106}]$ $[Y_5, Y_{11}, Y_{91}]$
S_5	$[X_{12}, X_{23}, X_{87}, X_{104}, X_{135}, X_{145}, X_{149}, X_{151}, X_{176}, X_{177}]$ $[Y_{27}, Y_{93}, Y_{96}]$
S_6	$[X_{112}]$ $[Y_{34}, Y_{35}, Y_{39}, Y_{59}, Y_{62}, Y_{72}, Y_{73}, Y_{84}]$
S_7	$[X_1, X_{28}, X_{34}, X_{61}, X_{84}, X_{87}, X_{92}, X_{155}, X_{174}]$ $[Y_7, Y_{41}, Y_{60}, Y_{74}, Y_{83}, Y_{97}]$
S_8	$[\emptyset]$ $[Y_8, Y_{10}, Y_{28}, Y_{30}, Y_{32}, Y_{48}, Y_{49}, Y_{67}, Y_{89}, Y_{94}]$
S_9	$[\emptyset]$ $[Y_{20}, Y_{31}, Y_{52}, Y_{61}, Y_{92}]$
S_{10}	$[\emptyset]$ $[Y_{21}, Y_{40}, Y_{50}, Y_{53}, Y_{55}, Y_{63}, Y_{76}, Y_{78}, Y_{95}, Y_{99}, Y_{100}]$

Table 1: The simulated configuration analysed in the text for a data set with 200 covariates ($\mathcal{X}$) and 100 responses ($\mathcal{Y}$).

nine components. All seven components of type (m, n) with $m > 0$ were successfully identified. The three components for which the relevant predictors were not provided in the X matrix were well separated from the others and were not associated with any predictor, but they were grouped into two components of type $(0, n)$ and not into three, as in Table 1. The outcomes in the $(0, n)$ components in the simulation have however very similar variances, which is probably not sufficient to discriminate between them, because in these components we cluster in terms of the scale factor only. In fact, a similar simulation on data enjoying greater variability across $(0, n)$ components was able to recover the $(0, n)$ components of the underlying model with greater accuracy. The $\rho = 0.1$ and $\rho = 0.5$ runs identified correctly the (m, n) components, but many covariates were associated with the responses that in the underlying model were in $(0, n)$ components. This indeed exemplifies that both types of moves are necessary. We also assessed the sensitivity of the results to the choice of other hyperparameters by varying h_0 and h from 0.1 to 10, σ_0^2 and ν from 0.1 to 1. The values of h, σ_0^2 and ν did not appear to affect the results. There was a bit of sensitivity to h_0: the MAP configurations obtained in runs with $h_0 = 1$ had one or two predictors in the components that are of type $(0, n)$ in Table 1 and for smaller values still, some of the largest (m, n) components of the true configuration were split into subcomponents: for example $S_1 = (10, 19)$ in Table 1 was split into two components $(10, 16)$ and $(11, 3)$ the latter having an additional regressor which is therefore a false positive; similarly, $S_3 = (9, 26)$ appeared as two distinct components $(9, 23)$ and $(10, 3)$.

As we have already noted, inference for the association between X and Y variables can also be drawn based on the marginal probabilities that each outcome be associated with each covariate. Figure 1 displays a heatmap of these marginal probabilities. We note that the locations with high marginal probabilities correspond to the variables in Table 1 that are in the same component. Similarly, the pairwise posterior probabilities that two outcome variables be allocated to the same component can be used to identify Y variables that have similar characteristics (see Figure 2).

Comparison with existing multivariate method

We compare the performance of our method with the multivariate method of Brown et al. (1998). As we have pointed out in the introduction, the latter method selects one set of covariates for all outcomes, and, consequently, if we were to apply it to our data set, we would not recover the underlying model. To make a fair comparison, we thus apply it to each component separately. That is, we assume that we have perfectly clustered the Ys by some method, a result in itself not always easily attainable, and then we select the covariates for the outcomes in each cluster (that is, in each component). When we employed their algorithm, we successfully identified all the correct predictors for components with few outcome variables, such as S_4 and S_5. However, no predictor was identified for components with a relatively large number of response variables, such as S_1 and S_3. Indeed, for the latter two components, the models with highest posterior probabilities contained no regressor and all covariates had marginal posterior probabilities of inclusion less than 0.1. Thus, it appears that the method performs well

Simulated components	Recovered components					
	$RAR = 0.12$ $\rho = 0.001$	$RAR = 0.22$ $\rho = 0.005$	$RAR = 0.54$ $\rho = 0.01$	$RAR = 33$ $\rho = 0.05$	$RAR = 108$ $\rho = 0.1$	$RAR = 121$ $\rho = 0.5$
$\mathcal{S}_1$ $(10, 19)$	✓	✓	✓	✓	✓	$(43, 18) \oplus (18, 1)$ $1\ FN, 40\ FP$
$\mathcal{S}_2$ $(8, 9)$	(0,9) $8\ FN$	✓	✓	✓	✓	(35, 9) $27\ FP$
$\mathcal{S}_3$ $(9, 26)$	$(3, 17) \oplus (2, 9)$ $5\ FN$	(5, 26) $6\ FN, 2\ FP$	✓	✓	✓	(44,26) $27\ FP$
$\mathcal{S}_4$ $(5, 3)$	✓	✓	✓	✓	✓	(7,3) $2\ FP$
$\mathcal{S}_5$ $(10, 3)$	(1, 3) $9\ FN$	✓	✓	✓	✓	(12,3) $2\ FP$
$\mathcal{S}_6$ $(1, 8)$	✓	✓	✓	✓	✓	$(1, 7) \oplus (12, 1)$ $11\ FP$
$\mathcal{S}_7$ $(9, 6)$	✓	✓	✓	✓	✓	(34, 6) $27\ FP$
$\mathcal{S}_8$ $(0, 10)$	(0,8)	(0,15)	(0,15)	(26,10) $26\ FP$	$(31, 7) \oplus (23, 3)$ $54\ FP$	$(55, 1) \oplus (41, 9)$ $96\ FP$
$\mathcal{S}_9$ $(0, 5)$	(0,5)	(0,11)	(0,11)	(25,5) $25\ FP$	$(25, 3) \oplus (18, 2)$ $43\ FP$	(37, 5) $37\ FP$
$\mathcal{S}_{10}$ $(0, 11)$	(0,13)			(21,11) $21\ FP$	$(8, 11) \oplus (36, 10)$ $44\ FP$	(41, 11) $41\ FP$

Table 2: Sensitivity analysis to hyperparameter ρ for one set of runs carried out on simulated data from normal mixtures with $p = 200$, $q = 100$, whose underlying model is given in Table 1. RAR indicates the ratio of acceptance rate of type 1 over type 2 moves. The symbol ✓ indicates that the component was perfectly identified. FN designates false negatives, *i.e.*, relevant X's that were not selected, and FP corresponds to false positives, *i.e.* wrongly selected Xs.

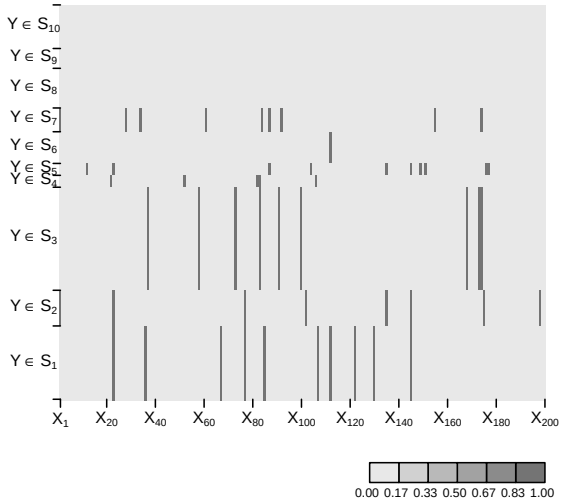

Figure 1: Heat map of marginal posterior probabilities for association of X and Y variables in simulated data from normal mixtures with $p = 200$, $q = 100$, whose underlying model is given in Table 1.

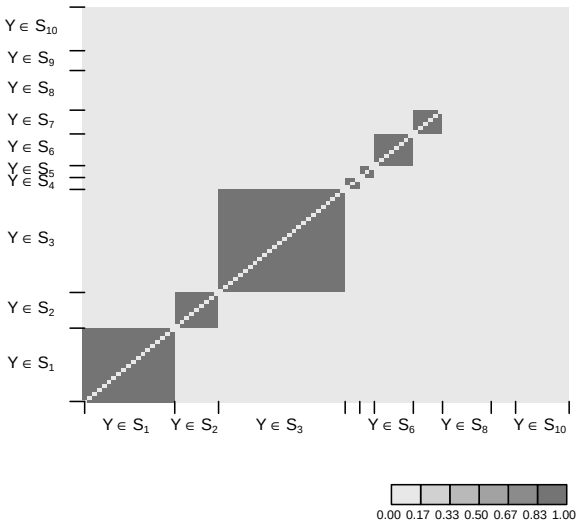

Figure 2: Heat map of pairwise posterior probabilities for Y variables being assigned to the same component in simulated data from normal mixtures with $p = 200$, $q = 100$, whose underlying model is given in Table 1.

in the presence of few correlated outcomes, but is not as suited as ours for variable selection with many response variables.

Performance in the presence of high collinearity

It is reasonable to assume that the presence of highly correlated covariates may complicate the identification of relevant predictors. To investigate the performance of the method in these circumstances, we generated two new covariates

$$\begin{aligned} Z_1 &= X_{37} + X_{58} \\ Z_2 &= X_{37} + X_{58} + X_{73} + X_{83}, \end{aligned}$$

as sums of predictors from one component (S_3) of the configuration summarized in Table 1. We were interested in seeing whether the two variables Z_1 and Z_2 would appear in the two MAP configurations found by the algorithm using as new covariates $(\mathcal{X}, Z_1)$ and $(\mathcal{X}, Z_1, Z_2)$ and the same outcomes $\mathcal{Y}$. We carried out one run for each extended data set with parameters $\alpha_0 = \beta_0 = 0$, $h_0 = 10$, $h = 1$, $\nu = \sigma_0^2 = 0.1$, $\rho = 0.01$. The MAP configurations selected by the algorithm in both cases recovered all components of Table 1 except for substituting Z_1 for X_{58} in the component S_3, which shows that the algorithm is reasonably resistant to collinearity between predictors.

Performance under deviations from normality

We assessed the robustness of the method to deviations from the assumption of normality. We generated the outcome variables, Y, from a mixture of t-distributions, rather than a mixture of normal densities. The n_k outcomes in component $\mathcal{S}_k$ were simulated using the m_k regressors in the same component according to the model

$$Y_{ji}|\mathcal{S}_k = \alpha_j + \sum_{r=1}^{m_k} \beta_{ks_r} X_{s_r i} + \varepsilon_{ij}, \qquad \text{where } \varepsilon_{ij} \sim t_{df}.$$

The algorithm was able to discriminate between the different components, although they were split into smaller subcomponents. This is to be expected since the t-distribution has heavier tails than the normal density. Similarly to what happens in the standard model-based clustering with Gaussian mixtures, observations in the tail area of the t-distribution appear atypical for a component under the normal assumption and tend to be allocated to new components (Peel and McLachlan 2000). We also noticed that the subcomponents contained some, but not all, of the relevant m_k covariates. We repeated this simulation using t distributions with various degrees of freedom and, as anticipated, the performance improved with larger df since the t distribution then approaches the normal distribution.

Performance under nonlinear relationships

We also considered nonlinear relationships between the response variables and the covariates. We used the matrix of covariates, $\mathcal{X}$, generated in the previous simulated

example and the same configuration provided in Table 1. The n_k outcomes in each component (m_k, n_k), however, were drawn from the following model:

$$\begin{aligned} Y_{ji}|\mathcal{S}_k &= \sum_{r=1}^{m_{k_1}} \beta_{k,s_r} \exp\left(X_{s_r,i}\right) + \sum_{r=m_{k_1}+1}^{m_{k_2}} \beta_{k,s_r} \left(X_{s_r,i}\right)^2 + \sum_{r=_{k_2}+1}^{m_k} \beta_{k,s_r} \left(X_{s_r,i}\right)^3 + \varepsilon_{ji}, \\ \varepsilon_{ji} &\sim \mathcal{N}(0,1) \end{aligned}$$

with $1 \leq m_{k_1} < m_{k_2} < m_k$. We successfully clustered the Ys in ten groups. However, for most of these components only two or three relevant predictors were identified and for a few components several false positives were selected. These results are not surprising since the linearity assumption inherent to the model is violated. In such situation, transformations of the variables would be required (Breiman and Friedman 1985; Tibshirani 1988).

Analysis of high-dimensional simulated data

Finally, we tested our method on a large data set, with $p = 1000$ covariates, $q = 1000$ outcomes, $N = 50$ samples, and smaller effect sizes, with the regression coefficients, β_{s_r}, sampled in the range $[-1.5, -0.5] \cup [0.5, 1.5]$. As shown, with the smaller simulated data, the multivariate Bayesian variable selection method of Brown et al. (1998) does not perform well in the presence of many response variables. Here, we compare the performance of our method with the univariate Bayesian stochastic search variable selection (SSVS) algorithm (George and McCulloch 1997) applied to each response variable separately. The results we report for the SSVS were obtained using the same hyperparameter values for all 1000 regression models: $\alpha_0 = \beta = 0, h_0 = 10, h = 1, \nu = \sigma_0^2 = 0.1$ and $\omega = 20$, where ω is the number of covariates expected *a priori* in the model. In theory, one could tune these values for each univariate analysis and improve on the results. However, this is not practically feasible with 1000 outcomes. With regard to this, we should emphasize that the method we have presented has the same number of hyperparameters to contend with as one univariate model, irrespectively of the number of outcomes. We carried out several MCMC runs and we compare the results of the univariate analyses with those obtained by our method using the hyperparameter choices that led to the worst performing run. The configuration we constructed had 35 components: five $(0, n)$, two $(1, n)$ and 28 $(m > 1, n)$ components. We were able to recover with good accuracies 14 of the $(m > 1, n)$ components and only eight regressors were wrongly identified, four of which were in one component. Among the remaining 14 components of type $(m > 1, n)$, 13 were each split into two subcomponents with overlapping regressors among the latter, and only 20 false positives were included across all components. The component of the true model that was recovered with the least accuracy had most of its Ys assigned to one component in the MAP configuration, but the few remaining outcomes were grouped in $(0, n)$ components with other outcomes belonging to $(0, n)$ components in the true model. The almost totality of the true $(0, n)$ components were split into smaller components and were not associated with any covariate. Only two subcomponents expected to be $(0, n)$ contained one regressor. In other runs, our algorithm identified a MAP configuration with fewer split components and fewer false

positives, although some do remain. Our result based on non-tuned hyperparameters, however, is quite good, and much better than the results from the univariate SSVS. For many outcomes, the univariate analyses missed many of the predictors used to simulate the data; generally, only one to three regressors were correctly identified and several false positives were selected. Outcomes that in the true model were in the same component had at best few common covariates. In fact, for components with many regressors, there was often no overlap among the covariates selected by each univariate model. The problem was even acuter for the $(1, n)$ and $(0, n)$ components. For most of the Ys in the $(1, n)$ components, the correct regressor was not identified; instead, different sets of false positives were selected. Similarly, different sets of covariates were associated to the outcomes of the $(0, n)$ components, while no covariate should have been selected. To summarize, the univariate SSVS clearly does not compete with our method in identifying relevant predictors. Furthermore, trying to reconstruct the components of the underlying model by grouping outcomes that are associated to the same set of regressors by the SSVS procedure fails. Our model achieves both grouping of correlated outcomes and identification of their associated regressors with few hyperparameters to tune.

4.2 Real data: CGH and gene expression profiles

Array comparative genomic hybridization (CGH) technologies are designed to measure DNA copy numbers and allow the detection of gains or losses of chromosomal segments. Some of these structural changes may have no obvious phenotypic consequences. Others may affect mRNA transcript levels and, in turn, cause genetic diseases. In an attempt to identify relationships between DNA copy number and mRNA expression level in cancer tissues, Bussey et al. (2006) used the NCI-60 cell line panel, which consists of 60 human cancer cell lines from nine tissue types. They computed Pearson correlation coefficients between all pairs of CGH and mRNA expression levels collected on these samples. This procedure raises a problem of multiplicity. In addition, it does not assess the joint effect of multiple markers nor does it make use of the correlation among transcripts. Our method overcomes these limitations, although it retains the normality and linear association assumptions underlying their analysis.

The processed CGH data and the Affymetrix HG-U133A RMA gene expression estimates were downloaded from CellMiner (discover.nci.nih.gov/cellminer). The X matrix of covariates consists of the CGH data, which are continuous and correspond to $\log_2$ CY3/CY5 intensity ratios. One of the cell lines did not have gene expression estimates and was removed, leaving $N = 59$ samples for the analysis. We considered $q = 3291$ probe sets that showed variability across tissue types, which are representative of 2500 genes, and $p = 261$ CGH clones. The goal of this analysis is to identify groups of correlated gene expression profiles ($\mathcal{Y}$) and their associated DNA copy number variations, represented by some continuous surrogates, the intensity ratios ($\mathcal{X}$).

The model was fit with hyperparameters $\alpha_0 = \beta_0 = 0$, $h_0 = h = 1$, $\sigma_0^2 = \nu = 0.1$, $\rho = 0.01$, and the sampler was run for 20 million iterations. A number of components of type (m, n) were in the MAP configuration. Some components captured known associations and grouped probe sets of genes which are involved in similar biological processes.

Some genes were represented in one component by more than one of their probe sets. Figure 3 shows the 4 outcomes of a $(2, 4)$ component of the MAP configuration, which are the transcript abundances of *CD69, HIST1H3D, LMO2*, and *TAL1*. These are genes involved in hematopoietic development and lymphocyte proliferation, known to be implicated in a subset of human T-cell leukemia. Indeed, one can see that they have higher transcript abundance in some of the NCI-60 leukemia cell lines. The two CGH clones (covariates) in the component are *ABL1*, which is deleted in leukemia cells, and *GNG10*. Figure 4 displays the expression profiles for four probe sets that correspond to *CD24*, a gene which is expressed in various tumors: the NCI-60 cell lines indicate in fact that it has consistently lower transcript abundance among the leukemia and melanoma samples. All these four probe sets were selected in one component that had the clone *RYBP* as a covariate. This same clone is also associated with changes in the expression levels of other genes. For example, *RYBP* appears as a covariate in a component of the MAP configuration that includes as outcomes the expression levels of *GPNMB, MLANA*, and *SOX10*, which are consistently higher in the melanoma tissues, as the NCI-60 data show (Figure 5). We also considered the inference using marginal and pairwise posterior probabilities. We observed good agreement between the associations identified by the two inference strategies. For example, any two outcomes described above as being in the same component of the MAP configuration had also pairwise posterior probabilities greater than 0.75 of being allocated to the same component. Similarly, any clone-probe pair had marginal posterior probability of association greater than 0.75.

As we outlined in Section 3.2, to assess convergence we analysed the trace-plots of the log-posterior probabilities of the visited models and we monitored the effectiveness of the tempering exchanges. We also compared the pairwise posterior probabilities of pairs of variables computed using sets of models sampled at different distant times. In addition, we ran several MCMC chains with the same hyperparameter setting but different initial points. There was good concordance across the results: Figure 6 displays a representation of a subset of the pairwise posterior probabilities obtained from two MCMC chains with different initial configurations.

5 Conclusion

In this paper, we have described a Bayesian stochastic approach designed to find sets of covariates associated with correlated outcomes. This is implemented via an MCMC procedure which sweeps through the space of possible configurations, by attempting to partition or combine subsets of variables. Owing to the large space of possible configurations and the multimodal nature of the posterior distribution, the ergodicity of the simulated Markov chain may be compromised. We have implemented a parallel tempering algorithm in order to overcome this problem.

Like all multivariate methods, the one we have presented here provides substantial improvement over the radicated practice of fitting univariate regression models on each covariate, or of applying variable selection separately on each response variable. More-

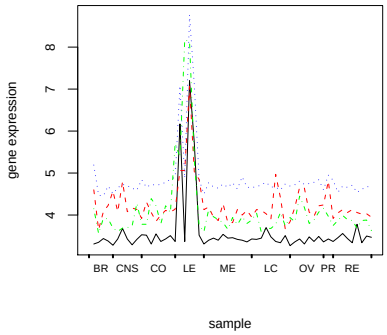

Figure 3: Expression profiles of *CD69* (solid/black), *HIST1H3D* (dashed/red), *LMO2* (dotted/blue), *TAL1* (dot-dashed/green). These genes are the 4 outcomes of a $(2, 4)$ component of the MAP configuration for the CGH data.

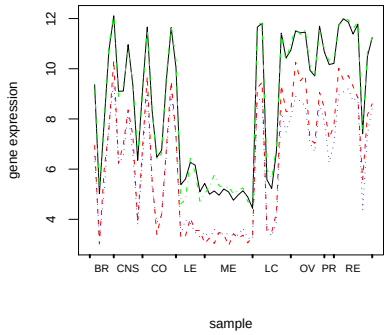

Figure 4: Expression profiles for 4 probe sets of gene *CD24*, which appear in the same component of the MAP configuration for the CGH data.

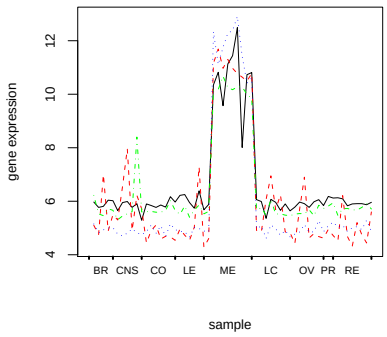

Figure 5: Expression profiles of *DCT* (solid/black), *GPNMB* (dashed/red), *MLANA* (dotted/blue), *SOX10* (dot-dashed/green).

The tissue type labels in Figures 3, 4 and 5 correspond to: BR – breast; CNS – central nervous system; CO – colon; LE – leukemia; ME – melanoma; LC – lung; OV – ovarian; PR – prostate; RE – renal.

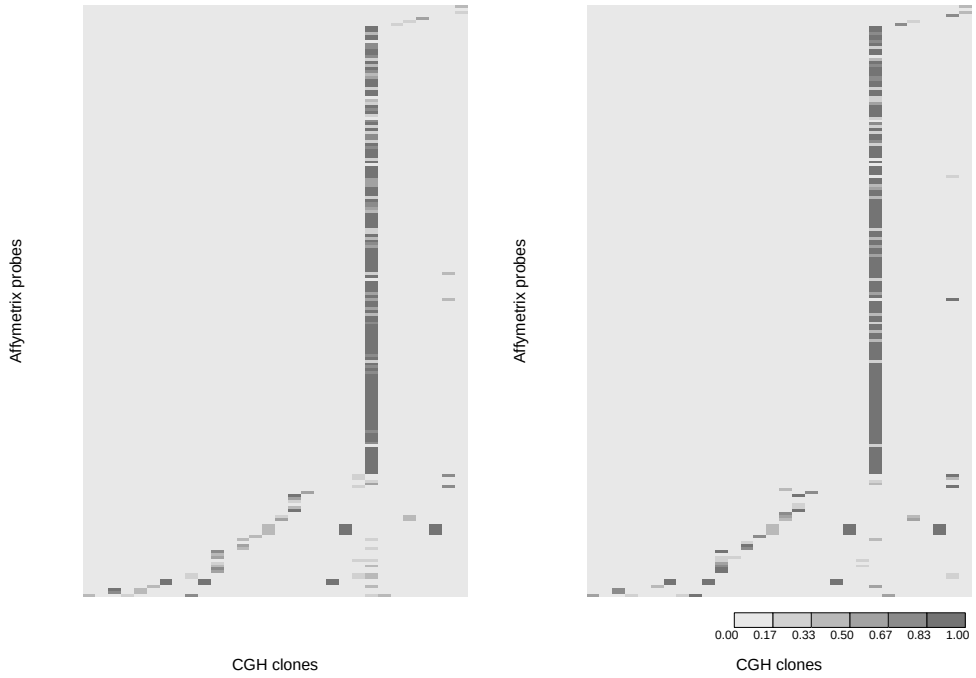

Figure 6: Heat maps of a subset of the posterior probabilities of association between CGH clones and gene expression probes for two MCMC chains started from different initial configurations.

over, it is a method that takes account of the correlation between response variables and assesses the joint effect of multiple covariates on these correlated outcomes, while allowing different subsets of predictors to be associated with different sets of outcomes. For this reason, it is more general and more flexible than the few existing variable selection methods in multivariate regression. We have formulated a model which is quite simple and yet can still capture the key features and the relationships between two high-dimensional data sets. We have verified this fact by evaluating its performance on some simulated examples. Furthermore, to show that the method is suited to analyse data where the number of responses (in the thousands) is comparable with that of the covariates, we have applied it to two genomic data , which consist of 261 continuous regressors and 3291 outcomes. We notice that the method is also applicable to categorical data by suitably re-expressing the variables. For example, we have applied the method to identify polymorphisms in DNA sequences that explain changes in mRNA transcript levels. In this analysis, known as eQTL, which we have not presented in this paper, we used a data set with 3554 responses, gene expression levels, and 2455 single nucleotide polymorphisms (SNPs), which are categorical covariates. Statistical methods for analyzing such data are especially needed now that several high-throughput genomic experiments are being conducted with the goal of integrating the data to understand molecular processes better. Indeed, many efforts are being carried out to correlate molecular data from SNP genotyping, DNA microarray technology and proteomics.

We have considered two standard ways of inference. On the one hand, we have considered the MAP configuration, which provides important information on higher order relationships between variables. But since it is a single configuration, if we limit our analysis to it, we neglect additional information coming from potentially very different configurations with similar posterior probabilities. On the other hand, we have used marginal and pairwise posterior probabilities to identify covariates associated with particular outcomes and to locate correlated outcomes. We thus average over different configurations, but we have to forgo higher-order statistics. We feel it best to employ both inferential strategies in light of their somewhat complementary features.

One can modify the method in different directions. There may be additional information that can be incorporated to elicit the priors and that could be used to design proposals for merging/splitting components. We have, however, preferred to use standard priors, as we do not want to obfuscate the general applicability of the method. In situations where one chooses to use other prior distributions, the model parameters could be updated in the MCMC procedure instead of integrating them out. This may not be practical in high-dimensional problems, since all parameters would need to be updated at each MCMC iteration and appropriate reallocations would need to be devised at each merge and split moves. The marginalization over the model parameters provides a substantial gain in computational speed and efficiency.

The method is based on some assumptions, such as normality of the mixture components, which may not always be adequate and which one could modify, still maintaining the same computational framework. One could, for example, consider mixtures of t-distributions or mixtures of gamma distributions. More general procedures for handling situations where the normality and/or linearity assumptions are not satisfied

would require nonparametric methods, such as spline models, or, even transformations of outcomes and covariates (Breiman and Friedman 1985; Tibshirani 1988). Data transformation can reasonably be implemented when the dimensions of the data are small. However, when dealing with thousands of variables, estimating the optimal transformation for each variable is a daunting task and is not practically feasible. Further research is required to develop efficient algorithms for identifying structures and nonlinear relationships between high-dimensional data sets with arbitrary density.

Appendix

In this appendix we sketch the equivalence between formulae (2) and (3). By definition we have

$$H_0^{-1} + W^T W = \begin{bmatrix} (N + h_0^{-1}) I_{n_k \times n_k} & \left(X 1_N \otimes 1_{n_k}^T\right)^T \\ X 1_N \otimes 1_{n_k}^T & h^{-1} I_{m_k \times m_k} + n_k X X^T \end{bmatrix}.$$

One can easily show that the Schur complement of the first block matrix $(N+h_0^{-1})I_{n_k \times n_k}$ is indeed the matrix A given in (4). By applying standard properties of the determinant, we then have

$$\det\left(W H_0 W^T + I\right) = \det(I)\det(H_0)\det\left(H_0^{-1} + W^T W\right) = h_0^{n_k} h^{m_k} (N + h_0^{-1})^{n_k} \det(A).$$

Similarly, it can be shown that

$$\Omega = (Y - W\theta_0)^T \left(W H_0 W^T + I\right)^{-1} (Y - W\theta_0)$$

using the equality

$$\left(W H_0 W^T + I\right)^{-1} = I - W \left(H_0^{-1} + W^T W\right)^{-1} W^T,$$

and

$$\left[\frac{N h_0 + 1}{h_0}\left(H_0^{-1} + W^T W\right)\right]^{-1} =$$

$$\begin{bmatrix} I_{n_k \times n_k} + \frac{h_0}{(N h_0 + 1)} \left(X 1_N \otimes 1_{n_k}^T\right)^T A^{-1} \left(X 1_N \otimes 1_{n_k}^T\right) & -\left(X 1_N \otimes 1_{n_k}^T\right)^T A^{-1} \\ -A^{-1}\left(X 1_N \otimes 1_{n_k}^T\right) & \frac{N h_0 + 1}{h_0} A^{-1} \end{bmatrix}.$$

References

Abramowitz, M. and Stegun, I. (1972). "Stirling Numbers of the Second Kind." In Abramowitz, M. and Stegun, I. (eds.), *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, 9th printing*, 824–825. New York: Dover. 418

Breiman, L. and Friedman, J. (1985). "Estimating optimal transformations for multiple regression and correlation (with discussion)." *Journal of the American Statistical Association*, 80: 580–619. 428, 434

Breiman, L., Friedman, J., Olshen, R., and Stone, C. (1984). *Classification and Regression Trees*. Belmont, CA: Wadsworth. 414

Brown, P., Vannucci, M., and Fearn, T. (1998). "Multivariate Bayesian Variable Selection and Prediction." *Journal of the Royal Statistical Society, Ser. B*, 60: 627–641. 413, 424, 428

Bryant, P. and Williamson, J. (1978). "Asymptotic Behaviour of Classification Maximum Likelihood Estimates." *Biometrika*, 65: 273–281. 414

Bussey, K., Chin, K., Lababidi, S., Reimers, M., Reinhold, W., Ku, W., Gwadry, F., Kouros-Mehr, A., Fridlyand, J., Jain, A., Collins, C., Nishizuka, S., Tonon, G., Roschke, A., Gehlhaus, K., Kirsch, I., Scudiero, D., Gray, J., and Weinstein, J. (2006). "Integrating data on DNA copy number with gene expression levels and drug sensitivities in the NCI-60 cell line panel." *Molecular Cancer Therapeutics*, 5: 853–867. 429

Chipman, H., George, E., and McCulloch, R. (1998). "Bayesian CART Model Search." *Journal of the American Statistical Association*, 93: 935–948. 414

George, E. and McCulloch, R. (1997). "Approaches for Bayesian variable selection." *Statistica Sinica*, 7: 339–373. 413, 428

Geyer, C. (1991). "Markov Chain Monte Carlo Maximum Likelihood." In Keramigas, E. (ed.), *Computing Science and Statistics*, 156–163. Fairfax: Interface Foundation. 421

Holmes, C., Denison, D., Ray, S., and Mallick, B. (2005). "Bayesian Prediction via Partitioning." *Journal of Computational and Graphical Statistics*, 14: 811–830. 414

Jiang, W. and Tanner, M. (1999). "Hierarchical mixtures-of-experts for exponential family regression models: approximation and likelihood estimation." *The Annals of Statistics*, 27: 987–1011. 416

Lau, J. and Green, P. (2007). "Bayesian model based clustering procedures." *Journal of Computational and Graphical Statistics*, 16: 526–558. 414

Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., and Teller, E. (1953). "Equations of State Calculations by Fast Computing Machines." *Journal of Chemical Physics*, 21: 1087–1091. 419

Peel, D. and McLachlan, G. (2000). "Robust mixture modelling using the t distribution." *Statistics and Computing*, 10: 339–348. 427

Tibshirani, R. (1988). "Estimating transformations for regression via additivity and variance stabilization." *Journal of the American Statistical Association*, 83: 394–404. 428, 434

Zou, H. and Hastie, T. (2005). "Regularization and Variable Selection via the Elastic Net." *Journal of the Royal Statistical Society, Ser. B*, 67: 301–320. 413

Acknowledgments

This work was carried out while S. Monni was in the Department of Biostatistics and Epidemiology at the University of Pennsylvania. The authors thank the Editor, the Associate Editor and the reviewer for their comments and suggestions to improve the paper.