

BAYESIAN AUXILIARY VARIABLE MODELS FOR BINARY AND MULTINOMIAL REGRESSION

CHRIS C HOLMES

University of Oxford

`cholmes@stats.ox.ac.uk`

LEONHARD HELD

Ludwig-Maximilians-University Munich

`leonhard.held@stat.uni-muenchen.de`

SUMMARY

In this paper we discuss auxiliary variable approaches to Bayesian binary and multinomial regression. These approaches are ideally suited to automated Markov chain Monte Carlo simulation. In the first part we describe a simple technique using joint updating which improves the performance of the conventional probit regression algorithm. In the second part we discuss auxiliary variable methods for inference in Bayesian logistic regression, including covariate set uncertainty. Finally we show how the logistic method is easily extended to multinomial regression models. All of the algorithms are fully automatic with no user set parameters and no necessary Metropolis-Hastings accept/reject steps.

Some Key Words: Auxiliary variables, Bayesian binary and multinomial regression, Markov chain Monte Carlo, Model averaging, Scale mixture of normals, Variable selection.

1 Introduction

Binary and polychotomous (or multinomial) regression using Generalised Linear Models (GLMs) is a widely used technique in applied statistics and the Bayesian approach to this subject is well documented, see e.g. Dey, Gosh and Mallick (1999). Inference in Bayesian GLMs is complicated by the fact that no conjugate prior exists for the parameters in the model other than for normal regression and this makes simulation difficult. In a seminal paper Albert & Chib (1993) demonstrated an auxiliary variable approach for binary probit regression models which renders the conditional distributions of the model parameters equivalent to those under the Bayesian normal linear regression model with Gaussian noise. In this case, conjugate priors are available to the conditional likelihood and the block Gibbs sampler can then be used to great effect. In this paper we describe three extensions to the Albert and Chib approach. Firstly, we highlight a simple technique to improve performance in probit regression simulation by jointly updating the regression coefficients and the auxiliary variables. Secondly, we show that the auxiliary approach is also possible for logistic regression, by using a scale mixture of normals representation for the noise process. The logistic model is an important extension as typically the logit link is the preferred method of choice for most statistical applications. The preference for logistic regression is due to the strong interpretation of the regression coefficients which then quantify the change to the log-odds of one class over another for unit change in the associated covariate. Finally we show that the logistic model is easily generalised to accommodate covariate set uncertainty and to multinomial response data.

We believe the methods discussed here offer a valuable extension to the current literature by offering fully automatic multivariate sampling schemes for Bayesian binary and polychotomous regression methods. Chen and Dey (1998) described a logistic regression model based on the scale mixture representation. However, their approach requires the evaluation of the mixing density, which is only known as an infinite series expansion. Hence they must resort approximate numerical techniques. Moreover, their method requires a Metropolis sampler which introduces a data dependent accept-reject stage into their algorithm. In contrast, our approach is exact, fully-automatic (no accept-reject) and we present extensions to multinomial (multi-class, polychotomous) regression. Alternative strategies for the logistic model include Gamerman (1997) who uses a normal approximation to the posterior density of the regression coefficients found using iterative-reweighted-least-squares and Dellaportas and Smith (1993) who suggest the use of adaptive-rejection sampling (ARS) from the univariate conditional densities of the coefficients. The approach of Gamerman requires Metropolis-Hastings updates and hence data dependent accept-reject steps. Our auxiliary variable method uses direct sampling from the condi-

tional distributions. The ARS algorithm does not suffer from Metropolis-Hastings updates but does have the weakness of univariate updating of the coefficients. Hence if there is strong posterior dependence between the coefficients the sampler will mix very poorly. In contrast we provide a **joint multivariate update scheme for the regression parameters.**

In Section 2 we present the methods and algorithms. The approach is also well suited to generalisations of the standard binary regression model and in Section 2.4 we describe such an application, namely, in covariate set uncertainty. In Section 3 we extend our approach to deal with polychotomous data. Finally, in Section 4 we offer a brief discussion, contrasting the approach to existing methods and pointing to possible extensions. An implementation of the various procedures written in MATLAB will be made available on the web site of the first author.

2 Data augmentation in binary regression models

To begin, consider the Bayesian binary regression model,

$$\begin{aligned} y_i &\sim \text{Bernoulli}(g^{-1}(\eta_i)) \\ \eta_i &= \mathbf{x}_i \boldsymbol{\beta} \\ \boldsymbol{\beta} &\sim \pi(\boldsymbol{\beta}) \end{aligned} \tag{1}$$

where $y_i \in \{0, 1\}$, $i = 1, \dots, n$ is a binary response variable for a collection of n objects with associated p covariate measurements $\mathbf{x}_i = (x_{i1}, \dots, x_{ip})$, $g(u)$ is a link function, η_i denotes the linear predictor and $\boldsymbol{\beta}$ represents a $(p \times 1)$ column vector of regression coefficients which *a priori* are from some distribution $\pi(\cdot)$.

2.1 Probit regression using auxiliary variables

For the probit link, $g^{-1}(u) = \Phi(u)$ where $\Phi(u)$ denotes the cdf of a standard normal random variable, the model in (1) has a well known representation using auxiliary variables,

$$\begin{aligned} y_i &= \begin{cases} 1 & \text{if } z_i > 0 \\ 0 & \text{otherwise} \end{cases} \\ z_i &= \mathbf{x}_i \boldsymbol{\beta} + \epsilon_i \\ \epsilon_i &\sim N(0, 1) \\ \boldsymbol{\beta} &\sim \pi(\boldsymbol{\beta}) \end{aligned} \tag{2}$$

where y_i is now deterministic conditional on the sign of the stochastic auxiliary variable z_i . Under independence of ϵ_i , $i = 1, \dots, n$, the marginal likelihood $L(\boldsymbol{\beta}|\mathbf{y})$ in model (2) is the same as in (1).

The advantage of working with representation (2) is that, for judicious choice of $\pi(\boldsymbol{\beta})$, we can perform efficient simulation using the block Gibbs sampler as reported in Albert and Chib (1993), hereafter A&C. In particular, in the case of a normal prior on $\boldsymbol{\beta}$, $\pi(\boldsymbol{\beta}) = N(\mathbf{b}, \mathbf{v})$, the full conditional distribution of $\boldsymbol{\beta}$ is still normal,

$$\begin{aligned}\boldsymbol{\beta}|\mathbf{z} &\sim N(\mathbf{B}, \mathbf{V}) \\ \mathbf{B} &= \mathbf{V}(\mathbf{v}^{-1}\mathbf{b} + \mathbf{x}'\mathbf{z}) \\ \mathbf{V} &= (\mathbf{v}^{-1} + \mathbf{x}'\mathbf{x})^{-1},\end{aligned}\tag{3}$$

where $\mathbf{x} = (\mathbf{x}'_1, \mathbf{x}'_2, \dots, \mathbf{x}'_n)'$. The full conditional for each element z_i is then truncated normal,

$$z_i|\boldsymbol{\beta}, \mathbf{x}_i, y_i \propto \begin{cases} N(\mathbf{x}_i\boldsymbol{\beta}, 1) I(z_i > 0) & \text{if } y_i = 1 \\ N(\mathbf{x}_i\boldsymbol{\beta}, 1) I(z_i \leq 0) & \text{otherwise,} \end{cases}\tag{4}$$

which is straightforward to sample from, see for example Robert (1995).

The auxiliary variable method offers a convenient framework for Markov chain Monte Carlo (MCMC) simulation by iteratively sampling from the conditional densities in (3) and (4). However, a potential problem lurks in that there is strong posterior correlation between $\boldsymbol{\beta}$ and \mathbf{z} , clearly indicated in the model (2). In the standard A&C iterative updating this correlation is likely to cause slow mixing in the chain.

To combat this we suggest a simple approach that reduces autocorrelation and improves mixing in the Markov chain. We propose to update $\boldsymbol{\beta}$ and \mathbf{z} jointly, making use of the factorisation,

$$\pi(\boldsymbol{\beta}, \mathbf{z}|\mathbf{y}) = \pi(\mathbf{z}|\mathbf{y})\pi(\boldsymbol{\beta}|\mathbf{z}),$$

where the distribution $\pi(\boldsymbol{\beta}|\mathbf{z})$ is unchanged from above in (3) but now \mathbf{z} is updated from it's marginal distribution having integrated over $\boldsymbol{\beta}$. We assume from here on that the prior for $\boldsymbol{\beta}$ is a mean zero normal density, $N(\mathbf{0}, \mathbf{v})$. From standard matrix algebra we then obtain,

$$\pi(\mathbf{z}|\mathbf{y}) \propto N(\mathbf{0}, \mathbf{x}\mathbf{v}\mathbf{x}')Ind(\mathbf{y}, \mathbf{z})$$

where $Ind(\mathbf{y}, \mathbf{z})$ is an indicator function which truncates the multivariate normal distribution of \mathbf{z} to the appropriate region. Direct sampling from the multivariate truncated normal is known to be difficult. However, it is straightforward to Gibbs sample the dis-

tribution,

$$z_i | \mathbf{z}_{-i}, y_i \propto \begin{cases} N(m_i, v_i) I(z_i > 0) & \text{if } y_i = 1 \\ N(m_i, v_i) I(z_i \leq 0) & \text{otherwise,} \end{cases}$$

where \mathbf{z}_{-i} denotes the auxiliary variables \mathbf{z} with the i th variable removed. The means m_i and variances v_i , $i = 1, \dots, n$ are obtained from the leave-one-out marginal predictive densities. Using, for example, Henderson & Searle (1981) we can calculate the parameters efficiently as,

$$\begin{aligned} m_i &= \mathbf{x}_i \mathbf{B} - w_i (z_i - \mathbf{x}_i \mathbf{B}) \\ v_i &= 1 + w_i \\ w_i &= h_i / (1 - h_i) \end{aligned} \tag{5}$$

where z_i is the current value for z_i , \mathbf{B} is taken from (3) and h_i is the i th diagonal element of the Bayesian hat matrix, $h_i = (\mathbf{H})_{ii}$, $\mathbf{H} = \mathbf{x} \mathbf{V} \mathbf{x}'$, with \mathbf{V} defined in (3).

Following an update to each z_i we must recalculate the posterior mean \mathbf{B} using, for example, the relationship,

$$\mathbf{B} = \mathbf{B}^{\text{old}} + \mathbf{S}_i (z_i - z_i^{\text{old}})$$

where \mathbf{B}^{old} and z_i^{old} denote the values of \mathbf{B} and z_i prior to the update of z_i , and \mathbf{S}_i denotes the i th column of $\mathbf{S} = \mathbf{V} \mathbf{x}'$.

Note that the variance v_i in (5) is always greater than one, which is the variance of the conventional iterative sampler. During simulation, the calculation of \mathbf{S} , w_i and v_i need only be performed once prior to the MCMC loop. The procedure is best illustrated by considering the pseudo-code listed in Appendix A1, with notation defined in Appendix A0.

The algorithm carries little increase in computational burden over the conventional approach (see the comments at the bottom of the pseudo-code in A1). The use of joint updating should improve mixing and sampling efficiency in the Markov chain. In the next section we present an analysis on four data sets that bears witness to this.

2.2 Empirical test on four data sets

To illustrate the relative efficiency gains of the joint updating scheme we present results from the analysis of four binary regression data sets. The first data set is the Pima Indian

example used in Ripley (1996). The other three data sets are the Australian Credit, Heart, and German Credit data sets used in the STATLOG project (Michie *et al.*, 1994).

To test our procedure we simulated a Gibbs sampler for 10,000 iterations for both the conventional iterative algorithm and our joint update scheme. Unless otherwise stated we take $\pi(\boldsymbol{\beta}) = N(0, 100I_p)$ from here on. To measure efficiency we recorded the total CPU run time and the average Euclidean update distance for $\boldsymbol{\beta}$ between iterations, measured as,

$$\text{Dist.} = \frac{1}{N-1} \sum_{i=1}^{N-1} \|\boldsymbol{\beta}^{(i)} - \boldsymbol{\beta}^{(i+1)}\| \quad (6)$$

where $\boldsymbol{\beta}^{(i)}$ denotes the i th of N MCMC samples.

We also calculated the “effective sample size”, ESS, see Kass *et al.* (1998). The effective sample size for a single coefficient is calculated as,

$$ESS = M / (1 + 2 \sum_{j=1}^k \rho(k)) \quad (7)$$

where M is the number of *post burn in* MCMC samples and $\sum_{j=1}^k \rho(k)$ is the sum of the k monotone sample autocorrelations, estimated by the initial monotone sequence estimator (Geyer, 1992). The ESS was calculated for each coefficient and then averaged.

The results, averaged over 10 runs, along with some characteristics of the data sets are given in Table 1. The standard deviation around the mean was less than 10^{-2} times the mean value for all results. The programs were written in MATLAB 6.5 and run on a desktop PC. The final two columns in Table 1 records the relative efficiency of the joint updating approach over the iterative scheme having standardised for CPU run time. The penultimate column lists the relative updating distance to $\boldsymbol{\beta}$ while the final column lists the relative effective sample size.

The improvement of the joint updating scheme is substantial, giving up to a two-fold improvement in the parameter distance jumped between updates to $\boldsymbol{\beta}$ while also reducing the autocorrelation in the chain, leading to around a 50% improvement in the effective sample size. These results are obtained for minimal increase in algorithmic complexity as shown in Appendix A1. The results are perhaps not surprising given the extent of the posterior correlation in the model (2).

2.3 Logistic regression with auxiliary variables

Consider the model in (2). If we now take $\epsilon_i \sim \pi(\epsilon_i)$ to be the standard logistic distribution then we obtain the logistic regression model. As it stands we loose the conditional conju-

Data			Itr.			Joint				
	n	p	CPU (s)	Dist.	ESS	CPU (s)	Dist.	ESS	Rel. Dist.	Rel. ESS
Pima	532	8	20.17	29.92	831	20.72	58.25	1270	1.97	1.55
A. Credit	690	14	34.82	3.83	546	36.04	8.32	903	2.10	1.60
Heart	270	13	12.29	10.63	634	12.67	21.49	1050	1.96	1.61
G. Credit	1000	24	82.46	3.27	830	83.62	6.01	1265	1.81	1.50

Table 1: Table listing performance measures for Section 2.2 on four data sets, comparing the conventional iterative updating (Itr.) with our joint update scheme. The final two columns present the relative efficiency of the schemes having standardised for CPU run time. The measures Dist. and ESS are defined in equations (6) and (7) respectively.

gacy for updating β . However we can introduce a further set of variables, λ_i , $i = 1, \dots, n$, and note the additional representation

$$\begin{aligned}
y_i &= \begin{cases} 1 & \text{if } z_i > 0 \\ 0 & \text{otherwise} \end{cases} \\
z_i &= \mathbf{x}_i \boldsymbol{\beta} + \epsilon_i \\
\epsilon_i &\sim N(0, \lambda_i) \\
\lambda_i &= (2\psi_i)^2 \\
\psi_i &\sim KS \\
\boldsymbol{\beta} &\sim \pi(\boldsymbol{\beta})
\end{aligned} \tag{8}$$

where ψ_i , $i = 1, \dots, n$, are independent random variables following the Kolmogorov-Smirnov (KS) distribution, e.g. Devroye (1986). In this case, ϵ_i has a scale mixture of normal form with a marginal logistic distribution (Andrews & Mallows, 1974), so that the marginal likelihood $L(\boldsymbol{\beta}|\mathbf{y})$ for models (8) and (1) with logit link are equivalent.

As before, in the case of a normal prior on $\boldsymbol{\beta}$, $\pi(\boldsymbol{\beta}) = N(\mathbf{b}, \mathbf{v})$, the full conditional distribution of $\boldsymbol{\beta}$ given \mathbf{z} and $\boldsymbol{\lambda}$ is still normal,

$$\begin{aligned}
\boldsymbol{\beta}|\mathbf{z}, \boldsymbol{\lambda} &\sim N(\mathbf{B}, \mathbf{V}) \\
\mathbf{B} &= \mathbf{V}(\mathbf{v}^{-1}\mathbf{b} + \mathbf{x}'\mathbf{W}\mathbf{z}) \\
\mathbf{V} &= (\mathbf{v}^{-1} + \mathbf{x}'\mathbf{W}\mathbf{x})^{-1}, \\
\mathbf{W} &= \text{diag}(\lambda_1^{-1}, \dots, \lambda_n^{-1}),
\end{aligned} \tag{9}$$

and the full conditional for z_i is still truncated normal, but now with individual variances

λ_i ,

$$z_i | \boldsymbol{\beta}, \mathbf{x}_i, y_i, \lambda_i \propto \begin{cases} N(\mathbf{x}_i \boldsymbol{\beta}, \lambda_i) I(z_i > 0) & \text{if } y_i = 1 \\ N(\mathbf{x}_i \boldsymbol{\beta}, \lambda_i) I(z_i \leq 0) & \text{otherwise.} \end{cases} \quad (10)$$

Finally, the conditional distribution $\pi(\lambda_i | z_i, \boldsymbol{\beta})$ does not have a standard form. However, it is simple to generate from, which is the only important issue, using rejection sampling as outlined in Appendix A4.

The above specification allows for automatic sampling from the Bayesian logistic regression model using iterative updates, say, $(\boldsymbol{\beta} | \mathbf{z}, \boldsymbol{\lambda})$ followed by $(\mathbf{z} | \boldsymbol{\beta}, \boldsymbol{\lambda})$ and then $(\boldsymbol{\lambda} | \mathbf{z}, \boldsymbol{\beta})$. The sampling scheme will be slower than that for the probit case as not only must we sample $\boldsymbol{\lambda}$ but also the posterior variance-covariance matrix \mathbf{V} in (9) will change for each update of $\boldsymbol{\lambda}$.

As in Section 2.1 we can look to improve matters through joint updating. Interestingly there are two options here. On the one hand we can follow the procedure outlined in Section 2.1 and update $\{\mathbf{z}, \boldsymbol{\beta}\}$ jointly given $\boldsymbol{\lambda}$,

$$\pi(\mathbf{z}, \boldsymbol{\beta} | \mathbf{y}, \boldsymbol{\lambda}) = \pi(\mathbf{z} | \mathbf{y}, \boldsymbol{\lambda}) \pi(\boldsymbol{\beta} | \mathbf{z}, \boldsymbol{\lambda}),$$

followed by an update to $\boldsymbol{\lambda} | \mathbf{z}, \boldsymbol{\beta}$. The pseudo-code for this method is given in Appendix A2.

On the other hand we can update $\{\mathbf{z}, \boldsymbol{\lambda}\}$ jointly given $\boldsymbol{\beta}$,

$$\pi(\mathbf{z}, \boldsymbol{\lambda} | \boldsymbol{\beta}, \mathbf{y}) = \pi(\mathbf{z} | \boldsymbol{\beta}, \mathbf{y}) \pi(\boldsymbol{\lambda} | \mathbf{z}, \boldsymbol{\beta})$$

followed by an update to $\boldsymbol{\beta} | \mathbf{z}, \boldsymbol{\lambda}$. In this latter case the marginal densities for the z_i 's are independent truncated logistic distributions,

$$z_i | \boldsymbol{\beta}, \mathbf{x}_i, y_i \propto \begin{cases} \text{Logistic}(\mathbf{x}_i \boldsymbol{\beta}, 1) I(z_i > 0) & \text{if } y_i = 1 \\ \text{Logistic}(\mathbf{x}_i \boldsymbol{\beta}, 1) I(z_i \leq 0) & \text{otherwise,} \end{cases} \quad (11)$$

where $\text{Logistic}(a, b)$ denotes the density function of the logistic distribution with mean a and scale parameter b (Devroye, 1986, p. 39). An advantage of this latter approach is that sampling from the truncated logistic distribution can be done efficiently by the inversion method, because both the distribution function and its inverse have simple analytic forms. The pseudo-code for this approach is given in Appendix A3.

In Table 2 we repeat the analysis of Section 2.2 but here we compare the two joint update schemes for the logistic model. We ran the Gibbs sampler 10 times each for 10,000

iterations, 5,000 of which are taken as a burn in. The averaged results are presented in Table 2. The standard deviation around the mean was no greater than 10^{-2} times the mean value for all results. In Table 2, scheme A, columns 2-4, refers to the results for updating $\{\mathbf{z}, \boldsymbol{\lambda}\}$ jointly; while scheme B, columns 5-7, lists the results for updating $\{\mathbf{z}, \boldsymbol{\beta}\}$ jointly. The final two columns compares the relative efficiency after we have standardised for CPU run time.

It is first interesting to compare Tables 1 and 2. We see, as expected, that the logistic sampler takes considerably longer to run due to the extra computational burden of sampling the auxiliary mixing variances and having to invert the posterior covariance matrix for $\boldsymbol{\beta}$ at each iteration. Having said that the longest run time for the logistic sampling was still under 10 minutes for 10,000 samples using a non-compiled language (MATLAB). Another striking feature is that the averaged distance jumped between iterations is much larger for the logistic model. This is due to the larger variance $\pi^2/3$ of the logistic distribution compared to the standard normal.

Comparing the procedures within Table 2 we see that scheme A is consistently faster to run than scheme B. This is due to a combination of the simple form for logistic sampling of \mathbf{z} (by inversion) and also that this procedure can be written in vector form in MATLAB which proves much faster than looping. It would be interesting to see how they fair under a compiled language.

The relative efficiencies shown in the rightmost two columns of Table 2 are also interesting. Scheme A is more efficient with regards to the effective sample size measure. This suggests that the logistic updates to \mathbf{z} appear to reduce the autocorrelation in the $\boldsymbol{\beta}$ samples. However, the expected jump distance per iteration is greater under scheme B. Thus, although there is greater autocorrelation in the $\boldsymbol{\beta}$ samples under scheme B, they move larger distances (with greater persistence in direction).

In summary there may not be much to separate the two schemes. Our recommendation at present is to use scheme A as this is simpler to code.

2.4 Extension to covariate set uncertainty

In this section we discuss an extension of the above methods to accommodate covariate set uncertainty. Auxiliary variable approaches are ideal for these scenarios as they allow for joint updates to the covariate set \mathbf{x} and the regression coefficients $\boldsymbol{\beta}$ which leads to efficient dimension jumping moves.

The standard approach to covariate set uncertainty is to adopt a prior distribution on the covariate set $\pi(\mathbf{x})$ via a covariate indicator vector $\boldsymbol{\gamma} = \{\gamma_1, \dots, \gamma_p\}$, $\gamma_i \in \{0, 1\}$, $i = 1, \dots, p$, such that $\gamma_i = 1$ if the i th covariate is present in the model and $\gamma_i = 0$ if

Data	A: $\{\mathbf{z}, \boldsymbol{\lambda}\}$			B: $\{\mathbf{z}, \boldsymbol{\beta}\}$			(B / A)	CPU s^{-1}
	CPU (s)	Dist.	ESS	CPU (s)	Dist.	ESS	Rel. Dist.	Rel. ESS
Pima	190.2	87.19	1131	249.5	151.33	996	1.33	0.67
A. Credit	255.6	10.80	740	340.8	21.64	638	1.50	0.65
Heart	145.1	30.10	890	181.4	56.93	795	1.51	0.71
G. Credit	384.3	9.74	1236	545.9	15.85	1044	1.15	0.60

Table 2: Table listing performance measures of the two joint sampling schemes in Section 2.3, on the four data sets described in Section 2.2. Scheme A uses a joint update to $\{\mathbf{z}, \boldsymbol{\lambda}\}$ while Scheme B uses a joint update to $\{\mathbf{z}, \boldsymbol{\beta}\}$. The measures Dist. and ESS are defined in equations (6) and (7)

it is not. A prior on the model space can then be specified via a prior on the covariate indicator, $\pi(\boldsymbol{\gamma})$. The parameter vector $\boldsymbol{\gamma}$ is then included in the model specification and updated as part of the simulation.

Bayesian analysis of models of random dimension have become extremely popular following the introduction of sampling techniques such as Green (1995). Clyde (1999) provides a good overview of this so called Bayesian model averaging approach. However, it is well known that simulation of variable dimensional models can be problematic as a change to the model structure typically causes a large change to the likelihood of any parameter values carried through to the new model, see Brooks *et al.* (2003). A key advantage of using auxiliary variables is that when updating the model we can condition on $\{\mathbf{z}, \boldsymbol{\lambda}\}$ and jointly update $\{\boldsymbol{\gamma}, \boldsymbol{\beta}\}$ from the full conditional distribution given a change to the covariates. The vector \mathbf{z} retains information about the likelihood which allows for optimal updates to be made to $\boldsymbol{\beta}$, given a change in the covariate set. Updating the $\boldsymbol{\beta}$ vector jointly with the covariate set is extremely important as typically, when the covariates are non-orthogonal, there is strong linear dependence between the regression coefficients.

To sample from the posterior model space we use joint updates,

$$\pi(\boldsymbol{\gamma}, \boldsymbol{\beta} | \mathbf{z}, \boldsymbol{\lambda}) = \pi(\boldsymbol{\gamma} | \mathbf{z}, \boldsymbol{\lambda}) \pi(\boldsymbol{\beta} | \boldsymbol{\gamma}, \mathbf{z}).$$

To generate from $\pi(\boldsymbol{\gamma} | \mathbf{z})$ we could use Gibbs sampling on the indicator variables. However, under Peskun ordering it turns out to be more efficient to use a Metropolis-Hastings step to update the current covariate set, defined by $\boldsymbol{\gamma}$, with a joint update to $\boldsymbol{\beta}$ as well,

$$q(\boldsymbol{\gamma}^*, \boldsymbol{\beta}^*) = \pi(\boldsymbol{\beta}^* | \boldsymbol{\gamma}^*, \mathbf{z}, \boldsymbol{\lambda}) q(\boldsymbol{\gamma}^*),$$

where $q()$ denotes a proposal distribution, $\pi(\boldsymbol{\beta}^*|\boldsymbol{\gamma}^*, \mathbf{z}, \boldsymbol{\lambda})$ is the conditional multivariate normal posterior distribution (9) given the covariate set defined by $\boldsymbol{\gamma}^*$, and $q(\boldsymbol{\gamma}^*)$ is a, possibly symmetric, Metropolis-Hastings proposal density that may, or may not, be based on the current covariate set $\boldsymbol{\gamma}$. In this case, some straightforward algebra leads to the acceptance probability of the joint move as,

$$\alpha = \min \left\{ 1, \frac{|\mathbf{V}_{\boldsymbol{\gamma}^*}|^{1/2} |\mathbf{v}_{\boldsymbol{\gamma}}|^{1/2} \exp(0.5 \mathbf{B}_{\boldsymbol{\gamma}^*}' \mathbf{V}_{\boldsymbol{\gamma}^*}^{-1} \mathbf{B}_{\boldsymbol{\gamma}^*}) \pi(\boldsymbol{\gamma}^*) q(\boldsymbol{\gamma}|\boldsymbol{\gamma}^*)}{|\mathbf{V}_{\boldsymbol{\gamma}}|^{1/2} |\mathbf{v}_{\boldsymbol{\gamma}^*}|^{1/2} \exp(0.5 \mathbf{B}_{\boldsymbol{\gamma}}' \mathbf{V}_{\boldsymbol{\gamma}}^{-1} \mathbf{B}_{\boldsymbol{\gamma}}) \pi(\boldsymbol{\gamma}) q(\boldsymbol{\gamma}^*|\boldsymbol{\gamma})} \right\} \quad (12)$$

where α denotes the acceptance probability of the proposal and $\{\mathbf{B}_{\boldsymbol{\gamma}}, \mathbf{V}_{\boldsymbol{\gamma}}\}$ are defined in (9), where the subscripts indicate that they are conditioned on the covariate set defined by $\boldsymbol{\gamma}$. Note that the realised (drawn) values of $\{\boldsymbol{\beta}, \boldsymbol{\beta}^*\}$ do not appear in the acceptance probability (12), which resembles the Bayes factor of a standard Bayesian linear model. This implicit marginalisation of $\boldsymbol{\beta}$ in the proposal step leads to efficient dimension sampling, as the $\boldsymbol{\beta}$'s are being updated from their full conditional distributions given the change to the covariate set. Following an update to $\{\boldsymbol{\gamma}, \boldsymbol{\beta}\}$ we then update $\{\mathbf{z}, \boldsymbol{\lambda}\}|\{\boldsymbol{\gamma}, \boldsymbol{\beta}\}$ using the scheme outlined above in Section 2.3.

To illustrate the approach we consider again the Pima Indian data from Ripley (1996). The regression task is to predict whether patients will test positive or negative for diabetes using a set of seven covariate measurements, observed on a group of adult females of Pima Indian heritage. There are 532 records, selected from a larger data set, with the following predictor variables: number of pregnancies (NP); plasma glucose concentration (Gl); distolic blood pressure (BP); triceps skin fold thickness (TST); body mass index (BMI); diabetes pedigree function (DP); and, age (Ag). In Ripley (1996) they used a classical (non-Bayesian) logistic regression model and noted that some of the covariates appeared irrelevant. Ripley (1996) went on to perform stepwise variable selection using an AIC model choice criteria and found that the covariates blood pressure and skin thickness were dropped from the final model. We performed a Bayesian analysis using independent priors on the covariates and regression coefficients as, $\pi(\boldsymbol{\gamma}) = \prod_i \pi(\gamma_i)$, with $\pi(\gamma_i = 1) = 0.5$ for $i = 1, \dots, p$ and $\pi(\boldsymbol{\beta}) = N(0, 100I_p)$. Updates to the covariate set were made using a Metropolis proposal as follows. We select a covariate at random and propose $\gamma_i^* = 1$, if the current $\gamma_i = 0$, $\gamma_i^* = 0$ otherwise. This results in the final term, $\frac{\pi(\boldsymbol{\gamma}^*) q(\boldsymbol{\gamma}|\boldsymbol{\gamma}^*)}{\pi(\boldsymbol{\gamma}) q(\boldsymbol{\gamma}^*|\boldsymbol{\gamma})}$, in (12) being one. Following updates to $\{\boldsymbol{\gamma}, \boldsymbol{\beta}\}$ we jointly update $\{\mathbf{z}, \boldsymbol{\lambda}\}$ using the marginal truncated logistic sampler, Appendix A3.

We performed a simulation of 10,000 iterations and discarded the first 5,000 as a burn-in. In Table 3, we show the estimates of the posterior probabilities, $\pi(\gamma_i = 1|y)$, for the seven covariates, along with the standard deviations in these MCMC estimates taken

Covariates	NP	Gl	BP	TST	BMI	DP	Ag
$E[\gamma_i]$	0.923	0.999	0.009	0.037	0.993	0.944	0.129
MCMC Std	0.083	0.001	0.008	0.011	0.001	0.031	0.108

Table 3: Row 1, lists the covariate acronyms for the Pima Indian data set example in Section 3: (NP), number of pregnancies; (Gl), plasma glucose concentration; (BP), distolic blood pressure; (TST), triceps skin fold thickness; (BMI), body mass index; (DP), diabetes pedigree function; and, (Ag), age. Row 2, lists the posterior probabilities of covariate selection. In row 3, we report the MCMC standard deviations of the estimates $\pi(\gamma_i = 1|y)$, taken across nine consecutive post burn-in regions of size 1,000 MCMC samples.

from nine consecutive regions of the MCMC samples, $\{(1001, 2000), \dots, (9001, 10000)\}$. The chain appears to be mixing well under the data augmentation approach. The overall acceptance rate of the covariate update proposals was around 4% which is good when considering the posterior probabilities $\pi(\gamma_i|y)$ shown in Table 3. The estimates of $\pi(\gamma_i = 1|y)$ are in accordance with the observations of Ripley (1996) though we find there also appears to be some doubt as to the relevance of age.

3 Bayesian polychotomous regression

In this section we highlight another useful extension of the logistic auxiliary variable approach to data where the response is multicategorical. It is straightforward to extend the logistic regression models of Section 2.3 for ordinal data, such as the cumulative or the sequential model, following the algorithmic approach discussed by Albert & Chib, (1993, 2001) for the probit link. However, the logistic model also allows for a simple extension to polychotomous data. That is, when $y_i \in \{1, \dots, Q\}$, is an unordered category indicator of one of Q classes. This is known as polychotomous regression (McCullagh & Nelder, 1989).

The polychotomous generalisation of the logistic regression model is defined via,

$$y_i \sim \mathcal{M}(1; \theta_{i1}, \dots, \theta_{iQ})$$

$$\theta_{ij} = \frac{\exp(\mathbf{x}_i \boldsymbol{\beta}_j)}{\sum_{k=1}^Q \exp(\mathbf{x}_i \boldsymbol{\beta}_k)} \quad (13)$$

where $\mathcal{M}(1; \cdot)$ denotes the single sample multinomial distribution. We note in (13) that there is now a separate set of coefficients $\boldsymbol{\beta}_j$ for each category. It is usual to fix one set of coefficient, say $\boldsymbol{\beta}_Q$, to be zero, so that the logistic regression model is recovered for

$Q = 2$ and the interpretation of the coefficients are in terms of the change to the log-odds relative to category Q .

The ability to extend the methods discussed in Section 2.3 to the polychotomous case arises by considering the conditional likelihood of a set of coefficients, say β_j , having fixed the other coefficients, $\beta_{-j} = \{\beta_1, \dots, \beta_{j-1}, \beta_{j+1}, \dots, \beta_Q\}$, in the model. In this case we find,

$$\begin{aligned} L(\beta_j | \mathbf{y}, \beta_{-j}) &\propto \prod_{i=1}^n \prod_{k=1}^Q [\theta_{ik}]^{I(y_i=k)}, \\ &\propto \prod_{i=1}^n [\eta_{ij}]^{I(y_i=j)} [w_i(1 - \eta_{ij})]^{I(y_i \neq j)}, \\ &\propto \prod_{i=1}^n [\eta_{ij}]^{I(y_i=j)} [1 - \eta_{ij}]^{I(y_i \neq j)}, \end{aligned} \tag{14}$$

where $I(\cdot)$ is the logical indicator function, w_i is a weight function independent of β_j and,

$$\begin{aligned} \eta_{ij} &= \frac{\exp(\mathbf{x}_i \beta_j - C_{ij})}{1 + \exp(\mathbf{x}_i \beta_j - C_{ij})}, \\ C_{ij} &= \log \sum_{k \neq j} \exp(\mathbf{x}_i \beta_k). \end{aligned} \tag{15}$$

The point here is that the conditional likelihood $L(\beta_j | \mathbf{y}, \beta_{-j})$ has the form of a logistic regression on class indicator $I(y_i = j)$. This allows us to use the logistic sampling technique highlighted in Section 2.3 embedded within a Gibbs step looping over the $Q - 1$ classes. Appendix A5 lists the polychotomous pseudo-code, generalising the logistic scheme of A3.

To illustrate the polychotomous method we analysed the data discussed in Bull (1994). The study in Bull (1994) relates to workplace attitudes to smoking restrictions within Toronto and in particular to changes in attitude before and after the introduction of a byelaw regulating smoking in the workplace. The data was collected by random telephone surveys of households in the Toronto region. The response variable, y_i , has three categories relating to the subjects beliefs that smoking in the workplace should be “prohibited”, “restricted” or “unrestricted”. Bull (1994) provides full details and background to the data. In the original study Bull analysed a variety of models constructed from subsets of 12 covariates recorded on each telephone response. Here we restrict our attention to the four “byelaw-related” covariates, namely, time-of-survey, $x_1 \in \{0, 1\}$ denoting whether the data was recorded before or after the introduction of the byelaw; place-of-work, $\{x_2, x_3\}$, where $x_2 \in \{0, 1\}$, denotes workplace outside the city or not, and $x_3 \in \{0, 1\}$, denotes

workplace outside the home or not; finally place-of-residence $x_4 \in \{0, 1\}$, denotes residence within the city of Toronto or otherwise.

The primary interest is in quantifying changes to attitudes from before to after the implementation of the bylaw. We ran the polychotomous Gibbs sampler on the data and following Bull (1994) we used “restricted” as the baseline category with associated β set to zero. The sampler was run for 10,000 iterations the first 5,000 being discarded. We then examine the marginal posterior effect of time-of-survey.

Before discussing the results we note that the auxiliary variable approach is well tailored to marginal posterior density estimation through Rao-Blackwellization (Gelfand and Smith, 1990). For example, it is often of interest to estimate the marginal posterior density $p(\beta^*|\mathbf{y})$ for some subset β^* of β , for example for the computation of simultaneous credible regions (Held, 2004). In higher dimensions, the only feasible way to do this is based on an average of the corresponding full conditional distributions of the regression coefficients β^* ,

$$\hat{p}(\beta^*|\mathbf{y}) = \frac{1}{M} \sum_{j=1}^M p(\beta^*|\mathbf{y}, \Theta^{(j)}) \quad (16)$$

where $\Theta^{(j)}$ denotes all other parameters in the model and the upper index j denotes the j -th sample from the posterior distribution and M is the number of samples. Without our auxiliary variable approach, a fundamental problem is, firstly, that $p(\beta^*|\mathbf{y}, \Theta^{(j)})$ may not have closed form and secondly, even if it does, the denominator in

$$p(\beta^*|\mathbf{y}, \Theta) = \frac{p(\mathbf{y}|\beta^*, \Theta) \times p(\beta^*)}{p(\mathbf{y}|\Theta)},$$

which depends on Θ , is typically not known, hence (16) cannot be applied. However, in the auxiliary variable method for MCMC simulation, then $p(\beta^*|\mathbf{y})$ is estimated via

$$\hat{p}(\beta^*|\mathbf{y}) = \frac{1}{M} \sum_{j=1}^J N(\mathbf{B}^{*(j)}, \mathbf{V}^{*(j)}) \quad (17)$$

where $N(\mathbf{B}^{*(j)}, \mathbf{V}^{*(j)})$ denotes the multivariate normal distribution (9) with appropriate submatrices obtained from the MCMC samples. Clearly the normalizing constant is then known. Estimation of the marginal posterior of the regression coefficients can thus be simply achieved through the Gaussian mixture (17).

In Figure 1 we show the marginal densities for the regression coefficients associated with the time-of-study covariate for the “unrestricted” category (top) and the “prohibited” category (bottom) relative to the “restricted” case. These plots were obtained as Gaussian mixture models (17). Using the model (13) we interpret these plots as quantifying the

uncertainty in the change over time to the log-odds of the two categories relative to the “restricted” response. This interpretation is gained from using the conditional logit link (13). The two plots are interesting. The top plot indicates that the “unrestricted” subjects were more likely to prefer restrictions following the introduction of the byelaw while those subjects previously preferring prohibition see a hardening of attitudes, being more likely to vote for prohibition after the byelaw came into action.

We can use the Gaussian mixture representation (17) to also estimate the sign of the effect, $Pr(\beta \leq 0)$, by using the distribution function of the normal density. From this we estimate the probability of a negative time-of-survey effect on the log odds of unrestricted to restricted is 0.7552; while the probability of a positive time-of-study effect on the log odds of prohibited to restricted is 0.7426.

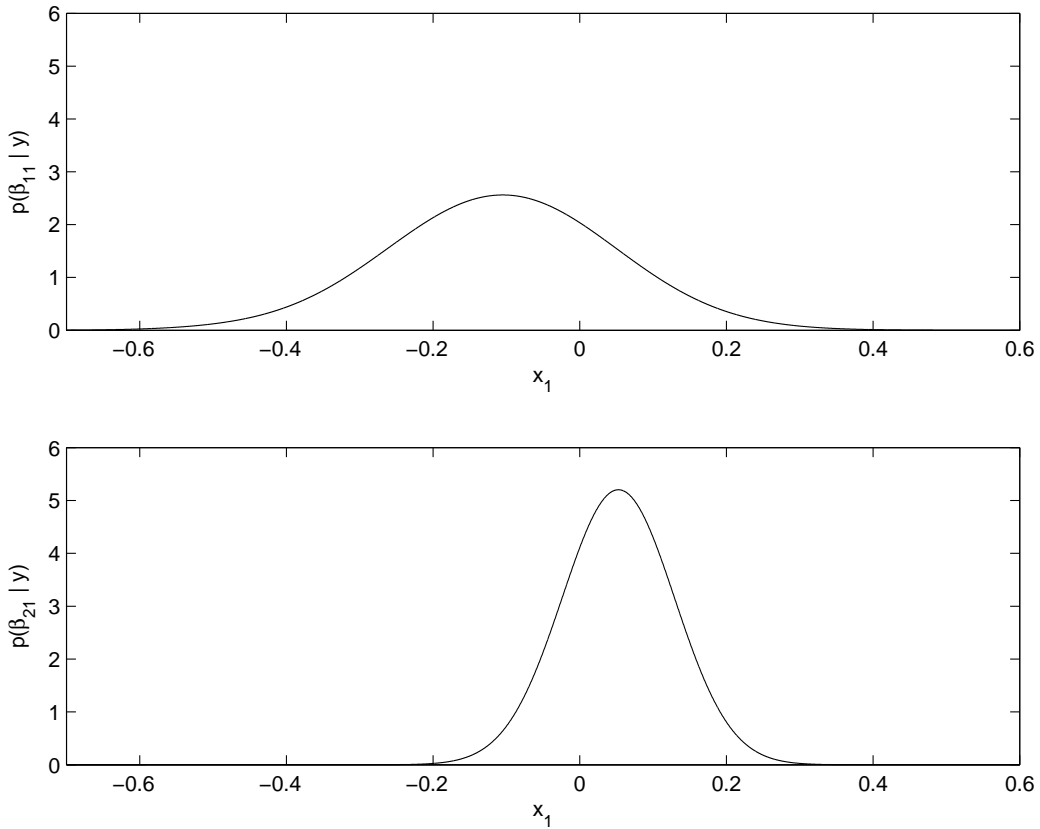


Figure 1: Plot showing the marginal densities of β_{11} and β_{21} , the effect of covariate time-of-survey for the unrestricted (top plot) and prohibited (bottom plot) relative to the restricted class, using the data from Section 3. The plots are obtained using the Gaussian mixture representation (17).

4 Discussion

We have discussed a variety of auxiliary variable methods for Bayesian binary and polychotomous regression. All of the algorithms are fully automatic with no user set parameters and as such they are ideal for embedding in statistical software. Although concentrating here on GLMs, the methods are readily applicable to non-linear modelling using free-knot regression splines (see Denison *et al*, 2002).

Popular current alternatives for MCMC simulation in Bayesian logistic regression models are found in Albert & Chib (1993) and Gamerman (1997). In Albert & Chib (1993) it was noted that specifying a scale mixture for λ_i in (8) as $\lambda_i \sim \text{Gamma}(4, 4)$ induces a t-distribution for ϵ_i with 8 degrees of freedom which gives a good approximation to the logistic distribution (up to a change in scale). However, this remains an approximation and a qq-plot of the true logistic distribution against that found using the Student approximation reveals considerable departure in the tails, see Figure 2.

In applications it will be difficult to assess the effect of this bias on the posterior distribution of the regression coefficients. Our approach, however, is exact and provides a fast, efficient and automatic algorithm for inference in logistic regression models.

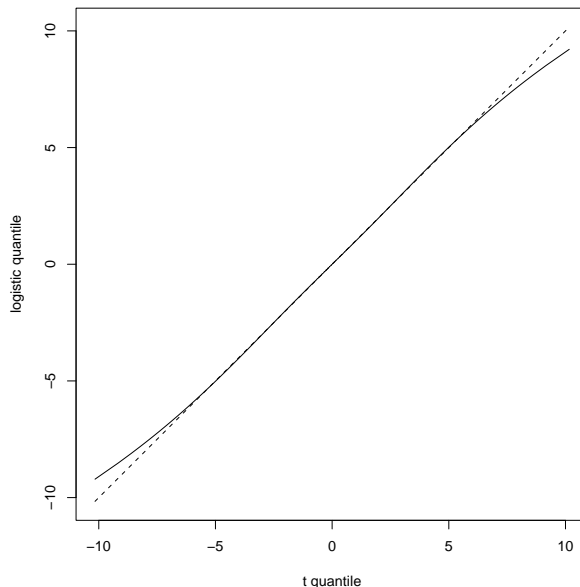


Figure 2: Plot of t -quantiles against logistic quantiles for probabilities between 0.0001 and 0.9999 (Solid line). The dashed line gives the reference line if the two distributions are identical.

An alternative algorithm without auxiliary variables is described in Gamerman (1997). Gamerman suggests a “weighted least squares” Metropolis-Hastings proposal based on a linear Taylor-approximation of the likelihood. This algorithm works well in practice, in

particular if the number of parameters to be updated is not too large. However, there are no guarantees on the acceptance rates and this detracts significantly when implementing this in generic software. In contrast, due to the use of auxiliary variables the corresponding acceptance rates in our approach will always be unity, other than in the variable dimension case Section 2.4 where we specifically choose the Metropolis-Hastings over the available Gibbs sampler. Moreover, the extension of Gamerman's approach to variable dimension settings is non-trivial whereas we have shown in 2.4 this to be straightforward for our approach.

ACKNOWLEDGMENTS

The authors would like to express their thanks to Maria De Iorio, John Kent, Stefan Lang and especially to Dave Stephens for their helpful comments and suggestions.

References

- Andrews, D.F. & Mallows, C.L. (1974). Scale mixtures of normal distributions. *J. R. Statist. Soc. B* **36**, 99-102.
- Albert, J. & Chib, S. (1993). Bayesian analysis of binary and polychotomous response data. *Journal of the American Statistical Association* **88**, 669-679.
- Albert, J. & Chib, S. (2001). Sequential ordinal modeling with applications to survival data. *Biometrics*, **57**, 829-836.
- Brooks, S.P., Giudici, P. & Roberts, G. O. (2003). Efficient construction of reversible jump MCMC proposal distributions (with discussion). *J. R. Statist. Soc. B*, **65**, 3-55.
- Bull, S. (1994). Analysis of attitudes toward workplace smoking restrictions. In *Case Studies in Biometry*. Eds. Lange, N., Ryan, L., Billard, L., Brillinger, D., Conquest, L. and Greenhouse, J. Published by Wiley.
- Chen, M.-H. and Dey, D.K. (1998) Bayesian modeling of correlated binary responses via scale mixture of multivariate normal link functions. *Sankhyā: The Indian Journal of Statistics*, **60**, 322-343.
- Clyde, M. (1999). Bayesian model averaging and model search strategies (with discussion). In *Bayesian Statistics 6* (ed. J.M. Bernardo, J.O. Berger, A.P. Dawid, A.F.M. Smith). Oxford: Clarendon Press.

- Dellaportas, P. and Smith, A.F.M. (1993). Bayesian inference for generalized linear and proportional hazard models via Gibbs sampling. *Appl. Statist.*, **42**, 443-459.
- Denison, D.G.T., Holmes, C.C., Mallick, B.K. & Smith, A.F.M. (2002). *Bayesian methods for nonlinear classification and regression*. Chichester: Wiley.
- Dey, D.P., Gosh, S. & Mallick, B. (1999). *Generalized Linear Models: A Bayesian Perspective*. New York: Marcel Dekker.
- Devroye, L. (1986). *Non-Uniform Random Variate Generation*. New York: Springer.
- Gamerman, D. (1997). Efficient sampling from the posterior distribution in generalized linear mixed models. *Statistics and Computing*, **7**, 57-68.
- Gelfand, A.E. & Smith, A.F.M. (1990). Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association* **85**, 398-409.
- Geyer, C.J. (1992). Practical Markov chain Monte Carlo. *Statistical Science*, **7**, 473-511.
- Green, P.J. (1995). Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, **82**, 711-732.
- Held, L. (2004). Simultaneous posterior probability statements from Monte Carlo output. *Journal of Computational and Graphical Statistics*, **13**, 20-35.
- Henderson, H.V. & Searle, S.R. (1981). On deriving the inverse of a sum of matrices. *SIAM Rev.*, **23** 53-60.
- Kass, R.E., Carlin, B.P., Gelman A. and Neal, R.M. (1998). Markov Chain Monte Carlo in Practice: A Roundtable Discussion, *The American Statistician*, **52**, 93-100.
- McCullagh, P. & Nelder, J.A. (1989). *Generalised Linear Models*, 2nd ed. London: Chapman and Hall.
- Michie, D., Spiegelhalter, D.J. & Taylor, C.C. (1994). *Machine Learning, Neural and Statistical Classification*. Ellis Horwood.
- Ripley, B.D. (1996). *Pattern Recognition and Neural Networks*. Cambridge University Press.
- Robert, C.P. (1995). Simulation of truncated normal variables. *Statistics and Computing* **5**, 121-125.

Appendix: A0 Pseudo-code

This section lists the pseudo-code for the algorithms. The code assumes that the prior on β is $\pi(\beta) = N(\mathbf{0}, \mathbf{v})$ and the design matrix X is of dimension $(n \times p)$. Comment lines are preceded by % %.

$A[i]$ denotes the i th element of a column matrix A ; $A[i, j]$ denotes the i th, j th element of a matrix A ; $A[i,]$ denotes the i th row of A , $A[, j]$ denotes the j th column; AB denotes matrix multiplication of A and B ; $A[i,]B[, j]$ denotes the row, column inner product.

A1: Procedure for joint sampling in Bayesian probit, $\beta \sim \pi(\beta|Y, X, v)$

% % First record constants unaltered within MCMC loop

$V \leftarrow (X^T X + v^{-1})^{-1}$

$L \leftarrow \text{Chol}(V)$

% % L stores the *lower* triangular Cholesky factorisation of V

$S \leftarrow V X^T$

FOR j=1 to number of observations

$H[j] \leftarrow X[j,] S[, j]$

% % H stores the diagonal elements of hat matrix ($X S$)

$W[j] \leftarrow H[j] / (1 - H[j])$

$Q[j] \leftarrow W[j] + 1$

END

% % Initialise latent variable Z , from truncated normal

$Z \sim N(0, I_n) \text{Ind}(Y, Z)$

$B \leftarrow S Z$ % % B denotes the conditional mean of β .

FOR i=1 to MCMC iterations

FOR j=1 to number of observations

$z_{old} \leftarrow Z[j]$

$m \leftarrow X[j,] B$

$m \leftarrow m - W[j](Z[j] - m)$

% % now draw $Z[j]$ from truncated normal

$Z[j] \sim N(m, Q[j]) \text{Ind}(Y[j], Z[j])$

% % make change to B

$B \leftarrow B + (Z[j] - z_{old}) S[, j]$

END

% % now draw new value of β

$T \sim N(0, I_p)$

$\beta[, i] \leftarrow B + L T$

END MCMC iterations; RETURN β

% % Note, to convert to conventional iterative sampling: Leave out initial loop on line 6 to line 11 which calculates $\{W[j], Q[j]\}$ prior to the MCMC loop; Set $B \leftarrow 0$ as the first line within the MCMC loop; Change innermost loop to: $m \leftarrow X[j,] \beta[, i]$; $Z[j] \sim N(m, 1) \text{Ind}(Y[j])$; $B \leftarrow B + Z[j] S[, j]$.

A2: Procedure for sampling Bayesian logistic model, $\beta \sim \pi(\beta|Y, X, v)$, using joint update to $\{z, \beta\}$.

```
% % Initialise mixing weights  $\Lambda$  to the  $(n \times n)$  identity matrix
 $\Lambda \leftarrow I_n$ 
% % draw  $Z$  from truncated normal
 $Z \sim N(0, I_n) \text{Ind}(Y, Z)$ 
FOR i = 1 to number of MCMC iterations
     $V \leftarrow (X^T \Lambda^{-1} X + v^{-1})^{-1}$ 
    % % note that  $\Lambda^{-1}$  is a diagonal matrix and
    hence simple to invert
     $L \leftarrow \text{Chol}(V)$ 
    % % So  $L$  stores the lower triangular Cholesky factorisation of  $V$ 
     $S \leftarrow V X^T$ 
     $B \leftarrow S \Lambda^{-1} Z$ 
    FOR j=1 to number of observations
         $z_{old} \leftarrow Z[j]$ 
         $H[j] \leftarrow X[j, ] S[, j]$ 
         $W[j] \leftarrow H[j] / (\Lambda[j, j] - H[j])$ 
         $m \leftarrow X[j, ] B$ 
         $m \leftarrow m - W[j] (Z[j] - m)$ 
         $q \leftarrow \Lambda[j, j] (W[j] + 1)$ 
        % % draw  $Z[j]$  from truncated normal
         $Z[j] \sim N(m, q) \text{Ind}(Y[j], Z[j])$ 
        % % make change to  $B$ 
         $B \leftarrow B + \left( \frac{Z[j] - z_{old}}{\Lambda[j, j]} \right) S[, j]$ 
    END
    % % now draw new value of  $\beta$ 
     $T \sim N(0, I_p)$ 
     $\beta[, i] \leftarrow B + L T$ 
    % % now draw new values for mixing variances
    FOR j=1 to number of observations
         $R \leftarrow (Z[j] - X[j, ] \beta[, i])$ 
         $\Lambda[j][j] \sim \pi(\lambda | R^2)$ 
    END
    % % See program A4.
END
END MCMC iterations; RETURN  $\beta$ 
```

A3: Procedure for sampling Bayesian logistic model, $\beta \sim \pi(\beta|Y, X, v)$, using joint update to $\{z, \lambda\}$.

```
% % Initialise mixing weights  $\Lambda$  to the
       $(n \times n)$  identity matrix
 $\Lambda \leftarrow I_n$ 
% % draw  $Z$  from truncated logistic
 $Z \sim Lo(0, 1)Ind(Y, Z)$ 
FOR i = 1 to number of MCMC iterations
    % % draw value of  $\beta$ 
     $V \leftarrow (X^T \Lambda^{-1} X + v^{-1})^{-1}$ 
    % % note that  $\Lambda^{-1}$  is a diagonal matrix and
    hence simple to invert
     $L \leftarrow \text{Chol}(V)$ 
    % % So  $L$  stores the lower triangular Cholesky factorisation of  $V$ 
     $B \leftarrow V X^T \Lambda^{-1} Z$ 
     $T \sim N(0, I_p)$ 
     $\beta[, i] \leftarrow B + LT$ 
    % % Now update  $\{Z, \Lambda\}$ 
    FOR j=1 to number of observations
         $m \leftarrow X[j, ] \beta[, i]$ 
        % % draw  $Z[j]$  from truncated logistic
         $Z[j] \sim Lo(m, 1)Ind(Y[j], Z[j])$ 
        % % now draw new value for mixing variance
         $R \leftarrow (Z[j] - m)$ 
         $\Lambda[j][j] \sim \pi(\lambda | R^2)$ 
        % % See program A4.
    END
END MCMC iterations; RETURN  $\beta$ 
```

Sampling the mixing weights, $\pi(\Lambda|R^2)$

In this section we describe how to sample from the full conditional distribution of the auxilliary variables $\Lambda = \{\lambda_1, \dots, \lambda_n\}$ in the logistic regression model of Section 2.3. The conditional distribution does not have a standard form, however, sampling from the density can be achieved efficiently using rejection sampling.

As a rejection sampling density we suggest using the Generalised Inverse Gaussian distribution $\text{GIG}(\lambda, \psi, \chi)$. Using the parameterisation of Devroye (1986, p. 479), we set $\lambda = 0.5$, $b\psi = 1$ and $\chi = (z_i - \mathbf{x}_i\boldsymbol{\beta})^2 = R^2$.

When sampling from the GIG we make use of the equality, $\text{GIG}(0.5, 1, r^2) = r/\text{IG}(1, r)$ where IG denotes the inverse Gaussian. The inverse Gaussian is easier to sample from than the GIG as it can be done using an inversion algorithm (p. 148, section IV, 4.3 Devroye, 1986).

Let $g(\lambda)$ denote the $\text{GIG}(0.5, 1, r^2)$ rejection sampling density, where $r^2 = (z_i - \mathbf{x}_i\boldsymbol{\beta})^2$. Following a draw from $g(\cdot)$ the sample is accepted with probability $\alpha(\cdot)$,

$$\alpha(\lambda) = \frac{l(\lambda)\pi(\lambda)}{Mg(\lambda)} \quad (18)$$

where $M \geq \sup_{\lambda} \frac{l(\lambda)\pi(\lambda)}{g(\lambda)}$, $l(\lambda)$ denotes the likelihood, $l(\lambda) \propto \lambda^{-1/2} \exp(-0.5r^2/\lambda)$, and $\pi(\lambda)$ is the prior,

$$\pi(\lambda) = \frac{1}{4}\lambda^{-1/2}KS\left(\frac{1}{2}\lambda^{1/2}\right) \quad (19)$$

where $KS(\cdot)$ denotes the Kolmogorov-Smirnov density. The prior (19) follows from the transformation of random variables $\lambda_i = (2\psi_i)^2$ in (8).

We note that we can set $M = 1$ and cancelling terms leaves the acceptance probability (18) as,

$$\alpha(\lambda) = \exp(0.5\lambda)\pi(\lambda).$$

The direct evaluation of $\alpha(\lambda)$ is problematic as the KS density is only known up to an infinite series. However, there is an alternating series representation given in Devroye (1986, p. 161-165) that allows for an efficient set of squeezing functions to be adopted for the rejection sampling algorithm. Following Devroye (1986) we partition the λ space into two regions within which we can construct a monotone alternating series. The breakpoint for this mixture method can be anywhere in the interval $[4/3, \pi^2]$. We have used the value $4/3$ as the rightmost interval is faster to evaluate.

The pseudo-code follows below. The method is least efficient as $r_i^2 \rightarrow 0$, though we still observe an acceptance rate of around 0.25 for r_i^2 as small as 10^{-10} . For $r_i^2 \approx 1$ the acceptance is around 0.5 rising to nearly one for $r_i^2 > 10$.

A4: Procedure to sample $\lambda \sim \pi(\lambda|r^2)$

REPEAT

```
% % Note,  $r^2 = (z_i - \mathbf{x}_i\beta)^2$ 
% % To begin we must
draw a sample from the rejection sampling density
/
 $Y \sim N(0, 1)$ 
 $Y \leftarrow Y^2$ 
 $Y \leftarrow 1 + (Y - \sqrt{Y(4r + Y)})/(2r)$ 
 $U \sim U[0, 1]$ 
IF  $U \leq 1/(1 + Y)$ 
  THEN  $\lambda \leftarrow r/Y$ 
  ELSE  $\lambda \leftarrow rY$ 
% % Now,  $\lambda \sim GIG(0.5, 1, r^2)$ 
 $U \sim U[0, 1]$ 
IF  $\lambda > 4/3$ 
  OK  $\leftarrow$  rightmost-interval( $U, \lambda$ )
ELSE
  OK  $\leftarrow$  leftmost-interval( $U, \lambda$ )
```

WHILE NOT OK

The procedure above calls two functions, `rightmost-interval()` and `leftmost-interval()` depending on the value of the proposed λ . The pseudo-code for these functions follows:

OK \leftarrow rightmost-interval(U, λ)

```
 $Z \leftarrow 1$ 
 $X \leftarrow \exp(-0.5\lambda)$ 
 $j \leftarrow 0$ 
REPEAT
  % % Squeezing
   $j \leftarrow j + 1$ 
   $Z \leftarrow Z - (j + 1)^2 X^{(j+1)^2 - 1}$ 
  IF  $Z > U$  THEN RETURN OK  $\leftarrow 1$ 
   $j \leftarrow j + 1$ 
```



```

 $Z \leftarrow Z + (j + 1)^2 X^{(j+1)^2-1}$ 
IF  $Z < U$  THEN RETURN OK  $\leftarrow 0$ 
END

```

The pseudo-code for the left region is

```

OK  $\leftarrow$  leftmost-interval( $U, \lambda$ )

 $H \leftarrow 0.5 \log(2) + 2.5 \log(\pi) - 2.5 \log(\lambda) - \frac{\pi^2}{2\lambda} + 0.5\lambda$ 
 $lU \leftarrow \log(U)$ 
 $Z \leftarrow 1$ 
 $X \leftarrow \exp(-\pi^2/(2\lambda))$ 
 $K \leftarrow \lambda/\pi^2$ 
 $j \leftarrow 0$ 
REPEAT
    % % Squeezing
     $j \leftarrow j + 1$ 
     $Z \leftarrow Z - KX^{j^2-1}$ 
    IF  $H + \log(Z) > lU$  THEN RETURN OK  $\leftarrow 1$ 
     $j \leftarrow j + 1$ 
     $Z \leftarrow Z + (j + 1)^2 X^{(j+1)^2-1}$ 
    IF  $H + \log(Z) < lU$  THEN RETURN OK  $\leftarrow 0$ 
END

```

A5: Procedure for sampling the Bayesian polychotomous model,

$\beta \sim \pi(\beta|Y, X, v)$, using joint update to $\{z, \lambda\}$.

```
% % Let,  $Y[i][j]$  denote the category indicator variable,  $Y[i][j] = 1$  if the  $i$ th
observation is of class  $j$ ,  $j \in \{1, \dots, Q\}$ ,  $Y[i][j] = 0$  otherwise.
% % Initialise mixing weights,  $\Lambda[:, q]$  for each category to the  $(n \times n)$  identity
matrix
FOR q=1 to  $Q - 1$ 
     $\Lambda[:, q] \leftarrow I_n$ 
    % % draw  $Z$  from truncated logistic
     $Z[:, q] \sim Lo(0, 1)Ind(Y[:, q], Z[:, q])$ 
END
FOR i = 1 to number of MCMC iterations
    FOR q=1 to  $Q - 1$ 
         $V \leftarrow (X^T \Lambda[:, q]^{-1} X + v^{-1})^{-1}$ 
        % % note that  $\Lambda^{-1}$  is a diagonal matrix and
        hence simple to invert
         $L \leftarrow \text{Chol}(V)$ 
        % % So  $L$  stores the lower triangular Cholesky factorisation of  $V$ 
         $B \leftarrow V X^T \Lambda[:, q]^{-1} Z[:, q]$ 
         $T \sim N(0, I_p)$ 
         $\beta[:, q, i] \leftarrow B + LT$ 
        % % Now update  $\{Z, \Lambda\}$ 
        FOR j=1 to number of observations
             $m \leftarrow X[j, :] \beta[:, q, i]$ 
             $C \leftarrow \text{sum}(\exp(X[j, :] \beta[:, -q, i]))$ 
            % % Hence,  $C$  records the sum of the  $Q - 2$  terms,  $\exp(X[j, :] \beta[:, t, i])$ ,
            % % for,  $t \in \{1, \dots, q - 1, q + 1, \dots, Q - 1\}$ .
            % % Now draw  $Z[j, q]$  from truncated logistic
             $Z[j, q] \sim Lo(m - \log C, 1)Ind(Y[j, q], Z[j, q])$ 
            % % now draw new value for mixing variance
             $R \leftarrow (Z[j, q] - m)$ 
             $\Lambda[j, j, q] \sim \pi(\lambda | R^2)$ 
        % % See program A4.
    END
END
END MCMC iterations; RETURN  $\beta$ 
```