



Journal of Statistical Software

April 2011, Volume 40, Issue 5.

<http://www.jstatsoft.org/>

DPpackage: Bayesian Semi- and Nonparametric Modeling in R

Alejandro Jara

Pontificia Universidad
Católica de Chile

Timothy E. Hanson

University of
South Carolina

Fernando A. Quintana

Pontificia Universidad
Católica de Chile

Peter Müller

University of Texas at Austin

Gary L. Rosner

The Johns Hopkins University

Abstract

Data analysis sometimes requires the relaxation of parametric assumptions in order to gain modeling flexibility and robustness against mis-specification of the probability model. In the Bayesian context, this is accomplished by placing a prior distribution on a function space, such as the space of all probability distributions or the space of all regression functions. Unfortunately, posterior distributions ranging over function spaces are highly complex and hence sampling methods play a key role. This paper provides an introduction to a simple, yet comprehensive, set of programs for the implementation of some Bayesian nonparametric and semiparametric models in R, **DPpackage**. Currently, **DPpackage** includes models for marginal and conditional density estimation, receiver operating characteristic curve analysis, interval-censored data, binary regression data, item response data, longitudinal and clustered data using generalized linear mixed models, and regression data using generalized additive models. The package also contains functions to compute pseudo-Bayes factors for model comparison and for eliciting the precision parameter of the Dirichlet process prior, and a general purpose Metropolis sampling algorithm. To maximize computational efficiency, the actual sampling for each model is carried out using compiled C, C++ or Fortran code.

Keywords: Bayesian semiparametric analysis, random probability measures, random functions, Markov chain Monte Carlo, R.

1. Introduction

In many practical situations, a parametric model cannot be expected to properly describe

the chance mechanism generating an observed dataset. Unrealistic features of some common models (e.g., the thin tails of the normal distribution when compared to the distribution of the observed data) can lead to unsatisfactory inferences. Constraining the analysis to a specific parametric form may limit the scope and type of inferences that can be drawn from such models. In these situations, we would like to relax parametric assumptions in order to gain modeling flexibility and robustness against mis-specification of a parametric statistical model. In the Bayesian context such flexible inference is typically achieved by placing a prior distribution on infinite-dimensional spaces, such as the space of all probability distributions for a random variable of interest. These models are usually referred to as Bayesian semiparametric (BSP) or nonparametric (BNP) models depending on whether the problem can be specified in such a way that the infinite-dimensional parameter θ can be written as $\theta = (\theta_1, \theta_2)$, where θ_1 is a finite-dimensional parameter and θ_2 is an infinite-dimensional parameter, or not (see, e.g., Dey, Müller, and Sinha 1998; Walker, Damien, Laud, and Smith 1999; Ghosh and Ramamoorthi 2003; Müller and Quintana 2004; Hanson, Branscum, and Johnson 2005; Hjort, Holmes, Müller, and Walker 2010).

BNP is a relatively young research area in statistics. First advances were made in the sixties and seventies, and were primarily mathematical formulations. It was only in the early nineties with the advent of sampling based methods, in particular Markov chain Monte Carlo (MCMC) methods, that substantial progress has been made. Posterior distributions ranging over function spaces are highly complex and hence sampling methods play a key role. The introduction of MCMC methods in the area began with the work of Escobar (1994) for Dirichlet process mixtures. A number of themes are still undergoing development, including issues in theory, methodology and applications. We refer to Walker *et al.* (1999), Müller and Quintana (2004), Hanson *et al.* (2005) and Hjort *et al.* (2010) for recent overviews.

While BNP and BSP are extremely powerful and have a wide range of applicability, they are not as widely used as one might expect. One reason for this has been the gap between the type of software that many users would like to have for fitting models and the software that is currently available. The most general programs currently available for Bayesian inference are **BUGS** (see, e.g., Gilks, Thomas, and Spiegelhalter 1994) and **OpenBugs** (Thomas, O'Hara, Ligges, and Sibylle 2006). **BUGS** can be accessed from the publicly available R program (R Development Core Team 2011), using the **R2WinBUGS** package (Sturtz, Ligges, and Gelman 2005). **OpenBugs** can run on Windows and Linux, as well as from inside R. In addition, various R packages exist that directly fit particular Bayesian models. We refer to Appendix C in Carlin and Louis (2008), for an extensive list of software for Bayesian modeling. Although the number of fully Bayesian programs continues to burgeon, with many available at little or no cost, they generally do not include semiparametric models. An exception to this rule is the R package **bayesm** (Rossi, Allenby, and McCulloch 2005; Rossi and McCulloch 2011), including functions for some models based on Dirichlet process priors (Ferguson 1973). The range of different Bayesian semiparametric models is huge. It is practically impossible to build flexible and efficient software for the generality of such models.

In this paper we present an up to date introduction to a publicly available R (R Development Core Team 2011) package designed to help bridge the previously mentioned gap, the **DPpackage**, originally presented in Jara (2007). Although the name of the package is due to the most widely used prior on the space of the probability distributions, the Dirichlet process (DP Ferguson 1973), the package includes many other priors on function spaces. Currently, **DPpackage** includes models considering DP (Ferguson 1973), mixtures of DP (MDP

Antoniak 1974), DP mixtures (DPM Lo 1984; Escobar and West 1995), linear dependent DP (LDDP De Iorio, Müller, Rosner, and MacEachern 2004; De Iorio, Johnson, Müller, and Rosner 2009), linear dependent Poisson-Dirichlet processes (LDPD Jara, Lesaffre, De Iorio, and Quintana 2010), weight dependent DP (WDDP Müller, Erkanli, and West 1996), hierarchical mixture of DPM of normals (HDPM Müller, Quintana, and Rosner 2004), centrally standardized DP (CSDP Newton, Czado, and Chapell 1996), Polya trees (PT Ferguson 1974; Mauldin, Sudderth, and Williams 1992; Lavine 1992, 1994), mixtures of Polya trees (MPT Lavine 1992, 1994; Hanson and Johnson 2002; Hanson 2006; Christensen, Hanson, and Jara 2008), mixtures of triangular distributions (Perron and Mengersen 2001), random Bernstein polynomials (Petrone 1999a,b; Petrone and Wasserman 2002) and dependent Bernstein polynomials (Barrientos, Jara, and Quintana 2011). The package also includes models considering penalized B-splines (Lang and Brezger 2004) and a general purpose function implementing an independence chain Metropolis-Hastings algorithm with a proposal density function generated using PT ideas (Hanson, Monteiro, and Jara 2011). The package is available from the Comprehensive R Archive Network at <http://CRAN.R-project.org/package=DPpackage>.

The article is organized as follows. Section 2 reviews the general syntax and design philosophy. Although the material in this section was presented in Jara (2007), its inclusion here is necessary in order to make the paper self-contained. In Sections 3, 4 and 5 the main features and usages of **DPpackage** are illustrated by means of simulated and real life data analyses. We conclude with additional comments and discussion in Section 6.

2. Design philosophy and general syntax

The design philosophy behind **DPpackage** is quite different from the one of a general purpose language. The most important design goal has been the implementation of model-specific MCMC algorithms. A direct benefit of this approach is that the sampling algorithms can be made dramatically more efficient than in a general purpose function based on black-box algorithms.

Fitting a model in **DPpackage** begins with a call to an R function, for instance, `DPmodel`, or `PTmodel`. Here “`model`” denotes a descriptive name for the model being fitted. Typically, the model function will take a number of arguments that control the specific MCMC sampling strategy adopted. In addition, the model(s) formula(s), data, and prior parameters are passed to the model function as arguments. The common arguments in every model function are listed next.

- (i) **prior**: An object list which includes the values of the prior hyper-parameters.
- (ii) **mcmc**: An object list which must include the integers `nburn` giving the number of burn-in scans, `nskip` giving the thinning interval, `nsave` giving the total number of scans to be saved, and `ndisplay` giving the number of saved scans to be displayed on screen, that is, the function reports on the screen when every `ndisplay` iterations have been carried out and returns the process runtime in seconds. For some specific models, one or more tuning parameters for Metropolis steps may be needed and must be included in this list. The names of these tuning parameters are explained in each specific model description in the associated help files.
- (iii) **state**: An object list giving the current value of the parameters, when the analysis is

the continuation of a previous analysis, or giving the starting values for a new Markov chain, which is useful to run multiple chains starting from different points.

- (iv) **status**: A logical variable indicating whether it is a new run (**TRUE**) or the continuation of a previous analysis (**FALSE**). In the latter case, the current value of the parameters must be specified in the object **state**.

Inside the R model function the inputs are organized in a more useable form, the MCMC sampling is performed by calling a shared library written in a compiled language, and the posterior sample is summarized, labeled, assigned into an output list, and returned. The output list includes,

- (i) **state**: An object list containing the current value of the parameters.
- (ii) **save.state**: An object list containing the MCMC samples for the parameters. This list contains two matrices **randsave** and **thetasave**, which contain the MCMC samples of the variables with random distribution (errors, random effects, etc.) and the parametric part of the model, respectively.

In order to exemplify the extraction of the output elements, consider the abstract model fit:

```
fit <- DPmodel(..., prior, mcmc, state, status, ...)
```

The lists can be extracted using the following code:

```
fit$state
fit$save.state$randsave
fit$save.state$thetasave
```

Based on these output objects, it is possible to use, for instance, the **boa** (Smith 2007) or the **coda** (Plummer, Best, Cowles, and Vines 2006) R packages to perform convergence diagnostics. For illustration, we consider the **coda** package here. It requires a matrix of posterior draws for relevant parameters to be saved as a **mcmc** object. Assume that we have obtained **fit1**, **fit2**, and **fit3**, by independently running a model function three times, specifying different starting values each. To compute the Gelman-Rubin convergence diagnostic statistic for the first parameter stored in the **thetasave** object, the following commands may be used:

```
library("coda")
coda.obj <- mcmc.list(
  chain1 = mcmc(fit1$save.state$thetasave[,1]),
  chain2 = mcmc(fit2$save.state$thetasave[,1]),
  chain3 = mcmc(fit3$save.state$thetasave[,1]))
gelman.diag(coda.obj, transform = TRUE)
```

The second command line above saves the results as a **mcmc.list** object class and the third command line computes the Gelman-Rubin statistic from these three chains.

Generic R functions such as **print**, **plot**, **summary** and **anova** have methods to display the results of the **DPpackage** model fit. The function **print** displays the posterior means of the

parameters in the model, and `summary` displays posterior summary statistics (mean, median, standard deviation, naive standard errors, and credibility intervals). By default, the function `summary` computes the 95% highest posterior density (HPD) intervals using the Monte Carlo method proposed by [Chen and Shao \(1999\)](#). The user can display the order statistic estimator of the 95% credible interval by using the following code:

```
summary(fit, hpd = FALSE)
```

The `plot` function displays the trace plots and a kernel-based estimate of the posterior distribution for the parameters of the model. Similarly to `summary`, the `plot` function displays the 95% HPD regions in the density plot and the posterior mean. The same plot but considering the 95% credible region can be obtained by using the following code:

```
plot(fit, hpd = FALSE)
```

The `anova` function computes simultaneous credible regions for a vector of parameters from the MCMC sample using the method described by [Besag, Green, Higdon, and Mengersen \(1995\)](#). The output of the `anova` function is an anova-like table containing the pseudo-contour probabilities for each of the factors included in the linear part of the model.

The following sections will show how model fitting functions in **DPpackage** are implemented in the context of density regression, survival data and education data. A complete description of the currently available functions is given in the supplementary material file.

3. Bayesian density regression

Density estimation is perhaps the most traditional inference problem addressed by BNP inference. **DPpackage** provides several models to implement density estimation, including DPM, PT, Bernstein polynomials and more. Details are described in the on-line supplementary material and in the package documentation. We describe in more detail an extension of the basic density estimation problem to density regression. Density regression is the fully nonparametric version of traditional regression problems for data $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$, where $\mathbf{x}_i \in \mathcal{X} \subset \mathbb{R}^p$ is a set of predictors, and $y_i \in \mathbb{R}$ is the response variable. Rather than assuming a functional form for the mean function and/or a common error distribution the problem is cast as inference for a family of conditional distributions

$$\{G_{\mathbf{x}} : \mathbf{x} \in \mathcal{X} \subset \mathbb{R}^p\},$$

where $y_i \mid \mathbf{x}_i \stackrel{ind.}{\sim} G_{\mathbf{x}_i}$. The current version of **DPpackage** considers several BNP models for related random probability distributions based on particular implementations of the dependent DP (DDP) proposed by [MacEachern \(1999, 2000\)](#), a natural generalization of the approach discussed by [Müller *et al.* \(1996\)](#) for nonparametric regression to the context of conditional density estimation, and the hierarchical mixture of DPM models (HDPM) proposed by [Müller *et al.* \(2004\)](#). In this section we show how to perform conditional density estimation using BNP models for related probability distributions, also referred to as Bayesian density regression, using the `DPdensity` and `LDDPdensity` functions.

3.1. The linear dependent DP model

MacEachern (1999, 2000), proposed the DDP as an approach to define a prior model for an uncountable set of random measures indexed by a single continuous covariate, say x , $\{G_x : x \in \mathcal{X} \subset \mathbb{R}\}$. The key idea behind the DDP is to create an uncountable set of DPs (Ferguson 1973) and to introduce dependence by modifying the Sethuraman (1994)'s stick-breaking representation of each element in the set. If G follows a DP prior with precision parameter α and base measure G_0 , denoted by $G \sim DP(\alpha G_0)$, then the stick-breaking representation of G is

$$G(B) = \sum_{l=1}^{\infty} \omega_l \delta_{\theta_l}(B), \quad (1)$$

where B is a measurable set, $\delta_a(\cdot)$ is the Dirac measure at a , $\theta_l \mid G_0 \stackrel{iid}{\sim} G_0$ and $\omega_l = V_l \prod_{j < l} (1 - V_j)$, with $V_l \mid \alpha \stackrel{iid}{\sim} \text{Beta}(1, \alpha)$. MacEachern (1999, 2000) generalizes (1) by assuming the point masses $\theta(x)_l$, $l = 1, \dots$, to be dependent across different levels of x , but independent across l .

De Iorio *et al.* (2004) and De Iorio *et al.* (2009) proposed a particular version of the DDP where the component of the atoms defining the location in a DDP mixture model follows a linear regression model $\theta_l(\mathbf{x}) = (\mathbf{x}^\top \boldsymbol{\beta}_l, \sigma_l^2)$, where \mathbf{x} is a p -dimensional design vector. An advantage of this model for related random probability measures, referred to as the linear DDP (LDDP), is that it can be represented as DPM of linear (in the coefficients) regression models, when the model is convolved with a normal kernel. This approach is implemented in the LDDPdensity function, where for the regression data (y_i, \mathbf{x}_i) , $i = 1, \dots, n$, the following model is considered:

$$y_i \mid G \stackrel{ind.}{\sim} \int N(y_i \mid \mathbf{x}_i^\top \boldsymbol{\beta}, \sigma^2) dG(\boldsymbol{\beta}, \sigma^2),$$

and

$$G \mid \alpha, G_0 \sim DP(\alpha G_0),$$

where $G_0 \equiv N_p(\boldsymbol{\beta} \mid \boldsymbol{\mu}_b, \mathbf{S}_b) \Gamma(\sigma^{-2} \mid \tau_1/2, \tau_2/2)$. The LDDP model specification is completed with the following hyper-priors:

$$\alpha \mid a_0, b_0 \sim \Gamma(a_0, b_0), \quad \tau_2 \mid \tau_{s_1}, \tau_{s_2} \sim \Gamma(\tau_{s_1}/2, \tau_{s_2}/2),$$

$$\boldsymbol{\mu}_b \mid \mathbf{m}_0, \mathbf{S}_0 \sim N_p(\mathbf{m}_0, \mathbf{S}_0) \quad \text{and} \quad \mathbf{S}_b \mid \nu, \boldsymbol{\Psi} \sim IW_p(\nu, \boldsymbol{\Psi}).$$

3.2. The weight dependent DP model

Let $\mathbf{x}_i^\top = (1, \mathbf{z}_i^\top)$, where \mathbf{z}_i is a p -dimensional vector of continuous predictors. The LDDP of the previous section defines a mixture model where the weights are independent of the predictors \mathbf{z} , given by

$$f_{\mathbf{z}}(\cdot) = \sum_{l=1}^{\infty} \omega_l N(\cdot \mid \beta_{0l} + \mathbf{z}^\top \boldsymbol{\beta}_l, \sigma_l^2),$$

where the weights ω_l follow a stick-breaking construction and $(\beta_{0l}, \boldsymbol{\beta}_l, \sigma_l^2) \stackrel{iid}{\sim} G_0$. Motivated by regression problems with continuous predictors, different extensions have been proposed by making the weights dependent on covariates (see, e.g., Griffin and Steel 2006; Duan, Guindani, and Gelfand 2007; Dunson, Pillai, and Park 2007a; Dunson and Park 2008), such that

$$f_{\mathbf{z}}(\cdot) = \sum_{l=1}^{\infty} \omega_l(\mathbf{z}) N(\cdot | \beta_{0l} + \mathbf{z}^\top \boldsymbol{\beta}_l, \sigma_l^2). \quad (2)$$

An earlier approach that is related to the latter references and that also induces a weight-dependent DP model, as in expression (2), was discussed by Müller *et al.* (1996). These authors fitted a “standard” DPM of multivariate Gaussian distributions to the complete data $\mathbf{d}_i = (y_i, \mathbf{z}_i)^\top$, $i = 1, \dots, n$, and looked at the induced conditional distributions. Although Müller *et al.* (1996) focused on the mean function only, $m(\mathbf{z}) = E(y|\mathbf{z})$, their method can be easily extended to provide inferences for the conditional density at covariate level \mathbf{z} , that is, a “density regression” model in the spirit of Dunson *et al.* (2007a). The extension of the approach of Müller *et al.* (1996) for related probability measures is implemented in the `DPdensity` function, where the model is given by

$$\mathbf{d}_i | G \stackrel{iid}{\sim} \int N_k(\mathbf{d}_i | \boldsymbol{\mu}, \boldsymbol{\Sigma}) dG(\boldsymbol{\mu}, \boldsymbol{\Sigma}),$$

and

$$G | \alpha, G_0 \sim DP(\alpha G_0),$$

where $k = p + 1$ is the dimension of the vector of complete data \mathbf{d}_i , the baseline distribution G_0 is the conjugate normal-inverted-Wishart (IW) distribution $G_0 \equiv N_k(\boldsymbol{\mu} | \mathbf{m}_1, \kappa_0^{-1} \boldsymbol{\Sigma}) IW_k(\boldsymbol{\Sigma} | \nu_1, \boldsymbol{\Psi}_1)$. To complete the model specification, the following hyper-priors are assumed

$$\alpha | a_0, b_0 \sim \Gamma(a_0, b_0),$$

$$\mathbf{m}_1 | \mathbf{m}_2, \mathbf{S}_2 \sim N_k(\mathbf{m}_2, \mathbf{S}_2),$$

$$\kappa_0 | \tau_1, \tau_2 \sim \Gamma(\tau_1/2, \tau_2/2),$$

and

$$\boldsymbol{\Psi}_1 | \nu_2, \boldsymbol{\Psi}_2 \sim IW_k(\nu_2, \boldsymbol{\Psi}_2).$$

This model induce a weight dependent mixture models, as in expression (2), where the components are given by

$$\omega_l(\mathbf{z}) = \frac{\omega_l N_p(\mathbf{z} | \boldsymbol{\mu}_{2l}, \boldsymbol{\Sigma}_{22l})}{\sum_{j=1}^{\infty} \omega_j N_p(\mathbf{z} | \boldsymbol{\mu}_{2j}, \boldsymbol{\Sigma}_{22j})}$$

$$\beta_{0l} = \mu_{1l} - \boldsymbol{\Sigma}_{12l} \boldsymbol{\Sigma}_{22l}^{-1} \mu_{2l},$$

$$\boldsymbol{\beta}_l = \boldsymbol{\Sigma}_{12l} \boldsymbol{\Sigma}_{22l}^{-1},$$

and

$$\sigma_l^2 = \sigma_{11l}^2 - \Sigma_{12l} \Sigma_{22l}^{-1} \Sigma_{21l},$$

where the weights ω_l follow a DP stick-breaking construction and the remaining elements arise from the standard partition of the vectors of means and (co)variance matrices given by

$$\boldsymbol{\mu}_l = \begin{pmatrix} \mu_{1l} \\ \mu_{2l} \end{pmatrix} \quad \text{and} \quad \boldsymbol{\Sigma}_l = \begin{pmatrix} \sigma_{11l}^2 & \Sigma_{12l} \\ \Sigma_{21l} & \Sigma_{22l} \end{pmatrix},$$

respectively.

The `DPdensity` function fits a marginalized version of the model, where the random probability measure G is integrated out. Full inference on the conditional density at covariate level \boldsymbol{z} is obtained by using the ϵ -DP approximation proposed by [Muliere and Tardella \(1998\)](#), with $\epsilon = 0.01$.

3.3. Simulated example

We replicate the results reported by [Dunson *et al.* \(2007a\)](#), where a different approach is proposed. Following [Dunson *et al.* \(2007a\)](#), we simulate $n = 500$ observations from a mixture of two normal linear regression models, with the mixture weights depending on the predictor, different error variances and a non-linear mean function for the second component,

$$y_i | x_i \stackrel{ind.}{\sim} \exp\{-2x_i\} N(y_i | x_i, 0.01) + (1 - \exp\{-2x_i\}) N(y_i | x_i^4, 0.04), \quad i = 1, \dots, n.$$

The predictor values x_i are simulated from a uniform distribution, $x_i \stackrel{iid}{\sim} U(0, 1)$. The following code is useful to plot the true conditional densities and the mean function:

```
R> dtrue <- function(grid, x) {
+   exp(-2 * x) * dnorm(grid, mean = x, sd = sqrt(0.01)) +
+   (1 - exp(-2 * x)) * dnorm(grid, mean = x^4, sd = sqrt(0.04))
+ }
R> mtrue <- function(x) exp(-2 * x) * x + (1 - exp(-2 * x)) * x^4
```

The data were simulated using the following code:

```
R> set.seed(0)
R> nrec <- 500
R> x <- runif(nrec)
R> y1 <- x + rnorm(nrec, 0, sqrt(0.01))
R> y2 <- x^4 + rnorm(nrec, 0, sqrt(0.04))
R> u <- runif(nrec)
R> prob <- exp(-2 * x)
R> y <- ifelse(u < prob, y1, y2)
```

The extension of the DPM of normals approach of [Müller *et al.* \(1996\)](#) considered by the `DPdensity` function, was fitted using the hyper-parameters $a_0 = 10$, $b_0 = 1$, $\nu_1 = \nu_2 = 4$, $\boldsymbol{m}_2 = (\bar{y}, \bar{x})^\top$, $\tau_1 = 6.01$, $\tau_2 = 3.01$ and $\boldsymbol{S}_2 = \boldsymbol{\Psi}_2^{-1} = 0.5\boldsymbol{S}$, where \boldsymbol{S} is the sample covariance matrix for the response and predictor. The following code illustrates how the hyper-parameters are specified:


```
R> w <- cbind(y, x)
R> wbar <- apply(w, 2, mean)
R> wcov <- var(w)
R> prior <- list(a0 = 10, b0 = 1, nu1 = 4, nu2 = 4, s2 = 0.5 * wcov,
+   m2 = wbar, psiinv2 = 2 * solve(wcov), tau1 = 6.01, tau2 = 3.01)
```

A total number of 25,000 scans of the Markov chain cycle implemented in the `DPcdensity` function were completed. A burn-in period of 5,000 samples was considered and the chain was subsampled every 4 iterates to get a final sample size of 5,000. The following code illustrate the MCMC specification:

```
R> mcmc <- list(nburn = 5000, nsave = 5000, nskip = 3, ndisplay = 1000)
```

The following commands were used to fit the model, where the conditional density estimates were evaluated on a grid of 100 points on the range of the response:

```
R> fitWDDP <- DPcdensity(y = y, x = x, xpred = seq(0, 1, 0.02),
+   ngrid = 100, compute.band = TRUE, type.band = "HPD",
+   prior = prior, mcmc = mcmc, state = NULL, status = TRUE)
```

Using the same MCMC specification, the LDDP model was also fitted to the data. The `LDDPdensity` function was used to fit a mixture of B-splines models with $\mathbf{x}^\top \boldsymbol{\beta} = \beta_0 + \sum_{j=1}^6 \psi_j(x) \beta_j$, where $\psi_k(x)$ corresponds to the k th B-spline basis function evaluated at x , as implemented in the `bs` function of the `splines` R package. The LDDP model was fitted using Zellner's g-prior (Zellner 1983), with $g = 10^3$. The following values for the hyperparameters were considered: $a_0 = 10$, $b_0 = 1$, $\mathbf{m}_0 = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$, $\mathbf{S}_0 = g (\mathbf{X}^\top \mathbf{X})^{-1}$, $\tau_1 = 6.01$, $\tau_{s1} = 6.01$, $\tau_{s2} = 2.01$, $\nu = 9$, and $\boldsymbol{\Psi}^{-1} = \mathbf{S}_0$. The following piece of code illustrate the prior specification:

```
R> library("splines")
R> W <- cbind(rep(1, nrec), bs(x, df = 6, Boundary.knots = c(0, 1)))
R> S0 <- 1000 * solve(t(W) % * % W)
R> m0 <- solve(t(W) % * % W) % * % t(W) % * % y
R> prior <- list(a0 = 10, b0 = 1, m0 = m0, S0 = S0, tau1 = 6.01,
+   taus1 = 6.01, taus2 = 2.01, nu = 9, psiinv = solve(S0))
```

The following commands were used to fit the model, where the conditional density estimates were evaluated on a grid of 100 points on the range of the response,

```
R> xpred <- seq(0, 1, 0.02)
R> Wpred <- cbind(rep(1, length(xpred)),
+   predict(bs(x, df = 6, Boundary.knots = c(0, 1)), xpred))
R> fitLDDP <- LDDPdensity(formula = y ~ W - 1, zpred = Wpred, ngrid = 100,
+   compute.band = TRUE, type.band = "HPD", prior = prior, mcmc = mcmc,
+   state = NULL, status = TRUE)
```

Figures 1 and 2 show the true density, the estimated density and pointwise 95% HPD intervals for a range of values of the predictor for the WDDP and LDDP model, respectively. The

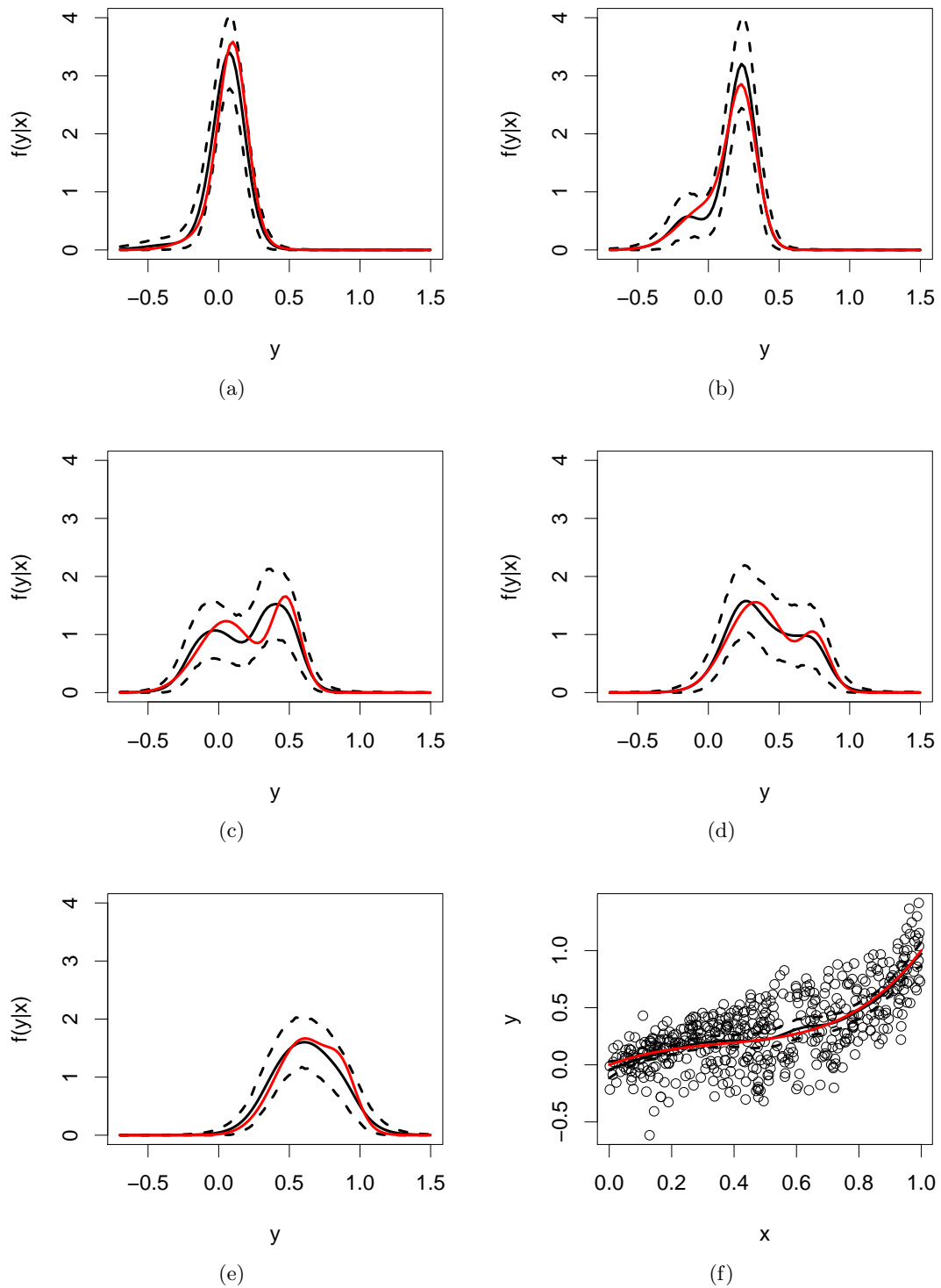


Figure 1: Simulated data - WDDP model: True conditional densities of $y|x$ (in red), posterior mean estimates (black continuous line) and pointwise 95% HPD intervals (black dashed lines) for: (a) $x = 0.1$, (b) $x = 0.25$, (c) $x = 0.48$, (d) $x = 0.76$, and (e) $x = 0.88$. Panel (f) shows the data, along with the true and estimated mean regression curves.

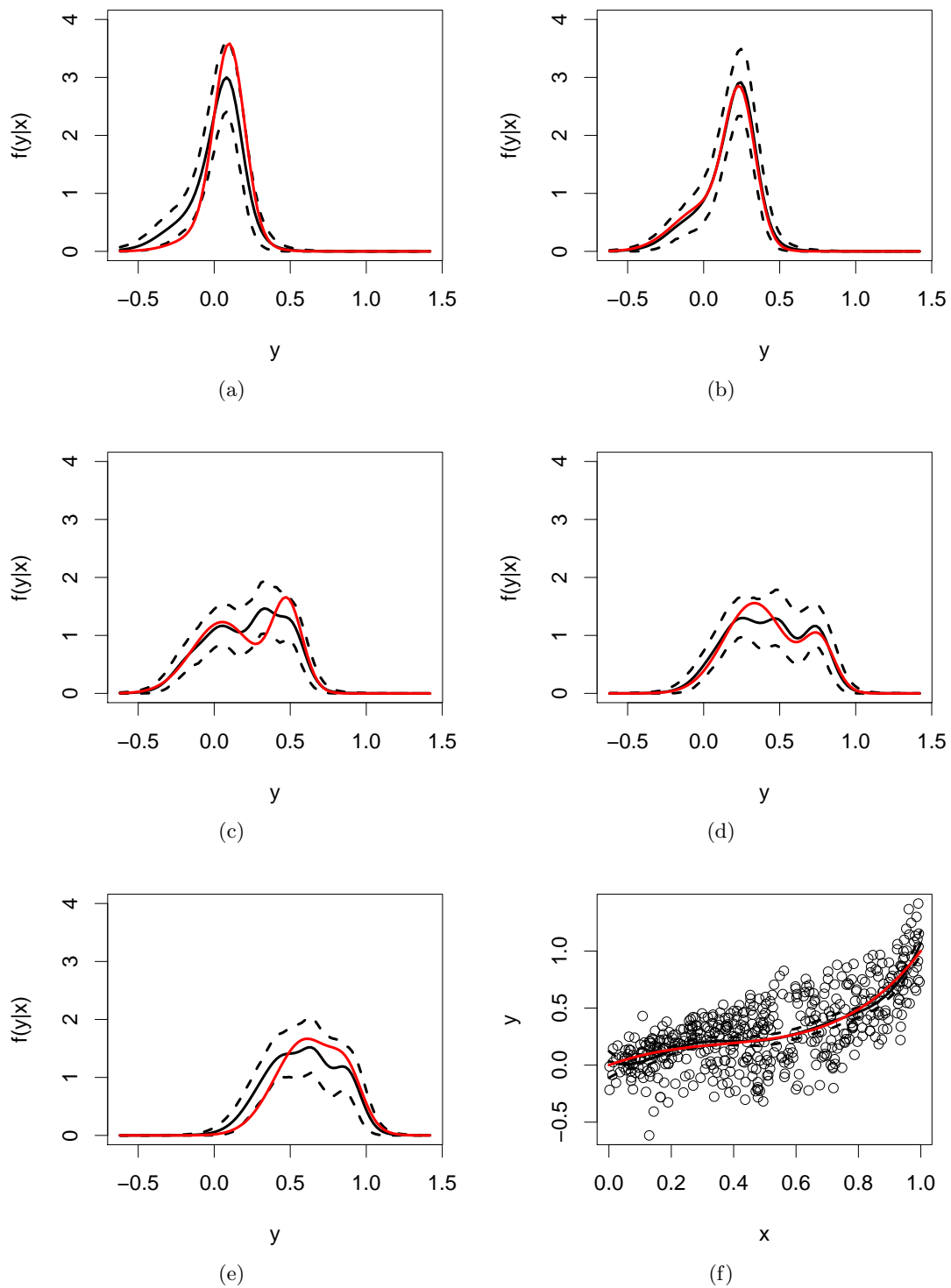


Figure 2: Simulated data - LDDP model: True conditional densities of $y|x$ (in red), posterior mean estimates (black continuous line) and pointwise 95% HPD intervals (black dashed lines) for: (a) $x = 0.1$, (b) $x = 0.25$, (c) $x = 0.48$, (d) $x = 0.76$, and (e) $x = 0.88$. Panel (f) shows the data, along with the true and estimated mean regression curves.

estimates correspond approximately to the true densities in each case. The figures also display the plot of the data along with the estimated mean function, which is very close to the true one under both models.

In both functions, the posterior mean estimates and the limits of pointwise 95% HPD intervals for the conditional density for each value of the predictors are stored in the model objects `densp.m`, and `densp.l` and `densp.h`, respectively. The following code illustrates how these objects can be used in order to get the posterior estimates for $x = 0.1$ in the LDDP model. This code was used to draw the plots displayed in Figures 1 and 2.

```
R> par(cex = 1.5, mar = c(4.1, 4.1, 1, 1))
R> plot(fitLDDP$grid, fitLDDP$densp.h[6,], lwd = 3, type = "l", lty = 2,
+      main = "", xlab = "y", ylab = "f(y|x)", ylim = c(0, 4))
R> lines(fitLDDP$grid, fitLDDP$densp.l[6,], lwd = 3, type = "l", lty = 2)
R> lines(fitLDDP$grid, fitLDDP$densp.m[6,], lwd = 3, type = "l", lty = 1)
R> lines(fitLDDP$grid, dtrue(fitLDDP$grid, xpred[6]), lwd = 3,
+      type = "l", lty = 1, col = "red")
```

Finally, both functions return the posterior mean estimates and the limits of point-wise 95% HPD intervals for the mean function in the model objects `meanfp.m`, and `meanfp.l` and `meanfp.h`, respectively. The following code was used to obtain the estimated mean function under the LDDP model, along with the true function.

```
R> par(cex = 1.5, mar = c(4.1, 4.1, 1, 1))
R> plot(x, y, xlab = "x", ylab = "y", main = "")
R> lines(xpred, fitLDDP$meanfp.m, type = "l", lwd = 3, lty = 1)
R> lines(xpred, fitLDDP$meanfp.l, type = "l", lwd = 3, lty = 2)
R> lines(xpred, fitLDDP$meanfp.h, type = "l", lwd = 3, lty = 2)
R> lines(xpred, mtrue(xpred), col = "red", lwd = 3)
```

4. Mixed-effects models with nonparametric random effects

Standard implementations of generalized linear mixed models (GLMM) typically assume independent and identically distributed random effects from a parametric distribution. Different BNP strategies have been proposed to relax the parametric assumption, including DP, DPM of normals, and PT (see, [Jara, Hanson, and Lesaffre 2009](#), for a review and methods based on PT), which are available in the current version of **DPpackage**. In this section we show how to fit a semiparametric GLMM using the `PTglm` function. The example further illustrates that a proportional-hazards model with nonparametric frailties can be fitted using any of the BNP-GLMM functions.

4.1. A semiparametric GLMM

Assume that for each of m experimental units the regression data $(Y_{ij}, \mathbf{x}_{ij}, \mathbf{z}_{ij})$, $1 \leq i \leq m$, $1 \leq j \leq n_i$, is recorded, where Y_{ij} is a response variable, and $\mathbf{x}_{ij} \in \mathbb{R}^p$ and $\mathbf{z}_{ij} \in \mathbb{R}^q$ are p - and q -dimensional design vectors, respectively. The observations are assumed to be conditionally independent with exponential family distribution,

$$p(Y_{ij} | \vartheta_{ij}, \tau) = \exp \{ [Y_{ij} \vartheta_{ij} - b(\vartheta_{ij})] / \tau \} c(Y_{ij}, \tau).$$

The means $\mu_{ij} = E(Y_{ij} | \vartheta_{ij}, \tau)$ and variances $\sigma_{ij}^2 = \text{Var}(Y_{ij} | \vartheta_{ij}, \tau)$ are related to the canonical ϑ_{ij} and dispersion parameter τ via $\mu_{ij} = b'(\vartheta_{ij})$ and $\sigma_{ij}^2 = \tau b''(\vartheta_{ij})$, respectively. The means μ_{ij} are related to the p -dimensional and q -dimensional “fixed” effects vectors β^F and β^R , respectively, and the q -dimensional “random” effects vector \mathbf{b}_i via the link relation

$$h(\mu_{ij}) = \eta_{ij} = \mathbf{x}_{ij}^\top \beta^F + \mathbf{z}_{ij}^\top \beta^R + \mathbf{z}_{ij}^\top \mathbf{b}_i, \quad (3)$$

where, $h(\cdot)$ is a known monotonic differentiable link function, and η_{ij} is called the linear predictor. Due to software limitations, the analyses are often restricted to the setting in which the random effects follow a multivariate normal distribution, $\mathbf{b}_1, \dots, \mathbf{b}_m | \Sigma \stackrel{iid}{\sim} N_q(\mathbf{0}, \Sigma)$. BNP extensions incorporate a probability model for the random effects distribution in order to better represent the distributional uncertainty and to avoid the effects of the miss-specification of an arbitrary parametric random effects distribution. Under these approaches, the parametric assumption is relaxed by considering

$$\mathbf{b}_1, \dots, \mathbf{b}_m | G \stackrel{iid}{\sim} G,$$

and

$$G | H \sim H,$$

where H is one of the previously mentioned probability models for probability distributions (e.g., DP, DPM, PT). Even though any of the BNP models can be considered with this aim, its implementation is not direct and it is necessary to discuss some important issues regarding the specification of the model. Specifically, it is important to stress that under parameterization given by expression (3), β^R represents the mean of random effects, and \mathbf{b}_i represents the subject-specific deviation from the mean. It follows that fixing the mean of the normal prior distribution for the random effects \mathbf{b} at zero in the parametric context corresponds to an identification restriction for the model parameters (see, e.g., [Newton 1994](#); [San Martín, Jara, Rolin, and Mouchart 2011](#)). Equivalently, the random probability measure must be appropriately restricted in a semiparametric GLMM specification. In our settings, the location of G is “confounded” with the parameters β^R . Although such identification issues present no difficulties for a Bayesian analysis in the sense that a prior is transformed into a posterior using the sampling model and the probability calculus, if the interest focuses on a “confounded” parameter, then such formal assurances have little practical value. Furthermore, as more data become available, the posterior mass will not concentrate on a point in the model, making asymptotic analysis difficult. As pointed out by [Newton \(1994\)](#), from a computational point of view, identification problems imply ridges in the posterior distribution and MCMC methods can be difficult to implement in these situations.

Following [Jara *et al.* \(2009\)](#), we consider the following re-parameterization of the model

$$\eta_{ij} = \mathbf{x}_{ij}^\top \beta + \mathbf{z}_{ij}^\top \boldsymbol{\theta}_i,$$

$$\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_m | G \stackrel{iid}{\sim} G,$$

and

$$G | H \sim H$$

where $\boldsymbol{\beta} = \boldsymbol{\beta}^F$, and $\boldsymbol{\theta}_i = \boldsymbol{\beta}^R + \mathbf{b}_i$, and we center the nonparametric priors for G at an $N_q(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ distribution. Notice that samples under the original parameterization can be obtained in a straightforward manner from the MCMC samples, as discussed by Jara *et al.* (2009). When a DP or DPM prior is used to model the random effects distribution, Dunson, Yang, and Baird (2007b) and Li, Müller, and Lin (2007) proposed alternative strategies to avoid the identifiability problem described above, but these approaches are not implemented in the current version of **DPpackage**.

The `PTglmm` function considers the PT prior for G as described in Jara *et al.* (2009), such that

$$G \mid \alpha, \boldsymbol{\mu}, \boldsymbol{\Sigma}, \mathbf{O} \sim PT^M(\Pi^{\boldsymbol{\mu}, \boldsymbol{\Sigma}, \mathbf{O}}, \mathcal{A}^\alpha),$$

where M is the maximum level of the partition to be updated, $\Pi^{\boldsymbol{\mu}, \boldsymbol{\Sigma}, \mathbf{O}} = \{\pi_j\}_{j \geq 0}$ is a set of partitions of \mathbb{R}^q , indexed by the centering mean $\boldsymbol{\mu}$, centering covariance matrix $\boldsymbol{\Sigma}$ and the matrix \mathbf{O} , and \mathcal{A}^α is a family of non-negative vectors controlling the variability of the process indexed by $\alpha > 0$. Here \mathbf{O} is a $q \times q$ orthogonal matrix defining the “direction” of the partition sets and the PT prior is centered around $N_q(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ distribution.

The models in `PTglmm` are completed by assuming the following prior distributions:

$$\boldsymbol{\beta} \sim N_p(\boldsymbol{\beta}_0, \mathbf{S}_{\boldsymbol{\beta}_0}),$$

$$\tau^{-1} \mid \tau_1, \tau_2 \sim \Gamma(\tau_1/2, \tau_2/2),$$

$$\boldsymbol{\mu} \mid \boldsymbol{\mu}_b, \mathbf{S}_b \sim N_q(\boldsymbol{\mu}_b, \mathbf{S}_b),$$

$$\boldsymbol{\Sigma} \mid \nu_0, \mathbf{T} \sim IW_k(\nu_0, \mathbf{T}),$$

$$\mathbf{O} \sim \text{Haar}(q),$$

and

$$\alpha \mid a_0, b_0 \sim \Gamma(a_0, b_0),$$

where Γ , IW , and $Haar$ refer to the Gamma, inverted Wishart, and Haar distributions, respectively. Notice that the inverted Wishart prior is parameterized such that $E(\boldsymbol{\Sigma}) = \mathbf{T}^{-1}/(\nu_0 - q - 1)$. Notice also that the Haar measure induces a uniform prior on the space of the orthogonal matrices.

4.2. A proportional hazards model with nonparametric frailties

We show that **DPpackage** functions for fitting GLMM can be used to fit the Cox proportional hazards model (Cox 1972) with nonparametric frailties in this section. Consider right-censored survival data where failure times are repeatedly observed within a group or subject. Let $i = 1, \dots, n$ denote the strata over which repeated times-to-event are recorded, and $j = 1, \dots, n_i$ denote the repeated observations within stratum i . The data are denoted by $\{(\mathbf{w}_{ij}, t_{ij}, \delta_{ij}) : i = 1, \dots, n, j = 1, \dots, n_i\}$, where t_{ij} is the recorded event time, $\delta_i = 1$ if t_{ij} is an observed

failure time and $\delta_{ij} = 0$ if the failure time is right censored at t_{ij} , and \mathbf{w}_{ij} is a p -dimensional vector of covariates.

The baseline hazard function $\lambda_0(t)$ corresponds to an individual with covariates $\mathbf{w} = \mathbf{0}$ and survival time T_0 . Given that the baseline individual has made it up to t , $T_0 \geq t$, the baseline hazard is how the probability of expiring in the next instant is changing. In terms of the baseline survival function $S_0(t) = P(T_0 > t)$ and density $f_0(t)$, this is given by

$$\lambda_0(t) = \lim_{\epsilon \rightarrow 0^+} \frac{P(t \leq T_0 < t + \epsilon | T_0 \geq t)}{\epsilon} = \frac{f_0(t)}{S_0(t)}.$$

The conditional proportional hazards assumption stipulates that

$$\lambda(t_{ij} | \mathbf{z}_{ij}) = \lambda_0(t) \exp(\mathbf{w}_{ij}^\top \boldsymbol{\gamma} + \theta_i),$$

where $\boldsymbol{\theta} = (\theta_1, \dots, \theta_n)^\top$ are random effects, termed *frailties* in the survival literature. Often the frailties θ_i , or exponentiated frailties e^{θ_i} , are assumed to be *iid* from some parametric distribution such as $N(0, \sigma^2)$, gamma, positive stable, etc. We consider a PT prior for the frailties distribution below.

The specification is conditional because proportionality only holds for survival times within a given group i , not across groups unless the distribution of θ_i is positive stable (see, e.g., [Qiou, Ravishanker, and Dey 1999](#)). Precisely, for individuals j_1 and j_2 within group i ,

$$\frac{\lambda(t_{ij_1} | \mathbf{w}_{ij_1})}{\lambda(t_{ij_2} | \mathbf{w}_{ij_2})} = \exp\{(\mathbf{w}_{ij_1} - \mathbf{w}_{ij_2})^\top \boldsymbol{\gamma}\}.$$

Often the baseline hazard is assumed to be piecewise constant on a partition of \mathbb{R}^+ comprised of K intervals, yielding the piecewise exponential model. References are too numerous to list. See, for instance, [Walker and Mallick \(1997\)](#), [Aslanidou, Dey, and Sinha \(1998\)](#), and [Qiou et al. \(1999\)](#). Assume that

$$\lambda_0(t) = \sum_{k=1}^K \lambda_k I\{a_{k-1} < t \leq a_k\},$$

where $a_0 = 0$ and $a_K = \infty$, although in practice $a_K = \max\{t_{ij}\}$ is sufficient. The prior hazard is specified by cutpoints $\{a_k\}_{k=0}^K$ and hazard values $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_K)^\top$. If the prior on $\boldsymbol{\lambda}$ is taken to be independent gamma distributions, the model can approximate the gamma process on a fine mesh ([Kalbfleisch 1978](#)). Regardless, the resulting model implies a Poisson likelihood for “data” y_{ijk} , taking values $y_{ijk} = 0$ when $t_{ij} \notin (a_{k-1}, a_k]$ or $\delta_{ij} = 0$, and $y_{ijk} = 1$ when $t_{ij} \in (a_{k-1}, a_k]$ and $\delta_{ij} = 1$, for $k = 1, \dots, K(t_{ij})$, where $K(t) = \max\{k : a_k \leq t\}$. The likelihood for $(\boldsymbol{\beta}, \boldsymbol{\lambda}, \boldsymbol{\gamma})$ is given by

$$\begin{aligned} \mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\lambda}, \boldsymbol{\gamma}) &= \prod_{i=1}^n \prod_{j=1}^{n_i} \left[\prod_{k=1}^{K(t_{ij})} e^{-\exp\{\log(\lambda_k) + \mathbf{w}_{ij}^\top \boldsymbol{\beta} + \gamma_i\} \Delta_{ijk}} \right] \left[e^{\log\{\lambda_{K(t_{ij})}\} + \mathbf{w}_{ij}^\top \boldsymbol{\beta} + \gamma_i} \right]^{\delta_{ij}}, \\ &\propto \prod_{i=1}^n \prod_{j=1}^{n_i} \prod_{k=1}^{K(t_{ij})} p(y_{ijk} | \mu_{ijk}), \end{aligned}$$

where $p(y|\mu)$ is the probability mass function for a $\text{Poisson}(\mu)$ random variable, $\mu_{ijk} = \exp\{\log(\lambda_k) + \mathbf{w}_{ij}^\top \boldsymbol{\beta} + \gamma_i\} \Delta_{ijk}$, and $\Delta_{ijk} = \min\{a_k, t_{ij}\} - a_{k-1}$. Thus, the Cox model assuming a piecewise constant baseline hazard can be fitted in any software which allows for Poisson regression. Note that if covariates are time dependent as well, and change only at values included in $\{a_k\}_{k=0}^K$, the likelihood is trivially extended to include \mathbf{w}_{ijk} above for $k = 1, \dots, K(t_{ij})$, rather than \mathbf{w}_{ij} .

4.3. Kidney patient data

We consider data on $n = 38$ kidney patients discussed by [McGilchrist and Aisbett \(1991\)](#). Each of the patients provides $n_i = 2$ infection times, some of which are right censored. [McGilchrist and Aisbett \(1991\)](#) found that only gender was significant, and so we follow [Aslanidou et al. \(1998\)](#), [Walker and Mallick \(1997\)](#), [Qiou et al. \(1999\)](#), and [Hemming and Shaw \(2005\)](#) in considering only this covariate in what follows. We fitted the semiparametric proportional hazards regression model using a nonparametric prior for the frailties distribution. The commands used to prepare the data to fit the model are given in the supplementary material. The original dataset, $d[i, j]$, is a 38 by 6 matrix, which for each row (from left to right) contains the subject indicator, t_{i1} , δ_{i1} , t_{i2} , δ_{i2} , and the gender indicator. Ten intervals were considered with cutpoints $\{a_1, \dots, a_{10}\}$ taken from the empirical distribution of the data. We performed the analysis using the `PTglm` function for the responses

$$\mathbf{y}_i = (y_{i11}, \dots, y_{i1K(t_{i1})}, \dots, y_{i21}, \dots, y_{i2K(t_{i2})}),$$

where \mathbf{x}_{ij} is a 11-dimensional design vector containing the gender indicator and the indicator for the interval associated with the corresponding response. Finally, we set $\boldsymbol{\beta}^\top = (\boldsymbol{\gamma}^\top, \boldsymbol{\lambda}^\top)$, and assume

$$G \sim PT^M(\Pi^{\sigma^2}, \mathcal{A}^\alpha).$$

We consider a $M = 5$ finite PT prior which was centered around a $N(0, \sigma^2)$ distribution and constrained to have median-0 (`frstlprob = TRUE` in the prior object below). The values for the hyper-parameters $\boldsymbol{\beta}_0$ and $\mathbf{S}_{\boldsymbol{\beta}_0}$ were obtained from a penalized quasi-likelihood (PQL) fit, using the `glmPQL` function available from the `MASS` package ([Venables and Ripley 2002](#)). The matrix $\mathbf{S}_{\boldsymbol{\beta}_0}$ was inflated by a factor of 100. The remaining hyper-parameters were $a_0 = b_0 = 1$, $\nu_0 = 3$, and $\mathbf{T} = \mathbf{I}_1$. The following code illustrate the prior specification:

```
R> library("MASS")
R> fit0 <- glmPQL(fixed = y ~ gender + loghazard - 1 + offset(log(off)),
+   random = ~ 1 | id, family = poisson(log))
R> beta0 <- fit0$coefficients$fixed
R> Sbeta0 <- vcov(fit0)
R> prior <- list(M = 5, a0 = 1, b0 = 1, nu0 = 3, tinv = diag(1, 1),
+   mu = rep(0, 1), beta0 = beta0, Sbeta0 = Sbeta0, frstlprob = TRUE)
```

Starting values for the model parameters were obtained from the PQL fit. The following code illustrate how the starting values are specified in the `state` object.

```
R> beta <- fit0$coefficients$fixed
R> b <- as.vector(fit0$coefficients$random$id)
```



```
R> mu <- rep(0, 1)
R> sigma <- getVarCov(fit0)[1, 1]
R> state <- list(alpha = 1, beta = beta, b = b, mu = mu, sigma = sigma)
```

A single Markov chain cycle of length 25,000 was completed. The full chain was sub-sampled every 20 steps after a burn in period of 5,000 samples, to give a reduced chain of length 5,000. In the code below, `tune3 = 1.5`, corresponds to the standard deviation of the lognormal candidate generating distribution used in the Metropolis-Hastings step for the precision parameter of the PT.

```
R> mcmc <- list(nburn = 5000, nsave = 5000, nskip = 19, ndisplay = 1000,
+   tune3 = 1.5)
```

The code for fitting the model, extracting the posterior inferences and the frailties densities estimates using `PTglmm` is given below.

```
R> fitPT <- PTglmm(fixed = y ~ gender + loghazard,
+   offset = log(off), random = ~ 1 | id, family = poisson(log),
+   prior = prior, mcmc = mcmc, state = state, status = FALSE)
R> summary(fitPT)
R> predPT <- PTrandom(fitPT, predictive = TRUE, grid1 = c(-2.3, 2.3))
R> plot(predPT)
```

The abridged output is given below. The output lists the estimated effect for gender $\hat{\beta}_1 = -1.13$ followed by $K = 10$ estimated log-hazard values. Notice that the intercept term in the posterior information for the “fixed” effects (regression coefficients in the output), corresponds to the mean of the frailties distribution G . The posterior median estimate of the centering variance was $\hat{\sigma}^2 = 0.35$ and close to the posterior median of the frailties variance (0.33). Further, the posterior median (95% credible interval) for α was 0.75 (0.04; 3.77). The trace plots of the parameters (not shown) indicate a good mixing of the chain. The acceptance rates for the Metropolis-Hastings steps associated with the regression coefficients, frailties, centering variance and precision parameter was 36, 61, 43 and 0.46%, respectively. Notice that the 0 values for the acceptance rates in the output corresponds to the centering mean, which is not sampled, and the decomposition of the centering covariance matrix. The latter is only sampled for dimensions greater or equal than 2.

Walker and Mallick (1997) analyzed these data with piecewise exponential model and frailties following a PT with fixed centering variance, $PT^8(\Pi^{100}, \mathcal{A}^{0.1})$ and find $\hat{\beta}_1 = -1.0$. McGilchrist and Aisbett (1991) obtain $\hat{\beta}_1 = -1.8$, but with other nonsignificant covariates included. Aslanidou *et al.* (1998) also reportes $\hat{\beta} = -1.0$. Hemming and Shaw (2005) obtain $\hat{\beta} = -1.7$ and Qiou *et al.* (1999) obtain $\hat{\beta} = -1.1$ under the positive stable and $\hat{\beta} = -1.6$ under gamma frailties, respectively. The deviance information criterion (DIC), as presented by Spiegelhalter, Best, Carlin, and Van der Linde (2002), was 398 for either PT or normal model (not shown), so the normal model does about the same from a predictive standpoint based on the DIC.

Bayesian semiparametric generalized linear mixed effect model

Call:

```
PTglmm.default(fixed = y ~ gender + loghazard, random = ~1 |
  id, family = poisson(log), offset = log(off), prior = prior,
  mcmc = mcmc, state = state, status = FALSE)
```

Posterior Predictive Distributions (log):

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
-5.99200	-0.22250	-0.10970	-0.48500	-0.05714	-0.01381

Model's performance:

Dbar	Dhat	pD	DIC	LPML
379.21	360.63	18.58	397.79	-200.29

Regression coefficients:

	Mean	Median	Std.Dev.	Naive.Std.Err.	95%CI-L	95%CI-U
(Intercept)	-0.0004	0.0015	0.0960	0.0013	-0.2066	0.2021
gender	-1.1321	-1.1296	0.3219	0.0045	-1.7762	-0.5117
loghazard1	-4.2608	-4.2375	0.4412	0.0062	-5.1598	-3.4611
loghazard2	-3.7898	-3.7638	0.5018	0.0070	-4.8383	-2.8794
loghazard3	-3.9792	-3.9691	0.4556	0.0064	-4.9028	-3.1213
loghazard4	-3.0627	-3.0526	0.4526	0.0064	-4.0124	-2.2353
loghazard5	-3.2581	-3.2477	0.4219	0.0059	-4.1039	-2.4603
loghazard6	-3.9951	-3.9805	0.4544	0.0064	-4.9103	-3.1403
loghazard7	-4.9343	-4.9183	0.5365	0.0075	-6.0496	-3.9150
loghazard8	-3.6883	-3.6845	0.4479	0.0063	-4.5692	-2.8232
loghazard9	-3.6723	-3.6673	0.4810	0.0068	-4.6112	-2.7315
loghazard10	-4.1246	-4.1272	0.4966	0.0070	-5.0749	-3.1886

Baseline distribution:

	Mean	Median	Std.Dev.	Naive.Std.Err	95%CI-L	95%CI-U
mu-(Intercept)	0.000	0.000	0.000	0.000	0.000	0.000
sigma-(Intercept)	0.430	0.354	0.294	0.004	0.119	1.212

Precision parameter:

	Mean	Median	Std.Dev.	Naive.Std.Err	95%CI-L	95%CI-U
alpha	1.058	0.751	1.022	0.014	0.044	3.769

Random effects variance:

	Mean	Median	Std.Dev.	Naive.Std.Err	95%CI-L	95%CI-U
R.E.Cov-(Intercept)	0.378	0.331	0.222	0.003	0.096	0.948

Acceptance Rate for Metropolis Steps = 0.3570 0.6072 0 0.428 0.463 0

Number of Observations: 413

Number of Groups: 38

Figure 3 shows the estimated frailty distribution from these data along with the posterior mean of the frailty term for each patient. The distribution is remarkably Gaussian-shaped,

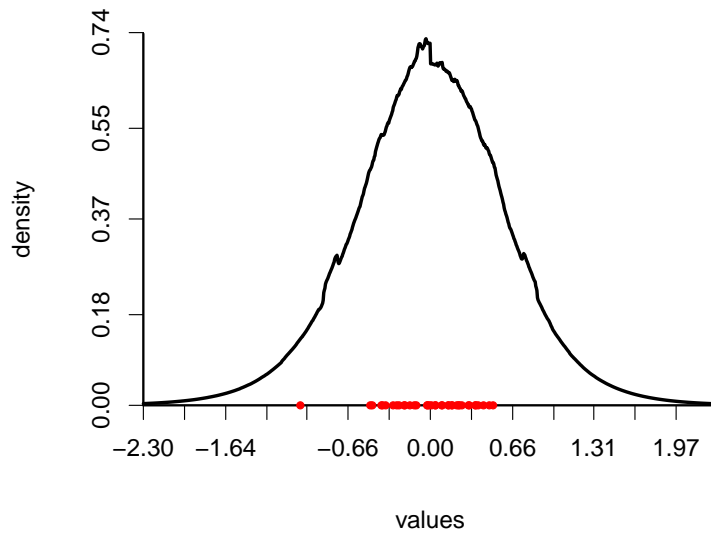


Figure 3: Kidney data: Posterior mean of the frailty distribution. The posterior mean of the individual frailty terms are shown in red dots.

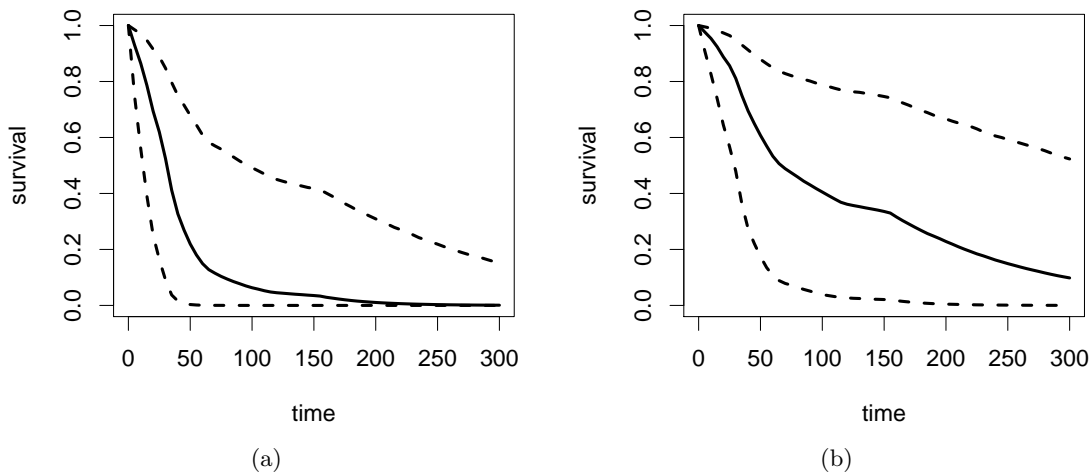


Figure 4: Kidney data: Posterior estimates (median and point-wise 95% credible intervals) for the survival function for time to infection. The results for males and females are shown in panels (a) and (b), respectively.

in contrast to the analysis presented in [Walker and Mallick \(1997\)](#), which showed two well defined density modes corresponding to men and women. We were unable to duplicate this result across several sets of hyper-prior values, including the consideration of $PT^8(\Pi^{100}, \mathcal{A}^{0.1})$. In retrospect, this is not surprising. Two well separated modes would typically indicate an omitted covariate, although gender was included as a risk factor in the model.

Finally, [Figure 4](#) shows the posterior median and 95% credible interval for survival curves for males and females, taking the individual-level heterogeneity modeled through the frailty distribution into account.

5. Dependent random effects distributions

Although different BNP strategies have been proposed to relax the parametric assumption of the random effects distribution in mixed models, the assumption of exchangeable random effects has rarely been tested or relaxed, possibly obscuring important features of the among-subjects variation. We show how to fit dependent GLMM in the context of educational data using the `LDDPrasch` function in this section.

5.1. The dependent semiparametric Rasch model

The statistical analyses of educational measurement data are usually based in Rasch-type models (see, e.g., [De Boeck and Wilson 2004](#)). Rasch-type models ([Rasch 1960](#)) can be viewed as a particular case of GLMM (see, e.g., [Doran, Bates, Bliese, and Dowling 2007](#); [De Boeck, Bakker, Zwitser, Nivard, Hofman, Tuerlinckx, and Partchev 2011](#)), where the linear predictor, η_{ij} , is modeled as a linear function of two parameters, $\eta_{ij} = \theta_i - \beta_j$, where $\theta_i \in \mathbb{R}$ corresponds to the ability of subject i , $i = 1, \dots, n$, and $\beta_j \in \mathbb{R}$ corresponds to the difficulty of probe/item j , $j = 1, \dots, m$. The difficulty and ability parameters are interpreted as fixed and random effects, respectively. The structural assumptions underlying the model specification implies an important property of the models, namely, that the vectors of individual response patterns, $\mathbf{Y}_i = (Y_{i1}, \dots, Y_{im})$, and then total number of correct responses (total score), form an *iid* process conditionally on the difficulty parameters, $\boldsymbol{\beta} = (\beta_1, \dots, \beta_m)$, and the abilities distribution G (see, e.g., [San Martín et al. 2011](#)). The common distribution of \mathbf{Y}_i , for all $i \in \mathbb{N}$, is given by

$$\begin{aligned} P[\mathbf{Y}_i = \mathbf{y} \mid \boldsymbol{\beta}, G] &= E \{ P[\mathbf{Y}_i = \mathbf{y} \mid \boldsymbol{\theta}, \boldsymbol{\beta}] \mid \boldsymbol{\beta}, G \}, \\ &= \int \left\{ \prod_{1 \leq j \leq m} P[Y_{ij} = y_{ij} \mid \theta, \beta_j] \right\} G(d\boldsymbol{\theta}), \end{aligned}$$

for $\mathbf{y} \in \{0, 1\}^m$.

In many practical applications the exchangeability assumption is not appropriate and a partially exchangeable model that emphasizes the explanatory use of the model may be more appropriated. [De Boeck and Wilson \(2004\)](#) discussed a wide variety of explanatory Rasch-type models that incorporate predictor information in the model. When the predictors represent characteristics of the individuals, the abilities parameters have been modeled using a location model given by

$$\theta_i = \mathbf{x}_i^\top \boldsymbol{\gamma} + u_i, \quad (4)$$

and

$$u_1, \dots, u_n \mid G \sim G, \quad (5)$$

where $\boldsymbol{\gamma}$ are “fixed” regression coefficients and the u_i represent “residual” random effects. Therefore, the model becomes a more general GLMM. Although a semiparametric specification of the model given by expressions (4) and (5), obtained by considering a BNP model for G , provides flexibility in capturing different distributional shapes and, at the same time, allows for non-exchangeable abilities, it implies that the vector of covariates \mathbf{x}_i acts modifying the location of the distribution of the subject specific behaviors only. However, a more

thorough evaluation of the effect of the predictors should account for potential changes in other characteristics of the abilities distribution than just the location. For instance, it is useful to examine for potential changes in the symmetry and multimodality of the abilities distribution.

The `LDDPrasch` functions considers the LDDP mixture of normals prior for the abilities in a Rasch model, as described in [Fariña, Quintana, San Martín, and Jara \(2011\)](#). The model is given by

$$Y_{ij} | \pi_{ij} \stackrel{ind}{\sim} \text{Bernoulli}(\pi_{ij}),$$

$$\text{logit}(\pi_{ij}) = \theta_i - \beta_j,$$

$$\boldsymbol{\beta}_{2:k} | \boldsymbol{\beta}_0, \mathbf{S}_{\beta_0} \sim N_{k-1}(\boldsymbol{\beta}_0, \mathbf{S}_{\beta_0}),$$

$$\theta_i | G \stackrel{ind}{\sim} \int N(\theta_i | \mathbf{x}_i^\top \boldsymbol{\gamma}, \sigma^2) dG(\boldsymbol{\gamma}, \sigma^2).$$

and

$$G | \alpha, G_0 \sim DP(\alpha G_0),$$

where $G_0 \equiv N_p(\boldsymbol{\gamma} | \boldsymbol{\mu}_b, \mathbf{S}_b) \Gamma(\sigma^{-2} | \tau_1/2, \tau_2/2)$. Notice that the first difficulty parameter is fixed at 0 in order to avoid identification problems (see, [San Martín et al. 2011](#)). The LDDP model specification is completed with the following hyper-priors:

$$\alpha | a_0, b_0 \sim \Gamma(a_0, b_0),$$

$$\tau_2 | \tau_{s_1}, \tau_{s_2} \sim \Gamma(\tau_{s_1}/2, \tau_{s_2}/2),$$

$$\boldsymbol{\mu}_b | \mathbf{m}_0, \mathbf{S}_0 \sim N_p(\mathbf{m}_0, \mathbf{S}_0),$$

and

$$\mathbf{S}_b | \nu, \boldsymbol{\Psi} \sim IW_p(\nu, \boldsymbol{\Psi}).$$

5.2. Chilean educational data

We consider data from the Chilean system for educational quality measurement (sistema de medición de la calidad de la educación, SIMCE). The Chilean education system is subject to several performance evaluations regularly at the school, teacher and student level. In the latter case, SIMCE has developed mandatory census-type tests to regularly assess the educational progress at three stages: 4th and 8th grades in primary school (9 and 13 years old children, respectively), and 2nd grade in secondary school (16 years old children). The SIMCE instruments are designed to assess the achievement of fundamental goals and minimal contents of the curricular frame in different areas of knowledge, currently Spanish, mathematics and

science. Here we focus on data from the math test applied in 2004 to 8 grader examinees in primary school. The test consists of 45 multiple choice items questions with 4 alternatives. The response $Y_{ij} \in \{0, 1\}$ is a binary variable indicating whether the individual i answers item j correctly.

The main purpose of collecting these data is to monitor standards and progress of educational systems, focusing on characterizing the population (and its evolution) rather than individual examinees. It is of particular interest to understand the way in which some factors at individual and/or school level could explain systematic differences in the performance of students in order to establish policies to improve the education system. For instance, a significant characteristic of the Chilean elementary and secondary education system is a variety of different school types. These are grouped as Public I, financed by the state and administered by county governments; Public II, financed by the state and administered by county corporations; Private I, financed by the state and administered by the private sector; Private II, fee-paying schools that operate solely on payments from parents and administered by the private sector.

In order to evaluate the effect of the type of school and gender on the student performance we consider the `LDDPrasch` function. For illustration purposes, we consider a subset of 500 children. We refer to [Fariña et al. \(2011\)](#) for a full analysis of the complete data. In this analysis, \mathbf{x}_i includes an intercept term, three dummy variables for the type of school and the gender indicator. The LDDP Rasch model was fitted assuming $\beta_{2:45} \sim N_{44}(\mathbf{0}, 10^3 \mathbf{I}_{44})$, $\alpha = 1$, $\mu_0 = \mathbf{0}_5$, $\mathbf{S}_0 = 100 \mathbf{I}_5$, $\tau_1 = 6.01$, $\tau_{s1} = 6.01$, $\tau_{s2} = 2.01$, $\nu = 8$, $\Psi = \mathbf{I}_5$. The following code illustrates the prior specification in the `LDDPrasch` function.

```
R> prior <- list(alpha = 1, beta0 = rep(0, 44), Sbeta0 = diag(1000, 44),
+   mu0 = rep(0, 5), S0 = diag(100, 5), tau1 = 6.01, taus1 = 6.01,
+   taus2 = 2.01, nu = 8, psiinv = diag(1, 5))
```

A single Markov chain cycle of length 25,000 was completed. The full chain was sub-sampled every 4 steps after a burn in period of 5,000 samples, to give a reduced chain of length 5,000.

```
R> mcmc <- list(nburn = 5000, nskip = 3, ndisplay = 1000, nsave = 5000)
```

For each gender and type of school the density of the abilities distribution was evaluated on a grid of 100 equally spaced points in the range $(-3, 8)$. The following code was used to create the design matrix for the prediction, where the first column corresponds to the intercept, the next three columns correspond to the dummy variable for the type of school (types), and the last column corresponds to the gender indicator (1= girl).

```
R> zpred <- matrix(c(1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 1, 0, 0,
+   1, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 1, 0, 0, 1, 1, 0, 1, 0, 1,
+   1, 0, 0, 1, 1), nrow = 8, ncol = 5, byrow = TRUE)
```

Based on the previous specification, the `LDDPrasch` function was used to fit the model, using the following code:

```
R> fitLDDP <- LDDPrasch(formula = y ~ types + gender, prior = prior,
+   mcmc = mcmc, state = NULL, status = TRUE, zpred = zpred,
+   grid = seq(-3, 8, len = 100), compute.band = TRUE)
```

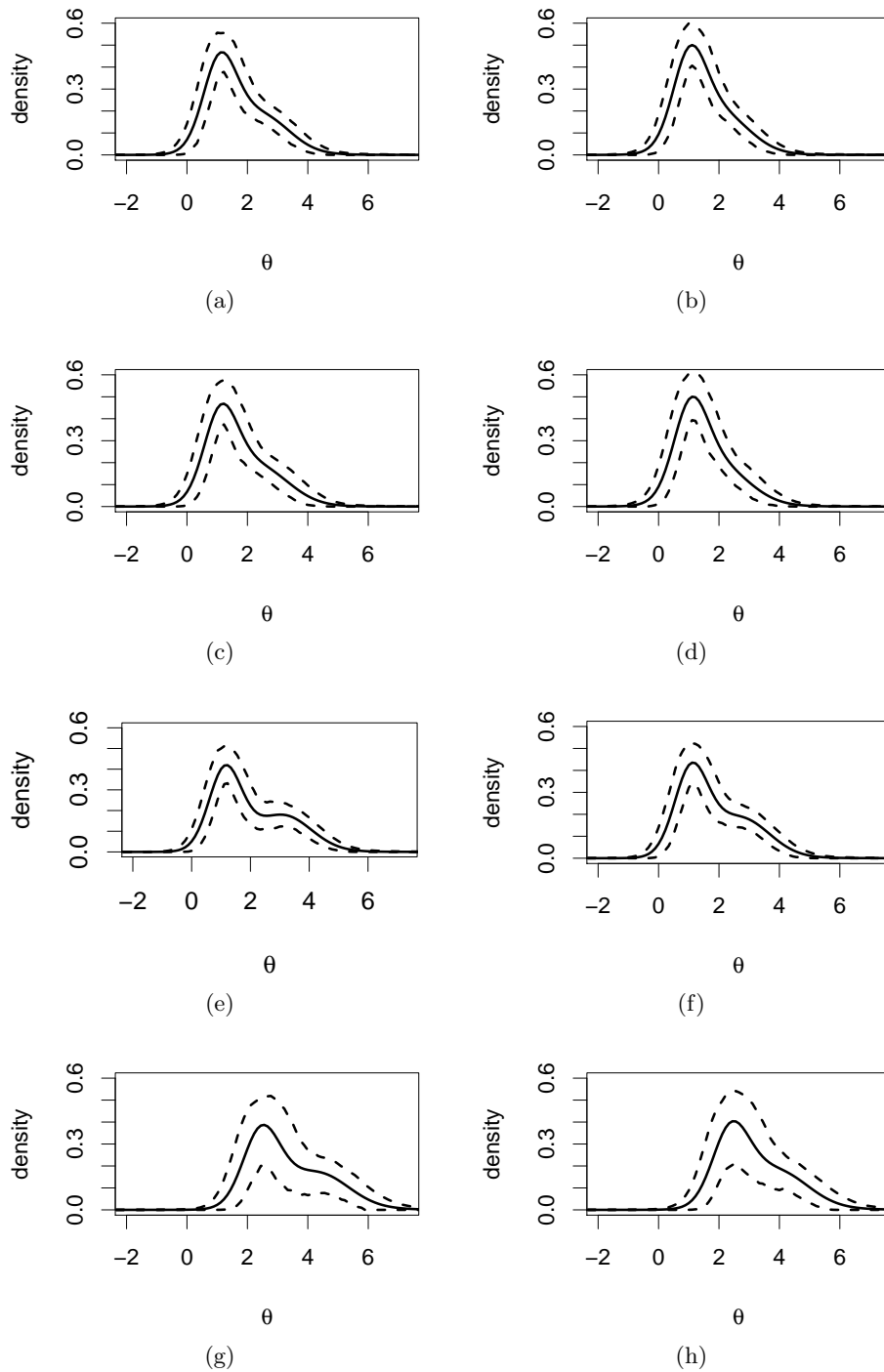


Figure 5: SIMCE data: Posterior estimates (mean and point-wise 95% HPD intervals) for the ability distribution for type of school and gender. The results for boys are shown in panels (a), (c), (e) and (g) for type of school Public I, Public II, Private I, and Private II, respectively. The results for girls are shown in panels (b), (d), (f) and (h) for type of school Public I, Public II, Private I, and Private II, respectively.

Different shapes in the resulting posterior densities were observed. Figure 5 displays the posterior mean and point wise 95%HPD interval for the random effects distribution for different combinations of the predictors. The density estimates show a clear departure from the commonly assumed normality of the random effects distributions. We found no important differences in the behavior of boys and girls. Children in Public I and II schools showed a similar skewed to the right random effects distribution. The estimated abilities distributions for children in private schools were shifted to the right in comparison with the distribution observed for children from public schools. This shift was more pronounced for children in fee-paying schools that operate solely on payments from parents and administered by the private sector (Private II) than those from schools financed by the state and administered by the private sector (Private I). A bimodal random effects distribution was observed in the abilities distributions from private schools.

6. Concluding remarks

Because the main obstacle for the practical use of BSP and BNP methods has been the lack of estimation tools, we presented an R package for fitting some frequently used models. Until the release of **DPpackage**, the two options for researchers who wished to fit a BSP or BNP model were to write their own code or to rely heavily on particular parametric approximations to some specific processes using the **BUGS** code given in Peter Congdon's books (see, e.g., Congdon 2001). **DPpackage** is geared primarily towards users who are not willing to bear the costs associated with either of these options.

Chambers (2000) conceptualized statistical software as a set of tools to organize, analyze and visualize data. Data organization and visualization of results is based on R capabilities. Chambers (2000) also proposed requirements and guidelines for developing and assessing statistical softwares. These requirements may be discussed with respect to **DPpackage**.

1. *Easy specification of simple tasks:* The documentation contains examples, and similar problems can be analyzed by moderate modifications of the model description files. The examples have been chosen so that they demonstrate the functionality of **DPpackage** with well-known datasets.
2. *Gradual refinement of the tasks:* The user can enhance a nonparametric model by adding covariates, and by fixing part of the baseline distributions and the precision parameters.
3. *Arbitrarily extensive programming:* **DPpackage** has a programming environment for implementing sophisticated proposal distributions, if the default proposals are not sufficient. Each source file contains a detailed specification of the corresponding function which is useful for potential user/developers who wish to modify the source files. In addition, we are currently working on a **DPpackage** developer's manual that will help users who would like to add new functionality to the package.
4. *Implementing high-quality computations:* Also, because the source code in a compiled language is available, new procedures can be added and the old ones modified to improve performance and flexibility.
5. *Embedding the results of items 2–4 as new simple tools:* **DPpackage** has the capability of continuing a Markov chain from the last value of the parameters of a previous analysis.

As the MCMC samples are saved in matrix objects, both parts of the Markov chain can be easily merged.

Many improvements to the current status of the package can be made. For example, all **DPpackage** modeling functions compute conditional predictive ordinates for model comparison. However, only some of them compute the effective number of parameters pD and the DIC. These and other model comparison criterion will be included for all functions in future versions of **DPpackage**.

The implementation of more models, the development of general-purpose sampling functions, real-time visualization of simulation progress, and the ability to handle large data set problems, through the use of sparse matrix techniques (George and Liu 1981), are the topic of further improvements.

Acknowledgments

The first author is supported by Fondecyt 3095003 and 11100144 grants. Partial support from the KUL-PUC bilateral (Belgium-Chile) grant BIL05/03 and of the IAP research network grant Nr P6/03 of the Belgian government (Belgian Science Policy) for previous versions of **DPpackage** is also acknowledged. The work of the second author was supported in part by NIH grant 2-R01-CA95955-05. The third author was partially supported by grant Fondecyt 1060729. The last two authors were partially supported by grant NIH/NCI R01CA75981. The SIMCE Office from the Chilean Government kindly allowed us access to the databases used in this paper.

References

- Antoniak CE (1974). “Mixtures of Dirichlet Processes with Applications to Bayesian Non-parametric Problems.” *The Annals of Statistics*, **2**(6), 1152–1174.
- Aslanidou H, Dey DK, Sinha D (1998). “Bayesian Analysis of Multivariate Survival Data Using Monte Carlo Methods.” *The Canadian Journal of Statistics*, **26**(1), 33–48.
- Barrientos F, Jara A, Quintana FA (2011). “Dependent Bernstein Polynomials for Bounded Density Regression.” *Technical report*, Department of Statistics, Pontificia Universidad Católica de Chile. URL <http://www.mat.puc.cl/~ajara/Publications.html>.
- Besag J, Green P, Higdon D, Mengersen K (1995). “Bayesian Computation and Stochastic Systems.” *Statistical Science*, **10**(1), 3–66.
- Carlin BP, Louis TA (2008). *Bayesian Methods for Data Analysis*. 3rd edition. Chapman and Hall/CRC, New York.
- Chambers JM (2000). “Users, Programmers, and Statistical Software.” *Journal of Computational and Graphical Statistics*, **9**(3), 404–422.
- Chen MH, Shao QM (1999). “Monte Carlo Estimation of Bayesian Credible and HPD Intervals.” *Journal of Computational and Graphical Statistics*, **8**(1), 69–92.

- Christensen R, Hanson T, Jara A (2008). “Parametric Nonparametric Statistics: An Introduction to Mixtures of Finite Polya Trees.” *The American Statistician*, **62**(4), 296–306.
- Congdon P (2001). *Bayesian Statistical Modelling*. John Wiley & Sons, New York.
- Cox DR (1972). “Regression Models and Life-Tables.” *Journal of the Royal Statistical Society B*, **34**(2), 187–220.
- De Boeck P, Bakker M, Zwitser R, Nivard M, Hofman A, Tuerlinckx F, Partchev I (2011). “The Estimation of Item Response Models with the `lmer` Function from the `lme4` Package in R.” *Journal of Statistical Software*, **39**(12), 1–28. URL <http://www.jstatsoft.org/v39/i12/>.
- De Boeck P, Wilson M (2004). *Explanatory Item Response Models: A Generalized Linear and Nonlinear Approach*. Springer-Verlag, New York.
- De Iorio M, Johnson WO, Müller P, Rosner GL (2009). “Bayesian Nonparametric Nonproportional Hazards Survival Modelling.” *Biometrics*, **65**(3), 762–771.
- De Iorio M, Müller P, Rosner GL, MacEachern SN (2004). “An ANOVA Model for Dependent Random Measures.” *Journal of the American Statistical Association*, **99**(465), 205–215.
- Dey D, Müller P, Sinha D (1998). *Practical Nonparametric and Semiparametric Bayesian Statistics*. Springer-Verlag, New York.
- Doran H, Bates D, Bliese P, Dowling M (2007). “Estimating the Multilevel Rasch Model: With the `lme4` Package.” *Journal of Statistical Software*, **20**(2), 1–18. URL <http://www.jstatsoft.org/v20/i02/>.
- Duan JA, Guindani M, Gelfand AE (2007). “Generalized Spatial Dirichlet Process Models.” *Biometrika*, **94**(4), 809–825.
- Dunson DB, Park JH (2008). “Kernel Stick-Breaking Processes.” *Biometrika*, **95**(2), 307–323.
- Dunson DB, Pillai N, Park JH (2007a). “Bayesian Density Regression.” *Journal of the Royal Statistical Society B*, **69**(2), 163–183.
- Dunson DB, Yang M, Baird D (2007b). “Semiparametric Bayes Hierarchical Models with Mean and Variance Constraints.” *Technical report*, Department of Statistical Science, Duke University. URL <http://ftp.stat.duke.edu/WorkingPapers/07-08.html>.
- Escobar MD (1994). “Estimating Normal Means with a Dirichlet Process Prior.” *Journal of the American Statistical Association*, **89**(425), 268–277.
- Escobar MD, West M (1995). “Bayesian Density Estimation and Inference Using Mixtures.” *Journal of the American Statistical Association*, **90**(430), 577–588.
- Fariña P, Quintana FA, San Martín E, Jara A (2011). “A Dependent Semiparametric Rasch Model for the Analysis of Chilean Educational Data.” *Technical report*, Department of Statistics, Pontificia Universidad Católica de Chile. URL <http://www.mat.puc.cl/~ajara/Publications.html>.
- Ferguson TS (1973). “A Bayesian Analysis of Some Nonparametric Problems.” *The Annals of Statistics*, **1**(2), 209–230.

- Ferguson TS (1974). “Prior Distributions on Spaces of Probability Measures.” *The Annals of Statistics*, **2**(4), 615–629.
- George A, Liu JW (1981). *Computer Solution of Large Sparse Positive Definite Systems*. Prentice-Hall, New York.
- Ghosh JK, Ramamoorthi RV (2003). *Bayesian Nonparametrics*. Springer-Verlag, New York.
- Gilks WR, Thomas A, Spiegelhalter DJ (1994). “A Language and Program for Complex Bayesian Modelling.” *Journal of the Royal Statistical Society Series D*, **43**(1), 169–177.
- Griffin JE, Steel MFJ (2006). “Order-Based Dependent Dirichlet Processes.” *Journal of the American Statistical Association*, **101**(473), 179–194.
- Hanson T (2006). “Inference for Mixtures of Finite Polya Tree Models.” *Journal of the American Statistical Association*, **101**(476), 1548–1565.
- Hanson T, Branscum A, Johnson W (2005). “Bayesian Nonparametric Modeling and Data Analysis: An Introduction.” In DK Dey, CR Rao (eds.), *Handbook of Statistics, Volume 25: Bayesian Thinking, Modeling and Computation*, pp. 245–278. Elsevier, Amsterdam, The Netherlands.
- Hanson T, Johnson WO (2002). “Modeling Regression Error with a Mixture of Polya Trees.” *Journal of the American Statistical Association*, **97**(460), 1020–1033.
- Hanson T, Monteiro JVD, Jara A (2011). “The Polya Tree Sampler: Toward Efficient and Automatic Independent Metropolis-Hastings Proposals.” *Journal of Computational and Graphical Statistics*, **20**(1), 41–62.
- Hemming K, Shaw JEH (2005). “A Class of Parametric Dynamic Survival Models.” *Lifetime Data Analysis*, **11**(1), 81–98.
- Hjort NL, Holmes C, Müller P, Walker S (2010). *Bayesian Nonparametrics*. Cambridge University Press, Cambridge, UK.
- Jara A (2007). “Applied Bayesian Non- and Semi-parametric Inference Using **DPpackage**.” *R News*, **7**(3), 17–26.
- Jara A, Hanson T, Lesaffre E (2009). “Robustifying Generalized Linear Mixed Models Using a New Class of Mixture of Multivariate Polya Trees.” *Journal of Computational and Graphical Statistics*, **18**(4), 838–860.
- Jara A, Lesaffre E, De Iorio M, Quintana FA (2010). “Bayesian Semiparametric Inference for Multivariate Doubly-Interval-Censored Data.” *The Annals of Applied Statistics*, **4**(4), 2126–2149.
- Kalbfleisch JD (1978). “Non-Parametric Bayesian Analysis of Survival Time Data.” *Journal of the Royal Statistical Society B*, **40**(2), 214–221.
- Lang S, Brezger A (2004). “Bayesian P-Splines.” *Journal of Computational and Graphical Statistics*, **13**(1), 183–212.

- Lavine M (1992). “Some Aspects of Polya Tree Distributions for Statistical Modelling.” *The Annals of Statistics*, **20**(3), 1222–1235.
- Lavine M (1994). “More Aspects of Polya Tree Distributions for Statistical Modelling.” *The Annals of Statistics*, **22**(3), 1161–1176.
- Li Y, Müller P, Lin X (2007). “Center-Adjusted Inference for a Nonparametric Bayesian Random Effect Distribution.” *Technical report*, Department of Biostatistics, The MD Anderson Cancer Center. URL http://www-stat.wharton.upenn.edu/statweb/Conference/0Bayes09/AbstractPapers/muller_paper.pdf.
- Lo AY (1984). “On a Class of Bayesian Nonparametric Estimates: I. Density Estimates.” *The Annals of Statistics*, **12**(1), 351–357.
- MacEachern SN (1999). “Dependent Nonparametric Processes.” In *ASA Proceedings of the Section on Bayesian Statistical Science, Alexandria, VA*. American Statistical Association.
- MacEachern SN (2000). “Dependent Dirichlet Processes.” *Technical report*, Department of Statistics, The Ohio State University.
- Mauldin RD, Sudderth WD, Williams SC (1992). “Polya Trees and Random Distributions.” *The Annals of Statistics*, **20**(3), 1203–1221.
- McGilchrist CA, Aisbett CW (1991). “Regression with Frailty in Survival Analysis.” *Biometrics*, **47**(2), 461–466.
- Muliere P, Tardella L (1998). “Approximating Distributions of Random Functionals of Ferguson-Dirichlet Priors.” *The Canadian Journal of Statistics*, **26**(2), 283–297.
- Müller P, Erkanli A, West M (1996). “Bayesian Curve Fitting Using Multivariate Normal Mixtures.” *Biometrika*, **83**(1), 67–79.
- Müller P, Quintana FA (2004). “Nonparametric Bayesian Data Analysis.” *Statistical Science*, **19**(1), 95–110.
- Müller P, Quintana FA, Rosner G (2004). “A Method for Combining Inference Across Related Nonparametric Bayesian Models.” *Journal of the Royal Statistical Society B*, **66**(3), 735–749.
- Newton MA (1994). “Computing with Priors that Support Identifiable Semiparametric Models.” *Technical Report 905*, Department of Statistics, University of Wisconsin-Madison.
- Newton MA, Czado C, Chapell R (1996). “Bayesian Inference for Semiparametric Binary Regression.” *Journal of the American Statistical Association*, **91**(433), 142–153.
- Perron F, Mengersen K (2001). “Bayesian Nonparametric Modeling Using Mixtures of Triangular Distributions.” *Biometrics*, **57**(2), 518–528.
- Petrone S (1999a). “Bayesian Density Estimation Using Bernstein Polynomials.” *The Canadian Journal of Statistics*, **27**(1), 105–126.
- Petrone S (1999b). “Random Bernstein Polynomials.” *Scandinavian Journal of Statistics*, **26**(3), 373–393.

- Petrone S, Wasserman L (2002). “Consistency of Bernstein Polynomial Posteriors.” *Journal of the Royal Statistical Society B*, **64**(1), 79–100.
- Plummer M, Best N, Cowles K, Vines K (2006). “**coda**: Convergence Diagnosis and Output Analysis for MCMC.” *R News*, **6**(1), 7–11. URL <http://CRAN.R-project.org/doc/Rnews/>.
- Qiou Z, Ravishanker N, Dey DK (1999). “Multivariate Survival Analysis with Positive Stable Frailties.” *Biometrics*, **55**(2), 637–644.
- Rasch G (1960). *Probabilistic Models for Some Intelligence and Attainment Tests*. The Danish Institute for Educational Research, Chicago. Expanded Edition, 1980, The University Chicago Press.
- 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/>.
- Rossi P, Allenby G, McCulloch R (2005). *Bayesian Statistics and Marketing*. John Wiley & Sons, New York.
- Rossi P, McCulloch R (2011). **bayesm**: *Bayesian Inference for Marketing/Micro-Econometrics*. R package version 2.2-4, URL <http://CRAN.R-project.org/package=bayesm>.
- San Martín E, Jara A, Rolin JM, Mouchart M (2011). “On the Analysis of Bayesian Semi-parametric IRT-Type Models.” *Psychometrika*. Forthcoming.
- Sethuraman J (1994). “A Constructive Definition of Dirichlet Priors.” *Statistica Sinica*, **4**, 639–650.
- Smith BJ (2007). “**boa**: An R Package for MCMC Output Convergence Assessment and Posterior Inference.” *Journal of Statistical Software*, **21**(11), 1–37. URL <http://www.jstatsoft.org/v21/i11/>.
- Spiegelhalter SD, Best NG, Carlin BP, Van der Linde A (2002). “Bayesian Measures of Model Complexity and Fit.” *Journal of the Royal Statistical Society B*, **64**(4), 583–639.
- Sturtz S, Ligges U, Gelman A (2005). “**R2WinBUGS**: A Package for Running **WinBUGS** from R.” *Journal of Statistical Software*, **12**(3), 1–16. URL <http://www.jstatsoft.org/v12/i03/>.
- Thomas A, O’Hara B, Ligges U, Sibylle S (2006). “Making **BUGS** Open.” *R News*, **6**(1), 12–17.
- Venables WN, Ripley BD (2002). *Modern Applied Statistics with S*. 4th edition. Springer-Verlag, New York.
- Walker SG, Damien P, Laud PW, Smith AFM (1999). “Bayesian Nonparametric Inference for Random Distributions and Related Functions.” *Journal of the Royal Statistical Society B*, **61**(3), 485–527.

Walker SG, Mallick BK (1997). “Hierarchical Generalized Linear Models and Frailty Models with Bayesian Nonparametric Mixing.” *Journal of the Royal Statistical Society B*, **59**(4), 845–860.

Zellner A (1983). “Applications of Bayesian Analysis in Econometrics.” *Journal of the Royal Statistical Society D*, **32**(1/2), 23–34.

Affiliation:

Alejandro Jara
Department of Statistics
Facultad de Matemáticas
Pontificia Universidad Católica de Chile
Casilla 306, Correo 22
Santiago, Chile
Telephone: +56/2/3544506
Fax: +56/2/3547729
E-mail: atjara@uc.cl
URL: <http://www.mat.puc.cl/~ajara/>