Computational Statistics and Data Analysis 73 (2014) 87–102

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

journal homepage: www.elsevier.com/locate/csda

Efficient estimation of the link function parameter in a robust Bayesian binary regression model

Vivekananda Roy *

Department of Statistics, Iowa State University, Ames, IA 50011, United States

ARTICLE INFO

Article history:
Received 17 April 2013
Received in revised form 17 October 2013
Accepted 20 November 2013
Available online 4 December 2013

Keywords:
Data augmentation
Empirical Bayes
Importance sampling
Markov chain
Robit regression
Robust regression

ABSTRACT

It is known that the robit regression model for binary data is a robust alternative to the more popular probit and logistic models. The robit model is obtained by replacing the normal distribution in the probit regression model with the Student's t distribution. Unlike the probit and logistic models, the robit link has an extra degrees of freedom (df) parameter. It is shown that in practice it is important to estimate (rather than use a prespecified fixed value) the df parameter. A method for effectively selecting the df parameter of the robit model is described. The proposed method becomes computationally more effective if efficient MCMC algorithms are available for exploring the posterior distribution associated with a Bayesian robit model. Fast mixing parameter expanded DA (PX–DA) type algorithms based on an appropriate Haar measure are developed for significantly improving the convergence of DA algorithms for the robit model. The algorithms built for sampling from the Bayesian robit model shed new light on the construction of efficient PX–DA type algorithms in general. In spite of the fact that Haar PX–DA algorithms are known to be asymptotically "optimal", through an empirical study it is shown that it may take millions of iterations before they provide improvement over the DA algorithms. Contrary to the popular belief, it is demonstrated that a partially reparameterized DA algorithm can outperform a fully reparameterized DA algorithm. The proposed methodology of selecting the df parameter is illustrated through two detailed examples.

1. Introduction

Suppose $y = (y_1, y_2, \ldots, y_n)$ are n independent observations where y_i is either 0 or 1. Binary regression models using Generalized Linear Models (GLMs) assume that

$$Y_i \sim \text{Ber}(p_i),$$

where $F^{-1}(p_i) = x_i^T \beta$ for some link function $F^{-1}(\cdot)$, the x_i's, $i = 1, 2, \ldots, n$ are $p \times 1$ covariate vectors and β is the $p \times 1$ vector of regression coefficients. The two most popular choices for the link function are $F(\cdot) = \Phi(\cdot)$, the standard normal cdf which leads to the probit model and $F(\eta) = e^{\eta}/(1 + e^{\eta})$, the cdf of the standard logistic distribution which leads to the logistic model. It is well known that the estimates of regression coefficients for logistic and probit models are not robust to outliers (Pregibon, 1982). A robust alternative to logistic and probit models is obtained by assuming $F(\cdot) = F_{\nu}(\cdot)$, where F_{ν} is the cdf of the standard Student's t distribution with degrees of freedom ν (Liu, 2004). Following Liu (2004), this model is called the robit regression model. Both logit and probit link functions are well approximated by a robit link function with an appropriate degrees of freedom parameter ν (Mudholkar and George, 1978; Albert and Chib, 1993). In fact, a robit link with about seven degrees of freedom provides an excellent approximation to the logit link, and the probit link can be well

* Tel.: +1 515 294 8701; fax: +1 515 294 4040.
E-mail address: vroy@iastate.edu.

0167-9473/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.csda.2013.11.013

approximated by a robit model with large degrees of freedom. Gelman and Hill (2007, Chapter 6) showed that in the presence of outliers, the robit model, unlike logistic and probit models, can effectively downweight the discordant data points, for a better model fitting. Therefore, if the degrees of freedom parameter is chosen appropriately, the robit model will replicate the logistic or probit models if the data follows one of those models, but will provide a robust alternative when outliers are present. Here we consider a Bayesian robit model for analyzing binary data.

For a Bayesian analysis we need a prior distribution for the vector of regression coefficients β. We consider the multivariate t prior on β given by

$$\pi(\beta) = \frac{\Gamma((\nu_0+p)/2)}{\Gamma(\nu_0/2)\nu_0^{p/2}\pi^{p/2}}|\Sigma_0|^{1/2}\left[1+\nu_0^{-1}\beta^T\Sigma_0\beta\right]^{-\frac{p+\nu_0}{2}}. \tag{1.1}$$

Here, the prior for β is $t_p(0, \Sigma_0^{-1}, \nu_0)$, the p dimensional multivariate Student's t distribution with a known $p \times p$ positive definite scatter matrix Σ_0, and known degrees of freedom ν_0, centered at 0. As mentioned in Gelman et al. (2008), the t family of prior distributions allow for robust inference. Let X be the $n \times p$ design matrix whose ith row is x_i^T. Note that if $\Sigma_0 = cX^TX$ for some constant c, then $\pi(\beta)$ is the marginal prior for β under Zellner's g-prior for the normal linear model (Zellner, 1983). The posterior density is

$$\pi_\nu(\beta|y) = \frac{1}{m_\nu(y)}\ell_\nu(\beta|y) \times \pi(\beta), \tag{1.2}$$

where $\ell_\nu(\beta|y)$ is the likelihood function given by

$$\ell_\nu(\beta|y) = \prod_{i=1}^{n}\left(F_\nu(x_i'\beta)\right)^{y_i}\left(1-F_\nu(x_i'\beta)\right)^{1-y_i} \tag{1.3}$$

and $m_\nu(y) := \int_{\mathbb{R}^p}\ell_\nu(\beta|y) \times \pi(\beta)d\beta$ is the normalizing constant.

As we will see in Section 4 that in practice it is important to estimate (rather than use a prespecified fixed value) the degrees of freedom parameter ν for the robit model to provide a robust alternative to more popular logistic and probit models. Here we consider an empirical Bayes approach for estimating ν, that is, we select that value of ν which maximizes the marginal likelihood of the data $m_\nu(y)$. Henceforth we simply write m_ν instead of $m_\nu(y)$. If we are interested in a family of robit models indexed by $\nu \in \mathcal{N}$ for some set $\mathcal{N} \subset (0, \infty)$, we can calculate m_ν for all $\nu \in \mathcal{N}$ and then select that value of ν which maximizes m_ν. Note that the value of ν that maximizes m_ν, is same as the value of ν that maximizes am_ν for $\nu \in \mathcal{N}$ where a is a constant. It is often much easier to calculate am_ν than m_ν for all $\nu \in \mathcal{N}$ if a is properly chosen. For selecting models that are better than other models when ν varies across $\mathcal{N}$, we can calculate and subsequently compare the values of m_ν/m_{ν_1}, where ν_1 is a suitably chosen fixed value of the degrees of freedom parameter. (Note that, in this case $a = 1/m_{\nu_1}$.) We denote m_ν/m_{ν_1} by B_{ν,ν_1}. Ideally we would like to calculate and compare B_{ν,ν_1} for a large number of values of ν. Recently, Doss (2010) described a method based on importance sampling for selecting prior hyperparameters by estimating a large family of Bayes factors. Following Doss (2010) we consider a method that is based on importance sampling and control variates to efficiently estimate B_{ν,ν_1} for a large set of possible values of ν. The method proposed here is not specific to the robit model and can be used for any model especially when it may be difficult to effectively specify a prior on certain parameters.

Availability of fast mixing MCMC algorithms with stationary density $\pi_\nu(\beta|y)$ (for fixed ν) makes the above method of estimating ν more computationally efficient. Using the fact that the t distribution can be represented as a scale mixture of normal distributions a data augmentation (DA) algorithm can be constructed for $\pi_\nu(\beta|y)$ (see Liang et al., 2010, Section 2.4.2). It is well known that DA algorithms often converge to their stationary distributions very slowly. On the other hand, over the last decade several authors have shown that convergence of DA algorithms can be significantly improved by introducing an efficient parameter expansion step into the DA algorithm (see, e.g., Meng and van Dyk, 1999, Liu and Wu, 1999, van Dyk and Meng, 2001, Hobert and Marchev, 2008). Following these works we construct three parameter expanded DA (PX–DA) type algorithms, which are similar to the DA algorithm in terms of computational complexity. We show that *two* of these algorithms significantly improve the convergence of the DA algorithm, while the third algorithm does not show improvement over the DA algorithm. The PX–DA type algorithms that we build in the context of our Bayesian robit model shed new light on the construction of efficient PX–DA type algorithms in general. Firstly, it has been shown in the literature that the Haar PX–DA algorithms, which are PX–DA type algorithms based on appropriate Haar measures, are always asymptotically at least as efficient as their underlying DA algorithms (Roy, 2012b; Hobert and Marchev, 2008). Also it is known that Haar PX–DA algorithms are "optimal" PX–DA algorithms in the sense that the Haar PX–DA algorithm is at least as good as any other PX–DA algorithm in terms of asymptotic convergence rate and asymptotic efficiency (Hobert and Marchev, 2008) (see Section 2.2.1 for details). We construct a Haar PX–DA algorithm and empirically show that it fails to provide improvement over the DA algorithm even when the algorithm is run for several thousands of iterations. This example shows that care must be taken while constructing effective PX–DA type algorithms by introducing a parameter expansion step into the DA algorithms. Secondly, we show that a Haar PX–DA algorithm with a partial reparameterization of the augmented space can outperform a Haar PX–DA algorithm where all the augmented variables are reparameterized, which is commonly done in practice.

The rest of this article is organized as follows. In Section 2 we present different Markov chain Monte Carlo (MCMC) algorithms that can be used to explore the posterior density (1.2). Section 3 contains our methods for efficiently estimating B_{ν,ν_1}. To illustrate the methodology we present results from the analysis of two datasets in Section 4. Some concluding remarks appear in Section 5 and the proofs of the technical results are given in the Appendices.

2. MCMC algorithms

In this section we present several MCMC algorithms with stationary density $\pi_\nu(\beta|y)$ defined in (1.2). As mentioned in the Introduction a DA algorithm for $\pi_\nu(\beta|y)$ can be derived using the scale mixture representation of the t distribution. Originally Albert and Chib (1993) used this to construct a Gibbs sampler for a Bayesian robit model. More recently, Liang et al. (2010, Section 2.4.2) constructed a DA algorithm for exploring $\pi(\beta|y)$. In Section 2.1 we give the details of this construction for completeness and also since the derivation is used to construct improved DA algorithms later in this section.

2.1. DA algorithm

Let $t_\nu(\mu, 1)$ denote the univariate Student's t distribution with location μ, scale 1 and degrees of freedom ν. Following Albert and Chib (1993), let $z_i \sim t_\nu(x_i'\beta, 1)$. Then $P(Y_i = 1) = F_\nu(x_i^T\beta) = P(z_i > 0)$, that is, the z_i's can be thought as the latent variables underlying the Y_i's and we observe $Y_i = 1$ if $z_i > 0$, and otherwise we observe $Y_i = 0$. As in Albert and Chib (1993), we introduce additional random variables $\lambda_i,\ i = 1, 2, \ldots, n$ where λ_i's are independent with $\lambda_i \sim$ Gamma $(\nu/2, \nu/2)$ and note that if $z_i|\lambda_i \sim N(x_i^T\beta, 1/\lambda_i)$ then marginally we have $z_i \sim t_\nu(x_i'\beta, 1)$. Here $W \sim$ Gamma (a, b) means that the pdf of W evaluated at ω is given by $q(\omega; a, b) = b^a\omega^{a-1}e^{-b\omega}/\Gamma(a)$. Similarly the multivariate t prior on β, $\pi(\beta)$, can be represented as the following scale mixture of multivariate normal distributions:

$$\tau_0 \sim \text{Gamma}\left(\frac{\nu_0}{2}, \frac{\nu_0}{2}\right) \quad \text{and} \quad \beta|\tau_0 \sim N_p\left(0, \frac{\Sigma_0^{-1}}{\tau_0}\right).$$

Let $z = (z_1, z_2, \ldots, z_n)^T$, $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_n)^T$, $\mathbb{R}_+ = (0, \infty)$ and $\mathbb{R}_- = (-\infty, 0]$. We use z, λ and τ_0 as the *augmented data* to construct the following joint posterior density

$$\pi(\beta, (\lambda, z, \tau_0)|y) = \frac{1}{m_\nu}\left[\prod_{i=1}^n \left\{I_{\mathbb{R}_+}(z_i)I_{\{1\}}(y_i) + I_{\mathbb{R}_-}(z_i)I_{\{0\}}(y_i)\right\}\phi\left(z_i; x_i^T\beta, \frac{1}{\lambda_i}\right)q\left(\lambda_i; \frac{\nu}{2}, \frac{\nu}{2}\right)\right]$$
$$\times\, \phi_p\left(\beta; 0, (\tau_0\Sigma_0)^{-1}\right)q\left(\tau_0; \frac{\nu_0}{2}, \frac{\nu_0}{2}\right); \quad \lambda_i, \tau_0 \in \mathbb{R}_+,\ z_i \in \mathbb{R},\ \beta \in \mathbb{R}^p, \tag{2.1}$$

where $I_A(\cdot)$ is the indicator function of the set A and $\phi_p(x; a, b)$ is the density of the p dimensional normal distribution with mean a, dispersion matrix b, evaluated at the point x with $\phi \equiv \phi_1$. From the above discussion it follows that

$$\int_{\mathbb{R}_+}\int_{\mathbb{R}^n}\int_{\mathbb{R}_+^n} \pi(\beta, (\lambda, z, \tau_0)|y)d\lambda dz d\tau_0 = \pi_\nu(\beta|y),$$

that is, the β marginal density of $\pi(\beta, (\lambda, z, \tau_0)|y)$ is our target posterior density $\pi_\nu(\beta|y)$. So, we can use the complete posterior density $\pi(\beta, (\lambda, z, \tau_0)|y)$ to construct a DA algorithm for $\pi_\nu(\beta|y)$ as long as we are able to make draws from the conditional densities $\pi(\beta|\lambda, z, \tau_0, y)$ and $\pi(\lambda, z, \tau_0|\beta, y)$. Simple calculations show that

$$\beta|\lambda, z, \tau_0, y \sim N_p\left(\hat\beta, (X^T\Lambda X + \tau_0\Sigma_0)^{-1}\right),$$

where $\hat\beta = (X^T\Lambda X + \tau_0\Sigma_0)^{-1}X^T\Lambda z$ and Λ is an $n \times n$ diagonal matrix with diagonal elements $\lambda_1, \lambda_2, \ldots, \lambda_n$. It is easy to see that τ_0 and (λ, z) are conditionally independent given β and $\tau_0|\beta, y \sim$ Gamma$((\nu_0 + p)/2, (\nu_0 + \beta^T S_0\beta)/2)$. We can draw from $\pi(\lambda, z|\beta, y)$ by first drawing from $\pi(z|\beta, y)$ and then from $\pi(\lambda|z, \beta, y)$. It can be shown that conditional on (β, y), $z_1, \ldots, z_n$ are independent with $z_i|\beta, y \sim Tt_\nu(x_i^T\beta, y_i)$, where $Tt_\nu(x_i'\beta, y_i)$ denotes the truncated t distribution with ν degrees of freedom and location parameter $x_i^T\beta$ that is truncated left at 0 if $y_i = 1$ and truncated right at 0 if $y_i = 0$. Finally, conditional on (z, β, y), λ_i's are independent with $\lambda_i|z_i, \beta, y \sim$ Gamma $((\nu + 1)/2, (\nu + (z_i - x_i'\beta)^2)/2)$. Note that the order in which λ and z are drawn from $\pi(\lambda, z|\beta, y)$ is important since $\pi(\lambda|\beta, y)$ is not a standard density. So if the current state of the DA algorithm is β, then the following two steps are used to move to the next state β':

DA Algorithm:
DA (a) Draw $\{(\lambda_i, z_i),\ i = 1, 2, \ldots, n\}$ by first drawing $z_i \sim Tt_\nu(x_i'\beta, y_i)$ and then draw $\lambda_i \sim$ Gamma $\left(\frac{\nu+1}{2}, \frac{\nu+(z_i - x_i'\beta)^2}{2}\right)$ and independently draw $\tau_0 \sim$ Gamma$\left(\frac{\nu_0+p}{2}, \frac{\nu_0+\beta^T\Sigma_0\beta}{2}\right)$.
DA (b) Then draw $\beta' \sim N_p\left(\hat\beta, (X^T\Lambda X + \tau_0\Sigma_0)^{-1}\right)$.

The above DA algorithm is reversible with respect to the posterior density $\pi_v(\beta|y)$. It can be easily shown that the DA algorithm is Harris ergodic and converges to the target density (1.2) (see e.g. Hobert, 2011).

2.2. Improving the DA algorithm

As we mentioned before, the DA algorithms are often criticized for their slow convergence. In this section we consider some alternatives to the DA algorithm presented in Section 2.1.

2.2.1. Sandwich algorithms

Graphically the two steps of the DA algorithm can be viewed as $\beta \rightarrow (\lambda, z, \tau_0) \rightarrow \beta'$. It has been shown that the rate of convergence of a DA algorithm can be significantly increased by inserting a properly chosen extra step in between the two steps of the DA algorithm. The idea of introducing an extra step was developed independently by Liu and Wu (1999), who called it *parameter expanded-data augmentation* (PX–DA), and Meng and van Dyk (1999), who called it *marginal augmentation*. Following these two papers, Hobert and Marchev (2008) recently developed general versions of PX–DA type algorithms and Yu and Meng (2011) call these *sandwich algorithms* since these algorithms involve an additional step which is sandwiched between the two steps of the original DA algorithm. Graphically a sandwich algorithm can be represented as $\beta \rightarrow (\lambda, z, \tau_0) \rightarrow (\lambda', z', \tau_0') \rightarrow \beta'$, where the first and the third steps are basically the two steps of the DA algorithm and we now describe how to construct a valid middle step. Let $\pi(\lambda, z, \tau_0|y)$ be the marginal density of (λ, z, τ_0) from (2.1). The middle step $(\lambda, z, \tau_0) \rightarrow (\lambda', z', \tau_0')$ is implemented by making a draw according to *any* Markov transition function (Mtf) $R(\cdot, \cdot)$ defined on the support of the density $\pi(\lambda, z, \tau_0|y)$ and which is reversible with respect to $\pi(\lambda, z, \tau_0|y)$. Hobert and Marchev (2008) showed that the sandwich algorithm has the posterior density $\pi_v(\beta|y)$ as its stationary distribution and asymptotically the sandwich algorithm is *always* at least as good as the underlying DA algorithm in the efficiency ordering. (See Hobert and Marchev, 2008 for the definition of *efficiency ordering*.) Intuitively, the extra step R reduces the correlation between β and β' and thus improves the *mixing* of the DA algorithm. On the other hand, since the extra step in a sandwich algorithm involves more computational effort compared to the DA algorithm any gain in mixing should be weighed against this increased computational burden. Let Λ' be the $n \times n$ matrix with λ_i''s in the diagonal and $\tilde{\beta} = (X^T\Lambda' X + \tau_0'\Sigma_0)^{-1}X^T\Lambda' z'$. Then if the current state is denoted by β, a generic sandwich algorithm uses the following three steps to move to the new state β'.

A generic sandwich algorithm:
SA (a) Draw (λ, z, τ_0) as in DA (a).
SA (b) Draw $(\lambda', z', \tau_0') \sim R((\lambda, z, \tau_0), \cdot)$.
SA (c) Then draw $\beta' \sim N_p\left(\tilde{\beta}, (X^T\Lambda' X + \tau_0'\Sigma_0)^{-1}\right)$.

The three sandwich algorithms that we present in this section are all equivalent to the DA algorithm in terms of computational cost, but as we will show through empirical evidence, two of these algorithms significantly improve the mixing of the DA algorithm, while the third algorithm does not provide any improvement over the DA algorithm.

We now use Hobert and Marchev's (2008) recipe using group action to construct three viable choices for the extra step $(\lambda, z, \tau_0) \rightarrow (\lambda', z', \tau_0')$. In order to use Hobert and Marchev's (2008) recipe we need the density $\pi(\lambda, z, \tau_0|y)$. Simple calculations show that

$$\pi(\lambda, z, \tau_0|y) \propto \frac{\exp\left\{-\frac{1}{2}\left[z^T\Lambda^{1/2}(I-Q)\Lambda^{1/2}z\right]\right\}}{|X^T\Lambda X + \tau_0\Sigma_0|^{\frac{1}{2}}}|\Lambda|^{\frac{\nu+1}{2}-1}e^{-\frac{\nu}{2}\sum\lambda_i}\tau_0^{(p+\nu_0)/2-1} \times e^{-\frac{\tau_0\nu_0}{2}}\prod_{i=1}^{n}\left[I_{\mathbb{R}_+}(z_i)I_{\{1\}}(y_i) + I_{\mathbb{R}_-}(z_i)I_{\{0\}}(y_i)\right],$$

where $Q = \Lambda^{1/2}X(X^T\Lambda X + \tau_0\Sigma_0)^{-1}X^T\Lambda^{1/2}$. Now let G be the multiplicative group $\mathbb{R}_+$ which is unimodular with Haar measure $\varrho(dg) = dg/g$, where dg denotes the Lebesgue measure on $\mathbb{R}_+$. Let $\mathcal{Z}$ denote the subset of $\mathbb{R}^n$ where z lives, that is, $\mathcal{Z}$ is the Cartesian product of n positive and negative half lines ($\mathbb{R}_+$ and $\mathbb{R}_-$), where the ith component is either $\mathbb{R}_+$ (if $y_i = 1$) or $\mathbb{R}_-$ (if $y_i = 0$). As in Liu and Wu's (1999) probit regression example (see also Roy and Hobert, 2007), we let the group G act on $\mathbb{R}_+^n \times \mathcal{Z} \times \mathbb{R}_+$ (the support of the density $\pi(\lambda, z, \tau_0|y)$) through component wise multiplication, that is, we consider the transformation $(\lambda', z', \tau_0') = T_g(\lambda, z, \tau_0) = (g\lambda, gz, g\tau_0)$. If $g \in G$ and $h : \mathbb{R}_+^n \times \mathcal{Z} \times \mathbb{R}_+ \rightarrow \mathbb{R}$ is an integrable function then

$$\int_{\mathbb{R}_+^n}\int_{\mathcal{Z}}\int_{\mathbb{R}_+} h(\lambda, z, \tau_0)d\tau_0 dz d\lambda = \chi(g)\int_{\mathbb{R}_+^n}\int_{\mathcal{Z}}\int_{\mathbb{R}_+} h(T_g(\lambda, z, \tau_0))d\tau_0 dz d\lambda, \tag{2.2}$$

where $\chi(g) = g^{2n+1}$. Also it is easy to see that $\chi(g^{-1}) = 1/\chi(g)$ and $\chi(g_1 g_2) = \chi(g_1)\chi(g_2)$ for all $g, g_1, g_2 \in G$. Here $\chi(g)$ plays the role of the function j defined in Hobert and Marchev (2008, p. 543). Following Hobert and Marchev (2008), consider a density function $r(g)$ defined on G, and which depends on (λ, z, τ_0), where

$$\begin{aligned} r(g)dg &\propto \pi(g\lambda, gz, g\tau_0 \mid y)\, \chi(g)\varrho(dg) \\ &\propto g^{(3n+n\nu+\nu_0)/2-1} \exp\left\{-\frac{g}{2}\left[\left(\nu\sum_{i=1}^{n}\lambda_i + \tau_0\nu_0\right) + g^2 z^T \Lambda^{1/2}(I-Q)\Lambda^{1/2} z\right]\right\} dg. \end{aligned} \tag{2.3}$$

Since Σ_0 is positive semidefinite, we know that $(X^T\Lambda X)^{-1} - (X^T\Lambda X + \tau_0\Sigma_0)^{-1}$ is positive semidefinite (Harville, 2008, p. 438). So

$$\begin{aligned} z^T\Lambda^{1/2}(I-Q)\Lambda^{1/2}z &= z^T\Lambda z - z^T\Lambda X(X^T\Lambda X + \tau_0\Sigma_0)^{-1}X^T\Lambda z \\ &\geq z^T\Lambda z - z^T\Lambda X(X^T\Lambda X)^{-1}X^T\Lambda z \\ &= z^T\Lambda^{\frac{1}{2}}\left(I - \Lambda^{\frac{1}{2}}X(X^T\Lambda X)^{-1}X^T\Lambda^{\frac{1}{2}}\right)\Lambda^{\frac{1}{2}}z \geq 0, \end{aligned}$$

where the last inequality follows since $I - \Lambda^{\frac{1}{2}}X(X^T\Lambda X)^{-1}X^T\Lambda^{\frac{1}{2}}$ is an idempotent matrix. Since $z^T\Lambda^{1/2}(I-Q)\Lambda^{1/2}z \geq 0$, it implies that $r(g)$ is a valid density.

In the second step of our first sandwich algorithm, we make a draw $g \sim r(g)$ and set $(\lambda', z', \tau_0') = g(\lambda, z, \tau_0)$. From Hobert and Marchev (2008) we know this step is reversible with respect to $\pi(\lambda, z, \tau_0|y)$ and since $r(g)$ is log-concave we can use Gilks and Wild's (1992) adaptive rejection sampling algorithm to efficiently sample from it. As mentioned before, in the last (third) step of a sandwich algorithm, we draw β' from a multivariate normal distribution as in the second step of the DA algorithm with (λ, z, τ_0) now replaced with (λ', z', τ_0'). Straightforward calculations show that the mean, $\tilde{\beta}$, of this normal distribution can be obtained by multiplying $\hat{\beta}$ by g, i.e., $\tilde{\beta} = g\hat{\beta}$ and the dispersion matrix becomes $(X^T\Lambda' X + \tau_0'\Sigma_0)^{-1} = (X^T\Lambda X + \tau_0\Sigma_0)^{-1}/g$. As before, letting β be the current state, we use following three steps to move to the new state β'.

Sandwich Algorithm 1:
SA1 (a) Draw (λ, z, τ_0) as in DA (a).
SA1 (b) Draw $g \sim r(g)$ where $r(g)$ is given in (2.3).
SA1 (c) Then draw $\beta' \sim N_p\left(g\hat{\beta}, \frac{1}{g}(X^T\Lambda X + \tau_0\Sigma_0)^{-1}\right)$.

We now consider two other sandwich algorithms using the same multiplicative group G as before but with the group actions defined differently. If we define $T_g(\lambda, z, \tau_0) = (\lambda, gz, \tau_0)$, then (2.2) holds with $\chi(g) = g^n$. Straightforward calculations show that

$$\pi(\lambda, gz, \tau_0 \mid y)\, g^n \varrho(dg) \propto g^{n-1} \exp\left\{-\frac{g^2}{2} z^T\Lambda^{1/2}(I-Q)\Lambda^{1/2}z\right\} dg.$$

Given (λ, z, τ_0), the above defines a valid density of g. In fact given (λ, z, τ_0), we have $g^2 \sim \text{Gamma}(n/2, (z^T\Lambda^{1/2}(I - Q)\Lambda^{1/2}z)/2)$. In this case, we leave λ and τ_0 unaltered in step (b) and only the z component of the augmented vector (λ, z, τ) is changed: $z' = gz$ where g is drawn from the above density. This transformation is used in Roy (2012a) to construct a sandwich algorithm for robit model with normal prior on β. Since λ and τ_0 remain unchanged, the dispersion matrix of β' in the step (c) of the sandwich algorithm is same as in the DA algorithm and the mean $\tilde{\beta}$ becomes $g\hat{\beta}$. Again from Hobert and Marchev (2008) it follows that the above step is reversible with respect to $\pi(\lambda, z, \tau_0|y)$ and our second sandwich algorithm entails the following three steps.

Sandwich Algorithm 2:
SA2 (a) Draw (λ, z, τ_0) as in DA (a).
SA2 (b) Draw $g^2 \sim \text{Gamma}\left(\frac{n}{2}, \frac{z^T\Lambda^{1/2}(I-Q)\Lambda^{1/2}z}{2}\right)$.
SA2 (c) Then draw $\beta' \sim N_p\left(g\hat{\beta}, (X^T\Lambda X + \tau_0\Sigma_0)^{-1}\right)$.

Lastly, we consider a group action as $T_g(\lambda, z, \tau_0) = (g\lambda, z, g\tau_0)$. In this case (2.2) is satisfied if we define $\chi(g) = g^{n+1}$. Note that

$$\pi(g\lambda, z, g\tau_0 \mid y)\, g^{n+1}\varrho(dg) \propto g^{\frac{n(\nu+1)+\nu_0}{2}-1} \exp\left\{-\frac{g}{2}\left[\nu\sum_{i=1}^{n}\lambda_i + \tau_0\nu_0 + z^T\Lambda^{1/2}(I-Q)\Lambda^{1/2}z\right]\right\} dg,$$

which, given (λ, z, τ_0), is a Gamma density as a function of g. Here in step (b) we leave z unchanged and (λ, τ_0) is reparameterized. In this case, $\tilde{\beta} = \hat{\beta}$, i.e., the mean of β' as in the DA algorithm, but the dispersion matrix is changed to $(X^T \Lambda X + \tau_0 \Sigma_0)^{-1}/g$. Sandwich Algorithm 3 uses the following three steps to move from the current state β to the new state β'.

Sandwich Algorithm 3:
SA3 (a) Draw (λ, z, τ_0) as in DA (a).
SA3 (b) Draw $g \sim \text{Gamma}\left(\frac{n(\nu+1)+\nu_0}{2}, \frac{\nu\sum_{i=1}^n \lambda_i + \tau_0\nu_0 + z^T\Lambda^{1/2}(I-Q)\Lambda^{1/2}z}{2}\right)$.
SA3 (c) Then draw $\beta' \sim N_p\left(\hat{\beta}, \frac{1}{g}(X^T\Lambda X + \tau_0\Sigma_0)^{-1}\right)$.

Note that the Mtf $R(\cdot,\cdot)$ used in step (b) of each of the three sandwich algorithms presented here is reducible and $\{T_g(\lambda, z, \tau_0) : g \in G\}$ is a one dimensional subspace of $\mathbb{R}_+^n \times \mathcal{Z} \times \mathbb{R}_+$ (the support of the density $\pi(\lambda, z, \tau_0|y)$). But the resulting sandwich algorithms are irreducible and Harris ergodic with stationary density $\pi_\nu(\beta|y)$. Step (b) of each of these sandwich algorithms only requires making a single draw g from a univariate distribution—which is computationally insignificant compared to the two steps of the DA algorithm. Also, none of these three algorithms involves any *new* computationally demanding calculations such as matrix inversions. So all of the above four algorithms are essentially equivalent in terms of computational complexity. (In fact this is why we did not consider other group actions based on G, for example, the sandwich algorithm corresponding to the group action $T_g(\lambda, z, \tau_0) = (g\lambda, z, \tau_0)$ requires making draws from a very complicated density.) We compared the performance of these algorithms using simulated data. We took $n = 50$ and generated $x = (x_1, x_2, \ldots, x_{50})$ where x_i's are independent with $x_i \sim N(0, 1)$. We then simulated 50 binary variables $y_1, y_2, \ldots, y_{50}$ where $y_i \sim \text{Bernoulli}(F_\nu(\beta_0 + \beta_1 x_i))$ with $\beta_0 = 0$ and for each of the following nine combinations of true values of ν and β_1

$$(\nu, \beta_1) \in \{1, 7, 20\} \times \{1, 1.5, 2.5\}.$$

For the prior density in (1.1), we took $\nu_0 = 3$ and $\Sigma_0 = cI_2$ with $c = 0.0001$. We now state the following proposition which provides a simple condition for finiteness of moments with respect to the posterior density (1.2).

Proposition 1. *Let $r_0, r_1, \ldots, r_{p-1}$ be p nonnegative integers. If $\sum_{j=0}^{p-1} r_j < \nu_0$ then the product moment of order $r_0, r_1, \ldots, r_{p-1}$ of the posterior density (1.2) is finite.*

The proof of Proposition 1 is given in the Appendix A. Since $\nu_0 > 2$ from Proposition 1 it follows that $E(\beta_1^2|y) < \infty$ and since our MCMC algorithms are Harris ergodic, the (stationary) autocovariances can be consistently estimated by empirical autocovariance functions. Fig. 1 shows the autocorrelation functions of the draws of β_1 for all four algorithms under different true values of ν and β_1. We ran the chains longer when we needed to estimate lag-k autocovariances for larger k. For example, the plot corresponding to $\nu = 1$, $\beta_1 = 2.5$ (1st row, third plot from the left) is based on 10 million iterations of the algorithms, whereas the plot for $\nu = 20$, $\beta_1 = 2.5$ (3rd row, 3rd plot from the left) is based on 1 million iterations. Fig. 2 gives lag-one scatterplots for β_1 based on 3000 draws after the first 2000 iterations when the true values of ν and β_1 are 7 and 2.5 respectively. It is clear from the autocorrelation plots in Fig. 1 that as the true value for β_1 increases, the improvement of SA1 and SA2 over the DA algorithm becomes more significant. We see that as β_1 increases there is no considerable change in the performance of these two sandwich algorithms, whereas the DA algorithm is substantially slowed down. Also as the true value of the degrees of freedom parameter ν decreases, the mixing of SA3 and DA algorithms becomes slower. From the plots we see that autocorrelations of SA2 die down faster than that of SA1. Note that to construct SA2 we only partially reparameterized the augmented variables (λ, z, τ_0), that is, we left λ and τ_0 unaltered and only changed z, whereas SA1 is obtained by a scale transformation of the entire augmented vector, which is commonly done in practice. So we show that a Haar PX–DA algorithm with a partial reparameterization of the augmented space can outperform a Haar PX–DA algorithm where all the augmented variables are reparameterized (see also Section 2.2.2). Secondly, while our empirical autocorrelation plots show that both SA1 and SA2 dramatically improve the mixing of the DA algorithm, the third algorithm SA3 *fails* to provide an improvement over the DA algorithm. Hobert and Marchev (2008) showed that under usual regularity conditions the PX–DA algorithms are more efficient than the DA algorithms in the sense that PX–DA results in a smaller asymptotic variance for time average estimators than the DA algorithms. Fig. 3 shows the plot of $\hat{v}(\beta_1, DA)/\hat{v}(\beta_1, SA3)$, where $\hat{v}(\beta_1, DA)$ denotes the initial sequence estimator (Geyer, 1992) of the asymptotic variance of the time average estimator of β_1 based on the DA algorithm. (The batch means estimators of asymptotic variance produced similar plots.) From Fig. 3 we see that it may take millions of iterations before $\hat{v}(\beta_1, SA3)$ is less than $\hat{v}(\beta_1, DA)$. The left panel in Fig. 3 shows that even after 5 million iterations, the empirical autocovariances for SA3 are larger than those for DA (although from Fig. 1 we know that it is not the case after 10 million iterations). All of our sandwich algorithms are Haar PX–DA algorithms since these algorithms are based on the "diffuse" Haar measure on the group G, whereas a PX–DA algorithm is based on a proper density function on G. Hobert and Marchev (2008) also showed that given a group action T_g, the Haar PX–DA algorithm is at least as good as any other PX–DA algorithm in terms of both asymptotic convergence rate and asymptotic efficiency. In other words, Haar PX–DA algorithms are "optimal" parameter expanded DA algorithms (Liu and Wu, 1999;

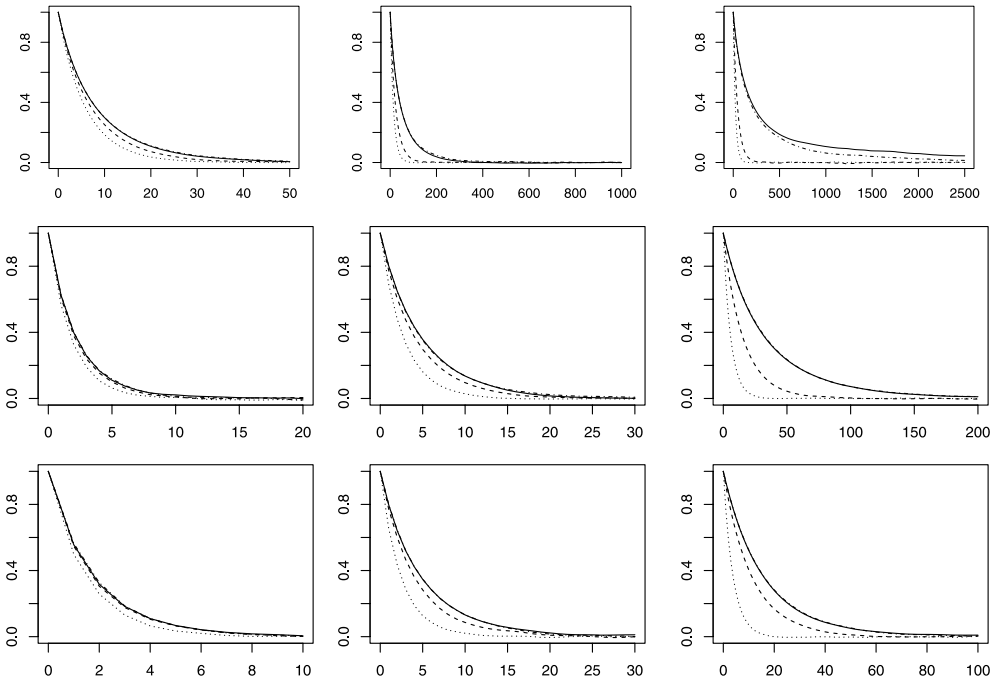

Fig. 1. Autocorrelation functions for DA (solid lines), SA1 (dashed lines), SA2 (dotted lines) and SA3 (dot dashed lines) with various values of ν and β_1: 1st row $\nu = 1$ and $\beta_1 = 1, 1.5, 2.5$ (from left to right); 2nd row $\nu = 7$ and $\beta_1 = 1, 1.5, 2.5$ (from left to right); 3rd row $\nu = 20$ and $\beta_1 = 1, 1.5, 2.5$ (from left to right).

Hobert and Marchev, 2008). So we show that an "optimal" PX–DA algorithm can fail to provide any improvement over the DA algorithm even when the algorithm is run for several thousands of iterations.

Fig. 4 plots $\sqrt{\hat{R}}$, the potential scale reduction factor (Brooks and Gelman, 1998), of the draws of β_1 based on five chains with well dispersed starting points for each of the four algorithms. Note that as in the autocorrelation plot, SA3 does not show any improvement over the DA algorithm in terms of $\sqrt{\hat{R}}$. In fact, in all three situations presented here $\sqrt{\hat{R}}$ for both SA1 and SA2 goes below 1.01 within one thousand iterations, while even after 50,000 iterations $\sqrt{\hat{R}}$ values for the other two algorithms are still above 1.36 (SA3), 1.28 (DA) (for $\beta_1 = 1$), 1.44 (SA3), 1.44 (DA) (for $\beta_1 = 1.5$) and 1.61 (SA3), 1.46 (DA) (for $\beta_1 = 2.5$) respectively.

We also applied the four MCMC algorithms described here to analyze a real dataset, namely, the Pima Indian dataset used in Ripley (1996). A population of women who were at least 21 years old, of Pima Indian heritage and living near Phoenix, Arizona, was tested for diabetes according to World Health Organization criteria. The data were collected by the US National Institute of Diabetes and Digestive and Kidney Diseases. We used the 532 complete records, selected from a larger dataset, with the binary observation denoting the presence or absence of diabetes, and the following 7 predictor variables: number of pregnancies, plasma glucose concentration in an oral glucose tolerance test, diastolic blood pressure (mm Hg), triceps skin fold thickness (mm), body mass index (weight in kg/(height in m)^2), diabetes pedigree function, and age (in years). We used all four algorithms for simulating from a robit model (1.2) with $\nu = 1.49$ for the Pima Indian data. (See Section 4.2 for the reason for taking this particular value of ν.) As before, we took $\nu_0 = 3$ and $\Sigma_0 = cI_8$ with $c = 0.0001$ for the prior density in (1.1). As in the simulated data example, we see that the sandwich algorithm SA2 performs much better than both the DA and SA3. Also, SA2 performs better than the fully reparameterized sandwich algorithm SA1. Fig. 5 shows the autocorrelation plot for the regression coefficient corresponding to glucose concentration, and the plot of Brooks and Gelman's (1998) $\sqrt{\hat{R}}$ for glucose concentration and blood pressure. We observe similar behavior in the autocorrelation plots and $\sqrt{\hat{R}}$ plots for other covariates. We also considered other values of ν, e.g. $\nu = 1, 7, 40$; SA2 again outperformed other algorithms in these situations.

2.2.2. CA-DA algorithms

To avoid the need for group action Liu (2003) describes an alternative to PX–DA known as covariance-adjusted DA (CA-DA) algorithms to accelerate DA. In every iteration of CA-DA, the augmented variables (*missing data*) (λ, z, τ_0) are simulated as in DA (a). Then to reduce autocorrelations in the 2nd step of CA-DA, the parameter (β) is drawn together with a function of the augmented variables $S(\lambda, z, \tau_0)$ from their joint distribution given a complement of $(S(\lambda, z, \tau_0), \beta)$ with respect to the space of $(\lambda, z, \tau_0, \beta)$. So compared with the DA algorithm, the CA-DA algorithm involves an extra step that *reimputes* the function $S(\lambda, z, \tau_0)$. By adjusting z with a sufficient statistic for its common scale, a CA-DA algorithm for the posterior density

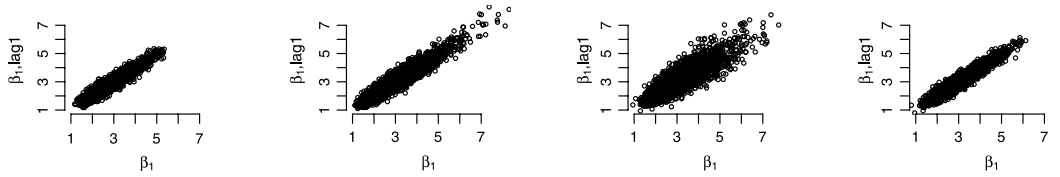

Fig. 2. Lag-one scatterplots for β_1 (corresponding to $\nu = 7$ and $\beta_1 = 2.5$) for DA, SA1, SA2 and SA3 (from left to right).

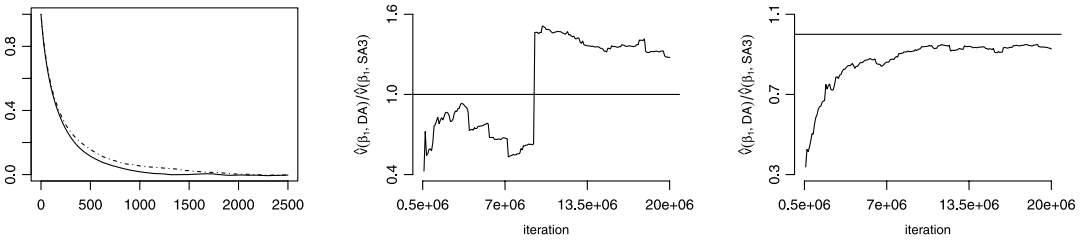

Fig. 3. Autocorrelation functions (left panel) based on 5 million iterations for DA (solid line), and SA3 (dot dashed line) corresponding to $\nu = 1$ and $\beta_1 = 2.5$. The middle and the right panel show the plots of $\hat{v}(\beta_1, DA)/\hat{v}(\beta_1, SA3)$ for $\nu = 1$, $\beta_1 = 2.5$, and 1.5 respectively.

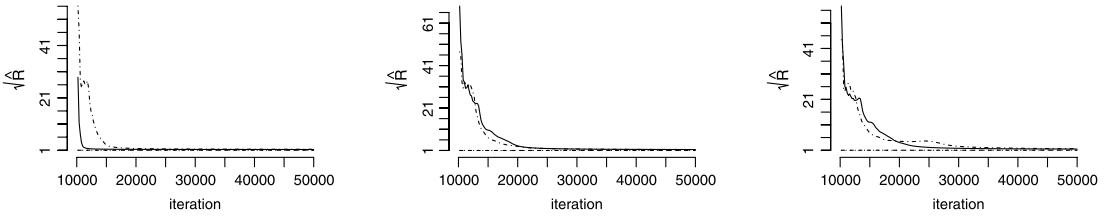

Fig. 4. Iterative $\sqrt{\hat{R}}$ plot for β_1 (from five parallel chains) corresponding to DA (solid lines), SA1 (dashed lines), SA2 (dotted lines) and SA3 (dot dashed lines) with $\nu = 1$ and various true values of $\beta_1 = 1, 1.5, 2.5$ (from left to right).

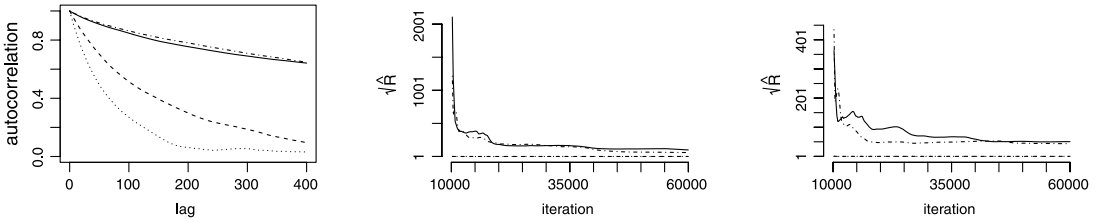

Fig. 5. Autocorrelation functions and iterative $\sqrt{\hat{R}}$ plot (from five parallel chains) for DA (solid lines), SA1 (dashed lines), SA2 (dotted lines) and SA3 (dot dashed lines) for regression coefficient corresponding to glucose concentration (left and middle panel) and $\sqrt{\hat{R}}$ plot for blood pressure (right panel).

$\pi_\nu(\beta|y)$ is derived in Liang et al. (2010, Section 2.4.2) which turns out to be the SA2 algorithm developed in Section 2.2.1. There is no theoretical study available in the literature comparing CA-DA and DA algorithms in general. However since for the robit model one of the CA-DA algorithms is equivalent to SA2, it is more efficient than the DA algorithm. Below we present another CA-DA algorithm given in Liang et al. (2010, Section 2.4.2) that is constructed by considering a transformation on the entire augmented space (λ, z, τ_0).

CA-DA Algorithm:
CA-DA (a) Draw (λ, z, τ_0) as in DA (a).
CA-DA (b) Draw $g \sim \text{Gamma}\left(\frac{n\nu+\nu_0}{2}, \frac{\nu\sum_{i=1}^{n}\lambda_i+\tau_0\nu_0}{2}\right)$.
CA-DA (c) Then draw β as in DA (b) and set $\beta' = (1/\sqrt{g})\beta$.

Following popular belief, Liang et al. (2010) claims that the above CA-DA algorithm converges faster than the other CA-DA algorithm (equivalently SA2) that only adjusts z. Fig. 6 shows that, as in Section 2.2.1, a partially adjusted CA-DA algorithm outperforms a fully adjusted CA-DA.

In the next section we discuss different methods for estimating the degrees of freedom parameter ν using the MCMC algorithms discussed in this section.

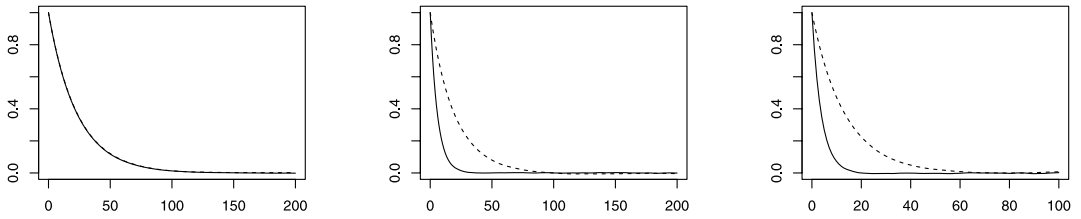

Fig. 6. Autocorrelation functions for SA2 (solid lines), and CA-DA (dashed lines) with $\beta_1 = 2.5$, and $\nu = 1, 7, 20$ (from left to right).

3. Estimation of the degrees of freedom ν

As mentioned in the Introduction, we can select appropriate value of the degrees of freedom parameter ν by comparing B_{ν,ν_1} for $\nu \in \mathcal{N}$ and a fixed ν_1. In particular, we estimate the degrees of freedom parameter by the value of ν resulting in largest value of B_{ν,ν_1}. A simple consistent estimator of B_{ν,ν_1} can be obtained as follows.

Let $\{\beta^{(i)}\}_{i=1}^N$ be a Harris ergodic Markov chain with stationary distribution $\pi_{\nu_1}(\beta|y)$. For example, any of the four MCMC algorithms presented in the previous section produce Harris ergodic Markov chains. Since $\{\beta^{(i)}\}_{i=1}^N$ is Harris ergodic, the ergodic theorem implies that, no matter what the distribution of the starting value $\beta^{(1)}$, as $N \to \infty$

$$\begin{aligned}\frac{1}{N}\sum_{i=1}^{N}\frac{\ell_\nu(\beta^{(i)}|y)}{\ell_{\nu_1}(\beta^{(i)}|y)} &\xrightarrow{\text{a.s.}} \int \frac{\ell_\nu(\beta|y)}{\ell_{\nu_1}(\beta|y)}\pi_{\nu_1}(\beta|y)d\beta \\ &= \frac{m_\nu}{m_{\nu_1}}\int \frac{\ell_\nu(\beta|y)\pi(\beta)/m_\nu}{\ell_{\nu_1}(\beta|y)\pi(\beta)/m_{\nu_1}}\pi_{\nu_1}(\beta|y)d\beta \\ &= \frac{m_\nu}{m_{\nu_1}}\int \frac{\pi_\nu(\beta|y)}{\pi_{\nu_1}(\beta|y)}\pi_{\nu_1}(\beta|y)d\beta \\ &= \frac{m_\nu}{m_{\nu_1}}. \end{aligned} \tag{3.1}$$

So B_{ν,ν_1} can be consistently estimated by (3.1) for $\nu \in \mathcal{N}$. It is important to notice that we need samples from only one posterior distribution namely $\pi_{\nu_1}(\beta|y)$ to calculate the entire family, $\{B_{\nu,\nu_1} : \nu \in \mathcal{N}\}$. As we mentioned before, ideally we would like to calculate and compare B_{ν,ν_1} for a large number of values of ν. But, when the likelihood $\ell_\nu(\beta|y)$ differs greatly from the likelihood $\ell_{\nu_1}(\beta|y)$, for example when $\ell_\nu(\beta|y)$ is nearly singular with respect to $\ell_{\nu_1}(\beta|y)$ over the region where $\beta^{(i)}$'s are likely to be, then the estimate (3.1) can be unstable.

A natural approach for dealing with the instability of the estimator in (3.1) is to choose k points $\nu_1, \nu_2, \ldots, \nu_k \in \mathcal{N}$. Then replace $\ell_{\nu_1}(\beta|y)$ in the denominator of (3.1) by a linear combination $\sum_{i=1}^k w_i\ell_{\nu_i}(\beta|y)/r_i$ with $r_i = m_{\nu_i}/m_{\nu_1}$ for $i = 2, 3, \ldots, k$, $r_1 = 1$ and appropriately chosen weights $w_i \geq 0$ such that $\sum_{i=1}^k w_i = 1$. In particular, note that

$$\int \frac{\ell_\nu(\beta|y)}{\sum_{i=1}^{k} w_i\ell_{\nu_i}(\beta|y)/r_i}\pi_{\text{mix}}(\beta|y)d\beta = B_{\nu,\nu_1},$$

where $\pi_{\text{mix}}(\beta|y) = \sum_{i=1}^k w_i\pi_{\nu_i}(\beta|y)$. Suppose that $\mathbf{r} = (r_1, r_2, \ldots, r_k)$ is known and if we have a sample (i.i.d. or a Harris ergodic Markov chain) $\{\beta^{(l)}\}_{l=1}^N$ from $\pi_{\text{mix}}(\beta|y)$. Then,

$$\frac{1}{N}\sum_{l=1}^{N}\frac{\ell_\nu(\beta^{(l)}|y)}{\sum_{i=1}^{k} w_i\ell_{\nu_i}(\beta^{(l)}|y)/r_i} \xrightarrow{\text{a.s.}} B_{\nu,\nu_1}.$$

Now assume that we have Harris ergodic Markov chain samples $\beta_j^{(l)}$, $l = 1, 2, \ldots, N_j$ from each of the posterior densities $\pi_{\nu_j}(\beta|y)$ for $j = 1, 2, \ldots, k$. Let $w_i = N_i/N$ where $N = \sum_{i=1}^k N_i$, then the pooled sample $\{\beta_j^{(l)}\}_{l,j}$ form a stratified sample from $\pi_{\text{mix}}(\beta|y)$. Define

$$\hat{B}_{\nu,\nu_1} = \sum_{j=1}^{k}\sum_{l=1}^{N_j}\frac{\ell_\nu(\beta_j^{(l)}|y)}{\sum_{i=1}^{k} N_i\ell_{\nu_i}(\beta_j^{(l)}|y)/r_i}. \tag{3.2}$$

As in Doss (2010), by the ergodic theorem we have

$$\hat{B}_{\nu,\nu_1} = \frac{1}{m_{\nu_1}} \sum_{j=1}^{k} \frac{1}{N_j} \sum_{l=1}^{N_j} \frac{\frac{N_j}{N} \ell_\nu(\beta_j^{(l)}|y)\pi(\beta)}{\sum_{i=1}^{k} \frac{N_i}{N} \ell_{\nu_i}(\beta_j^{(l)}|y)\pi(\beta)/m_{\nu_i}}$$

$$\xrightarrow{\text{a.s.}} \frac{m_\nu}{m_{\nu_1}} \sum_{j=1}^{k} \int \frac{w_j \pi_\nu(\beta|y)}{\sum_{i=1}^{k} w_i \pi_{\nu_i}(\beta|y)} \pi_{\nu_j}(\beta|y) d\beta = B_{\nu,\nu_1}.$$

So $\hat{B}_{\nu,\nu_1}$ is a consistent estimator of B_{ν,ν_1}. Note that the above asymptotics holds even if we assume $N_i/N \to w_i \in (0, 1)$. Later in this section, we discuss how to efficiently choose the sample size N_i for $i = 1, 2, \ldots, k$. Doss (2010, p. 548) gives some guidelines for choosing good values of k and the skeleton points $\nu_1, \nu_2, \ldots, \nu_k$.

In practice $\mathbf{r} = (r_1, r_2, \ldots, r_k)$ is of course unknown. Geyer (1994) proposes an estimator of $\mathbf{r}$ based on the "reverse logistic regression" method. As mentioned in Buta and Doss (2011), the estimate of $\mathbf{r}$ obtained by Geyer (1994) is same as the estimates given in Meng and Wong (1996) and Kong et al. (2003). In this paper we use a computationally fast iterative procedure developed in Meng and Wong (1996) to calculate the ratios of marginal likelihoods, $\mathbf{r}$. Let $\hat{\mathbf{r}} = (\hat{r}_1, \hat{r}_2, \ldots, \hat{r}_k)$ with $\hat{r}_1 = 1$ be the estimate of $\mathbf{r}$ produced by any of the above mentioned methods. Then B_{ν,ν_1} can be estimated by

$$\hat{B}_{\nu,\nu_1}(\hat{\mathbf{r}}) = \sum_{j=1}^{k} \sum_{l=1}^{N_j} \frac{\ell_\nu(\beta_j^{(l)}|y)}{\sum_{i=1}^{k} N_i \ell_{\nu_i}(\beta_j^{(l)}|y)/\hat{r}_i}. \tag{3.3}$$

Buta and Doss (2011) give conditions under which $\hat{B}_{\nu,\nu_1}(\hat{\mathbf{r}})$ is asymptotically normal. In particular, one of the conditions in Buta and Doss (2011) is that the Markov chains $\{\beta_j^{(l)}\}_{l=1}^{N_j}$ are geometrically ergodic for each $j = 1, 2, \ldots, k$. (See Meyn and Tweedie, 1993 for definition of geometric ergodicity.) We are not aware of any results establishing the geometric convergence of any of the MCMC algorithms presented in the previous section. But, recently Roy (2012a) showed that under certain conditions, the DA algorithm for the Bayesian robit model with a multivariate normal prior on β converges at a geometric rate. Another condition that we need for asymptotic normality of $\hat{B}_{\nu,\nu_1}(\hat{\mathbf{r}})$ is that the ratio $\ell_\nu(\beta|y)/(\sum_{i=1}^{k} w_i \ell_{\nu_i}(\beta|y)/r_i)$ has finite $2 + \epsilon$ moment with respect to $\pi_{\nu_j}(\beta|y)$ for each $j = 1, 2, \ldots, k$ for some $\epsilon > 0$. Corollary 1 (presented at the end of this section) shows that if the skeleton points $(\nu_1, \nu_2, \ldots, \nu_k)$ are properly chosen then these ratios are in fact bounded and hence have moments of all orders.

We can use control variates to improve the accuracy of $\hat{B}_{\nu,\nu_1}(\hat{\mathbf{r}})$ without increasing any computational cost. Following Doss (2010) we define

$$Y(\beta) = \frac{\ell_\nu(\beta|y)}{\sum_{i=1}^{k} w_i \ell_{\nu_i}(\beta|y)/r_i} \quad \text{and} \quad Z^{(j)}(\beta) = \frac{\ell_{\nu_j}(\beta|y)/r_j - \ell_{\nu_1}(\beta|y)}{\sum_{i=1}^{k} w_i \ell_{\nu_i}(\beta|y)/r_i} \quad \text{for } j = 2, \ldots, k. \tag{3.4}$$

Note that $E(Y(\beta)) = B_{\nu,\nu_1}$ and $E(Z^{(j)}(\beta)) = 0$ where the expectation is taken with respect to the mixture density $\pi_{\text{mix}}(\beta|y)$. For a fixed $\alpha = (\alpha_2, \ldots, \alpha_k)$, let

$$\hat{I}_\alpha = \frac{1}{N} \sum_{j=1}^{k} \sum_{l=1}^{N_j} \left(Y(\beta_j^{(l)}) - \sum_{i=2}^{k} \alpha_i Z^{(i)}(\beta_j^{(l)}) \right).$$

Note that for a fixed α, under stationarity $\hat{I}_\alpha$ is an unbiased estimator of B_{ν,ν_1}. The idea of using control variates is to consider that value of α in $\hat{I}_\alpha$ that minimizes the latter's variance. In the absence of control variates, the estimate $\hat{I}_\alpha$ simply becomes $\frac{1}{N} \sum_{j=1}^{k} \sum_{l=1}^{N_j} Y(\beta_j^{(l)})$ which is $\hat{B}_{\nu,\nu_1}$. As in Owen and Zhou (2000), the estimate of α is generally obtained by doing multiple linear regression with response variables $Y(\beta_j^{(l)})$ and predictor variables $Z^{(i)}(\beta_j^{(l)})$, $i = 2, \ldots, k$. In fact, in this case $\hat{I}_\alpha = \hat{\alpha}_0$, the least squares estimate of the intercept term if it is included in the regression and this can be easily implemented in standard statistical packages (For example, the lm() function in R can be used for this purpose R Development Core Team, 2011.). Note that the estimate $\hat{I}_\alpha$ depends on the unknown vector $\mathbf{r}$ which can be estimated by say, Geyer's (1994) reverse logistic regression as mentioned before. We denote the corresponding estimator as $\hat{I}_\alpha(\hat{\mathbf{r}})$ when $\mathbf{r}$ is replaced by $\hat{\mathbf{r}}$ in (3.4).

We now discuss how to efficiently choose the sample size N_i. We find the two-stage procedure proposed by Doss (2010) useful for estimating a large number of values of B_{ν,ν_1}; $\nu \in \mathcal{N}$. In stage I, we draw a large sample $\{\beta_j^{(l)}\}_{l=1}^{N_j}$ from $\pi_{\nu_j}(\beta|y)$, for each $j = 1, 2, \ldots, k$ which does not take much time since our MCMC algorithms are quite fast and calculate $\hat{\mathbf{r}}$ based on this sample. Independently of stage I, in stage II we get *new* samples $\{\beta_j^{(l)}\}$ from each of the posterior densities

$\pi_{\nu_j}(\beta|y),\ j = 1, 2, \ldots, k$. We then use this stage II samples to estimate B_{ν,ν_1} for all $\nu \in \mathcal{N}$ using either $\hat{B}_{\nu,\nu_1}(\hat{\mathbf{r}})$ or $\hat{I}_\alpha(\hat{\mathbf{r}})$. The reason for using this two-stage procedure is that in stage II we calculate B_{ν,ν_1} for a large number of values of ν and for each ν the amount of computation required to calculate B_{ν,ν_1} is linear in N and this rules out a large N. On the other hand, a large N_j in stage I assists in getting an accurate estimate of $\mathbf{r}$. In the next section, we give the exact breakup of total computational time for the two stages in a real data analysis.

Lastly, we present the following proposition which shows that for fixed ν_1 the ratio $\frac{\ell_\nu(\beta|y)}{\ell_{\nu_1}(\beta|y)}$ has moments of all orders with respect to any pdf of β as long as $\nu_1 \le \nu \le \infty$.

Proposition 2. *If $0 < \nu_1 < \nu \le \infty$, then the ratio $\frac{\ell_\nu(\beta|y)}{\ell_{\nu_1}(\beta|y)}$ is a bounded function of β, where $\ell_\nu(\beta|y)$ is the likelihood function of the robit model given in (1.3).*

The following corollary generalizes the above proposition to the case when the denominator is a mixture of robit likelihoods.

Corollary 1. *If there exists at least one $j' \in \{1, 2, \ldots, k\}$ such that $\nu_{j'} < \nu$, then the ratio $\frac{\ell_\nu(\beta|y)}{\sum_{j=1}^k w_j \ell_{\nu_j}(\beta|y)/r_j}$ is a bounded function of β, where w_j's and r_j's are as defined before.*

4. Illustrations using real data analysis

To illustrate our methods we consider two data sets—Finney's (1947) vaso-constriction data and the Pima Indian data used in Ripley (1996). Finney's (1947) vaso-constriction dataset seems ideal since it has been studied in several papers already so we can compare our results with previous analyses, and also because it contains some influential observations (see Section 4.1). This example shows that in the presence of outliers, the robit model along with our method for selecting its degrees of freedom parameter, provides a robust alternative to the probit and logistic regression models by effectively downweighting the influence of the outliers. Next in Section 4.2 we illustrate our methods on the Pima Indian dataset. This dataset has been used as a benchmark data in several articles. For example, recently Holmes and Held (2006) and Douc and Robert (2011) used logit and probit links respectively for analyzing the Pima Indian dataset. Recall that a robit model with around 7 degrees of freedom provides a good approximation to the logistic model, while the probit link is basically the robit link with ∞ degrees of freedom. We show that for the Pima Indian dataset our methodology selects a robit link with small (around 1.5) degrees of freedom resulting in steeper (fitted) response curves than the probit or logistic models.

4.1. Finney's (1947) vaso-constriction data

Finney's (1947) vaso-constriction data consists of 39 binary responses denoting the presence ($y = 1$) or absence ($y = 0$) of vaso-constriction of the skin of the subjects after inspiration of volume V of air at inspiration rate R. The test results were obtained from repeated measurements on three individuals with number of observations per subject being 9, 8 and 22. He found no evidence of intersubject variability and treated the data as 39 independent observations. He observed that the occurrence of a response was largely determined by the magnitude of VR, the product of volume (V) and rate (R) of inspiration and analyzed the data using the probit regression model with V and R in the logarithm scale as covariates, i.e., he considered the model $\Phi^{-1}(p_i) = \beta_0 + \beta_1 \log V_i + \beta_2 \log R_i$ for $i = 1, 2, \ldots, 39$, where $p_i = P(Y_i = 1)$. This dataset was reanalyzed by Pregibon (1982) who identified that two observations (observations 4 and 18) are influential in the maximum likelihood estimation of the logistic model and considered robust procedures (called resistant fitting methods) as alternatives to logistic regression. More recently Liu (2004) analyzed this dataset by the frequentist robit regression model, and using an EM algorithm estimated ν to be about 0.11. Here we consider a Bayesian analysis of the data using our robit model (1.2) with $\nu_0 = 3$ and $\Sigma_0 = cX^TX$ with $c = 0.0001$.

We took $k = 10$ and the skeleton set as $\{0.05, 0.1, 0.3, 0.5, 0.7, 0.9, 1.2, 2.5, 4, 8\}$. We used SA2 to get MCMC samples. All programs are written in R language (R Development Core Team, 2011). In stage I we ran 10 chains with the above degrees of freedom parameter values, each for 10,000 iterations. This took less than 3 min on an old Intel Q9550 2.83 GHz machine running Windows 7, that is, it took around one and a half seconds to draw 1000 samples. Then we used Meng and Wong's (1996) iterative procedure to estimate $\mathbf{r}$, which took around 9 min. We used $\nu_1 = 0.5$ as the base value, since preliminary MCMC runs suggested m_ν attains its maximum when ν is around 0.5. Finally, we ran 10 *new* chains each of length 1500, corresponding to the ν values as in stage I, and calculated $\hat{I}_\alpha(\hat{\mathbf{r}})$ for 513 different values of ν. (We took ν ranging from 0.05 to 5 by increments of 0.01, from 5 to 10 with increment of 0.3, and ∞.) This final step took around 10 min. The left plot in Fig. 7 suggests that a robit model with degrees of freedom around 0.5 should be used (maximum of $\hat{I}_\alpha(\hat{\mathbf{r}})$ is attained at 0.48). To obtain an estimate of the variability of our estimates, we repeated the entire procedure 15 times and the maximum root mean squared error (rmse) for $\hat{I}_\alpha(\hat{\mathbf{r}})$ over the entire range of the graph was less than 0.01. The estimates of B_{ν,ν_1} obtained by $\hat{B}_{\nu,\nu_1}(\hat{\mathbf{r}})$ were similar to the control variates estimator and the maximizing value ν was 0.48, but the maximum rmse for estimates based on $\hat{B}_{\nu,\nu_1}(\hat{\mathbf{r}})$ was 0.06.

A scatterplot of Finney's data is displayed in the right panel of Fig. 7. The lines in the figure represent contours of constant fitted probability of vaso-constriction. We used estimates of posterior means of the regression coefficients to construct the

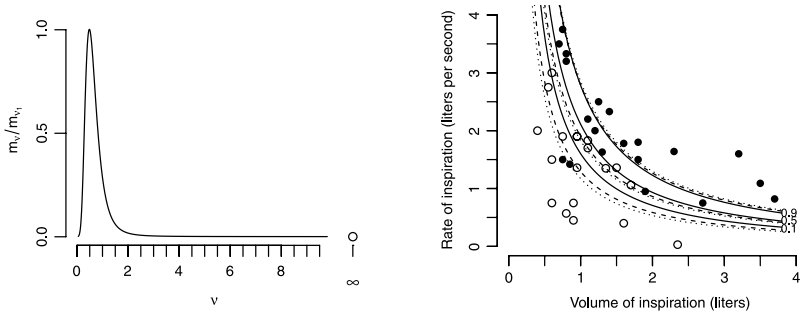

Fig. 7. The left plot shows the estimates of B_{ν,ν_1} for the vaso-constriction dataset. The plot suggests a robit link with degrees of freedom parameter around 0.5 and essentially rules out both the probit and logistic models. The right plot is the scatterplot of the vaso-constriction data (• and ∘ indicating presence and absence of vaso-constriction). The 0.1, 0.5 and 0.9 probability contours are obtained by fitting our Bayesian robit model with $\nu = 0.48$ (solid line), $\nu = 7$ (dashed line) and $\nu = 40$ (dotted line).

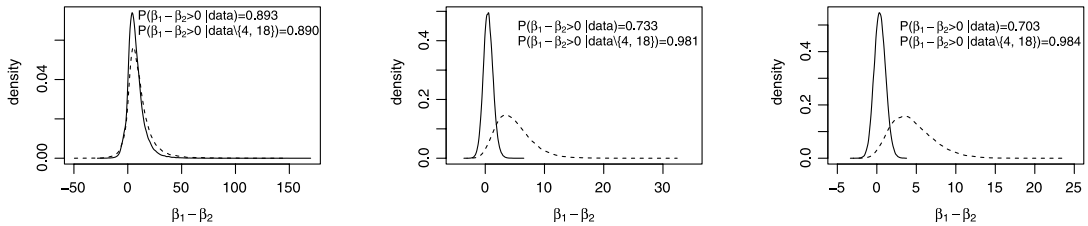

Fig. 8. Posterior densities of $\beta_1 - \beta_2$ for vaso-constriction data with (solid lines) and without (dashed lines) the observations 4 and 18 corresponding to robit models with degrees of freedom parameters $\nu = 0.48, 7, 40$ (from left to right).

probability contours. From the plot we see that probability contour plots corresponding to small degrees of freedom more successfully separates the positive and negative responses. Notice that for the robit model with $\nu = 0.48$, the 0.1 contour is much closer to its 0.5 probability contour than the robit models with larger degrees of freedoms, indicating a much steeper fitted sigmoid curve. The fitted contours corresponding to $\nu = 0.48$ are similar to those obtained by Pregibon (1982) using his resistant fitting methods. On the other hand, although the 0.5 probability contour for the robit model with $\nu = 0.11$ (Liu's (2004) estimate) is similar to that of the robit model with $\nu = 0.48$, the 0.1 and 0.9 probability contours with $\nu = 0.11$ are very different and they lie much below and above all points in the scatterplot. We also computed the Deviance Information Criterion (DIC) of Spiegelhalter et al. (2002) for both robit models with $\nu = 0.11$, and 0.48. The smaller the value of DIC, the better the model fits the data. The DIC for robit models with $\nu = 0.11$, and 0.48 are 40.79 and 28.24 respectively.

Finally, Finney (1947) was interested in determining whether a simple regression model $\Phi^{-1}(p_i) = \beta_0' + \beta_1'(\log V_i + \log R_i)$ can be considered instead of the three parameters model described above. He found the corresponding χ^2 test to be not significant, even though he questioned the validity of the χ^2 test here. Fig. 8 shows the posterior probabilities $P(\beta_1 - \beta_2 > 0|\text{data})$ and $P(\beta_1 - \beta_2 > 0|\text{data} \setminus \{4, 18\})$ for $\nu = 0.48, 7$ and 40, where "data $\setminus$ {4, 18}" signifies the subset of the dataset without observations 4 and 18. Fig. 8 demonstrates that the robit model with $\nu = 0.48$ provides robust inference for $\beta_1 - \beta_2$. For the robit model with $\nu = 0.11$ (Liu's (2004) estimate), the estimates of $P(\beta_1 - \beta_2 > 0|\text{data})$ and $P(\beta_1 - \beta_2 > 0|\text{data} \setminus \{4, 18\})$ are 0.696 and 0.760 respectively. This example shows that if the degrees of freedom parameter of the robit model is chosen by the methodology presented in this article, then the robit model provides a robust alternative to the probit and logistic regression models when outliers are present.

4.2. Diabetes in Pima Indians data

To illustrate our approach to estimation of the degrees of freedom parameter ν next we consider the Pima Indian data from Ripley (1996). As mentioned in Section 2.2.1, here we use $n = 532$ observations with 7 predictor variables mentioned before and an intercept term (i.e. $p = 8$). We took $k = 10$ and the skeleton points as {0.5, 0.7, 1, 1.2, 1.5, 2, 2.5, 3, 4, 8}. For the prior (1.1) we used $\nu_0 = 3$, $\Sigma_0 = cI_8$ with $c = 0.0001$. In stage I we ran 10 chains for 10,000 iterations each and ran SA2 to get MCMC samples. This took around three hours on the machine mentioned in Section 4.1, that is, it took around one and a half minutes to draw 1000 samples. The covariates were standardized to improve convergence of the MCMC algorithms. Meng and Wong's (1996) iterative procedure to estimate $\mathbf{r}$ took around another one hour. Here we used $\nu_1 = 0.5$ as the base value. Finally, in stage II we ran 10 new chains each of length 3000 and calculated $\hat{I}_\alpha(\hat{\mathbf{r}})$ for 306 values of ν. (These ν values range from 0.1 to 3 by increments of 0.01, from 3 to 7 with increment 0.3, and ∞.) This final step took a little over two hours. Fig. 9 shows plot $\hat{I}_\alpha(\hat{\mathbf{r}})$ values. Fig. 9 suggests a robit model with about 1.5 degree of freedom should be used (maximum of $\hat{I}_\alpha(\hat{\mathbf{r}})$ is attained at 1.49). The maximum root mean squared error (rmse) (based on 10 repetitions) for $\hat{I}_\alpha(\hat{\mathbf{r}})$ over the entire range of the graph was less than 0.3.

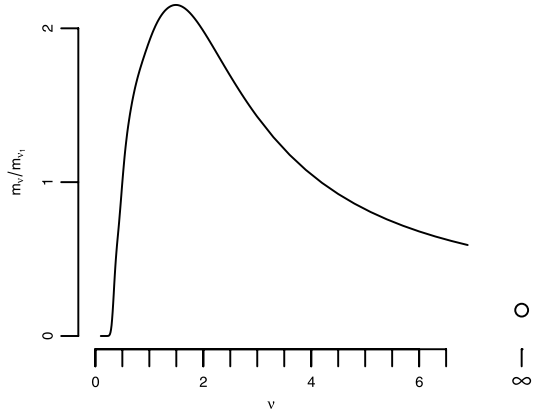

Fig. 9. Estimates of B_{ν,ν_1} for the Pima Indian dataset. The plot suggests a robit link with degrees of freedom parameter around 1.5 and essentially rules out both the probit and logistic models.

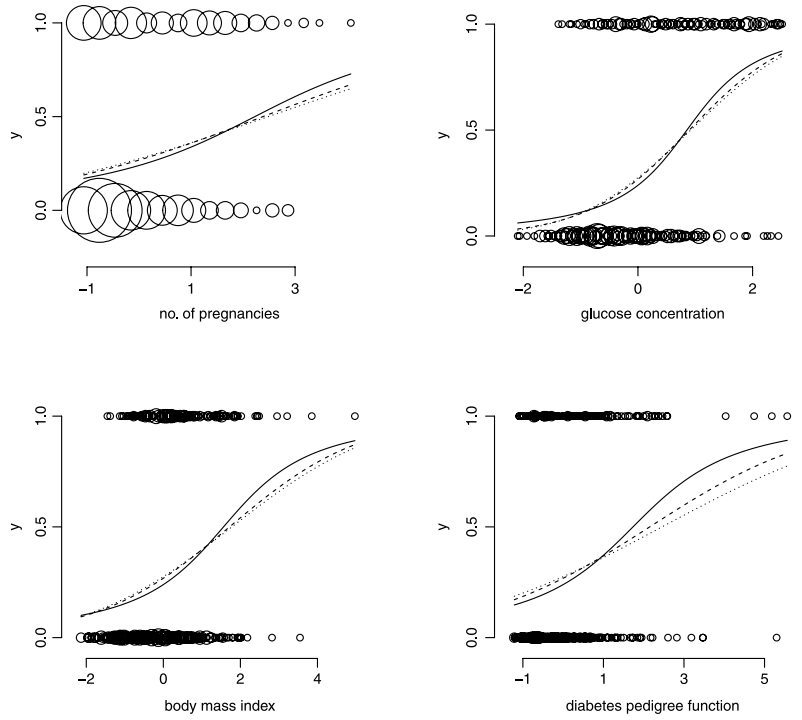

Fig. 10. Estimated response curves for Pima Indian data. Top row left panel shows the data along with the estimated response curve plotted against no. of pregnancies (controlling for other covariates by fixing them at their mean values). The size of the points indicates the number of repeats. We used our Bayesian robit model with $\nu = 1.49$ (solid line), $\nu = 7$ (dashed line) and $\nu = 40$ (dotted line). Similarly other panels show the estimated response curves plotted against other important covariates. In each case, the robit link with degrees of freedom estimated by our methodology responds more rapidly to shifts in the proportion of observed 1s for greater values of the covariates.

Fig. 10 shows the plots of the fitted response curves against the four covariates, number of pregnancies, glucose concentration, body mass index, and diabetes pedigree function. These are the four variables (out of seven covariates) which are found to be important in multiple analysis of these data (see e.g. Ripley, 1996, Holmes and Held, 2006). From the plots in Fig. 10 we see that the fitted response curves corresponding to the robit model with 1.49 degrees of freedom are steeper than the ones corresponding to the logistic and probit models. The response functions corresponding to $\nu = 1.49$ are able to curve rapidly enough to capture the observed proportion of positive responses for greater values of the covariates.

5. Discussion

One way to deal with the uncertainty regarding the choice of degrees of freedom parameter is to put a prior on it. In fact, Albert and Chib (1993) developed a Gibbs sampling algorithm for a robit model with a prior on ν. This approach falls

under so-called "Bayesian model averaging", which can be very useful. On the other hand, this might not be appropriate for all settings. First, the choice of prior might highly influence the analysis. For example, one can use an *improper* uniform prior for ν. (Note that in this case the posterior density of ν is proportional to m_ν and hence the posterior mode is exactly the point at which B_{ν,ν_1} is maximized.) It is well known that, for certain parameters, flat priors can be very informative. Here, using a flat prior on ν skews the results in favor of $\nu = \infty$ (probit model) (Doss, 2012, p. 20). Secondly, as Albert and Chib (1993) mention, it is difficult to simulate from the full conditional distribution of ν and the subsequent inference is less parsimonious and interpretable (Robert, 2007, Chapter 7). Albert and Chib (1993) were interested in a small finite number of values of ν and in a numerical example they used a uniform prior for ν on the set $\{4, 8, 16, 32\}$. The methodology proposed in this article works even when we are interested in a large number of values of ν. The value ν maximizing m_ν can also be obtained by EM-type algorithms (Liu, 2004), but, as mentioned in Doss and Hobert (2010) often it is also of interest to know the entire marginal likelihood function. In the illustrations in Section 4, from the marginal likelihood function we see that the values of $\nu = 7$ (logit case) and $\nu = \infty$ (probit case) are essentially ruled out. On the other hand, if the likelihood at the maximum were nearly the same as the likelihood at ∞ (or 7), we could use a probit (or a logistic) model if it was appropriate. Lastly, the choice of parametric link function in a GLM was also considered in Czado and Raftery (2006) who used approximate Bayes factors based on Laplace approximation to assess model fitting. The accuracy of the Laplace approximation depends on the sample size of the data, which is fixed, whereas the accuracy of our estimates depends on the length of the MCMC sample, which is in our control.

The method proposed in this article for selecting degrees of freedom parameter of robit link can be applied to many settings. Obviously, the method for estimating ν presented here can be appropriately modified to jointly estimate both ν and ν_0. The robit link is a symmetric link function. But, sometimes we may need a skewed link function to fit the dataset in hand. There are several flexible parametric families of skewed link functions available in the literature, e.g. Czado's (1994) generalized probit regression model, Kim et al. (2008) generalized skewed t-link model, and Wang and Dey's (2010) generalized extreme value model. We can use our empirical Bayes method to efficiently estimate link function parameters for these families of skewed link functions as well.

Acknowledgments

The author thanks Hani Doss, Mark Kaiser and two reviewers for helpful comments and suggestions.

Appendix A. Proof of Proposition 1

Proof of Proposition 1. Let $\beta = (\beta_0, \beta_1, \ldots, \beta_{p-1})$. The posterior product moments of $\beta_0, \beta_1, \ldots, \beta_{p-1}$ are given by

$$E\left(\prod_{j=0}^{p-1}\beta_j^{r_j}|y\right) = \int_{\mathbb{R}^p}\prod_{j=0}^{p-1}\beta_j^{r_j}\pi_\nu(\beta|y)d\beta = \frac{1}{m_\nu}\int_{\mathbb{R}^p}\ell_\nu(\beta|y)\prod_{j=0}^{p-1}\beta_j^{r_j}\pi(\beta)d\beta.$$

We know that if $u = (u_0, u_1, \ldots, u_{p-1})$ and v are independent and distributed as $u \sim N_p(0, \Sigma_0^{-1})$ and $v \sim \chi^2_{\nu_0}$, and if $\beta = u\sqrt{\nu_0/v}$, then $\beta \sim \pi(\beta)$. So

$$\int_{\mathbb{R}^p}\prod_{j=0}^{p-1}\beta_j^{r_j}\pi(\beta)d\beta = \nu_0^{\left(\sum_{j=0}^{p-1}r_j\right)/2}E\left(\prod_{j=0}^{p-1}u_j^{r_j}\right)E\left(v^{-\left(\sum_{j=0}^{p-1}r_j\right)/2}\right),$$

which is finite if $\nu_0 > \sum_{j=0}^{p-1} r_j$. Now the proof follows since the likelihood function, $\ell_\nu(\beta|y)$ is bounded. □

Appendix B. Proofs of the technical results in Section 3

Proof of Proposition 2. We first assume that $\nu < \infty$. Let $f_\nu(t)$ denote the pdf of $t_\nu(0, 1)$, i.e.,

$$f_\nu(t) = \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\Gamma\left(\frac{\nu}{2}\right)\sqrt{\nu\pi}}\left(1+\frac{t^2}{\nu}\right)^{-\frac{\nu+1}{2}}.$$

So $f_\nu(t)/f_{\nu_1}(t) = C_{\nu,\nu_1}h_{\nu,\nu_1}(t)$, where

$$h_{\nu,\nu_1}(t) = \frac{(\nu_1+t^2)^{\frac{\nu_1+1}{2}}}{(\nu+t^2)^{\frac{\nu+1}{2}}}$$

and C_{ν,ν_1} is a constant depending only on ν and ν_1. Straightforward calculations show that

$$\frac{d}{dt}\log h_{\nu,\nu_1}(t) = \frac{(\nu-\nu_1)t(1-t^2)}{(\nu+t^2)(\nu_1+t^2)},$$

and since $\nu > \nu_1$, the above implies that $\sup_{t\in\mathbb{R}} h_{\nu,\nu_1}(t) = h_{\nu,\nu_1}(1)(=h_{\nu,\nu_1}(-1)) = (\nu_1+1)^{\frac{\nu_1+1}{2}}/(\nu+1)^{\frac{\nu+1}{2}} = D_{\nu,\nu_1}$, say. So we have

$$\sup_{t\in\mathbb{R}} \frac{f_\nu(t)}{f_{\nu_1}(t)} = C_{\nu,\nu_1}\times D_{\nu,\nu_1} = \frac{f_\nu(1)}{f_{\nu_1}(1)} = M_{\nu,\nu_1}, \text{ say,}$$

and hence

$$\int_A f_\nu(t)dt \le M_{\nu,\nu_1}\int_A f_{\nu_1}(t)dt \quad \forall A\subset\mathbb{R}.$$

In particular, we have $F_\nu(t) \le M_{\nu,\nu_1}F_{\nu_1}(t)$ and $1-F_\nu(t) \le M_{\nu,\nu_1}(1-F_{\nu_1}(t))$ for all $t\in\mathbb{R}$. Finally we have

$$\frac{\ell_\nu(\beta|y)}{\ell_{\nu_1}(\beta|y)} = \prod_{i=1}^n \left(\frac{F_\nu(x_i'\beta)}{F_{\nu_1}(x_i'\beta)}\right)^{y_i}\left(\frac{1-F_\nu(x_i'\beta)}{1-F_{\nu_1}(x_i'\beta)}\right)^{1-y_i} \le (M_{\nu,\nu_1})^n.$$

If $\nu=\infty$, then $f_\nu(t)$ becomes $\phi(t)$. In this case we can similarly show that $\phi(t)/f_{\nu_1}(t) \le \phi(1)/f_{\nu_1}(1)$ and hence the lemma is proved. □

Proof of Corollary 1. The proof of the Corollary 1 follows since

$$\frac{\ell_\nu(\beta|y)}{\sum_{j=1}^k w_j\ell_{\nu_j}(\beta|y)/r_j} \le \frac{m_{\nu_{j'}}}{m_{\nu_1}}\frac{1}{w_{j'}}\frac{\ell_\nu(\beta|y)}{\ell_{\nu_{j'}}(\beta|y)} \le \frac{m_{\nu_{j'}}}{m_{\nu_1}}\frac{1}{w_{j'}}(M_{\nu,\nu_{j'}})^n,$$

where the last inequality is due to the proof of Proposition 2. □

References

Albert, J.H., Chib, S., 1993. Bayesian analysis of binary and polychotomous response data. J. Amer. Statist. Assoc. 88, 669–679.
Brooks, S.P., Gelman, A., 1998. General methods for monitoring convergence of iterative simulations. J. Comput. Graph. Statist. 7, 434–455.
Buta, E., Doss, H., 2011. Computational approaches for empirical Bayes methods and Bayesian sensitivity analysis. Ann. Statist. 39, 2658–2685.
Czado, C., 1994. Parametric link modification of both tails in binary regression. Statist. Papers 35, 189–201.
Czado, C., Raftery, A.E., 2006. Choosing the link function and accounting for link uncertainty in generalized linear models using Bayes factors. Statist. Papers 47, 419–442.
Doss, H., 2010. Estimation of large families of Bayes factors from Markov chain output. Statist. Sinica 20, 537–560.
Doss, H., 2012. Hyperparameter and model selection for nonparametric Bayes problems via Radon–Nikodym derivatives. Statist. Sinica 22, 1–26.
Doss, H., Hobert, J.P., 2010. Estimation of Bayes factors in a class of hierarchical random effects models using a geometrically ergodic MCMC algorithm. J. Comput. Graph. Statist. 19, 295–312.
Douc, R., Robert, C.P., 2011. A vanilla Rao–Blackwellization of Metropolis–Hastings algorithms. Ann. Statist. 39, 261–277.
Finney, D.J., 1947. The estimation from individual records of the relationship between dose and quantal response. Biometrika 34, 320–334.
Gelman, A., Hill, J., 2007. Data Analysis Using Regression and Multilevel/Hierarchical Models. Cambridge University Press, Cambridge.
Gelman, A., Jakulin, A., Pittau, M.G., Su, Y.-S., 2008. A weakly informative default prior distribution for logistic and other regression models. Ann. Appl. Stat. 2, 1360–1383.
Geyer, C.J., 1992. Practical Markov chain Monte Carlo (with discussion). Statist. Sci. 7, 473–511.
Geyer, C.J., 1994. Estimating Normalizing Constants and Reweighting Mixtures in Markov Chain Monte Carlo. Tech. Rep. 568. School of Statistics, University of Minnesota.
Gilks, W.R., Wild, P., 1992. Adaptive rejection sampling for Gibbs sampling. Appl. Stat. 41, 337–348.
Harville, D.A., 2008. Matrix Algebra from a Statistician's Perspective. Springer, New York, USA.
Hobert, J.P., 2011. Handbook of Markov Chain Monte Carlo. CRC Press, Boca Raton, FL, pp. 253–293 (Chapter) The data augmentation algorithm: theory and methodology.
Hobert, J.P., Marchev, D., 2008. A theoretical comparison of the data augmentation, marginal augmentation and PX-DA algorithms. Ann. Statist. 36, 532–554.
Holmes, C.C., Held, L., 2006. Bayesian auxiliary variable models for binary and multinomial regression. Bayesian Anal. 1, 145–168.
Kim, S., Chen, M.H., Dey, D.K., 2008. Flexible generalized t-link models for binary response data. Biometrika 95, 93–106.
Kong, A., McCullagh, P., Meng, X.-L., Nicolae, D., Tan, Z., 2003. A theory of statistical models for Monte Carlo integration (with discussion). J. R. Stat. Soc. Ser. B 65, 585–618.
Liang, F., Liu, C., Carroll, R.J., 2010. Advanced Markov Chain Monte Carlo Methods: Learning From Past Samples. Wiley, UK.
Liu, C., 2003. Alternating subspace-spanning resampling to accelerate Markov chain Monte Carlo simulation. J. Amer. Statist. Assoc. 98, 110–117.
Liu, C., 2004. Robit regression: a simple robust alternative to logistic and probit regression. In: Gelman, A., Meng, X.L. (Eds.), Applied Bayesian Modeling and Casual Inference from Incomplete-Data Perspectives. Wiley, London, pp. 227–238.
Liu, J.S., Wu, Y.N., 1999. Parameter expansion for data augmentation. J. Amer. Statist. Assoc. 94, 1264–1274.
Meng, X.-L., van Dyk, D.A., 1999. Seeking efficient data augmentation schemes via conditional and marginal augmentation. Biometrika 86, 301–320.
Meng, X.-L., Wong, W.H., 1996. Simulating ratios of normalizing constants via a simple identity: a theoretical exploration. Statist. Sinica 6, 831–860.
Meyn, S.P., Tweedie, R.L., 1993. Markov Chains and Stochastic Stability. Springer-Verlag, London.
Mudholkar, G.S., George, E.O., 1978. A remark on the shape of the logistic distribution. Biometrika 65, 667–668.
Owen, A., Zhou, Y., 2000. Safe and effective importance sampling. J. Amer. Statist. Assoc. 95, 135–143.
Pregibon, D., 1982. Resistant fits for some commonly used logistic models with medical applications. Biometrics 38, 485–498.

R Development Core Team 2011. R: a Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria, ISBN: 3-900051-07-0, URL: http://www.R-project.org.
Ripley, B., 1996. Pattern Recognition and Neural Networks. Cambridge University Press.
Robert, C.P., 2007. The Bayesian Choice: from Decision-Theoretic Foundations to Computational Implementation, second ed. Springer, New York.
Roy, V., 2012a. Convergence rates for MCMC algorithms for a robust Bayesian binary regression model. Electron. J. Stat. 6, 2463–2485.
Roy, V., 2012b. Spectral analytic comparisons for data augmentation. Stat. and Prob. Letters 82, 103–108.
Roy, V., Hobert, J.P., 2007. Convergence rates and asymptotic standard errors for MCMC algorithms for Bayesian probit regression. J. R. Stat. Soc. Ser. B 69, 607–623.
Spiegelhalter, D.J., Best, N.G., Carlin, B.P., Linde, A.V.D., 2002. Bayesian measures of model complexity and fit (with discussion). J. R. Stat. Soc. Ser. B 64, 583–639.
van Dyk, D.A., Meng, X.-L., 2001. The art of data augmentation (with discussion). J. Comput. Graph. Statist. 10, 1–50.
Wang, X., Dey, D.K., 2010. Generalized extreme value regression for binary response data: an application to B2B electronic payments system adoption. Ann. Appl. Stat. 4, 2000–2023.
Yu, Y., Meng, X.-L., 2011. To center or not to center: that is not the question—an ancillarity-sufficiency interweaving strategy (ASIS) for boosting MCMC efficiency. J. Comput. Graph. Statist. 20, 531–570.
Zellner, A., 1983. Applications of Bayesian analysis in econometrics. Statistician 32, 23–34.