Bayesian Analysis (0000) **00**, Number 0, pp. 1

Supplementary Materials: Bayesian Functional Linear Regression with Sparse Step Functions

Paul-Marie Grollemund[*,†] Christophe Abraham[*] Meïli Baragatti[*] Pierre Pudlo[‡]

Contents

[*]IMAG UMR 5149, Université de Montpellier, CNRS, Place E. Bataillon, 34095 Montpellier CEDEX, France paul-marie.grollemund@umontpellier.fr
[†]MISTEA UMR 729, INRA, Montpellier SupAgro, Place Pierre Viala, 34060 Montpellier CEDEX, France christophe.abraham@supagro.fr meili.baragatti@supagro.fr
[‡]I2M UMR 7373, Aix-Marseille Université, CNRS, Centrale Marseille, Rue F. Joliot Curie, 13453 Marseille CEDEX 13, France pierre.pudlo@univ-amu.fr

 DOI: 0000

1 Implementation

1.1 Tuning the Hyperparameters

We discuss our recommendation on the hyperparameters of the model, given at the end of Section 2.2 of the paper. For this study, we applied our methodology on Dataset 1 and fixed the hyperparameters v_0, v, a around the recommended values. Remember that Dataset 1 is a synthetic dataset simulated with a coefficient function that is a Step Function (the black curve of Figure 2 of the paper), with a high level of signal over noise ($r = 5$) and with a low level of autocorrelation within the covariate ($\zeta = 1$). The following values are considered for each hyperparameter:

- for a: $2K$, $5K$, $10K$, $15K$ and $20K$;
- for v: 10, 5, 2, 1 and 0.5;
- and for K: any integer between 1 and 10.

The numerical results are given in Table 1. The default values we recommend are not the best values here, but we have done numerous other trials on many synthetic datasets and these choices are relatively robust. should) be chosen with the Bayesian model choice machinery.

1.2 Computation and visualization of the posterior distribution

The full posterior distribution can be written explicitly from the Bayesian model given in Equations (9). As usual with hierarchical models, sampling from the posterior distribution $\pi_K(\theta|\mathcal{D})$ can be done with a Gibbs algorithm (see, e.g., Robert and Casella, 2013, Chapter 7). The details of the MCMC algorithm are given in Section 1.3 for the case of one single functional covariate and in Section 1.4 for the case of several functional covariates.

Now, for simplicity of the notation, we focus on the single functional covariate case. Let $\theta(s)$, $s = 1, \ldots, N$, denote the output of the MCMC sampler after the burn-in period. The computation of the Bayes estimate $\widehat{S}_\gamma(\mathcal{D})$ of the support as defined in Theorem 1 depends on the probabilities $\alpha(t|\mathcal{D})$. With the Monte Carlo sample from the MCMC, we can easily approximate these posterior probabilities by the frequencies

$$\alpha(t|\mathcal{D}) \approx \frac{1}{N} \sum_{s=1}^{N} \mathbb{1}\{\beta_{\theta(s)}(t) \neq 0\}.$$

Table 1: Performances of Bliss with respect to the tuning of the hyperparameters.

	Error on the β		Error on the support	
	Bliss estimate	L^2-estimate	Support of the stepwise Bliss estimate	Bayes support estimate
$a = 2K$	1.000	0.698	0.222	0.439
$a = 5K$ ♡	1.013	1.135	0.222	0.192
$a = 10K$	1.642	1.364	0.242	0.202
$a = 15K$	3.060	1.645	0.364	0.212
$a = 20K$	2.032	1.888	0.263	0.263
$v = 10$	1.628	1.125	0.242	0.192
$v = 5$ ♡	1.711	1.131	0.242	0.192
$v = 2$	1.082	1.143	0.273	0.192
$v = 1$	1.207	1.119	0.273	0.192
$v = 0.5$	1.675	1.129	0.263	0.192
$K = 1$	1.798	1.782	0.424	0.449
$K = 2$	0.993	1.101	0.222	0.222
$K = 3$	1.696	1.124	0.242	0.192
$K = 4$	1.736	1.159	0.283	0.172
$K = 5$	2.081	1.233	0.303	0.172
$K = 6$	2.177	1.243	0.283	0.202
$K = 7$	2.135	1.221	0.303	0.232
$K = 8$	1.343	1.184	0.263	0.242
$K = 9$	1.439	1.166	0.263	0.328
$K = 10$	1.897	1.089	0.364	0.348

The ♡ symbol indicates the default values.

What remains to be computed are the approximations of $\widehat{\beta}_{L^2}(\cdot)$ and $\widehat{\beta}^{\varepsilon}_{K_0}(\cdot)$ based on the MCMC sample. First, the Monte Carlo approximation of (14) is given by

$$\widehat{\beta}_{L^2}(t) \approx \frac{1}{N}\sum_{s=1}^{N} \beta_{\theta(s)}(t).$$

More interestingly, the Bayes estimate $\widehat{\beta}^{\varepsilon}_{K_0}(\cdot)$ can be computed by minimizing

$$\left\| d(\cdot) - \widehat{\beta}_{L^2}(t) \right\|^2$$

over the set $\mathcal{E}^{\varepsilon}_{K_0}$. To this end we run a Simulated Annealing algorithm (Kirkpatrick et al., 1983), described in Section 1.5.

We also provide a striking graphical display of the posterior distribution on the set $\mathcal{E}_K$ with a heat map. More precisely, the aim is to sketch all marginal posterior distributions $\pi^t_K(\cdot|\mathcal{D})$ of $\beta_\theta(t)$ for any value of $t \in \mathcal{T}$ in one single figure. To this end we introduce the probability measure Q on $\mathcal{T} \times \mathbb{R}$ defined as follows. Its marginal distribution over $\mathcal{T}$ is uniform, and given the value t of the first coordinate, the second coordinate is distributed according to the posterior distribution of $\beta_\theta(t)$. In other words,

$$(t, b) \sim Q \quad \Longleftrightarrow \quad t \sim \text{Unif}(\mathcal{T}),\ b|t \sim \pi^t_K(\cdot|\mathcal{D}).$$

We can easily derive an empirical approximation of Q from the MCMC sample $\{\theta(s)\}$ of the posterior. Indeed, the first marginal distribution of Q, namely Unif($\mathcal{T}$) can be approximated by a regular grid t_i, $i = 1, \ldots, M$. And, for each value of i, set $b_{is} =$

$\beta_{\theta(s)}(t_i)$, $s = 1, \ldots, N$. The resulting empirical measure is

$$\widehat{Q} = \frac{1}{M\,N} \sum_{i=1,\ldots,M} \sum_{s=1,\ldots,N} \delta_{(t_i, b_{is})},$$

where $\delta_{(t,b)}$ is the Dirac measure at (t, b). The graphical display we propose is representing $\widehat{Q}$ with a heat map on $\mathcal{T} \times \mathbb{R}$. Each small area of $\mathcal{T} \times \mathbb{R}$ is thus coloured according to its $\widehat{Q}$-probability. This should be done cautiously as the marginal posterior distribution $\pi_K^t(\cdot|\mathcal{D})$ has a point mass at zero: $\pi_K^t(b = 0|\mathcal{D}) > 0$ by construction of the prior distribution. Finally the colour scale can be any monotone function of the probabilities, in particular non linear functions to handle the atom at 0. Examples are provided in Section 3 in Figures 1 and 2.

Remark In practice the whole function x_i may be unknown and only observed at a finite set of time points $\{t_{ij}, j = 1, \ldots, n_i\}$. The time points may be irregularly spaced and vary between individuals. This common situation of applied functional data analysis is usually handled by converting the discrete measures $\{x_i(t_{ij}), j = 1, \ldots, n_i\}$ to a function computable for any time point by using interpolation or smoothing techniques (see Ramsay and Silverman, 2005 page 9 and chapter 15 of the second edition, or Crambes, Kneip, and Sarda, 2009 page 41). In the present paper, it is worth noting that the whole curve x_i is actually not needed. It is only required to compute the value of the integral $\int_{\mathcal{I}_k} x_i(t)\mathrm{d}t$ for any given interval $\mathcal{I}_k$. Several numerical techniques are available for this purpose when the observed time points are irregular (see, for example, Deheuvels, 1980, Chapter V or Phythian and Williams, 1986). For simplicity, we choose the trapezoidal rule in the simulations as the derived precision is sufficient in our context.

1.3 Gibbs algorithm and Full conditional distributions for a single functional covariate

The full conditional distributions for the Gibbs Sampler in Section 1.2 are the following,

$$\begin{aligned}
\mu, \beta^* | y, \sigma^2, m, \ell &\sim \mathcal{N}_{K+1}\left((\underline{\mathrm{x}}^T\underline{\mathrm{x}} + \underline{\mathrm{V}})^{-1}\underline{\mathrm{x}}y \ , \ \sigma^2(\underline{\mathrm{x}}^T\underline{\mathrm{x}} + \underline{\mathrm{V}})^{-1}\right), \\
\sigma^2 | y, \mu, \beta^*, m, \ell &\sim \Gamma^{-1}\left(\frac{n+K+1}{2}, \frac{1}{2}\mathrm{RSS} + \frac{1}{2}(\mu, \beta^*)^T\underline{\mathrm{V}}^{-1}(\mu, \beta^*)\right), \\
\pi\left(m_k | y, \mu, \beta^*, \sigma^2, m_{-k}, \ell\right) &\propto \exp\left(-\mathrm{RSS}/2\sigma^2\right) \times \pi(\beta^* | m, \ell, \sigma^2) \\
\pi\left(\ell_k | y, \mu, \beta^*, \sigma^2, m, \ell_{-k}\right) &\propto \exp\left(-\mathrm{RSS}/2\sigma^2\right) \times \pi(\ell_k) \times \pi(\beta^* | m, \ell, \sigma^2)
\end{aligned}$$

where $\mathrm{RSS} = \|y - \mu\mathbf{1}_n - x_{\cdot}(\mathcal{I}_{\cdot})\beta^*\|^2$, $\underline{\mathrm{x}} = \Big(\mathbf{1}_n \ \Big| \ x_{\cdot}(\mathcal{I}_{\cdot})\Big)$, and

$$\underline{\mathrm{V}} = \begin{pmatrix} v_0^{-1} & 0 \\ 0 & n^{-1}\Big(x_{\cdot}(\mathcal{I}_{\cdot})^T x_{\cdot}(\mathcal{I}_{\cdot}) + vI_K\Big) \end{pmatrix}.$$

The full conditional distributions for the hyperparameters m_k and ℓ_k are unusual distributions. As the covariate curves x_i are observed on a grid $\mathcal{T}_G = (t_j)_{j=1,\ldots,p}$, we consider

that m_k belongs to $\mathcal{T}_G$ and ℓ_k is defined so that $m_k \pm \ell_k \in \mathcal{T}_G$. Thus, the number of possible values for m_k and ℓ_k is finite and the full conditional distributions of m_k and ℓ_k are easily computable.

1.4 Gibbs Algorithm and Full Conditional Distributions for q Functional Covariates

Remenber $K = \sum_{j=1}^{q} K_j$. We denote

- $\beta_j^* = (\beta_{1j}^*, \ldots, \beta_{K_j j}^*)$ and $\beta^* = (\beta_1^*, \ldots, \beta_q^*)$,
- $m_j = (m_{1j}, \ldots, m_{K_j j})$ and $m = (m_1, \ldots, m_q)$,
- $\ell_j = (\ell_{1j}, \ldots, \ell_{K_j j})$ and $\ell = (\ell_1, \ldots, \ell_q)$.

The full conditional distributions are

$$\begin{aligned}
\mu, \beta^* | y, \sigma^2, m, \ell &\sim \mathcal{N}_{K+1}\left((\underline{\mathrm{x}}^T\underline{\mathrm{x}} + \underline{\mathrm{V}})^{-1}\underline{\mathrm{x}}y \ , \ \sigma^2(\underline{\mathrm{x}}^T\underline{\mathrm{x}} + \underline{\mathrm{V}})^{-1}\right), \\
\sigma^2 | y, \mu, \beta^*, m, \ell &\sim \Gamma^{-1}\left(\frac{n+K+1}{2}, \frac{1}{2}\mathrm{RSS} + \frac{1}{2}(\mu, \beta^*)^T\underline{\mathrm{V}}^{-1}(\mu, \beta^*)\right), \\
\pi\left(m_{kj} | y, \mu, \beta^*, \sigma^2, m_{-(kj)}, \ell\right) &\propto \exp\left(-\mathrm{RSS}/2\sigma^2\right) \times \pi(\beta^* | m, \ell, \sigma^2) \\
\pi\left(\ell_{kj} | y, \mu, \beta^*, \sigma^2, m, \ell_{-(kj)}\right) &\propto \exp\left(-\mathrm{RSS}/2\sigma^2\right) \times \pi(\ell_{kj}) \times \pi(\beta^* | m, \ell, \sigma^2)
\end{aligned}$$

where $\mathrm{RSS} = \left\| y - \mu\mathbf{1}_n - \sum_{j=1}^{q} x_{\cdot j}(\mathcal{I}_{\cdot j})\beta_j^* \right\|^2$, $\underline{\mathrm{x}} = \left(\mathbf{1}_n \ \middle| \ x_{\cdot 1}(\mathcal{I}_{\cdot 1}) \ \middle| \ \ldots \ \middle| \ x_{\cdot q}(\mathcal{I}_{\cdot q})\right)$ and

$$\underline{\mathrm{V}} = \begin{pmatrix} v_0^{-1} & 0 & \ldots & 0 \\ 0 & n^{-1}\left(x_{\cdot 1}(\mathcal{I}_{\cdot 1})^T x_{\cdot 1}(\mathcal{I}_{\cdot 1}) + vI_{K_1}\right) & & 0 \\ \vdots & & \ddots & \\ 0 & 0 & & n^{-1}\left(x_{\cdot q}(\mathcal{I}_{\cdot q})^T x_{\cdot q}(\mathcal{I}_{\cdot q}) + vI_{K_q}\right) \end{pmatrix}.$$

1.5 Simulated Annealing Algorithm

We give in this section the details of the Simulated Annealing algorithm we use. In presence of more than one functional covariate, the following algorithm is used to determine the estimate of each coefficient function one by one.

Let $\tilde{\Theta}_{K_0} = \bigotimes_{K=1}^{K_0}\left(K, \Theta_K\right)$ where Θ_K is the space of all $\theta = (\beta_1^*, \ldots, \beta_K^*, m_1, \ldots, m_K, \ell_1, \ldots, \ell_K)$ and let the function $C(d(\cdot)) = \left\| d(\cdot) - \hat{\beta}_{L^2}(\cdot) \right\|^2$.

Algorithm : *Simulated Annealing*

- Initialize: a deterministic decreasing schedule of temperature $(\tau_i)_{i=1,\ldots,N_{\mathrm{SANN}}}$, a value of K_0 and an initial vector $(K_{(0)}, \theta_{(0)}) \in \tilde{\Theta}_{K_0}$.

- Compute the function $\beta_{(0)}(t)$ from $(K_{(0)}, \theta_{(0)})$.
- Repeat for i from 1 to N_{SANN} :
  - Choose randomly a move from $(K_{(i-1)}, \theta_{(i-1)})$ to (K', θ') among :
    1. propose a new $\beta_k^{*\prime}$ for an arbitrary $k \leq K_{(i-1)}$,
    2. propose a new m'_k for an arbitrary $k \leq K_{(i-1)}$,
    3. propose a new ℓ'_k for an arbitrary $k \leq K_{(i-1)}$,
    4. propose to append a new interval $(\beta^{*\prime}, m', \ell')$ or
    5. propose to drop out an interval (β_k^*, m_k, ℓ_k) for an arbitrary $k \leq K_{(i-1)}$.
  - Compute the function $\beta'(t)$ from the proposal (K', θ').
  - Compute the acceptance ratio

$$\alpha = \min\left\{1, \exp\left(\frac{C(\beta'(\cdot)) - C\big(\beta_{(i)}(\cdot)\big)}{\tau_i}\right)\right\}.$$

  - Draw u from Unif$(0, 1)$.
  - If $u < \alpha$, $(K_{(i)}, \theta_{(i)}) = (K', \theta')$ (move accepted), else $(K_{(i)}, \theta_{(i)}) = (K_{(i-1)}, \theta_{(i-1)})$ (move rejected).
  - Compute the function $\beta_{(i)}(t)$ from $(K_{(i)}, \theta_{(i)})$.
- Return the iteration $(K_{(i)}, \theta_{(i)})$ minimizing the criteria $C(.)$.

For the schedule of temperature, we use by default a logarithmic schedule (see Bélisle, 1992), which is given for each iteration i by

$$\text{Te}/\log\left((i-1)+e\right), \tag{1}$$

where Te is a parameter to calibrate and corresponds to the initial temperature. The result of the Simulated Annealing algorithm is sensitive to the scale of Te and it is quite difficult to find an a priori suitable value. For example, if the initial temperature is too small, almost all the proposed moves are rejected during the algorithm. On the other hand, if it is too large, they are almost all accepted. So, we run the algorithm a few times and each time Te is determined with respect to the previous runs. For instance, if for a run the moves are always rejected or always accepted, the initial temperature for the next run is accordingly adjusted. Only 2 or 3 runs are necessary to find a suitable scale of Te.

2 Theoretical results

2.1 Proof of Theorem (1)

Without loss of generality we can assume that $\mathcal{T} = [0;1]$. We begin the proof with the following lemma whose simple proof is left to the reader.

Lemma S.2.1. *Set* $\psi^*(\gamma,\alpha) = \min\{\gamma(1-\alpha)\,;\,(1-\gamma)\alpha\}$ *for any* $\alpha,\gamma \in [0;1]$. *We have*

$$\psi^*(\gamma,\alpha) = \begin{cases} \gamma(1-\alpha) & \text{if } \gamma \le \alpha, \\ (1-\gamma)\alpha & \text{if } \gamma \ge \alpha. \end{cases}$$

Remember that the posterior loss we optimise is given in (12), where S is any Borel subset of $\mathcal{T} = [0;1]$. Using Fubini's theorem (for non-negative functions) and the definition of $\alpha(t|\mathcal{D})$ given in (10), we have

$$\begin{aligned} \int_{\Theta_K} L_\gamma(S,S_\theta)\pi_K(\theta|\mathcal{D})\mathrm{d}\theta &= \gamma \int_0^1 \int_{\Theta_K} \mathbb{1}\{t \in S \setminus S_\theta\}\pi_K(\theta|\mathcal{D})\mathrm{d}\theta\mathrm{d}t \\ &\quad + (1-\gamma)\int_0^1 \int_{\Theta_K} \mathbb{1}\{t \in S_\theta \setminus S\}\pi_K(\theta|\mathcal{D})\mathrm{d}\theta\mathrm{d}t \\ &= \int_0^1 \psi_S\big(t,\gamma,\alpha(t|\mathcal{D})\big)\mathrm{d}t \end{aligned} \tag{S.2}$$

where, for all $\alpha \in [0;1]$ we have set

$$\psi_S(t,\gamma,\alpha) = \mathbb{1}\{t \in S\}\gamma\big(1-\alpha\big) + \mathbb{1}\{t \notin S\}(1-\gamma)\alpha.$$

Now, whatever the set S, $\psi_S(t,\gamma,\alpha) \ge \psi^*(\gamma,\alpha)$. Reporting this bound in (S.2) yields

$$\int_{\Theta_K} L_\gamma(S,S_\theta)\pi_K(\theta|\mathcal{D})\mathrm{d}\theta \ge \int_0^1 \psi^*\big(\gamma,\alpha(t|\mathcal{D})\big)\mathrm{d}t$$

whatever the Borel set S. Moreover, this inequality is an equality if and only if the Borel set S is chosen so that, for almost all $t \in [0;1]$, $\psi_S\big(t,\gamma,\alpha(t|\mathcal{D})\big) = \psi^*\big(\gamma,\alpha(t|\mathcal{D})\big)$. Using Lemma S.2.1, the last condition is equivalent to saying that for almost all $t \in [0;1]$, either $\alpha(t|\mathcal{D}) = \gamma$ or $\big(t \in S \iff \gamma \le \alpha(t|\mathcal{D})\big)$. This concludes the proof of Theorem (1). □

2.2 Proof of Proposition (2)

Obviously, $\widehat{\beta}_{L^2}(\cdot)$ minimizes

$$\int\int_{\mathcal{T}} \big(\beta_\theta(t) - d(t)\big)^2\,\mathrm{d}t\;\pi_K(\theta|\mathcal{D})\mathrm{d}\theta = \int_{\mathcal{T}}\int \big(\beta_\theta(t) - d(t)\big)^2\,\pi_K(\theta|\mathcal{D})\mathrm{d}\theta\;\mathrm{d}t$$

because it does optimize $\int \big(\beta_\theta(t) - d(t)\big)^2\,\pi_K(\theta|\mathcal{D})\mathrm{d}\theta$ for all $t \in \mathcal{T}$. It remains to show that $\widehat{\beta}_{L^2}(\cdot) \in L^2(\mathcal{T})$. We have

$$\|\widehat{\beta}_{L^2}(\cdot)\|^2 = \int_{\mathcal{T}} \left(\int \beta_\theta(t)\pi_K(\theta|\mathcal{D})\mathrm{d}\theta\right)^2 \mathrm{d}t$$

$$
\begin{aligned}
&= \iint \int_{\mathcal{T}} \beta_\theta(t)\beta_{\theta'}(t)\mathrm{d}t \ \pi_K(\theta|\mathcal{D})\pi_K(\theta'|\mathcal{D})\mathrm{d}\theta\mathrm{d}\theta' \\
&\leq \iint \|\beta_\theta(\cdot)\|\|\beta_{\theta'}(\cdot)\| \ \pi_K(\theta|\mathcal{D})\pi_K(\theta'|\mathcal{D})\mathrm{d}\theta\mathrm{d}\theta' \quad \text{with Cauchy-Schwarz inequality} \\
&\leq \left(\int \|\beta_\theta(\cdot)\|\pi_K(\theta|\mathcal{D})\mathrm{d}\theta \right)^2
\end{aligned}
$$

And the last integral is finite because of the assumption. Hence $\widehat{\beta}_{L^2}(\cdot)$ is in $L^2(\mathcal{T})$. □

2.3 Proof of Proposition (3)

First, the norm $\|d(\cdot) - \widehat{\beta}_{L^2}(\cdot)\|$ is non negative, hence the set

$$
\left\{ \|d(\cdot) - \widehat{\beta}_{L^2}(\cdot)\|, \ d(\cdot) \in \mathcal{E}^{\varepsilon}_{K_0} \right\}
$$

admits an infimum. Let m denote this infimum. We have to prove that m is actually a minimum of the above set, namely that there exists a function $d(\cdot) \in \mathcal{E}^{\varepsilon}_{K_0}$ so that $m = \|d(\cdot) - \widehat{\beta}_{L^2}(\cdot)\|$.

To this end, we introduce a minimizing sequence $\{d_n(\cdot)\}$ and we will show that one of its subsequences admits a limit within $\mathcal{E}^{\varepsilon}_{K_0}$. Let $d_n(\cdot)$ be so that

$$
m = \inf \left\{ \|d(\cdot) - \widehat{\beta}_{L^2}(\cdot)\|, \ d(\cdot) \in \mathcal{E}^{\varepsilon}_{K_0} \right\} \leq \|d_n(\cdot) - \widehat{\beta}_{L^2}(\cdot)\| \leq m + 2^{-n}. \tag{S.4}
$$

The step function $d_n(\cdot)$ can be written as

$$
d_n(t) = \sum_{k=1}^{L} \alpha_{k,n} \mathbb{1}\{t \in (a_{k,n}, \ b_{k,n})\}
$$

where the $(a_{k,n}, \ b_{k,n})$, $k = 1, \ldots, L$ are non-overlapping intervals. Note that their number L does not depend on n because all $d_n(\cdot)$ lie in $\mathcal{E}_{K_0}$ for some fixed value of K_0, and we can always choose $L = 2K_0 - 1$. Moreover, because $d_n(t)$ is in $\mathcal{F}^{\varepsilon}$, we can assume that

$$
b_{k,n} - a_{k,n} \geq \varepsilon, \quad \text{for all } k, n. \tag{S.5}
$$

Now the sequence $\{a_{1,n}\}_n$ has its elements in the compact interval $\mathcal{T}$ hence we extract a subsequence (still denoted $\{a_{1,n}\}_n$) which converges an element $a_{1,\infty}$ of $\mathcal{T}$. Likewise, by extracting subsequences $2L$ times, we can assume that all sequences $\{a_{1,n}\}_n, \ldots$, $\{a_{L,n}\}_n$, $\{b_{1,n}\}_n$, $\ldots$, $\{b_{L,n}\}_n$ are convergent, and that

$$
a_{k,\infty} = \lim_{n\to\infty} a_{k,n}, \quad b_{k,\infty} = \lim_{n\to\infty} b_{k,n}, \quad \text{and} \quad b_{k,\infty} - a_{k,\infty} \geq \varepsilon, \quad k = 1, \ldots, L
$$

where the last inequalities come from (S.5).

The sequence $d_n(\cdot)$ is bounded (in L^2-norm):

$$\|d_n(\cdot)\| \leq \|\widehat{\beta}_{L^2}(\cdot)\| + \|d_n(\cdot) - \widehat{\beta}_{L^2}(\cdot)\| \leq R + \sqrt{m+1}$$

with (S.4), where $R = \|\widehat{\beta}_{L^2}(\cdot)\|$. Moreover

$$\|d_n(\cdot)\|^2 = \sum_{k=1}^{L} \alpha_{k,n}^2 \big(b_{k,n} - a_{k,n}\big) \geq \varepsilon \sum_{k=1}^{L} \alpha_{k,n}^2.$$

Hence, each sequence $\{\alpha_{1,n}\}_n$, ..., $\{\alpha_{L,n}\}_n$ is bounded. Thus, by further extracting subsubsequences, we can assume that, for $k = 1, \ldots, L$,

$$\lim_{n\to\infty} \alpha_{k,n} = \alpha_{k,\infty}.$$

Finally, by setting

$$d_\infty(\cdot) = \sum_{k=1}^{L} \alpha_{k,\infty} \mathbb{1}\{t \in (a_{k,\infty},\ b_{k,\infty})\}$$

we can easily prove that $d_n(\cdot)$ tends to $d_\infty(\cdot)$ in L^2-norm and that $d_\infty(\cdot) \in \mathcal{E}_{K_0}^{\varepsilon}$. And, with (S.4)

$$m = \|d_\infty(\cdot) - \widehat{\beta}_{L^2}(\cdot)\|$$

which concludes the proof. □

2.4 Topological Properties of $\mathcal{E}_K$

Proposition S.2.2. *Let $K \geq 1$.*

(i) The convex hull of $\mathcal{E}_K$ is $\mathcal{E}$.

(ii) Under the $L^2(\mathcal{T})$-topology, the closure of $\mathcal{E}$ is $L^2(\mathcal{T})$.

Proof. The result of *(ii)* is rather classical, see, e.g., Rudin (1986). The convex hull of $\mathcal{E}_K$ includes any step function. Indeed, any step function can be written as a convex combination of simple $a\mathbf{1}\{t \in I\}$'s which all belongs to $\mathcal{E}_K$. Moreover, $\mathcal{E}$ is convex because it is a linear space. Hence claim *(i)* is proven. □

For a given K, the set of functions $\mathcal{E}_K$ is not suitable to define a projection of $\hat{\beta}_{L^2}(\cdot)$. Indeed, let $\{d_n(\cdot)\}$ be a minimizing sequence of the set $\big\{\|d(\cdot) - \hat{\beta}_{L^2}(\cdot)\|,\ d(\cdot) \in \mathcal{E}_K(\cdot)\big\}$, so

$$m = \inf\Big\{\|d(\cdot) - \widehat{\beta}_{L^2}(\cdot)\|,\ d(\cdot) \in \mathcal{E}_K\Big\} \leq \|d_n(\cdot) - \widehat{\beta}_{L^2}(\cdot)\| \leq m + 2^{-n}.$$

Knowing that $\hat{\beta}_{L^2}(\cdot)$ and $d_n(\cdot)$ belong to L^2 for all n, we have

$$d_n(.) \in \mathcal{E}_K \cap \mathcal{B}_{L^2}(R + m + 1), \qquad \text{for all } n,$$

where $\mathcal{B}_{L^2}(r)$ is the L^2-ball of radius r around the origin. Note that $\mathcal{E}_K \cap \mathcal{B}_{L^2}(R+m+1)$ is not a compact set, for example consider $d_n(t) = \sqrt{n}\,\mathbf{1}\{t \in [0, \frac{1}{n}]\}$. Hence it is not possible to extract a subsequence of $\{d_n(\cdot)\}$ which converges to a $d_\infty(\cdot) \in \mathcal{E}_K$ so that $\|d(\cdot) - \hat{\beta}_{L^2}(\cdot)\| = m$.

3 Application

3.1 Comparison of Bliss and competitors

In order to compare the methods for the estimation of the coefficient function, we use the L^2-loss, namely

$$\int_0^1 (\widehat{\beta}(t) - \beta_0(t))^2 \mathrm{d}t \tag{9}$$

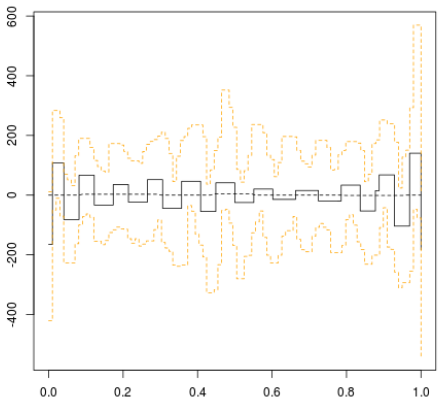
where $\widehat{\beta}(t)$ is an estimate we compare to the true coefficient function $\beta_0(t)$. Table 2 of the paper shows the results of Bliss and its competitors on these simulated datasets. It appears that the numerical results of the three methods have the same order of magnitude although the three methods may have different accuracy, depending on the shape of the coefficient function that generated the dataset.

Regarding Fused Lasso, we can see in Table 2 that its accuracy worsens when the problem is not sparse, that is to say when the true function is the "smooth" function (the red curve of Figure 2 of the paper). Next, we observe that Flirti is very sensitive. Its numerical results can sometimes be quite accurate, but sometimes the L^2-error can blow up (to exceed 100) because the method did not manage to tune its parameters. Concerning the BFDA method, we note that the estimate becomes irrelevant when the autocorrelation increases (i.e. ζ decreases). In particular, for the Step function and Smooth shapes, the L^2-errors can exceed 10^3. The L^2-estimate defined in Proposition 2 frequently overperforms the other methods. This first conclusion is not surprising because the L^2-estimate has been defined to optimize the L^2-loss integrated over the posterior distribution.

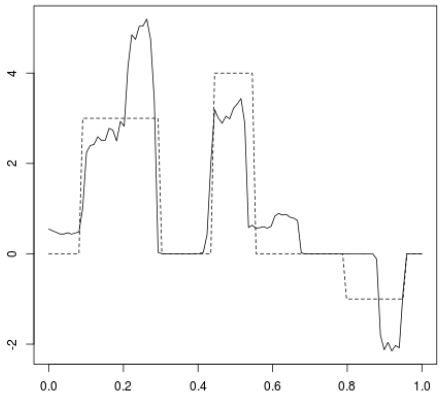
Even in situations where the true function is stepwise, the stepwise Bliss estimate of Proposition 3 is less accurate than the L^2-estimate, except for two examples (datasets 6 and 9). Nevertheless we do insist that the stepwise Bliss estimate was built to provide a trade off between accuracy regarding the support estimate and accuracy regarding the coefficient function estimate. Thus the stepwise estimate is a balance between support estimate and coefficient function estimate that can help the statistician who can then obtain an interpretation of the underlying phenomena that generated the data. In other words, the stepwise Bliss estimate is not the best either at estimating the support or at approximating the coefficient function, but provides a tradeoff.

To show more detailed results we have presented the estimate of the coefficient function in three cases.

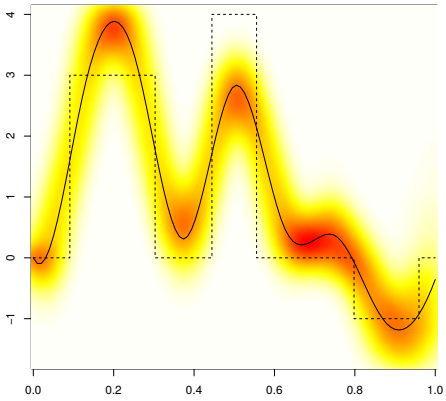
- Figure 1 displays the numerical results on Dataset 4 (medium level of signal, low level of autocorrelation with the covariate). As can be expected when the true coefficient is a stepwise function, the stepwise Bliss estimate behaves nicely. The representation of the marginals of the posterior distribution with a heat map shows the confidence we can have in the Bayes estimate of the coefficient function. The smooth L^2-estimate nicely follows the regions of high posterior density. Here, the stepwise estimate clearly highlights two time periods (the first two intervals of the true support) and the sign of the coefficient function on these intervals.



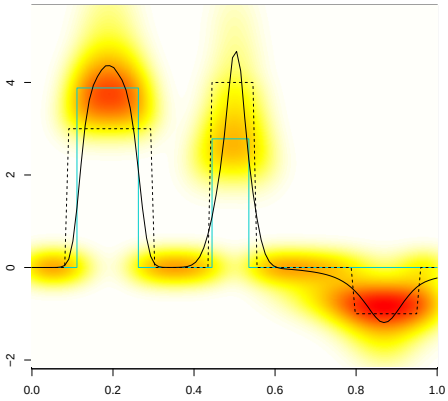
Flirti



Fused Lasso

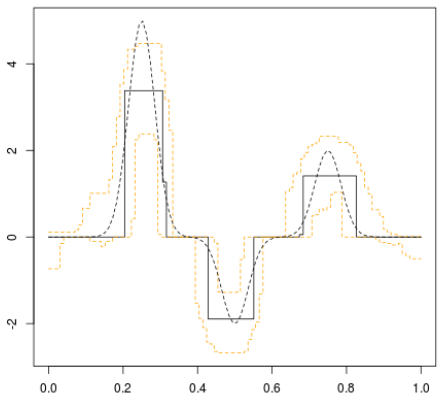


BFDA

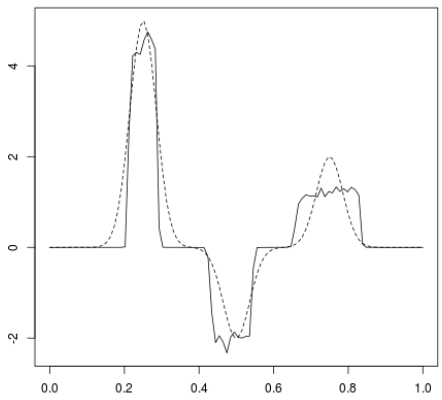


Bliss

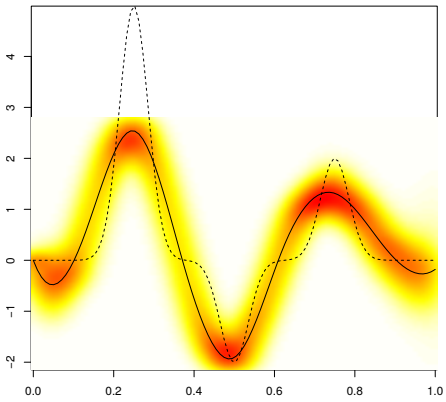
Figure 1: Estimates of the coefficient function on Dataset 4 ($r = 3$, $\zeta = 1$). *For each plot, the black dotted line is the true coefficient function (Step function, in this case) and the solid black lines are the estimates of each method. Concerning the Flirti plot, the orange dotted lines correspond to the confidence bands of the estimate. For the Bayesian methods (BFDA and Bliss) a representation of the marginal posterior distributions of* $\beta(t)$ *are represented using heat maps, as described in Section 1.2. Red (resp. white) colour is used to represent high (resp. low) posterior densities. For the Bliss plot, the solid black line is the* L^2*-estimate and the light blue line is the stepwise Bliss estimate.*



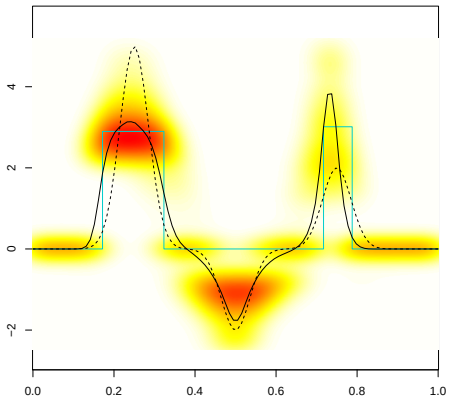
Flirti



Fused Lasso

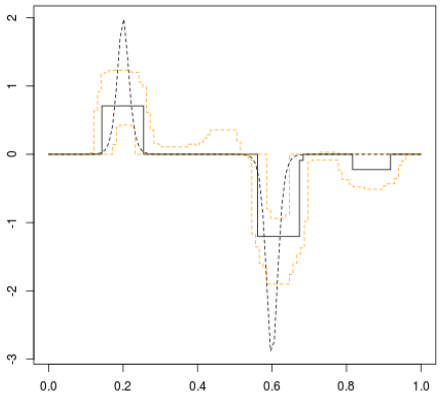


BFDA

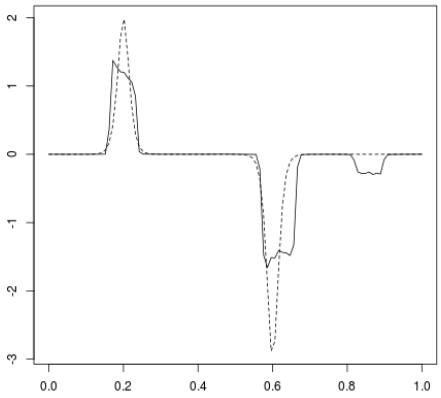


Bliss

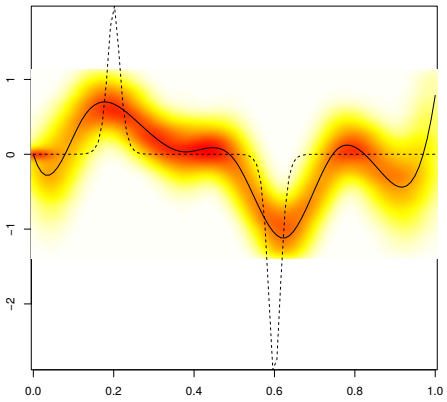
Figure 2: Estimates of the coefficient function on Dataset 13 ($r = 3$, $\zeta = 1$)) *For each plot, the black dotted line is the true coefficient function (Smooth, in this case) and the solid black lines are the estimates of each method. Concerning the Flirti plot, the orange dotted lines correspond to the confidence bands of the estimate. For the Bayesian methods (BFDA and Bliss) a representation of the marginal posterior distributions of* $\beta(t)$ *are represented using heat maps, as described in Section 1.2. Red (resp. white) colour is used to represent high (resp. low) posterior densities. For the Bliss plot, the solid black line is the* L^2*-estimate and the light blue line is the stepwise Bliss estimate.*



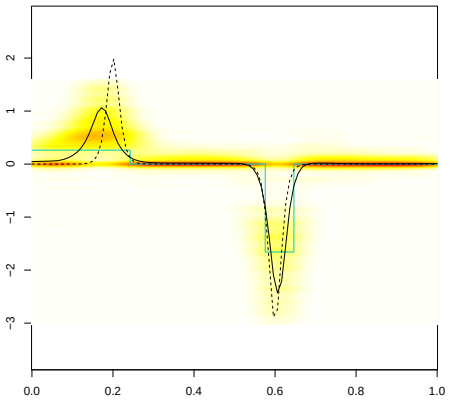
Flirti



Fused Lasso



BFDA



Bliss

Figure 3: Estimates of the coefficient function on Dataset 25 ($r = 1$, $\zeta = 1$)) *For each plot, the black dotted line is the true coefficient function (Spiky, in this case) and the solid black lines are the estimates of each method. Concerning the Flirti plot, the orange dotted lines correspond to the confidence bands of the estimate. For the Bayesian methods (BFDA and Bliss) a representation of the marginal posterior distributions of* $\beta(t)$ *are represented using heat maps, as described in Section 1.2. Red (resp. white) colour is used to represent high (resp. low) posterior densities. For the Bliss plot, the solid black line is the* L^2*-estimate and the light blue line is the stepwise Bliss estimate.*

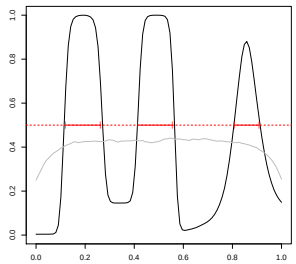
We can compare our proposal with its competitors. Flirti did not manage to tune its own parameters, and the Flirti estimate is irrelevant. Fused Lasso on a discretized version of the functional covariate provides a relatively nice estimate of the coefficient function. BFDA is not that bad, although the estimate is clearly too smooth to match the true coefficient function. As for the Bliss method, we give a representation of the marginals of the posterior distribution based on the MCMC sample provided by the BFDA method. In this case, the true coefficient function is not in high posterior density regions, especially for t for which it equals 0.

- Figure 2 displays the numerical results on Dataset 13 (medium level of signal, low level of autocorrelation with the covariate). In this example, the true coefficient is not stepwise, but smooth, and is around zero on small time periods. Flirti and Fused Lasso performed nicely. Their estimates are exactly 0 on intervals where the true coefficient function is 0. Hence they clearly highlight the intervals. The BFDA estimate decently fits the true coefficient function but it is less accurate when the coefficient function is around 0. The L^2-estimate performed as well as the BFDA estimate, except it is around 0 for t in $[0, 0.1]$ and $[0.9, 1]$. The stepwise Bliss estimate performed relatively poorly because it does not detect a negative interval around $t = 0.5$. With respect to Proposition 3, it is a projection of the L^2-estimate (which is clearly different to 0 around $t = 0.5$), hence we could expect that the stepwise Bliss estimate is negative for t around 0.5. Therefore, in this case, the simulated annealing algorithm does not correctly converge.

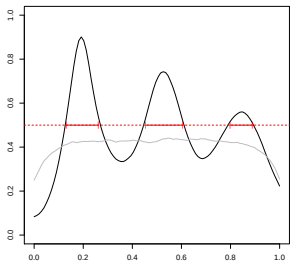
- Figure 3 displays the numerical results on Dataset 25 (low level of signal, and low level of autocorrelation within the covariate). In this example, the true coefficient is not stepwise, but smooth, and is around zero on large time periods. The L^2-estimate of Proposition 2 matches approximately the true coefficient function. The stepwise Bliss estimate is a little bit poorer (maybe because of the difficult calibration of the simulated annealing algorithm). When comparing these results with other estimates on this dataset, we see that Flirti and Fused Lasso performed decently also, even if they both highlight a third time period (around $t = 0.85$) where they infer a negative coefficient function instead of 0. In this case, Flirti has managed to tune its own parameters in a relevant way. The confidence bands of Flirti are therefore reliable, but we stress here that they are relatively wide around periods where the Flirti estimate is null and does not reflect high confidence in any support estimate based on Flirti. Finally, the comments on BFDA are the same as Dataset 4, the BFDA estimate has clearly been too smoothed to match the true coefficient function, especially for t for which it is around 0.

3.2 All the graphical results for a single functional covariate

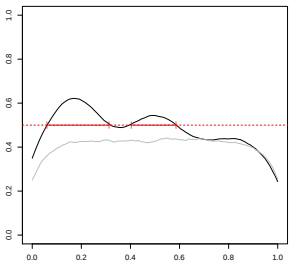
Below, Figure 4 (resp. 5) shows the estimates of the support (resp. de coefficient function) when the simulated $\beta(\cdot)$ is the step function described in Section 3.1 (see Figure 2 of the paper). Figures 6 and 7) show the estimates when the simulated $\beta(\cdot)$ is the Smooth function. Next, Figures 8 and 9 show the estimates when the simulated $\beta(\cdot)$ is the Spiky function.



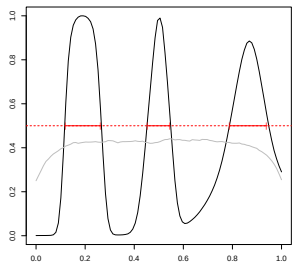
Dataset 1 ($r = 5$, $\zeta = 1$)



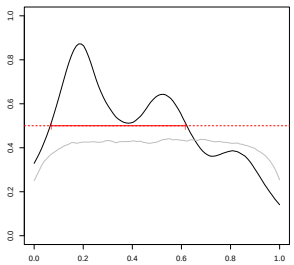
Dataset 2 ($r = 5$, $\zeta = 1/3$)



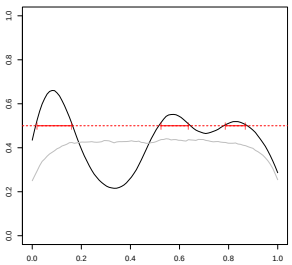
Dataset 3 ($r = 5$, $\zeta = 1/5$)



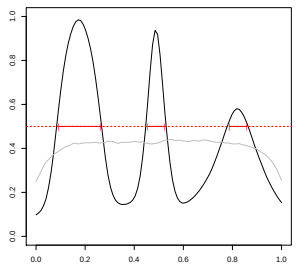
Dataset 4 ($r = 3$, $\zeta = 1$)



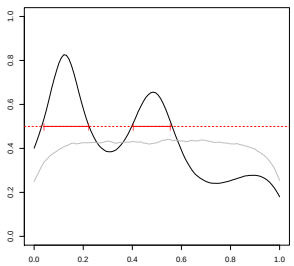
Dataset 5 ($r = 3$, $\zeta = 1/3$)



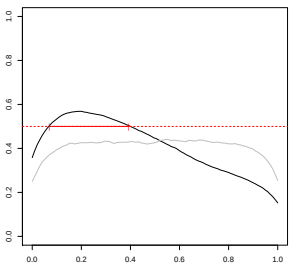
Dataset 6 ($r = 3$, $\zeta = 1/5$)



Dataset 7 ($r = 1$, $\zeta = 1$)

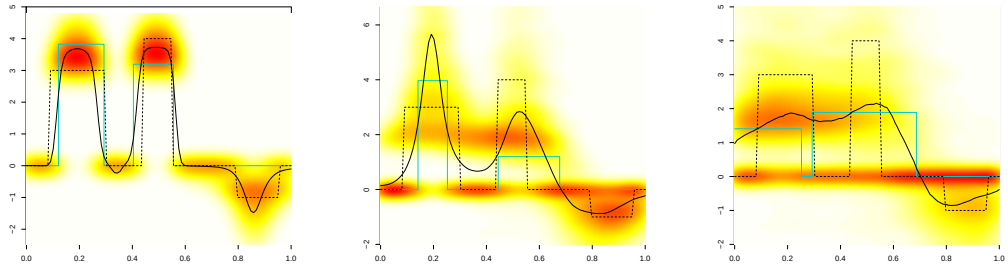


Dataset 8 ($r = 1$, $\zeta = 1/3$)

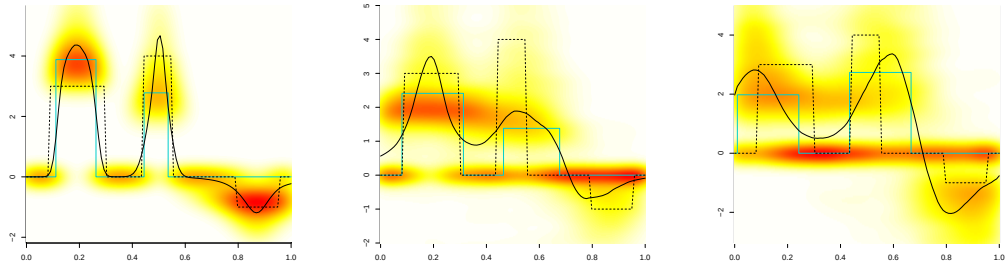


Dataset 9 ($r = 1$, $\zeta = 1/5$)

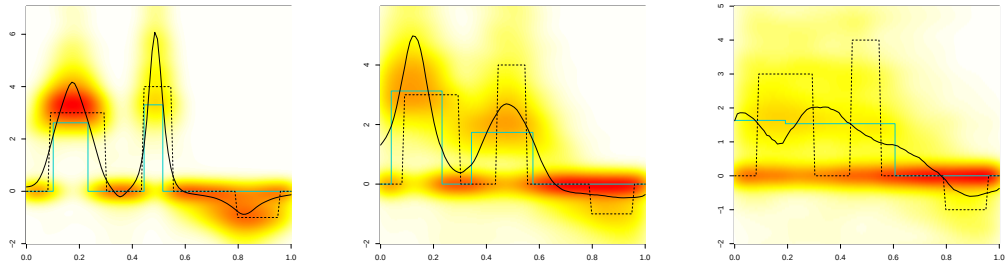
Figure 4: Prior (in gray) and posterior (in black) probabilities of being in the support, computed on Datasets 1 to 9. *Bayes estimate of support using Theorem 1 with* $\gamma = 1/2$ *are given in red.*



Dataset 1 ($r = 5$, $\zeta = 1$) Dataset 2 ($r = 5$, $\zeta = 1/3$) Dataset 3 ($r = 5$, $\zeta = 1/5$)

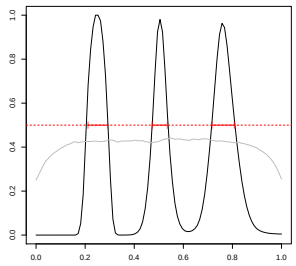


Dataset 4 ($r = 3$, $\zeta = 1$) Dataset 5 ($r = 3$, $\zeta = 1/3$) Dataset 6 ($r = 3$, $\zeta = 1/5$)

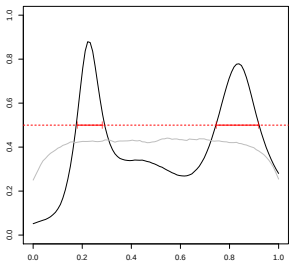


Dataset 7 ($r = 1$, $\zeta = 1$) Dataset 8 ($r = 1$, $\zeta = 1/3$) Dataset 9 ($r = 1$, $\zeta = 1/5$)

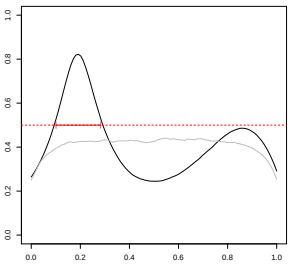
Figure 5: Estimates of the coefficient functions on the Datasets 1 to 9. *For each plot, the black dotted line is the true coefficient function (Step function, in this case), the solid black line is the L^2-estimate and the light blue line is the stepwise Bliss estimate. The heat maps represents the marginal posterior distributions of $\beta(t)$, as described in Section 1.2. Red (resp. white) colour is used to represent high (resp. low) posterior densities.*



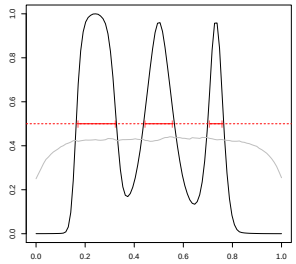
Dataset 10 ($r = 5$, $\zeta = 1$)



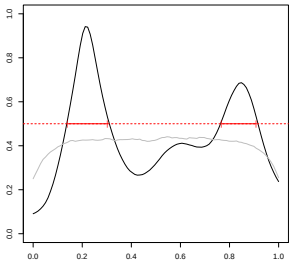
Dataset 11 ($r = 5$, $\zeta = 1/3$)



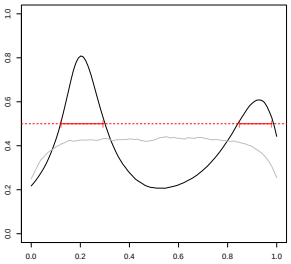
Dataset 12 ($r = 5$, $\zeta = 1/5$)



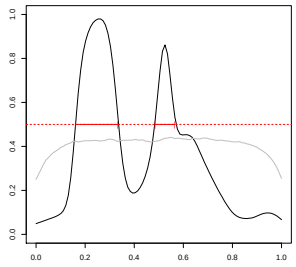
Dataset 13 ($r = 3$, $\zeta = 1$)



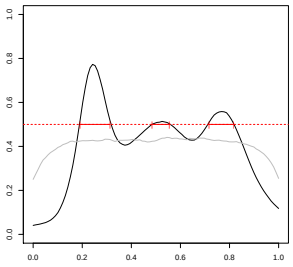
Dataset 14 ($r = 3$, $\zeta = 1/3$)



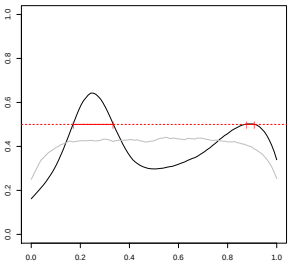
Dataset 15 ($r = 3$, $\zeta = 1/5$)



Dataset 16 ($r = 1$, $\zeta = 1$)

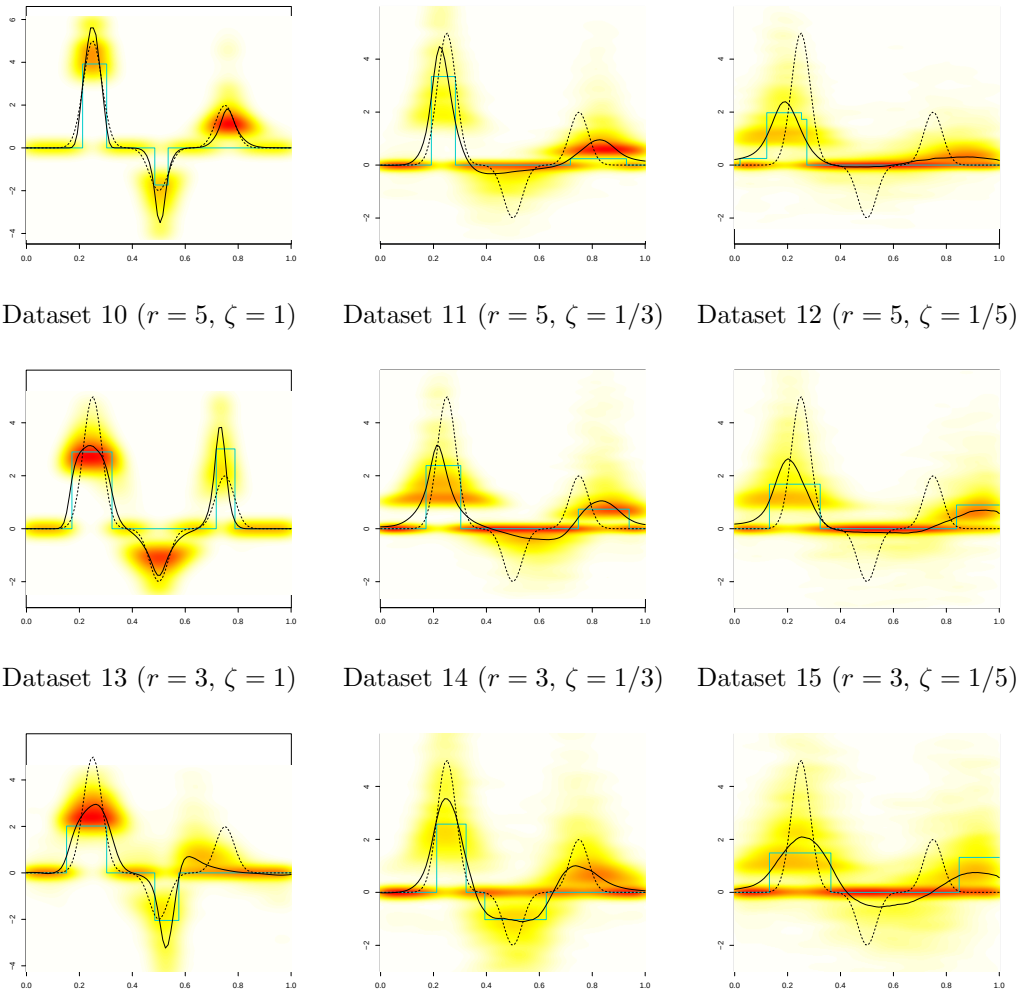


Dataset 17 ($r = 1$, $\zeta = 1/3$)



Dataset 18 ($r = 1$, $\zeta = 1/5$)

Figure 6: Prior (in gray) and posterior (in black) probabilities of being in the support, computed on Datasets 10 to 18. *Bayes estimate of support using Theorem 1 with* $\gamma = 1/2$ *are given in red.*

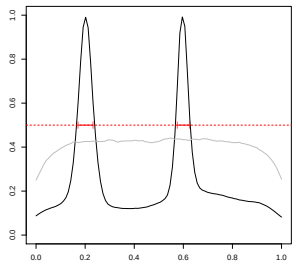


Dataset 10 ($r = 5$, $\zeta = 1$) Dataset 11 ($r = 5$, $\zeta = 1/3$) Dataset 12 ($r = 5$, $\zeta = 1/5$)

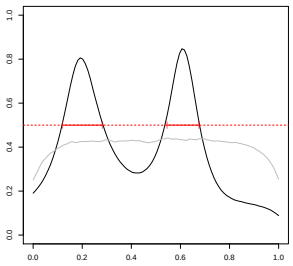
Dataset 13 ($r = 3$, $\zeta = 1$) Dataset 14 ($r = 3$, $\zeta = 1/3$) Dataset 15 ($r = 3$, $\zeta = 1/5$)

Dataset 16 ($r = 1$, $\zeta = 1$) Dataset 17 ($r = 1$, $\zeta = 1/3$) Dataset 18 ($r = 1$, $\zeta = 1/5$)

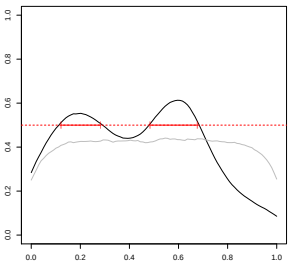
Figure 7: Estimates of the coefficient functions on the Datasets 10 to 18. *For each plot, the black dotted line is the true coefficient function (Smooth, in this case), the solid black line is the L^2-estimate and the light blue line is the stepwise Bliss estimate. The heat maps represents the marginal posterior distributions of $\beta(t)$, as described in Section 1.2. Red (resp. white) colour is used to represent high (resp. low) posterior densities.*



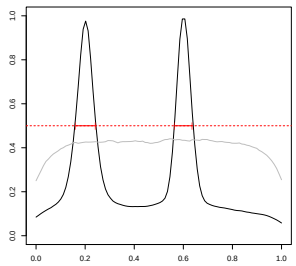
Dataset 19 ($r = 5$, $\zeta = 1$)



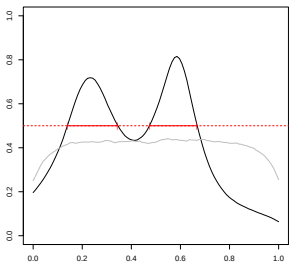
Dataset 20 ($r = 5$, $\zeta = 1/3$)



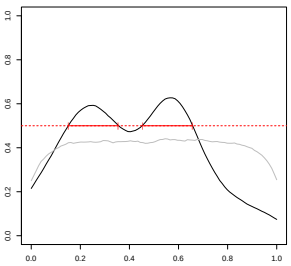
Dataset 21 ($r = 5$, $\zeta = 1/5$)



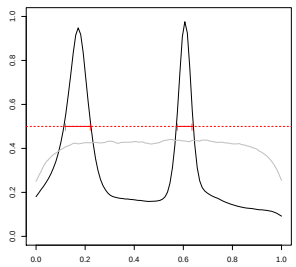
Dataset 22 ($r = 3$, $\zeta = 1$)



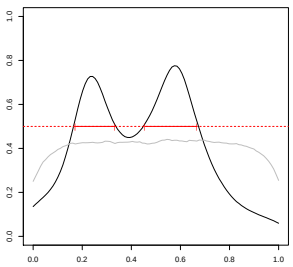
Dataset 23 ($r = 3$, $\zeta = 1/3$)



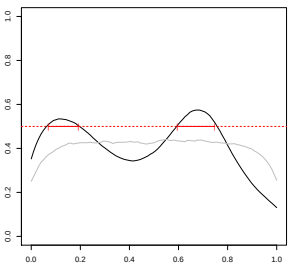
Dataset 24 ($r = 3$, $\zeta = 1/5$)



Dataset 25 ($r = 1$, $\zeta = 1$)

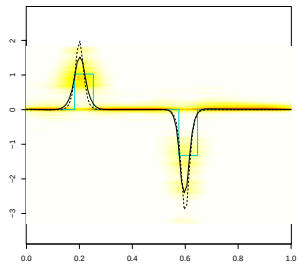


Dataset 26 ($r = 1$, $\zeta = 1/3$)

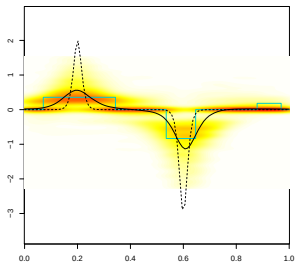


Dataset 27 ($r = 1$, $\zeta = 1/5$)

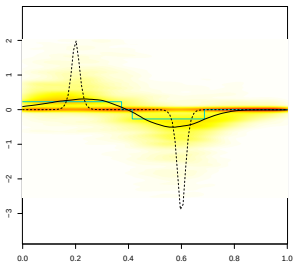
Figure 8: Prior (in gray) and posterior (in black) probabilities of being in the support, computed on Datasets 19 to 27. *Bayes estimate of support using Theorem 1 with* $\gamma = 1/2$ *are given in red.*



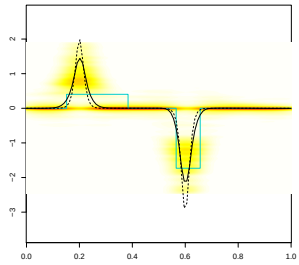
Dataset 19 ($r = 5$, $\zeta = 1$)



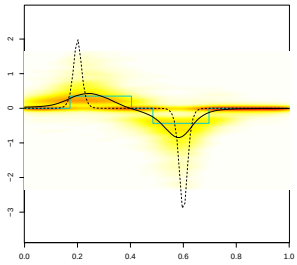
Dataset 20 ($r = 5$, $\zeta = 1/3$)



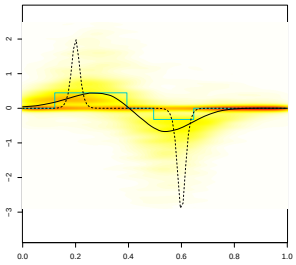
Dataset 21 ($r = 5$, $\zeta = 1/5$)



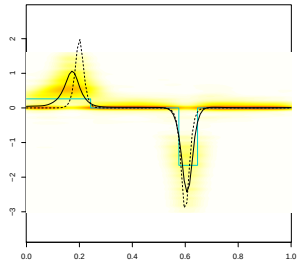
Dataset 22 ($r = 3$, $\zeta = 1$)



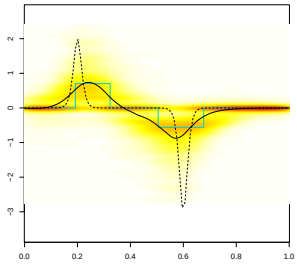
Dataset 23 ($r = 3$, $\zeta = 1/3$)



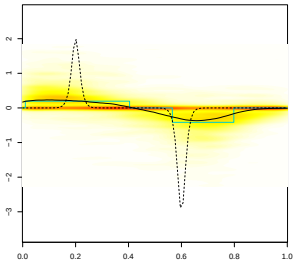
Dataset 24 ($r = 3$, $\zeta = 1/5$)



Dataset 25 ($r = 1$, $\zeta = 1$)



Dataset 26 ($r = 1$, $\zeta = 1/3$)



Dataset 27 ($r = 1$, $\zeta = 1/5$)

Figure 9: Estimates of the coefficient functions on the Datasets 19 to 27. *For each plot, the black dotted line is the true coefficient function (Spiky, in this case), the solid black line is the L^2-estimate and the light blue line is the stepwise Bliss estimate. The heat maps represents the marginal posterior distributions of $\beta(t)$, as described in Section 1.2. Red (resp. white) colour is used to represent high (resp. low) posterior densities.*

3.3 Model Choice by using BIC

Below, we illustrate the model choice procedure in order to fix the hyperparameter K. We apply the procedure on different simulated datasets and in Section 3.5 it is applied on the truffle dataset.

We simulate four kind of datasets to evaluate the performance of BIC in different situations.

First, we show the results of BIC when the true interval's number varies. Therefore, we simulate datasets with the coefficient functions given in Figure 10 for which the true K varies from 1 to 5. The values of BIC are given in Figure 11. When the true K is 1 or 2, BIC selects the true model. Otherwise, when the true K is greater than 3, it underestimates K which can be due to the small sample size ($n = 100$).

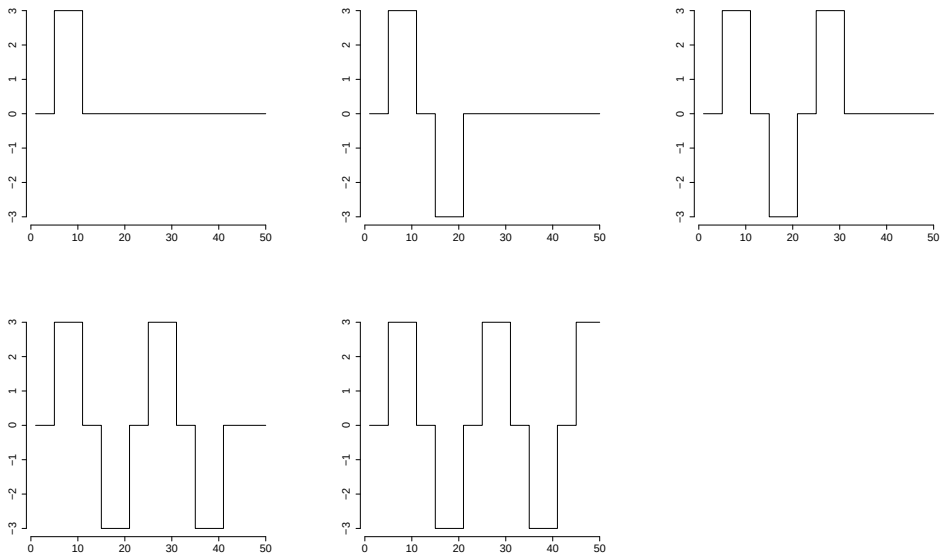


Figure 10: Coefficient functions used to simulate datasets with different number of intervals.

Second, we illustrate the variation of BIC when the autocorrelation of the curves $x_i(\cdot)$ increases, i.e. ζ decreases (see Section 3.1). We simulate datasets with $\zeta = 1, 0.8, 0.6, 0.4$ and 0.2, with $n = 100$ and the true coefficient function is the third plot given in Figure 10. Plot (a) Figure 12 shows the values of BIC for different values of ζ. When the autocorrelation is high (blue and light blue lines), BIC trends to underestimate K, since in this example the true K is 3. The results

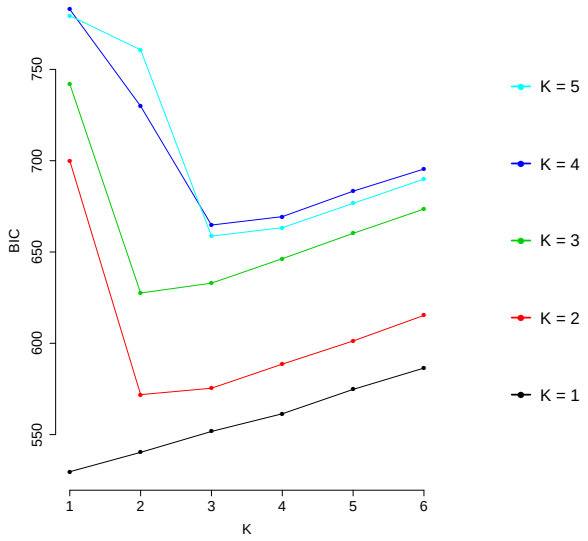


Figure 11: The values of BIC for different values of the number of intervals of the true coefficient function.

are the same if the true coefficient function is the Smooth function described in Section 3.1 of the main paper, see Plot (b) Figure 12.

Third, we illustrate the variation of BIC when the sample size n increases. Figure 13 shows the values of BIC for $n = 50, 100, 200$ and 300. As expected, when the sample size increases, BIC is more accurate for selecting the true model $K = 3$.

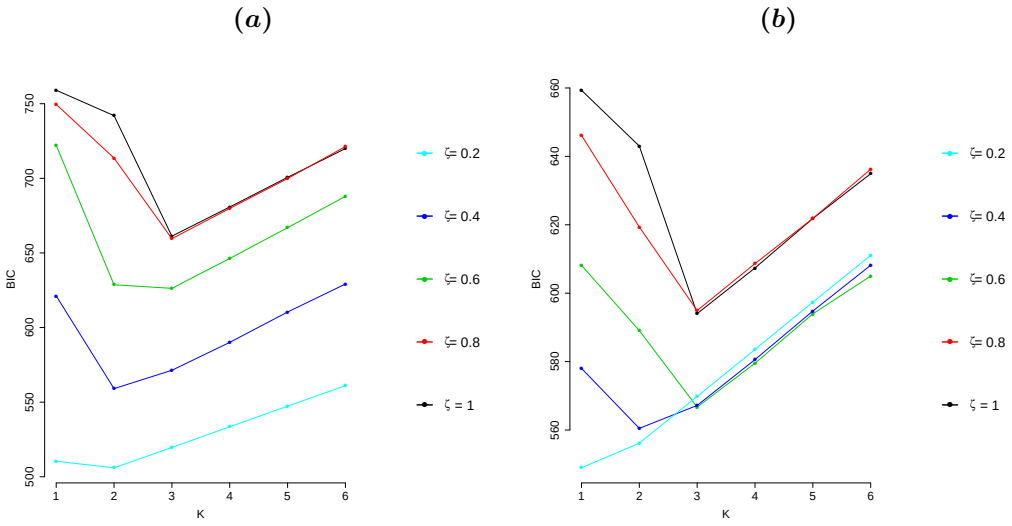


Figure 12: The values of BIC for different levels of autocorrelation ζ. *The left plot concerns datasets simulated for which the coefficient function is a step function with three intervals (see Figure 10). The right plot concerns simulated datasets for which the coefficient function is the Smooth function described in Section 3.1.*

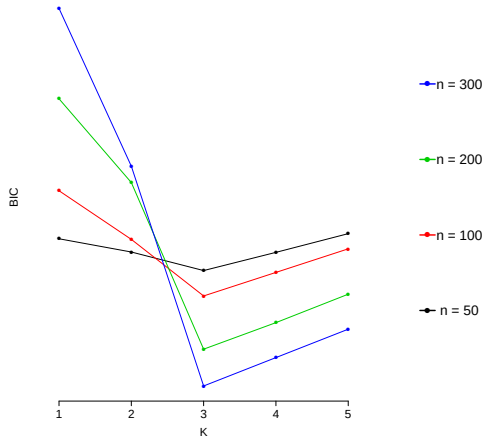


Figure 13: The values of BIC for different values of n.

3.4 Simulation Study for Two Functional Covariates

Performances regarding the coefficient function

Figure 14 shows the estimators of $\beta_1(\cdot)$ and $\beta_2(\cdot)$ for $c = 0$ and for $c = 0.9$. We notice that the estimates behave poorly in the presence of correlation between the covariates ($c = 0.9$), as in a classical multiple linear regression model with scalar covariates. The investigation of the Plots (a), (b) and (c), (d) Figure 14 leads us to almost the same remarks as in the previous paragraph. When the cross-covariance between the covariates increases, the estimates become less accurate. We notice additionnaly that the posterior distributions are flatter for $c = 0.9$ than for $c = 0$, hence the estimates have a higher variability when there is an important cross-covariance between the covariates.

Graphical results for two functional covariates

In Section 3.4 of the main paper, we present the estimates when there are two functional covariates and for different levels of cross-correlation. In the main paper, we show the estimates when $c = 0$ and $c = 0.9$ (low cross-correlation and high cross-correlation). Below, we show the estimates when $c = 0.3$ or 0.6 with Figures 15 and 16.

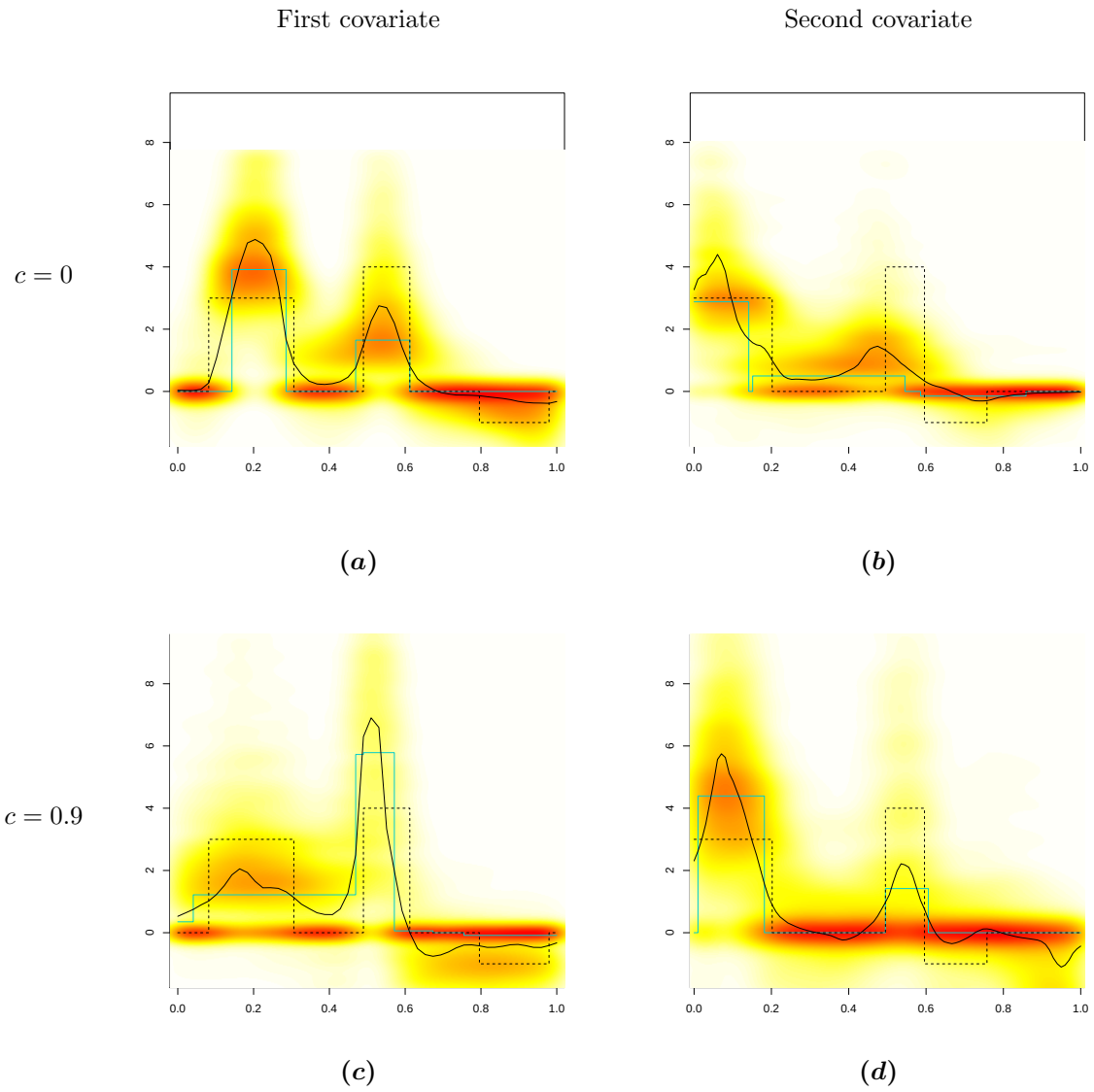
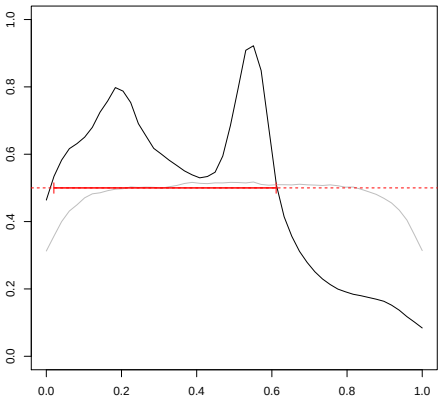
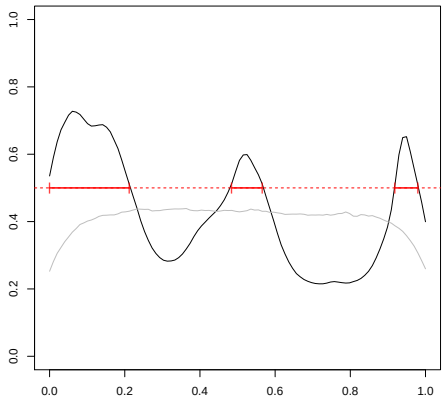


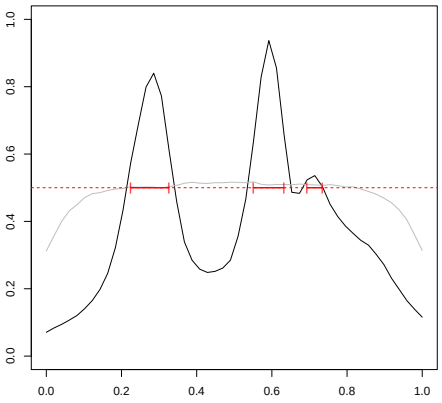
Figure 14: Estimates of the coefficient functions for $c = 0$ and $c = 0.9$. *For each plot, the black dotted line is the true coefficient function, the solid black line is the* L^2*-estimate and the light blue line is the stepwise Bliss estimate. The marginal posterior distributions of* $\beta_\theta(t)$ *are represented by using heat maps, as described in Section 1.2. Red (resp. white) colour is used to represent high (resp. low) posterior densities.*



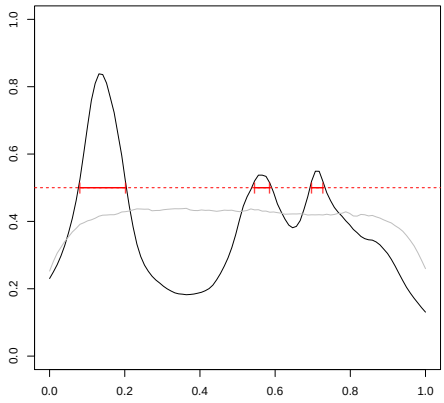
First covariate ($c = 0.3$)



Second covariate ($c = 0.3$)

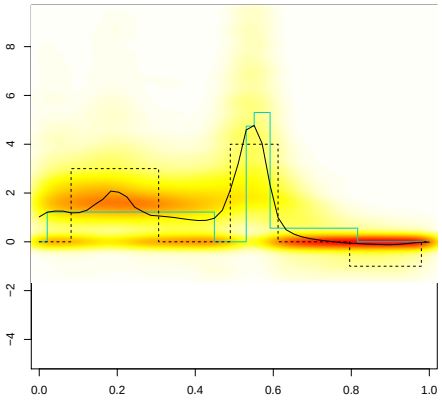


First covariate ($c = 0.6$)

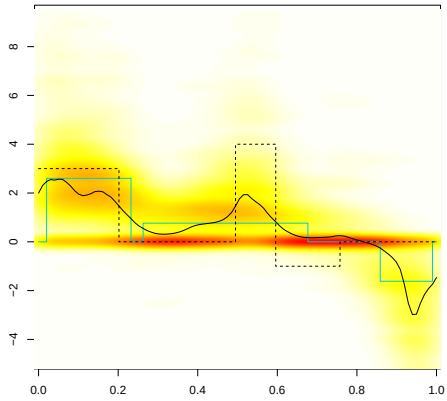


Second covariate ($c = 0.6$)

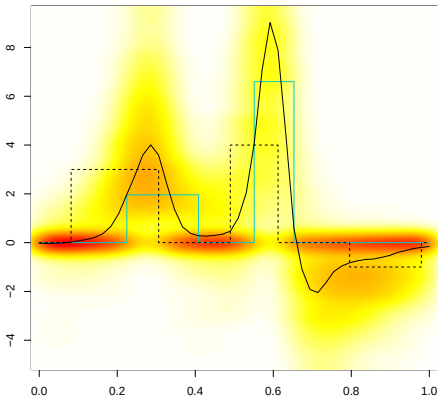
Figure 15: Prior (in gray) and posterior (in black) probabilities of being in the support when $c = 0.3$ or $c = 0.6$. *Bayes estimate of support using Theorem 1 with* $\gamma = 1/2$ *are given in red.*



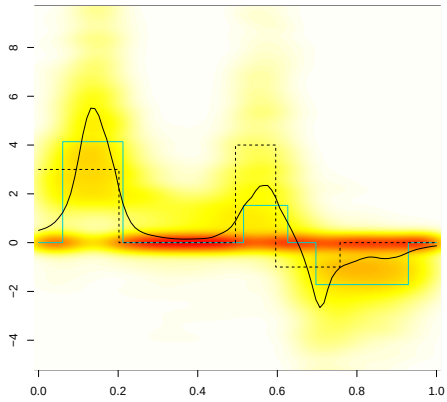
First covariate ($c = 0.3$)



Second covariate ($c = 0.3$)



First covariate ($c = 0.6$)



Second covariate ($c = 0.6$)

Figure 16: Estimates of the coefficient functions when $c = 0.3$ or $c = 0.6$. *For each plot, the black dotted line is the true coefficient function, the solid black line is the L^2-estimate and the light blue line is the stepwise Bliss estimate. The heat maps represents the marginal posterior distributions of $\beta(t)$, as described in Section 1.2. Red (resp. white) colour is used to represent high (resp. low) posterior densities.*

3.5 Application on the truffle dataset

Figure 17 shows the values of BIC for the truffle dataset described in Section 4.

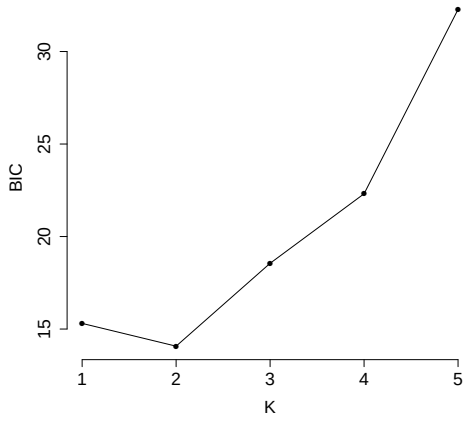


Figure 17: The values of BIC for the truffle dataset described in Section 4 of the paper.

3.6 Computational Time

In this Section, we provde the computational time of the 3 algorithms used in this paper:

1. a Gibbs sampler, for which the full conditional distributions are given in Section 1.3 and 1.4,
2. an algorithm, denoted *density estimation*, which computes the heat map described in Section 1.2 and
3. a simulated annealing algorithm described in Appendix 1.5.

The computational time of the simulated annealing algorithm is negligible ($\sim 1s$). Moreover, the computational time of the *density estimation* algorithms is around one minute and it is mainly due to use of the *kde2d* function. Therefore, below we discuss only the Gibbs sampler.

First, remember that we observe n curves for q covariates, evaluated on the regular grid $\boldsymbol{t^j} = (t_1^j, \ldots, t_p^j)$ for the j^{th} covariates. Remenber also that K_j are the fixed number of intervals in (5) for the j^{th} dimension and $K = \sum_j^q K_j$. We give a sketch of the main steps of the Gibbs sampler algorithm we use. The Gibbs sampler consists of two major steps in terms of computational time. Firstly, we compute the integrals $\int_{\mathcal{I}} x_{ij}(t)\mathrm{d}t$ on all possible intervals $\mathcal{I}$ for $j = 1, \ldots, q$. The intervals $\mathcal{I}$ depend on a center m and a

half-length ℓ. In practice, we consider that m belongs to the grid $\boldsymbol{t}$ and we consider p possible values for ℓ. Hence, we have to compute $n \times p^2 \times q$ integrals. Secondly, we perform a loop of N iterations for which each parameter is updated with regards to full conditional distributions. At each iteration, the longer required computations are:

- q Singular Value Decompositions of matrix $n \times K_j$ to compute the matrix G_j in (18),
- inversions of a matrix $(K+1) \times (K+1)$ and
- determinants of a matrix $(K+1) \times (K+1)$.

For one single functional covariate

For the case of one single functional covariates, we use a data set described in Section 3.1 with different values for n and p (50,100 and 200). We apply the Bliss method with $K = 3$ or $K = 6$ and different number iterations (10 000, 20 000 and 50 000). The computational time are given in Table 2.

Table 2: The computational time of the Gibbs sampler for different values of n, p, K and the number iterations in the one single functional covariate case.

n	p	K	*iter*	computational time	n	p	K	*iter*	computational time
50	50	3	10 000	1.3 min	50	50	6	20 000	11.5 min
100	50	3	10 000	3.3 min	100	50	6	20 000	26.8 min
200	50	3	10 000	11.5 min	200	50	6	20 000	1.5 h
50	100	3	10 000	2.5 min	50	100	6	20 000	23.6 min
100	100	3	10 000	6.7 min	100	100	6	20 000	54.7 min
200	100	3	10 000	23.3 min	200	100	6	20 000	3 h
50	200	3	10 000	5.2 min	50	200	6	20 000	48 min
100	200	3	10 000	13.7 min	100	200	6	20 000	1.9 h
200	200	3	10 000	47.5 min	200	200	6	20 000	6 h
50	50	6	10 000	5.7 min	50	50	3	50 000	6.4 min
100	50	6	10 000	13.4 min	50	100	3	50 000	17 min
200	50	6	10 000	44.2 min	50	200	3	50 000	59.3 min
50	100	6	10 000	11.8 min	50	100	3	50 000	12.6 min
100	100	6	10 000	27.4 min	100	100	3	50 000	33.3 min
200	100	6	10 000	1.5 h	200	100	3	50 000	1.9 h
50	200	6	10 000	24.5 min	50	200	3	50 000	25.5 min
100	200	6	10 000	55.8 min	100	200	3	50 000	1.1 h
200	200	6	10 000	3 h	200	200	3	50 000	3.9 h
50	50	3	20 000	2.5 min	50	50	6	50 000	28.8 min
100	50	3	20 000	6.6 min	100	50	6	50 000	1.1 h
200	50	3	20 000	23.1 min	200	50	6	50 000	3.6 h
50	100	3	20 000	5 min	50	100	6	50 000	59.1 min
100	100	3	20 000	13.3 min	100	100	6	50 000	2.3 h
200	100	3	20 000	46.2 min	200	100	6	50 000	7.4 h
50	200	3	20 000	10.2 min	50	200	6	50 000	2 h
100	200	3	20 000	27.2 min	100	200	6	50 000	4.6 h
200	200	3	20 000	1.6 h	200	200	6	50 000	14.8 h

For two functional covariates

For the case of two functional covariates, we use a data set described in Section 3.1 with different values for n, p_1 and p_2. We vary the value of K and the number iterations as in the previous paragraph. The computational time are given in Table 3.

Table 3: The computational time of the Gibbs sampler for different values of n, p_1, p_2, K and the number iterations in the one single functional covariate case.

n	p_1 and p_2	K	*iter*	computational time	n	p_1 and p_2	K	*iter*	computational time
50	50	3	10 000	6 min	50	50	6	20 000	59 min
100	50	3	10 000	13.7 min	100	50	6	20 000	2 h
200	50	3	10 000	44 min	200	50	6	20 000	5.8 h
50	100	3	10 000	12.2 min	50	100	6	20 000	2.1 h
100	100	3	10 000	27.7 min	100	100	6	20 000	4.1 h
200	100	3	10 000	1.5 h	200	100	6	20 000	11.9 h
50	200	3	10 000	25.1 min	50	200	6	20 000	4.3 h
100	200	3	10 000	56.4 min	100	200	6	20 000	8.3 h
200	200	3	10 000	3 h	200	200	6	20 000	24 h
50	50	6	10 000	29.8 min	50	50	3	50 000	30.6 min
100	50	6	10 000	59 min	50	100	3	50 000	1.2 h
200	50	6	10 000	2.9 h	50	200	3	50 000	3.7 h
50	100	6	10 000	1 h	50	100	3	50 000	1 h
100	100	6	10 000	2 h	100	100	3	50 000	2.3 h
200	100	6	10 000	5.9 h	200	100	3	50 000	7.4 h
50	200	6	10 000	2.2 h	50	200	3	50 000	2.1 h
100	200	6	10 000	4.2 h	100	200	3	50 000	4.6 h
200	200	6	10 000	12 h	200	200	3	50 000	14.8 h
50	50	3	20 000	11.8 min	50	50	6	50 000	2.5 h
100	50	3	20 000	27.4 min	100	50	6	50 000	4.9 h
200	50	3	20 000	1.5 h	200	50	6	50 000	14.6 h
50	100	3	20 000	24.3 min	50	100	6	50 000	5.1 h
100	100	3	20 000	55.6 min	100	100	6	50 000	10.1 h
200	100	3	20 000	3 h	200	100	6	50 000	29.6 h
50	200	3	20 000	49.9 min	50	200	6	50 000	10.7 h
100	200	3	20 000	1.9 h	100	200	6	50 000	20.7 h
200	200	3	20 000	6 h	200	200	6	50 000	60.2 h

References

Bélisle, C. (1992). "Convergence Theorems for a Class of Simulated Annealing Algorithms on $\mathbb{R}^d$." *Journal of Applied Probability*, 29(4): 885–895. 6

Crambes, C., Kneip, A., and Sarda, P. (2009). "Smoothing Splines Estimators for Functional Linear Regression." *The Annals of Statistics*, 37(1): 35–72. 4

Deheuvels, P. (1980). *L'intégrale*. Presse Universitaire de France. 4

Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. (1983). "Optimization by Simulated Annealing." *Science*, 220(4598): 671–680. 3

Phythian, J. E. and Williams, R. (1986). "Direct Cubic Spline Approximation to Integrals with Applications in Nautical Science." *International Journal for Numerical Methods in Engineering*, 23: 305–315. 4

Ramsay, J. and Silverman, B. (2005). *Functional Data Analysis*. Springer-Verlag New York. 4

Robert, C. P. and Casella, G. (2013). *Monte Carlo statistical methods*. Springer-Verlag New York. 2

Rudin, W. (1986). *Real and complex analysis*. New York: McGraw-Hill Inc, 3rd edition. 9