

Johns Hopkins University, Dept. of Biostatistics Working Papers

12-1-2007

Bayesian Analysis for Penalized Spline Regression Using Win BUGS

Ciprian M. Crainiceanu
Johns Hokins Bloomberg School of Public Health, Department of Biostatistics, ccrainic@jhsph.edu

David Ruppert
Cornell University, School of Operational Research & Industrial Engineering, rcarroll@tamu.edu

M.P. Wand
Department of Statistics, School of Mathematics, University of South Wales, wand@maths.unsw.edu.au

Suggested Citation

Crainiceanu, Ciprian M.; Ruppert, David; and Wand, M.P., "Bayesian Analysis for Penalized Spline Regression Using Win BUGS" (December 2007). *Johns Hopkins University, Dept. of Biostatistics Working Papers*. Working Paper 40.
http://biostats.bepress.com/jhubiostat/paper40

This working paper is hosted by The Berkeley Electronic Press (bepress) and may not be commercially reproduced without the permission of the copyright holder.
Copyright © 2011 by the authors

Bayesian Analysis for Penalized Spline Regression Using WinBUGS

Ciprian M. Crainiceanu* David Ruppert† M.P. Wand‡

December 1, 2007

Abstract

Penalized splines can be viewed as BLUPs in a mixed model framework, which allows the use of mixed model software for smoothing. Thus, software originally developed for Bayesian analysis of mixed models can be used for penalized spline regression. Bayesian inference for nonparametric models enjoys the flexibility of nonparametric models and the exact inference provided by the Bayesian inferential machinery. This paper provides a simple, yet comprehensive, set of programs for the implementation of nonparametric Bayesian analysis in WinBUGS. MCMC mixing is substantially improved from the previous versions by using low–rank thin–plate splines instead of truncated polynomial basis. Simulation time per iteration is reduced 5 to 10 times using a computational trick.

Keywords: MCMC, Semiparametric regression, Software

1 Introduction

The virtues of nonparametric regression models have been discussed extensively in the statistics literature. Competing approaches to nonparametric modeling include, but are not limited to, smoothing splines (Eubank [9]; Wahba [28]; Green and Silverman [15]), series-based smoothers (Tarter and Lock [26]; Ogden [21]), kernel methods (Wand and Jones [30]; Fan and Gijbels [10]); regression splines (Hastie and Tibshirani [18]; Friedman [11]; Hansen and Kooperberg [17]; penalized splines (Eilers and Marx, [8],

*Department of Biostatistics, Johns Hopkins University, 615 N. Wolfe St. E3037 Baltimore, MD 21205 USA. E-mail: ccrainic@jhsph.edu

†School of Operational Research and Industrial Engineering, Cornell University, Rhodes Hall, NY 14853, USA. E-mail: ruppert@orie.cornell.edu

‡Department of Statistics, School of Mathematics, University of New South Wales, Sydney 2052, Australia E-mail: wand@maths.unsw.edu.au

Collection of Biostatistics Research Archive

Hosted by The Berkeley Electronic Press

Ruppert, Wand and Carroll [23]). The main advantage of nonparametric over parametric models is their flexibility. In the nonparametric framework the shape of the functional relationship between covariates and the dependent variables is determined by the data, whereas in the parametric framework the shape is determined by the model.

In this paper we focus on semiparametric regression models using penalized splines (Ruppert, Wand and Carroll [23]), but the methodology can be extended to other penalized likelihood models. It is becoming more widely appreciated that penalized likelihood models can be viewed as particular cases of Generalized Linear Mixed Models (GLMMs): Eilers and Marx [8]; Brumback, Ruppert and Wand 1999, [3]; Ruppert, Wand and Carroll [23]. We discuss this in more details in section 2. Given this equivalence, statistical software developed for mixed models, such as S-plus (function `lme`) or SAS (`PROC MIXED` and the `GLIMMIX` macro) can be used for smoothing (Wand [29], Ngo and Wand [20]). There are at least two potential problems when using such software for inference in mixed models. Firstly, in the case of GLMMs the likelihood of the model is a high dimensional integral over the unobserved random effects and, in general, cannot be computed exactly and has to be approximated. This can have a sizeable effect on parameter estimation, especially on the variance components. The second problem is that confidence intervals are obtained by replacing the estimated parameters instead of the true parameters and ignoring the additional variability. This results in tighter than normal confidence intervals and could be avoided by using bootstrap. However, standard software does not have bootstrap capabilities and favors the "plug-in" method.

Bayesian analysis treats all parameters as random, assigns prior distributions to characterize knowledge about parameter values prior to data collection, and uses the joint posterior distribution of parameters given the data as the basis of inference. Often the posterior density is analytically unavailable but can be simulated using Markov Chain Monte Carlo (MCMC). Moreover, the posterior distribution of any explicit function of the model parameters can be obtained as a by-product of the simulation algorithm.

The Bayesian inference for nonparametric models enjoys the flexibility of nonpara-

COBRA BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

http://biostats.bepress.com/jhubiostat/paper40

metric models and the exact inference provided by the Bayesian inferential machinery. It is this combination that makes Bayesian nonparametric modeling so attractive (e.g. Berry, Carroll, and Ruppert [2]; Ruppert, Wand and Carroll [23]).

The goal of this paper is not to discuss Bayesian methodology, nonparametric regression or provide novel modeling techniques. Instead, we provide a simple, yet comprehensive, set of programs for the implementation of nonparametric Bayesian analysis in WinBUGS (Spiegelhalter, Thomas and Best [24]), which has become the standard software for Bayesian analysis. Special attention is given to the choice of spline basis and MCMC mixing properties. The R package R2WinBUGS developed by Sturtz, Ligges and Gelman [25] is used to call WinBUGS 1.4 and export results in R. This is especially helpful when studying the frequentist properties of Bayesian inference using simulations.

2 Low–rank thin–plate splines

Ruppert, Wand and Carroll [23] present the general methodology of semiparametric modeling using the equivalence between penalized splines mixed models. This paper shows how to do the Bayesian analysis of semiparametric models using WinBUGS.

We now introduce low–rank thin–plate splines. Consider the regression model

$$y_i = m(x_i) + \epsilon_i \ ,$$

where ϵ_i are i.i.d. $N\left(0, \sigma_\epsilon^2\right)$, ϵ_i is independent x_i, and $m(\cdot)$ is a smooth function. The smooth function could be modeled using natural cubic splines, B-splines, truncated polynomials, radial splines etc. In Bayesian analysis, the particular choice of basis has important consequences for the mixing properties of the MCMC chains. We will focus on low–rank thin–plate splines which tend to have very good numerical properties. In particular, the posterior correlation of parameters of the thin–plate splines is much smaller than for other basis (e.g. truncated polynomials) which greatly improves mixing.

The low–rank thin–plate spline representation of $m(\cdot)$ is

$$m(x, \boldsymbol{\theta}) = \beta_0 + \beta_1 x + \sum_{k=1}^{K} u_k \left|x - \kappa_k\right|^3 \ ,$$

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

Hosted by The Berkeley Electronic Press

where $\boldsymbol{\theta} = (\beta_0, \beta_1, u_1, \ldots, u_K)^T$ is the vector of regression coefficients, and $\kappa_1 < \kappa_2 < \ldots < \kappa_K$ are fixed knots. Following Ruppert (2002), [22] we consider a number of knots that is large enough (typically 5 to 20) to ensure the desired flexibility, and κ_k is the sample quantile of x's corresponding to probability $k/(K+1)$, but results hold for any other choice of knots. To avoid overfitting, we minimize

$$\sum_{i=1}^{n} \{y_i - m(x_i, \boldsymbol{\theta})\}^2 + \frac{1}{\lambda}\boldsymbol{\theta}^T \boldsymbol{D}\boldsymbol{\theta}\,, \tag{1}$$

where λ is the smoothing parameter and $\boldsymbol{D}$ is a known positive semi-definite penalty matrix. The thin–plate spline penalty matrix is

$$\boldsymbol{D} = \begin{bmatrix} \mathbf{0}_{2\times 2} & \mathbf{0}_{2\times K} \\ \mathbf{0}_{K\times 2} & (\boldsymbol{\Omega}_K^{1/2})^T \boldsymbol{\Omega}_K^{1/2} \end{bmatrix}$$

where the (l,k)th entry of $\boldsymbol{\Omega}_K$ is $|\kappa_l - \kappa_k|^3$ and penalizes only coefficients of $|x - \kappa_k|^3$.

Let $\boldsymbol{Y} = (y_1, y_2, \ldots, y_n)^T$, $\boldsymbol{X}$ be the matrix with the ith row $\boldsymbol{X}_i = (1, x_i)$, and $\boldsymbol{Z}_K$ be the matrix with ith row $\boldsymbol{Z}_{Ki} = \left\{|x_i - \kappa_1|^3, \ldots, |x_i - \kappa_K|^3\right\}$. If we divide (1) by the error variance one obtains

$$\frac{1}{\sigma_\epsilon^2}\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{Z}_K\|^2 + \frac{1}{\lambda\sigma_\epsilon^2}\boldsymbol{u}^T(\boldsymbol{\Omega}_K^{1/2})^T\boldsymbol{\Omega}_K^{1/2}\boldsymbol{u},$$

where $\boldsymbol{\beta} = (\beta_0, \beta_1)^T$ and $\boldsymbol{u} = (u_1, \ldots, u_K)^T$. Define $\sigma_u^2 = \lambda\sigma_\epsilon^2$, consider the vector $\boldsymbol{\beta}$ as fixed parameters and the vector $\boldsymbol{u}$ as a set of random parameters with $E(\boldsymbol{u}) = 0$ and $\text{cov}(\boldsymbol{u}) = \sigma_u^2\boldsymbol{\Omega}_K^{-1/2}(\boldsymbol{\Omega}_K^{-1/2})^T$. If $(\boldsymbol{u}^T, \boldsymbol{\epsilon}^T)^T$ is a normal random vector and $\boldsymbol{u}$ and $\boldsymbol{\epsilon}$ are independent then one obtains an equivalent model representation of the penalized spline in the form of a LMM (Brumback et al., 1999). Specifically, the P-spline is equal to the best linear predictor (BLUP) in the LMM

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}_K\boldsymbol{u} + \boldsymbol{\epsilon}, \;\; \text{cov}\begin{pmatrix} \boldsymbol{u} \\ \boldsymbol{\epsilon} \end{pmatrix} = \begin{bmatrix} \sigma_u^2\boldsymbol{\Omega}_K^{-1/2}(\boldsymbol{\Omega}_K^{-1/2})^T & 0 \\ 0 & \sigma_\epsilon^2\boldsymbol{I}_n \end{bmatrix}. \tag{2}$$

Using the reparemeterization $\boldsymbol{b} = \boldsymbol{\Omega}_K^{1/2}u$ and defining $\boldsymbol{Z} = \boldsymbol{Z}_K\boldsymbol{\Omega}_K^{-1/2}$ the mixed model (2) is equivalent to

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{b} + \boldsymbol{\epsilon}, \;\; \text{cov}\begin{pmatrix} \boldsymbol{b} \\ \boldsymbol{\epsilon} \end{pmatrix} = \begin{bmatrix} \sigma_b^2\boldsymbol{I}_K & 0 \\ 0 & \sigma_\epsilon^2\boldsymbol{I}_n \end{bmatrix}. \tag{3}$$

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

http://biostats.bepress.com/jhubiostat/paper40

3 The Canadian Age–Income Data

Figure 1 is a scatterplot of age versus log(income) for a sample of $n = 205$ Canadian workers, all of whom were educated to grade 13. These data were used by Ullah [27], who identifies their source as a 1971 Canadian Census Public Use Tape.

3.1 Model and priors

The mean of log(income) as a function of age was modeled using thin–plate splines with $K = 20$ knots chosen so that the k-th knot is the sample quantile of age corresponding to probability $k/(K+1)$. We used model (3) where y_i, x_i denote the log income and age of the i-th worker. The following priors were used

$$\begin{cases} \beta_0,\ \beta_1 & \sim & N(0, 10^6) \\ \sigma_b^{-2},\ \sigma_\epsilon^{-2} & \sim & \text{Gamma}\left(10^{-6}, 10^{-6}\right) \end{cases}, \tag{4}$$

where the second parameter of the normal distribution is the variance. In many applications a normal prior distribution centered at zero with a standard error equal to 1000 is sufficiently noninformative. If there are reasons to suspect, either using alternative estimation methods or prior knowledge, that the true parameter is in another region of the space, then the prior should be adjusted accordingly. The parameterization of the Gamma(a, b) distribution is chosen so that its mean is $a/b = 1$ and its variance is $a/b^2 = 10^6$. In Section 8 we discuss several issues related to prior choice for nonparametric smoothing.

3.2 WinBUGS program for age–income data

We now describe the WinBUGS program that follows closely the description of the Bayesian nonparametric model in equation (3) with the priors defined in (4). We provide the entire program in Appendix A1. While this program was designed for the age–income data, it can be used for other penalized spline regression models with minor adjustments. Many features of the program will be repeated in other examples and changes will be described, as needed.

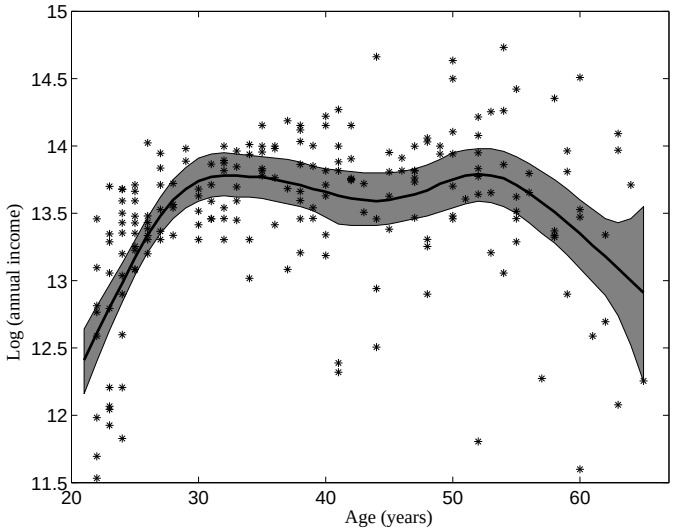
The likelihood part of the model (3) is specified in WinBUGS as follows

```
for (i in 1:n)
```

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

Hosted by The Berkeley Electronic Press

Figure 1: Scatterplot of log(income) versus age for a sample of $n = 205$ Canadian workers with posterior median (solid) and 95% credible intervals for the mean regression function




```
{response[i]~dnorm(m[i],taueps)
 m[i]<-mfe[i]+mre110[i]+mre1120[i]
 mfe[i]<-beta[1]*X[i,1]+beta[2]*X[i,2]
 mre110[i] <-b[1]*Z[i,1]+b[2]*Z[i,2]+b[3]*Z[i,3]+b[4]*Z[i,4]+
            b[5]*Z[i,5]+b[6]*Z[i,6]+b[7]*Z[i,7]+b[8]*Z[i,8]+
            b[9]*Z[i,9]+b[10]*Z[i,10]
 mre1120[i]<-b[11]*Z[i,11]+b[12]*Z[i,12]+b[13]*Z[i,13]+b[14]*Z[i,14]+
            b[15]*Z[i,15]+b[16]*Z[i,16]+b[17]*Z[i,17]+b[18]*Z[i,18]+
            b[19]*Z[i,19]+b[20]*Z[i,20]}
```

The number of subjects, n, is a constant in the program. The first statement specifies that the i-th response (log income of the i-th worker) has a normal distribution with mean m_i and precision $\tau_\epsilon = \sigma_\epsilon^{-2}$. The second statement provides the structure of the conditional mean function, $m_i = m(x_i)$. Here `beta[]` denotes the 2×1 dimensional vector $\boldsymbol{\beta} = (\beta_0, \beta_1)$, which is the vector of fixed effects parameters. The ith row of matrix $\boldsymbol{X}$ is $\boldsymbol{X}_i = (1, x_i)$. Similarly, `b[]` denotes the 20×1 dimensional vector $\boldsymbol{b} = (b_1, \ldots, b_k)$ of random coefficients. Both the matrix $\boldsymbol{X}$ and $\boldsymbol{Z} = \boldsymbol{Z}_K \boldsymbol{\Omega}_K^{-1/2}$ are design

CODRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

http://biostats.bepress.com/jhubiostat/paper40

matrices obtained outside WinBUGS and are entered as data. In Section 9 we discuss an auxiliary R program that calculates these matrices and uses the `R2WinBUGS` package to call WinBUGS from R. Such programs would be especially useful in a simulation study. The formulae for `mre110[]` and `mre1120[]` could be shortened using the inner product function `inprod`. However, depending on the application, computation time can be 5 to 10 times longer when `inprod` is used.

The distribution of the random coefficients $\boldsymbol{b}$ is represented in WinBUGS as

```
for (k in 1:num.knots){b[k]~dnorm(0,taub)}
```

This specifies that the b_k are independent and normally distributed with mean 0 and precision $\tau_b = \sigma_b^{-2}$. Here `num.knots` is the number of knots ($K = 20$) and is introduced in WinBUGS as a constant. The prior distributions of model parameters described in equation (4) are specified in WinBUGS as follows

```
for (l in 1:2){beta[l]~dnorm(0,1.0E-6)}
taueps~dgamma(1.0E-6,1.0E-6)
taub~dgamma(1.0E-6,1.0E-6)
```

The prior normal distributions for the $\boldsymbol{\beta}$ parameters are expressed in terms of the precision parameter and the Gamma distributions are specified for the precision parameters $\tau_\epsilon = \sigma_\epsilon^{-2}$ and $\tau_b = \sigma_b^{-2}$.

Note that the code is very short and intuitive presenting the model specification in rational steps. After writing the program one needs to load the data: the n–dimensional vector `response` (y) and the design matrices `X[,]` ($\boldsymbol{X}$) and `Z[,]` ($\boldsymbol{Z}$), the sample size `n` (n), the number of knots `num.knots` (K). At this stage the program needs to be compiled and initial values for all random variables have to be loaded.

3.3 Model inference

Convergence to the posterior distributions was assessed using several initial values of model parameters and visually inspecting several chains corresponding to the model parameters. Convergence was attained in less than $1,000$ simulations, but we discarded the first $10,000$ burn-in simulations. For inference we used $90,000$ simulations. These simulations took approximately 6 minutes on a PC (3.6GB RAM, 3.4GHz CPU).

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

Hosted by The Berkeley Electronic Press

Table 1: Posterior median and 95% credible interval for some parameters of model (3) for the Canadian age–income data

Parameter	2.5%	50%	97.5%
β_0	10.12	14.47	19.79
β_1	−0.14	−0.02	0.08
σ_b	0.0029	0.0066	0.0161
σ_ϵ	0.48	0.53	0.59

Table 1 shows the posterior median and a 95% credible interval for some of the model parameters. We also obtained the posterior distributions of the mean function of the response, $m_i = m(x_i)$. Figure 1 displays the median, 2.5% and 97.5% quantiles of these posterior distributions for each value of the covariate x_i. The greyed area corresponds to pointwise credible intervals for each $m(x_i)$ and is not a joint credible band for the mean function. An important advantage of Bayesian over the typical frequentist analysis is that in the Bayesian case the credible intervals take into account the variability of each parameter and do not use the "plug-in" method. Prediction intervals at an in-sample x value can be obtained very easily by monitoring random variables of the type

$$y_i^* = m_i + \epsilon_i^* \ ,$$

with ϵ_i^* being independent realizations of the distribution $N(0, \sigma_\epsilon^2)$. This can be implemented by adding the following lines to the WinBUGS code

```
for (i in 1:n)
    {epsilonstar[i]~dnorm(0,taueps)
     ystar[i]<-m[i]+epsilonstar[i]}
```

4 The Wage–Union Membership Data

Figure 2 displays data on wages and union membership for 534 workers described by Berndt [1]. The data were taken from the Statlib website at Carnegie Mellon University `lib.stat.cmu.edu/`. This data set was analyzed by Ruppert, Wand and Carroll [23] who show that standard linear, quadratic and cubic logistic regression are

COBRA
Collection of Biostatistics
Research Archive

http://biostats.bepress.com/jhubiostat/paper40

not appropriate in this case. Instead, they model the logit of the union membership probability as a penalized spline, which allows identification of features that are not captured by standard regression techniques. In this section we show how to implement a semiparametric Bernoulli regression in WinBUGS using low–rank thin plate splines.

4.1 Generalized P–spline model

Denote by y the binary union membership variable, by x the continuous wage variable and by $p(x)$ the union membership probability for a worker with wage x in dollars/hour. The logit of $p(x)$ is modeled nonparametrically using a linear ($p = 1$) penalized spline with $K = 20$ knots. We used the following model

$$\begin{cases} y_i|x_i & \sim & \text{Bernoulli}\{p(x_i)\} \\ \text{logit}\{p(x_i)\} & = & \beta_0 + \beta_1 x_i + \sum_{k=1}^{K} b_k z_{ik} \\ b_k & \sim & N(0, \sigma_b^2) \\ \epsilon_i & \sim & N(0, \sigma_\epsilon^2) \end{cases}, \tag{5}$$

where z_{ik} is the (i,k)th entry of the design matrix $\boldsymbol{Z} = \boldsymbol{Z}_K \boldsymbol{\Omega}_K^{-1/2}$ defined in Section 2. The following prior distributions were used

$$\begin{cases} \beta_0,\ \beta_1 & \sim & N(0, 10^6) \\ \sigma_b^{-2} & \sim & \text{Gamma}\left(10^{-6}, 10^{-6}\right) \end{cases}. \tag{6}$$

4.2 WinBUGS program for wage–union data

While model (5) is very similar to model (3) the Bayesian analysis implementation in MATLAB, C or other software is significantly different. Typically, when the model is changed one needs to rewrite the entire code and make sure that all code bugs have been removed. This is a lengthy process that requires a high level of expertise in statistics and MCMC coding. WinBUGS cuts short this difficult process, thus making Bayesian analysis appealing to a larger audience.

In this case, changing the model from (3) to (5) requires only small changes in the WinBUGS code. Specifically, the two lines specifying the conditional distribution of the response variable are replaced with

```
for (i in 1:n)
    {response[i]~dbern(p[i])
     logit(p[i])<-mfe[i]+mre110[i]+mre1120[i]}
```

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

Hosted by The Berkeley Electronic Press

while the rest of the code remains practically unchanged. Given this very simple change, we do not provide the rest of the code here, but we provide a commented version in the accompanying software file.

4.3 Model inference

Table 2 shows the posterior median and a 95% credible interval for some of the model parameters. We also obtained the posterior distributions of $p_i = p(x_i)$ and Figure 2 displays the median, 2.5% and 97.5% quantiles of these distributions. The greyed area corresponds to credible intervals for each $p(x_i)$ and is not a joint credible band. The credible intervals take into account the variability of each parameter. Convergence was attained in less than $1,000$ simulations, but we discarded the first $10,000$ burn-in simulations. For inference we used $90,000$ simulations. These simulations took approximately 80 minutes on a PC (3.6GB RAM, 3.4GHz CPU).

Table 2: Posterior median and 95% credible interval for some parameters of the model presented in equations (5) and (6)

Parameter	2.5%	50%	97.5%
β_0	-7.48	-4.15	-2.43
β_1	-0.03	0.34	1.08
σ_b	0.045	0.100	0.229

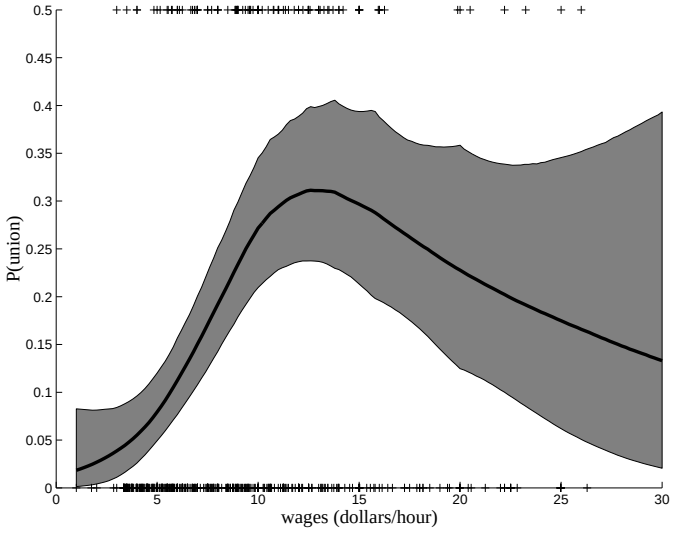
5 The Sitka spruce data

The mixed model representation of penalized splines allows simple extensions additive mixed models. As an example we will use data on the growth of Sitka spruces displayed in Figure 1.3 of Diggle, Heagerty, Liang and Zeger (2002) [7]. The data consist of growth measurements of 79 trees over two seasons: 54 trees were grown in an ozone–enriched atmosphere while the remaining 25 comprise the control group.

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

http://biostats.bepress.com/jhubiostat/paper40

Figure 2: Logistic spline fit to the union and wages scatterplot (solid) with 95% credible sets. Raw data are plotted as pluses, but with values of 1 for union replaced by 0.5 for graphical purposes. A worker making $44.50/hour was used in the fitting but not shown to increase detail.



5.1 Additive mixed models

A useful mixed model for the Sitka data is

$$\begin{cases} y_{ij} &= U_i + \alpha w_i + f(x_{ij}) + \epsilon_{ij} \\ U_i &\sim N(0, \sigma_U^2) \end{cases} , \tag{7}$$

where y_{ij}, $1 \leq i \leq 79$, $1 \leq j \leq 13$, is the log size of spruce i at the time of measurement j taken on day x_{ij}. Also U_i are independent random intercepts for each tree, w_i is the ozone exposure indicator and ϵ_{ij} are random errors. We model $f(\cdot)$ using low–rank thin–plate splines

$$\begin{cases} f(x_{ij}) &= \beta_0 + \beta_1 x_{ij} + \sum_{k=1}^{K} b_k z_{ijk} \\ b_k &\sim N(0, \sigma_b^2) \end{cases} , \tag{8}$$

where the x_{ij} observations are stacked in one vector and (ij) corresponds to the $\{13 * (i-1) + j\}$th observation. Here z_{ijk} is the $(13 * (i-1) + j, k)$th entry of the design matrix $\boldsymbol{Z} = \boldsymbol{Z}_K \boldsymbol{\Omega}_K^{-1/2}$ defined in Section 2. The random parameters b_k are assumed independent normal with σ_b^2 controling the shrinkage of the thin–plate spline function towards the first degree polynomial.

COBRA
Columbia Biostatistics
Research Archive

Hosted by The Berkeley Electronic Press

5.2 WinBUGS program for the Sitka spruce data

The WinBUGS program has essentially the same structure as the previous programs. The likelihood part of the program is

```
for (k in 1:n)
    {log.size[k]~dnorm(mu[k],tauepsilon)
     mu[k]<-U[id.num[k]]+alpha*ozone[k]+m[k]
     m[k]<-beta[1]*X[k,1]+beta[2]*X[k,2]+b[1]*Z[k,1]+b[2]*Z[k,2]+
           b[3]*Z[k,3]}
```

The indexing structure is induced by stacking the vectors of observations corresponding to trees. For example the kth observation corresponds to the index (i, j) such that $k = 13 * (i - 1) + j$. The first line of the program specifies that, conditional on its mean y_{ij} are independent with mean μ_{ij} and precision $\tau_\epsilon = 1/\sigma^2_\epsilon$. The second line of of code specifies the structure of the mean function as the sum between a random intercept U_i, the ozone treatment effect αw_i and a nonparametric mean function $f(\cdot)$. The third line describes the mean function $f(\cdot)$ as a low–rank thin–plate spline.

Nested indexing is a powerful feature of WinBUGS and was used here to define the clusters corresponding to trees. To achieve this we defined a new vector `id.num[]` which is the tree indicator. More precisely, `id.num[k]=i` if and only if the kth observation corresponds to tree i. In this way `U[id.num[k]]` is U_i, the random intercept corresponding to tree i, if and only if kth observation corresponds to tree i.

The distribution of random intercepts is specified as

```
for (i in 1:M){U[i]~dnorm(0,tauU)}
```

where $M = 79$ is a constant in the program and `tauU` is the precision $\tau_U = 1/\sigma^2_U$ of the random intercept. The rest of the program is identical to the program for age–log income data and is omitted. A file containing the commented program and the corresponding R programs is attached.

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

http://biostats.bepress.com/jhubiostat/paper40

5.3 Model inference

Table 3 shows the posterior median and a 95% credible interval for some of the model parameters. We also obtained the posterior distributions of $f(x_{ij})$ and Figure 3 displays the median, 2.5% and 97.5% quantiles of these distributions. The greyed area corresponds to credible intervals for each $f(x_{ij})$) and is not a joint credible band. Convergence was attained in less than $1,000$ simulations, but we discarded the first $10,000$ burn-in simulations. For inference we used $90,000$ simulations. These simulations took approximately 5.5 minutes on a PC (3.6GB RAM, 3.4GHz CPU).

Table 3: Posterior median and 95% credible interval for some parameters of the model presented in equations (7) and (8)

Parameter	2.5%	50%	97.5%
α	-0.61	-0.31	-0.007
β_0	3.39	7.20	10.99
β_1	-0.92	-0.30	0.32
σ_U	1.87	2.62	3.55
σ_b	0.34	0.70	2.35
σ_ϵ	0.178	0.187	0.195

6 The Coronary sinus potassium data

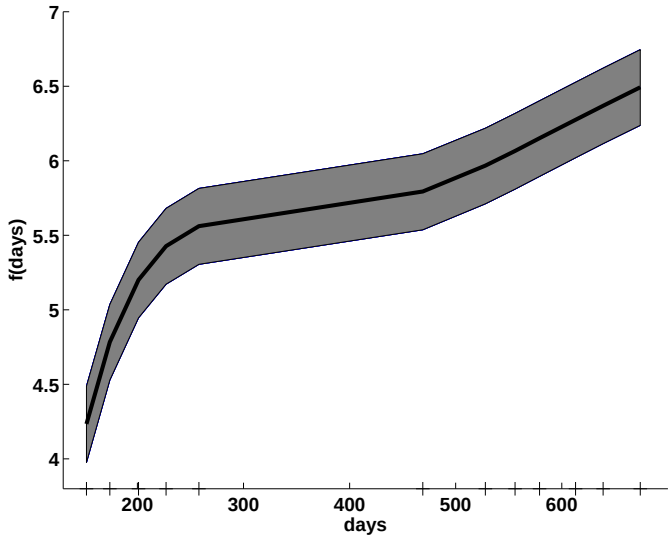
We consider the coronary sinus potassium concentration data measured on 36 dogs published by Grizzle and Allan [16] and Wang [31]. The measurements on each dog were taken every 2 minutes from 1 to 13 minute (7 observations per dog). The 36 dogs come from 4 treatment groups.

Wang [31] presents four smoothing spline analyses of variance models for this data. Crainiceanu and Ruppert [5] also present a hierarchical model of curves including a nonparametric overall mean, nonparametric treatment deviations from the overall curve, and nonparametric subject deviations from the treatment curves. In this section we show how to implement such a complex model in WinBUGS.

CORRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

Hosted by The Berkeley Electronic Press

Figure 3: Thin–plate spline fit for the function $f(\cdot)$ for Sitka spruce data (solid) with 95% credible sets. Sampling days are plotted as pluses.



6.1 Longitudinal Nonparametric ANOVA model

Denote by y_{ij} and t_{ij} the potassium concentration and time for dog i at time j (in this example $t_{ij} = 2j - 1$, but we keep the presentation more general). Consider the following model for potassium concentration

$$y_{ij} = f(t_{ij}) + f_{g(i)}(t_{ij}) + f_i(t_{ij}) + \epsilon_{ij} \ , \tag{9}$$

where $f(\cdot)$ is the overall curve, $f_{g(i)}(\cdot)$ are the deviations of the treatment group from the overall curve and $f_i(\cdot)$ are the deviations of the subject curves from the group curves. Here $g(i)$ represents the treatment group index corresponding to subject i. All three functions are modeled as low–rank thin–plate splines as follows

$$\left\{ \begin{array}{lcl} f(t) & = & \beta_0 + \beta_1 t + \sum_{k=1}^{K_1} b_k z_{tk} \\ f_g(t) & = & \gamma_{0g} I_{(g>1)} + \gamma_{1g} t I_{(g>1)} + \sum_{k=1}^{K_2} c_{gk} z_{tk}^{(g)} \\ f_i(t) & = & \delta_{0i} + \delta_{1i} t + \sum_{k=1}^{K_3} d_{ik} z_{tk}^{(i)} \end{array} \right. \tag{10}$$

where $I_{(g>1)}$ is the indicator that $g > 1$, that is that the treatment group is $g = 2$, or 3 or 4. Here z_{tk} is the (t,k)th entry of the design matrix for the thin–plate spline random coefficients, $\boldsymbol{Z} = \boldsymbol{Z}_K \boldsymbol{\Omega}_K^{-1/2}$ corresponding to the overall mean function $f(\cdot)$. Similarly, we defined $z_{tk}^{(g)}$ and $z_{tk}^{(i)}$ as the (t,k)th entries of the design matrices for

COBRA
Collection of Biostatistics
Research Archive

http://biostats.bepress.com/jhubiostat/paper40

random coefficients corresponding to the group level $f_g(\cdot)$, $1 \leq g \leq 4$, and subject level curves $f_i(\cdot)$, $1 \leq i \leq 36$. The number of knots can be different for each curve and one can choose, for example, more knots to model the overall curve than each subject specific curve. However, in our example we used the same knots for each curve ($K_1 = K_2 = K_3 = 3$).

The model also assumes that the b, c, d and δ parameters are mutually independent and

$$\begin{cases} b_k & \sim & N(0, \sigma_b^2),\ k = 1, \ldots, K_1 \\ c_{gk} & \sim & N(0, \sigma_c^2),\ g = 1, \ldots, 4,\ k = 1, \ldots, K_2 \\ d_{ik} & \sim & N(0, \sigma_d^2),\ i = 1, \ldots, N,\ k = 1, \ldots, K_3 \\ \delta_{0i} & \sim & N(0, \sigma_0^2),\ i = 1, \ldots, N \\ \delta_{1i} & \sim & N(0, \sigma_1^2),\ i = 1, \ldots, N \end{cases}, \tag{11}$$

where σ_b^2, σ_c^2 and σ_d^2 control the amount of shrinkage of the overall, group and individual curves respectively and σ_0^2 and σ_1^2 are the variance components of the subject random intercepts and slopes. We could also add other covariates that enter the model parametrically or nonparametrically, consider different shrinkage parameters for each treatment group, etc. All these model transformations can be done very easily in WinBUGS.

To completely specify the Bayesian nonparametric model one needs to specify prior distributions for all model parameters. The following priors were used

$$\begin{cases} \beta_0,\ \beta_1,\ \gamma_{0g}, \gamma_{1g} & \sim & N(0, 10^6),\ \ g = 1, \ldots, 4 \\ \sigma_b^{-2},\ \sigma_c^{-2}, \sigma_d^{-2},\ \sigma_\epsilon^{-2}, \sigma_0^{-2},\ \sigma_1^{-2} & \sim & \text{Gamma}\left(10^{-6}, 10^{-6}\right) \end{cases}. \tag{12}$$

6.2 WinBUGS program for the dog data

We provide the entire WinBUGS code for this model in Appendix A3. Equation (10) is coded in WinBUGS as

```
for (k in 1:n)
    {response[k]~dnorm(m[k],taueps)
     m[k]<-f[k]+fg[k]+fi[k]
     f[k]<-beta[1]*X[k,1]+beta[2]*X[k,2]+b[1]*Z[k,1]+
           b[2]*Z[k,2]+b[3]*Z[k,3]
     fg[k]<-(gamma[group[k],1]*X[k,1]+gamma[group[k],2]*X[k,2])
            *step(group[k]-1.5)+c[group[k],1]*Z[k,1]+
```

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

Hosted by The Berkeley Electronic Press

```
            c[group[k],2]*Z[k,2]+c[group[k],3]*Z[k,3]
fi[k]<-delta[dog[k],1]*X[k,1]+delta[dog[k],2]*X[k,2]+
            d[dog[k],1]*Z[k,1]+d[dog[k],2]*Z[k,2]+d[dog[k],3]*Z[k,3]}
```

The response is organized as a column vector obtained by stacking the information for each dog. Because there are 7 observations for each dog, the observation number k can be written explicitly in terms of (i, j), that is $k = 7(i-1)+j$. The number of observations is $n = 36 \times 7 = 252$.

We used two $n \times 1$ column vectors with entries `dog[k]` and `group[k]`, that store the dog and treatment group indexes corresponding to the k-th observation.

The first two lines of code in the `for` loop correspond to equation (9), where `dnorm` specifies that `response[k]` has a normal distribution with mean `m[k]` and precision `taueps`. The mean of the response is specified to be the sum of `f[k]`, `fg[k]` and `fi[k]`, which are the variables for the overall mean, treatment group deviation from the mean and individual deviation from the group curves.

The following lines of code in the `for` loop describe the structure of these curves in terms of splines. We keep the same notations from the previous sections. Because in this example we use the same knots and covariates the matrices `X` and `Z` do not change for the three types of curves.

The definition of the overall curve `f[k]` follows exactly the same procedure with the one described in Section 3.2. The definition of `fg[k]` follows the same pattern but it involves two WinBUGS specific tricks. The first one is the use of the `step` function, described in Section 5.2. Here `step(group[k]-1.5)` is 1 if the index of the group corresponding to the k-th observation is larger than 1.5 and zero otherwise. This captures the structure of the $f_g(\cdot)$ function in equation (10) because the possible values of `group[k]` are 1, 2, 3 and 4. The second trick is the nested indexing used in the definition of the $\boldsymbol{\gamma}$ and $\boldsymbol{c}$ parameters using the `dogs` vector described above. For example, the $\boldsymbol{\gamma}$ parameters are stored in a 4×2 matrix `gamma[,]` with the g-th line `gamma[g,]` corresponding to the parameters γ_{0g}, γ_{1g} of the $f_g(\cdot)$ function. Note that if g is replaced by `group[k]` we obtain the parameters corresponding to the k-th observation. Similarly, `c[,]` stores the c_{gk} parameters of $f_g(\cdot)$ and is a 4×3 matrix

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

http://biostats.bepress.com/jhubiostat/paper40

because there are 4 treatment groups and 3 knots. The definition of `fi[k]` curve uses the same ideas, with the only difference that the vector `dog[k]` is used instead of `group[k]`. Here, `delta[,]` is a 36×2 matrix with the i-th line containing the δ_{0i} and δ_{1i}, the random slope and intercept corresponding to the i-th dog. Also, `d[,]` is a 36×3 matrix with the i-th line storing the d_{i1}, d_{i2} and d_{i3}, the parameters of the truncated polynomial functions for the i-th dog.

The WinBUGS coding of the distributions of $\boldsymbol{b}$, $\boldsymbol{c}$, $\boldsymbol{d}$ and $\boldsymbol{\delta}$ follows almost literally the definitions provided in equation (11)

```
for (k in 1:num.knots){b[k]~dnorm(0,taub)}
for (k in 1:num.knots)
    {for (g in 1:ngroups){c[g,k]~dnorm(0,tauc)}}
for (i in 1:ndogs)
    {for (k in 1:num.knots){d[i,k]~dnorm(0,taud)}}
for (i in 1:ndogs)
    {for (j in 1:2){delta[i,j]~dnorm(0,taudelta[j])}}
```

For example, the parameters $c_{j,k}$ are assumed to be independent with distribution $N(0, \sigma_c^2)$ and the WinBUGS code is `c[g,k]~dnorm(0,tauc)`. Here `num.knots`, `ngroups` and `ndogs` are the number of knots of the spline, the number of treatment groups and the number of dogs respectively. These are constants and are entered as data in the program. Using the same notations as in Section 3.2 the normal prior distributions described in equation (12) are coded as

```
for (l in 1:2){beta[l]~dnorm(0,1.0E-6)}
for (l in 1:2)
    {for (j in 1:ngroups){gamma[j,l]~dnorm(0,1.0E-6)}}
```

and the prior gamma distributions on the precision parameters are coded as

```
taub~dgamma(1.0E-6,1.0E-6)
tauc~dgamma(1.0E-6,1.0E-6)
taud~dgamma(1.0E-6,1.0E-6)
```

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

Hosted by The Berkeley Electronic Press

```
taueps~dgamma(1.0E-6,1.0E-6)
for (j in 1:2){taudelta[j]~dgamma(1.0E-6,1.0E-6)}
```

Here `taub`, `tauc`, `taud` and `taueps` are the precisions σ_b^{-2}, σ_c^{-2}, σ_d^{-2} and σ_ϵ^{-2} respectively. `taudelta[1]` and `taudelta[2]` are the precisions σ_0^{-2} and σ_1^{-2} for the δ-parameters.

6.3 Model inference

Figure 4 shows the data for the 36 dogs corresponding to each treatment group together with the posterior mean and 90% credible interval for the treatment group mean functions. Recall that the treatment group functions are the sums between the overall mean function and the functions for the treatment group deviations from the mean functions, that is

$$f_{\text{group}}(t) = f(t) + f_g(t)$$

This is achieved in WinBUGS by monitoring a new variable `fgroup[]` defined as

```
for (k in 1:n){fgroup[k]<-f[k]+fg[k]}
```

For inference we used $90,000$ simulations. These simulations took approximately 4.5 minutes on a PC (3.6GB RAM, 3.4GHz CPU).

7 Improving mixing

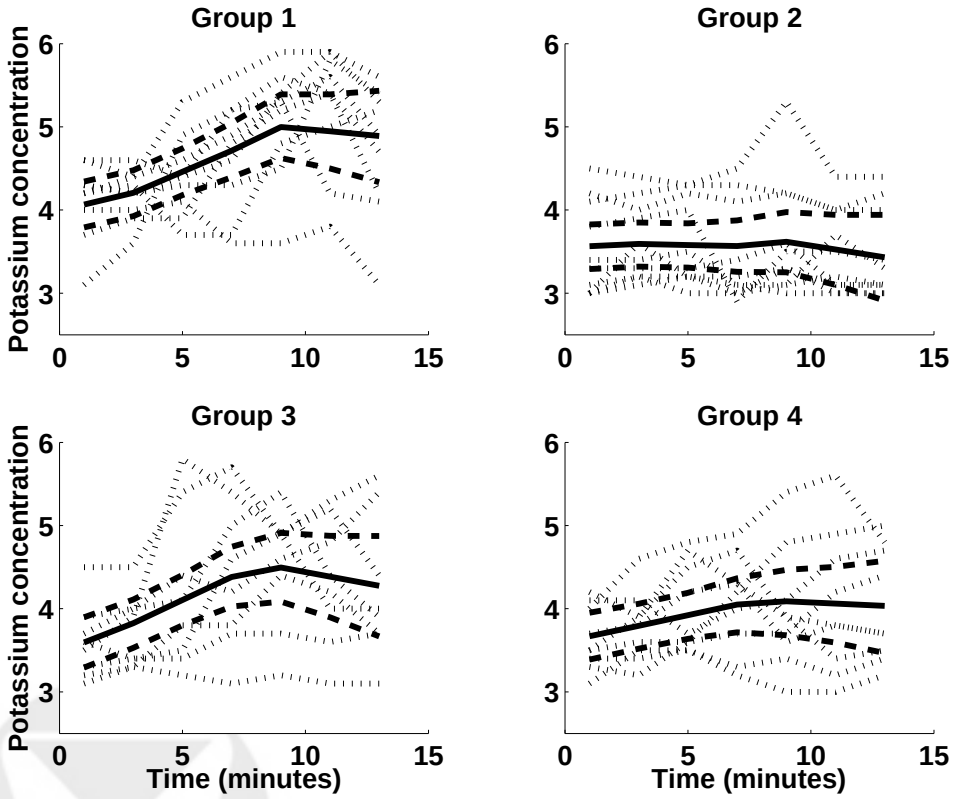
Mixing is the property of the Markov chain to move rapidly throughout the support of the posterior distribution of the parameters. Improving mixing is very important especially when computation speed is affected by the size of data set or model complexity. In this section we present a few simple but effective techniques that help improve mixing.

Model parameterization can dramatically affect MCMC mixing even for simple parametric models. Therefore careful consideration should be given to the complex semiparametric models, such as those considered in this paper. Probably the most important step for improving mixing in this framework is careful choice of the spline basis. While we have experimented with other spline bases, the low–rank thin–plate

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

http://biostats.bepress.com/jhubiostat/paper40

Figure 4: Coronary sinus potassium concentrations for 36 dogs in four treatment groups with posterior median and 90% credible intervals of the group means



COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

Hosted by The Berkeley Electronic Press

splines seem best suited for the MCMC sampling in WinBUGS. This is probably due to the reduced posterior correlation between the spline parameters. The truncated polynomial basis provides similar inferences about the mean function but mixing tends to be very poor with serious implications about the coverage probabilities of the pointwise confidence intervals.

In our experience with WinBUGS, centering and standardizing the covariates also improve, sometimes dramatically, mixing properties of simulated chains.

Another, less known technique is hierarchical centering (Gelfand et al. [12]; Gelfand et al. [13]). Many statistical models contain random effects that are ordered in a natural hierarchy (e.g. observation/site/region). The hierarchical centering of random effects generally has a positive effect on simulation mixing and we recommend it whenever the model contains a natural hierarchy. Bayesian smoothing models presented in this paper also contain the exchangeable random effects, $\boldsymbol{b}$, which are not part of an hierarchy and they cannot be "hierarchically centered".

Crainiceanu et al. [6] show that even for a simple Poisson–Log Normal model the amount of information has a strong impact on the mixing properties of parameters. A practical recommendation in these cases is to improve mixing, as much as possible, for a subset of parameters of interest. These model specification refinements pay off especially in slow WinBUGS simulations.

8 Prior Specification

Any smoother depends heavily on the choice of smoothing parameter, and for P-splines in a mixed model framework, the smoothing parameter is the ratio of the error variance to the prior variance on the mean (Ruppert, Wand and Carroll, 2003 [23]). The smoothness of the fit depends on how these variances are estimated. For example, Crainiceanu and Ruppert (2004), [4] showed that, in finite samples, the (RE)ML estimator of the smoothing parameter is biased towards oversmoothing.

In Bayesian mixed models, the estimates of the variance components are known to be sensitivity to the prior specification, e.g., see Gelman (2004). To study the effect of this sensitivity upon Bayesian P-splines, consider model (3) with one smoothing parameter and homoscedastic errors so that σ_b^2 and σ_ϵ^2 are constant. In terms of the

COBRA
Collection of Biostatistics
Research Archive

http://biostats.bepress.com/jhubiostat/paper40

precision parameters $\tau_b = 1/\sigma_b^2$ and $\tau_\epsilon = 1/\sigma_\epsilon^2$, the smoothing parameter is $\lambda = \tau_\epsilon/\tau_b = \sigma_b^2/\sigma_\epsilon^2$ and a small (large) λ corresponds to oversmoothing (undersmoothing).

It is standard to assume that the fixed effects parameters, β_i, are apriori independent, with prior distributions either $[\beta_i] \propto 1$ or $\beta_i \propto N(0, \sigma_\beta^2)$, where σ_β^2 is very large. In our applications we used $\sigma_\beta^2 = 10^6$, which we recommend if x and y have been standardized or at least have standard deviations with order of magnitude one.

As just mentioned, the priors for the precisions τ_b and τ_ϵ are crucial. We now show how critically the choice of τ_b may depend upon the scaling of the variables. The gamma family of priors for the precisions is conjugate. If $[\tau_b] \sim \text{Gamma}(A_b, B_b)$ and, independently of τ_b, $[\tau_\epsilon] \sim \text{Gamma}(A_\epsilon, B_\epsilon)$ where $\text{Gamma}(A, B)$ has mean A/B and variance A/B^2, then

$$[\tau_b|\boldsymbol{Y}, \boldsymbol{\beta}, \boldsymbol{b}, \tau_\epsilon] \sim \text{Gamma}\left(A_b + \frac{K_m}{2}, B_b + \frac{||\boldsymbol{b}||^2}{2}\right) \tag{13}$$

and

$$[\tau_\epsilon|\boldsymbol{Y}, \boldsymbol{\beta}, \boldsymbol{b}, \tau_\epsilon] \propto \text{Gamma}\left(A_\epsilon + \frac{n}{2}, B_\epsilon + \frac{||\boldsymbol{Y} - \boldsymbol{X\beta} - \boldsymbol{Zb}||^2}{2}\right).$$

Also,

$$\text{E}(\tau_b|\boldsymbol{Y}, \boldsymbol{\beta}, \boldsymbol{b}, \tau_\epsilon) = \frac{A_b + K_m/2}{B_b + ||\boldsymbol{b}||^2/2}, \quad \text{Var}(\tau_b|\boldsymbol{Y}, \boldsymbol{\beta}, \boldsymbol{b}, \tau_\epsilon) = \frac{A_b + K_m/2}{(B_b + ||\boldsymbol{b}||^2/2)^2},$$

and similarly for τ_ϵ.

The prior does not influence the posterior distribution of τ_ϵ when both A_b and B_b are small compared to $K_m/2$ and $||\boldsymbol{b}||^2/2$ respectively. Since the number of knots is $K_m \geq 1$ and in most problems considered $K_m \geq 5$, it is safe to choose $A_b \leq 0.01$. When $B_b << ||\boldsymbol{b}||^2/2$ the posterior distribution is practically unaffected by the prior assumptions. When B_b increases compared to $||\boldsymbol{b}||^2/2$, the conditional distribution is increasingly affected by the prior assumptions. $\text{E}(\tau_b|\boldsymbol{Y}, \boldsymbol{\beta}, \boldsymbol{b}, \tau_\epsilon)$ is decreasing in B_b so large B_b compared to $||\boldsymbol{b}||^2/2$ correspond to undersmoothing. Since the posterior variance of τ_b is also decreasing in B_b a poor choice of B_b will likely result in underestimating the variability of the smoothing parameter $\lambda = \tau_\epsilon/\tau_b$ causing too narrow confidence intervals for m. The condition $B_b << ||\boldsymbol{b}||^2/2$ shows that the "noninformativeness" of the gamma prior depends essentially on the scale of the problem, because the size of $||\boldsymbol{b}||^2/2$ depends upon the scaling of the x and y variables. If y is rescaled to $a_y y$

COBRA
Berkeley Electronic Press
University of California, Berkeley
Division of Biostatistics
Research Archive

Hosted by The Berkeley Electronic Press

and x to $a_x x$, then the regression function becomes $a_y m(a_x x)$ whose p-th derivative is $a_y a_x^p m^{(p)}(a_x x)$ so that $||\boldsymbol{b}||^2/2$ is rescaled by the factor $a_y^2 a_x^{2p}$. Thus, $||\boldsymbol{b}||^2/2$ is particularly sensitive to the scaling of x.

A similar discussion holds true for τ_ϵ but now large B_ϵ corresponds to oversmoothing and τ_ϵ does not depend on the scaling of x. In applications it is less likely that B_ϵ is comparable in size to $||\boldsymbol{Y} - \boldsymbol{X\beta} - \boldsymbol{Zb}||^2$, because the latter is an estimator of $n\sigma_\epsilon^2$. If $\widehat{\sigma}_\epsilon^2$ is an estimator of σ_ϵ^2 a good rule of thumb is to use values of B_ϵ smaller than $n\widehat{\sigma}_\epsilon^2/100$. This rule should work well when $\widehat{\sigma}_\epsilon^2$ does not have an extremely large variance.

Alternative to gamma priors are discussed by, for example, Natarajan and Kass (2000), [19] and Gelman (2004), [14]. These have the advantage of requiring less care in the choice of the hyperparameters. However, we find that with reasonable care, the conjugate gamma priors can be used in practice. Nonetheless, exploration of other prior families for P-splines would be well worthwhile, though beyond the scope of this paper.

9 Interface with and processing in R

WinBUGS 1.4 provides a Graphical User Interface (GUI) that is user friendly and provides important information including the chain histories that can be used to asses mixing. However, the WinBUGS script language is relatively limited and is hard to use for effective simulation studies involving repeated calls for WinBUGS.

R2WinBUGS is an R package developed by Sturtz, Ligges and Gelman [25] that calls WinBUGS 1.4 and export results into R. We used this package into our own R function that also does processing of data. R functions for each model described in this paper are attached to this paper. We present here important parts of the R code, while commented R programs are attached to this paper.

The R program starts with

```
data.file.name="smoothing.norm.txt"
program.file.name="scatter.txt"
inits.b=rep(0,20)
inits<-function(){list(beta=c(0,0),b=inits.b,taub=0.01,taueps=0.01)}
```

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

http://biostats.bepress.com/jhubiostat/paper40

```
parameters<-list("lambda","sigmab","sigmaeps","beta","b","ystar")
```

The first two code lines define the file names for data and WinBUGS program respectively. The third and fourth lines define the initial values to be used in the WinBUGS program and the fifth line indicates the name of the parameters to be monitored in the MCMC sampling. These parameters must correspond to parameters in the WinBUGS program. The R program continues with

```
data<-read.table(file=data.file.name,header=TRUE)
attach(data)
n<-length(covariate)
X<-cbind(rep(1,n),covariate)
knots<-quantile(unique(covariate),
seq(0,1,length=(num.knots+2))[-c(1,(num.knots+2))])
```

The first and second lines read and attach the data, the third line defines the sample size, and the fourth line defines the X matrix of fixed effects for the thin–plate spline. The last assignment defines the `num.knots` number of knots at the sample quantiles of the covariate. An important step in using thin–plate splines is to define the $\boldsymbol{Z}_K$, $\boldsymbol{\Omega}_K$ and the design matrix of random coefficients $\boldsymbol{Z} = \boldsymbol{Z}_K \boldsymbol{\Omega}_K^{-1/2}$. The following lines of code achieve this

```
Z_K<-(abs(outer(covariate,knots,"-")))^3
OMEGA_all<-(abs(outer(knots,knots,"-")))^3
svd.OMEGA_all<-svd(OMEGA_all)
sqrt.OMEGA_all<-t(svd.OMEGA_all$v %*%
(t(svd.OMEGA_all$u)*sqrt(svd.OMEGA_all$d)))
Z<-t(solve(sqrt.OMEGA_all,t(Z_K)))
```

At this stage data is defined, WinBUGS is called from R and the output of the program is loaded into R for further processing. The main function for doing this is `bugs()` implemented in the R2WinBUGS package.

```
data<-list("response","X","Z","n","num.knots")
```

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

Hosted by The Berkeley Electronic Press

```
Bayes.fit<- bugs(data, inits, parameters, model.file = program.file.name,
        n.chains = 1, n.iter = n.iter, n.burnin = n.burnin,
        n.thin = n.thin,debug = FALSE, DIC = FALSE, digits = 5,
        codaPkg = FALSE,bugs.directory = "c:/Program Files/WinBUGS14/")
attach.all(Bayes.fit)
```

10 Pros and cons

An advantage of WinBUGS is the simple programming that translates almost literally the Bayesian model into code. This saves time by avoiding the usually lengthy implementations of the MCMC simulation algorithms. For example, total programming time for one model is approximately 1–2 hours. Programs designed by experts for specific problems can be more refined by taking into account properties of the model and using a combination of art and experience to improve mixing and computation time. However, when we compare a WinBUGS with an expert program in terms of computation speed, programming time needs to be taken into account.

WinBUGS allows simple model changes to be reflected in simple code changes, which encourages the practitioner or the expert to investigate a much wider spectrum of models. Expert programs are usually restrictive in this sense.

Our recommendation is to start with WinBUGS, implement the model for the specific data set. If it runs in a reasonable time and has good mixing properties, then continue with WinBUGS. Otherwise consider designing an expert program. Even if one decides to use the expert program we still recommend using WinBUGS as a method of checking results. Programming errors and debugging time are also dramatically reduced in WinBUGS.

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

http://biostats.bepress.com/jhubiostat/paper40

Appendix A1: WinBUGS code for the age-income example.

Model. This is the complete code for scatterplot smoothing used in the age-income example.

```
    model{                                  #Begin model
#This model can be used for any simple scatterplot smoothing. It
#can be easily modified to accommodate other covariates and/or
#random effects

#Likelihood of the model
    for (i in 1:n)
        {response[i]~dnorm(m[i],taueps)
         m[i]<-mfe[i]+mre110[i]+mre1120[i]
         mfe[i]<-beta[1]*X[i,1]+beta[2]*X[i,2]
         mre110[i]<-b[1]*Z[i,1]+b[2]*Z[i,2]+b[3]*Z[i,3]+b[4]*Z[i,4]+
                    b[5]*Z[i,5]+b[6]*Z[i,6]+b[7]*Z[i,7]+b[8]*Z[i,8]+
                    b[9]*Z[i,9]+b[10]*Z[i,10]
         mre1120[i]<-b[11]*Z[i,11]+b[12]*Z[i,12]+b[13]*Z[i,13]+b[14]*Z[i,14]+
                     b[15]*Z[i,15]+b[16]*Z[i,16]+b[17]*Z[i,17]+b[18]*Z[i,18]+
                     b[19]*Z[i,19]+b[20]*Z[i,20]}

#Prior distributions of the random effects parameters
    for (k in 1:num.knots){b[k]~dnorm(0,taub)}

#Prior distribution of the fixed effects parameters
    for (l in 1:2){beta[l]~dnorm(0,1.0E-6)}

#Prior distributions of the precision parameters
    taueps~dgamma(1.0E-6,1.0E-6); taub~dgamma(1.0E-6,1.0E-6)
```

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

Hosted by The Berkeley Electronic Press

```
#Deterministic transformations. Obtain the standard deviations and
#the smoothing parameter
    sigmaeps<-1/sqrt(taueps);sigmab<-1/sqrt(taub)
    lambda<-pow(sigmab,2)/pow(sigmaeps,2)

#Predicting new observations
    for (i in 1:n)
        {epsilonstar[i]~dnorm(0,taueps)
         ystar[i]<-m[i]+epsilonstar[i]}

    }                                    #end model
```

Data. Data consists of the response variable (`response[]`) design matrix for fixed effects (`X[,]`) design matrix of random effects (`Z[,]`) sample size (`n`), and number of knots (`num.knots`).

Initial values. Initial values are provided for the fixed effects β (`beta[]`) random coefficients b (`b[]`) precision τ_b (`taub`) and precision τ_ϵ (`taueps`). All other initial values are generated by WinBUGS from their prior distributions.

Both data and initial values are specified and processed in R and then used in WinBUGS through the `bugs()` function implemented in the R2WinBUGS package as described in Section 9.

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

http://biostats.bepress.com/jhubiostat/paper40

Appendix A2: WinBUGS code for the wage–union and Sitka spruce examples.

Omitted here because they are similar to the age–income example. For complete commented programs see the models "P-spline fitting with Bernoulli variation" and "Additive mixed model" in the attached WinBUGS file.

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

Hosted by The Berkeley Electronic Press

Appendix A3: WinBUGS code for coronary sinus potassium example.

Model. This is the complete code for the Bayesian semiparametric model for coronary sinus potassium example presented in Section 6.

```
    model{                                  #Begin model
#This model was designed for the coronary sinus potassium model
#described in this paper. However, the basic coding ideas can be
#applied more generally to longitudinal models that involve a
#hierarchy of parametric and/or nonparametric curves


#Likelihood of the model
    for (k in 1:n)
        {response[k]~dnorm(m[k],taueps)
         m[k]<-f[k]+fg[k]+fi[k]
         f[k]<-beta[1]*X[k,1]+beta[2]*X[k,2]+b[1]*Z[k,1]+
               b[2]*Z[k,2]+b[3]*Z[k,3]
         fg[k]<-(gamma[group[k],1]*X[k,1]+gamma[group[k],2]*X[k,2])
            *step(group[k]-1.5)+c[group[k],1]*Z[k,1]+
            c[group[k],2]*Z[k,2]+c[group[k],3]*Z[k,3]
         fi[k]<-delta[dog[k],1]*X[k,1]+delta[dog[k],2]*X[k,2]+
                d[dog[k],1]*Z[k,1]+d[dog[k],2]*Z[k,2]+d[dog[k],3]*Z[k,3]}

#Prior for the random parameters of the overall curve
    for (k in 1:num.knots){b[k]~dnorm(0,taub)}

#Prior for the random parameters for the curves describing group
#deviations from the overall curve
    for (k in 1:num.knots)
        {for (g in 1:ngroups){c[g,k]~dnorm(0,tauc)}}

#Prior for the random parameters for the individual deviations
```

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

http://biostats.bepress.com/jhubiostat/paper40

```
#from the group curve
    for (i in 1:ndogs)
        {for (k in 1:num.knots){d[i,k]~dnorm(0,taud)}}

#Prior for monomial parameters of the overall curve
    for (l in 1:2){beta[l]~dnorm(0,1.0E-6)}

#Prior for monomial parameters of curves describing the group
#deviations from the overall curve
    for (l in 1:2)
        {for (j in 1:ngroups){gamma[j,l]~dnorm(0,1.0E-6)}}

#Prior for monomial parameters of curves describing the individual
#deviations from the group curve
    for (i in 1:ndogs)
        {for (j in 1:2){delta[i,j]~dnorm(0,taudelta[j])}}

#Priors of precision parameters
    taub~dgamma(1.0E-6,1.0E-6)
    tauc~dgamma(1.0E-6,1.0E-6)
    taud~dgamma(1.0E-6,1.0E-6)
    taueps~dgamma(1.0E-6,1.0E-6)
    for (j in 1:2){taudelta[j]~dgamma(1.0E-6,1.0E-6)}

#Define the group curves
    for (i in 1:n){fgroup[i]<-f[i]+fg[i]}
        }                                #End model
```

Data. Data consists of the response variable (`response[]`) design matrix for fixed effects (`X[,]`) and design matrix of random effects (`Z[,]`), sample size (`n`), number of knots (`num.knots`), number of subjects (`nsubjects`), number of groups (`ngroups`),

Research Archive

Hosted by The Berkeley Electronic Press

subject indicator vector (`dog`), and group vector indicator (`group`).

Initial values. Initial values are provided for the fixed effects for all curves β (`beta[]`), γ (`gamma[,]`), δ (`delta[,]`), random coefficients for all curves b (`b[]`), c (`c[,]`), d (`d[,]`), precisions τ_b (`taub`), τ_c (`tauc`), τ_d (`taud`) and precision τ_ϵ (`taueps`).

Both data and initial values are specified and processed in R and then used in WinBUGS through the `bugs()` function implemented in the R2WinBUGS package as described in Section 9.

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

http://biostats.bepress.com/jhubiostat/paper40

References

[1] E.R. Berndt. *The Practice of Econometrics: Classical and and Contemporary*. Addison-Wesley, Reading, MA, 1991.

[2] S. Berry, R. J. Carroll, and D. Ruppert. Bayesian smoothing and regression splines for measurement error problems. *JASA*, 97:160–169, 2002.

[3] B. Brumback, D. Ruppert, and M. P. Wand. Comment on variable selection and function estimation in additive nonparametric regression using data-based prior by shively, kohn, and wood. *Journal of the American Statistical Association*, 94:794–797, 1999.

[4] C. M. Crainiceanu and D. Ruppert. Asymptotic distribution of likelihood ratio tests in linear mixed models. *Journal of the Royal Statistical Society, Series B*, 66:165–185, 2004.

[5] C. M. Crainiceanu and D. Ruppert. Restricted likelihood ratio tests in nonparametric longitudinal models. *submitted*, 2004.

[6] C. M. Crainiceanu, D. Ruppert, J. R. Stedinger, and C. T. Behr. *Improving MCMC mixing for a GLMM Describing Pathogen Concentrations in Water Supplies. In Case Studies in Bayesian Statistics 6 (eds. C. Gatsonis, R.E. Kass, A. Carriquiry, A. Gelman, D. Higdon, D.K Pauler, I. Verdinelli)*. Springer Verlag, New York, 2002.

[7] P.J. Diggle, P. Heagerty, K.Y. Liang, and S.L. Zeger. *Analysis of Longitudinal Data*. Oxford University Press, New York, 2002.

[8] P.H.C. Eilers and B.D. Marx. Flexible smoothing with b-splines and penalties. *Statistical Science*, 11(2):89–121, 1996.

[9] R.L. Eubank. *Spline smoothing and Nonparametric Regression*. New York: Marcel Dekker, 1988.

[10] J. Fan and I. Gijbels. *Local Polynomial Modeling and its Applications*. Chapman and Hall, London, 1996.

[11] J.H. Friedman. Multivariate adaptive regression splines (with discussion). *The Annals of Statistics*, 19:1–141, 1991.

COBRA
Collection of Biostatistics
Research Archive

Hosted by The Berkeley Electronic Press

[12] A.E. Gelfand, S.K. Sahu, and B.P. Carlin. Efficient parameterizations for normal linear mixed model. *Biometrika*, 3:479–488, 1995.

[13] A.E. Gelfand, S.K. Sahu, and B.P. Carlin. *Efficient parameterizations for normal linear mixed model. In Bayesian Statistics 5 (eds. J.M. Bernardo, J. Berger, A.P. Dawid and A.F.M. Smith).* Oxford University Press, Oxford, 1995.

[14] A. Gelman. Prior distributions for variance parameters in hierarchical models. *manuscript*, 2004.

[15] P.J. Green and B.W. Silverman. *Nonparametric regression and Generalized Linear Models.* Chapman and Hall, London, 1994.

[16] J.E. Grizzle and D.M. Allan. Analysis of dose and dose response curves. *Biometrics*, 25:357–381, 1969.

[17] M.H. Hansen and C. Kooperberg. Spline adaptation in extended linear models (with discussion). *Statistical Science*, 17:2–51, 2002.

[18] T.J. Hastie and R. Tibshirani. *Generalized Additive Models.* Chapman and Hall, 1990.

[19] R. Natarajan and R.E. Kass. Reference bayesian methods for generalized linear mixed models. *J. of the Amer. Statist. Assoc.*, 95:227–237, 2000.

[20] L. Ngo and M.P. Wand. Smoothing with mixed model software. *Journal of Statistical Software*, 9, 2004.

[21] R.T. Ogden. *Essential Wavelets for Statistical Applications and Data Analysis.* Birkhauser, Boston, 1996.

[22] D. Ruppert. Selecting the number of knots for penalized splines. *Journal of Computational and Graphical Statistics*, 2002.

[23] D. Ruppert, M.P. Wand, and R. Carroll. *Semiparametric Regression.* Cambridge University Press, 2003.

[24] D. Spiegelhalter, A. Thomas, and N. Best. WinBugs Version 1.3 User Manual*, Medical Research Council Biostatistics Unit, Cambridge.* http://www.mrc-bsu.cam.ac.uk/bugs/winbugs/contents.shtml, 2000.

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

http://biostats.bepress.com/jhubiostat/paper40

[25] S. Sturtz, U. Ligges, and A. Gelman. R2winbugs: A package for running winbugs from r. *Journal of Statistical Software*, 12, 2005.

[26] M.E. Tarter and M.D. Lock. *Model-free Curve Estimation*. Chapman and Hall, New York, 1993.

[27] A. Ullah. Specification analysis of econometric models. *Journal of Quantitative Economics*, 2:187–209, 1985.

[28] G. Wahba. *Spline Models for Observational Data*. Wiley, 1990.

[29] M.P. Wand. Smoothing and mixed models. *submitted to Computational Statistics*, 2002.

[30] M.P. Wand and M.C. Jones. *Kernel Smoothing*. Chapman and Hall, London, 1995.

[31] Y. Wang. Mixed effects smoothing spline analysis of variance. *J. Royal Statistical Society, Series B*, 60:159–174, 1998.

COBRA
A BEPRESS REPOSITORY
Collection of Biostatistics
Research Archive

Hosted by The Berkeley Electronic Press