Bayesian Application of Penalized Thin-Plate Spline Regression Using WinBUGS

Luigi Barba

Bayesian Modelling Final Project
15th February 2021

1 Introduction

The virtues of nonparametric regression models have been discussed extensively in the statistics literature. The main advantage of nonparametric over parametric models resides in 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 work we focus on a particular semiparametric regression model using thin-plate penalized splines. It is becoming more widely appreciated that penalized likelihood models can be viewed as particular cases of Generalized Linear Mixed Models (GLMMs).

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 nonparametric models and the exact inference provided by the Bayesian inferential machinery. It is this combination that makes Bayesian nonparametric modeling so attractive.

2 Model overview

Consider the regression model:

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

where ϵ_i are i.i.d. $N(0, \sigma_\epsilon^2)$, ϵ_i is independent x_i, and $m(\cdot)$ is a smoothing function.

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.

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

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

where $\theta = (\beta_0, \beta_1, u_1, \ldots, u_K)^T$ is the vector of regression coefficients, and $\kappa_1 < \kappa_2 < \cdots < \kappa_K$ are fixed knots. To ensure the desired flexibility, we consider a number of knots that is large enough (typically 5 to 20) and 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 introduce a penalized loss function:

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

where λ is the smoothing parameter and $\mathbf{D}$ is a known positive semi-definite penalty matrix.

The thin-plate spline penalty matrix is:

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

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

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

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

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

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}_K \boldsymbol{\upsilon} + \boldsymbol{\epsilon}, \ \operatorname{cov} \begin{pmatrix} \mathbf{u} \\ \boldsymbol{\epsilon} \end{pmatrix} = \begin{bmatrix} \sigma_u^2 (\boldsymbol{\Omega}_K^{1/2})^T \boldsymbol{\Omega}_K^{1/2} & 0 \\ 0 & \sigma_\epsilon^2 I_n \end{bmatrix}.$$

Using the reparametrization $\mathbf{b} = \boldsymbol{\Omega}_K^{1/2} u$ and defining $\mathbf{Z} = \mathbf{Z}_K \Omega_K^{1/2}$, the mixed model (2) is equivalent to:

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{b} + \boldsymbol{\epsilon}, \ \operatorname{cov} \begin{pmatrix} \mathbf{u} \\ \boldsymbol{\epsilon} \end{pmatrix} = \begin{bmatrix} \sigma_b^2 I_K & 0 \\ 0 & \sigma_\epsilon^2 I_n \end{bmatrix}. \tag{3}$$

The mixed model (3) could be fit in a frequentist framework using Best Linear Unbiased Predictor (BLUP) or Penalized Quasi-Likelihood (PQL) estimation. In this work we adopt a Bayesian inferential perspective, by placing priors on the model parameters and simulating their joint posterior distribution.

3 Model Implementation

In most of the simulations, I used a WinBUGS implementation of the model (3) provided by the reference paper.

```
model{

#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)

#Deterministic transformations
```

```
  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]}
  }
```

This model takes in input the vector of the dependent variabile `response`, the design matrices `X` (in the form $(\underline{\mathbf{1}}, \underline{\mathbf{x}})$) and `Z`, the sample size `n` and the number of knots `num.knots`. A predictor `ystar` is also generated.

The prior specification is also crucial, especially for the precision parameters. In our experiment we want to choose a non-informative prior. In the reference paper (Ch. 8) it is shown how to choose the right precision prior parameters, which has a dependence on the scale of the data. In this work I adopted the following priors:

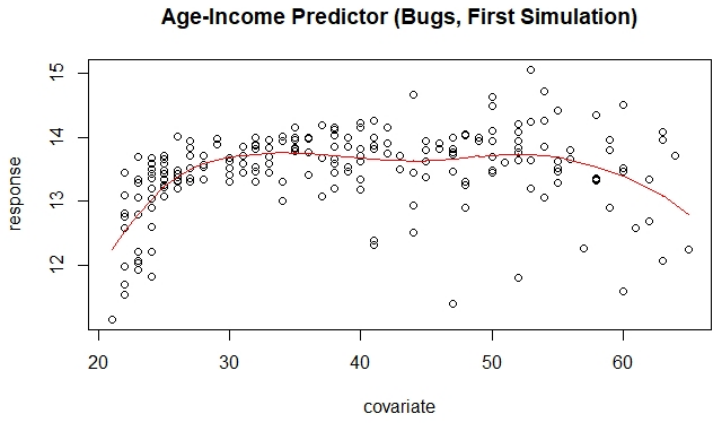
$$\begin{cases} \beta_0, \beta_1 & \sim & N(0,1) \\ \sigma_b^{-2}, \sigma_\epsilon^{-2} & \sim & \text{Gamma}(10^{-6}, 10^{-6}) \end{cases}$$

4 Experimental simulations

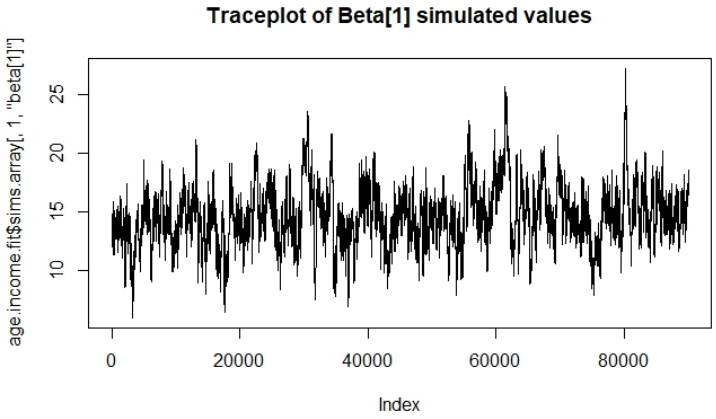
I tested two different datasets, and with each of them I tried to accomplish different goals. In the first one I try to test the robustness of the Hastings-Metropolis algorithm, while in the second I test the sensitivity of the model to the number of knots.

4.1 The Canadian age–income data

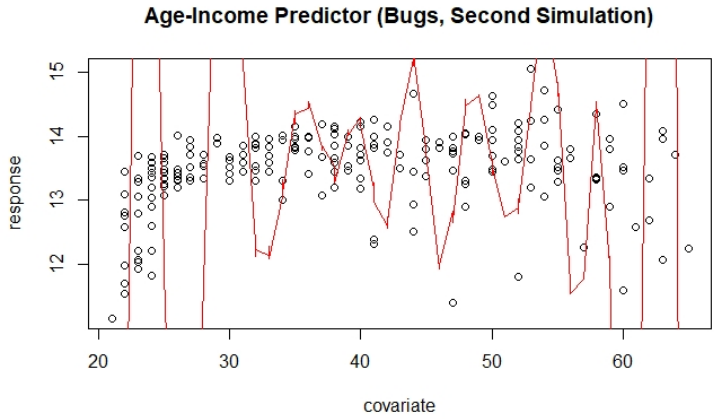
The `age.income` dataset from the `SemiPar` package is the one proposed by the reference paper, and it contains the age and the log(income) of a sample of $n = 205$ Canadian workers, all of whom were educated to grade 13. I run a total of 4 different simulations using the model with 20 knots; two of them implemented the `bugs` function which makes use of the Metropolis-Hastings algorythm, while the remaining two using the `jags` function from the `R2jags` package, which implements the Gibbs Sampling algorythm. As I said before, for both the algorythms I run two simulations each, one with the starting points proposed by the paper, while the other one with some "bad" ones.



(a)

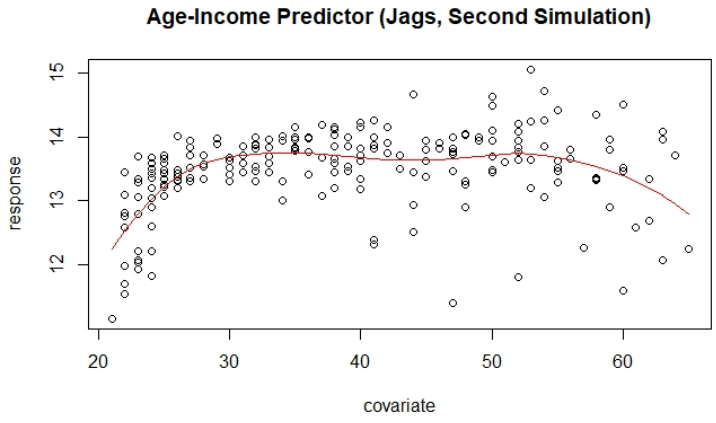


(b)

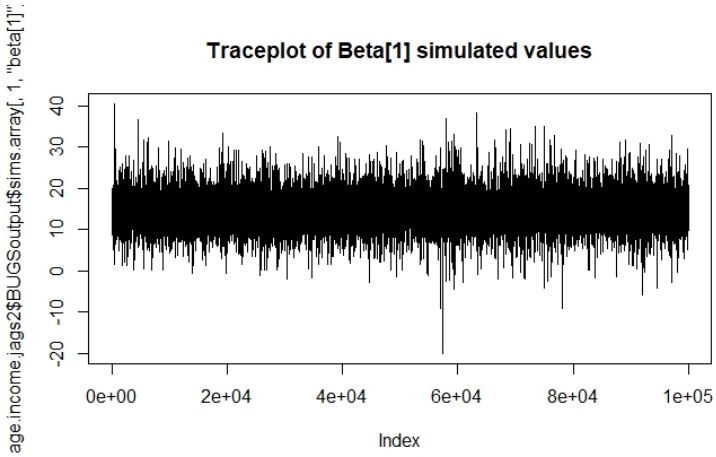


(c)

Figure 1: The Metropolis-Hastings algorithm is very dependent on the starting points: in the first simulation I used the points provided by the paper, while in the second some "bad" ones. This is caused by the fact that the parameters' values of the simulations do not converge to a certain value, as we can see in the traceplot.



(a)



(b)

Figure 2: Jags' Gibbs-Sampler on the contrary provides a more robust convergence. In both the simulations I run the same model was delivered. We also can see the traceplot resembling a stationary process

Results showed that, while the `jags` function achieves the same results of the `bugs` function, the latter is too susceptible to the choice of starting points. Gibbs-Sampling proved to be a more robust method, and this can be clearly seen by the simulations plot, which converges to the same results in both of the simulations I run.

Age-Income Data (Bugs, DIC = 331.386)			
Parameter	2.5%	50%	97.5%
Beta[1]	10.12	14.47	19.79
Beta[2]	0.14	0.02	0.08
σ_b	0.0029	0.0066	0.0161
σ_ϵ	0.48	0.53	0.59

Age-Income Data (Jags, DIC = 334.693)			
Parameter	2.5%	50%	97.5%
Beta[1]	9.44	14.37	20.22
Beta[2]	-0.15	-0.018	0.09
σ_b	0.0029	0.0069	0.0178
σ_ϵ	0.48	0.53	0.59

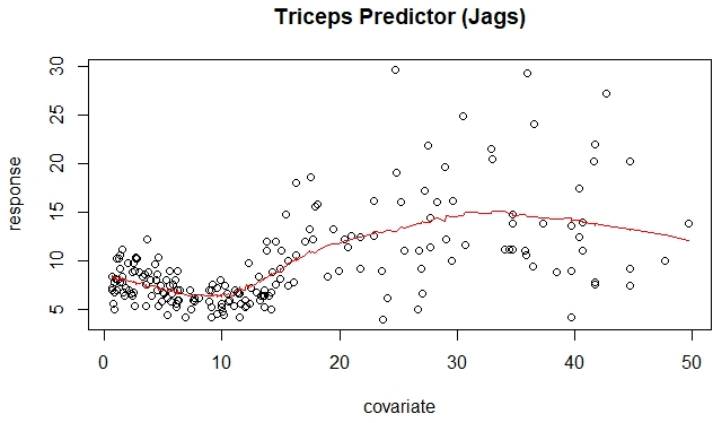
4.2 The Triceps data

The `triceps` dataset is derived from an anthropometric study of 892 females under 50 years in three Gambian villages in West Africa whose the triceps skinfold thichness was measured andreported along with their age. In my experiment I sampled 200 observations.

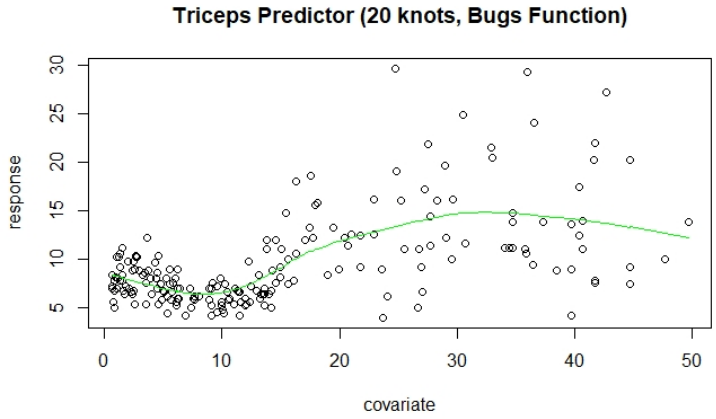
In the context of this dataset, the Jags' Gibbs-Sampler provides a very noisy and undersmoothed predictor. Because of this, I conducted two simulations with the Bugs' Metropolis-Hastings, implementing first the usual model with 20 knots and anotherone with only 10 knots. As we can see from the Figure 2, the models are approximately identical. The DIC criterion has only a 0.1% improvement with the introduction of the 10 additional knots.

Triceps Data (Bugs, 20 Knots, DIC = 1094.91)			
Parameter	2.5%	50%	97.5%
Beta[1]	-11.3802	3.2	23.5402
Beta[2]	-0.6435	0.4091	1.141
σ_b	0.01887	0.04868	0.1471
σ_ϵ	3.319	3.662	4.065

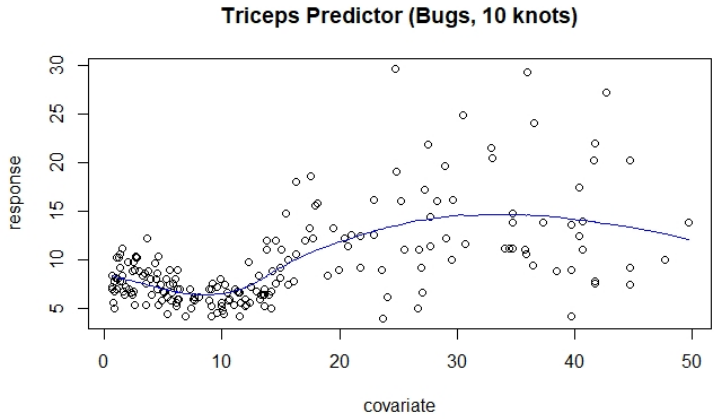
Triceps Data (Bugs, 10 Knots, DIC = 1095.03)			
Parameter	2.5%	50%	97.5%
Beta[1]	-15.5	2.275	33.6602
Beta[2]	-1.526	0.4332	1.588
σ_b	0.019	0.05221	0.1946
σ_ϵ	3.323	3.6650	4.07



(a)



(b)



(8)

Figure 3: Jags' predictor is very noisy and undersmoothed. For the Triceps dataset, th Bugs function delivered better models.

References

[1] Crainiceanu, Ciprian, David Ruppert, & Matthew P. Wand. *Bayesian Analysis for Penalized Spline Regression Using WinBUGS* Journal of Statistical Software, 14.14 (2005): 1 - 24.

[2] Ioannis Ntzoufras, *Bayesian Modeling Using WinBUGS*. Wiley, 2009.