Nonparametric Covariance Estimation for Longitudinal Data via Penalized Tensor Product Splines

Tayler A. Blake[*] Yoonkyung Lee[†]

February 18, 2018

1 Smoothing Spline Varying-coefficient Models for Covariance Estimation

A predominant difficulty in the estimation of covariance matrices is the potentially high dimensionality of the problem, as the number of unknown elements in the covariance matrix grows quadratically with the size of the matrix. It is well-known that the sample covariance matrix can be unstable in high dimensions; ways for controlling the complexity of estimates is highly desirable for improving stability of estimates. In the longitudinal-data literature, it is a common practice to use parametric models for the covariance structure. Many have specified parsimonious parametric models for ϕ_{ijk} to overcome the issue of dimensionality.

We naturally accommodate irregularly spaced data and unequal sample sizes between subjects by defining the autoregressive parameters as the values of a smooth function evaluated at within-subject pairs of observed time points. Furthermore, by viewing $\phi\left(t,s\right)$ as a smooth *bivariate* function, we can utilize the information across the subdiagonals of T to inform the fit, rather than treating each subdiagonal separately. As in the classical nonparametric function estimation setting, we assume ϕ to vary in a high-dimensional (possibly infinite) function space. We propose two representations of $\phi\left(\cdot,\cdot\right)$ and $\sigma\left(\cdot,\cdot\right)$: approximation by smoothing splines and approximation by B-spline basis expansion.

We assume $Y\left(t\right)$ has covariance function $G\left(t,s\right)$ and that $\epsilon\left(t\right)$ follows a zero mean Gaussian white noise process with unit variance. Under mild assumptions regarding the behaviour of Y, then $G\left(t,s\right)$ satisfies some smoothness conditions, where smoothness is defined in terms of square integrability of certain derivatives. We view the entries of Σ as values of G evaluated at the distinct pairs of within-subject observed time points. If we consider the Cholesky decomposition of Σ

within such functional context, it is natural to extent the same notion to the elements of T and D.

[*] The Ohio State University, 1958 Neil Avenue, Columbus, OH 43201

[†] The Ohio State University, 1958 Neil Avenue, Columbus, OH 43201

We view the GARPs $\{\phi_{ijk}\}$ and innovation variances as the evaluation of the smooth functions $\phi(t,s)$ and $\sigma^2(t)$ at observed time points, where

$$\phi_{ijk} = \phi(t_{ij}, t_{ik}), \quad t_{ik} < t_{ij}$$

and

$$\sigma^2_{ij} = \sigma^2(t_{ij}).$$

Following the notation in Yao et al. (2005a), let Ui j and Vi j the jth observations of the random trajectories Xi() and Yi() at a random time points Ti j , respectively, where Ti j are independently drawn from a distribution on compact domain T ? R.

With this definition of ϕ and σ^2, Model **??** becomes

$$y(t_{ij}) = \sum_{k=1}^{j-1} \phi(t_{ij}, t_{ik}) \, y(t_{ik}) + \sigma(t_{ij}) \, \epsilon(t_i) \quad i = 1, \ldots, N, \tag{1}$$

Our formulation transforms the task of estimating a covariance matrix is to estimating the function $\phi(t,s)$ (and $\sigma(t)$) using bivariate (univariate) smoothing. Like other nonparametric situations, we make no assumption about the functional form of ϕ other than that ϕ is smooth, with smoothness defined in terms of square integrability of certain derivatives. Our proposed method for covariance estimation defines a flexible, general framework which makes all of the existing techniques for penalized regression accessible for the seemingly far different task of estimating a covariance matrix.

To add to the interpretability of our proposed models, we propose a rotation of the (t,s) axes:

$$\begin{aligned} l &= t - s \\ m &= \frac{t+s}{2} \end{aligned}$$

The l axis, the "lag" between time points t and s is of special interest within the context of covariance modeling. The class of isotropic covariance functions M is isotropic are defined as those functions $G(t,s) = G(||t-s||)$ that depend on time points t and s only through the Euclidean distance $||t-s||$ between them. Isotropic covariance functions obey some notion of parsimony, so it is convenient to apply regularized estimation to ϕ in terms of the transformed points (l,m). Along with this class of functions, others have proposed parsimonious models for the elements of the Cholesky decomposition in terms of the difference between Re-expressing ϕ in terms of these new arguments, our goal is to estimate

$$\phi^*(l,m) = \phi^*\left(s-t, \frac{1}{2}(s+t)\right) = \phi(s,t) \tag{2}$$

Rather than imposing structure on the unconstrained values of $\phi(s,t)$, we instead consider a rotation of the axes of the input space and re-express ϕ in terms of the transformed coordinates $\begin{bmatrix} l \\ m \end{bmatrix} = \begin{bmatrix} s-t \\ \frac{(s+t)}{2} \end{bmatrix}$

$$\begin{aligned} \phi(s,t) &= \phi^*\left(s-t, \frac{s+t}{2}\right) \\ &= \phi^*(l,m) \end{aligned}$$

While our framework allows for estimation of the autoregressive coefficient function and the innovation variance function via any nonparametric regression setup, we focus on two primary approaches for representing ϕ and σ. First, we assume that ϕ belongs to a reproducing kernel Hilbert space, $\mathcal{H}$ and employ the smoothing spline methods of Kimeldorf and Wahba (see **?** and **?** for comprehensive presentation.) To enhance the statistical interpretability of model parameters, we decompose ϕ into functional components similar to the notion of the main effect and the interaction terms in classical analysis of variance. We adopt the smoothing spline analogue of the classical ANOVA model proposed by Gu (see **?**), and estimation is achieved through similar computational strategies.

Due to rapid development of science and technology, it is possible to collect data that are naturally functions. This type of data , that are referred as functional data, has many applications in various fields of science, including, for example, environmental science, chemometrics, engineering, biomedical studies, public health, and econometrics. Functional data analysis deals with situations in which the individual observed data are infinite-dimensional, such as curves. See Ramsay and Silverman (2002, 2005) and Ferraty and Vieu (2006) for comprehensive discussions on methods and applications for functional data. The functional linear model has been studied extensively in the literature. While flexible enough to closely approximate nearly any functional relationship between a set of predictors and a response, they can also be incredibly interpretable, making them a pragmatic modeling choice when understanding the underlying data generating mechanism is of as much importance as strong predictive capability. By leveraging a likelihood-based estimation procedure, we can naturally relax the restriction that the data on individual curves $Y_1, \ldots, Y_N$ be observed on a grid of time points which are equally spaced and common to all subjects. Simultaneously, by employing a functional view of the autoregressive coefficients $\phi_{t,j}$, we can seamlessly incorporate regularization framework that accompanies the usual nonparametric function estimation problem.

In **??**, we reviewed a number of parsimonious covariance structures which parameterize the dependence between two measurements y_t and y_{t-l} in terms of the time lapse between measurements only, and omit dependence on t itself. This motivates characterizing the GARPs as a function of not (t,s), but of the continuous analogue of lag, $l = t-s$, and its orthogonal direction $m = (t+s)/2$.

(Xi , Yi) : i = 1, 2, ..., n corresponds to functional varying coefficient model, i.e

1.1 Smoothing spline representation of ϕ, σ

Let $\mathcal{H} = \mathcal{H}_0 \oplus \mathcal{H}_1$ be the reproducing kernel Hilbert space (r.k.h.s) corresponding to the tensor product of the first-order and second-order Sobolev spaces:

$$\mathcal{H} = \mathcal{H}_l \otimes \mathcal{H}_m, \;\; \mathcal{H}_l = W_2\left(0,1\right), \;\; \mathcal{H}_m = W_1\left(0,1\right) \text{ where}$$

$$W_m\left(0,1\right) \equiv \{f: \;\; f', \ldots, f^{(m-1)} \text{absolutely continuous}, \int_0^1 \left(f^{(m)}\right)^2 dt < \infty\}$$

We seek $\phi^*\left(\cdot,\cdot\right) \in \mathcal{H}$ which minimizes

$$\frac{1}{2}\sum_{i=1}^{N}\sum_{j=2}^{n_i}\sigma_{ij}^{-2}\left(y\left(t_{ij}\right) - \sum_{k=1}^{n_i-1}\phi^*\left(l_{jk}^i, m_{jk}^i\right)y\left(t_{ik}\right)\right)^2 + \lambda J\left(\phi^*\right) \tag{3}$$

where $P_1\phi^*$ is the projection of ϕ^* onto $\mathcal{H}_1$, $J\left(\phi^*\right) = ||P_1\phi^*||^2$. Define the differential operator $M_\nu f = \int_0^1 f^{(m)}\left(x\right)dx$, $\nu = 1, \ldots, m$ and endow $W_m\left(0,1\right)$ with inner product

$$\langle f, g\rangle = \langle f, g\rangle_0 + \langle f, g\rangle_1 = \sum_{\nu=0}^{m-1} M_\nu f M_\nu g + \int_0^1 f^{(m)}\left(x\right)g^{(m)}\left(x\right)dx \tag{4}$$

which induces norm

$$||f||^2 = \langle f, f\rangle = \langle f, f\rangle_0 + \langle f, f\rangle_1 = ||P_0 f||^2 + ||P_1 f||^2$$

Let $k_j\left(x\right) = B_j\left(x\right)/j!$ for $x \in [0,1]$, where $B_j\left(x\right)$ is the j^{th} Bernoulli polynomial which can be defined according to the recursive relationship:

$$B_0\left(x\right) = 1, \qquad \frac{d}{dx}B_r\left(x\right) = rB_{r-1}\left(x\right)$$

Noting that $M_\nu B_r = \delta_{\nu-r}$, W_m can be written as a direct sum of the m orthogonal subspaces: $\{k_r\}_{r=0}^{m-1}$ and W_m^1. Here, $\{k_r\}$ is the subspace spanned by k_r and W_m^1 is the space orthogonal to $W_m^0 \equiv \{1\} \oplus \{k_1\} \oplus \cdots \oplus \{k_{m-1}\}$ which satisfies

$$W_m^1 = \{f: M_\nu f = 0, \;\; \nu = 0, 1, \ldots, m-1\}$$

Writing $\mathcal{H}$ as the tensor product of the two decomposed Sobolev spaces, we have

$$\begin{aligned}
\mathcal{H} = \mathcal{H}_l \otimes \mathcal{H}_m &= W_2 \otimes W_1 \\
&= \left[W_2^0 \oplus W_2^1\right] \otimes \left[W_1^0 \oplus W_1^1\right] \\
&= \left[\left[\{1\} \oplus \{k_1\}\right] \oplus W_2^1\right] \otimes \left[\{1\} \oplus W_1^1\right] \\
&= \left[\{1\} \oplus \{k_1\}\right] \oplus W_2^1 \oplus W_1^1 \oplus \left[\{k_1\} \otimes W_1^1\right] \oplus \left[W_2^1 \otimes W_1^1\right] \\
&\equiv \left[\mathcal{H}_{\mu^*} \oplus \mathcal{H}_l^0\right] \oplus \left[\mathcal{H}_l^1 \oplus \mathcal{H}_m^1 \oplus \mathcal{H}_{lm}^{01} \oplus \mathcal{H}_{lm}^{11}\right] \\
&= \mathcal{H}_0 \oplus \mathcal{H}_1
\end{aligned} \tag{5}$$

where the functional components corresponding to $\mathcal{H}_{\mu}$, $\mathcal{H}_{l}^{0}$, $\mathcal{H}_{l}^{1}$, $\mathcal{H}_{m}^{1}$, and $[\mathcal{H}_{lm}^{01} \oplus \mathcal{H}_{lm}^{11}]$ are the overall mean, the nonparametric main effect of l, the parametric main effect of l, the parametric main effect of m, the nonparametric-parametric interaction, and the parametric-parametric interaction (between l and m). Given this decomposition of the function space, any $\phi^{*} \in \mathcal{H}$ may be written as a sum of components from each of the

$$\phi^{*}(l,m) = \mu^{*} + \phi_{l}^{*}(l) + \phi_{m}^{*}(m) + \phi_{lm}^{*}(l,m) \tag{6}$$

where $\int_0^1 \phi_l^{*}(l)\,dl = \int_0^1 \phi_m^{*}(m)\,dm = 0$, $\int_0^1 \phi_{lm}^{*}(l,m)\,dl = \int_0^1 \phi_{lm}^{*}(l,m)\,dm = 0$. The reproducing kernel (r.k.) for $\{k_r\}$ is $k_r(x)\,k_r(x')$. It can be verified that the r.k. for W_m^1 (Craven and Wahba 1979) is given by $R^1(x,x') = k_m(x)\,k_m(x') + (-1)^{m-1} k_{2m}([x-x'])$ where $[\alpha]$ is the fractional part of α. The r.k. for W_m is given by

$$\begin{aligned}
R(x,x') &= R^0(x,x') + R^1(x,x') \\
&= \left[\sum_{\nu=1}^{m-1} k_{\nu}(x)\,k_{\nu}(x')\right] + \left[k_m(x)\,k_m(x') + (-1)^{m-1} k_{2m}([x-x'])\right]
\end{aligned}$$

Using the fact that the r.k. for a tensor product space is the product of the corresponding reproducing kernels, the r.k. for $\mathcal{H}$ is given by

$$\begin{aligned}
R((l,m),(l',m')) &= R_l(l,l') \times R_m(m,m') \\
&= \left[R_l^0(l,l') + R_l^1(l,l')\right] \times \left[R_m^0(l,l') + R_m^1(l,l')\right] \\
&= R_l^0(l,l')\,R_m^0(m,m') + R_l^0(l,l')\,R_m^1(m,m') \\
&\qquad\qquad + R_l^1(l,l')\,R_m^0(m,m') + R_l^1(l,l')\,R_m^1(m,m') \\
&= \left[k_1(l)\,k_1(l')\right] + \left[R_l^1(l,l') + k_1(l,l')\,R_m^1(m,m') + R_l^1(l,l')\,R_m^1(m,m')\right] \\
&= R^0((l,m),(l',m')) + R^1((l,m),(l',m'))
\end{aligned} \tag{7}$$

We must introduce some notation to simplify the following expression of the form of the elements in $\mathcal{H}$. Denote the set of unique pairs of observed within-subject time points and the corresponding set of unique transformed coordinates by $\mathcal{W}$ and $\mathcal{W}^{*}$, respectively:

$$\begin{aligned}
\mathcal{W} &= \bigcup_{i=1}^{N} \bigcup_{j>k} (t_{ij}, t_{ik}) \\
\mathcal{W}^{*} &= \bigcup_{i=1}^{N} \bigcup_{j>k} \left(t_{ij} - t_{ik}, \frac{1}{2}(t_{ij} + t_{ik})\right) = \bigcup_{i=1}^{N} \bigcup_{j>k} (l_{jk}^{i}, m_{jk}^{i})
\end{aligned}$$

with $|\mathcal{W}| = |\mathcal{W}^{*}| = N_{\phi^{*}}$. For simplicity of presentation, relabel the elements of $\mathcal{W}^{*}$ so that

$$\mathcal{W}^{*} = \{(l_1, m_1), (l_2, m_2), \ldots, \left(l_{N_{\phi^{*}}}, m_{N_{\phi^{*}}}\right)\}$$

One may verify that any $\phi^* \in \mathcal{H}$ can be written

$$\phi^*(l,m) = d_0 + d_1 k_1(l) + \sum_{i=1}^{n} c_i R_1((l,m),(l_i,m_i)) + \rho(l,m) \tag{8}$$

where $\rho \perp \mathcal{H}_0 = \{1\} \oplus \{k_1\},\ \mathrm{span}\{R_1((l_i,m_i),\cdot)\}$. It can be shown that the minimizer of 3 has $\rho = 0$, so that the $\phi^* \in \mathcal{H}$ minimizing 3 can be written as a (finite) linear combination of inner products:

$$\phi^*(l,m) = d_0 + d_1 k_1(l) + \sum_{i=1}^{n} c_i R_1((l,m),(l_i,m_i)) \tag{9}$$

The proof entails demonstrating that ρ does not improve the first term in (3) (the data fit functional) and only adds to the penalty term, $J(\phi^*)$. Details are left to the appendix **??**. Let Φ^* be the $N_{\phi^*} \times 1$ vector of regression coefficients given by (**??**) corresponding to ϕ^* evaluated at the elements of $\mathcal{W}^*$, $\Phi^* = \left(\phi^*_1, \phi^*_2, \ldots, \phi^*_{N_{\phi^*}}\right)^T$. Let $d = (d_0, d_1)^T$, $c = \left(c_1, \ldots, c_{N_{\phi^*}}\right)^T$, and $b = (b_1, \ldots, b_{N_m})^T$. Define K_{11}, K_{12}, K_{22}, B_1, and B_2 as follows:

$$\begin{array}{ll}
K_{11}[i,j] = R_1((l_i,m_i),(l_j,m_j)) & i,j = 1,\ldots,N_{\phi^*} \\
K_{12}[i,j] = R_1((l_i,m_i),(0,m_j)) & i = 1,\ldots,N_{\phi^*},\ j = 1,\ldots,N_m \\
K_{22}[i,j] = R_1((0,m_i),(0,m_j)) & i,j = 1,\ldots,N_m \\
B_1[i,j] = k_j(l_i) & i = 1,\ldots,N_{\phi^*},\ j = 1,2 \\
B_2[i,j] = k_j(0) & i = 1,\ldots,N_m,\ j = 1,2
\end{array}$$

$$K = \begin{bmatrix} K_{11} & K_{12} \\ K_{12}^T & K_{22} \end{bmatrix} = \begin{bmatrix} K_1 \\ K_2 \end{bmatrix}; \quad B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}$$

In matrix notation:

$$\Phi^* = Sd + Rc$$

2 A B-spline representation for pp functions

Definition 2.1. Let $t = \{t_i\}$ denote a non-decreasing sequence. The i^{th} B-spline of order k which corresponds to the knot sequence t is defined by

$$B_{i,k,t}(x) = (t_{i+k} - t_i)[t_i, \ldots, t_{i+k}](\cdot - x)_+^{k-1} \tag{10}$$

The placeholder notation, $(\cdot - x)_+^{k-1}$, is used to indicate that the k^{th} divided difference of the function $g(t) = (t-x)_+^{k-1}$ is obtained by fixing x and applying the divided difference to $g(t)$ as a function of t alone. Henceforth, we will write B_i rather than $B_{i,k,t}$ when the spline order and knot sequence can be inferred from surrounding context.

2.1 Properties of B-splines

I. $B_i(x)$ has isolated support:

$$B_i(x) = 0, \quad x \notin [t_i, t_{i+k}]$$

To see this, note that if $x \notin [t_i, t_{i+k}]$, then $g(t) = (t-x)_+^{k-1}$ is a polynomial of degree $< k$ on $[t_i, t_{i+k}]$, thus by **?? ??**,

$$[t_i, \ldots, t_{i+k}]\, g = 0.$$

As a result, for a set of B-splines of order k corresponding to the knot sequence t, only k of them are nonzero on $[t_j, t_{j+k}]$: $B_{j-k+1}, B_{j-k+2}, \ldots, B_j$.

II. The i^{th} B-spline of order is defined as the k^{th} divided difference of $(\cdot - x)_+^{k-1}$ times a normalization factor: $(t_{i+k} - t_i)$. This normalization, using **?? ??**, allows us to write

$$B_i(x) = [t_{i+1}, \ldots, t_{i+k}]\,(\cdot - x)_+^{k-1} - [t_i, \ldots, t_{i+k-1}]\,(\cdot - x)_+^{k-1} \tag{11}$$

For $x \in (t_j, t_{j+1})$, by **?? ??**,

$$\begin{aligned}
\sum_i B_i(x) &= \sum_{i=j+1-k}^{j} B_i(x) \\
&= \sum_{i=j+1-k}^{j} [t_{i+1}, \ldots, t_{i+k}]\,(\cdot - x)_+^{k-1} - \sum_{i=j+1-k}^{j} [t_i, \ldots, t_{i+k-1}]\,(\cdot - x)_+^{k-1} \\
&= [t_{j+1}, \ldots, t_{j+k}]\,(\cdot - x)_+^{k-1} - [t_{j+1-k}, \ldots, t_j]\,(\cdot - x)_+^{k-1} \\
&= 1 - 0
\end{aligned} \tag{12}$$

The last equality in 12 is a consequence of the following: for $x \in (t_j, t_{j+1})$, $g(t) = (t-x)_+^{k-1}$ is a $k-1$ degree polynomial with unit leading coefficient on $[t_{j+1}, t_{j+k}]$, so by **?? ??**,

$$[t_{j+1}, \ldots, t_{j+k}]\, g = 1.$$

On $[t_{j+1-k}, t_j]$, g is identically 0, hence $[t_{j+1-k}, \ldots, t_j]\, g = 0$.

III. Each $B_i(x)$ is positive on its support. Applying Leibnitz's formula (**?? ??**) to the product

$$[t_i, \ldots, t_{i+k}]\,(t-x)_+^{k-1} = [t_i, \ldots, t_{i+k}]\,(t-x)\,(t-x)_+^{k-2},$$

we have

$$\begin{aligned}
\left[t_i, \dots, t_{i+k}\right](t-x)_+^{k-1} &= \left[t_i, \dots, t_{i+k}\right](t-x)(t-x)_+^{k-2} \\
&= \sum_{r=i}^{i+k} \left[t_i, \dots, t_{i+r}\right](t-x)\left[t_r, \dots, t_{i+k}\right](t-x)_+^{k-2} \\
&= \Bigg[[t_i](t-x) \Bigg] \Bigg[\left[t_i, \dots, t_{i+k}\right](t-x)_+^{k-2} \Bigg] \\
&\qquad + \Bigg[[t_i, t_{i+1}](t-x) \Bigg] \Bigg[\left[t_{i+1}, \dots, t_{i+k}\right](t-x)_+^{k-2} \Bigg] \\
&= (t_i - x)\left[t_i, \dots, t_{i+k}\right](t-x)_+^{k-2} \\
&\qquad + 1 \cdot \left[t_{i+1}, \dots, t_{i+k}\right](t-x)_+^{k-2} \qquad (13)
\end{aligned}$$

since $[t_i, \dots, t_j](\cdot - x) = 0$ for $j > i+1$. By **?? ??**,

$$(t_i - x)\left[t_i, \dots, t_{i+k}\right]g = \frac{t_i - x}{t_{i+k} - t_i}\Bigg[\left[t_{i+1}, \dots, t_{i+k}\right]g - \left[t_i, \dots, t_{i+k-1}\right]g \Bigg],$$

and we may express 13 as

$$\begin{aligned}
\left[t_i, \dots, t_{i+k}\right](\cdot - x)_+^{k-1} = &\frac{x - t_i}{t_{i+k} - t_i}\left[t_i, \dots, t_{i+k-1}\right](\cdot - x)_+^{k-2} \\
&+ \frac{t_{i+k} - x}{t_{i+k} - t_i}\left[t_{i+1}, \dots, t_{i+k}\right](\cdot - x)_+^{k-2}
\end{aligned}$$

which we can write in terms of the normalized B-spline:

$$\frac{B_{i,k}(x)}{t_{i+k} - t_i} = \frac{x - t_i}{t_{i+k} - t_i}\frac{B_{i,k-1}(x)}{t_{i+k-1} - t_i} + \frac{t_{i+k} - x}{t_{i+k} - t_i}\frac{B_{i+1,k-1}(x)}{t_{i+k} - t_{i+1}} \qquad (14)$$

This shows that we can write the i^{th} B-spline of order k as a convex combination of the i^{th} and $(i+1)^{st}$ B-splines of order $k-1$ since

$$\frac{x - t_i}{t_{i+k} - t_i} + \frac{t_{i+k} - x}{t_{i+k} - t_i} = 1,$$

and each of these weights are positive for $t_i < x < t_{i+1}$. If

$$B_{j,k-1}(x) > 0, \quad t_j < x < t_{j+k-1} \text{ for all } j,$$

then by 14, we have that

$$B_{i,k}(x) > 0, \qquad t_i < x < t_{i+k}$$

since $B_{j,k-1} = 0$ for $x \notin [t_j, t_{j+k}]$ by 2.1 I and by induction over k, starting with the fact that

$$B_j, 1(x) = \begin{cases} 1 & t_j \le x < t_{j+1} \\ 0 & otherwise \end{cases}$$

Properties I, II, and III demonstrate that a sequence of B-splines form a *partition of unity*: a set of non-negative functions which sum, pointwise, to one.

Definition 2.2. The *B-representation of* $f \in \mathcal{P}_{k,\xi,\nu}$ consists of

I. integers k and n specifying the order of f as a pp function and the number of linear parameters,

$$n = kl - \sum_i \nu_i = \dim\left(\mathcal{P}_{k,\xi,\nu}\right),$$

respectively.

II. The knot vector $t = \{t_i\}$, $i = 1, \ldots, n+k$ with elements arranged in increasing order, constructed according to Theorem **??**, via ξ and ν.

III. The B-spline coefficients $\alpha = \{\alpha_i\}$, $i = 1, \ldots, n$ for the knot sequence, t.

Given I, II, and III in 2.2, the function value at $x \in [t_k, t_{n+1}]$ is given by

$$f(x) = \sum_{i=1}^{n} \alpha_i B_i(x),$$

and in particular, by I, for $x \in [t_j, t_{j+1}]$,

$$f(x) = \sum_{i=j}^{j+k-1} \alpha_i B_i(x).$$

3 Single-regressor varying coefficient models via B-spline basis expansions

Hastie and Tibshirani were the first to introduce the varying coefficient model, which supplies a modeling approach which permits interpolation of regressors and response variables which varying according to an *indexing variable* at values of this indexing variable where there is either missing data of only a single observation and slope estimation is not feasible. In the section that follows, we will discuss the approach to smoothing the coefficient vector (and *not* the regressor, $x(t)$) first, for mechanical demonstration of parameterization and estimation of the coefficient function via B-spline basis expansion, at a predetermined set of values of an indexing variable, t (knots), then following the approach of Eilers and Marx by assuming that the number and position of the knots are unknown and using penalized B-splines, or P-splines.

Consider data of the form

$$(x_i, y_i, t_i), \qquad i = 1, \ldots, m$$

where y_i is the response, x_i is the single (univariate) regressor variable, and t_i is an indexing variable. We first consider a simple situation as an introductory warmup for demonstrating the mechanics of the varying coefficient model. Suppose we wish to fit a scatterplot smoother to the points (t_i, y_i) using a B-spline basis expansion. Assume that we can model

$$y(t) = f(t) + \epsilon(t) \tag{15}$$

where ϵ is a zero-mean error process. Modeling the mean function as a q^{th}-order B-spline, we can rewrite 15 as

$$y(t) = \sum_{j=1}^{K} \alpha_j B_j(t) + \epsilon(t) \tag{16}$$

Assume we use K of basis functions in our expansion of f. Let $y = (y_1, \ldots, y_m)^T$, and let B denote the $m \times K$ design matrix with $i - j^{th}$ element given by the j^{th} order-q B-spline evaluated at the i^{th} value of t:

$$b_{ij} = B_j(t_i),$$

$i = 1, \ldots, m$, $j = 1, \ldots, K$. Then in matrix notation, we may write the mean vector

$$\mu = E[y] = B\alpha$$

where α is the vector of K unknown basis coefficients. We take $\hat{\alpha}$ to be the minimizer of

$$\begin{aligned} S &= \sum_{i=1}^{m} \left(y_i - \sum_{j=1}^{K} \alpha_j B_j(t_i) \right)^2 \\ &= |y - B\alpha|^2 \end{aligned} \tag{17}$$

$$B^T B\alpha = B^T y$$

which has explicit solution

$$\hat{\alpha} = \left(B^T B\right)^{-1} B^T y$$

Given $\hat{\alpha}$, one may estimate the response at any new value of t, say t^*, by

$$\hat{y}(t^*) = \sum_{j=1}^{K} \hat{\alpha}_j B_j(t^*).$$

3.1 B-spline estimators for varying coefficient models with fixed knots

To extend the varying intercept model 15 to accommodate for controlling for another regressor, it is natural to consider the varying coefficient model; the single regressor varying-coefficient (VC) model extends the classical linear model by allowing the slope coefficient to vary smoothly in the dimension of the indexing variable, t. The single-index varying coefficient model assumes that the mean response is of the form

$$E[Y(t)] = \beta_0(t) + \beta_1(t)\, x(t) \tag{18}$$

where $\beta_0(t)$ is the smooth varying intercept function and $\beta_1(t)$ is the smooth slope function of interest. This model generalizes the well known simple linear regression model

$$E[Y(t)] = \beta_0 + \beta_1 x(t)$$

by trading the static regression coefficients for smooth coefficient functions which are assumed to varying across an indexing variable, t. This allows for the regressor variable to having modified effect, depending on the value of t. Using a set of predetermined knots along the t axis, the VC model can be fit in a fashion similar to that required for fitting model 15, requiring only minor adjustments to the design matrix. In matrix notation as described in 17, the mean vector may be written

$$\mu = B\alpha_0 + diag\{x(t)\} B\alpha_1 \tag{19}$$

where $diag\{x(t)\}$ is the $m \times m$ diagonal matrix of regressor measurements which ensures that the varying coefficients are appropriately weighted according to the correct value of x by aligning the regressor function with the corresponding slope value. Letting $U = diag\{x(t)\} B$, 19 becomes

$$\begin{aligned} \mu &= [B|U] \left(\alpha_0^T, \alpha_1^T\right)^T && (20) \\ &\equiv Q\alpha && (21) \end{aligned}$$

where α is the augmented vector of basis coefficients. Here, the same basis is used for smoothing both the varying intercept as well as the varying slope function; this is feasible because both components varying along the same indexing variable. One can relax this structure and allow each additive term to varying according to its own indexing variable. This, of course, requires a separate B-spline basis for each component. Again using least squares techniques as with the varying intercept-only model, we take $\hat{\alpha}$ to minimize

$$S = |y - Q\alpha|^2 \tag{22}$$

which has explicit solution

$$\hat{\alpha} = \left(Q^T Q\right)^{-1} Q^T y.$$

It is of interest to notice that Q is simply a row scaling of the original B-spline design matrix, B; thus, accommodating a varying slope function equates to the simple basis function regression setting with a modified basis, UB. Using the modified basis functions as covariates, estimation of model the varying coefficient model equates to a multiple regression problem. Each of the estimated smooth components are given by

$$\hat{\beta}_k(t) = B\hat{\alpha}_k, \qquad k = 0, 1$$

and the estimate of the smooth mean function is obtained via

$$\begin{aligned} \hat{\mu} &= Q\hat{\alpha} \\ &= Hy \end{aligned}$$

where $H = Q\left(Q^TQ\right)^{-1}Q^T$ is the "hat" matrix. This will be discussed in further detail in later sections on smoothing parameter selection and model tuning.

4 P-spline estimators for regularized estimation of fitted curves

The mechanics in the previous section rely on apriori knowledge of the number and locations of the knots $\{t_j\}$, $j = 1, \ldots, K$. In practice this information is readily available, but has a considerable impact on the behaviour of the estimated coefficient functions, as the smoothness of a fitted curve can be controlled by the number of B-splines used in the basis expansion used to approximate the curve. Fewer knots (thus, fewer basis functions) lead to smoother fits. This choice presents a model selection problem, as too many knots lead to overfitting while too few knots lead to underfitting. Optimal knot placement has been closely examined, with some authors proposing automatic methods for optimizing the number and the positions of the knots (Friedman and Silverman, 1989; Kooperberg and Stone, 1991,1992). This is a difficult numerical problem requiring nonlinear optimization, and is still an open problem today. However, limiting the number of B-splines is not the only approach to controlling the complexity of the fitted function.

As in chapter smoothing spline chapter, we can append a penalty on the coefficients of the basis functions to the goodness of fit measure, and by optimizing this augmented objective function, we can achieve as much smoothness in the fitted function as desired. citeo1986statistical was the first to propose using a rich B-spline basis and applying a discrete penalty to the spline coefficients.

He proposed a penalty on the second derivative to restrict the flexibility of the fitted curve, similar to the penalty pioneered for smoothing splines by Reinsch (1967). This penalty has become the standard in much of the spline literature; see Eubank (1988), Wahba (1990) and Green and Silverman (1994). This measure of roughness of a curve is given by

$$J = \int_l^u \left[f''(x)\right]^2 \, dx$$

where l and u are the bounds on the domain of x. Using the properties of B-splines, if $f(x) = \sum_j \beta_j B_j(x)$, one can derive a banded matrix P such that

$$J = \beta' P \beta$$

where $\beta = (\beta_1, \ldots, \beta_n)$, and the i-j^{th} element of P is given by

$$p_{ij} = \int_l^u B_i''(x) B_j''(x) \, dx.$$

He then proposed minimizing

$$\begin{aligned} Q(\beta, \lambda) &= \sum_{i=1}^m \left(y_i - \sum_j \beta_j B_j(x_i)\right)^2 + \lambda \int_l^u \left[f''(x)\right]^2 \, dx \\ &= |y - B\beta||^2 + \lambda \beta' P \beta \end{aligned}$$

The computation of P is nontrivial and becomes very tedious when the third and fourth derivative are used as the roughness measure. citewand2008semiparametric extend O'Sullivan's work to higher order derivatives for general degree B-splines and derive an exact matrix algebraic expression for the penalty matrices. In the cubic case, the expression is a result of the application of Simpson's Rule applied to the inter-knot differences since each $B_i''B_j''$ is a piecewise quadratic function. The penalty may be written

$$P = (B'')' \operatorname{diag}(\omega)\, B'',$$

where B'' is the $3\,(n+7) \times (n+4)$ matrix with i-j^{th} entry given by $B_j''\,(x_i^*)$, x_i^* is the i^{th} element of

$$\left(\phi_1, \frac{\phi_1+\phi_2}{2}, \phi_2, \phi_2, \frac{\phi_2+\phi_3}{2}, \phi_3, \ldots, \phi_{n+7}, \frac{\phi_{n+7}+\phi_{n+8}}{2}, \phi_{n+8}\right),$$

and ω is the $3\,(n+7) \times 1$ vector given by

$$\omega = \left(\frac{1}{6}(\Delta\phi)_1, \frac{4}{6}(\Delta\phi)_1, \frac{1}{6}(\Delta\phi)_1, \frac{1}{6}(\Delta\phi)_2, \frac{4}{6}(\Delta\phi)_2,\right.$$
$$\left.\frac{1}{6}(\Delta\phi)_2, \ldots, \frac{1}{6}(\Delta\phi)_{n+7}, \frac{4}{6}(\Delta\phi)_{n+7}, \frac{1}{6}(\Delta\phi)_{n+7}\right)$$

where $(\Delta\phi)_j = \phi_{j+1} - \phi_j$. They generalize this to the case of any order penalty and present a table of formulas for constructing any arbitrary penalty matrix, P.

4.1 Difference penalties

Imposing difference penalties on B-spline basis expansions generalizes and simplifies the approach outlined in the previous section in a way that permits application in any context where regression on B-splines is useful. Penalized B-splines, or *P-splines*, are an alternative an approach to nonparametric smoothing which circumvent any complexity associated with constructing such penalty matrices by ommitting derivatives and integrals altogether. Instead, smoothness is imposed via a discrete penalty matrix based on finite difference formulas which is simple to compute. This approach achieves smoothness in fitted functions in two ways:

I. To avoid the difficulty of choosing the optimal set of knots, use a B-spline basis with a large number of equally spaced knots, purposefully overfitting the smooth coefficient vectors.

II. Augment the goodness of fit measure with a difference penalty to prevent overfitting and accomodate a potentially ill-conditioned fitting procedure.

Using the properties of B-splines derived in B-spline section, it is relatively straightforward to show that the simplified penalty is nearly equivalent to the derivative-based penalty and that for second order differences, P-splines are very similar to O'Sullivan's approach. In some applications, it can be useful to use differences of a smaller or higher order in the penalty, and the P-spline framework makes the use of a penalty of any arbitrary order nearly seamless.

Consider the varying intercept-only model defined in 15 for the regression of M data points (t_i, y_i) on a set of K B-splines, $\{B_j\}$. By letting the number of knots, K, be relatively large, we allow more variation in fitted curve than the data reasonably justify. To make the result less flexible and avoid overfitting, O'Sullivan imposed a penalty on the second derivative of the fitted curve and appended this to the residual sum of squares, giving way to the objective function

$$\sum_{i=1}^{m} \left\{ y_i - \sum_{j=1}^{K} \beta_j B_j\left(t_i\right) \right\}^2 + \lambda \int_{t_{min}}^{t_{max}} \left\{ \sum_{j=1}^{K} \beta_j B_j''\left(t\right) \right\}^2 dt. \tag{23}$$

The integral of the square of the second derivative of a fitted function has become common as a smoothness penalty since the seminal work on smoothing splines by Reinsch (1967), though it is useful to note that there is nothing particularly special about the second derivative. One could easily specify higher or lower order derivatives in smoothness penalties. In the context of smoothing splines, the first derivative leads to simple equations and a piecewise linear fit, while higher derivatives lead to systems of equations with a high bandwidth and a very smooth fit.

Proposed for smoothing curves by citewhittaker1922new, difference penalties have been utilized for nearly a century, with more recent applications outlined in citeeilers1991penalized, citeeilers1991nonparametric, and citeeilers1995indirect. The finite difference penalty is easily introduced into regression equations, making it feasible to evaluate the impact of different orders of the differences on the fitted model. In some applications, it is useful to work with third and fourth order differences, since for high values of λ, the fitted curve approaches a parametric polynomial model. Detailed discussion on the effect of the smoothing parameter on fitted functions will follow. Let D_d denote the matrix difference operator; that is, $D_d\beta = \Delta^d\beta$, where

$$\begin{aligned} \Delta\alpha_j &= \alpha_j - \alpha_{j-1}, \\ \Delta^2\alpha_j &= \Delta\left(\Delta\alpha_j\right) = \alpha_j - 2\alpha_{j-1} + \alpha_{j-2}, \end{aligned}$$

and in general,

$$\Delta^d\alpha_j = \Delta\left(\Delta^{d-1}\alpha_j\right)$$

The $(K-d) \times K$ differencing matrix D_d is sparse for reasonably small values of d; for example, D_1 and D_2 for small dimensions are given by

$$D_1 = \begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}; \qquad D_2 = \begin{bmatrix} 1 & -2 & 1 & 0 \\ 0 & - & -2 & 1 \end{bmatrix}$$

citeeilers1996flexible propose to base the penalty on (higher-order) finite differences of the coefficients of adjacent B-splines:

$$\lambda|D_d\alpha|^2 = \lambda\alpha' D_d' D_d\alpha = \lambda\alpha' P\alpha,$$

Replacing O'Sullivan's penalty with the difference penalty, we can control the smoothness of the fitted mean function $\mu = \beta_0(t) = B\alpha$ by minimizing

$$S_\lambda = |y - B\alpha|^2 + \lambda|D_d\alpha|^2$$

This approach reduces the dimensionality of the problem to the number of B-splines, K instead of the number of observations, M , as with smoothing splines. The tuning parameter λ permits continuous control over smoothness of the fit. We will demonstrate that the difference penalty is a good discrete approximation to the integrated square of the k^{th} derivative, and with this penalty, moments of the data are conserved and polynomial regression models occur as limits for large values of λ. We will explore the connection between a penalty on second-order differences of the B-spline coefficients and O'Sullivan's choice of a penalty on the second derivative of the fitted function. However, the difference penalty can be handled mechanically for any order of the differences. citeo1986statistical used third-degree B-splines and the following penalty:

$$h^2P = \lambda \int_{t_{min}}^{t_{max}} \left\{ \sum_j \alpha_j B''_{j,3}(t) \right\}^2 dt \tag{24}$$

From the derivative properties of B-splines, it follows that

$$h^2P = \lambda \int_{t_{min}}^{t_{max}} \sum_j \sum_k \Delta^2\alpha_j \Delta^2\alpha_k B_{j,1}(t) B_{k,1}(t)\, dt \tag{25}$$

Most of the cross products of $B_{j,1}(t)$ and $B_{k,1}(t)$ vanish since B-splines of degree 1 only overlap when j is $k-1$, k, or $k+1$. Thus, we have that

$$\begin{aligned} h^2P &= \lambda \int_{t_{min}}^{t_{max}} \left[\left\{ \sum_j \Delta^2\alpha_j B_j(t,1) \right\}^2 + 2\sum_j \Delta^2\alpha_j \Delta^2\alpha_{j-1} B_j(t,1) B_{j-1}(t,1) \right] dt \\ &= \lambda \left[\sum_j \left(\Delta^2\alpha_j\right)^2 \int_{t_{min}}^{t_{max}} B_j^2(t,1)\ dt + 2\sum_j \Delta^2\alpha_j \Delta^2\alpha_{j-1} \right] \end{aligned} \tag{26}$$

or

$$\begin{aligned} h^2P = \lambda \sum_j \left(\Delta^2\alpha_j\right)^2 \int_{t_{min}}^{t_{max}} B_{j,1}^2(t)\, dt + 2\lambda \sum_j \Delta^2\alpha_j \Delta^2\alpha_{j-1} \\ + \int_{t_{min}}^{t_{max}} B_{j,1}(t) B_{j-1,1}(t)\, dt \end{aligned} \tag{27}$$

which can be written as

$$h^2P = \lambda \left\{ c_1 \sum_j \left(\Delta^2\alpha_j\right)^2 + c_2 \sum_j \Delta^2\alpha_j \Delta^2\alpha_{j-1} \right\} \tag{28}$$

where, for given equidistant knots, c_1 and c_2 are constants given by

$$\begin{aligned} c_1 &= \int_{t_{min}}^{t_{max}} B_{j,1}^2 (t)\, dt \\ c_2 &= \int_{t_{min}}^{t_{max}} B_{j,1} (t)\, B_{j-1,1} (t)\, dt \end{aligned} \tag{29}$$

O'Sullivan's ridge-like B-spline penalty in Equation 24 can be written as a linear combination of a difference penalty (**??**) and the sum of the cross products of neighboring second differences. The second term in Equation 28 leads to a complex objective function when minimizing the penalized likelihood, where seven adjacent spline coefficients occur, as opposed to five if only the first term in Equation 28 is used in the penalty. The additional complexity is due to overlapping B-splines, which quickly increases when using higher order differences and higher order B-splines. The use of a difference penalty allows us to sidestep the difficulty of constructing a procedure for incorporating the penalty in the likelihood equations.

Define $\hat{\alpha} = (\hat{\alpha}_1, \hat{\alpha}_2, \dots, \hat{\alpha}_K)$ to be the minimizer of S_λ:

$$S_\lambda = \sum_{i=1}^{m} \left\{ y_i - \sum_{j=1}^{K} \alpha_j B_j (t_i) \right\}^2 + \lambda \sum_{j=d+1}^{K} \left(\Delta^d \alpha_j \right)^2$$

In vector notation, this may be written

$$\begin{aligned} S_\lambda &= |y - B\alpha|^2 + \lambda |D_d \alpha|^2 \\ &= (y - B\alpha)^T (y - B\alpha) + \lambda \alpha^T P \alpha \end{aligned} \tag{30}$$

where

$$P = D_d^T D_d$$

and the elements of B are given by $b_{ij} = B_j (t_i)$, as defined in 17. Taking derivatives on both sides of 30 with respect to α gives

$$\begin{aligned} \frac{\partial}{\partial \alpha} S_\lambda &= \frac{\partial}{\partial \alpha} \left(\alpha^T B^T B \alpha - 2 y^T B^T \alpha + \lambda \alpha^T D_k^T D_k \alpha \right) \\ &= 2 B^T B \alpha - 2 B^T y + 2 \lambda D_d^T D_d \alpha \\ &= \left(B^T B + \lambda D_d^T D_d \right) \alpha - B^T y \end{aligned} \tag{31}$$

and setting equal to zero yields normal equations:

$$B^T y = \left(B^T B + \lambda D_d^T D_k \right) \alpha, \tag{32}$$

which has explicit solution

$$\hat{\alpha} = \left(B^T B + \lambda D_d^T D_d \right)^{-1} B^T y$$

The effective hat matrix is now

$$H_\lambda - B\left(B^TB + \lambda D_k^T D_k\right)^{-1} B^T$$

When $\lambda = 0$, we have the standard normal equations of linear regression with a B-spline basis, and with $k = 0$ 32 corresponds to the normal equations under the ridge regression penalty. When $\lambda > 0$, the penalty only influences the main diagonal and k sub-diagonals of the system of equations. The compact support and limited overlap of the B-spline basis functions gives this system a banded structure, though exploiting this structure is of little utility since the number of equations is equal to the number of splines, which is generally moderate by design.

4.1.1 P-splines for single-index VC models

The derivations in the previous section requiring little adjustment for accommodating a regressor and its corresponding varying slope function, as defined in Equation 18 with $\mu(t) = Q\alpha$, where

$$Q = [B|diag\{x(t)\}\, B]$$

but now B holds a rich B-spline basis with equally-spaced knots. If one wishes to allow for differing degrees of smoothing for each of the varying intercept term and the slope function, the P-spline objective function 30 must be further modified to accommodate multiple tuning parameters, λ_i, $i = 0, 1$. The objective function then becomes

$$\begin{aligned} S_\lambda^* &= |y - Q\alpha|^2 + \lambda_0 |D_{d_0}\alpha_0|^2 + \lambda_1 |D_{d_1}\alpha_1|^2 \\ &= |y - Q\alpha|^2 + |\alpha^T P\alpha|^2 \end{aligned} \tag{33}$$

where the penalty has form $P =$ block diag $\left(\lambda_0 D_{d_0}^T D_{d_0}, \lambda_1 D_{d_1}^T D_{d_1}\right)$. The minimizer of 33 is given by

$$\hat{\alpha} = \left(Q^TQ + P\right)^{-1} Q^T y.$$

The block diagonal structure of the penalty separates the penalization of each individual smooth component. The estimated mean function is then given by

$$\hat{\mu} = Q\hat{\alpha} = Hy$$

where

$$H = Q\left(Q^TQ + P\right)^{-1} Q^T. \tag{34}$$

[Figure 1 Need to explain figure 3 here.]

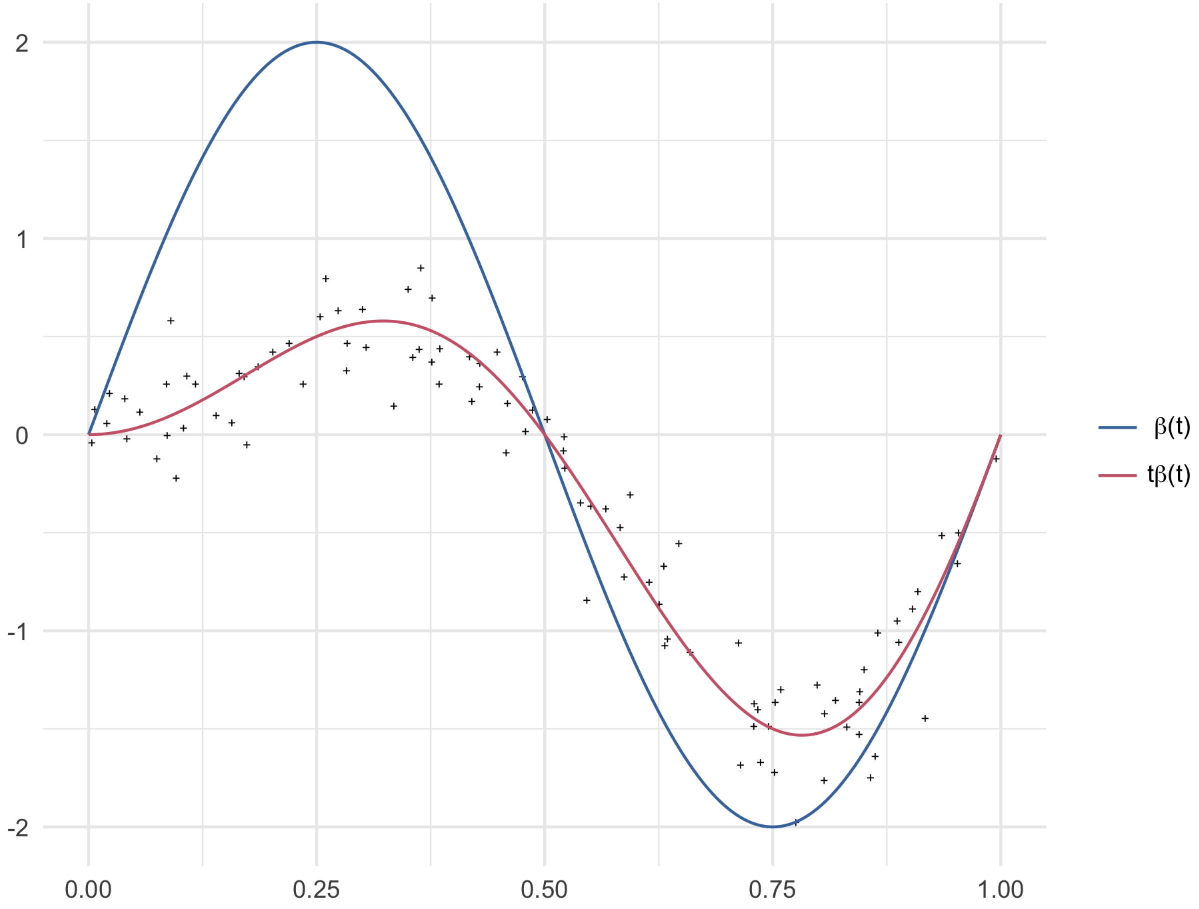
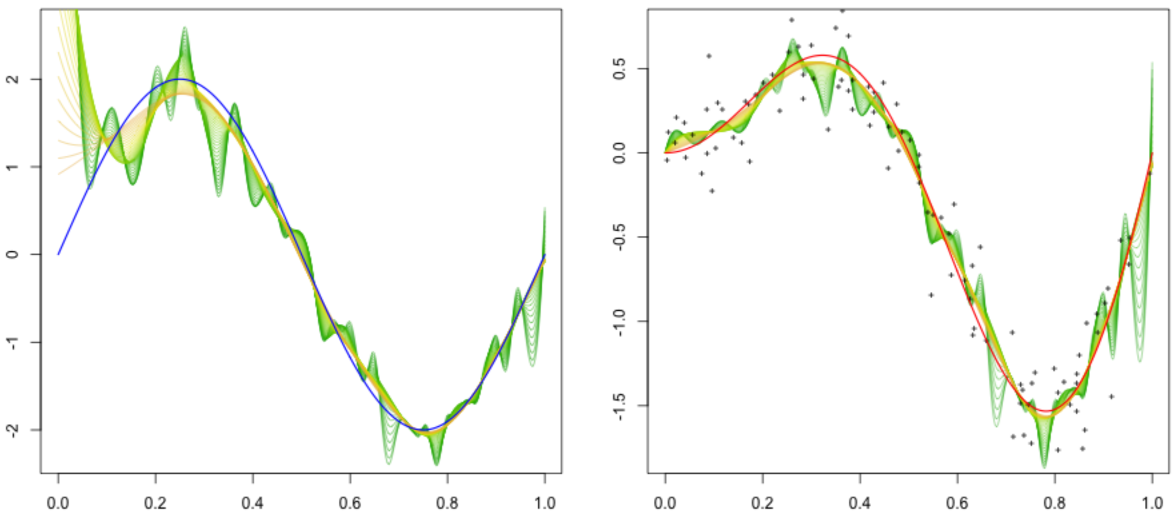


Figure 1: *100 simulated data points where* $y(t) = t\beta(t) + 0.2\epsilon(t)$ *where* ϵ *is a white noise process with unit variance, and* $\beta(t) = 2\sin(2\pi t)$.



(a) $\hat{\beta}\,(t)$

(b) $\hat{\mu}\,(t) = t\hat{\beta}\,(t)$

Figure 2: *Estimated coefficient function* $\hat{\beta}\,(t)$ *and mean curve* $\hat{\mu}\,(t) = t\sin(2\pi t)$ *using a 80 B-splines basis functions of order 5 and a difference penalty of order* $k = 3$.

The properties discussed in Section **??** allude to how controlling the coefficients of a spline $f \in \mathcal{S}_{k,t}$ influences the shape of the overall function. Specifically, the form of the j^{th} derivative provides an avenue of understanding how the differenced B-spline coefficient sequence is related to the volatility of the function on a given interval of its domain. The following figure visually explore the impact of the squared distance on adjacent basis coefficients on the function; a useful way of examining at P-splines is to consider the coefficients as the skeleton of the function, then draping the B-splines over them to put the flesh over the bones. A smoother sequence of coefficients leads to a smoother curve, which is clearly illustrated in Figure 3. As long as the coefficient sequence is smooth, the number of basis functions (and coefficients) is unimportant. The penalty ensures the smoothness of the skeleton.

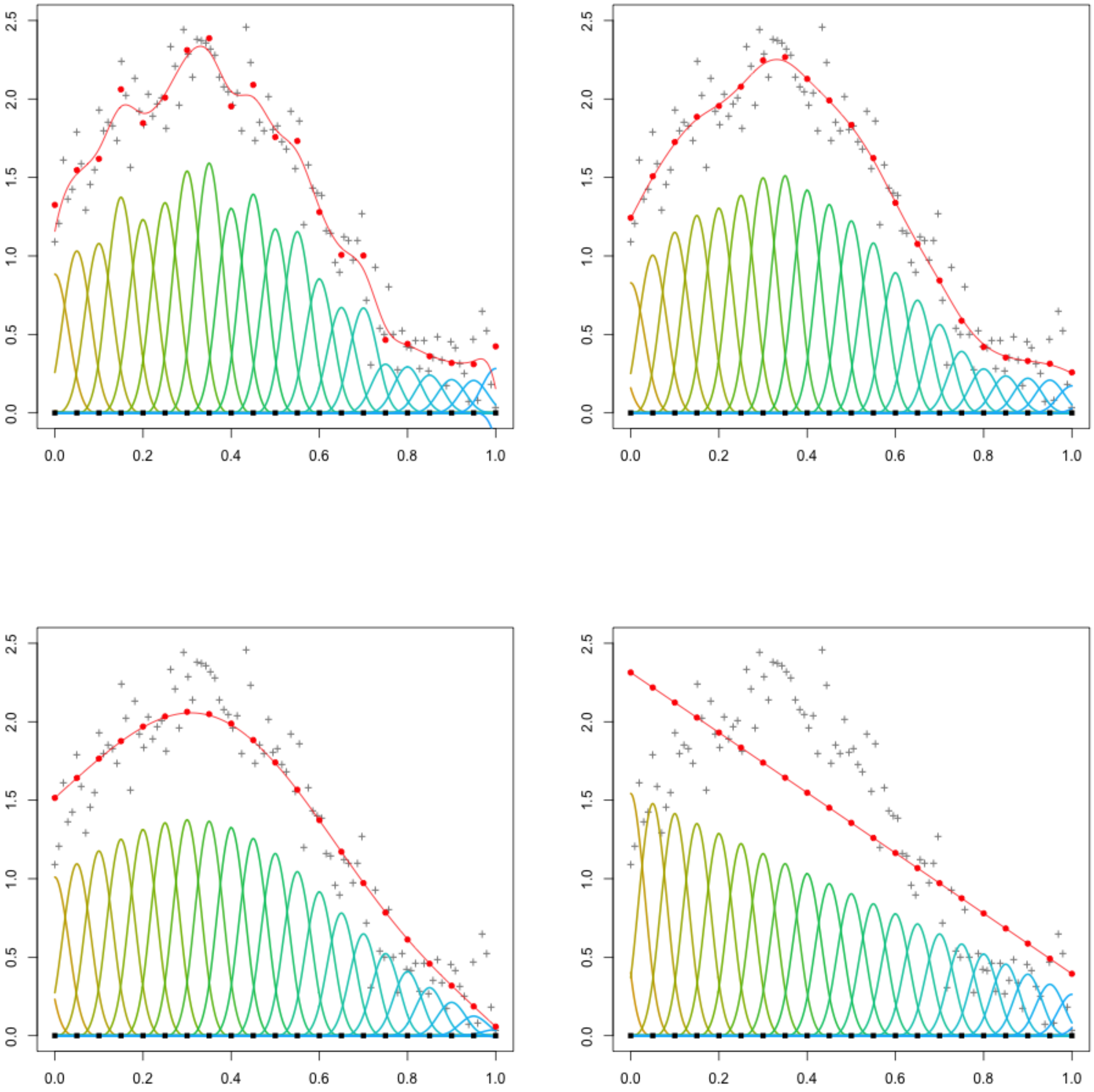


Figure 3: *Illustration of the impact of the second order difference penalty. The number of B-splines used is the same in each plot, with the value of the penalty parameter increasing from left to right and top to bottom across each plot. The fitted curve in the upper left plot is the most "wiggly" of any of the fits, as the penalty plays the weakest roll in the fitted coefficients there. The red circles are the values of each of the B-spline coefficients; as the penalty increases, they form as smoother sequence as we move across the four plots, which results in a smoother fitted function. As the penalty parameter approaches infinity, the fit approaches a linear function as shown in the bottom right plot.*

The number of B-splines can be much larger than the number of observations because penalty ensures that the fitting procedure well-conditioned. One could literally use a thousand splines to fit ten observations without problems. Figure **??** illustrates this utility of the penalty for simulated data. There are $m = 10$ observations and $40 + 3$ cubic B-splines. This property of P-splines cannot be overly appreciated, as it allows us to completely circumvent the nontrivial task of the optimal selection of knot placement. But one simply cannot have too many B-splines. Unless computational constraints are of concern, which is possible with large models, it is prudent to use even more. Figure **??** shows how the fitted function changes as the tuning parameter λ is varied in the presence of sparsely sampled data.

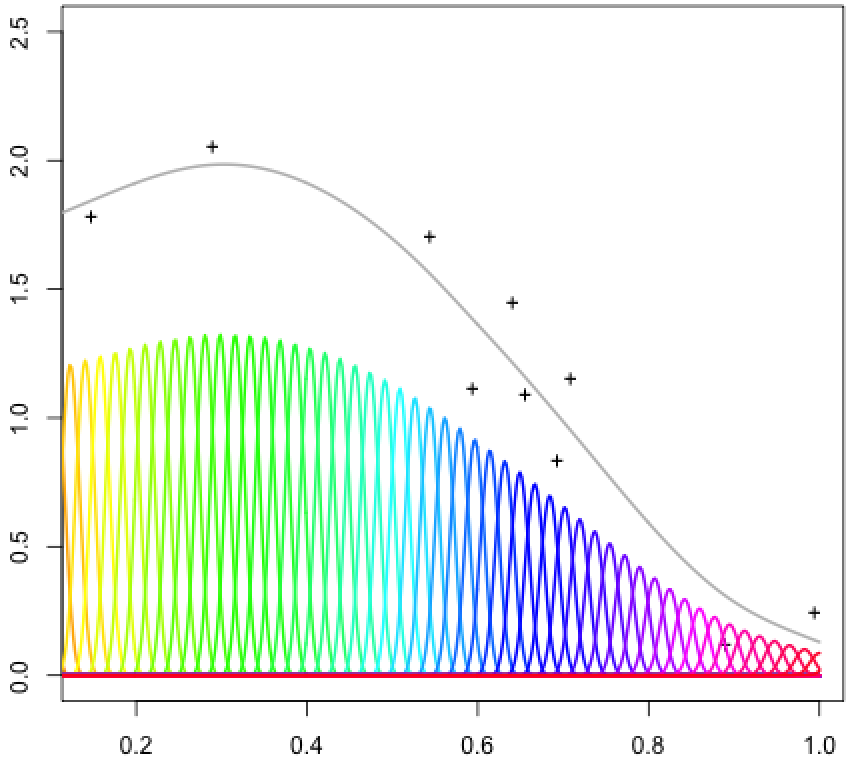


Figure 4: P-spline smoothing of 10 observations using 60 B-spline basis functions.

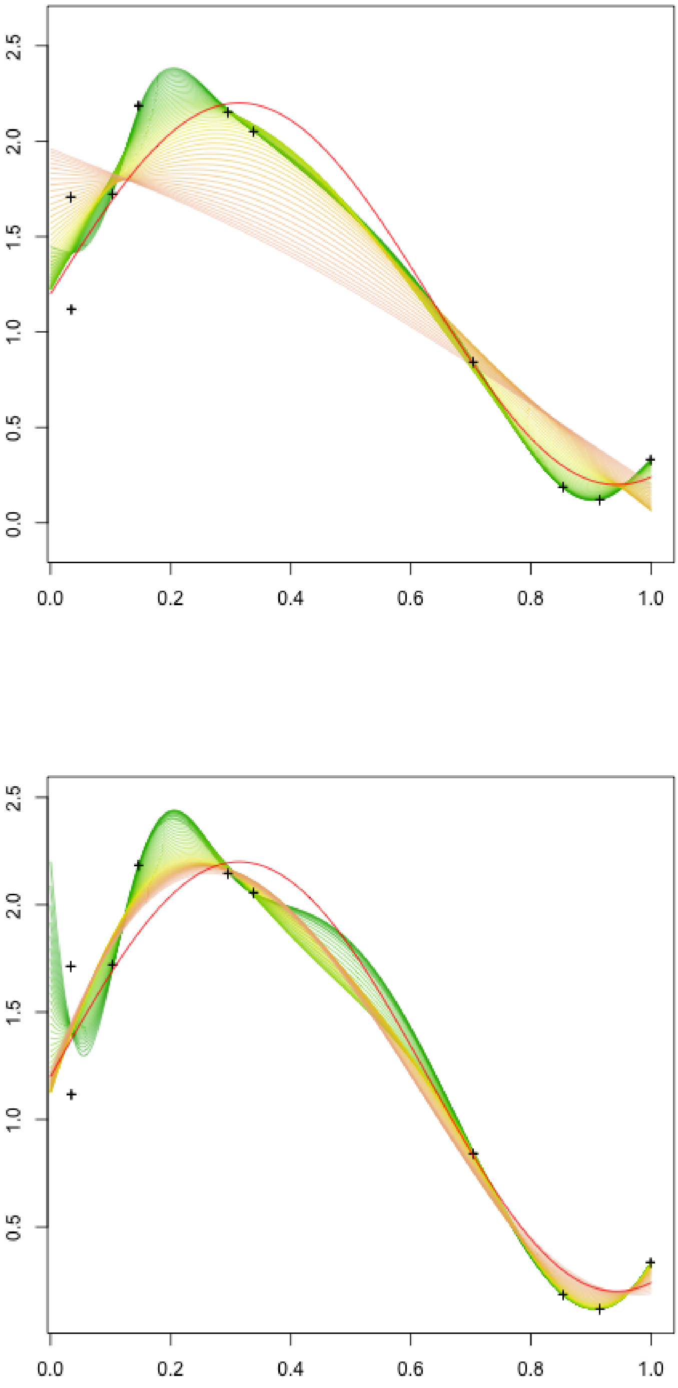


Figure 5: Fitted mean curves using a second (top) and third (bottom) order difference penalty for simulated data, sparsely sampled along the indexing variable: $y\,(t) = 1.2 + \sin{(5t)} + 0.2\epsilon_t$, where $\epsilon_t \overset{i.i.d.}{\sim}$ N $(0, 1)$. A total of 10 data points were fit using a basis of 60 B-splines of degree $k = 3$.

4.2 Properties of P-splines

P-splines enjoy many advantageous properties, many due in part to the inherited properties of the B-spline basis functions on which a generous portion of their foundation is constructed.

I. **Boundary effects** P-splines show no boundary effects, as many types of kernel smoothers do. By this, we mean the spreading of a fitted curve or density outside of the (physical) domain of the data, generally accompanied by bending toward zero.

II. **P-splines fit polynomial data exactly.** P-splines can fit polynomial data exactly. Given data (t_i, y_i), if the y_i are a polynomial in t of degree k, then B-splines of degree k or higher will fit the data exactly.

Proof. This statement is equivalent to the claim that given $\xi = \{\xi_i\}$, $i = 1, \dots, l+1$, and g such that $y(t) = g(t)$, we can find an $f \in \mathcal{P}_{k,\xi} \bigcap \mathcal{C}^{(k-2)}$ which agrees with g at the points $\tau_1 < \cdots < \tau_n$ with $\tau_i \in [\xi_1, \xi_{l+1}]$ for all i, where

$$n = k + l - 1$$

The solution, f is constructed as follows: generate the knot sequence $t = \{t_i\}$ as per the recipe in Theorem **??**:

$$\begin{aligned} t_1 = t_2 = \cdots = t_k &= \xi_1 \\ t_{k+i} &= \xi_{i+1}, & i = 1, \dots, l-1 \\ t_{n+1} = t_{n+2} = \cdots = t_{n+k} &= \xi_{l+1} \end{aligned}$$

Let $\{B_{ik}\}$, $i = 1, \dots, n$ be the corresponding sequence of B-splines of order k, which are a basis for $\mathcal{P}_{k,\xi} \bigcap \mathcal{C}^{(k-2)}$ by Theorem **??**. Here, $\mathcal{P}_{k,\xi} \bigcap \mathcal{C}^{(k-2)}$ denotes the space of pp functions with breakpoints ξ having two continuous (global) derivatives. Then, citeschoenberg1953polya have shown that there exists exactly one $f \in \mathcal{P}_{k,\xi} \bigcap \mathcal{C}^{(k-2)}$ agreeing with g at $\tau_1, \dots, \tau_n$ if and only if

$$B_{ik}(\tau_i) \neq 0, \qquad i = 1, \dots, n.$$

This f has a unique expansion of the form

$$f = \sum_{i=1}^{n} a_i B_{ik}$$

for coefficients $a_i, \dots, a_n$, which are the solution to the linear system

$$\sum_{j=1}^{n} a_j B_{jk}(\tau_i) = g(\tau_i), \qquad i = 1, \dots, n.$$

This system has a banded matrix of coefficients since $B_{jk}(\tau_i) \neq 0$ if and only if $\tau_i \in [t_j, t_{j+k}]$. So if $B_{jk}(\tau_i) \neq 0$ and thus $\tau_i \in (t_j, t_{j+k})$, then there are at most k of the j

indices such that $B_{jk}(\tau_i)$ is nonzero. And further, each of these indices j must be such that

$$(t_i, t_{i+k}) \bigcap (t_j, t_{j+k}) \neq \emptyset,$$

or such that $|i - j| < k$. At worst, the system corresponds to a banded matrix with $k-1$ lower and $k-1$ upper diagonals. □

The same is true for P-splines if the order of the penalty is $k+1$ or higher, irrespective of the value of λ. Consider imposing a first-order difference penalty and a fit to data y that is constant - a polynomial of degree 0. Since

$$\sum_{j=1}^{n} \hat{\alpha}_j B_j(x_i) = c,$$

we have that

$$\sum_{j=1}^{n} \hat{\alpha}_j B_j'(x) = 0,$$

for all x. From the relationship between differences and derivatives in **?? ??**,

$$0 = \sum_{j=1}^{n} B_{j,k}'(x) = \sum_{j=1}^{n} \Delta\alpha_{j+1} B_{j,k-1}(x),$$

so that we must have $\Delta\alpha_j = 0$ for all j, and

$$\sum_{j=2}^{n} \Delta\alpha_j = 0.$$

This shows that the penalty has no impact on the basis coefficients, and the resulting fit is identical to that when using unpenalized B-splines. Using induction, one can show that this is also true when the relationship between x and y is linear and a second order difference penalty is used, and for any values of the polynomial order and order of the difference penalty.

III. **Null models under difference penalties** The limiting P-spline fit approaches a polynomial under strongly enforced smoothing. As $\lambda \to \infty$, under a difference penalty of order d, the fitted function will approach a polynomial of degree $d-1$ as long as the degree of the B-splines is greater than or equal to k. To see this, we again need to use the relationship between the differenced coefficient sequence and the derivative of a B-spline as described in **?? ??**. Consider using the second-order difference penalty; when λ is large, the penalty dominates the P-spline objective function defined in 30, so that the minimizer α must be such that $\sum_{j=3}^{n} \left(\Delta^2 \alpha_j\right)^2$ is close to zero. Consequently, each of the individual second differences must also be nearly zero, and thus the second derivative of the fitted function must be close to zero over the entire domain.

IV. **The limiting behaviour of** H_λ The trace of the hat matrix,

$$H_\lambda = B\left(B^T B + \lambda D_k^T D_k\right)^{-1} B^T y$$

(or for H defined for the addition of a varying slope component as in 34) approaches k, the order of the differencing operator, as λ increases. We index H with the smoothing parameter to indicate that the elements of H are a function of λ. Let

$$Q_B = B^T B \qquad \text{and} \qquad Q_\lambda = \lambda D^T D. \tag{35}$$

Then using properties of the matrix trace, we can write

$$\begin{aligned} \text{tr}\left(H_\lambda\right) &= \text{tr}\left[\left(Q_B + Q_\lambda\right)^{-1} Q_B\right] \\ &= \text{tr}\left[Q_B^{1/2}\left(Q_B + Q_\lambda\right)^{-1} Q_B^{1/2}\right] \\ &= \text{tr}\left[\left(I + Q_B^{-1/2} Q_\lambda Q_B^{-1/2}\right)^{-1}\right] \end{aligned} \tag{36}$$

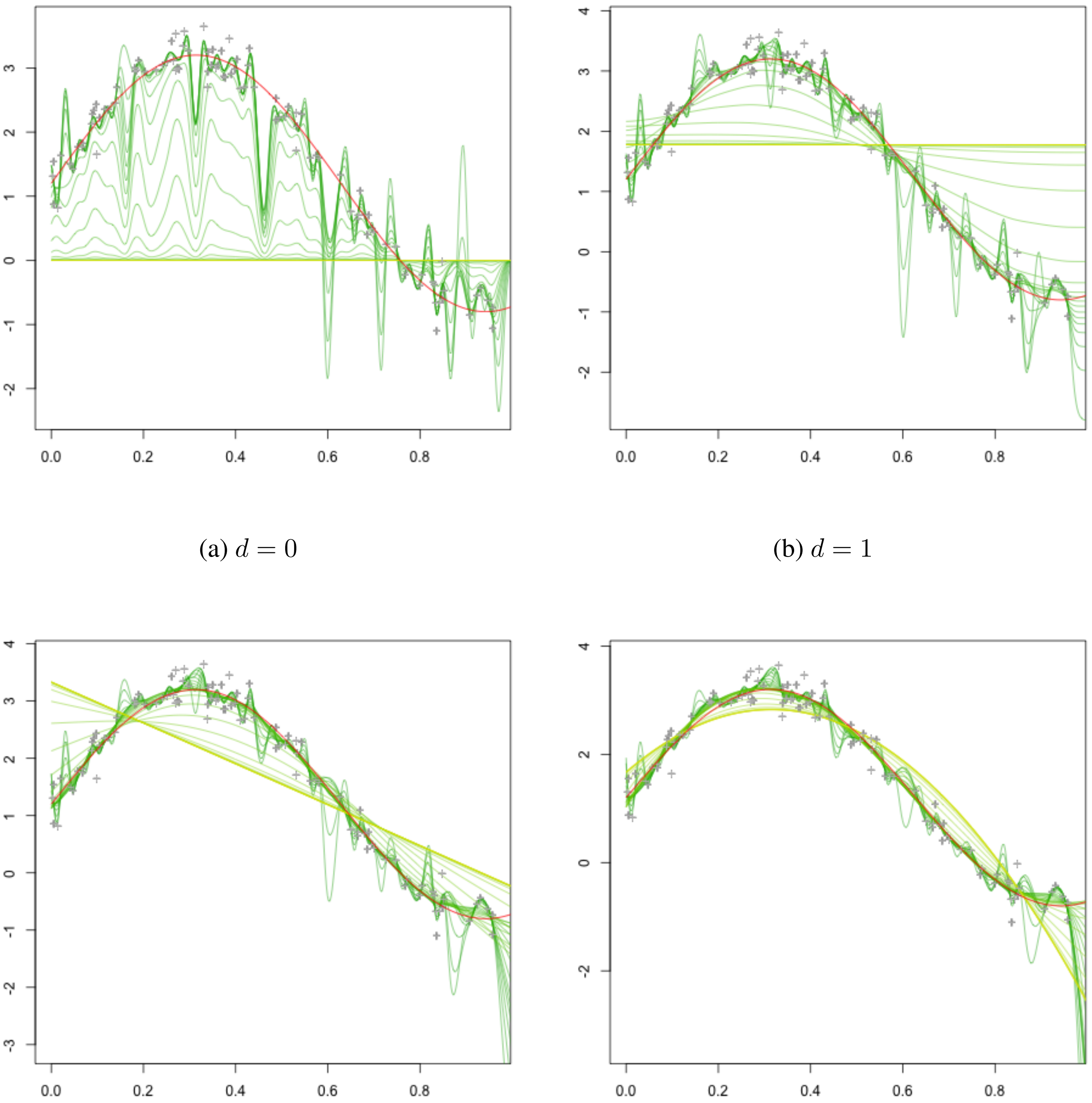
Define $L \equiv Q_B^{-1/2} Q_\lambda Q_B^{-1/2}$. Then

$$\text{tr}\left(H_\lambda\right) = \text{tr}\left[\left(I + \lambda L\right)^{-1}\right] = \sum_{j=1}^{n} \frac{1}{1 + \lambda\gamma_j} \tag{37}$$

where γ_j, $j = 1, \ldots, n$ are the eigenvalues of L. Q_λ has exactly k eigenvalues equal to zero, hence L has k zero eigenvalues. For large λ, only the k terms with $\gamma_j = 0$ contribute to the sum which gives the trace of H, so that

$$\lim_{\lambda \to \infty} \text{tr}\left(H\right) = k.$$

The previous derivations hold regardless of whether we are fitting the varying intercept-only model, with $\mu\left(t\right) = \beta_0\left(t\right)$ or accommodating a varying slope for a regressor by specifying $\mu\left(t\right) = \beta_0\left(t\right) + \beta_1\left(t\right) x\left(t\right)$. The inspection of the hat matrix H is a prelude to the following section, where we will discuss how to use the properties of H to tune the smoothing parameter for optimal model selection. We will later show that extension of these results can be extended in a rather straightforward manner to the case that is of our particular interest: when the smooth slope function is a two-dimensional surface rather than a curve.



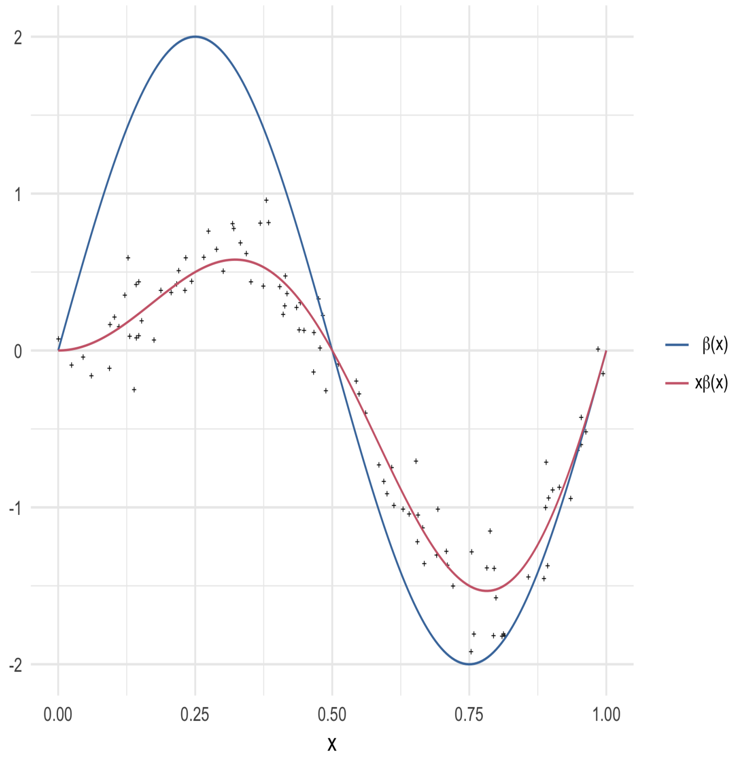
(a) $d = 0$

(b) $d = 1$

(c) $d = 2$

(d) $d = 3$

Figure 6: *Illustration of the impact of the order of the difference penalty. The number of B-splines used is the same in each plot, with the penalty parameter varying from across the same grid of values. The fitted curves in the upper left plot correspond to the difference penalty of order* 0*, where* $|D_0\alpha|^2 = \sum_{i=1}^{n} \alpha_i^2$*, analogous to ridge regression using the B-spline basis as regression covariates. The fitted curves approach polynomials of degree* $d-1$ *as* $\lambda \to \infty$*, as discussed in 4.2 III.*



5 Estimating ϕ via penalized maximum likelihood

We employ maximum likelihood for the estimation of the varying coefficient function $\phi\left(t,s\right)$ and the innovation variance function $\sigma\left(t\right)$, though neither the derivation the form of model **??** nor model 1 via the Cholesky decomposition rely on any assumptions about the distribution of Y.

For fixed $\left\{\sigma_j^2\right\}$, as a function of ϕ_{jk} the negative log-likelihood for a sample of N i.i.d. observations $Y_1, Y_2, \ldots, Y_N$ from a multivariate Gaussian distribution is proportional to the usual error sums of squares:

$$-2L\left(y_1,\ldots,y_N,\phi^*|\boldsymbol{\sigma^2}\right) \propto \sum_{i=1}^{N}\sum_{j=2}^{m_i}\sigma\left(t_j\right)^{-2}\left(y_{ij}-\sum_{k=1}^{j-1}\phi\left(t_{ij},t_{ik}\right)y_{ik}\right)^2 \qquad (38)$$

where

$$y_i = \left(y_{i1}, y_{i2}, \ldots, y_{i,m_i}\right), \quad i = 1,\ldots,N$$

denotes the vector of observations for subject i with corresponding measurement times

$$t_{i1} < t_{i2} < \cdots < t_{i,m_i}.$$

The form of the likelihood of $y_1,\ldots,y_N$ indicates that we allow both the number of measurements as well as the observation times to varying across subjects. The $\{t_{ij}\}$ need not be evenly-spaced within or across individuals. Denote the innovation variance function evaluated at the vector of observed time points by $\boldsymbol{\sigma^2}$, and similarly let $\boldsymbol{\phi}^*$ denote the resulting vector when evaluating ϕ^* at the observed grid of time points, transformed to the l-m axis. Estimation of the varying coefficient function and the innovation variance function may be accomplished in an iterative fashion:

I. Fix $\boldsymbol{\sigma^2} = \boldsymbol{\sigma^2}_{0}$;

II. find $\widehat{\boldsymbol{\phi}^*} = \underset{\boldsymbol{\phi}^*}{\arg\max} - 2L\left(y_1, \ldots, y_N, \boldsymbol{\phi}^* | \boldsymbol{\sigma^2}_0\right)$.

III. Fix $\boldsymbol{\phi}^* = \widehat{\boldsymbol{\phi}^*}$;

IV. find $\widehat{\boldsymbol{\sigma^2}} = \underset{\boldsymbol{\sigma^2}}{\arg\max} - 2L\left(y_1, \ldots, y_N, \boldsymbol{\sigma^2} | \widehat{\boldsymbol{\phi}^*}\right)$.

V. Iterate until convergence.

For ease of exposition, we first focus our attention on the estimation of ϕ assume that $\sigma^2\left(t\right)$ is fixed and known. Estimation of the innovation variance function is presented in Section **??**. In the case that subjects share a common set of observation times $t_1 < \cdots < t_m$, it is well known that the MLE for Σ, $S = \sum_{i=1}^N y_i y_i'$ is highly unstable in high dimensions, a condition that is potentially worsened when one or more subjects has at least one observation time that is unique from the set of observation times common across subjects. To mitigate instability due to high dimensionality and simultaneously permit the estimation of $\phi\left(\cdot,\cdot\right)$ as a smooth bivariate function, we obtain a covariance estimator by applying bivariate smoothing of the elements of the Cholesky factor.

Estimating the varying coefficient function ϕ^*, however, is quite different from the usual problem of estimating an arbitrary bivariate function. In the case of the latter, we most typically treat both arguments equally in terms of regularization, but in the case of covariance estimation and the generalized coefficient function equal treatment of l and m in terms of penalization perhaps is not the most appropriate approach. The lag component, l, has particularly significant meaning in terms of the covariance function and thus also in terms of ϕ^* and is of considerable more interest than the orthogonal component, m. We parameterize ϕ in terms of the transformed domain:

$$l = t - s, \qquad m = \frac{1}{2}\left(s + t\right),$$

so that the following relationship holds:

$$\phi\left(s,t\right) = \phi^*\left(s - t, \frac{1}{2}\left(s+t\right)\right) = \phi^*\left(l, m\right)$$

with

$$\frac{l}{2} < m < 1 - \frac{l}{2}, \quad 0 < l < 1. \tag{39}$$

The likelihood can be written in terms of the reparameterized varying coefficient function:

$$\begin{aligned}
-2L\left(y_1, \ldots, y_N, \phi^* | \boldsymbol{\sigma^2}\right) &= \sum_{i=1}^{n} \sum_{j=2}^{m_i} \sigma_{ij}^{-2} \left(y_{ij} - \sum_{k=1}^{j-1} \phi\left(t_{ij}, t_{ik}\right) y_{ik} \right)^2 \\
&= \sum_{i=1}^{n} \sum_{j=2}^{m_i} \sigma_{ij}^{-2} \left(y_{ij} - \sum_{k=1}^{j-1} \phi^*\left(l_{ijk}, m_{ijk}\right) y_{ik} \right)^2
\end{aligned} \tag{40}$$

We equip the l and m axes each with a B-spline basis to construct a basis for the varying coefficient function ϕ in **??** by taking the tensor product of the two marginal bases. Let

$$B_1\left(l\right), \ldots, B_K\left(l\right) \text{ and } B_1\left(m\right), \ldots, B_L\left(m\right)$$

denote the B-spline bases for l and m, each having a set of equally spaced knots along their respective domain. It is worth noting that while we have chosen not to distinguish between $\{B_k\}$ and $\{B_l\}$ for the sake of brevity, one is free to specify a different basis for each dimension either by using different order B-spline or, of course, using different numbers of knots, and hence entirely different knot sequences since P-splines rely on bases with equally spaced knots. The tensor product basis functions

$$T_{jk}\left(l, m\right) = B_j\left(l\right) B_k\left(m\right)$$

carve the l-m domain into rectangles. Figure 8 shows a thinned tensor product basis $\{T_{kl}\}$; a portion of the basis was omitted to eliminate overlapping of the basis functions so that the reader can identify individual tensor products. Each "hill" in Figure 8 is associated with an unknown coefficient θ_{ij} which determines the height of the hill. For a given knot grid, we can approximate a surface by

$$\phi^*\left(l, m\right) = \sum_{i=1}^{K} \sum_{j=1}^{L} \theta_{ij} B_i\left(l\right) B_j\left(m\right), \tag{41}$$

and the function evaluated at the observed $\left(l_{ijk}, m_{ijk}\right)$ may be written

$$\boldsymbol{\phi^*} = B_m \Theta B_l'$$

where Θ denotes the $K \times L$ matrix of tensor product coefficients, with elements θ_{ij}.

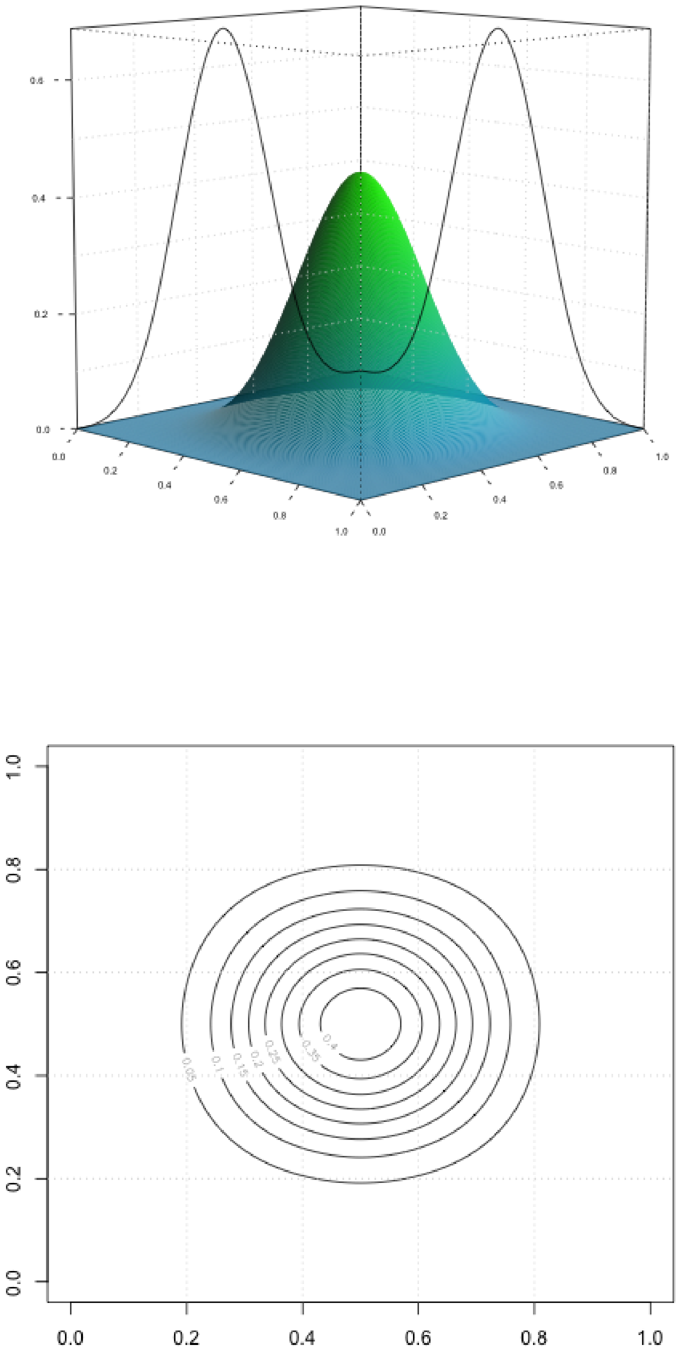


Figure 7: Tensor product of two cubic B-splines

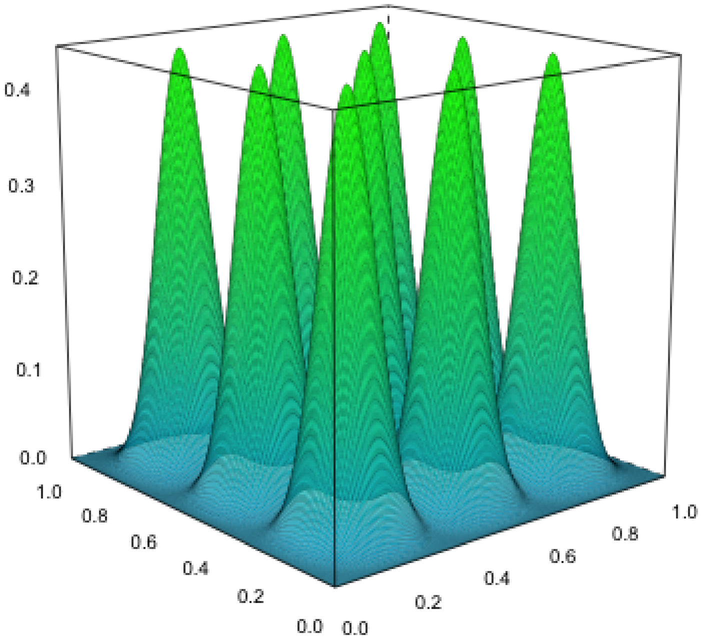


Figure 8: A subset of a full bivariate basis of cubic B-splines

5.1 Regularization with difference penalties

The minimizer of **??** honors the fidelity to the data, so to balance the complexity of the fitted function with the goodness of fit to the data, we can append a penalty to the negative log likelihood to control the fitted function. By using rich B-spline bases for l and m alongside discrete difference penalties on the spline coefficients, we can achieve as much smoothness of the fitted function in both the l and m dimensions as desired. **?** was the first to propose using a rich B-spline basis and using a penalty to restrict the flexibility of the fitted curve, like **?** applying a penalty to the second derivative of the fitted curve:

$$J = \int_0^1 \left[f''\left(l\right)\right]^2 \, dx.$$

For a B-spline of the form

$$f\left(x\right) = \sum_{j=1}^{n} \theta_i B_j\left(x\right),$$

one can derive a banded matrix P using the properties of B-splines such that

$$J = \theta' P \theta$$

where $\theta = (\theta_1, \ldots, \theta_n)$. The i-j^{th} element of P is given by

$$p_{ij} = \int_0^1 B_i''(x) \, B_j''(x) \; dx.$$

In some applications, it is useful to work with third and fourth order differences, since for large values of λ, the fitted curve approaches a parametric polynomial model. This may be of particular interest in the context of estimating the elements of the Cholesky factor, as many have proposed simple parametric functions of lag only for ϕ^*, such as low order polynomials. See **?**. However, with the use of higher order derivatives, the computation of P is nontrivial and becomes very tedious. **?** were the first to propose P-splines, or *penalized B-splines*, as an approach to non-parametric regression. P-splines circumvent complexity associated with constructing such penalty matrices by omitting derivatives and integrals altogether, replacing them with finite differences and sums.

Instead, flexibility of the fitted function is controlled by using a discrete penalty matrix based on finite difference formulas. Smoothness of the fitted function is achieved by first using a rich B-spline basis with equally spaced knots to purposefully overfit the smooth coefficient vectors; this eliminates the difficulty of choosing the optimal set of knots. Then by attaching a difference penalty to the goodness of fit measure, one may prevent overfitting and make a potentially ill-conditioned fitting procedure a well-conditioned one. The finite difference penalty is simple to compute and can be handled mechanically for any order of the differences. Since it is easily introduced into regression equations, it is feasible to evaluate the impact of different orders of the differences on the fitted model. Using the properties of B-splines, it is straightforward to show that the difference penalty of order d is a good discrete approximation to the integrated square of the d^{th} derivative, so little is lost by replacing the derivative-based penalty with

$$J_d(f) = \sum_{j=d}^{n} \left(\Delta^d \theta_j \right)^2 \tag{42}$$

where $\theta = (\theta_1, \ldots, \theta_n)$. Let D_d denote the matrix difference operator: $D_d \theta = \Delta^d \theta$, where

$$\Delta \theta_j = \theta_j - \theta_{j-1}, \quad \Delta^2 \theta_j = \Delta \left(\Delta \theta_j \right) = \theta_j - 2\theta_{j-1} + \theta_{j-2}$$

In general,

$$\Delta^d \theta_j = \Delta \left(\Delta^{d-1} \theta_j \right).$$

Then, 42 can be written in terms of the squared norm of the difference operator applied to the vector of B-spline coefficients:

$$\begin{aligned} J_d(f) &= ||D_d \theta||^2 \\ &= \theta' P_d \theta \end{aligned} \tag{43}$$

where $P_d = D_d' D_d$. To examine the connection between the second-derivative penalty to the penalty on second-order differences of the B-spline coefficients, we only need to employ straightforward calculus and exploit the recursive property of the B-spline basis functions:

$$\int_0^1 \left[f''(x)\right]^2 \, dx = \int_0^1 \left\{\sum_{j=1}^{n} \theta_j B_{j,3}''(l)\right\}^2 \, dl.$$

The derivative properties of B-splines permits this to be written as

$$\int_0^1 \left[f''(x)\right]^2 \, dx = \int_0^1 \left[\sum_{j=1}^{n}\sum_{k=1}^{n} \Delta^2\theta_j \Delta^2\theta_k B_{j,1}(l) \, B_{k,1}(l)\right] dl.$$

Most of the cross products of $B_{j,1}(x)$ and $B_{k,1}(x)$ vanish since B-splines of degree 1 only overlap when j is $k-1$, k, or $k+1$. Thus, we have that

$$\begin{aligned}\int_0^1 \left[f''(x)\right]^2 \, dx &= \int_0^1 \left[\left\{\sum_{j=1}^{n} \Delta^2\theta_j B_j(l,1)\right\}^2 + 2\sum_{j} \Delta^2\theta_j \Delta^2\theta_{j-1} B_j(l,1) \, B_{j-1}(l,1)\right] dl \\ &= \sum_{j=1}^{n} \left(\Delta^2\theta_j\right)^2 \int_0^1 B_j^2(l,1) \, dl \\ &\qquad + 2\sum_{j=1}^{n} \Delta^2\theta_j \Delta^2\theta_{j-1} \int_0^1 B_j(l,1) \, B_{j-1}(l,1) \, dl\end{aligned} \tag{44}$$

which can be written as

$$\int_0^1 \left[f''(x)\right]^2 \, dx = c_1 \sum_{j=2}^{n} \left(\Delta^2\theta_j\right)^2 + c_2 \sum_{j=3}^{n} \Delta^2\theta_j \Delta^2\theta_{j-1} \tag{45}$$

Given a set of equidistant knots, the constants c_1 and c_2 are given by

$$\begin{aligned} c_1 &= \int_0^1 B_{j,1}^2(x) \, dx \\ c_2 &= \int_0^1 B_{j,1}(x) \, B_{j-1,1}(x) \, dx. \end{aligned} \tag{46}$$

This gives us that the traditional smoothness penalty on the squared second derivative can be written as a linear combination of a penalty on the second-order differences of the B-spline coefficients 42 and the sum of the cross products of neighboring second differences. The second term in 45 leads to a complex objective function when minimizing the penalized likelihood, where seven adjacent spline coefficients occur, as opposed to five if only the first term in 45 is used in the penalty. The added complexity is a consequence of overlapping B-splines, which quickly increases when using higher order differences and higher order B-splines. We can seamlessly augment the likelihood with the difference penalty to achieve smooth fitted functions without the complexity posed by the derivative-based penalty.

A smoother sequence of coefficients leads to a smoother curve, as illustrated in Figure 3. The relationship between P-spline curves and their coefficients is easily characterized if we consider the coefficients as the skeleton of the function, and draping the B-splines over them puts the flesh on the bones. As long as the coefficient sequence is smooth, the number of basis functions (and coefficients) is unimportant since the penalty ensures the smoothness of the skeleton and that the fitting procedure is well-conditioned. Figure **??** illustrates this utility of the penalty for simulated data; there are $m = 10$ observations and 60 cubic B-splines. This property of P-splines cannot be overly appreciated because it frees us from the concern of choosing the optimal set of knots. Unless computational constraints are of concern, which is possible with large models, it is prudent to use even more B-splines. Figure **??** shows how the fitted function changes as the tuning parameter varies when the data are sparsely sampled. P-splines enjoy a number of additional advantageous properties, many of which are inherited from the attractive properties of B-splines. See **?** for a detailed presentation.

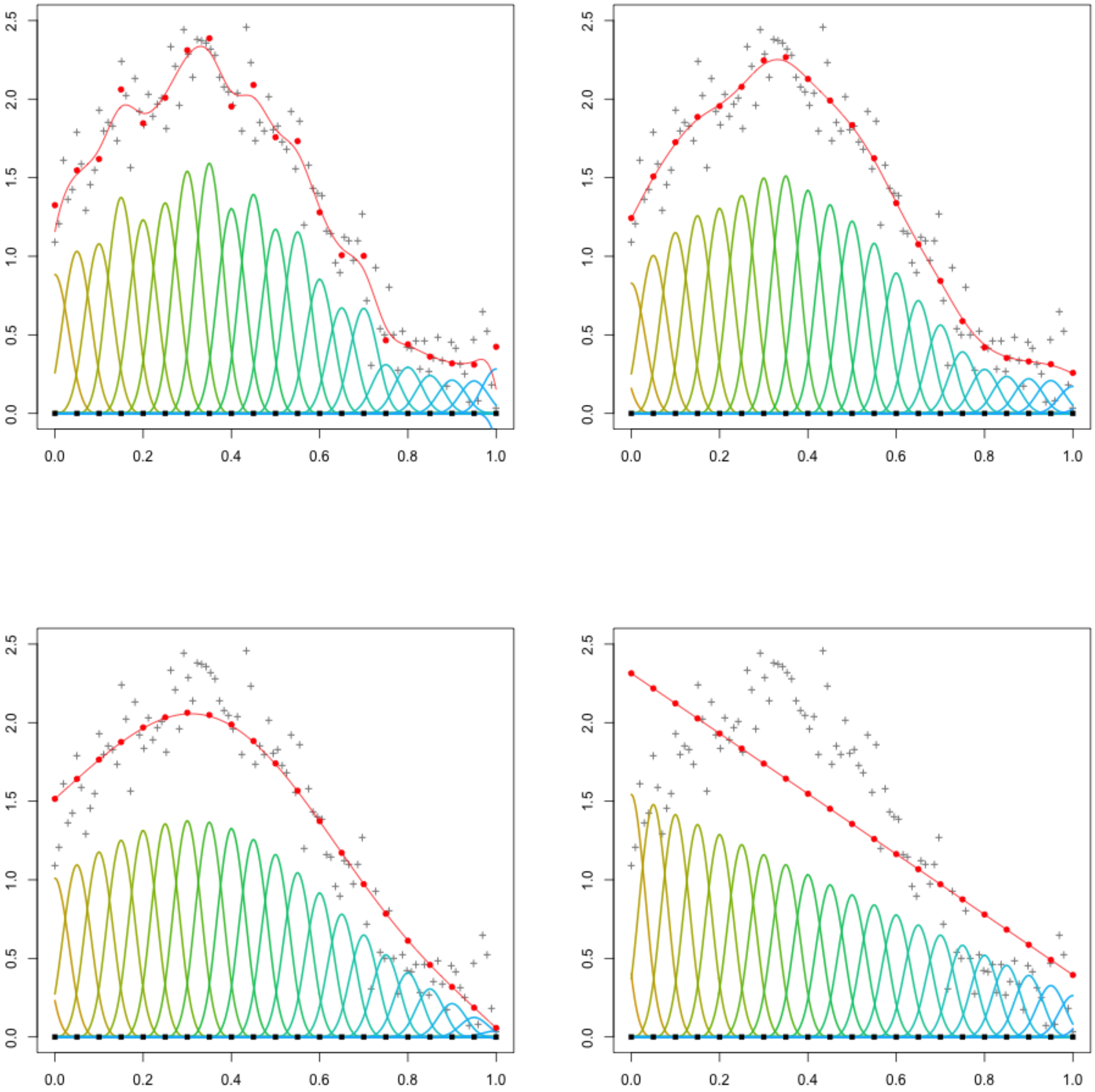
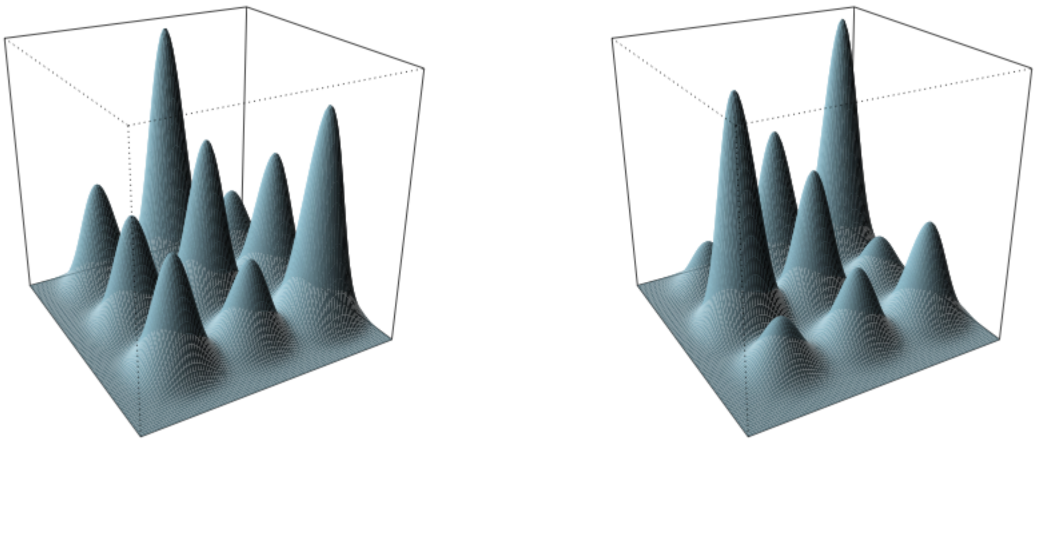


Figure 9: *Illustration of the impact of the second order difference penalty. The number of B-splines used is the same in each plot, with the value of the penalty parameter increasing from left to right and top to bottom across each plot. The fitted curve in the upper left plot is the most "wiggly" of any of the fits, as the penalty plays the weakest roll in the fitted coefficients there. The red circles are the values of each of the B-spline coefficients; as the penalty increases, they form as smoother sequence as we move across the four plots, which results in a smoother fitted function. As the penalty parameter approaches infinity, the fit approaches a linear function as shown in the bottom right plot.*

To extend these results to the bivariate setting for regularizing of ϕ^*, the only modification to the differencing procedure in one dimension necessary is the addition of a second difference penalty, one for each variable l and m. We append the pair of penalties to the negative log likelihood **??** and take the estimator of ϕ^* to correspond to the B-spline coefficients minimizing

$$\begin{aligned}-2L + J\left(\phi^*\right) = \sum_{i=1}^{N}\sum_{j=2}^{m_i} \sigma\left(t_j\right)^{-2}\left\{y_{ij} - \sum_{k=1}^{j-1}\left(\sum_{r=1}^{L}\sum_{s=1}^{K}\theta_{rs}B_r\left(l_{ijk}\right)B_s\left(m_{ijk}\right)\right)y_{ik}\right\}^2 \\ +\lambda_l\sum_{r'=1}^{K}|D_{d_l}\theta_{r'\cdot}|^2 + \lambda_m\sum_{s'=1}^{L}|D_{d_m}\theta_{\cdot s'}|^2.\end{aligned} \tag{47}$$

where $\theta_{k\cdot}$ and $\theta_{\cdot l}$ denote the k^{th} row and l^{th} column of Θ, respectively. The first term in**??** imposes a difference penalty of order d_l on the rows of the coefficient matrix while the second term places a difference penalty (of possible different order d_m) on the columns. We give each direction its own smoothing parameter to permit anisotropic smoothing; however, one could opt to use a single smoothing parameter for both directions and dodge the added work of optimizing the amount of smoothing with two separate parameters. Figure **??** shows a potential result of heavy column penalization (left) and heavy row penalization (right) under a second order difference penalty on each row and each column for large values of λ_l and λ_m. The figure demonstrates that the limiting behaviour of each row and column is linear, but the resulting surface may exhibit slope reversals from one row (column) to the next.



(a) *heavy column penalty*

(b) *heavy row penalty*

Figure 10: *Nine cubic B-spline tensor products with heavy linear column penalty and heavy linear row penalty*

We take the estimator of ϕ^* to be the minimizer of It is computationally advantageous to express the coefficient matrix in "unfolded" notation, which allows us to express the varying coefficient function at the observed coordinate grid as in the usual multiple regression form:

$$\text{vec}\left\{\phi^*\left(l,m\right)\right\} = B\theta$$

Stacking the columns of Θ gives the vectorized coefficient matrix $\theta = \text{vec}\left(\Theta\right)$. The $p \times KL$ tensor product basis B is constructed from the tensor product of the marginal B-spline bases defined in **?** as the *row-wise Kronecker product* of the individual bases:

$$B = B_l \Box B_m = \left(B_m \otimes 1_K'\right) \odot \left(1_L' \otimes B_l\right). \tag{48}$$

The operator $\odot$ denotes the element-wise matrix product; 1_K (1_L) denotes the column vector of ones having length K $(L.)$ The operations in 48 construct B such that the i^{th} row of $B_m \Box B_l$ is the Kronecker product of the corresponding rows of B_m and B_l. The penalty in **??** can also be compactly expressed:

$$\lambda_l ||P_l\theta||^2 + \lambda_m ||P_m\theta||^2$$

where $P_l = I_L \otimes D_{d_l}' D_{d_l}$ and $P_m = D_{dm}' D_{dm} \otimes I_K$. We define the matrix W of historical regressors so that 47 can be written in matrix form as

$$-2L + J\left(\phi^*\right) = \left(Y - WB\theta\right)' D^{-1} \left(Y - WB\theta\right) + \lambda_l ||P_l\theta||^2 + \lambda_m ||P_m\theta||^2, \tag{49}$$

with $\hat{\theta}$ solving the system of equations

$$\left[\left(WB\right)' D^{-1} WB + \lambda_l P_l + \lambda_m P_m\right] \theta = \left(WB\right)' D^{-1} Y \tag{50}$$

From 50, we note that the system of equations depends on basis coefficients remains fixed at KL, even as the number of observations increases. The grid of regression coefficients can be recovered by arranging the elements of $\hat{\theta}$ into a matrix of L columns having length K.

This recipe for constructing a tensor product basis for ϕ^* is an easy and convenient way to construct a two-dimensional basis for a bivariate function with domain corresponding to the unit square. However, the domain of the autoregressive coefficient function, specified in 39, lies on the lower triangle of the unit square:

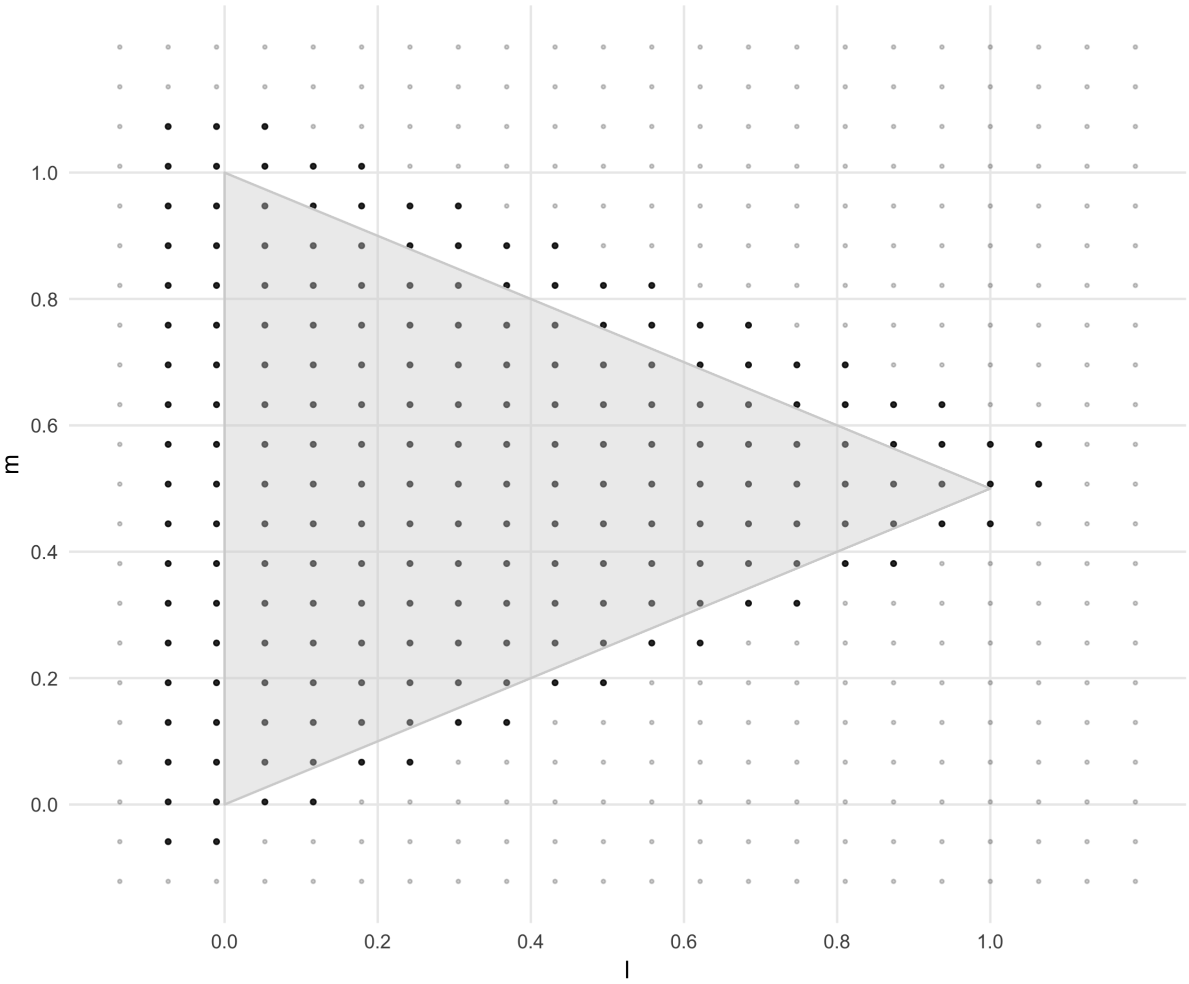


Figure 11: $\frac{l}{2} < m < 1 - \frac{l}{2}, \quad 0 < l < 1.$

The triangular domain of ϕ^* cannot be modeled by the tensor product basis as constructed due to singularities resulting from the large number of basis functions anchored at knots near which we have no data, and hence little information about the corresponding basis coefficient. Much work in computer graphics has been done proposing methods of smoothing over arbitrary function domains, which are approximated by triangulations. See **?** and **?** for details. These methods are, however, quite computationally intensive; to correct for the instability in the smoothed surface, we can simply remove the knots corresponding to tensor products functions which do not overlap with the function domain from the basis, B, and trimming the penalty matrices P_l and P_m as needed. With the trimmed basis and penalties, we can carry out optimization as previously discussed.

6 Model selection and tuning parameter estimation

6.1 The limiting behaviour of H_λ

The inspection of the hat matrix

$$H_\lambda = WB\left(WB'WB + \lambda_l P_l + \lambda_m P_m\right)^{-1}(WB)'\,D^{-1}.$$

and its properties are integral for assessing model complexity and selecting the optimal values of the tuning parameters λ_l and λ_m. Summarizing the complexity of a fitted P-spline is far from a trivial task; one must simultaneously consider the value of the smoothing parameter, the number of basis functions in the B-spline basis, as well as the order of the difference penalties. We follow **?** and**?** assess model complexity as discussed in citehastie1990generalized, who proposed to use the trace of the smoother matrix as an approximation to the effective dimensions of linear smoother. The *effective dimension* is easily obtained and combines the effect of all three of these elements:

$$\begin{aligned}\text{ED} &= \text{tr}\left[H_\lambda\right] \\ &= \text{tr}\Bigg[\,\left[WB\,(WB)'\,D^{-1}WB + \lambda_l P_l + \lambda_m P_m\right]^{-1}(WB)'\,D^{-1}\Bigg]\end{aligned} \tag{51}$$

When the number of basis functions is significantly smaller than the sample size, it is computationally advantageous to use the cyclic property of the trace:

$$\text{tr}\Bigg[\,\left[(WB)'\,D^{-1}WB + \lambda_l P_l + \lambda_m P_m\right]^{-1}(WB)'\,D^{-1}WB\Bigg],$$

which requires computing the trace of a $KL \times KL$ matrix. The effective dimension approaches $d_l + d_m$, the order of the differencing operator, as λ increases, where d_l and d_m denote the orders of the difference penalties in the l and m directions, respectively. Let

$$Q = (WB)'\,D^{-1}WB \qquad \text{and} \qquad Q_\lambda = P.$$

Using properties of the matrix trace, we can write

$$\begin{aligned}\text{tr}\left(H_\lambda\right) &= \text{tr}\Bigg[\,\left(Q + Q_\lambda\right)^{-1}Q\Bigg] \\ &= \text{tr}\Bigg[Q^{1/2}\left(Q + Q_\lambda\right)^{-1}Q^{1/2}\Bigg] \\ &= \text{tr}\Bigg[\,\left(I + Q^{-1/2}Q_\lambda Q^{-1/2}\right)^{-1}\,\Bigg]\end{aligned}$$

Define $L \equiv Q^{-1/2}Q_\lambda Q^{-1/2}$. Then

$$\text{tr}\left(H_\lambda\right) = \text{tr}\Bigg[\,(I + \lambda L)^{-1}\,\Bigg] = \sum_{j=1}^{n}\frac{1}{1+\lambda\gamma_j}$$

where γ_j, $j = 1, \ldots, n$ are the eigenvalues of L. Q_λ has exactly $d_l + d_m$ eigenvalues equal to zero. Hence, L has $d_l + d_m$ zero eigenvalues. For large λ, only the $d_l + d_m$ terms with $\gamma_j = 0$ contribute to the sum which gives the trace of H, so that

$$\lim_{\lambda \to \infty} \text{tr}\,(H) = d_l + d_m.$$