Electronic Journal of Statistics
Vol. 5 (2011) 1654–1717
ISSN: 1935-7524
DOI: 10.1214/11-EJS652

Penalized wavelets: Embedding wavelets into semiparametric regression

M.P. Wand

School of Mathematical Sciences,
University of Technology, Sydney, Broadway, 2007, Australia
e-mail: matt.wand@uts.edu.au

and

J.T. Ormerod

School of Mathematics and Statistics,
University of Sydney, Sydney, 2006, Australia
e-mail: jormerod@sydney.edu.au

Abstract: We introduce the concept of *penalized wavelets* to facilitate seamless embedding of wavelets into semiparametric regression models. In particular, we show that penalized wavelets are analogous to penalized splines; the latter being the established approach to function estimation in semiparametric regression. They differ only in the type of penalization that is appropriate. This fact is not borne out by the existing wavelet literature, where the regression modelling and fitting issues are overshadowed by computational issues such as efficiency gains afforded by the Discrete Wavelet Transform and partially obscured by a tendency to work in the wavelet coefficient space. With penalized wavelet structure in place, we then show that fitting and inference can be achieved via the same general approaches used for penalized splines: penalized least squares, maximum likelihood and best prediction within a frequentist mixed model framework, and Markov chain Monte Carlo and mean field variational Bayes within a Bayesian framework. Penalized wavelets are also shown have a close relationship with *wide data* ("$p \gg n$") regression and benefit from ongoing research on that topic.

Keywords and phrases: Bayesian inference, best prediction, generalized additive models, Gibbs sampling, maximum likelihood estimation, Markov chain Monte Carlo, mean field variational Bayes, sparseness-inducing penalty, wide data regression.

Received September 2011.

1. Introduction

Almost two decades have passed since wavelets made their debut in the statistics literature (Kerkycharian and Picard [40]). Articles that use wavelets in statistical problems now number in the thousands. A high proportion of this literature is concerned with the important statistical problem of nonparametric regression which, in turn, is a special case of semiparametric regression (e.g. Rup-

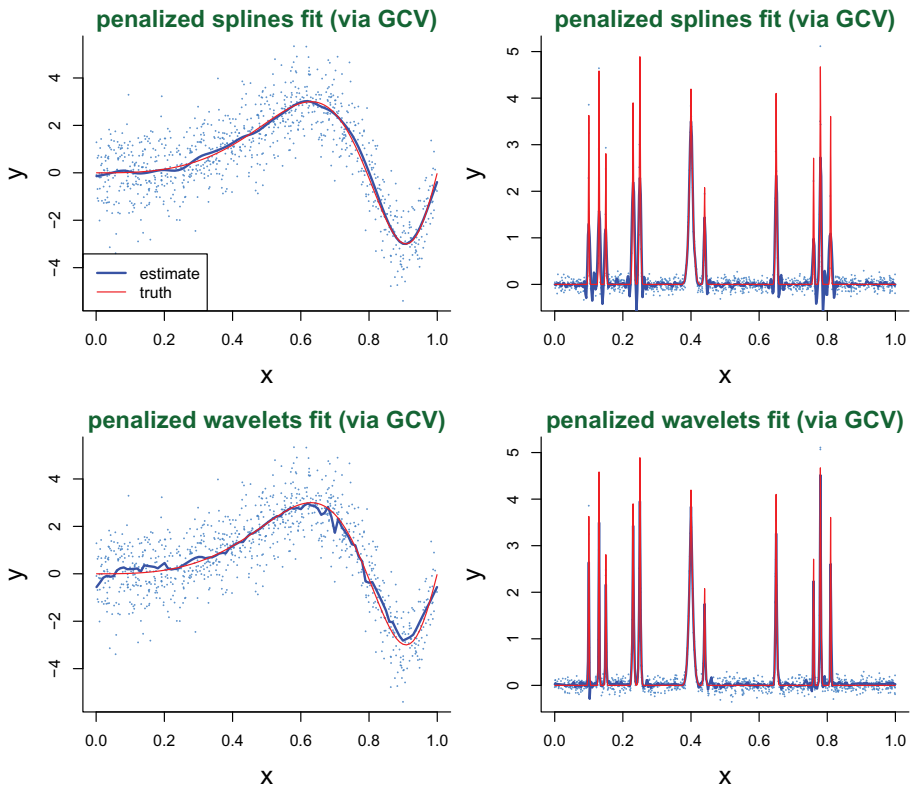

FIG 1. *Left panels: penalized spline and penalized wavelet fits (shown in blue) to a smooth regression function (shown in red). Right panels:penalized spline and penalized wavelet fits (shown in blue) to a jagged regression function (shown in red). For each fit the penalization parameter is chosen via generalized cross-validation, as described in Sections 2.4 and 3.4.*

pert, Wand and Carroll [56, 57]). Nevertheless, a chasm exists between wavelet-based nonparametric regression and the older and ubiquitous penalized splines-based nonparametric regression. In this article we remove this chasm and show that wavelets can be used in semiparametric regression settings in virtually the same way as splines. The only substantial difference is the type of penalization. The standard for splines is an L_2-type penalty, whilst for wavelets *sparseness-inducing* penalties, such as the L_1 penalty, are usually preferable. For mixed model and Bayesian approaches, this translates to the coefficients of wavelet basis functions having non-Gaussian (e.g. Laplacian) distributions, rather than the Gaussian distributions typically used for spline basis coefficients.

Figure 1 depicts two scatterplots: one of which is better suited to penalized spline regression, the other of which is more conducive to penalized wavelets. The data in the left panels is generated from a smooth regression function and penalized splines with generalized cross-validation (GCV) choice of the penalty

parameter (Section 2.4) appear to perform adequately. Adoption of an analogous strategy for penalized wavelets results in an overly rugged fit. The data in the right panels is generated from a jagged regression function and the data are more amenable to penalized wavelet analysis.

As will be clear by the end of this section, penalized splines and wavelet scatterplot smoothers are quite similar in the sense that each is simply a linear combination of basis functions. Apart from the basis functions themselves, the only difference between penalized splines and wavelets is the nature of the coefficient estimation strategy. However, this commonality is not clearly apparent from the literatures of each, as they have evolved largely independent of one another. The thrust of this article is putting penalized spline and wavelets on a common ground and explaining that variants of the same principles can be used for effective fitting and inference. One interesting payoff is semiparametric regression models containing both penalized splines and penalized wavelets (Sections 5.2 and 5.3).

1.1. Aspects of wavelets best left aside in the context of this article

Readers who have no previous exposure to wavelets could proceed to the second last paragraph of this section. Those who are are well-versed in wavelet theory and methodology are advised, in the context of the current article, to leave aside the following aspects of the wavelet nonparametric regression literature:

- Mallat's Pyramid Algorithm and the Discrete Wavelet Transform;
- the advantages of a predictor variable being equally-spaced and the sample size being a power of 2;
- the coefficient space approach to wavelet nonparametric and semiparametric regression;
- oracle and Besov space theory, and similar functional analysis theory.

We are not saying that these aspects of wavelets are unimportant. Indeed, some of them play crucial roles in the computation of penalized wavelets — see Section 3.1 on wavelet basis function construction. Rather, we are saying that these aspects have contributed to the aforementioned chasm between wavelet- and spline-based nonparametric regression, and thus has hindered cross-fertilization between the two areas of research. This is the reason for our plea to leave them aside for the remainder of this article.

The only aspect of wavelets that is of fundamental importance for semiparametric regression is that, as with splines, they can be used to construct a set of basis functions over an arbitrary compact interval $[a, b]$ in $\mathbb{R}$, and that linear combinations of such basis functions are able to estimate particular, usually jagged, regression functions better than spline bases.

We believe that this viewpoint of wavelet-based semiparametric regression is superior in terms of its accordance with *regression modelling*. That is: postulate models in terms of linear combinations of basis functions, with appropriate distributional assumptions, penalties and the like. But keep the numerical details in the background.

1.2. Relationship to existing wavelet nonparametric regression literature

The literature on wavelet approaches to nonparametric regression is now quite immense and we will not attempt to survey it here. Books on the topic include Vidakovic [62] and Nason [47]. The penalized wavelets that we develop in the present article are similar in substance to most wavelet-based nonparametric regression estimators already developed. The reason for this article, as the title suggests, is to show, explicitly, how wavelets can be integrated into existing semiparametric regression structures. A reader familiar with the first author's co-written expositions on semiparametric regression, Ruppert, Wand and Carroll [56, 57]), will immediately see how wavelets can be added to the semiparametric regression armory.

Despite the absence of a literature survey, we give special mention to Antoniadis and Fan [3], which crystallized the penalized least squares approaches to wavelet nonparametric regression and their connections with *wide data*, or "$p \gg n$", regression. That article, like this one, also proposed a way of handling non-equispaced predictor data. Finally, we note that our adoption of the term *penalized wavelets* for our proposed new wavelet regression paradigm is driven by the close analogues with *penalized splines*. This term has made at least one appearance in the literature: Antoniadis, Bigot and Gijbels [2], although their penalized wavelets are more in keeping with classical wavelet nonparametric regression.

1.3. Elements of penalized splines

Penalized splines are the building blocks of semiparametric regression models — a class of models that includes generalized additive models, generalized additive mixed models, varying coefficient models, geoadditive models, subject-specific curve models, among others (e.g. Ruppert, Wand and Carroll [56, 57]). Penalized splines include, as special cases, smoothing splines (e.g. Wahba [63]), P-splines (Eilers and Marx [21]), and pseudosplines (Hastie [34]). A distinguishing feature of penalized splines is that the number of basis functions does not necessarily match the sample sizes, and terminology such as *low-rank* or *fixed-rank* smoothing has emerged to describe this aspect. The R (R Development Core Team [54]) function `smooth.spline()` uses a low-rank modification of smoothing splines when the sample size exceeds 50. In the generalized additive (mixed) model R package `mgcv` (Wood [73]) the univariate function estimates use yet another variant of penalized splines: low-rank thin plate splines (Wood [71]).

In the early sections of this article we will confine discussion to the simple nonparametric regression model, and return to various semiparametric extensions in later sections. So, for now, we focus on the situation where we observe predictor/response pairs (x_i, y_i), $1 \le i \le n$, and consider the model

$$y_i = f(x_i) + \varepsilon_i \tag{1}$$

where the ε_i are a random sample from a distribution with mean zero and variance σ_ε^2. The regression function f is assumed to be "smooth" in some sense. There are numerous functional analytic ways by which this smoothness assumption can be formalized. See, for example, Chapter 1 of Wahba [63]. The *penalized spline* model for the regression function f is

$$f(x) = \beta_0 + \beta_1\, x + \sum_{k=1}^{K} u_k\, z_k(x)$$

where $\{z_k(\cdot) : 1 \le k \le K\}$ is a spline basis function.

The coefficients β_0, β_1 and $u_1, \ldots, u_K$ may be estimated in a number of ways (Section 2). The simplest is *penalized least squares*, which involves choosing the coefficients to minimize

$$\sum_{i=1}^{n} \left\{ y_i - \beta_0 - \beta_1\, x_i - \sum_{k=1}^{K} u_k z_k(x_i) \right\}^2 + \lambda \sum_{k=1}^{K} u_k^2 \tag{2}$$

where $\lambda > 0$ is usually referred to as the smoothing parameter or penalty parameter. The linear component $\beta_0 + \beta_1\, x$ is left unpenalized since the most popular spline basis functions have orthogonality properties with respect to lines. However, there is nothing special about lines, and other spline basis functions are such that other polynomial functions of x should be unpenalized. The default basis for penalized wavelets that we develop in Section 3.1 has only the constant component unpenalized.

Criterion (2) assumes that the $z_k(\cdot)$ have been linearly transformed to a *canonical form*, in that the penalty is simply a multiple of the sum of squares of the spline coefficients. For many spline basis functions it is appropriate that the penalty is a more elaborate quadratic form $\lambda \sum_{k=1}^{K} \sum_{k'=1}^{K} \Omega_{kk'} u_k u_{k'}$ where $\Omega_{kk'}$ depends on the basis functions. However, one can always linearly transform the $z_k(\cdot)$ so that the *canonical* penalty $\lambda \sum_{k=1}^{K} u_k^2$ is appropriate (see e.g. Wand and Ormerod [66], Section 4). Throughout this article we assume that the $z_k(\cdot)$ are in canonical form.

1.3.1. Basis construction

At the heart of contemporary penalized splines are algorithms, and corresponding software routines, for construction of design matrices for smooth function components in semiparametric regression — but also for plotting function estimates over a fine grid, and prediction at other locations in the predictor space. Algorithm 1 describes spline basis construction in its most elementary form.

The most obvious and common use of Algorithm 1 is to obtain the $z_k(x_i)$ values required for the fitting via the penalized least squares criterion (2). This involves setting $\boldsymbol{g} = (x_1, \ldots, x_n)$. The output matrix, usually denoted by $\boldsymbol{Z}$, is then the $n \times K$ design matrix containing the $z_k(x_i)$. However, Algorithm 1 is also relevant for prediction at other values of the x variable and for plotting

Algorithm 1 *Spline basis function construction in its most elementary form.*

Inputs: (1) $\boldsymbol{g} = (g_1, \ldots, g_M)$: vector of length M in the predictor space

(2) $a \leq \min(\boldsymbol{g})$ and $b \geq \max(\boldsymbol{g})$: end-points of compact interval $[a, b]$ over which basis functions are non-zero

(3) Knot locations $\kappa_1, \ldots, \kappa_K$

Inputs (2) and (3) are sufficient to define spline basis functions $z_k(\cdot)$, $1 \leq k \leq K$, over the interval $[a, b]$

Output: $$\boldsymbol{Z_g} = \begin{bmatrix} z_1(g_1) & \cdots & z_K(g_1) \\ \vdots & \ddots & \vdots \\ z_1(g_M) & \cdots & z_K(g_M) \end{bmatrix}$$

($M \times K$ design matrix containing the $z_k(\cdot)$ evaluated at $\boldsymbol{g}$)

estimates of f over a grid. For example, prediction at $x = x_{\text{new}}$ would require a call to Algorithm 1 with $\boldsymbol{g} = x_{\text{new}}$, in which case a $1 \times K$ matrix containing the values of $z_k(x_{\text{new}})$, $1 \leq k \leq K$, would be returned. This matrix, together with the estimated coefficients, could then be used to construct the prediction $\widehat{f}(x_{\text{new}})$.

Examples of Algorithm 1 include:

- the `smooth.spline()` function in R,
- the appendix of Eilers and Marx [21] on a discrete penalty (P-spline) approach, combined with the mixed model basis transformation described in Currie and Durbán [14],
- the $d = 1$ version of the algorithm described in Section 2 of Wood [71],
- special cases of the general model for polynomial splines given in Section 4 of Welham, Cullis, Kenward and Thompson [70],
- the O'Sullivan spline (O-spline) basis construction described in Wand and Ormerod [66] and Appendix A of the present article.

1.4. Proposed new penalized wavelet paradigm

The foundation stone for our proposed new paradigm for embedding penalized wavelets into semiparametric regression is an algorithm, Algorithm 2, taking almost the same form as Algorithm 1. A concrete version of Algorithm 2 is given in Section 3.1.

There are a few key differences between penalized wavelets and penalized splines:

1. computational considerations (see Section 3.1) dictate that once a, b and K are set, there are no other options for basis function specification. Hence, the analogue of knot placement is absent for penalized wavelets.

Algorithm 2 *Wavelet basis function construction in its most elementary form.*

Inputs: (1) $\boldsymbol{g} = (g_1, \ldots, g_M)$ (vector of M in the predictor space)

(2) $a \leq \min(\boldsymbol{g})$ and $b \geq \max(\boldsymbol{g})$ (end-points of compact interval $[a, b]$ over which basis functions are non-zero)

(3) $K = 2^L - 1, L$ positive integer

(These inputs are sufficient to define wavelet basis functions $z_k(\cdot)$, $1 \leq k \leq K$, over the interval $[a, b]$)

Output: $$\boldsymbol{Z}_{\boldsymbol{g}} = \begin{bmatrix} z_1(g_1) & \cdots & z_K(g_1) \\ \vdots & \ddots & \vdots \\ z_1(g_M) & \cdots & z_K(g_M) \end{bmatrix}$$

($M \times K$ design matrix containing the $z_k(\cdot)$ evaluated at $\boldsymbol{g}$)

2. symmetry conditions dictate that the number of basis functions K should satisfy $K = 2^L - 1$ for some positive integer L, which denotes the number of *levels* in the wavelet basis.
3. the unpenalized companion of $\boldsymbol{Z}$ consists of a *constant* rather than *linear* function of the x_is.
4. the coefficients of the basis functions in $\boldsymbol{Z}$ are subject to a *sparseness-inducing* penalty such as the L_1 penalty.

Section 3.1 gives details on computation of $\boldsymbol{Z}$.

The third and fourth of differences imply that, instead of (2), we work with a penalized least squares criterion

$$\sum_{i=1}^{n} \left\{ y_i - \beta_0 - \sum_{k=1}^{K} u_k z_k(x_i) \right\}^2 + \wp_\lambda(|u_k|) \tag{3}$$

where $\wp_\lambda$ induces a *sparse* solution, i.e. a solution for which many of the fitted u_ks are exactly zero. The simplest choice is $\wp_\lambda(x) = \lambda\, x$, corresponding to L_1 penalization. However, as discussed in Section 3.2, several other possibilities exist. As alluded to in Antoniadis and Fan [3], there is a lot of common ground between wavelet regression and wide data regression where the number of predictors exceeds the number of observations, and often labelled "$p \gg n$" regression. This connection is particularly strong for the penalized wavelet approach developed in the current article since we work with design matrices containing wavelet basis functions evaluated at the predictors. This means that the mechanics of fitting penalized wavelets are similar, and sometimes identical, to that used in fitting wide data regression models.

1.5. Common ground between penalized splines and penalized wavelets

The establishment of a wavelet basis algorithm for penalized wavelets puts them on the same footing as splines. For the nonparametric regression problem (1), the fitted values are

$$\widehat{\boldsymbol{f}} = \boldsymbol{X}\widehat{\boldsymbol{\beta}} + \boldsymbol{Z}\widehat{\boldsymbol{u}} \tag{4}$$

where $\boldsymbol{X} = [\mathbf{1}\ \boldsymbol{x}]$ for penalized splines and $\boldsymbol{X} = \mathbf{1}$ for penalized wavelets. In both cases, $\boldsymbol{Z}$ is an $n \times K$ matrix containing either K spline or K wavelet basis functions evaluated at the x_i. To re-affirm the fact that penalized wavelets are such close relatives of penalized splines we will use the

$$\{z_1(\cdot), \ldots, z_K(\cdot)\}$$

notation for the K basis functions over $[a, b]$ for both splines and wavelets, and only call upon distinguishing notation when there is a clash.

The only substantial difference between penalized splines and penalized wavelets is in the determination of the coefficients $\widehat{\boldsymbol{\beta}}$ and $\widehat{\boldsymbol{u}}$. Sections 2 and 3 lay out the differences and similarities for several fitting methods.

1.6. Outline of remainder of article

The remainder of this article is structured as follows:

2. Recap of Penalized Spline Fitting and Inference
   - 2.1 Default basis
   - 2.2 Fitting via penalized least squares
   - 2.3 Effective degrees of freedom
   - 2.4 Penalty parameter selection
   - 2.5 Fitting via frequentist fixed model representation
   - 2.6 Fitting via Bayesian inference and Markov chain Monte Carlo
   - 2.7 Fitting via mean field variational Bayes
3. Penalized Wavelet Fitting and Inference
   - 3.1 Default basis
   - 3.2 Fitting via penalized least squares
   - 3.3 Effective degrees of freedom
   - 3.4 Penalty parameter selection
   - 3.5 Fitting via frequentist mixed model representation
   - 3.6 Fitting via Bayesian inference and Markov chain Monte Carlo
   - 3.7 Fitting via mean field variational Bayes
4. Choice of Penalized Wavelet Basis Size

5. Semiparametric Regression Extensions
   5.1 Non-Gaussian response models
   5.2 Additive models
   5.3 Semiparametric longitudinal data analysis
   5.4 Non-standard semiparametric regression
6. R Software
7. Discussion

Note that Sections 2 and 3 have exactly the same subsection titles. These two sections are central to achieving our overarching goal of showing that penalized wavelet analysis can be performed in the same way as penalized spline analysis. Admittedly, most of the content of Section 2 has been described elsewhere. However, putting the various penalized spline analysis approaches in one place allows us to show the strong parallels between penalized splines and penalized wavelets.

Section 4 discusses the issue of choosing the number of penalized wavelet basis functions. We argue that this number should be of the form $2^L - 1$ where the integer L corresponds to the number of levels in the wavelet basis function hierarchy, and provide some suggestions for the choice of L. In Section 5 we discuss a number of semiparametric regression extensions of penalized wavelets including non-Gaussian response models, additive models and models for analysis of longitudinal data. R software relevant penalized wavelet semiparametric regression described in Section 6. Closing discussion is given in Section 7.

2. Recap of penalized spline regression fitting and inference

We now provide brief descriptions of the various ways by which the nonparametric regression model (1) can be fitted when f is modelled using penalized splines:

$$f(x) = \beta_0 + \beta_1\, x + \sum_{k=1}^{K} u_k\, z_k(x)$$

where $\{z_1(\cdot), \ldots, z_K(\cdot)\}$ is a set of spline basis functions appropriate for the linear component $\beta_0 + \beta_1\, x$ being unpenalized. Default choice of the $z_k(\cdot)$s is described in Section 2.1.

The following notation will be used throughout this section:

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix},\ \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix},\ \boldsymbol{u} = \begin{bmatrix} u_1 \\ \vdots \\ u_K \end{bmatrix},\ \boldsymbol{X} = \begin{bmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix},$$

$$\boldsymbol{Z} = \begin{bmatrix} z_1(x_1) & \cdots & z_K(x_1) \\ \vdots & \ddots & \vdots \\ z_1(x_n) & \cdots & z_K(x_n) \end{bmatrix},\quad \boldsymbol{C} = [\boldsymbol{X}\ \boldsymbol{Z}] \quad \text{and} \quad \boldsymbol{D} = \begin{bmatrix} 0 & 0 & \boldsymbol{0} \\ 0 & 0 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I}_K \end{bmatrix}$$

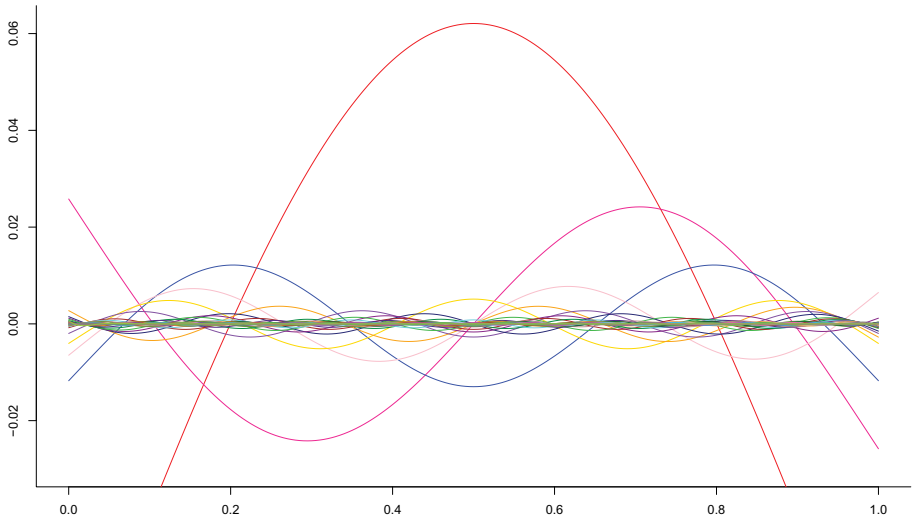

FIG 2. *Canonical O-spline basis functions for 25 equally-spaced interior knots on the unit interval.*

with $\boldsymbol{I}_K$ denoting the $K \times K$ identity matrix and $\mathbf{0}$ denoting a matrix of zeroes of appropriate size.

2.1. Default basis

For practice, it is prudent to have a default version of Algorithm 1. We believe that the B-spline basis and penalty set-up of O'Sullivan [51] is an excellent choice. It may be thought of as a low-rank version of smoothing splines (e.g. Green and Silverman [31]) and is used in the R function `smooth.spline()` when the sample size exceeds 50. Wand and Ormerod [66] describe conversion of the B-splines to canonical form. Appendix A provides details on the construction of the O'Sullivan penalized spline basis, or *O-splines* for short. Figure 2 shows the canonical O-spline basis functions with 25 equally-spaced interior knots on the unit interval.

2.2. Fitting via penalized least squares

The penalized spline criterion (2) has the matrix representation:

$$\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{Z}\boldsymbol{u}\|^2 + \lambda\|\boldsymbol{u}\|^2. \tag{5}$$

Noting that, in terms of $\boldsymbol{C}$ and $\boldsymbol{D}$, the criterion equals

$$\left\| \boldsymbol{y} - \boldsymbol{C} \begin{bmatrix} \boldsymbol{\beta} \\ \boldsymbol{u} \end{bmatrix} \right\|^2 + \lambda \begin{bmatrix} \boldsymbol{\beta} \\ \boldsymbol{u} \end{bmatrix}^T \boldsymbol{D} \begin{bmatrix} \boldsymbol{\beta} \\ \boldsymbol{u} \end{bmatrix}$$

the following solution is easily obtained:

$$\begin{bmatrix} \widehat{\boldsymbol{\beta}} \\ \widehat{\boldsymbol{u}} \end{bmatrix} = (\boldsymbol{C}^T\boldsymbol{C} + \lambda\boldsymbol{D})^{-1}\boldsymbol{C}^T\boldsymbol{y}. \tag{6}$$

The vector of fitted values is then

$$\widehat{\boldsymbol{f}}_\lambda = \begin{bmatrix} \widehat{f}_\lambda(x_1) \\ \vdots \\ \widehat{f}_\lambda(x_n) \end{bmatrix} = \boldsymbol{C}\begin{bmatrix} \widehat{\boldsymbol{\beta}} \\ \widehat{\boldsymbol{u}} \end{bmatrix}. \tag{7}$$

2.3. Effective degrees of freedom

The *effective degrees of freedom (edf)* of a nonparametric regression fit is defined to be the following function of the penalty parameter λ:

$$\text{edf}(\lambda) \equiv \frac{1}{\sigma_\varepsilon^2}\sum_{i=1}^{n}\text{Cov}(\widehat{f}_\lambda(x_i), y_i). \tag{8}$$

It provides a meaningful and scale-free measure of the amount of fitting (Buja, Hastie and Tibshirani [11]). Definition (8) has its roots in Stein's unbiased risk estimation theory (Stein [60], Efron [19]). If the vector of fitted values can be written as $\widehat{\boldsymbol{f}}_\lambda = \boldsymbol{S}_\lambda\boldsymbol{y}$ for some $n \times n$ matrix not depending on the y_is (known as the *smoother matrix*) then

$$\text{edf}(\lambda) = \text{tr}(\boldsymbol{S}_\lambda). \tag{9}$$

For the penalized least squares fit (7) it follows from (6) and (7) that $\boldsymbol{S}_\lambda = \boldsymbol{C}(\boldsymbol{C}^T\boldsymbol{C} + \lambda\boldsymbol{D})^{-1}\boldsymbol{C}^T$, which leads to the expression

$$\text{edf}(\lambda) = \text{tr}\{(\boldsymbol{C}^T\boldsymbol{C} + \lambda\boldsymbol{D})^{-1}\boldsymbol{C}^T\boldsymbol{C}\}.$$

Figure 3 shows penalized spline fits to some simulated data with four different edf(λ). Setting edf(λ) too low results in underfitting of the data, whilst excessively high edf(λ) produces overfitting. For these data, edf(λ) = 12 achieves a pleasing fit.

2.4. Penalty parameter selection

In the nonparametric regression literature there are numerous proposals for selection of the penalty parameter from the data. Many of these involve trade-offs between edf(λ) and the *residual sum of squares (RSS)*

$$\text{RSS}(\lambda) = \|\boldsymbol{y} - \widehat{\boldsymbol{f}}_\lambda\|^2.$$

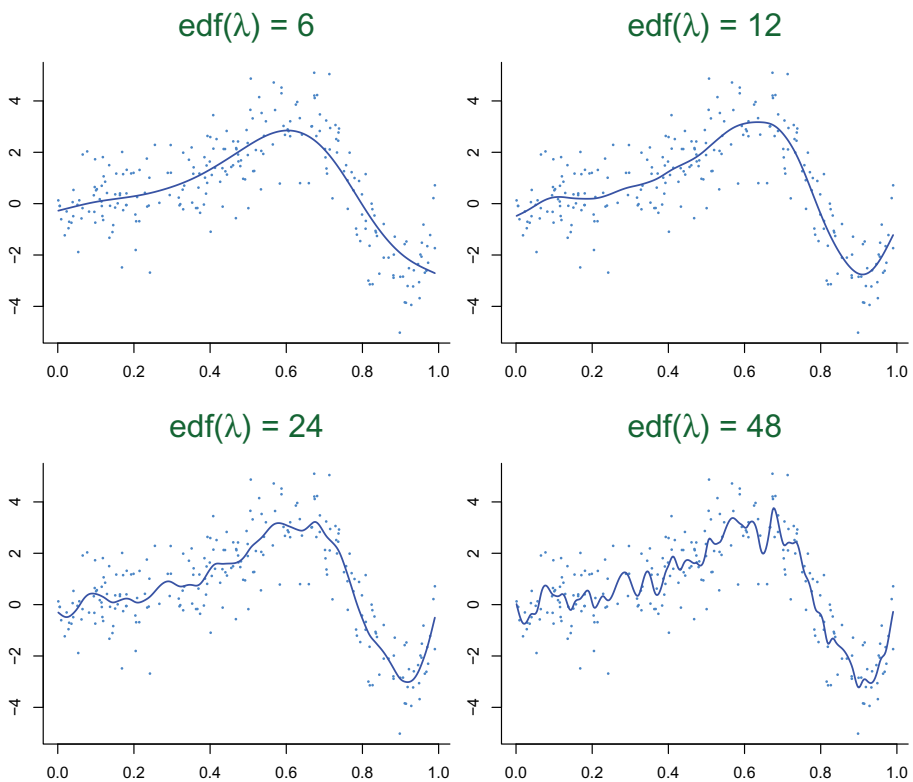

FIG 3. *Penalized spline fits to a simulated data set with four different values of the effective degrees of freedom edf*(λ).

Examples of popular penalty parameter selection criteria of this type are *Generalized Cross-Validation*,

$$\mathrm{GCV}(\lambda) = \mathrm{RSS}(\lambda)/[\{n - \mathrm{edf}(\lambda)\}^2]$$

(Craven and Wahba [13]) and *corrected Akaike's Information Criterion*,

$$\mathrm{AIC}_C(\lambda) = \log\{\mathrm{RSS}(\lambda)\} + \frac{2\{\mathrm{edf}(\lambda) + 1\}}{n - \mathrm{edf}(\lambda) - 2}$$

(Hurvich, Simonoff and Tsai [38]).

Another option for selection of λ is k-fold cross-validation, where k is a small number such as 5 or 10 (e.g. Hastie, Tibshirani and Friedman [37], Section 7.10.1). This selection method is defined, and computationally feasible, for general estimation methods and loss functions.

2.5. Fitting via frequentist mixed model representation

The frequentist mixed model representation of (5) is

$$\boldsymbol{y}|\boldsymbol{u} \sim N(\boldsymbol{X\beta} + \boldsymbol{Zu}, \sigma_\varepsilon^2 \boldsymbol{I}), \quad \boldsymbol{u} \sim N(\boldsymbol{0}, \sigma_u^2 \boldsymbol{I}). \tag{10}$$

(e.g. Ruppert *et al.* [56], Section 4.9). According to this model, the log-likelihood of the model parameters is

$$\ell(\boldsymbol{\beta}, \sigma_u^2, \sigma_\varepsilon^2) = -\tfrac{1}{2}\left\{n\log(2\pi) + \log|\boldsymbol{V}| + (\boldsymbol{y} - \boldsymbol{X\beta})^T \boldsymbol{V}^{-1}(\boldsymbol{y} - \boldsymbol{X\beta})\right\}$$

where

$$\boldsymbol{V} = \boldsymbol{V}(\sigma_u^2, \sigma_\varepsilon^2) \equiv \text{Cov}(\boldsymbol{y}) = \sigma_u^2 \boldsymbol{ZZ}^T + \sigma_\varepsilon^2 \boldsymbol{I}.$$

At the maximum we have the relationship

$$\boldsymbol{\beta} = (\boldsymbol{X}^T\boldsymbol{V}^{-1}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{V}^{-1}\boldsymbol{y} \tag{11}$$

which leads to the profile log-likelihood

$$\begin{aligned}\ell_P(\sigma_u^2, \sigma_\varepsilon^2) \quad = \quad & -\tfrac{1}{2}\left[\log|\boldsymbol{V}| + \boldsymbol{y}^T\boldsymbol{V}^{-1}\{\boldsymbol{I} - \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{V}^{-1}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{V}^{-1}\}\boldsymbol{y}\right] \\ & - \tfrac{n}{2}\log(2\pi).\end{aligned}$$

The modified profile log-likelihood, also known as the *restricted* log-likelihood, is

$$\ell_R(\sigma_u^2, \sigma_\varepsilon^2) = \ell_P(\sigma_u^2, \sigma_\varepsilon^2) - \tfrac{1}{2}\log|\boldsymbol{X}^T\boldsymbol{V}^{-1}\boldsymbol{X}|$$

and is usually preferred for estimation of the variance parameters σ_u^2 and σ_ε^2. Such estimators, which we denote by $\widehat{\sigma}_u^2$ and $\widehat{\sigma}_\varepsilon^2$, are known as restricted maximum likelihood (REML) estimators. Define

$$\widehat{\boldsymbol{V}} = \widehat{\sigma}_u^2 \boldsymbol{ZZ}^T + \widehat{\sigma}_\varepsilon^2 \boldsymbol{I}.$$

Then, in view of (11), an appropriate estimator for $\boldsymbol{\beta}$ is

$$\widehat{\boldsymbol{\beta}} = (\boldsymbol{X}^T\widehat{\boldsymbol{V}}^{-1}\boldsymbol{X})^{-1}\boldsymbol{X}^T\widehat{\boldsymbol{V}}^{-1}\boldsymbol{y}.$$

For estimation of $\boldsymbol{u}$ we appeal to the fact that its *best predictor* is

$$E(\boldsymbol{u}|\boldsymbol{y}) = \sigma_\varepsilon^2 \boldsymbol{Z}^T\boldsymbol{V}^{-1}(\boldsymbol{y} - \boldsymbol{X\beta})$$

and then plug in the above estimates to obtain

$$\widehat{\boldsymbol{u}} = \widehat{\sigma}_\varepsilon^2 \boldsymbol{Z}^T\widehat{\boldsymbol{V}}^{-1}(\boldsymbol{y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}}).$$

In summary:

- σ_u^2 and σ_ε^2 are estimated by maximum likelihood or restricted maximum likelihood,
- $\boldsymbol{\beta}$ is estimated by maximum likelihood,
- $\boldsymbol{u}$ is estimated via best prediction.

In practice, the second and third of these involve replacement of σ_u^2 and σ_ε^2 with the estimates $\widehat{\sigma}_u^2$ and $\widehat{\sigma}_\varepsilon^2$.

2.6. Fitting via Bayesian inference and Markov chain Monte Carlo

Bayesian approaches to penalized splines have been the subject of considerable research in the past decade. See, for example, Sections 2.3, 2.5 and 2.7 of Ruppert *et al.* [57]. Wand [65] describes a *graphical models* viewpoint of penalized splines and draws upon inference methods and software from that burgeoning area of research. We make use of such developments in this and the next subsections.

A Bayesian penalized spline model, corresponding to least squares penalization of $\boldsymbol{u}$, is:

$$\begin{gathered}\boldsymbol{y}|\boldsymbol{\beta},\boldsymbol{u},\sigma_\varepsilon \sim N(\boldsymbol{X\beta}+\boldsymbol{Zu},\sigma_\varepsilon^2\boldsymbol{I}),\quad \boldsymbol{u}|\,\sigma_u \sim N(\boldsymbol{0},\sigma_u^2\boldsymbol{I}),\\ \boldsymbol{\beta}\sim N(\boldsymbol{0},\sigma_\beta^2\boldsymbol{I}),\quad \sigma_u \sim \text{Half-Cauchy}(A_u),\quad \sigma_\varepsilon \sim \text{Half-Cauchy}(A_\varepsilon).\end{gathered} \tag{12}$$

The notation $\sigma \sim \text{Half-Cauchy}(A)$ means that σ has a Half Cauchy distribution with scale parameter $A > 0$. The corresponding density function is $p(\sigma) = 2/[\pi\, A\{1+(\sigma/A)^2\}],\ \sigma > 0$. As explained in Gelman [29], Half-Cauchy priors on scale parameters have the ability to achieve good non-informativity.

Approximate inference via Markov chain Monte Carlo (MCMC) is aided by the distribution theoretical result:

$$\begin{gathered}\sigma \sim \text{Half-Cauchy}(A) \text{ if and only if}\\ \sigma^2|\,a \sim \text{Inverse-Gamma}(\tfrac{1}{2},1/a)\quad\text{and}\quad a \sim \text{Inverse-Gamma}(\tfrac{1}{2},1/A^2)\end{gathered} \tag{13}$$

(e.g. Wand, Ormerod, Padoan and Frühwirth [67]). Here

$$\sigma^2 \sim \text{Inverse-Gamma}(A,B)$$

denotes that σ^2 has an Inverse Gamma distribution with shape parameter $A > 0$ and rate parameter $B > 0$. The Inverse Gamma density function is

$$p(\sigma^2) = \frac{B^A}{\Gamma(A)}\,(\sigma^2)^{-A-1}\,e^{-B/\sigma^2},\ \sigma^2 > 0.$$

Employment of (13) results in the following equivalent representation of (12):

$$\begin{gathered}\boldsymbol{y}|\boldsymbol{\beta},\boldsymbol{u},\sigma_\varepsilon^2 \sim N(\boldsymbol{X\beta}+\boldsymbol{Zu},\sigma_\varepsilon^2\boldsymbol{I}),\quad \boldsymbol{u}|\,\sigma_u^2 \sim N(\boldsymbol{0},\sigma_u^2\boldsymbol{I}),\quad \boldsymbol{\beta}\sim N(\boldsymbol{0},\sigma_\beta^2\boldsymbol{I}),\\ \sigma_u^2|\,a_u \sim \text{Inverse-Gamma}(\tfrac{1}{2},1/a_u),\ \sigma_\varepsilon^2|\,a_\varepsilon \sim \text{Inverse-Gamma}(\tfrac{1}{2},1/a_\varepsilon),\\ a_u \sim \text{Inverse-Gamma}(\tfrac{1}{2},A_u^{-2}),\ a_\varepsilon \sim \text{Inverse-Gamma}(\tfrac{1}{2},A_\varepsilon^{-2}).\end{gathered} \tag{14}$$

Figure 4 shows the directed acyclic graph (DAG) corresponding to (14).

In this Bayesian inference context, the most common choice for the vector of fitted values is the posterior mean

$$\widehat{\boldsymbol{f}} = E(\boldsymbol{X\beta}+\boldsymbol{Zu}|\boldsymbol{y}) = \boldsymbol{X}\,E(\boldsymbol{\beta}|\boldsymbol{y}) + \boldsymbol{Z}\,E(\boldsymbol{u}|\boldsymbol{y}).$$

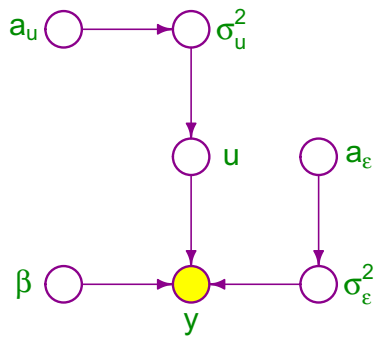

FIG 4. *Directed acyclic graph representation of the auxiliary variable Bayesian penalized spline model (14). The shaded node corresponds to observed data.*

The posterior distributions of $\boldsymbol{\beta}$ and $\boldsymbol{u}$, as well as the scale parameters σ_u and σ_ε, are not available in closed form. However, the *full conditionals* can be shown to have the following distributions:

$$\begin{aligned}
\left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{u}\end{array}\right]\Big|\text{rest} &\sim N\left(\left(\sigma_\varepsilon^{-2}\boldsymbol{C}^T\boldsymbol{C}+\left[\begin{array}{cc}\sigma_\beta^{-2}\boldsymbol{I} & \boldsymbol{0}\\ \boldsymbol{0} & \sigma_u^{-2}\boldsymbol{I}\end{array}\right]\right)^{-1}\sigma_\varepsilon^{-2}\boldsymbol{C}^T\boldsymbol{y},\right.\\
&\qquad\left.\left(\sigma_\varepsilon^{-2}\boldsymbol{C}^T\boldsymbol{C}+\left[\begin{array}{cc}\sigma_\beta^{-2}\boldsymbol{I} & \boldsymbol{0}\\ \boldsymbol{0} & \sigma_u^{-2}\boldsymbol{I}\end{array}\right]\right)^{-1}\right),\\
\sigma_u^2|\text{rest} &\sim \text{Inverse-Gamma}\left(\tfrac{1}{2}(K+1),\tfrac{1}{2}\|\boldsymbol{u}\|^2+a_u^{-1}\right),\\
\sigma_\varepsilon^2|\text{rest} &\sim \text{Inverse-Gamma}\left(\tfrac{1}{2}(n+1),\tfrac{1}{2}\|\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}-\boldsymbol{Z}\boldsymbol{u}\|^2+a_\varepsilon^{-1}\right),\\
a_u|\text{rest} &\sim \text{Inverse-Gamma}\left(1,\sigma_u^{-2}+A_u^{-2}\right)\\
\text{and}\quad a_\varepsilon|\text{rest} &\sim \text{Inverse-Gamma}\left(1,\sigma_\varepsilon^{-2}+A_\varepsilon^{-2}\right).
\end{aligned}$$

Here 'rest' denotes the set of other random variables in model (14). Since all full conditional are standard distributions *Gibbs sampling*, the simplest type of MCMC sampling, can be used to draw samples from the posterior distributions (see e.g. Robert and Casella [55]).

The DAG in Figure 4 is useful for determination of the above full conditional distributions. This is due to the fact that the full conditional distribution of any node on the graph is the same as the distribution of the node conditional on its *Markov blanket* (e.g. Pearl [52]). The Markov blanket of a node consists of its parent nodes, co-parent nodes and child nodes.

2.7. Fitting via mean field variational Bayes

Mean field variational Bayes (MFVB) (e.g. Attias [5], Wainwright and Jordan [64]) is a deterministic alternative to Markov chain Monte Carlo which allows

faster fitting and inference. In certain circumstances MFVB can be quite accurate and there is *prima facie* evidence that such is the case for the Bayesian penalized spline model (14). Moreover, MFVB algorithms are often very simple to implement. Each of the MFVB algorithms in the present article involve straightforward algebraic calculations. In Ormerod and Wand [49] we explained MFVB using statistical examples similar to those presented here.

For (14) we start by restricting the full posterior density function

$$p(\boldsymbol{\beta}, \boldsymbol{u}, \sigma_u^2, \sigma_\varepsilon^2, a_u, a_\varepsilon | \boldsymbol{y}) \tag{15}$$

to have the product form

$$q(\boldsymbol{\beta}, \boldsymbol{u}, \sigma_u^2, \sigma_\varepsilon^2) = q(\boldsymbol{\beta}, \boldsymbol{u})\, q(\sigma_u^2, \sigma_\varepsilon^2)\, q(a_u, a_\varepsilon) \tag{16}$$

where q denotes a density function over the appropriate parameter space. Let q^* denote the optimal q densities in terms minimum Kullback-Leibler distance between (15) and (16). Then, as shown in Appendix C,

$$\begin{array}{l} q^*(\boldsymbol{\beta}, \boldsymbol{u}) \text{ is a Multivariate Normal density function,} \\ q^*(\sigma_u^2),\ q^*(\sigma_\varepsilon^2),\ q^*(a_u) \text{ and } q^*(a_\varepsilon) \text{ are each Inverse Gamma} \\ \text{density functions.} \end{array} \tag{17}$$

Let $\boldsymbol{\mu}_{q(\boldsymbol{\beta},\boldsymbol{u})}$ and $\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{u})}$ denote the mean vector and covariance matrix for $q^*(\boldsymbol{\beta}, \boldsymbol{u})$ and $A_{q(\sigma_u^2)}$ and $B_{q(\sigma_u^2)}$ denote the shape and rate parameters for $q^*(\sigma_u^2)$. Apply similar definitions for the parameters in $q^*(\sigma_\varepsilon^2)$, $q^*(a_u)$ and $q^*(a_\varepsilon)$. Then the optimal values of these parameters are determined from Algorithm 3.

Algorithm 3 *Mean field variational Bayes algorithm for the determination of the optimal parameters in $q^*(\boldsymbol{\beta}, \boldsymbol{u})$, $q^*(\sigma_u^2)$, and $q^*(\sigma_\varepsilon^2)$ for the Bayesian penalized wavelet model (14)*

Initialize: $\mu_{q(1/\sigma_\varepsilon^2)},\ \mu_{q(1/\sigma_u^2)},\ \mu_{q(1/a_\varepsilon)},\ \mu_{q(1/a_u)} > 0$.
Cycle:

$$\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{u})} \leftarrow \left(\mu_{q(1/\sigma_\varepsilon^2)} \boldsymbol{C}^T \boldsymbol{C} + \begin{bmatrix} \sigma_\beta^{-2} \boldsymbol{I}_2 & \boldsymbol{0} \\ \boldsymbol{0} & \mu_{q(1/\sigma_u^2)} \boldsymbol{I}_K \end{bmatrix} \right)^{-1}$$

$$\boldsymbol{\mu}_{q(\boldsymbol{\beta},\boldsymbol{u})} \leftarrow \mu_{q(1/\sigma_\varepsilon^2)} \boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{u})} \boldsymbol{C}^T \boldsymbol{y}$$

$$\mu_{q(1/a_\varepsilon)} \leftarrow 1/\{\mu_{q(1/\sigma_\varepsilon^2)} + A_\varepsilon^{-2}\} \quad ; \quad \mu_{q(1/a_u)} \leftarrow 1/\{\mu_{q(1/\sigma_u^2)} + A_u^{-2}\}$$

$$B_{q(\sigma_u^2)} \leftarrow \tfrac{1}{2}\{\|\boldsymbol{\mu}_{q(\boldsymbol{u})}\|^2 + \text{tr}(\boldsymbol{\Sigma}_{q(\boldsymbol{u})})\} + \mu_{q(1/a_u)}$$

$$B_{q(\sigma_\varepsilon^2)} \leftarrow \tfrac{1}{2}\{\|\boldsymbol{y} - \boldsymbol{C}\boldsymbol{\mu}_{q(\boldsymbol{\beta},\boldsymbol{u})}\|^2 + \text{tr}(\boldsymbol{C}^T \boldsymbol{C} \boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{u})})\} + \mu_{q(1/a_\varepsilon)}$$

$$\mu_{q(1/\sigma_u^2)} \leftarrow \tfrac{1}{2}(K+1)/B_{q(\sigma_u^2)} \quad ; \quad \mu_{q(1/\sigma_\varepsilon^2)} \leftarrow \tfrac{1}{2}(n+1)/B_{q(\sigma_\varepsilon^2)}$$

until the increase in $\underline{p}(\boldsymbol{y}; q)$ is negligible.

The lower bound on the marginal log-likelihood is

$$\begin{aligned}
\log \underline{p}(\boldsymbol{y};q) \quad = \quad & \tfrac{1}{2}(K+2) - \tfrac{1}{2}\,n\log(2\pi) - 2\log(\pi) + \log\Gamma(\tfrac{1}{2}(K+1)) \\
& + \log\Gamma(\tfrac{1}{2}(n+1)) - \log(\sigma^2_{\boldsymbol{\beta}}) - \log(A_u) - \log(A_\varepsilon) \\
& - \tfrac{1}{2\sigma^2_{\boldsymbol{\beta}}}\{\|\boldsymbol{\mu}_{q(\boldsymbol{\beta})}\|^2 + \mathrm{tr}(\boldsymbol{\Sigma}_{q(\boldsymbol{\beta})})\} + \tfrac{1}{2}\log|\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{u})}| \\
& - \tfrac{1}{2}(K+1)\log\{B_{q(\sigma^2_u)}\} - \tfrac{1}{2}(n+1)\log\{B_{q(\sigma^2_\varepsilon)}\} \\
& - \log(\mu_{q(1/\sigma^2_u)} + A_u^{-2}) - \log(\mu_{q(1/\sigma^2_\varepsilon)} + A_\varepsilon^{-2}) \\
& + \mu_{q(1/\sigma^2_u)}\mu_{q(1/a_u)} + \mu_{q(1/\sigma^2_\varepsilon)}\mu_{q(1/a_\varepsilon)}
\end{aligned}$$

Figure 5 illustrates Bayesian penalized spline regression using both MCMC and MFVB approaches described in this and the preceding subsections. The

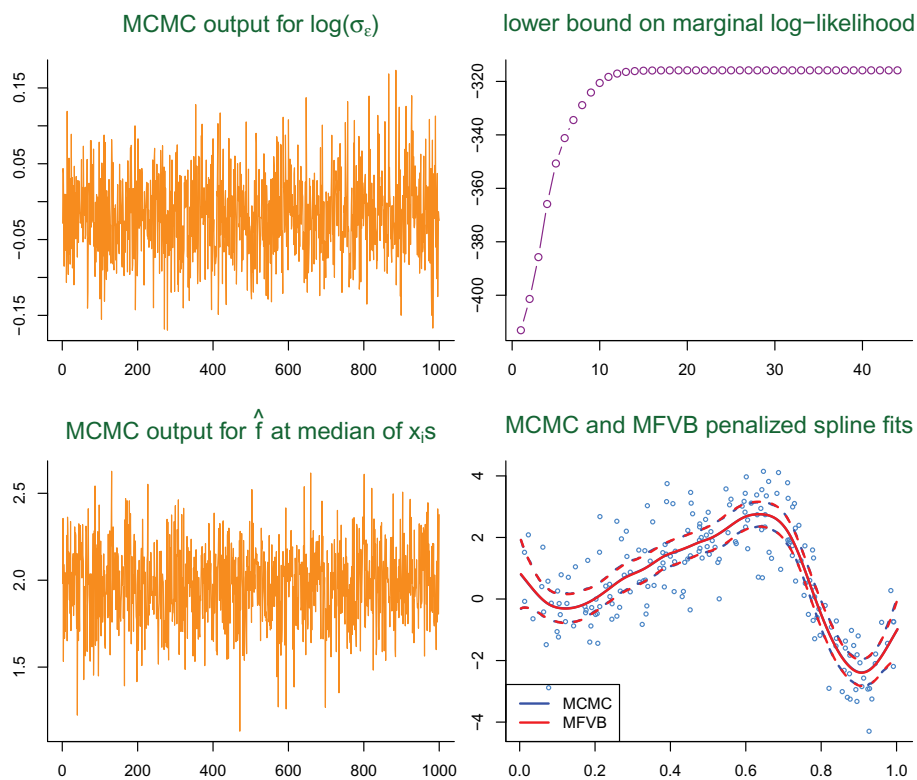

FIG 5. *Left panels: MCMC output for fitting Bayesian penalized spline model to simulated data. The upper left panel is for* $\log(\sigma_\varepsilon)$. *The lower left panel is for the estimated function at the median of the* x_i*s. Upper right panel: successive values of* $\log \underline{p}(\boldsymbol{y};q)$ *to monitor convergence of the MFVB algorithm. Lower right panel: Fitted function estimates and pointwise 95% credible sets for both MCMC and MFVB approaches.*

data were generated according to

$$y_i = 3\sin(2\pi x_i^3) + \varepsilon_i$$

with the x_is uniformly distributed on $(0,1)$ and $\varepsilon_i \overset{\text{ind.}}{\sim} N(0,1)$. Here and elsewhere $\overset{\text{ind.}}{\sim}$ stands for "independently distributed as". For the MCMC approach, samples of size 10000 were generated. The first 5000 values were discarded and the second 5000 values were thinned by a factor of 5. For the MFVB approach the iterations were terminated when the relative change in $\log \underline{p}(\boldsymbol{y}; q)$ fell below 10^{-10}. For this example the MCMC and MFVB fits and pointwise 95% credible sets are almost indistinguishable, suggesting that MFVB achieves high accuracy for Gaussian response Bayesian penalized spline regression.

Finally, we mention that the Bayesian penalized spline model treated here can be fitted via MFVB using the Infer.NET computing environment (Minka, Winn, Guiver and Knowles [44]). Wang and Wand [69] provide illustration of such implementation.

3. Penalized wavelet regression fitting and inference

This section parallels the previous with wavelets replacing splines. As we shall see, the approaches to fitting and inference are similar in many respects. The only substantial difference is the type of penalization.

Consider, again, the nonparametric regression model (1) but with with the smoothness assumption on f relaxed somewhat to allow for jumpier and spikier regression functions. Donoho [16], for example, discusses quantification of such relaxed smoothness assumptions via functional analytic structures such as Besov spaces. For the remainder of the present article we will simply say that f is a *jagged* function and refer the reader to articles such as Donoho [16] for mathematical formalization. For such jagged f we consider penalized wavelets models of the form:

$$f(x) = \beta_0 + \sum_{k=1}^{K} u_k\, z_k(x)$$

where $\{z_k(\cdot) : 1 \le k \le K\}$ is an appropriate set of *wavelet* basis functions. Default choice of the $z_k(\cdot)$s is described in Section 3.1.

The following notation will be used throughout this section:

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix},\ \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \end{bmatrix},\ \boldsymbol{u} = \begin{bmatrix} u_1 \\ \vdots \\ u_K \end{bmatrix},\ \boldsymbol{X} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix},$$

$$\boldsymbol{Z} = \begin{bmatrix} z_1(x_1) & \cdots & z_K(x_1) \\ \vdots & \ddots & \vdots \\ z_1(x_n) & \cdots & z_K(x_n) \end{bmatrix} \quad \text{and} \quad \boldsymbol{C} = [\boldsymbol{X}\ \boldsymbol{Z}].$$

The $\boldsymbol{\beta}$ vector and $\boldsymbol{X}$ matrix correspond to constants being unpenalized. Even though $\boldsymbol{\beta}$ is a scalar and $\boldsymbol{X}$ is a column vector, we continue to use such notation to allow easier comparison and contrast between penalized wavelets and splines.

3.1. Default basis

In this section we begin to fill in the missing details of Algorithm 2.

The assembly of a default basis for penalized wavelets relies on classical wavelet construction over equally-spaced grids on $[0,1)$ of length R, where R is a power of 2. Let the functions $\{z_k^U(\cdot) : 1 \le k \le R-1\}$, each defined on $[0,1)$, be such that

$$\boldsymbol{W} = R^{-1/2}[1 \;\; \underset{1\le k\le R-1}{z_k^U(\tfrac{i-1}{R})}]_{1\le i\le R} \tag{18}$$

where $\boldsymbol{W}$ is an $R \times R$ orthogonal matrix known as a *wavelet basis matrix*. We also insist that, for any fixed k, the $z_k^U(\cdot)$ do not depend on the value of R. Hence, if R is increased from 4 to 8 then the functions $z_1^U(\cdot)$, $z_2^U(\cdot)$ and $z_3^U(\cdot)$ remain unchanged. The "U" superscript denotes the fact the z_k^U are only defined over the unit interval.

If $\boldsymbol{y}$ is an $R \times 1$ vector of responses then it may be represented in terms of $\boldsymbol{W}$ as

$$\boldsymbol{y} = \boldsymbol{W}\widehat{\boldsymbol{\theta}}$$

where, using the orthogonality of $\boldsymbol{W}$,

$$\widehat{\boldsymbol{\theta}} = (\boldsymbol{W}^T\boldsymbol{W})^{-1}\boldsymbol{W}^T\boldsymbol{y} = \boldsymbol{W}^T\boldsymbol{y}. \tag{19}$$

A fast $O(R)$ algorithm, known as the *Discrete Wavelet Transform*, exists for determination of $\widehat{\boldsymbol{\theta}}$. If $\boldsymbol{y}$ corresponds to a signal contaminated by noise then a common denoising strategy involves annihilation or shrinkage of certain entries of $\widehat{\boldsymbol{\theta}}$. This is *not* the general approach to wavelet-based regression being studied in the present article and is only mentioned here to relate the $\boldsymbol{W}$ matrix to the established wavelet literature. Later in this section we will use (18) for computation of default penalized spline basis functions.

Until the mid-1980s the only known choice of $z_k^U(\cdot)$ having compact support over arbitrarily small intervals was the piecewise constant *Haar* basis. Starting with Daubechies [15], many continuous and arbitrarily smooth $z_k^U(\cdot)$ have been discovered and allowed efficient approximation of jagged functions. Each of the $z_k^U(\cdot)$, $1 \le k \le R-1$, are shifts and dilations of a single ("mother") wavelet function. Figure 6 shows four wavelet functions from the basic Daubechies family. The numbers correspond to the amount of smoothness. In the R package `wavethresh` (Nason [48]) this is referenced using `family="DaubExPhase"` and the smoothness number is denoted by `filter.number`. Note, however, that the Daubechies wavelet functions do not admit explicit algebraic expressions and can only be constructed via recursion over equally-spaced grids of size equal to a power of 2.

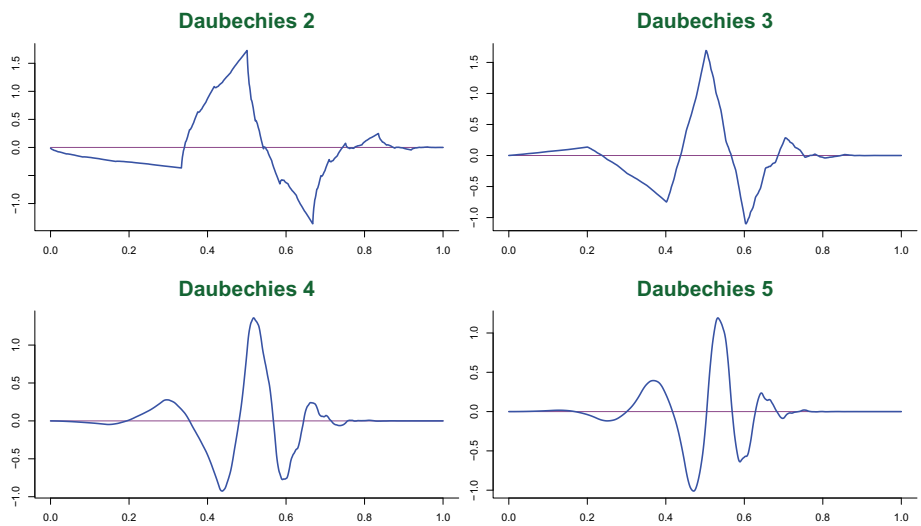

FIG 6. *Daubechies "mother" wavelets with smoothness values 2,3,4 and 5.*

The $z_k^U(\cdot)$ basis functions with the same amount of dilation, but differing shift, are said to be on the same level. The number of basis functions at level ℓ is $2^{\ell-1}$ for each of $\ell = 1, \ldots, \log_2(R)$. Our default basis definition requires that we impose the following ordering on the $z_k^U(\cdot)$, $1 \leq k \leq R-1$:

- $z_1(\cdot)$ is the single function on level 1
- $z_2(\cdot)$ and $z_3(\cdot)$ are on level 2, with ordering from left to right in terms of the support of the functions.
- Continue in this fashion for levels $3, \ldots, \log_2(R)$.

Figure 7 shows the z_k^U functions generated by the Daubechies 5 wavelet with resolution $R = 16$.

Let a and b be the end-point parameters defined in Algorithm 2 and $K = 2^L - 1$ be the required number of basis functions. We propose that default penalized wavelet basis functions take the form:

$$z_k(x) = z_k^U\left(\frac{x-a}{b-a}\right), \quad 1 \leq k \leq K.$$

where the z_k^Us are as in (18). We see no compelling reason to choose z_k^U from outside the basic Daubechies family. A reasonable default for the smoothness number is 5.

It remains to discuss computation of $z_k^U(x)$ for arbitrary $x \in [0,1)$. This simply involves choosing R to be a very large number such as $R = 2^{14} = 16384$ and then approximating via $z_k^U(x)$ linear interpolation over the grid $0, \frac{1}{R}, \ldots, \frac{R-1}{R}$. Specifically,

$$z_k^U(x) \approx \{1 - (xR - \lfloor xR \rfloor)\} z_k^U(\lfloor xR \rfloor / R) + (xR - \lfloor xR \rfloor) z_k^U((\lfloor xR \rfloor + 1)/R)$$

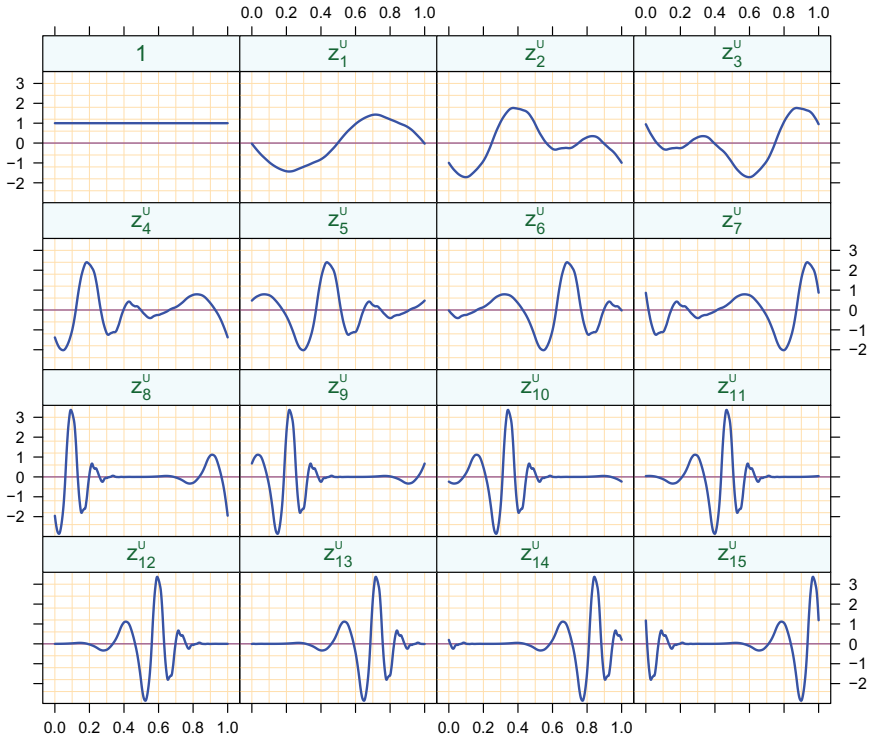

FIG 7. *Debauchies 5* $z_k^U(\cdot)$ *functions for* $R = 16$*, with ordering as prescribed in the text. The constant function, corresponding to the first column of the* $\boldsymbol{W}$ *matrix, is also shown.*

where $z_k^U(1) \equiv z_k^U(\frac{R-1}{R})$. All required calculations can be performed rapidly using the Discrete Wavelet Transform and without explicit construction of the $\boldsymbol{W}$ matrix. An R function that performs efficient default basis function computation is given in Appendix A.

Figure 8 illustrates approximation of the z_k^U functions for $K = 15$. The top-left panel shows values of z_k^U over a coarse grid with resolution $R = 16$. As R increases to 32, 64 and 128 the number of z_k^U functions increases to $R - 1$ and there is successive doubling of the resolution of the first 15 $z_k^U(\cdot)$ that are needed for the penalized wavelet basis.

Figure 9 shows the default basis functions for varying values of $K = 2^L - 1$. A significant aspect of the basis functions, apparent from Figure 9, is their hierarchical nature. To move from $L = L'$ to $L = L' + 1$ one simply adds $2^{L'}$ new basis functions corresponding to dilations of the highest level basis functions at level L'. This means that, for example, the basis functions for $L = 7$ are also present for $L = 4, 5, 6$.

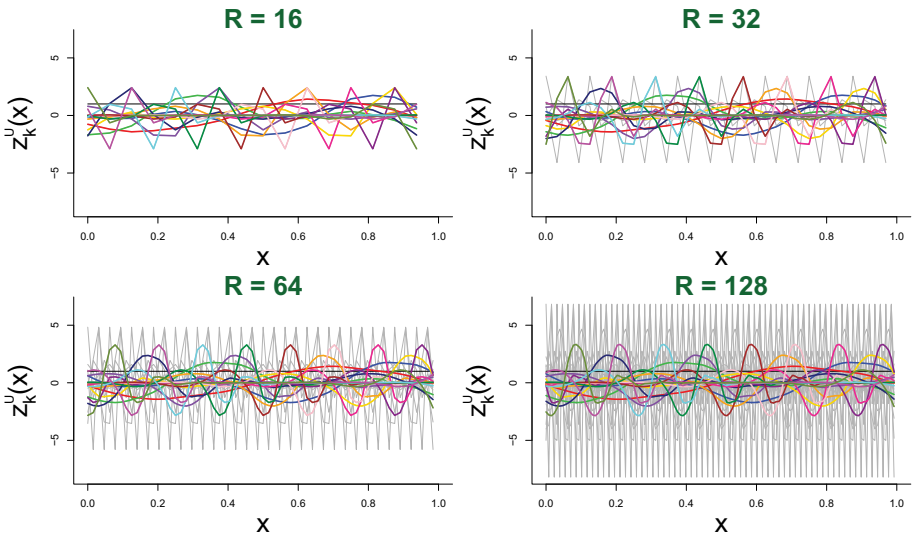

FIG 8. *Illustration of accurate approximation of z_k^U for $K = 15$. In each panel the z_k^U for $1 \le k \le 15$ are colored, whilst z_k^U for $k+1 \le k \le R-1$ are grey. As R increases the accuracy with which the colored functions can be approximated also increases.*

The use of penalized wavelet bases with the hierarchical structure is predicated on the belief that, for many signals of interest, higher-frequency basis functions can be ignored and that L can be set at a number considerably lower than $\log_2(n)$. In the penalized spline literature, Hastie [34] and Ruppert, Wand and Carroll [56] (Section 3.12) justify the omission of higher-frequency basis functions using the eigen-decomposition of the smoother matrix and the term *low-rank*, corresponding to the rank of the smoother matrix, is often used to describe this aspect of penalized splines.

We have constructed an example which suggest that the low-rank argument also applies to penalized wavelets. Consider the case of noiseless regression data generated according to

$$y_i = f_{\mathrm{WO}}(x_i), \qquad 1 \le i \le n,$$

where $x_i = (i-1)/n$, $n = 2^{12} = 4096$ and the function f_{WO}, introduced in this article and named after the initials of the authors' surnames, is given by

$$\begin{aligned} f_{\mathrm{WO}}(x) \quad \equiv \quad 18\Big[& \sqrt{x(1-x)}\sin(1.6\pi/(x+0.2)) + 0.4\, I(x > 0.13) \\ & - 0.7\, I(0.32 < x < 0.38) + 0.43\{(1 - |(x-0.65)/0.03|)_+\}^4 \\ & + 0.42\{(1 - |(x-0.91)/0.015|)_+\}^4\Big], \qquad 0 < x < 1. \end{aligned} \tag{20}$$

Here, and elsewhere, $I(\mathcal{P}) = 1$ if $\mathcal{P}$ is true and zero otherwise. Let $\boldsymbol{C}_L = [\mathbf{1}\ \boldsymbol{Z}_L]$ be the design matrix consisting of a column of ones for the constant term and

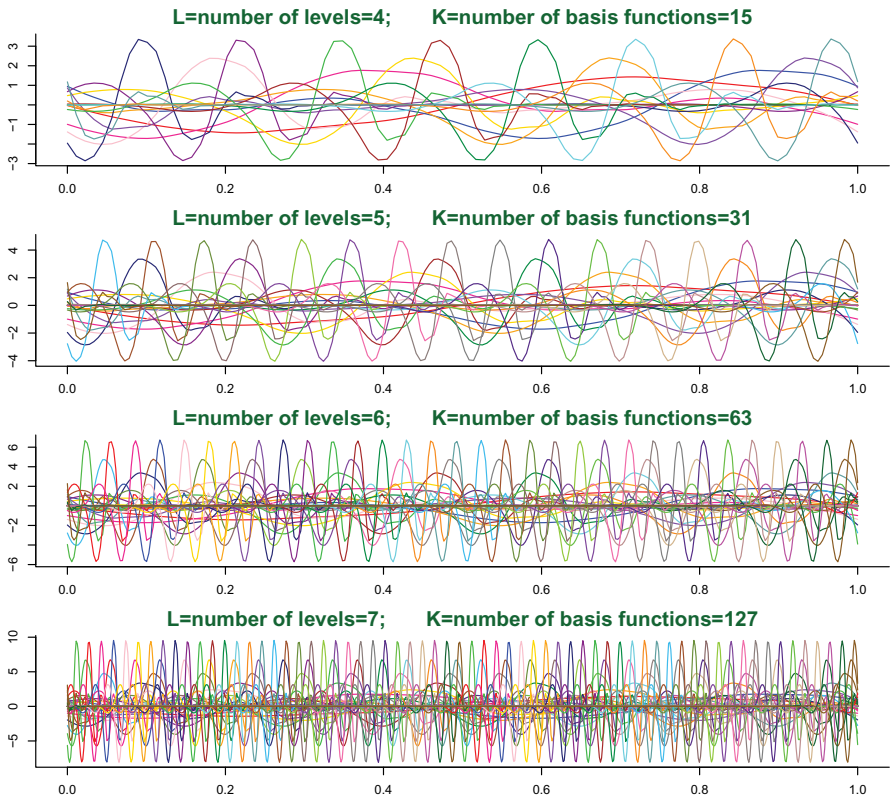

FIG 9. *Default penalized wavelet bases with varying values of* $K = 2^L - 1$.

our default wavelet basis functions evaluated at the x_is. Figure 10 shows the least squares regression fits

$$\widehat{\boldsymbol{y}} = \boldsymbol{C}_L(\boldsymbol{C}_L^T\boldsymbol{C}_L)^{-1}\boldsymbol{C}_L^T\boldsymbol{y}$$

and corresponding R^2 values. Notice the diminished returns as measured by R^2 when L is increased. An R^2 of 99.0% is achieved with only $2^7 - 1 = 127$ wavelet basis functions. It appears that that $L = 8$ ($K = 255$) is adequate for recovery for this particular signal, regardless of the sample size.

3.2. Fitting via penalized least squares

A generalization of the penalized spline criterion (5) is

$$\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{Z}\boldsymbol{u}\|^2 + \sum_{k=1}^{K} \wp_\lambda\left(|u_k|\right) \tag{21}$$

where $\wp(\cdot)$ is a non-decreasing function on $[0, \infty)$. For penalized splines, the choice $\wp_\lambda(x) = \lambda\, x^2$ is usually adequate, and has the advantage of admitting the

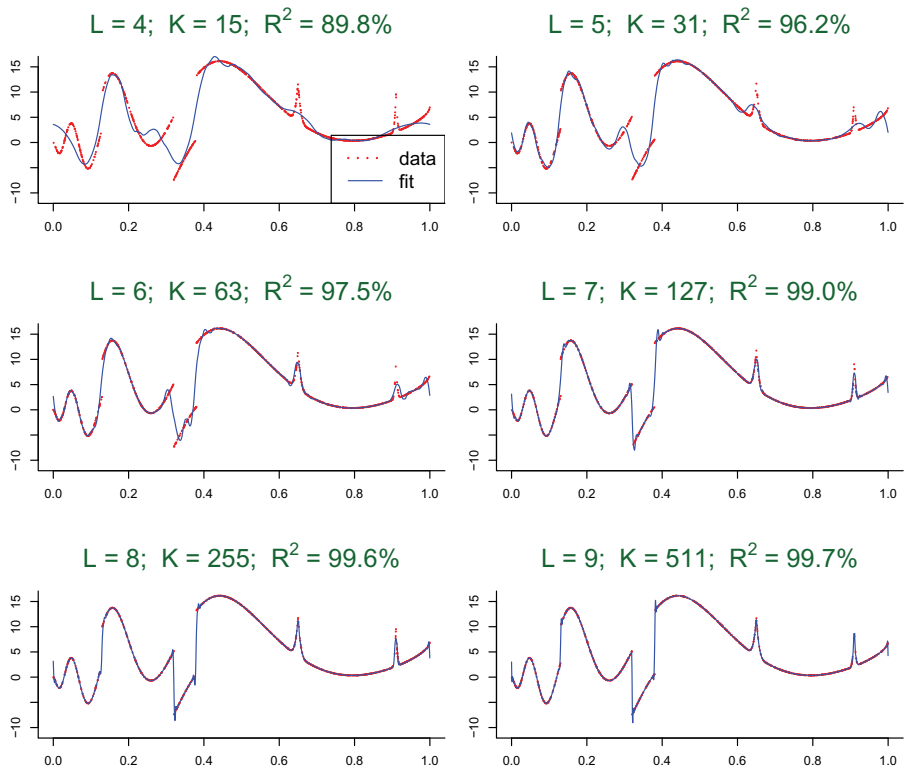

FIG 10. *Illustration of the ability of penalized wavelet basis functions with number of levels* $L \ll \log_2(n)$ *to estimate the* f_{WO} *function. In this case* $n = 2^{12} = 4096$ *and the data are observed without noise. Ordinary least squares is used for the fitting and the resultant* R^2 *value is shown.*

closed form solution (6). For wavelets, a more appropriate choice is $\wp_\lambda(x) = \lambda x$ since the corresponding L_1 penalty invokes a sparse solution. The L_1 penalty corresponds to the *least absolute shrinkage selection operator (LASSO)* (Tibshirani [61]) applied to the basis functions. Algorithms for solving (21) when $\wp_\lambda(x) = \lambda x$ are given in Osborne, Presnell and Turlach [50] and Efron *et al.* [20]. The algorithm in Efron *et al.* [20] efficiently computes the solutions over a grid of λ values.

There are several other possible contenders for $\wp_\lambda(\cdot)$. These include

$$\wp_\lambda(x) = \begin{cases} \lambda x^q, \ q < 1, & \textit{bridge} \text{ penalty} \\ \lambda^2 - (x-\lambda)^2 I(x < \lambda), & \textit{hard thresholding} \text{ penalty} \\ \mathrm{SCAD}(x; \lambda, a), \quad a > 2, & \textit{smoothly clipped absolute deviation (SCAD)} \text{ penalty} \\ \lambda \int_0^x (1 - t/a)_+ \, dt, \quad a > 0, & \textit{minimax concave} \text{ penalty} \end{cases} \tag{22}$$

where

$$\begin{aligned}\text{SCAD}(x;\lambda,a) \quad &\equiv \quad \lambda\, x\, I(x \le \lambda) - \frac{x^2 - 2a\lambda\, x + \lambda^2}{2(a-1)} I(\lambda < x \le a\lambda) \\ &\qquad + \tfrac{1}{2}(a+1)\lambda^2\, I(x > a\lambda).\end{aligned}$$

In each case $\wp_\lambda(x)$ is non-convex in x. Primary references for each of the penalties in (22) are, in order, Frank and Friedman [26], Donoho and Johnstone [17], Fan and Li [23] and Zhang [74].

Antoniadis and Fan [3] study the properties of wavelet nonparametric regression estimators for several such penalties. In particular, they provide a theorem that links the shape of $\wp_\lambda$ to the properties of the penalized least squares solution. The essence of this result is that non-convex penalties are sparseness-inducing. This sparseness property allows penalized wavelets to better handle jumps and jagged features. Figure 12 in Section 3.4 displays penalized least squares fits for three choices of $\wp_\lambda$.

3.3. Effective degrees of freedom

Penalized least squares with non-quadratic penalties does not lead to an explicit expression for the fitted values $\widehat{f}_\lambda(x_i)$ which means that the effective degrees of freedom $\text{edf}(\lambda)$, given by (8), is generally not tractable. In particular, $\widehat{f}_\lambda(x_i)$ is not a linear in the y_is and (9) no longer applies. However Zou, Hastie and Tibshirani [78] derived an unbiased estimator for $\text{edf}(\lambda)$ in the case of the L_1, or LASSO, penalty. For penalized wavelets their results lead to the following *estimated* effective degrees of freedom:

$$\widehat{\text{edf}}(\lambda) = 1 + (\text{number of non-zero } \widehat{u}_k\text{s when the penalty parameter is } \lambda). \quad (23)$$

Zou *et al.* [78] also point out that $\widehat{\text{edf}}(\lambda)$ is not unbiased for other penalties such as SCAD. Hence, effective degrees of freedom estimation is an open problem for penalized wavelets with non-L_1 penalization.

Figure 11 shows four L_1-penalized wavelet fits to data simulated according to

$$y_i = f_{\text{WO}}(x_i) + \varepsilon_i, \qquad 1 \le i \le 2000,$$

where the x_is are uniformly distributed on the unit interval and $\varepsilon_i \overset{\text{ind.}}{\sim} N(0,1)$. For this example it is seen that $\widehat{\text{edf}}(\lambda) = 100$ is the most visually pleasing among the four fits. This is much larger than the best $\text{edf}(\lambda)$ value of 12 for the example in Figure 3, and is to be expected given the complexity of the signal.

Figures 1 and 11 each include at least one visually pleasing penalized wavelet fit to simulated data sets. However, in each case, the error variance is relatively small and the sample size is quite large. If the error variance is increased by even a modest amount, whilst keeping the sample size fixed, then the quality of the penalized wavelet fit tends to deteriorate quite quickly in comparison with penalized splines. This phenomenon has been observed in the wavelet nonparametric regression literature. See, for example, Figure 6 of Marron *et al.* [43].

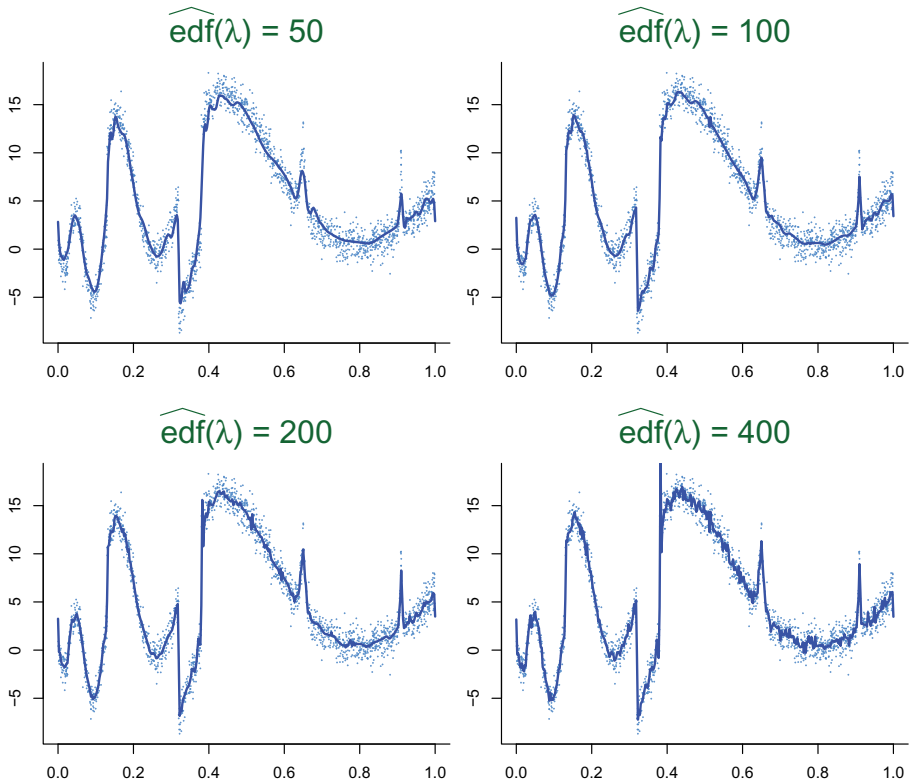

FIG 11. *Penalized wavelet fits to a simulated data set with four different values of the estimate effective degrees of freedom* $\widehat{edf}(\lambda)$.

3.4. Penalty parameter selection

As discussed in Section 2.3, many popular smoothing parameter selection methods trade off residual sum of squares against effective degrees of freedom. The same principle can be translated to penalized wavelets using the estimated effective degrees of freedom described in Section 3.3. For example, (23) suggests the estimated generalized cross-validation criterion:

$$\widehat{\text{GCV}}(\lambda) = \text{RSS}(\lambda)/[\{n - \widehat{\text{edf}}(\lambda)\}^2],$$

for selection of λ. In the case of L_1 penalization, the use of $\widehat{\text{edf}}(\lambda)$ in $\widehat{\text{GCV}}(\lambda)$ is justified by the theory of Zou *et al.* [78]. For other types of penalization, use of $\widehat{\text{GCV}}(\lambda)$ is somewhat tenuous. As mentioned in Section 3.4, k-fold cross-validation is always an option for selection of λ.

In Figure 12 we display three automatic penalized wavelet estimates for regression data of size $n = 500$ simulated from (20) with $N(0, 1)$ noise added. The

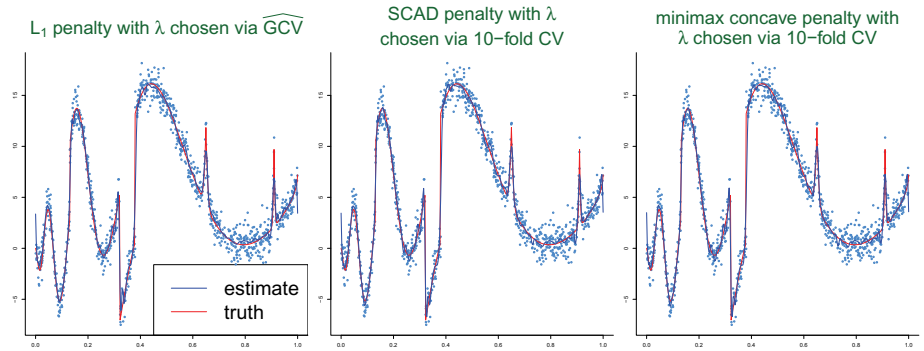

FIG 12. *Automatic penalized wavelet fits to the f_{WO} mean function with L_1 penalization and $\widehat{GCV}$ penalty parameter selection (left panel), SCAD penalization with 10-fold cross-validation penalty parameter selection (middle panel) and minimax concave penalization with 10-fold cross-validation penalty parameter selection (right panel). In each panel, the estimate is shown in blue and the true regression function is shown in red.*

estimates were obtained using (1) L_1 penalization with λ chosen to minimize $\widehat{\text{GCV}}(\lambda)$, (2) SCAD penalization with λ chosen via 10-fold cross-validation (CV) and (3) minimax concave penalization with λ chosen the same way. For these data, the estimates are seen to be quite similar. The R software used to produce Figure 12 is discussed in Section 6.

3.5. Fitting via frequentist mixed model representation

Penalized wavelet analogues of (10) take the general form

$$\boldsymbol{y}|\boldsymbol{u} \sim N(\boldsymbol{X\beta} + \boldsymbol{Zu}, \sigma_\varepsilon^2 \boldsymbol{I}), \quad u_k|\,\sigma_u, \boldsymbol{\theta} \overset{\text{ind.}}{\sim} p(u_k; \sigma_u, \boldsymbol{\theta}). \tag{24}$$

where $p(\cdot; \sigma_u, \boldsymbol{\theta})$ is a symmetric density function with scale parameter σ_u and shape parameter $\boldsymbol{\theta}$. There are numerous options for the choice of this density function. Some of them are:

$$\begin{aligned}
p(u;1) &= \tfrac{1}{2}\exp(-|u|) && \text{(Laplace)}\\
p(u;1,w) &= w\{\tfrac{1}{2}\exp(-|u|)\} + (1-w)\,\delta_0(u) && \text{(Laplace-Zero)}\\
p(u;1) &= (2\pi^3)^{-1/2}\exp(u^2/2)(-1)\text{Ei}(-u^2/2) && \text{(Horseshoe)}\\
p(u;1,\lambda) &= \frac{\lambda\, 2^\lambda \Gamma(\lambda+\frac{1}{2})}{\pi^{1/2}}\exp(u^2/4)\, D_{-2\lambda-1}(|u|) && \text{(Normal-Exponential -Gamma)}
\end{aligned} \tag{25}$$

The Laplace density is an obvious candidate because of its connection with L_1 penalization. Johnstone and Silverman [39] make a strong case for the use of penalty densities such as the Laplace-Zero family. The Horseshoe and Normal-Exponential-Gamma density functions correspond to non-convex penalization

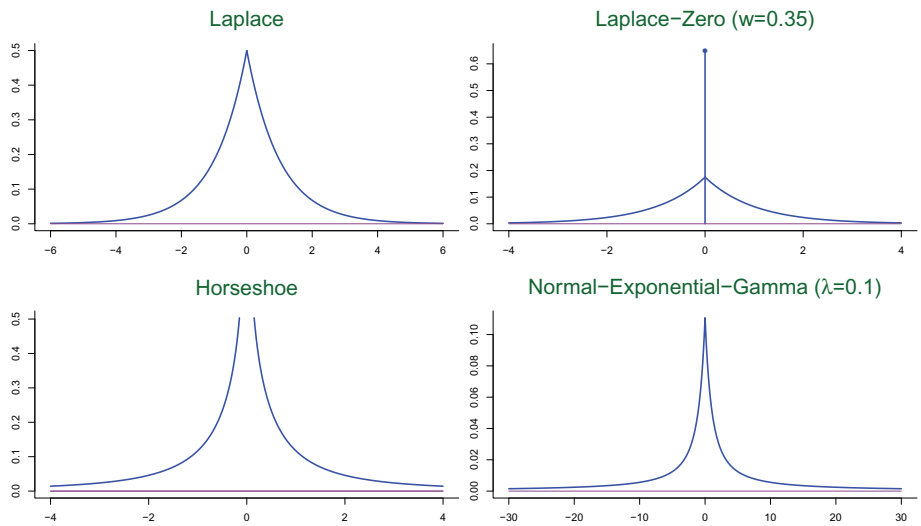

FIG 13. *Plots of the density functions listed in (25).*

and have been proposed in the wide data regression literature by, respectively, Carvalho, Polson and Scott [12] and Griffin and Brown [32]. The definitions involve the special functions Ei, the exponential integral function, and D_ν, the parabolic cylinder function of order ν. For these special functions, we follow the definitions used by Gradshteyn and Ryzhik [30]. Figure 13 plots the density functions listed at (25).

We now focus attention on the first and simplest of these penalty density function, the Laplace. Note the penalized least squares estimator of $\boldsymbol{u}$ with L_1 penalty $\lambda \sum_{k=1}^{K} |u_k|$ corresponds to the conditional *mode* of $\boldsymbol{u}$ given $\boldsymbol{y}$. The best (mean squared error) predictor of $\boldsymbol{u}$ is the conditional *mean*:

$$\widetilde{\boldsymbol{u}} \equiv E(\boldsymbol{u}|\boldsymbol{y}) = \frac{\int_{\mathbb{R}^K} \boldsymbol{u} \exp[-\frac{1}{2\sigma_\varepsilon^2}\{\|\boldsymbol{Z}\boldsymbol{u}\|^2 - 2\boldsymbol{u}^T\boldsymbol{Z}^T(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})\} - \frac{1}{\sigma_u}\boldsymbol{1}^T|\boldsymbol{u}|]\, d\boldsymbol{u}}{\int_{\mathbb{R}^K} \exp[-\frac{1}{2\sigma_\varepsilon^2}\{\|\boldsymbol{Z}\boldsymbol{u}\|^2 - 2\boldsymbol{u}^T\boldsymbol{Z}^T(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})\} - \frac{1}{\sigma_u}\boldsymbol{1}^T|\boldsymbol{u}|]\, d\boldsymbol{u}}.$$

For general $\boldsymbol{Z}$ this expression for $\widetilde{\boldsymbol{u}}$ cannot be reduced any further. However, if

$$\boldsymbol{Z}^T\boldsymbol{Z} = \alpha^2\boldsymbol{I} \quad \text{for some constant } \alpha > 0 \tag{26}$$

then a closed form expression for $\widetilde{\boldsymbol{u}}$ materializes. Appendix B contains the details. Whilst (26) does not hold for general regression data sets, it holds approximately when the x_is are close to being equally spaced or uniformly distributed. It holds *exactly* when n is a power of 2 and the x_is are equally spaced with $a = \min(x_i)$ and $b = \{n\max(x_i) - \min(x_i)\}/(n-1)$. Hence, the formulae in Appendix B could be used to perform approximate best prediction of $\boldsymbol{u}$ and maximum likelihood estimation of $\boldsymbol{\beta}$, σ_ε and σ_u.

The quality of penalized wavelet regression according to frequentist mixed model approaches, such as that using the formulae in Appendix B, is yet to be

studied in any depth. Apart from the fact that viability relies on conditions such as (26) approximately holding, there is the concern that the non-sparseness of the wavelet coefficient estimates may result in overly wiggly fits. In Sections 3.6 and 3.7 it is seen that Bayesian computing methods, MCMC and MFVB, with the random effects density containing a point mass at zero, such as the Laplace-Zero density, overcome this problem.

3.6. Fitting via Bayesian inference and Markov Chain Monte Carlo

Penalized wavelet analogues of (12) take the generic form:

$$\begin{aligned} &\boldsymbol{y}|\boldsymbol{\beta},\boldsymbol{u},\sigma_\varepsilon \sim N(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{Z}\boldsymbol{u},\sigma_\varepsilon^2\boldsymbol{I}), \quad u_k|\,\sigma_u,\boldsymbol{\theta} \overset{\text{ind.}}{\sim} p(u_k|\,\sigma_u,\boldsymbol{\theta}),\\ &\boldsymbol{\beta}\sim N(\boldsymbol{0},\sigma_\beta^2\boldsymbol{I}), \quad \sigma_u\sim \text{Half-Cauchy}(A_u), \quad \sigma_\varepsilon\sim \text{Half-Cauchy}(A_\varepsilon), \end{aligned} \tag{27}$$

where, $p(\,\cdot\,|\sigma_u,\boldsymbol{\theta})$ could be any of the random effects density functions contemplated in Section 3.5 such of those listed in (25) and displayed in Figure 13. (Note the use of the vertical line (|) rather than a semi-colon (;) since σ_u and $\boldsymbol{\theta}$ and now random.) In (27) we have not specified the form of the prior distribution on the shape parameter $\boldsymbol{\theta}$. This may be a fixed distribution, or involve further hierarchical modelling.

We have experimented with the choice of $p(\,\cdot\,|\,\sigma_u,\boldsymbol{\theta})$. The choice corresponding to L_1, or LASSO-type, penalization is the Laplace density function

$$p(u_k|\,\sigma_u) = (2\sigma_u)^{-1}\,\exp(-|u_k|/\sigma_u) \tag{28}$$

but the Bayes estimator of $\boldsymbol{u}$ is not sparse and, as a consequence, the resulting fits tend to be overly wiggly. However, sparse solutions are produced by a Laplace-Zero density function

$$p(u_k|\,\sigma_u,\rho) = \rho\,(2\sigma_u)^{-1}\,\exp(-|u_k|/\sigma_u) + (1-\rho)\,\delta_0(u_k) \tag{29}$$

where ρ is a random variable over $[0,1]$. Such priors are advocated by Johnstone and Silverman [39]. These authors also provide theoretical justification for use of (29). The fact that $E(u_k|\boldsymbol{y})$ is often exactly zero translates to better handling of jumps and sharp features in the underlying signal. Hence, for the remainder of this article we work with (29) for Bayesian penalized wavelets. Concurrent doctoral thesis research by Sarah E. Neville, supervised by the first author, is investigating the performance of the Horseshoe and Normal-Exponential-Gamma priors in this wavelet context.

MCMC handling of (29) is aided by introducing specially tailored auxiliary variables v_k, γ_k and b_k. Suppose that $u_k = \gamma_k\,v_k$ where

$$\gamma_k|\,\rho \overset{\text{ind.}}{\sim} \text{Bernoulli}(\rho), \quad v_k|b_k \overset{\text{ind.}}{\sim} N(0,\sigma_u^2/b_k) \;\text{ and }\; b_k \overset{\text{ind.}}{\sim} \text{Inverse-Gamma}(1,\tfrac{1}{2}).$$

Then, courtesy of elementary distribution theory manipulations, $u_k|\rho$ has density function (29). Because v_k is conditionally Gaussian, it is advantageous to

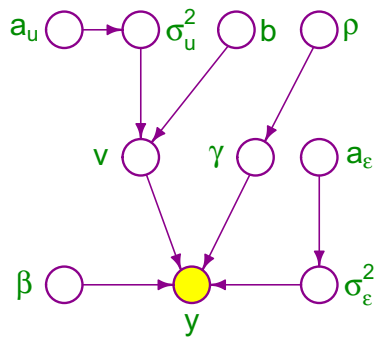

FIG 14. *Directed acyclic graph representation of the auxiliary variable Bayesian penalized wavelet model (30). The shaded node corresponds to observed data.*

work with the pairs (v_k, γ_k) rather than (u_k, γ_k) in the MCMC sampling strategy. As in Section 2.6 we use (13) to allow easier handling of the Half-Cauchy priors on σ_u and σ_ε. Let $\boldsymbol{a} \odot \boldsymbol{b}$ denote the elementwise product of equi-sized vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ and $\text{diag}(\boldsymbol{b})$ be the diagonal matrix with diagonal entries corresponding to those of $\boldsymbol{b}$. The full model, with appropriate auxiliary variables, is then

$$\begin{gathered}
\boldsymbol{y}|\boldsymbol{\beta}, \boldsymbol{v}, \sigma_\varepsilon^2 \sim N(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}(\boldsymbol{\gamma} \odot \boldsymbol{v}), \sigma_\varepsilon^2 \boldsymbol{I}), \;\; \boldsymbol{v}|\, \sigma_u^2, \boldsymbol{b} \sim N(\boldsymbol{0}, \sigma_u^2\, \text{diag}(\boldsymbol{b})^{-1}), \\
\sigma_u^2|\, a_u \sim \text{Inverse-Gamma}(\tfrac{1}{2}, a_u^{-1}), \;\; \sigma_\varepsilon^2|\, a_\varepsilon \sim \text{Inverse-Gamma}(\tfrac{1}{2}, a_\varepsilon^{-1}), \\
a_u \sim \text{Inverse-Gamma}(\tfrac{1}{2}, A_u^{-2}), \;\; a_\varepsilon \sim \text{Inverse-Gamma}(\tfrac{1}{2}, A_\varepsilon^{-2}), \\
\boldsymbol{\beta} \sim N(\boldsymbol{0}, \sigma_\beta^2 \boldsymbol{I}), \quad b_k \overset{\text{ind.}}{\sim} \text{Inverse-Gamma}(1, \tfrac{1}{2}), \\
\gamma_k|\, \rho \overset{\text{ind.}}{\sim} \text{Bernoulli}(\rho), \quad \rho \sim \text{Beta}(A_\rho, B_\rho).
\end{gathered} \tag{30}$$

The last of these distributional specifications corresponds to conjugate Beta priors being placed on Bernoulli probability parameters. The hyperparameters A_ρ and B_ρ are positive numbers corresponding to the usual parametrization of the Beta distribution. Figure 14 shows the DAG corresponding to (30).

As with penalized splines, the vector of fitted values is the posterior mean

$$\widehat{\boldsymbol{f}} = E(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{u}|\boldsymbol{y}) = \boldsymbol{X}\, E(\boldsymbol{\beta}|\boldsymbol{y}) + \boldsymbol{Z}\, E(\boldsymbol{u}|\boldsymbol{y}) = \boldsymbol{X}\, E(\boldsymbol{\beta}|\boldsymbol{y}) + \boldsymbol{Z}\, E(\boldsymbol{\gamma} \odot \boldsymbol{v}|\boldsymbol{y}).$$

The full conditionals for Markov chain Monte Carlo can be shown to be:

$$\begin{aligned}
\left[\begin{array}{c} \boldsymbol{\beta} \\ \boldsymbol{v} \end{array}\right] \Big| \text{rest} \quad \sim \quad & N\Bigg(\left(\sigma_\varepsilon^{-2} \boldsymbol{C}_{\boldsymbol{\gamma}}^T \boldsymbol{C}_{\boldsymbol{\gamma}} + \left[\begin{array}{cc} \sigma_\beta^{-2} & \boldsymbol{0} \\ \boldsymbol{0} & \sigma_u^{-2}\, \text{diag}(\boldsymbol{b}) \end{array}\right] \right)^{-1} \sigma_\varepsilon^{-2} \boldsymbol{C}_{\boldsymbol{\gamma}}^T \boldsymbol{y}, \\
& \left(\sigma_\varepsilon^{-2} \boldsymbol{C}_{\boldsymbol{\gamma}}^T \boldsymbol{C}_{\boldsymbol{\gamma}} + \left[\begin{array}{cc} \sigma_\beta^{-2} & \boldsymbol{0} \\ \boldsymbol{0} & \sigma_u^{-2}\, \text{diag}(\boldsymbol{b}) \end{array}\right] \right)^{-1} \Bigg),
\end{aligned}$$

$$\begin{aligned}
\sigma_u^2|\text{rest} &\sim \text{Inverse-Gamma}\left(\tfrac{1}{2}(K+1), \tfrac{1}{2}\boldsymbol{v}^T\text{diag}(\boldsymbol{b})\boldsymbol{v} + a_u^{-1}\right),\\
\sigma_\varepsilon^2|\text{rest} &\sim \text{Inverse-Gamma}\left(\tfrac{1}{2}(n+1), \tfrac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{Z}_{\boldsymbol{\gamma}}\boldsymbol{v}\|^2 + a_\varepsilon^{-1}\right),\\
a_u|\text{rest} &\sim \text{Inverse-Gamma}\left(1, \sigma_u^{-2} + A_u^{-2}\right),\\
a_\varepsilon|\text{rest} &\sim \text{Inverse-Gamma}\left(1, \sigma_\varepsilon^{-2} + A_\varepsilon^{-2}\right),\\
b_k|\text{rest} &\overset{\text{ind.}}{\sim} \text{Inverse-Gaussian}\left(\sigma_u/|v_k|, 1\right),\\
\rho|\text{rest} &\sim \text{Beta}(A_\rho + \gamma_\bullet, B_\rho + K - \gamma_\bullet)\\
\text{and } \gamma_k|\text{rest} &\overset{\text{ind.}}{\sim} \text{Bernoulli}\left(\frac{\exp(\eta_k)}{1+\exp(\eta_k)}\right)
\end{aligned}$$

where

$$\gamma_\bullet \equiv \sum_{k=1}^K \gamma_k\,, \qquad \boldsymbol{C}_{\boldsymbol{\gamma}} \equiv [\boldsymbol{X}\ \ \boldsymbol{Z}\,\text{diag}(\boldsymbol{\gamma})]$$

and

$$\begin{aligned}
\eta_k \quad \equiv \quad & -\tfrac{1}{2\sigma_\varepsilon^2}\big[\|\boldsymbol{Z}_k\|^2 v_k^2 - 2\boldsymbol{y}^T\,\boldsymbol{Z}_k\,v_k + 2\boldsymbol{X}^T\,\boldsymbol{Z}_k\,\boldsymbol{\beta}\,v_k\\
& \qquad + 2\,\boldsymbol{Z}_k^T\boldsymbol{Z}_{-k}\{\boldsymbol{\gamma}_{-k}\odot(v_k\boldsymbol{v}_{-k})\}\big] + \text{logit}(\rho).
\end{aligned}$$

Here, and elsewhere,

$$\boldsymbol{\gamma}_{-k} \equiv [\gamma_1, \ldots, \gamma_{k-1}, \gamma_{k+1}, \ldots, \gamma_K]^T.$$

The vector $\boldsymbol{v}_{-k}$ is defined analogously.

As for the Bayesian penalized spline model (14) all full conditional distributions are standard and MCMC reduces to ordinary Gibbs sampling.

3.7. Fitting via mean field variational Bayes

As in the penalized spline case, we now seek fast deterministic approximate inference for (30) based on MFVB. A tractable solution arises if we impose the product restriction

$$\begin{aligned}
q(\boldsymbol{\beta}, \boldsymbol{v}, \boldsymbol{b}, \boldsymbol{\gamma}, \rho, \sigma_u^2, \sigma_\varepsilon^2, a_u, a_\varepsilon) = \\
q(\boldsymbol{\beta}, \boldsymbol{v})\, q(\boldsymbol{b})\, q(a_u, a_\varepsilon, \rho)\, q(\sigma_u^2, \sigma_\varepsilon^2)\, \textstyle\prod_{k=1}^K q(\gamma_k).
\end{aligned} \tag{31}$$

Note that *induced* factorizations (e.g., Bishop [8], Section 10.2.5) lead to solution having the additional product structure

$$q(\boldsymbol{\beta}, \boldsymbol{v})\, q(\sigma_u^2)\, q(\sigma_\varepsilon^2)\, q(a_u)\, q(a_\varepsilon)\, q(\rho) \prod_{k=1}^K \{\, q(b_k)\, q(\gamma_k)\}.$$

Then, as shown in Appendix D,

$$\begin{array}{l} q^*(\boldsymbol{\beta},\boldsymbol{v}) \text{ is a Multivariate Normal density function,} \\ q^*(\sigma_u^2),\ q^*(\sigma_\varepsilon^2),\ q^*(a_u) \text{ and } q^*(a_\varepsilon) \text{ are each} \\ \text{Inverse Gamma density functions,} \\ q^*(\boldsymbol{b}) \text{ is a product of Inverse Gaussian density functions,} \\ q^*(\gamma_k), 1 \le k \le K, \text{ are Bernoulli probability mass functions,} \\ q^*(\rho) \text{ is a Beta density function.} \end{array} \tag{32}$$

Similarly to the penalized spline case, let $\boldsymbol{\mu}_{q(\boldsymbol{\beta},\boldsymbol{v})}$ and $\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{v})}$ denote the mean vector and covariance matrix for $q^*(\boldsymbol{\beta},\boldsymbol{v})$ and $A_{q(\sigma_u^2)}$ and $B_{q(\sigma_u^2)}$ denote the shape and rate parameters for $q^*(\sigma_u^2)$ with similar definitions for the parameters in $q^*(\sigma_\varepsilon^2)$, $q^*(a_u)$ and $q^*(a_\varepsilon)$. Then the optimal values of these parameters are determined from Algorithm 4, which is justified in Appendix D. Note that $\psi(x) \equiv \frac{d}{dx}\log\{\Gamma(x)\}$ denotes the digamma function.

Convergence of Algorithm 4 can be monitored using the following expression for the lower bound on the marginal log-likelihood:

$$\begin{aligned} \log \underline{p}(\boldsymbol{y};q) \;=\; & \tfrac{1}{2}(K+1) + \tfrac{1}{2}(K-n)\log(2\pi) - K\log(2) - 2\log(\pi) \\ & + \log\Gamma(\tfrac{1}{2}(K+1)) + \log\Gamma(\tfrac{1}{2}(n+1)) - \tfrac{1}{2}\log(\sigma_{\boldsymbol{\beta}}^2) \\ & - \log(A_u) - \log(A_\varepsilon) - \tfrac{1}{2\sigma_{\boldsymbol{\beta}}^2}\{\|\boldsymbol{\mu}_{q(\boldsymbol{\beta})}\|^2 + \mathrm{tr}(\boldsymbol{\Sigma}_{q(\boldsymbol{\beta})})\} \\ & + \tfrac{1}{2}\log|\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{v})}| - \tfrac{1}{2}(K+1)\log\{B_{q(\sigma_u^2)}\} \\ & - \tfrac{1}{2}(n+1)\log\{B_{q(\sigma_\varepsilon^2)}\} - \tfrac{1}{2}\sum_{k=1}^{K}\{1/\mu_{q(b_k)}\} \\ & - \log(\mu_{q(1/\sigma_u^2)} + A_u^{-2}) - \log(\mu_{q(1/\sigma_\varepsilon^2)} + A_\varepsilon^{-2}) \\ & + \mu_{q(1/\sigma_u^2)}\mu_{q(1/a_u)} + \mu_{q(1/\sigma_\varepsilon^2)}\mu_{q(1/a_\varepsilon)} \\ & - \sum_{k=1}^{K}[\mu_{q(\gamma_k)}\log\{\mu_{q(\gamma_k)}\} + (1-\mu_{q(\gamma_k)})\log\{1-\mu_{q(\gamma_k)}\}] \\ & + \log\Gamma(A_\rho + \mu_{q(\gamma_\bullet)}) + \log\Gamma(B_\rho + K - \mu_{q(\gamma_\bullet)}) \\ & - \log\Gamma(A_\rho + B_\rho + K) + \log(A_\rho + B_\rho) - \log\Gamma(A_\rho) - \log\Gamma(B_\rho). \end{aligned}$$

Illustration of Bayesian penalized wavelet regression, using both the MCMC and MFVB, is provided by Figure 15. The data were generated according to

$$y_i = f_{\mathrm{WO}}(x_i) + \varepsilon_i$$

with $x_i = (i-1)/n$ and $\varepsilon_i \overset{\text{ind.}}{\sim} N(0,1)$. The hyperparameters were set to be $\sigma_{\boldsymbol{\beta}}^2 = 10^8$, $A_u = A_\varepsilon = 25$, $A_\rho = 1$ and $B_\rho = 9$. MCMC samples of size 10000 were generated. The first 5000 values were discarded and the second 5000 values were thinned by a factor of 5. The MFVB iterations were terminated when the relative change in $\log \underline{p}(\boldsymbol{y};q)$ fell below 10^{-10}.

Algorithm 4 *Mean field variational Bayes algorithm for the determination of the optimal parameters in $q^*(\boldsymbol{\beta},\boldsymbol{v})$, $q^*(\boldsymbol{\gamma})$, $q^*(\sigma_u^2)$ and $q^*(\sigma_\varepsilon^2)$ for the Bayesian penalized wavelet model (30).*

Initialize: $\mu_{q(1/\sigma_\varepsilon^2)}$, $\mu_{q(1/\sigma_u^2)}$, $\mu_{q(1/a_\varepsilon)}$, $\mu_{q(1/a_u)}$, $\boldsymbol{\mu}_{q(\boldsymbol{b})}$, $\boldsymbol{\mu}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})}$ and $\boldsymbol{\Omega}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})}$.
Cycle:

$$\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{v})} \leftarrow \left(\mu_{q(1/\sigma_\varepsilon^2)}(\boldsymbol{C}^T\boldsymbol{C}) \odot \boldsymbol{\Omega}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})} + \begin{bmatrix} \sigma_\beta^{-2} & \boldsymbol{0} \\ \boldsymbol{0} & \mu_{q(1/\sigma_u^2)}\text{diag}(\boldsymbol{\mu}_{q(\boldsymbol{b})}) \end{bmatrix}\right)^{-1}$$

$$\boldsymbol{\mu}_{q(\boldsymbol{\beta},\boldsymbol{v})} \leftarrow \mu_{q(1/\sigma_\varepsilon^2)}\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{v})}\text{diag}\{\boldsymbol{\mu}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})}\}\boldsymbol{C}^T\boldsymbol{y}$$

For $k = 1,\ldots,K$:

$$\mu_{q(b_k)} \leftarrow \{\mu_{q(1/\sigma_u^2)}(\sigma^2_{q(v_k)} + \mu^2_{q(v_k)})\}^{-1/2}$$

$$\begin{aligned}\eta_{q(\gamma_k)} \quad \leftarrow \quad & -\tfrac{1}{2}\,\mu_{q(1/\sigma_\varepsilon^2)}\Big[\|\boldsymbol{Z}_k\|^2\{\sigma^2_{q(v_k)} + \mu^2_{q(v_k)}\} - 2\boldsymbol{Z}_k^T\boldsymbol{y}\,\mu_{q(v_k)} \\ & +2\boldsymbol{Z}_k^T\boldsymbol{X}\left\{(\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{v})})_{1,1+k} + \mu_{q(\boldsymbol{\beta})}\mu_{q(v_k)}\right\} \\ & +2\boldsymbol{Z}_k^T\boldsymbol{Z}_{-k}\left\{(\boldsymbol{\mu}_{q(\boldsymbol{\gamma})})_{-k} \odot \{(\boldsymbol{\Sigma}_{q(\boldsymbol{v})})_{-k,k} + \mu_{q(v_k)}(\boldsymbol{\mu}_{q(\boldsymbol{v})})_{-k}\}\right\}\Big] \\ & +\psi(A_\rho + \mu_{q(\gamma_\bullet)}) - \psi(B_\rho + K - \mu_{q(\gamma_\bullet)})\end{aligned}$$

$$\mu_{q(\gamma_k)} \leftarrow \frac{\exp(\eta_{q(\gamma_k)})}{1+\exp(\eta_{q(\gamma_k)})}$$

$$\boldsymbol{\mu}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})} \leftarrow \begin{bmatrix} 1 \\ \boldsymbol{\mu}_{q(\boldsymbol{\gamma})} \end{bmatrix} \quad ; \quad \mu_{q(\gamma_\bullet)} \leftarrow \textstyle\sum_{k=1}^K \mu_{q(\gamma_k)}$$

$$\boldsymbol{\Omega}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})} \leftarrow \text{diag}\{\boldsymbol{\mu}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})} \odot (\boldsymbol{1} - \boldsymbol{\mu}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})})\} + \boldsymbol{\mu}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})}\,\boldsymbol{\mu}^T_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})}$$

$$\mu_{q(1/a_\varepsilon)} \leftarrow 1/\{\mu_{q(1/\sigma_\varepsilon^2)} + A_\varepsilon^{-2}\} \quad ; \quad \mu_{q(1/a_u)} \leftarrow 1/\{\mu_{q(1/\sigma_u^2)} + A_u^{-2}\}$$

$$\begin{aligned}B_{q(\sigma_\varepsilon^2)} \leftarrow\ & \mu_{q(1/a_\varepsilon)} + \tfrac{1}{2}\,\|\boldsymbol{y}\|^2 - \boldsymbol{y}^T\boldsymbol{C}\left(\boldsymbol{\mu}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})} \odot \mu_{q(\boldsymbol{\beta},\boldsymbol{v})}\right) \\ & +\tfrac{1}{2}\text{tr}\left(\boldsymbol{C}^T\boldsymbol{C}\ \left[\boldsymbol{\Omega}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})} \odot \left\{\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{v})} + \boldsymbol{\mu}_{q(\boldsymbol{\beta},\boldsymbol{v})}\boldsymbol{\mu}^T_{q(\boldsymbol{\beta},\boldsymbol{v})}\right\}\right]\right)\end{aligned}$$

$$B_{q(\sigma_u^2)} \leftarrow \mu_{q(1/a_u)} + \tfrac{1}{2}\textstyle\sum_{k=1}^K \mu_{q(b_k)}\{\sigma^2_{q(v_k)} + \mu^2_{q(v_k)}\}$$

$$\mu_{q(1/\sigma_u^2)} \leftarrow \tfrac{1}{2}(K+1)/B_{q(\sigma_u^2)} \quad ; \quad \mu_{q(1/\sigma_\varepsilon^2)} \leftarrow \tfrac{1}{2}(n+1)/B_{q(\sigma_\varepsilon^2)}$$

until the increase in $\underline{p}(\boldsymbol{y};q)$ is negligible.

The left panels of Figure 15 show that the MCMC converges quite well. The upper right panel shows that MFVB converges after 79 iterations. R language implementation of the MCMC fit took about 45 minutes on the first author's laptop (Mac OS X; 2.33 GHz processor, 3 GBytes of random access memory) whereas the MFVB one took only 17 seconds with the same programming language. The lower right panel of Figure 15 indicates that the two fits are quite close.

In Figure 15 we zoom in on the fits for $0.6 \le x \le 0.7$. It is seen that both the MFVB and MCMC fits are quite close in terms of both point estimation and interval estimation. This suggests that MFVB is quite accurate for penalized wavelet model (30), although further simulation checks are warranted.

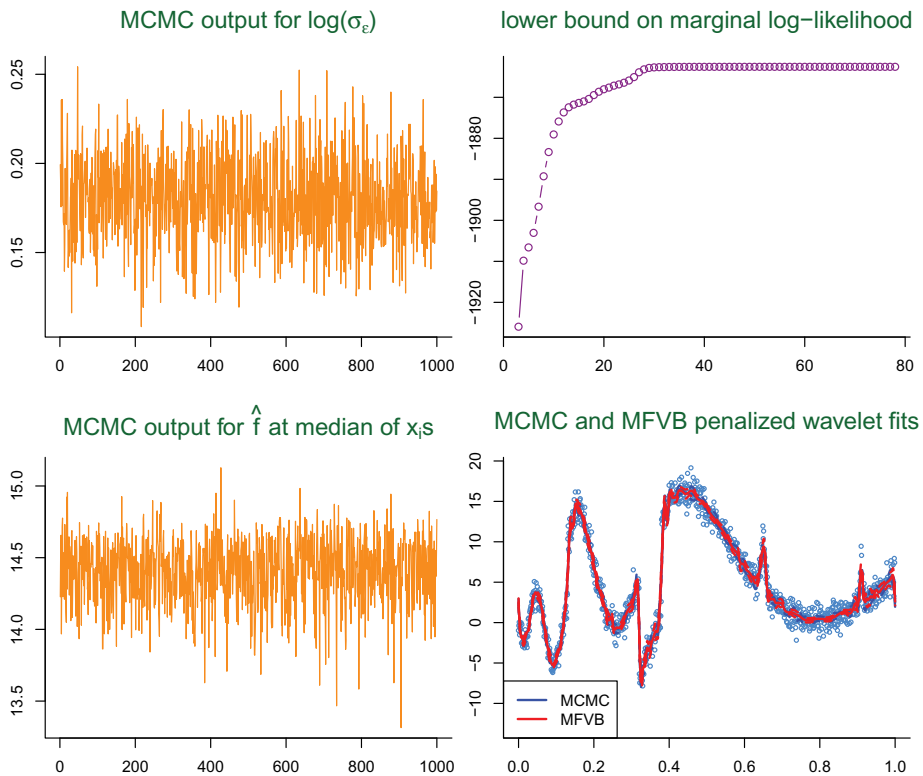

FIG 15. *Left panels: MCMC output for fitting Bayesian penalized wavelet model to simulated data. The upper left panel is for* $\log(\sigma_\varepsilon)$. *The lower left panel is for the estimated function at the median of the* x_i*s. Upper right panel: successive values of* $\log \underline{p}(\boldsymbol{y}; q)$ *to monitor convergence of the MFVB algorithm. Lower right panel: Fitted function estimates and pointwise 95% credible sets for both MCMC and MFVB approaches.*

4. Choice of penalized wavelet basis size

A remaining problem attached to our proposed new wavelet nonparametric paradigm is the choice of $L = \log_2(K+1)$. As demonstrated by Figure 10, it is often quite reasonable to have $L \ll \log_2(n)$. In the case of penalized spline regression is it usually enough to work with simple rules such as $K = \max(35, n/4)$. But this rule sometimes needs modification if it is believed that the underlying function is particularly wiggly. The same dilemma applies to penalized wavelets. Indeed, casual experimentation suggests that more care needs to be taken with choice of penalized wavelet basis size compared with the penalized splines counterpart. Further research is required to formalize the extent of the problem and to devise high-quality solutions. In the present article we flag it as an issue and make some brief remarks on possible approaches to choosing the penalized wavelet basis size.

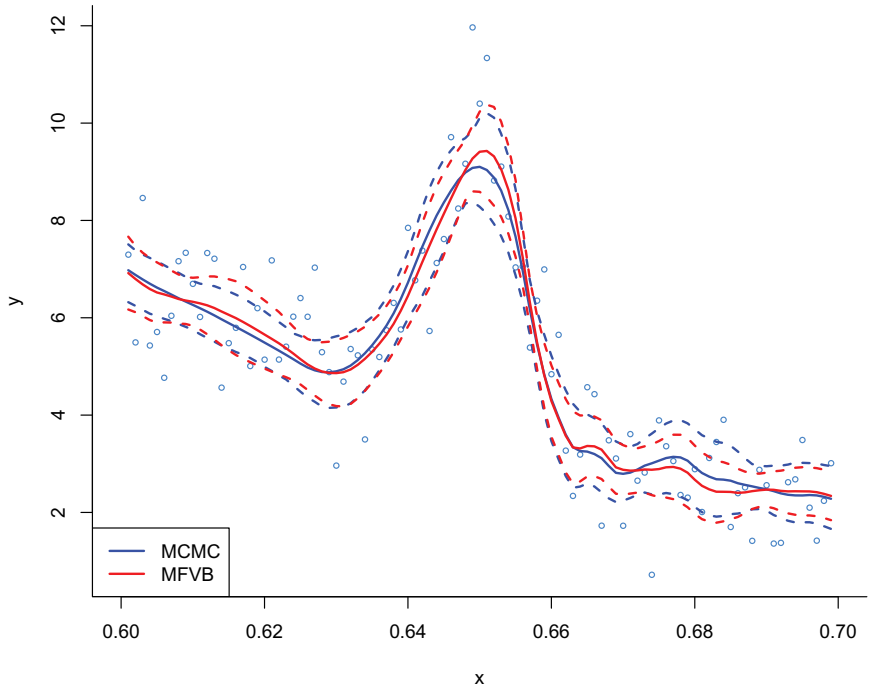

FIG 16. *Zoomed display of fits shown in lower right panel of Figure 15. The solid curves are the function estimates based on the pointwise posterior means and the dashed curves are pointwise 95% credible sets.*

In the low-noise situation, simple graphical checks could be used to guide the choice of L. If a more automatic method is required then each of the approaches to penalized wavelet fitting described in Sections 3.4 to 3.7 lend themselves to data-based rules choosing L. For example, an attractive by-product of the MFVB approach is an approximation to the marginal log-likelihood, which can be used to guide the choice of L.

Another possible approach to choice of L involves adaptation of classical wavelet thresholding methodology. If n is a power of 2 and the x_is are equally-spaced then the Discrete Wavelet Transform can be used to quickly obtain the n coefficients of the full set of wavelet basis functions, as elucidated by (19). Simple thresholds such as $\widehat{\sigma}_\varepsilon \sqrt{2\log_e(n)}$ (Donoho and Johnstone [17]) can be used to select L. Specifically, the L could correspond to the largest level having coefficients exceeding the threshold. Further development is required for general x_i.

5. Semiparametric regression extensions

The preceding sections put wavelets on the same footing as splines and, hence, facilitate straightforward embedding of penalized wavelets into semiparametric regression models (e.g. Ruppert, Wand and Carroll [56, 57]). Any existing semiparametric regression model containing penalized splines can be modified to instead contain penalized wavelets if there is reason to believe that the underlying functional effect is jagged. It is also conceivable that some components in the model are better handled using penalized splines, whilst penalized wavelets are more appropriate for other components. Illustrations of such composite models are given in Sections 5.2 and 5.3.

Bayesian approaches to semiparametric regression, with MCMC or MFVB fitting, are particularly amenable to such adaptation since replacement of splines by wavelets simply means modification of the corresponding DAG. Since the MCMC and MFVB algorithm updates are localized on the DAG (e.g. Wand *et al.* [67], Section 3) the spline to wavelet replacement can be made by replacement of penalized spline node structure (as in Figure 4) by penalized wavelet node structure (as in Figure 14).

The remainder of this section provides some concrete illustrations of such spline to wavelet adaptations. Given the ease with which these adaptations can be made using MCMC or MFVB, we will confine description to these approaches. The non-Bayesian approaches of Sections 2 and 3 can, at least in theory, be treated analogously. However, some of the implementational details may require further research.

5.1. Non-Gaussian response models

Non-Gaussian response models involving penalized wavelets can be treated analogously to those involving penalized splines. The only differences are the design matrices $\boldsymbol{X}$ and $\boldsymbol{Z}$ and the type of penalization applied to entries of the $\boldsymbol{u}$ vector. The non-Gaussian aspect means that penalized least squares is no longer appropriate and penalized log-likelihood should be used instead. Fan and Song [24] describe some of the properties of penalized log-likelihood estimators for penalties such as L_1 and SCAD. The extension of penalized wavelets to non-Gaussian response models via penalized log-likelihood applies quite generally. However, we will restrict further discussion to the important binary response case. See Antondiadis and Leblanc [4] for a classical wavelet treatment of binary response regression.

Figure 17 shows penalized wavelet estimates for binary response data simulated according to

$$\text{logit}\{P(y_i = 1)\} = 0.15\, f_{\text{WO}}(x_i) - \tfrac{1}{2}, \quad 1 \le i \le n. \tag{33}$$

where $x_i = (i-1)/n$ and n is set at 1000, 10000 and 100000. The estimates were obtained using the SCAD-penalized negative logistic log-likelihood

$$-\boldsymbol{y}^T(\boldsymbol{X\beta} + \boldsymbol{Zu}) + \boldsymbol{1}^T \log\{\boldsymbol{1} + \exp(\boldsymbol{X\beta} + \boldsymbol{Zu})\} + \lambda \sum_{k=1}^{K} \text{SCAD}(|u_k|, 3) \tag{34}$$

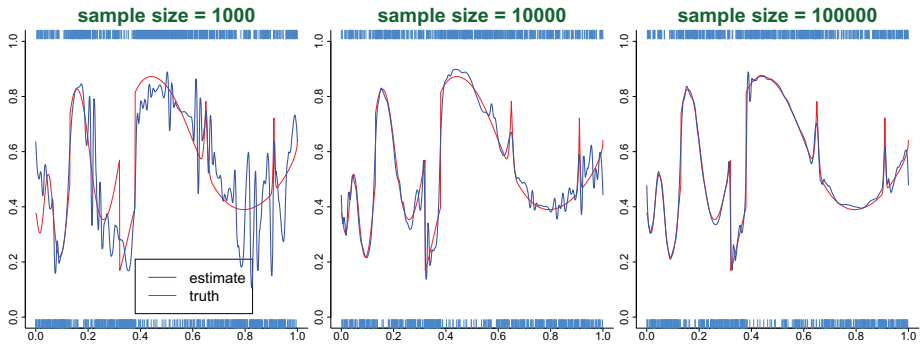

FIG 17. *Illustration of difficulty of binary response penalized wavelet regression. In each case, the penalized wavelet estimates of the mean, or probability, function are obtained using SCAD-penalized logistic log-likelihood with the penalty parameter chosen via 10-fold cross-validation and shown in blue. The true probability function is shown in red. In the leftmost panel* ($n = 1000$ *the data are shown as rugs. The rugs in the other two panels* ($n = 10000, 100000$) *correspond to sub-samples of size 1000.*

and λ chosen via 10-fold cross-validation. The R functions `ncvreg()` and `cv.ncvreg()` within the package `ncvreg` (Breheny [9]) were used to obtain the fits in Figure 17. The design matrices in $\boldsymbol{X}$ and $\boldsymbol{Z}$ (34) have exactly the same form as those used in Section 3 for Gaussian response penalized wavelet regression. A striking feature of Figure 17 is that quite large sample sizes are required to obtain visually pleasing estimates. This is a consequence of the low signal-to-noise ratio that is an inherent part of binary response regression and the difficulty that wavelets have in high-noise situations, as mentioned in Section 3.3.

Bayesian binary response penalized spline regression, with a probit rather than logit link function, has a Gibbsian MCMC solution courtesy of the auxiliary variable construction of Albert and Chib [1] (e.g. Ruppert *et al.* [56], Section 16.5.1). The same is true for penalized wavelets using, for example, a Laplace-Zero prior (29) on the wavelet coefficients. Specifically, consider the model

$$\begin{aligned} y_i \,|\, \boldsymbol{\beta}, \boldsymbol{u} &\overset{\text{ind.}}{\sim} \text{Bernoulli}\{\Phi((\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{u})_i)\}, \\ p(\boldsymbol{u} \,|\, \sigma_u, \gamma_k) &= \textstyle\prod_{k=1}^K \left\{\gamma_k \, (2\sigma_u)^{-1} \, \exp(-|u_k|/\sigma_u) + (1-\gamma_k)\, \delta_0(u_k)\right\}, \\ \boldsymbol{\beta} &\sim N(\boldsymbol{0}, \sigma_\beta^2 \boldsymbol{I}), \quad \sigma_u \sim \text{Half-Cauchy}(A_u), \\ \gamma_k \,|\, \rho &\overset{\text{ind.}}{\sim} \text{Bernoulli}(\rho), \quad \rho \sim \text{Beta}(A_\rho, B_\rho). \end{aligned} \tag{35}$$

Here $\Phi(x) \equiv \int_{-\infty}^x \phi(t)\, dt$ is the standard normal cumulative distribution function, with $\phi(x) \equiv (2\pi)^{-1/2} \exp(-x^2/2)$ denoting the corresponding density function. Introduce the vector of auxiliary variables $\boldsymbol{a} = (a_1, \ldots, a_n)$ such that $y_i = 1$ if and only if $a_i \geq 0$ and

$$\boldsymbol{a} \,|\, \boldsymbol{\beta}, \boldsymbol{u} \sim N(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{u}, \boldsymbol{I}).$$

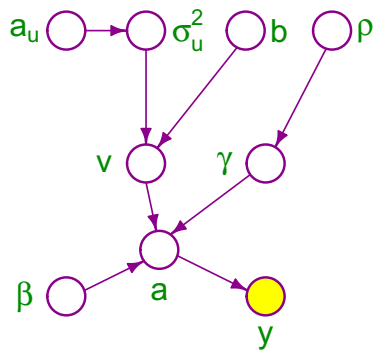

FIG 18. *Directed acyclic graph representation of the probit Bayesian penalized spline model (36). The shaded node corresponds to observed data.*

Then, with the auxiliary variables $\boldsymbol{v}$, $\boldsymbol{b}$ and a_u as in Section 3.6, we can write (35) as

$$\begin{aligned} y_i|a_i \overset{\text{ind.}}{\sim} \text{Bernoulli}(I(a_i \geq 0)), \quad \boldsymbol{a}|\boldsymbol{\beta}, \boldsymbol{v} \sim N(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}(\boldsymbol{\gamma} \odot \boldsymbol{v}), \boldsymbol{I}), \\ \boldsymbol{v}|\, \sigma_u^2, \boldsymbol{b} \sim N(\boldsymbol{0}, \sigma_u^2 \,\text{diag}(\boldsymbol{b})^{-1}), \\ \boldsymbol{\beta} \sim N(\boldsymbol{0}, \sigma_\beta^2 \boldsymbol{I}), \quad \sigma_u^2|\, a_u \sim \text{Inverse-Gamma}(\tfrac{1}{2}, a_u^{-1}), \\ a_u \sim \text{Inverse-Gamma}(\tfrac{1}{2}, A_u^{-2}), \quad b_k \overset{\text{ind.}}{\sim} \text{Inverse-Gamma}(1, \tfrac{1}{2}), \\ \gamma_k|\, \rho \overset{\text{ind.}}{\sim} \text{Bernoulli}(\rho), \quad \rho \sim \text{Beta}(A_\rho, B_\rho). \end{aligned} \tag{36}$$

Figure 18 is the DAG corresponding to (36).

The full conditionals for Markov chain Monte Carlo can be shown to be:

$$\begin{aligned} \begin{bmatrix} \boldsymbol{\beta} \\ \boldsymbol{v} \end{bmatrix} \Big| \text{rest} \;\sim\; & N\left(\left(\boldsymbol{C}_\gamma^T \boldsymbol{C}_\gamma + \begin{bmatrix} \sigma_\beta^{-2} & \boldsymbol{0} \\ \boldsymbol{0} & \sigma_u^{-2}\,\text{diag}(\boldsymbol{b}) \end{bmatrix} \right)^{-1} \boldsymbol{C}_\gamma^T \boldsymbol{a}, \right. \\ & \left. \left(\boldsymbol{C}_\gamma^T \boldsymbol{C}_\gamma + \begin{bmatrix} \sigma_\beta^{-2} & \boldsymbol{0} \\ \boldsymbol{0} & \sigma_u^{-2}\,\text{diag}(\boldsymbol{b}) \end{bmatrix} \right)^{-1} \right), \\ a_i|\text{rest} \overset{\text{ind.}}{\sim} & \begin{cases} N(\{\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}(\boldsymbol{\gamma} \odot \boldsymbol{v})\}_i, 1) \text{ truncated on } (-\infty, 0), & y_i = 0 \\ N(\{\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}(\boldsymbol{\gamma} \odot \boldsymbol{v})\}_i, 1) \text{ truncated on } (0, \infty), & y_i = 1 \end{cases} \\ \sigma_u^2|\text{rest} \;\sim\; & \text{Inverse-Gamma}\left(\tfrac{1}{2}(K+1), \tfrac{1}{2}\boldsymbol{v}^T \text{diag}(\boldsymbol{b})\, \boldsymbol{v} + a_u^{-1}\right), \\ a_u|\text{rest} \;\sim\; & \text{Inverse-Gamma}\left(1, \sigma_u^{-2} + A_u^{-2}\right), \\ b_k|\text{rest} \overset{\text{ind.}}{\sim} & \text{Inverse-Gaussian}\left(\sigma_u/|v_k|, 1\right), \\ \rho|\text{rest} \;\sim\; & \text{Beta}(A_\rho + \gamma_\bullet, B_\rho + K - \gamma_\bullet) \\ \text{and } \gamma_k|\text{rest} \overset{\text{ind.}}{\sim} & \text{Bernoulli}\left(\frac{\exp(\eta_k)}{1 + \exp(\eta_k)} \right) \end{aligned}$$

where $\boldsymbol{C}_{\boldsymbol{\gamma}}$ has the same definition as before and

$$\eta_k \equiv -\tfrac{1}{2}\left[\|\boldsymbol{Z}_k\|^2 v_k^2 - 2\boldsymbol{a}^T\,\boldsymbol{Z}_k\,v_k + 2\boldsymbol{X}^T\,\boldsymbol{Z}_k\,\boldsymbol{\beta}\,v_k + 2\,\boldsymbol{Z}_k^T\boldsymbol{Z}_{-k}\{\boldsymbol{\gamma}_{-k}\odot(v_k\boldsymbol{v}_{-k})\}\right] + \text{logit}(\rho).$$

The corresponding MFVB approach, summarised in Algorithm 5, requires only closed form updates. The optimal q^* density functions for all variables except $\boldsymbol{a}$ take the same forms as those given for the Gaussian response case in Section 3.7. Appendix E contains underpinning for Algorithm 5.

Figure 19 illustrates these MCMC and MFVB approaches to estimating the underlying probability function from data generated according to (33) with the sample size set at $n = 50000$. (Note, however, that the i linear predictor $(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{Z}\boldsymbol{u})_i$ estimates $\Phi^{-1}(\text{logit}^{-1}(0.15 f_{\text{WO}}(x_i)-\frac{1}{2}))$ since (35) is a *probit* regression model). Both approaches are seen to give similar fits.

Algorithm 5 *Mean field variational Bayes algorithm for the determination of the optimal parameters in $q^*(\boldsymbol{\beta},\boldsymbol{v})$, $q^*(\boldsymbol{\gamma})$ and $q^*(\sigma_u^2)$ for the probit Bayesian penalized wavelet model (36).*

Initialize: $\mu_{q(1/\sigma_u^2)}$, $\mu_{q(1/a_u)}$, $\boldsymbol{\mu}_{q(\boldsymbol{b})}$, $\boldsymbol{\mu}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})}$ and $\boldsymbol{\Omega}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})}$.
Cycle:

$$\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{v})} \leftarrow \left((\boldsymbol{C}^T\boldsymbol{C})\odot\boldsymbol{\Omega}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})} + \begin{bmatrix} \sigma_\beta^{-2} & \boldsymbol{0} \\ \boldsymbol{0} & \mu_{q(1/\sigma_u^2)}\text{diag}(\mu_{q(\boldsymbol{b})}) \end{bmatrix}\right)^{-1}$$

$$\boldsymbol{\mu}_{q(\boldsymbol{\beta},\boldsymbol{v})} \leftarrow \boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{v})}\,\text{diag}\{\boldsymbol{\mu}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})}\}\boldsymbol{C}^T\boldsymbol{\mu}_{q(\boldsymbol{a})}$$

$$\boldsymbol{\mu}_{q(\boldsymbol{a})} \leftarrow \boldsymbol{X}\mu_{q(\boldsymbol{\beta})} + \boldsymbol{Z}(\boldsymbol{\mu}_{q(\boldsymbol{\gamma})}\odot\boldsymbol{\mu}_{q(\boldsymbol{v})}) + \frac{(2\boldsymbol{y}-1)\odot\phi(\boldsymbol{X}\mu_{q(\boldsymbol{\beta})}+\boldsymbol{Z}(\boldsymbol{\mu}_{q(\boldsymbol{\gamma})}\odot\boldsymbol{\mu}_{q(\boldsymbol{v})}))}{\Phi((2\boldsymbol{y}-1)\odot\{\boldsymbol{X}\mu_{q(\boldsymbol{\beta})}+\boldsymbol{Z}(\boldsymbol{\mu}_{q(\boldsymbol{\gamma})}\odot\boldsymbol{\mu}_{q(\boldsymbol{v})})\})}$$

For $k = 1,\ldots,K$:

$$\mu_{q(b_k)} \leftarrow \{\mu_{q(1/\sigma_u^2)}(\sigma^2_{q(v_k)}+\mu^2_{q(v_k)})\}^{-1/2}$$

$$\begin{aligned}\eta_{q(\gamma_k)} \quad \leftarrow \quad & -\tfrac{1}{2}\Big[\|\boldsymbol{Z}_k\|^2\{\sigma^2_{q(v_k)}+\mu^2_{q(v_k)}\} - 2\boldsymbol{Z}_k^T\boldsymbol{\mu}_{q(\boldsymbol{a})}\,\mu_{q(v_k)} \\ & +2\boldsymbol{Z}_k^T\boldsymbol{X}\left\{(\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{v})})_{1,1+k}+\mu_{q(\boldsymbol{\beta})}\mu_{q(v_k)}\right\} \\ & +2\boldsymbol{Z}_k^T\boldsymbol{Z}_{-k}\left\{(\boldsymbol{\mu}_{q(\boldsymbol{\gamma})})_{-k}\odot\{(\boldsymbol{\Sigma}_{q(\boldsymbol{v})})_{-k,k}+\mu_{q(v_k)}(\boldsymbol{\mu}_{q(\boldsymbol{v})})_{-k}\}\right\}\Big] \\ & +\psi(A_\rho+\mu_{q(\gamma_\bullet)})-\psi(B_\rho+K-\mu_{q(\gamma_\bullet)})\end{aligned}$$

$$\mu_{q(\gamma_k)} \leftarrow \frac{\exp(\eta_{q(\gamma_k)})}{1+\exp(\eta_{q(\gamma_k)})}$$

$$\boldsymbol{\mu}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})} \leftarrow \begin{bmatrix} 1 \\ \boldsymbol{\mu}_{q(\boldsymbol{\gamma})} \end{bmatrix} \quad ; \quad \mu_{q(\gamma_\bullet)} \leftarrow \textstyle\sum_{k=1}^K \mu_{q(\gamma_k)}$$

$$\boldsymbol{\Omega}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})} \leftarrow \text{diag}\{\boldsymbol{\mu}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})}\odot(\boldsymbol{1}-\boldsymbol{\mu}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})})\}+\boldsymbol{\mu}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})}\,\boldsymbol{\mu}^T_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})}$$

$$B_{q(\sigma_u^2)} \leftarrow \mu_{q(1/a_u)} + \tfrac{1}{2}\textstyle\sum_{k=1}^K \mu_{q(b_k)}\{\sigma^2_{q(v_k)}+\mu^2_{q(v_k)}\}$$

$$\mu_{q(1/\sigma_u^2)} \leftarrow \tfrac{1}{2}(K+1)/B_{q(\sigma_u^2)} \quad ; \quad \mu_{q(1/a_u)} \leftarrow 1/\{\mu_{q(1/\sigma_u^2)}+A_u^{-2}\}$$

until the increase in $\underline{p}(\boldsymbol{y};q)$ is negligible.

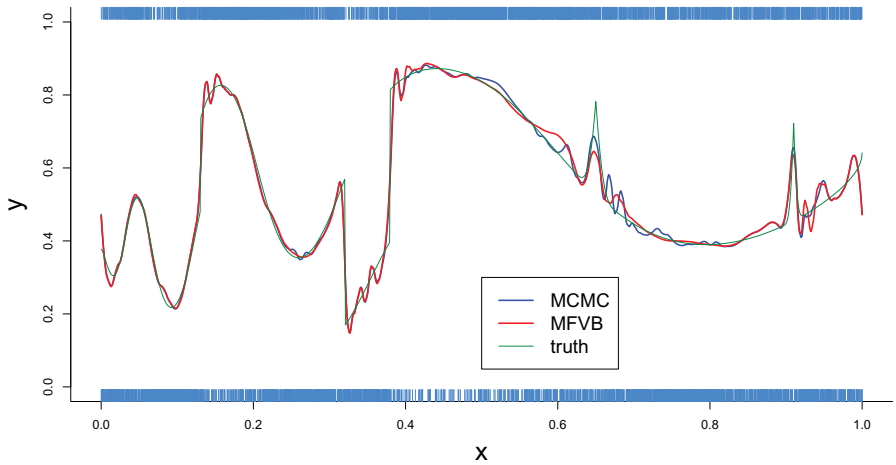

FIG 19. *Bayesian posterior mean estimates of the probability function (green curve) from data generated according to (33) with* $n = 50000$. *The Bayesian estimates were obtained via MCMC (blue curve) and MFVB (red curve). The rugs show a 10% random sub-sample of the data.*

5.2. Additive models and varying coefficient models

Additive models and *varying coefficient models* are a popular extensions of non-parametric regression when several continuous predictor variables are available. If the response is non-Gaussian then the term *generalized additive model* (Hastie and Tibshirani [36], Wood [72]) is commonly used for the former type.

With simplicity in mind, we will restrict discussion to the case of two predictor variables x_1 and x_2. The treatment of the general case is similar, but at the expense of additional notation. Generalized additive models take the generic form

$$g\{E(\boldsymbol{y})\} = f_1(\boldsymbol{x}_1) + f_2(\boldsymbol{x}_2) \tag{37}$$

whilst a varying coefficient model for such data is

$$g\{E(\boldsymbol{y})\} = f_1(\boldsymbol{x}_1) + f_2(\boldsymbol{x}_1) \odot \boldsymbol{x}_2. \tag{38}$$

Here g is a link function and f_1 and f_2 are arbitrary "well-behaved" functions. See, for example, Ruppert, Wand and Carroll [56], for details on penalized spline fitting of (37) and (38)

Given the preceding sections, the replacement of penalized splines by penalized wavelets is relatively straightforward and is appropriate if there is good reason to believe that either f_1 or f_2 is jagged. Models containing both penalized splines and penalized wavelets are also worthy of consideration.

To amplify this point, and to illustrate the embedding of penalized wavelets into additive models consider data simulated according to

$$y_i = \tfrac{1}{2}\,\Phi(6\,x_{1i} - 3) + \tfrac{1}{3}\,I(x_{2i} \geq 0.6) + \varepsilon_i, \quad 1 \leq i \leq n, \tag{39}$$

where the x_{i1} and x_{2i} are generated as completely independent samples from the Uniform distribution on $(0,1)$ and $\varepsilon_i \overset{\text{ind.}}{\sim} N(0, \sigma_\varepsilon^2)$ for some $\sigma_\varepsilon > 0$. Since, as known by the simulation set-up, the mean responses are a smooth function of the x_{1i}s and a step function of x_{2i}s an appropriate model in this example is

$$y_i = \beta_0 + \beta_1\,x_{1i} + \sum_{k=1}^{K^{\text{spl}}} u_k^{\text{spl}}\,z_k^{\text{spl}}(x_{1i}) + \sum_{k=1}^{K^{\text{wav}}} u_k^{\text{wav}}\,z_k^{\text{wav}}(x_{2i}) + \varepsilon_i$$

where the $z_k^{\text{spl}}(\cdot)$ are spline basis functions and the $z_k^{\text{wav}}(\cdot)$ are wavelet basis functions. Let

$$\boldsymbol{X} = [1\ x_{1i}\ x_{2i}]_{1\leq i\leq n}, \quad \boldsymbol{Z}^{\text{spl}} = \big[\,z_k^{\text{spl}}(x_{1i})\big]_{\substack{1\leq i\leq n\\ 1\leq k\leq K^{\text{spl}}}} \quad \text{and} \quad \boldsymbol{Z}^{\text{wav}} = \big[\,z_k^{\text{wav}}(x_{2i})\big]_{\substack{1\leq i\leq n\\ 1\leq k\leq K^{\text{wav}}}}$$

be the design matrices containing the linear functions, spline basis functions and wavelet basis functions of the data. Note that $\boldsymbol{Z}^{\text{spl}}$ and $\boldsymbol{Z}^{\text{wav}}$ can be obtained, respectively, by application of Algorithm 1 to the x_{1i}s and Algorithm 2 to the x_{2i}s. Given regularization parameters $\lambda^{\text{spl}} > 0$ and $\lambda^{\text{wav}} > 0$, an appropriate estimation strategy is one that minimizes the penalized least squares criterion

$$\big\|\boldsymbol{y} - \boldsymbol{X}\,\boldsymbol{\beta} - \boldsymbol{Z}^{\text{spl}}\boldsymbol{u}^{\text{spl}} - \boldsymbol{Z}^{\text{wav}}\boldsymbol{u}^{\text{wav}}\big\|^2 + \lambda^{\text{spl}}\big\|\boldsymbol{u}^{\text{spl}}\big\|^2 + \lambda^{\text{wav}}\sum_{k=1}^{K^{\text{wav}}} |u_k^{\text{wav}}|. \tag{40}$$

This takes a form similar to the *elastic net* penalty introduced by Zou and Hastie [77], and it is anticipated that the efficient computational algorithm that these authors developed is extendible to (40).

Alternatively a mixed model approach can be used by placing suitable distributions on the spline and wavelet coefficients. We will confine discussion to the Bayesian version of mixed model fitting, in which an appropriate hierarchical Bayesian model is

$$\begin{gathered}
\boldsymbol{y}|\,\boldsymbol{\beta}, \boldsymbol{u}^{\text{spl}}, \boldsymbol{u}^{\text{wav}}, \sigma_\varepsilon \sim N(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}^{\text{spl}}\boldsymbol{u}^{\text{spl}} + \boldsymbol{Z}^{\text{wav}}\boldsymbol{u}^{\text{wav}}, \sigma_\varepsilon^2\,\boldsymbol{I}),\\
\boldsymbol{\beta} \sim N(\boldsymbol{0}, \sigma_\beta^2\boldsymbol{I}), \quad \boldsymbol{u}^{\text{spl}} \sim N(\boldsymbol{0}, (\sigma_u^{\text{spl}})^2\,\boldsymbol{I}),\\
p(\boldsymbol{u}^{\text{wav}}\,|\,\sigma_u^{\text{wav}}, \gamma_k) = \textstyle\prod_{k=1}^{K}\big\{\gamma_k\,(2\sigma_u^{\text{wav}})^{-1}\,\exp(-|u_k^{\text{wav}}|/\sigma_u^{\text{wav}})\\
+ (1-\gamma_k)\,\delta_0(u_k^{\text{wav}})\big\},\\
\sigma_u^{\text{spl}} \sim \text{Half-Cauchy}(A_u^{\text{spl}}), \quad \sigma_u^{\text{wav}} \sim \text{Half-Cauchy}(A_u^{\text{wav}}),\\
\sigma_\varepsilon \sim \text{Half-Cauchy}(A_\varepsilon), \quad \gamma_k|\,\rho \overset{\text{ind.}}{\sim} \text{Bernoulli}(\rho), \quad \rho \sim \text{Beta}(A_\rho, B_\rho).
\end{gathered} \tag{41}$$

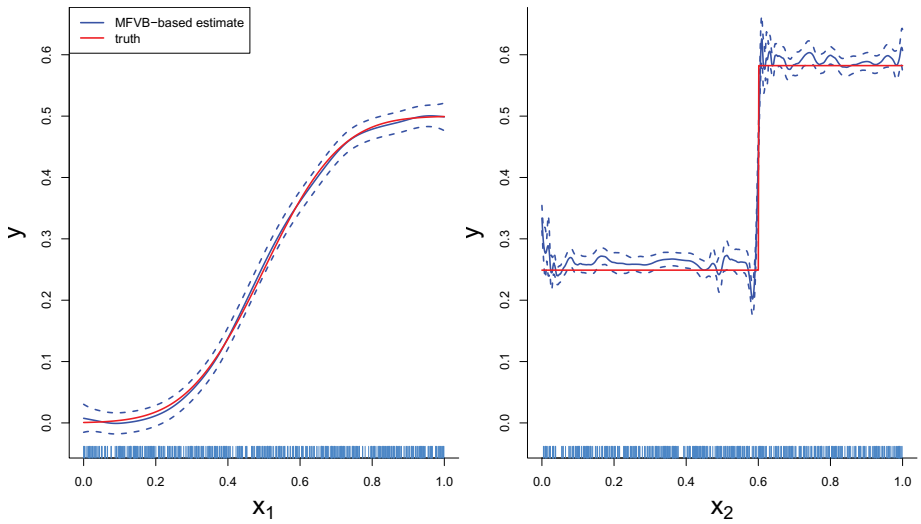

FIG 20. *Illustrative MFVB-based fit for the spline/wavelet additive model (41). The data were simulated from (39) with* $n = 5000$ *and* $\sigma_\varepsilon = 1$. *The true functions are shown in red. The blue solid curves are function estimates based on the pointwise posterior means. The blue dashed curves correspond to pointwise approximate 95% credible sets. All curves in the left panel correspond to the functions of* x_2 *evaluated at the sample mean of the* x_{2i}. *The reverse situation applies to the right panel. The rugs at the base of each panel show, respectively, 10% random sub-samples of the* x_{1i}*s and* x_{2i}*s.*

MCMC and MFVB algorithms for fitting (41) involve relatively straightforward marriage of those given in Sections 2.6, 2.7, 3.6 and 3.7 for Bayesian penalized spline and Bayesian penalized wavelet nonparametric regression.

Figure 20 shows a MFVB fit for (41), where the data is simulated from (39) with $n = 5000$ and $\sigma_\varepsilon = 1$. For this example, the combination of penalized splines and penalized wavelets is seen to capture the true functions quite well.

5.3. Semiparametric longitudinal data analysis

During the last fifteen years there has been much research on the use of splines to handle non-linear effects in the analysis of longitudinal data. See, for example, the *Non-Parametric and Semi-Parametric Methods for Longitudinal Data Analysis* section of Fitzmaurice, Davidian, Verbeke and Molenberghs [25] and the references therein. There is also a smaller literature on the incorporation of wavelets into longitudinal models, with contributions such as Aykroyd and Mardia [6], Morris, Vannucci, Brown and Carroll [46], Morris and Carroll [45] and Zhao and Wu [76]. A feature of the wavelet-based longitudinal data analysis literature is a tendency to work in the coefficient space (e.g. Morris *et al.* [46]). In this section we demonstrate that sound analyses can be conducted using direct approaches, analogous to those in the penalized spline longitudinal data analysis literature.

The penalized wavelet approach laid out in Section 3 facilitates straightforward modification of spline-based longitudinal models to handle data possessing jagged signals. One simply replaces spline basis functions by wavelet basis functions and modifies the penalties on the basis function coefficients. We will provide illustration via a modification of the subject-specific curve penalized spline model developed by Durbán, Harezlak, Wand and Carroll [18]. Earlier variants of this model, based on smoothing splines rather than penalized splines, were developed by Brumback and Rice [10], Wang [68] and Zhang, Lin, Raz and Sowers [75]. The model considered by Durbán *et al.* [18] takes the basic form

$$y_{ij} = f(x_{ij}) + g_i(x_{ij}) + \varepsilon_{ij}, \quad \varepsilon_{ij} \overset{\text{ind.}}{\sim} N(0, \sigma_\varepsilon^2)$$

where, for $1 \le i \le m$ and $1 \le j \le n_i$, (x_{ij}, y_{ij}) denotes the jth predictor/response pair for the ith subject. A Bayesian penalized spline model for f is

$$f(x) = \beta_0 + \beta_1 x + \sum_{k=1}^{K^{\text{gbl}}} u_k^{\text{gbl}} z_k^{\text{gbl}}(x), \qquad u_k^{\text{gbl}} | \sigma_u^{\text{gbl}} \overset{\text{ind.}}{\sim} N(0, (\sigma_u^{\text{gbl}})^2).$$

We could use penalized wavelets for f, but splines will often be adequate for the smoother global mean function. However, the subject-specific deviation functions could be quite jagged, in which case a penalized *wavelet* model such as

$$\begin{aligned} g_i(x) &= U_i + \textstyle\sum_{k=1}^{K^{\text{sbj}}} u_{ik}^{\text{sbj}} z_k^{\text{sbj}}(x), \quad U_i | \sigma_U \overset{\text{ind.}}{\sim} N(0, \sigma_U^2) \\ p(u_{ik}^{\text{sbj}} | \sigma_u^{\text{sbj}}, \gamma_{ik}) &= \gamma_{ik} (2\sigma_u^{\text{sbj}})^{-1} \exp(-|u_{ik}^{\text{sbj}}|/\sigma_u^{\text{sbj}}) + (1-\gamma_{ik}) \delta_0(u_{ik}^{\text{sbj}}) \end{aligned} \tag{42}$$

is appropriate for the subject specific deviations. We complete the model specification with

$$\begin{aligned} &\beta_0, \beta_1 \overset{\text{ind.}}{\sim} N(\mathbf{0}, \sigma_{\boldsymbol{\beta}}^2 \boldsymbol{I}), \quad \sigma_u^{\text{gbl}} \sim \text{Half-Cauchy}(A_u^{\text{gbl}}), \\ &\sigma_u^{\text{sbj}} \sim \text{Half-Cauchy}(A_u^{\text{sbj}}), \quad \sigma_\varepsilon \sim \text{Half-Cauchy}(A_\varepsilon), \\ &\gamma_{ik} | \rho_{ik} \overset{\text{ind.}}{\sim} \text{Bernoulli}(\rho_{ik}), \quad \rho_{ik} \overset{\text{ind.}}{\sim} \text{Beta}(A_\rho, B_\rho). \end{aligned} \tag{43}$$

Other possibilities for the last component (43) are $\rho_i \overset{\text{ind.}}{\sim} \text{Beta}(A_\rho, B_\rho)$ and $\rho \sim \text{Beta}(A_\rho, B_\rho)$, each of which are more natural generalizations of the Bayesian penalized wavelet models given in earlier sections of this article. A limited amount of experimentation with these alternatives led to poor results, whereas (43) has worked well (e.g. Figures 21 and 22 below). Note, however, that the last line of (43) is equivalent to $\gamma_{ik} \overset{\text{ind.}}{\sim} \text{Bernoulli}\big(\frac{A_\rho}{A_\rho + B_\rho}\big)$ which implies that the ρ_{ik} can be omitted. Further research on this part of the model is warranted.

Figure 21 shows data where model (43) is beneficial. The data are from a respiratory pneumonitis study (source: Hart *et al.* [33]) and the panels display the logarithm of normalized fluorodeoxyglucose uptake against radiation dose

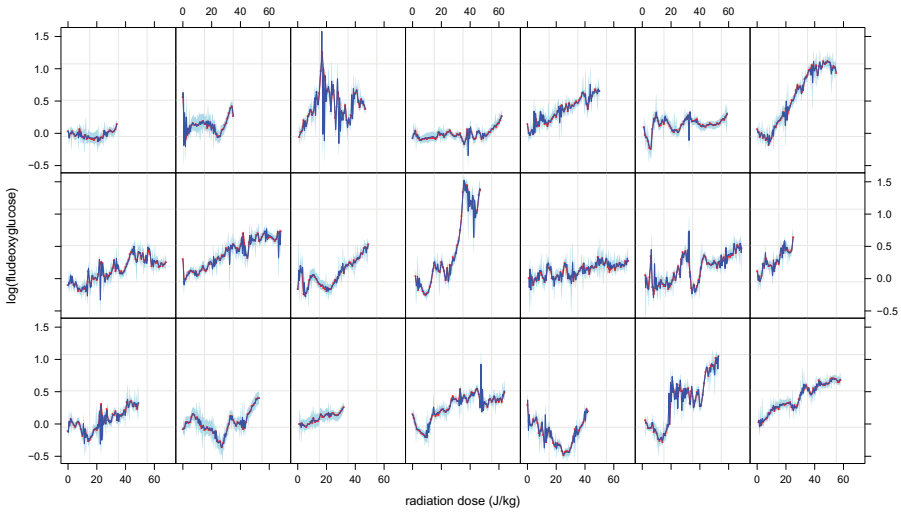

FIG 21. *Logarithm of normalized fluorodeoxyglucose uptake versus radiation dose (J/kg)for each of 21 lung cancer patients (source: Hart et al. [33]), shown as red points. The blue curves are posterior mean fits of the model given by (42) and (43). The light blue shading corresponds to pointwise 95% credible sets.*

for each of 21 lung cancer patients. The red points in Figure 21 show the data. The blue curves correspond to the posterior mean fit of (42) and (43). The light blue shading conveys pointwise 95% credible sets for each fitted curve. These fits were obtained using BUGS (Spiegelhalter *et al.* [58]), accessed from within R via the `BRugs` package (Ligges *et al.* [41]). The BUGS code is listed in Appendix F. A burnin of size 15000 was used, followed by 5000 iterations which were then thinned by a factor 5. The predictor and response data were each linearly transformed to the unit interval and the hyperparameters were set to the values $\sigma^2_{\boldsymbol{\beta}} = 10^8$, $A^{\text{gbl}}_u = A^{\text{sbj}}_u = A_\varepsilon = 25$, $A_\rho = B_\rho = 1$, corresponding to non-informativity. The inverse linear transformation was applied before displaying the fits.

Figure 22 highlights aspects of the fit shown in Figure 21. The top left panel is the penalized spline-based estimate of the global mean function f. The bottom left panel displays the penalized wavelet-based subject specific deviations. These are quite irregular and appear to benefit from the use of wavelets rather than splines. The top right panel is a zoomed version of one of the panels from Figure 21 and the bottom right panels shows the residuals against the fitted values. The residual plot shows no pronounced patterns, indicating that the model fits the data well.

In this article we do not delve into the scientific questions associated with these data and only use it to illustrate penalized wavelet-based semiparametric longitudinal data analysis. Further work is planned on the scientific ramifications of such analyses.

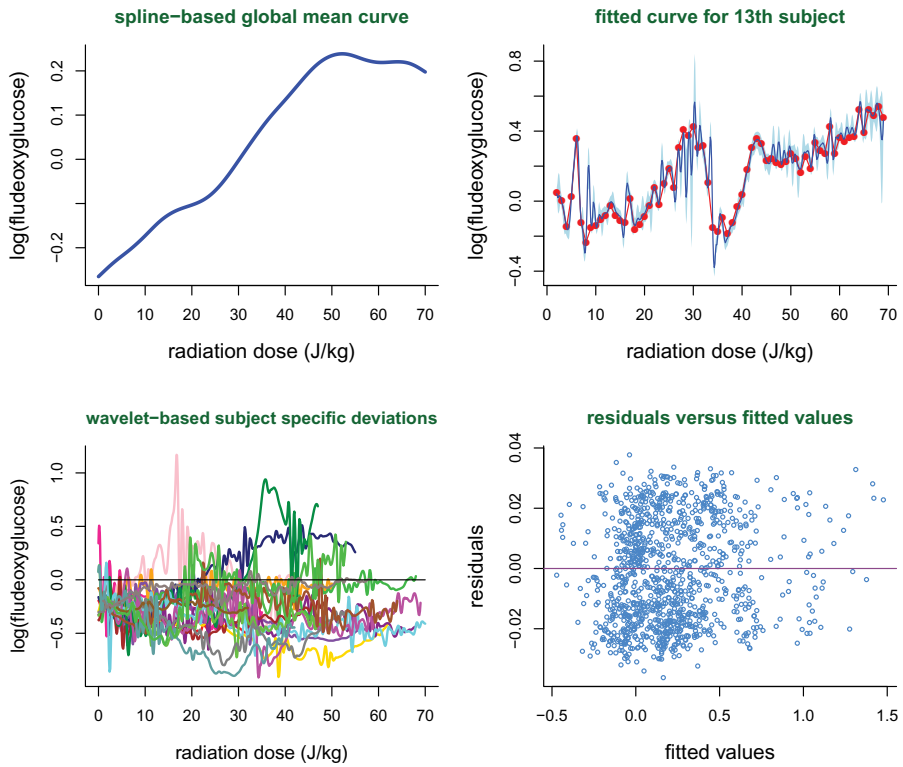

FIG 22. *Additional plots corresponding to the model fit shown in Figure 21. Top left: fitted penalized spline-based global mean curve. Bottom left: fitted penalized wavelet-based subject-specific deviation curves. Top right: Zoomed version of the for fit for the 13th subject. Bottom right: residuals versus fitted values.*

As of this writing our only implementation of model (42) and (43) is in BUGS, which has the disadvantage of taking 1–2 days to run on contemporary computing platforms. Ongoing work by Sarah E. Neville and the first author is aimed at developing faster MCMC and MFVB implementations for this and related models.

5.4. Non-standard semiparametric regression

As laid out in Section 2 penalized spline fitting and inference is now handled in a number of different ways. In particular, frequentist and Bayesian mixed model representations play an important role in accommodating various non-standard situations. Examples include measurement error (e.g. Berry, Carroll and Ruppert [7]), missing data (e.g. Faes, Ormerod and Wand [22]) and robustness (e.g. Staudenmayer, Lake and Wand [59]). In Marley and Wand [42]

we should how MCMC, with the help of BUGS, can handle a wide range of non-standard semiparametric regression problems.

As Section 3 shows, penalized wavelets can be handled using the same general approaches as penalized splines. It follows that modification to non-standard cases has similar parallels.

6. R software

Penalized wavelets benefit from particular software packages in the R language. We briefly describe some of them here.

The R package `wavethresh` (Nason [48]) plays a particularly important role in our proposed penalized wavelet paradigm since it supports efficient computation of the $\boldsymbol{Z}$ and $\boldsymbol{Z_g}$ design matrices, containing wavelet basis functions evaluated at predictor values or plotting grids. The function `ZDaub()`, described in Appendix A, contains the relevant code.

For the penalized least squares approach with L_1 penalization the function `lars()` within the package `lars` (Hastie and Efron [35]) efficiently computes a suite over a fine grid of λ values. The function also returns values of $\widehat{\text{edf}}(\lambda)$ which assists penalty parameter selection via criteria such as $\widehat{\text{GCV}}(\lambda)$.

The R package `ncvreg` (Breheny [9]) is similar to `lars` in that it efficiently computes penalized least squares fits over penalty parameter grids. However, it offers penalization using either the SCAD or minimax concave penalties. It also supports logistic regression loss and has k-fold cross-validation functionality for choice of the penalty parameter. Similar functionality is provided by the R package `glmnet` (Friedman, Hastie and Tibshirani [27, 28]), but with the elastic net family of penalties. This family includes the L_1 penalty as a special case.

A shortcoming of `lars`, `ncvreg` and `glmnet` in the context of the current article is that they support models only with a single penalty parameter. Hence, the multiple penalty parameter models described in Sections 5.2 and 5.3 require alternative routes for R implementation. As mentioned in Section 5.3, the `BRugs` package was used for the semiparametric longitudinal analysis done there and, of course, it can be used to handle the simpler Bayesian penalized models discussed earlier.

Finally, we mention that the matrix algebra features of the R language allow efficient implementation of the MFVB algorithms given in Sections 3 and 5.

7. Concluding remarks

The overarching theme of this article, that wavelets can be embedded in semiparametric regression in a way that is analogous to splines, is apparent from details provided in Sections 2 to 5. Two areas which have seen a great deal of recent activity in Statistics, wide data regression and mean field variational Bayes, are particularly relevant to penalized wavelets and can aid more widespread adoption. R packages for MCMC-based analyses, such as `BRugs`, also have an important role to play as demonstrated by the example in Section 5.3.

This new paradigm promises to be important for future analyses and developments in semiparametric regression since the benefits offered by wavelets can be enjoyed with relatively straightforward adaptation of existing penalized spline methodology.

Appendix A: R code for default basis computation

Algorithms 1 and 2, together with details given in Sections 2.1 and 3.1, describe construction of good default $\boldsymbol{Z}$ matrices for penalized splines and penalized wavelets, respectively. A web-supplement to this article is a ZIP archive titled `ZOSullandZDaub.zip` that includes two files `ZOSull.r` and `ZDaub.r`. These, respectively, contain the R functions, `ZOSull()` and `ZDaub()`, for computing these $\boldsymbol{Z}$ matrices. The first function uses O'Sullivan splines, or O-splines for short. The second uses Daubechies wavelets with the smoothness number an input parameter but defaulted to 5. Note that `ZDaub()` avoids computation and storage of large matrices, despite the description given in Section 3.1. Also included in `ZOSullandZDaub.zip` is an R script named

`ZOSullandZDaubDemo.Rs`

which demonstrates how `ZOSull()` and `ZDaub()` can be used for design matrix construction, prediction and plotting. The `README` file in the ZIP archive provides full details.

Appendix B: Details on frequentist mixed model-based penalized wavelet regression with Laplacian random effects

Consider the wavelet nonparametric regression model with frequentist mixed model representation:

$$\boldsymbol{y}|\boldsymbol{u} \sim N(\boldsymbol{X\beta} + \boldsymbol{Zu}, \sigma_\varepsilon^2 \boldsymbol{I})$$

where the u_k are independent with density function

$$p(u_k; \sigma_u) = (2\sigma_u)^{-1} \exp(-|u_k|/\sigma_u).$$

Theorem. *Suppose that $\boldsymbol{Z}^T\boldsymbol{Z} = \alpha^2 \boldsymbol{I}$. Then the log-likelihood of $(\boldsymbol{\beta}, \sigma_u^2, \sigma_\varepsilon^2)$ admits the explicit expression*

$$\begin{aligned}
&\ell(\boldsymbol{\beta}, \sigma_u^2, \sigma_\varepsilon^2) \\
&\quad = \tfrac{1}{2}(K-n)\log(2\pi\sigma_\varepsilon^2) - K\log(2\sigma_u) - \tfrac{1}{2\sigma_\varepsilon^2}\|\boldsymbol{y} - \boldsymbol{X\beta}\|^2 \\
&\qquad + \frac{1}{2\alpha^2\sigma_\varepsilon^2}\left\|\boldsymbol{Z}^T(\boldsymbol{y} - \boldsymbol{X\beta}) - \frac{\sigma_\varepsilon^2}{\sigma_u}\mathbf{1}\right\|^2 + \mathbf{1}^T \log\Phi\left(\frac{\boldsymbol{Z}^T(\boldsymbol{y} - \boldsymbol{X\beta}) - \frac{\sigma_\varepsilon^2}{\sigma_u}}{\alpha\sigma_\varepsilon}\right) \\
&\qquad + \mathbf{1}^T H\left(\frac{2\boldsymbol{Z}^T(\boldsymbol{y} - \boldsymbol{X\beta})}{\alpha^2\sigma_u} + \log\Phi\left(\frac{-\boldsymbol{Z}^T(\boldsymbol{y} - \boldsymbol{X\beta}) - \frac{\sigma_\varepsilon^2}{\sigma_u}}{\alpha\sigma_\varepsilon}\right)\right. \\
&\qquad\qquad \left. - \log\Phi\left(\frac{\boldsymbol{Z}^T(\boldsymbol{y} - \boldsymbol{X\beta}) - \frac{\sigma_\varepsilon^2}{\sigma_u}}{\alpha\sigma_\varepsilon}\right)\right)
\end{aligned}$$

where $H(x) \equiv \log(e^x+1)$. *In addition, the best predictor of* $\boldsymbol{u}$ *admits the explicit expression*

$$\begin{aligned} E(\boldsymbol{u}|\boldsymbol{y}) &= \alpha^{-2}\Big[w(\boldsymbol{y},\boldsymbol{\beta},\sigma_\varepsilon^2,\sigma_u^2)\{\boldsymbol{Z}^T(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})+\tfrac{\sigma_\varepsilon^2}{\sigma_u}\mathbf{1}\} \\ &\quad + \{1-w(\boldsymbol{y},\boldsymbol{\beta},\sigma_\varepsilon^2,\sigma_u^2)\}\{\boldsymbol{Z}^T(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})-\tfrac{\sigma_\varepsilon^2}{\sigma_u}\mathbf{1}\}\Big] \end{aligned}$$

where

$$\begin{aligned} w(\boldsymbol{y},\boldsymbol{\beta},\sigma_\varepsilon^2,\sigma_u^2) &\equiv \exp\Big(\frac{\boldsymbol{Z}^T(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})}{\alpha^2\sigma_u}\Big)\Phi\Big(\frac{-\boldsymbol{Z}^T(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})-\frac{\sigma_\varepsilon^2}{\sigma_u}}{\alpha\sigma_\varepsilon}\Big) \\ &\quad \times\Bigg\{\exp\Big(\frac{\boldsymbol{Z}^T(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})}{\alpha^2\sigma_u}\Big)\Phi\Big(\frac{-\boldsymbol{Z}^T(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})-\frac{\sigma_\varepsilon^2}{\sigma_u}}{\alpha\sigma_\varepsilon}\Big) \\ &\quad + \exp\Big(\frac{-\boldsymbol{Z}^T(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})}{\alpha^2\sigma_u}\Big)\Phi\Big(\frac{\boldsymbol{Z}^T(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})-\frac{\sigma_\varepsilon^2}{\sigma_u}}{\alpha\sigma_\varepsilon}\Big)\Bigg\}^{-1}. \end{aligned}$$

Remark 1. The expression for $\ell(\boldsymbol{\beta},\sigma_u^2,\sigma_\varepsilon^2)$ is given in terms of $H(x)=\log(e^x+1)$ for reasons of numerical stability. Note that $H(x)\approx x$, with this approximation being very accurate for $x \geq 20$. This approximation of $H(x)$ for large positive x should be used to avoid overflow in computation of $\ell(\boldsymbol{\beta},\sigma_u^2,\sigma_\varepsilon^2)$.

Remark 2. The expression for $E(\boldsymbol{u}|\boldsymbol{y})$ is similar to (6) of Pericchi and Smith [53] for the Bayes estimator of a normal location parameter with Laplacian prior.

Proof of Theorem. Define

$$\mathcal{C}(k,s_1,s_2,s_3) \equiv s_2^{-k-1}\int_{-\infty}^{\infty} x^k \exp\{-(x^2-2s_1x)/(2s_2^2)-|x|/s_3\}\,dx.$$

The proof uses the following two lemmas, each of which can be derived via elementary calculations:

Lemma 1. *For general* $s_1 \in \mathbb{R}$ *and* $s_2, s_3 > 0$

$$\mathcal{C}(0,s_1,s_2,s_3) = (\Phi/\phi)\left(-\frac{s_1}{s_2}-\frac{s_2}{s_3}\right)+(\Phi/\phi)\left(\frac{s_1}{s_2}-\frac{s_2}{s_3}\right)$$

and

$$\begin{aligned} \mathcal{C}(1,s_1,s_2,s_3) &= \left(\frac{s_1}{s_2}-\frac{s_2}{s_3}\right)(\Phi/\phi)\left(\frac{s_1}{s_2}-\frac{s_2}{s_3}\right) \\ &\quad -\left(-\frac{s_1}{s_2}-\frac{s_2}{s_3}\right)(\Phi/\phi)\left(-\frac{s_1}{s_2}-\frac{s_2}{s_3}\right) \end{aligned}$$

where $(\Phi/\phi)(x) \equiv \Phi(x)/\phi(x)$ *is the ratio of the standard normal cumulative distribution and density functions.*

Lemma 2. *For any* $a, b \in \mathbb{R}$

$$\log\{(\Phi/\phi)(-a-b) + (\Phi/\phi)(a-b)\} = \tfrac{1}{2}\log(2\pi) + \tfrac{1}{2}(a-b)^2 + \log\Phi(a-b) + H(2ab + \log\Phi(-a-b) - \log\Phi(a-b))$$

where $H(x) \equiv \log(e^x + 1)$.

The log-likelihood is

$$\begin{aligned}
\ell(\boldsymbol{\beta}, \sigma_u^2, \sigma_\varepsilon^2) &= \log p(\boldsymbol{y}; \boldsymbol{\beta}, \sigma_u^2, \sigma_\varepsilon^2) \\
&= \log \int_{\mathbb{R}^K} p(\boldsymbol{y}|\boldsymbol{u}; \boldsymbol{\beta}, \sigma_\varepsilon^2) p(\boldsymbol{u}|\sigma_u^2)\, d\boldsymbol{u} \\
&= \log \int_{\mathbb{R}^K} (2\pi\sigma_\varepsilon^2)^{-n/2} \exp\left\{-\frac{1}{2\sigma_\varepsilon^2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{Z}\boldsymbol{u}\|^2\right\} \\
&\qquad \times (2\sigma_u)^{-K} \exp\left\{-\sum_{k=1}^K \frac{|u_k|}{\sigma_u}\right\} d\boldsymbol{u}.
\end{aligned}$$

The assumption that $\boldsymbol{Z}^T\boldsymbol{Z} = \alpha^2\boldsymbol{I}$ leads to separation of the multivariate integral into K univariate integrals, resulting in

$$\begin{aligned}
\ell(\boldsymbol{\beta}, \sigma_u^2, \sigma_\varepsilon^2) &= -\tfrac{1}{2}\, n \log(2\pi\sigma_\varepsilon^2) + K\, \log\left(\frac{\sigma_\varepsilon}{2\sigma_u\alpha}\right) - \tfrac{1}{2\sigma_\varepsilon^2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|^2 \\
&\qquad + \sum_{k=1}^K \log \mathcal{C}(0, \{\boldsymbol{Z}^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})\}_k/\alpha, \sigma_\varepsilon, \alpha\,\sigma_u).
\end{aligned}$$

The stated result for $\ell(\boldsymbol{\beta}, \sigma_u^2, \sigma_\varepsilon^2)$ follows from the first part of Lemma 1 and Lemma 2.

Next note that

$$E(\boldsymbol{u}|\boldsymbol{y}) = \frac{\int_{\mathbb{R}^K} \boldsymbol{u}\, p(\boldsymbol{y}|\boldsymbol{u}; \boldsymbol{\beta}, \sigma_\varepsilon) p(\boldsymbol{u}; \sigma_u)\, d\boldsymbol{u}}{\int_{\mathbb{R}^K} p(\boldsymbol{y}|\boldsymbol{u}; \boldsymbol{\beta}, \sigma_\varepsilon) p(\boldsymbol{u}; \sigma_u)\, d\boldsymbol{u}}.$$

The denominator is the likelihood $\exp\{\ell(\boldsymbol{\beta}, \sigma_u^2, \sigma_\varepsilon^2)\}$ whilst the numerator is

$$\begin{aligned}
&\int_{\mathbb{R}^K} \boldsymbol{u}(2\pi\sigma_\varepsilon^2)^{-n/2} \exp\left\{-\tfrac{1}{2\sigma_\varepsilon^2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{Z}\boldsymbol{u}\|^2\right\} \\
&\qquad\qquad \times (2\sigma_u)^{-K} \exp\left\{-\textstyle\sum_{k=1}^K \frac{|u_k|}{\sigma_u}\right\} d\boldsymbol{u}.
\end{aligned} \tag{44}$$

As with the log-likelihood derivation, the assumption $\boldsymbol{Z}^T\boldsymbol{Z} = \alpha^2\boldsymbol{I}$ leads to separation of the multivariate integral into the following univariate integral expression for (44):

$$\frac{\mathcal{C}(1, \alpha^{-1}\boldsymbol{Z}^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}), \sigma_\varepsilon, \alpha\,\sigma_u)}{\mathcal{C}(0, \alpha^{-1}\boldsymbol{Z}^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}), \sigma_\varepsilon, \alpha\,\sigma_u)} \exp\{\ell(\boldsymbol{\beta}, \sigma_u^2, \sigma_\varepsilon^2)\}.$$

Application of Lemma 1 then leads to the explicit result for $E(\boldsymbol{u}|\boldsymbol{y})$. □

Appendix C: Derivation of (17) and Algorithm 3

We now derive result (17) and Algorithm 3 concerning MFVB fitting of the Bayesian penalized spline model (14). Throughout this appendix and the next two, additive constants with respect to the function argument are denoted by 'const'. The MFVB calculations heavily rely on the following results for the full conditional density functions:

$$\begin{aligned}
\log\, p(\boldsymbol{\beta},\boldsymbol{u}|\text{rest}) &= -\tfrac{1}{2}\left[\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix}^T\left(\sigma_\varepsilon^{-2}\boldsymbol{C}^T\boldsymbol{C}+\begin{bmatrix}\sigma_\beta^{-2}\boldsymbol{I}_2 & \boldsymbol{0}\\ \boldsymbol{0} & \sigma_u^{-2}\boldsymbol{I}_K\end{bmatrix}\right)\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix}\right.\\
&\qquad\left.-2\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix}^T\boldsymbol{C}^T\boldsymbol{y}\right]+\text{const},\\
\log\, p(\sigma_u^2|\text{rest}) &= \{-\tfrac{1}{2}(K+1)-1\}\log(\sigma_u^2)-\left(\tfrac{1}{2}\|\boldsymbol{u}\|^2+a_u^{-1}\right)/\sigma_u^2+\text{const},\\
\log\, p(\sigma_\varepsilon^2|\text{rest}) &= \{-\tfrac{1}{2}(n+1)-1\}\log(\sigma_\varepsilon^2)\\
&\qquad-\left(\tfrac{1}{2}\|\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}-\boldsymbol{Z}\boldsymbol{u}\|^2+a_\varepsilon^{-1}\right)/\sigma_\varepsilon^2+\text{const},\\
\log\, p(a_u|\text{rest}) &= -2\log(a_u)-(\sigma_u^{-2}+A_u^{-2})/a_u+\text{const}\\
\text{and}\ \log\, p(a_\varepsilon|\text{rest}) &= -2\log(a_\varepsilon)-(\sigma_\varepsilon^{-2}+a_\varepsilon^{-2})/a_\varepsilon+\text{const}.
\end{aligned}$$

Expressions for $q^*(\boldsymbol{\beta},\boldsymbol{u})$, $\boldsymbol{\mu}_{q(\boldsymbol{\beta},\boldsymbol{u})}$ *and* $\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{u})}$

$$q^*(\boldsymbol{\beta},\boldsymbol{u})\sim N(\boldsymbol{\mu}_{q(\boldsymbol{\beta},\boldsymbol{u})},\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{u})})$$

where

$$\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{u})}=\left(\mu_{q(1/\sigma_\varepsilon^2)}\boldsymbol{C}^T\boldsymbol{C}+\begin{bmatrix}\sigma_\beta^{-2}\boldsymbol{I}_2 & \boldsymbol{0}\\ \boldsymbol{0} & \mu_{q(1/\sigma_u^2)}\boldsymbol{I}_K\end{bmatrix}\right)^{-1}$$

and

$$\boldsymbol{\mu}_{q(\boldsymbol{\beta},\boldsymbol{v})}=\mu_{q(1/\sigma_\varepsilon^2)}\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{u})}\boldsymbol{C}^T\boldsymbol{y}.$$

Derivation:

$$\begin{aligned}
\log\, q^*(\boldsymbol{\beta},\boldsymbol{u}) &= E_q\{\log\, p(\boldsymbol{\beta},\boldsymbol{u}|\text{rest})\}+\text{const}\\
&= -\tfrac{1}{2}\left\{\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix}^T\left(\mu_{q(1/\sigma_\varepsilon^2)}\boldsymbol{C}^T\boldsymbol{C}+\begin{bmatrix}\sigma_\beta^{-2}\boldsymbol{I} & \boldsymbol{0}\\ \boldsymbol{0} & \mu_{q(1/\sigma_u^2)}\boldsymbol{I}_K\end{bmatrix}\right)\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix}\right.\\
&\qquad\left.-2\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix}^T\boldsymbol{C}^T\boldsymbol{y}\right\}+\text{const}.
\end{aligned}$$

The stated result then follows from standard 'completion of the square' manipulations.

Expressions for $q^*(\sigma_u^2)$, $B_{q(\sigma_u^2)}$ *and* $\mu_{q(1/\sigma_u^2)}$

$$q^*(\sigma_u^2) \sim \text{Inverse-Gamma}(\tfrac{1}{2}(K+1), B_{q(\sigma_u^2)})$$

where

$$B_q(\sigma_u^2) = \tfrac{1}{2}\{\|\boldsymbol{\mu}_{q(\boldsymbol{u})}\|^2 + \text{tr}(\boldsymbol{\Sigma}_{q(\boldsymbol{u})})\} + \mu_{q(1/a_u)}.$$

In addition,

$$\mu_{q(1/\sigma_u^2)} = \tfrac{1}{2}(K+1)/B_{q(\sigma_u^2)}.$$

Derivation:

$$\begin{aligned}\log q^*(\sigma_u^2) &= E_q\{\log p(\sigma_u^2|\text{rest})\} + \text{const}\\ &= \{-\tfrac{1}{2}(K+1)-1\}\log(\sigma_u^2) - \left(\tfrac{1}{2}\,E_q\|\boldsymbol{u}\|^2 + \mu_{q(1/a_u)}\right)/\sigma_u^2 + \text{const}.\end{aligned}$$

The form of $q^*(\sigma_u^2)$ and $B_{q(\sigma_u^2)}$ follows from this and the fact that

$$E_q\|\boldsymbol{u}\|^2 = \|E_q(\boldsymbol{u})\|^2 + \text{tr}\{\text{Cov}_q(\boldsymbol{u})\}.$$

The expression for $\mu_{q(1/\sigma_u^2)}$ follows from elementary manipulations involving Inverse Gamma density functions.

Expressions for $q^*(\sigma_\varepsilon^2)$, $B_{q(\sigma_\varepsilon^2)}$ *and* $\mu_{q(1/\sigma_\varepsilon^2)}$

$$q^*(\sigma_\varepsilon^2) \sim \text{Inverse-Gamma}(\tfrac{1}{2}(n+1), B_{q(\sigma_\varepsilon^2)})$$

where

$$B_q(\sigma_\varepsilon^2) = \tfrac{1}{2}\{\|\boldsymbol{y} - \boldsymbol{C}\boldsymbol{\mu}_{q(\boldsymbol{\beta},\boldsymbol{u})}\|^2 + \text{tr}(\boldsymbol{C}^T\boldsymbol{C}\,\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{u})})\} + \mu_{q(1/a_\varepsilon)}.$$

In addition,

$$\mu_{q(1/\sigma_\varepsilon^2)} = \tfrac{1}{2}(n+1)/B_{q(\sigma_\varepsilon^2)}.$$

Derivation:

This derivation is similar to that for $q^*(\sigma_u^2)$.

Expressions for $q^*(a_\varepsilon)$, $B_{q(a_\varepsilon)}$ *and* $\mu_{q(1/a_\varepsilon)}$

$$q^*(a_\varepsilon) \sim \text{Inverse-Gamma}(1, B_{q(a_\varepsilon)})$$

where

$$B_{q(a_\varepsilon)} = \mu_{q(1/\sigma_\varepsilon^2)} + A_\varepsilon^{-2} \quad \text{and} \quad \mu_{q(1/a_\varepsilon)} = 1/\{\mu_{q(1/\sigma_\varepsilon^2)} + A_\varepsilon^{-2}\}.$$

Derivation:

$$\begin{aligned}\log q^*(a_\varepsilon) &= -2\log(a_\varepsilon) - E_q(\sigma_\varepsilon^{-2} + A_\varepsilon^{-2})/a_\varepsilon + \text{const}\\ &= (-1-1)\log(a_\varepsilon) - (\mu_{q(1/\sigma_\varepsilon^2)} + A_\varepsilon^{-2})/a_\varepsilon + \text{const}.\end{aligned}$$

Therefore

$$q^*(a_\varepsilon) \sim \text{Inverse-Gamma}(1, \mu_{q(1/\sigma_\varepsilon^2)} + A_\varepsilon^{-2}).$$

The expressions for $B_{q(a_\varepsilon)}$ and $\mu_{q(1/a_\varepsilon)}$ follow immediately.

Expressions for $q^*(a_u)$, $B_{q(a_u)}$ *and* $\mu_{q(1/a_u)}$

$$q^*(a_u) \sim \text{Inverse-Gamma}(1, B_{q(a_u)})$$

where

$$B_{q(a_u)} = \mu_{q(1/\sigma_u^2)} + A_u^{-2} \quad \text{and} \quad \mu_{q(1/a_u)} = 1/\{\mu_{q(1/\sigma_u^2)} + A_u^{-2}\}.$$

Derivation:

The derivation is analogous to that for $q^*(a_\varepsilon)$ and related quantities.

Appendix D: Derivation of (32) and Algorithm 4

In this appendix we derive (32) and Algorithm 4 concerning MFVB fitting of the Bayesian penalized wavelet model (30).

The full conditionals satisfy

$$\begin{aligned}
\log p(\boldsymbol{\beta}, \boldsymbol{v}|\text{rest}) &= -\tfrac{1}{2}\left\{ \begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{v}\end{bmatrix}^T \left(\sigma_\varepsilon^{-2}\boldsymbol{C}_\gamma^T\boldsymbol{C}_\gamma + \begin{bmatrix}\sigma_\beta^{-2} & \boldsymbol{0}\\ \boldsymbol{0} & \sigma_u^{-2}\text{diag}(\boldsymbol{b})\end{bmatrix}\right)\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{v}\end{bmatrix} \right.\\
&\qquad \left. -2\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{v}\end{bmatrix}^T \boldsymbol{C}_\gamma^T\boldsymbol{y}\right\} + \text{const}\\
\log p(\sigma_u^2|\text{rest}) &= \{-\tfrac{1}{2}(K+1)-1\}\log(\sigma_u^2) - \left\{\tfrac{1}{2}\boldsymbol{v}^T\text{diag}(\boldsymbol{b})\boldsymbol{v} + a_u^{-1}\right\}/\sigma_u^2\\
&\qquad +\text{const},\\
\log p(\sigma_\varepsilon^2|\text{rest}) &= \{-\tfrac{1}{2}(n+1)-1\}\log(\sigma_\varepsilon^2)\\
&\qquad -\left(\tfrac{1}{2}\|\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}-\boldsymbol{Z}_\gamma\boldsymbol{u}\|^2 + a_\varepsilon^{-1}\right)/\sigma_\varepsilon^2 + \text{const},\\
\log p(a_u|\text{rest}) &= -2\log(a_u) - (\sigma_u^{-2} + A_u^{-2})/a_u + \text{const},\\
\log p(a_\varepsilon|\text{rest}) &= -2\log(a_\varepsilon) - (\sigma_\varepsilon^{-2} + a_\varepsilon^{-2})/a_\varepsilon + \text{const},\\
\log p(\boldsymbol{b}|\text{rest}) &= \sum_{k=1}^K \{-\tfrac{3}{2}\log(b_k) - (b_k - \sigma_u/|v_k|)^2/(2\,b_k\,\sigma_u^2/v_k^2)\} + \text{const},\\
\log p(\boldsymbol{\gamma}|\text{rest}) &= \sum_{k=1}^K \left\{ -\tfrac{1}{2\sigma_\varepsilon^2}\|\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}-\boldsymbol{Z}_{-k}(\boldsymbol{\gamma}_{-k}\odot\boldsymbol{v}_{-k}) - \boldsymbol{Z}_k\gamma_k\,v_k\|^2 \right.\\
&\qquad \left. +\gamma_k\,\text{logit}(\rho)\right\} + \text{const}\\
\text{and}\ \log p(\rho|\text{rest}) &= (A_\rho + \gamma_\bullet - 1)\,\log(\rho) + (B_\rho + K - \gamma_\bullet - 1)\,\log(1-\rho)\\
&\qquad +\text{const}.
\end{aligned}$$

where $\gamma_\bullet \equiv \sum_{k=1}^K \gamma_k$.

Expressions for $q^*(\boldsymbol{\beta}, \boldsymbol{v})$, $\boldsymbol{\mu}_{q(\boldsymbol{\beta},\boldsymbol{v})}$ *and* $\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{v})}$

$$q^*(\boldsymbol{\beta}, \boldsymbol{v}) \sim N(\boldsymbol{\mu}_{q(\boldsymbol{\beta},\boldsymbol{v})}, \boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{v})})$$

where

$$\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{v})} = \left(\mu_{q(1/\sigma_\varepsilon^2)}(\boldsymbol{C}^T\boldsymbol{C}) \odot \boldsymbol{\Omega}_{q(\boldsymbol{w}_\boldsymbol{\gamma})} + \begin{bmatrix} \sigma_\beta^{-2} & \boldsymbol{0} \\ \boldsymbol{0} & \mu_{q(1/\sigma_u^2)}\text{diag}(\mu_{q(\boldsymbol{b})}) \end{bmatrix} \right)^{-1},$$

$$\boldsymbol{\mu}_{q(\boldsymbol{\beta},\boldsymbol{v})} = \mu_{q(1/\sigma_\varepsilon^2)}\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{v})}\text{diag}\{\boldsymbol{\mu}_{q(\boldsymbol{w}_\boldsymbol{\gamma})}\}\boldsymbol{C}^T\boldsymbol{y}$$

and

$$\boldsymbol{\Omega}_{q(\boldsymbol{w}_\boldsymbol{\gamma})} \equiv \text{diag}\{\boldsymbol{\mu}_{q(\boldsymbol{w}_\boldsymbol{\gamma})} \odot (\boldsymbol{1} - \boldsymbol{\mu}_{q(\boldsymbol{w}_\boldsymbol{\gamma})})\} + \boldsymbol{\mu}_{q(\boldsymbol{w}_\boldsymbol{\gamma})}\,\boldsymbol{\mu}_{q(\boldsymbol{w}_\boldsymbol{\gamma})}^T. \tag{45}$$

Derivation:

$$\begin{aligned} \log\, q^*(\boldsymbol{\beta},\boldsymbol{v}) \quad = \quad & -\tfrac{1}{2}\Bigg\{ \begin{bmatrix} \boldsymbol{\beta} \\ \boldsymbol{v} \end{bmatrix}^T \Bigg(\mu_{q(1/\sigma_\varepsilon^2)} E_q(\boldsymbol{C}_\boldsymbol{\gamma}^T\boldsymbol{C}_\boldsymbol{\gamma}) \\ & + \begin{bmatrix} \sigma_\beta^{-2} & \boldsymbol{0} \\ \boldsymbol{0} & \mu_{q(1/\sigma_u^2)}\text{diag}(\mu_{q(\boldsymbol{b})}) \end{bmatrix} \Bigg) \begin{bmatrix} \boldsymbol{\beta} \\ \boldsymbol{v} \end{bmatrix} \\ & -2 \begin{bmatrix} \boldsymbol{\beta} \\ \boldsymbol{v} \end{bmatrix}^T E_q(\boldsymbol{C}_\boldsymbol{\gamma})^T\boldsymbol{y} \Bigg\} + \text{const.} \end{aligned}$$

The stated result then follows from standard 'completion of the square' manipulations and explicit expressions for $E_q(\boldsymbol{C}_\boldsymbol{\gamma})$ and $E_q(\boldsymbol{C}_\boldsymbol{\gamma}^T\boldsymbol{C}_\boldsymbol{\gamma})$ which we derive next.

Firstly,

$$E_q(\boldsymbol{C}_\boldsymbol{\gamma}) = \boldsymbol{C}\, E_q\{\text{diag}(w_\boldsymbol{\gamma})\} = \boldsymbol{C}\,\text{diag}\{\boldsymbol{\mu}_{q(\boldsymbol{w}_\boldsymbol{\gamma})}\}.$$

Secondly,

$$\boldsymbol{C}_\boldsymbol{\gamma}^T\boldsymbol{C}_\boldsymbol{\gamma} = \text{diag}(\boldsymbol{w}_\boldsymbol{\gamma})\,\boldsymbol{C}^T\boldsymbol{C}\text{diag}(\boldsymbol{w}_\boldsymbol{\gamma}) = (\boldsymbol{C}^T\boldsymbol{C}) \odot (\boldsymbol{w}_\boldsymbol{\gamma}\boldsymbol{w}_\boldsymbol{\gamma}^T).$$

Hence,

$$E_q(\boldsymbol{C}_\boldsymbol{\gamma}^T\boldsymbol{C}_\boldsymbol{\gamma}) = (\boldsymbol{C}^T\boldsymbol{C}) \odot \{\text{Cov}_q(\boldsymbol{w}_\boldsymbol{\gamma}) + \boldsymbol{\mu}_{q(\boldsymbol{w}_\boldsymbol{\gamma})}\,\boldsymbol{\mu}_{q(\boldsymbol{w}_\boldsymbol{\gamma})}^T\}.$$

Since the entries of $\boldsymbol{w}_\boldsymbol{\gamma}$ are binary and independent with respect to $q(\boldsymbol{\gamma})$ we have

$$\text{Cov}_q(\boldsymbol{w}_\boldsymbol{\gamma}) = \text{diag}\{\boldsymbol{\mu}_{q(\boldsymbol{w}_\boldsymbol{\gamma})} \odot (\boldsymbol{1} - \boldsymbol{\mu}_{q(\boldsymbol{w}_\boldsymbol{\gamma})})\}.$$

The stated results from these results via standard arguments.

Expressions for $q^*(\sigma_u^2)$, $B_{q(\sigma_u^2)}$ and $\mu_{q(1/\sigma_u^2)}$

$$q^*(\sigma_u^2) \sim \text{Inverse-Gamma}(\tfrac{1}{2}(K+1), B_{q(\sigma_u^2)})$$

where

$$B_{q(\sigma_u^2)} = \mu_{q(1/a_u)} + \tfrac{1}{2}\sum_{k=1}^{K} \mu_{q(b_k)}\{\sigma_{q(v_k)}^2 + \mu_{q(v_k)}^2\}.$$

In addition,

$$\mu_{q(1/\sigma_u^2)} = \tfrac{1}{2}(K+1)/B_{q(\sigma_u^2)}.$$

Derivation:

$$\begin{aligned}\log q^*(\sigma_u^2) &= \{-\tfrac{1}{2}(K+1)-1\}\log(\sigma_u^2) - \left[\tfrac{1}{2}\,E_q\{\boldsymbol{v}^T\mathrm{diag}(\boldsymbol{b})\boldsymbol{v}\} + \mu_{q(1/a_u)}\right]\Big/\sigma_u^2 \\ &\quad +\text{const.}\end{aligned}$$

It is apparent from this that $q^*(\sigma_u^2)$ is an Inverse Gamma density function with shape parameter $\frac{1}{2}(K+1)$ and rate parameter, $B_{q(\sigma_u^2)}$, equal to the term inside the square brackets. The remaining non-explicit term is

$$E_q\{\boldsymbol{v}^T\mathrm{diag}(\boldsymbol{b})\boldsymbol{v}\} = \mathrm{tr}\{\mathrm{diag}(\mu_q(\boldsymbol{b}))\,E_q(\boldsymbol{v}\boldsymbol{v}^T)\} = \sum_{k=1}^{K}\mu_{q(b_k)}\{\sigma^2_{q(v_k)} + \mu^2_{q(v_k)}\}.$$

Expressions for $q^*(\sigma_\varepsilon^2)$, $B_{q(\sigma_\varepsilon^2)}$ and $\mu_{q(1/\sigma_\varepsilon^2)}$

$$q^*(\sigma_\varepsilon^2) \sim \text{Inverse-Gamma}(\tfrac{1}{2}(n+1), B_{q(\sigma_\varepsilon^2)})$$

where

$$\begin{aligned}B_{q(\sigma_\varepsilon^2)} &= \mu_{q(1/a_\varepsilon)} + \tfrac{1}{2}\,\|\boldsymbol{y}\|^2 - \boldsymbol{y}^T\boldsymbol{C}\left(\boldsymbol{\mu}_{q(\boldsymbol{w}_\gamma)}\odot\mu_{q(\boldsymbol{\beta},\boldsymbol{v})}\right) \\ &\quad +\tfrac{1}{2}\mathrm{tr}\left(\boldsymbol{C}^T\boldsymbol{C}\,\left[\boldsymbol{\Omega}_{q(\boldsymbol{w}_\gamma)}\odot\left\{\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{v})} + \boldsymbol{\mu}_{q(\boldsymbol{\beta},\boldsymbol{v})}\boldsymbol{\mu}^T_{q(\boldsymbol{\beta},\boldsymbol{v})}\right\}\right]\right)\end{aligned}$$

and $\boldsymbol{\Omega}_{q(\boldsymbol{w}_\gamma)}$ is as given by (45). In addition,

$$\mu_{q(1/\sigma_\varepsilon^2)} = \tfrac{1}{2}(n+1)/B_{q(\sigma_\varepsilon^2)}.$$

Derivation:

$$\begin{aligned}\log q^*(\sigma_\varepsilon^2) &= E_q\{\log p(\sigma_\varepsilon^2|\text{rest})\} + \text{const} \\ &= \{-\tfrac{1}{2}(n+1)-1\}\log(\sigma_\varepsilon^2) \\ &\quad -\left\{\tfrac{1}{2}\,E_q\left\|\boldsymbol{y} - \boldsymbol{C}\left(\boldsymbol{w}_\gamma\odot\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{v}\end{bmatrix}\right)\right\|^2 + \mu_{q(1/a_\varepsilon)}\right\}\Big/\sigma_\varepsilon^2 + \text{const.}\end{aligned}$$

It is apparent from this that $q^*(\sigma_\varepsilon^2)$ is an Inverse Gamma density function with shape parameter $\frac{1}{2}(n+1)$ and rate parameter, $B_{q(\sigma_\varepsilon^2)}$, equal to the term inside the curly brackets. The remaining non-explicit term is

$$\begin{aligned}E_q\left\|\boldsymbol{y} - \boldsymbol{C}\left(\boldsymbol{w}_\gamma\odot\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{v}\end{bmatrix}\right)\right\|^2 &= \left\|\boldsymbol{y} - \boldsymbol{C}\left(\boldsymbol{\mu}_{q(\boldsymbol{w}_\gamma)}\odot\mu_{q(\boldsymbol{\beta},\boldsymbol{v})}\right)\right\|^2 \\ &\quad +\mathrm{tr}\left\{\boldsymbol{C}^T\boldsymbol{C}\;\mathrm{Cov}_q\left(\boldsymbol{w}_\gamma\odot\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{v}\end{bmatrix}\right)\right\}.\end{aligned}$$

Lemma 3 below implies that

$$\begin{aligned}\text{Cov}_q\left(\boldsymbol{w}_{\boldsymbol{\gamma}} \odot \left[\begin{array}{c}\boldsymbol{\beta}\\ \boldsymbol{v}\end{array}\right]\right) &= \text{Cov}_q(\boldsymbol{w}_{\boldsymbol{\gamma}}) \odot \{\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{v})} + \boldsymbol{\mu}_{q(\boldsymbol{\beta},\boldsymbol{v})}\boldsymbol{\mu}_{q(\boldsymbol{\beta},\boldsymbol{v})}^T\}\\ &\quad +(\boldsymbol{\mu}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})}\boldsymbol{\mu}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})}^T) \odot \boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{v})}.\end{aligned}$$

The stated result for $B_{q(\sigma_\varepsilon^2)}$ then follows quickly from this expression and the fact that

$$\text{Cov}_q(\boldsymbol{w}_{\boldsymbol{\gamma}}) = \text{diag}[\boldsymbol{\mu}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})} \odot \{\mathbf{1} - \boldsymbol{\mu}_{q(\boldsymbol{w}_{\boldsymbol{\gamma}})}\}].$$

Lemma 3. *If $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are independent random vectors of the same length then*

$$\begin{aligned}Cov(\boldsymbol{x}_1 \odot \boldsymbol{x}_2) &= Cov(\boldsymbol{x}_1) \odot Cov(\boldsymbol{x}_2) + \{E(\boldsymbol{x}_1)E(\boldsymbol{x}_1)^T\} \odot Cov(\boldsymbol{x}_2)\\ &\quad +\{E(\boldsymbol{x}_2)E(\boldsymbol{x}_2)^T\} \odot Cov(\boldsymbol{x}_1)\end{aligned}$$

Proof. First note that for any constant vector $\boldsymbol{a}$ having the same length as $\boldsymbol{x}$ we have $\text{Cov}(\boldsymbol{a} \odot \boldsymbol{x}) = (\boldsymbol{a}\boldsymbol{a}^T) \odot \text{Cov}(\boldsymbol{x})$. Then

$$\begin{aligned}\text{Cov}(\boldsymbol{x}_1 \odot \boldsymbol{x}_2) &= E\{\text{Cov}(\boldsymbol{x}_1 \odot \boldsymbol{x}_2|\, \boldsymbol{x}_1)\} + \text{Cov}\{E(\boldsymbol{x}_1 \odot \boldsymbol{x}_2|\, \boldsymbol{x}_1)\}\\ &= \{E(\boldsymbol{x}_1\boldsymbol{x}_1^T)\} \odot \text{Cov}(\boldsymbol{x}_2) + \text{Cov}\{E(\boldsymbol{x}_2) \odot \boldsymbol{x}_1\}\\ &= \{\text{Cov}(\boldsymbol{x}_1) + E(\boldsymbol{x}_1)E(\boldsymbol{x}_1)^T\} \odot \text{Cov}(\boldsymbol{x}_2)\\ &\quad +\{E(\boldsymbol{x}_2)E(\boldsymbol{x}_2)^T\} \odot \text{Cov}(\boldsymbol{x}_1).\end{aligned}$$

The lemma follows immediately. □

Expressions for $q^*(b_k)$ and $\mu_{q(b_k)}$

$$q^*(\boldsymbol{b}) = \prod_{k=1}^K q^*(b_k)$$

where

$$q^*(b_k) \sim \text{Inverse-Gaussian}(\{\mu_{q(1/\sigma_u^2)}(\sigma_{q(v_k)}^2 + \mu_{q(v_k)}^2)\}^{-1/2}, 1).$$

In addition,

$$\mu_{q(b_k)} = \{\mu_{q(1/\sigma_u^2)}(\sigma_{q(v_k)}^2 + \mu_{q(v_k)}^2)\}^{-1/2}.$$

Derivation:

We have

$$\log p(\boldsymbol{b}|\text{rest}) = \sum_{k=1}^K \{-\tfrac{3}{2}\log(b_k) - (b_k - \sigma_u/|v_k|)^2/(2\, b_k\, \sigma_u^2/v_k^2)\} + \text{const}$$

from which it follows that

$$\begin{aligned}\log q^*(\boldsymbol{b}) &= \sum_{k=1}^K [-\tfrac{3}{2}\log(b_k) - E_q\{(b_k - \sigma_u/|v_k|)^2/(2\, b_k\, \sigma_u^2/v_k^2)\}] + \text{const}\\ &= \sum_{k=1}^K [-\tfrac{3}{2}\log(b_k) - \tfrac{1}{2}\mu_{q(1/\sigma_u^2)}E_q(v_k^2)\, b_k - \tfrac{1}{2}(1/b_k)] + \text{const}.\end{aligned}$$

Straightforward manipulations then lead to the stated result.

Expressions for $q^*(\rho)$

$$q^*(\rho) \sim \text{Beta}(A_\rho + \mu_{q(\gamma_\bullet)}, B_\rho + K - \mu_{q(\gamma_\bullet)}).$$

Derivation:
Since

$$\begin{aligned}\log p(\rho|\text{rest}) &= (A_\rho + \gamma_\bullet - 1)\log(\rho) + (B_\rho + K - \gamma_\bullet - 1)\log(1-\rho) \\ &\quad + \text{const},\end{aligned}$$

we have

$$\begin{aligned}\log q^*(\rho) &= (A_\rho + \mu_{q(\gamma_\bullet)} - 1)\log(\rho) + (B_\rho + K - \mu_{q(\gamma_\bullet)} - 1)\log(1-\rho) \\ &\quad + \text{const}.\end{aligned}$$

which quickly leads to the stated result for $q^*(\rho)$.

Expressions for $q^*(\gamma_k)$ and $\mu_{q(\gamma_k)}$

$$q^*(\gamma_k) \overset{\text{ind.}}{\sim} \text{Bernoulli}\left(\frac{\exp(\eta_{q(\gamma_k)})}{1+\exp(\eta_{q(\gamma_k)})}\right)$$

where

$$\begin{aligned}\eta_{q(\gamma_k)} &= -\tfrac{1}{2}\,\mu_{q(1/\sigma_\varepsilon^2)}\Big[\|\boldsymbol{Z}_k\|^2\{\sigma^2_{q(v_k)} + \mu^2_{q(v_k)}\} - 2\boldsymbol{Z}_k^T\boldsymbol{y}\,\mu_{q(v_k)} \\ &\quad + 2\boldsymbol{Z}_k^T\boldsymbol{X}\left\{(\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{v})})_{1,1+k} + \mu_{q(\boldsymbol{\beta})}\mu_{q(v_k)}\right\} \\ &\quad + 2\boldsymbol{Z}_k^T\boldsymbol{Z}_{-k}\left\{(\boldsymbol{\mu}_{q(\boldsymbol{\gamma})})_{-k} \odot \{(\boldsymbol{\Sigma}_{q(\boldsymbol{v})})_{-k,k} + \mu_{q(v_k)}(\boldsymbol{\mu}_{q(\boldsymbol{v})})_{-k}\}\right\}\Big] \\ &\quad + \psi(A_\rho + \mu_{q(\gamma_\bullet)}) - \psi(B_\rho + K - \mu_{q(\gamma_\bullet)}).\end{aligned}$$

Derivation:

The full conditional density function for γ_k satisfies

$$\begin{aligned}\log p(\gamma_k|\text{rest}) &= -\tfrac{1}{2\sigma_\varepsilon^2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{Z}_{-k}(\boldsymbol{\gamma}_{-k} \odot \boldsymbol{v}_{-k}) - \boldsymbol{Z}_k\gamma_k\,v_k\|^2 \\ &\quad + \gamma_k\,\text{logit}(\rho) + \text{const} \\ &= \gamma_k\Big(-\tfrac{1}{2\sigma_\varepsilon^2}\big[\|\boldsymbol{Z}_k\|^2 v_k^2 - 2\boldsymbol{y}^T\boldsymbol{Z}_k\,v_k + 2\boldsymbol{X}^T\boldsymbol{Z}_k\,\boldsymbol{\beta}\,v_k \\ &\quad + 2\,\boldsymbol{Z}_k^T\boldsymbol{Z}_{-k}\{\boldsymbol{\gamma}_{-k} \odot (v_k\boldsymbol{v}_{-k})\}\big] + \text{logit}(\rho)\Big) + \text{const}.\end{aligned}$$

Hence

$$\begin{aligned}\log q^*(\gamma_k) &= \gamma_k\Big(-\tfrac{1}{2}\,\mu_{q(1/\sigma_\varepsilon^2)}\,E_q\big[\|\boldsymbol{Z}_k\|^2 v_k^2 - 2\boldsymbol{y}^T\boldsymbol{Z}_k\,v_k + 2\boldsymbol{X}^T\boldsymbol{Z}_k\,\boldsymbol{\beta}\,v_k \\ &\quad + 2\,\boldsymbol{Z}_k^T\boldsymbol{Z}_{-k}\{\boldsymbol{\gamma}_{-k} \odot (v_k\boldsymbol{v}_{-k})\}\big] + E_q\{\text{logit}(\rho)\}\Big) + \text{const}.\end{aligned}$$

We thus require four expectations with respect to the q-functions corresponding to the square brackets in this last expression. The first is

$$E_q(v_k^2) = \sigma^2_{q(v_k)} + \mu^2_{q(v_k)} = (\,\boldsymbol{\Sigma}_{q(\boldsymbol{v})})_{kk} + (\boldsymbol{\mu}_{q(\boldsymbol{v})})^2_k$$

whilst the second is $E_q(v_k) = (\boldsymbol{\mu}_{q(\boldsymbol{v})})_k$. The third is

$$E_q(\beta\, v_k) = (\,\boldsymbol{\Sigma}_{q(\boldsymbol{\beta},\boldsymbol{v})})_{1,1+k} + \mu_{q(\boldsymbol{\beta})}(\boldsymbol{\mu}_{q(\boldsymbol{v})})_k.$$

Next note that

$$\begin{aligned} E_q\{\boldsymbol{\gamma}_{-k} \odot (v_k \boldsymbol{v}_{-k})\} &= (\boldsymbol{\mu}_{q(\boldsymbol{\gamma})})_{-k} \odot E_q(v_k \boldsymbol{v}_{-k}) \\ &= (\boldsymbol{\mu}_{q(\boldsymbol{\gamma})})_{-k} \odot \{(\boldsymbol{\Sigma}_{q(\boldsymbol{v})})_{-k,k} + \mu_{q(v_k)}(\boldsymbol{\mu}_{q(\boldsymbol{v})})_{-k}\} \end{aligned}$$

where $(\boldsymbol{\Sigma}_{q(\boldsymbol{v})})_{-k,k}$ is the kth column of $\boldsymbol{\Sigma}_{q(\boldsymbol{v})}$ with the kth row omitted.

The remaining expectation is

$$\begin{aligned} E_q\{\text{logit}(\rho)\} &= E_q\{\log(\rho)\} - E_q\{\log(1-\rho)\} \\ &= \int_0^1 \frac{\rho^{A_\rho + \mu_{q(\gamma_\bullet)} - 1}(1-\rho)^{B_\rho + K - \mu_{q(\gamma_\bullet)} - 1} \log(\rho)}{B(A_\rho + \mu_{q(\gamma_\bullet)}, B_\rho + K - \mu_{q(\gamma_\bullet)})}\, d\rho \\ &\quad - \int_0^1 \frac{\rho^{Ap + \mu_{q(\gamma_\bullet)} - 1}(1-\rho)^{B_\rho + K - \mu_{q(\gamma_k)} - 1} \log(1-\rho)}{B(A_\rho + \mu_{q(\gamma_\bullet)}, B_\rho + K - \mu_{q(\gamma_\bullet)})}\, d\rho \end{aligned}$$

where $B(\cdot,\cdot)$ is the Beta function. Using the integral result

$$\int_0^1 \frac{x^{a-1}(1-x)^{b-1}}{B(a,b)} \log(x)\, dx = \psi(a) - \psi(a+b)$$

(Result 4.253 1. of Gradshteyn and Ryzhik [30]), where $\psi(x) \equiv \frac{d}{dx}\log\{\Gamma(x)\}$ is the digamma function, we eventually get

$$E_q\{\text{logit}(\rho)\} = \psi(A_\rho + \mu_{q(\gamma_\bullet)}) - \psi(B_\rho + K - \mu_{q(\gamma_\bullet)}).$$

On combining we see that

$$\log q^*(\gamma_k) = \gamma_k\, \eta_{q(\gamma_k)} + \text{const}, \quad \gamma_k = 0, 1.$$

The stated result follows immediately.

Expressions for $q^*(a_\varepsilon)$, $B_{q(a_\varepsilon)}$, $\mu_{q(1/a_\varepsilon)}$, $q^*(a_u)$, $B_{q(a_u)}$ and $\mu_{q(1/a_u)}$

Each of these expressions, and their derivations, are identical to the penalized spline case.

Appendix E: Derivation of Algorithm 5

In Algorithm 5, the MFVB calculations for σ_u^2, a_u, $\boldsymbol{b}$ and ρ are unaffected by the change from Gaussian $\boldsymbol{y}$ to binary $\boldsymbol{y}$. For $\boldsymbol{\beta}$, $\boldsymbol{v}$ and $\boldsymbol{\gamma}$ the algebra is very similar to the Gaussian case. The only modifications are

$$\mu_{q(1/\sigma_\varepsilon^2)} \text{ replaced by } 1$$

$$\text{and } \boldsymbol{y} \text{ replaced by } \boldsymbol{\mu}_{q(\boldsymbol{a})}.$$

It remains to determine the form of $q^*(\boldsymbol{a})$ and $\boldsymbol{\mu}_{q(\boldsymbol{a})}$. Firstly, if $y_i = 1$ then

$$\begin{aligned} q^*(a_i) &\propto \exp\left\{E_q\left(-\tfrac{1}{2}[a_i - (\boldsymbol{X}\boldsymbol{\beta})_i - \{\boldsymbol{Z}(\boldsymbol{\gamma}\odot\boldsymbol{v})\}_i]^2\right)\right\}, \quad a_i \geq 0 \\ &\propto \exp\left(-\tfrac{1}{2}\left[a_i - (\boldsymbol{X}\mu_{q(\boldsymbol{\beta})})_i - \{\boldsymbol{Z}(\boldsymbol{\mu}_{q(\boldsymbol{\gamma})}\odot\boldsymbol{\mu}_{q(\boldsymbol{v})})\}_i\right]^2\right), \quad a_i \geq 0. \end{aligned}$$

Hence, if $y_i = 1$,

$$q^*(a_i) = \frac{\phi(a_i - (\boldsymbol{X}\mu_{q(\boldsymbol{\beta})})_i - \{\boldsymbol{Z}(\boldsymbol{\mu}_{q(\boldsymbol{\gamma})}\odot\boldsymbol{\mu}_{q(\boldsymbol{v})})\}_i)}{\Phi((\boldsymbol{X}\mu_{q(\boldsymbol{\beta})})_i + \{\boldsymbol{Z}(\boldsymbol{\mu}_{q(\boldsymbol{\gamma})}\odot\boldsymbol{\mu}_{q(\boldsymbol{v})})\}_i)}, \quad a_i \geq 0,$$

which is a truncated normal density function on $(0,\infty)$. Similarly, if $y_i = 0$, then

$$q^*(a_i) = \frac{\phi(a_i - (\boldsymbol{X}\mu_{q(\boldsymbol{\beta})})_i - \{\boldsymbol{Z}(\boldsymbol{\mu}_{q(\boldsymbol{\gamma})}\odot\boldsymbol{\mu}_{q(\boldsymbol{v})})\}_i)}{1 - \Phi((\boldsymbol{X}\mu_{q(\boldsymbol{\beta})})_i + \{\boldsymbol{Z}(\boldsymbol{\mu}_{q(\boldsymbol{\gamma})}\odot\boldsymbol{\mu}_{q(\boldsymbol{v})})\}_i)}, \quad a_i < 0.$$

Using moment results such as $\int_0^\infty x\phi(x-\mu)/\Phi(x)\,dx = \mu + \phi(\mu)/\Phi(\mu)$ we eventually obtain the expression

$$\begin{aligned} \boldsymbol{\mu}_{q(\boldsymbol{a})} &= \boldsymbol{X}\mu_{q(\boldsymbol{\beta})} + \boldsymbol{Z}(\boldsymbol{\mu}_{q(\boldsymbol{\gamma})}\odot\boldsymbol{\mu}_{q(\boldsymbol{v})}) \\ &\quad + \frac{(2\boldsymbol{y}-1)\odot\phi(\boldsymbol{X}\mu_{q(\boldsymbol{\beta})} + \boldsymbol{Z}(\boldsymbol{\mu}_{q(\boldsymbol{\gamma})}\odot\boldsymbol{\mu}_{q(\boldsymbol{v})}))}{\Phi((2\boldsymbol{y}-1)\odot\{\boldsymbol{X}\mu_{q(\boldsymbol{\beta})} + \boldsymbol{Z}(\boldsymbol{\mu}_{q(\boldsymbol{\gamma})}\odot\boldsymbol{\mu}_{q(\boldsymbol{v})})\})}. \end{aligned}$$

Appendix F: BUGS code for Section 5.3 analysis

This last appendix lists the BUGS code used to fit the subject-specific curve model given by (42) and (43). The notation in the code matches that used in Section 5.3. For example `Zgbl` corresponds to $\boldsymbol{Z}^{\text{gbl}}$, and this design matrix is constructed outside of BUGS and inputted as data.

```
model
{
   for (i in 1:numObs)
   {
      mu[i] <- (beta0 + beta1*x[i] + inprod(uGbl[],Zgbl[i,])
                      + U[idnum[i]] + inprod(uSbj[idnum[i],],Zsbj[i,]))
```

```
      y[i] ~ dnorm(mu[i],tauEps)
   }
   for (iSbj in 1:numSbj)
   {
      U[iSbj] ~ dnorm(0,tauU)
   }
   for (kGbl in 1:ncZgbl)
   {
      uGbl[kGbl] ~ dnorm(0,tauGbl)
   }
   for (iSbj in 1:numSbj)
   {
      for (kSbj in 1:ncZsbj)
      {
         uSbj[iSbj,kSbj] <- gamma[iSbj,kSbj]*vSbj[iSbj,kSbj]
         vSbj[iSbj,kSbj] ~ ddexp(0,tauSbj)
         gamma[iSbj,kSbj] ~ dbern(rho[iSbj,kSbj])
         rho[iSbj,kSbj] ~ dbeta(Arho,Brho)
      }
   }
   beta0  ~ dnorm(0,tauBeta)   ; beta1  ~ dnorm(0,tauBeta)
   tauEps ~ dgamma(0.5,recipAeps) ; AepsRecSq <- pow(Aeps,-2)
   recipAeps ~ dgamma(0.5,AepsRecSq)
   tauGbl ~ dgamma(0.5,recipAgbl) ; AgblRecSq <- pow(Agbl,-2)
   recipAgbl ~ dgamma(0.5,AgblRecSq)
   tauU ~ dgamma(0.5,recipAlin) ; AlinRecSq <- pow(Alin,-2)
   recipAlin ~ dgamma(0.5,AlinRecSq)
   tauSbj ~ dgamma(0.5,recipAsbj) ; AsbjRecSq <- pow(Asbj,-2)
   recipAsbj ~ dgamma(0.5,AsbjRecSq)
}
```

Acknowledgments

We thank the editor and an associate editor for their comments on an earlier version of this article. We are grateful for advice received from Peter J. Green, Eric D. Kolaczyk, Thomas C.M. Lee, Jeff S. Morris and Rui Song during the course of this research. We also thank Josue G. Martinez for supplying the respiratory pneumonitis study data. The first author thanks the Department of Statistics, Colorado State University, U.S.A., for its hospitality during the course of this research. This research was partially supported by Australian Research Council Discovery Project DP110100061.

Supplementary Material

Supplementary materials to "Penalized wavelets: Embedding wavelets into semiparametric regression"
(doi: 10.1214/11-EJS652SUPP; .zip).

References

[1] Albert, J.H. and Chib, S. (1993). Bayesian analysis of binary and polychotomous response data. *Journal of the American Statistical Association*, **88**, 669–679. MR1224394

[2] Antoniadis, A., Bigot, J. and Gijbels, I. (2007). Penalized wavelet monotone regression. *Statistics and Probability Letters*, **77**, 1608–1621. MR2393599

[3] Antoniadis, A. and Fan, J. (2001). Regularization of wavelet approximations (with discussion). *Journal of the American Statistical Association*, **96**, 939–967. MR1946364

[4] Antoniadis, A. and Leblanc, F. (2000). Nonparametric wavelet regression for binary response. *Statistics*, **34**, 183–213. MR1802727

[5] Attias, H. (1999). Inferring parameters and structure of latent variable models by variational Bayes. *Proceedings of the 15th Conference on Uncertainty in Artificial Intelligence*, 21–30.

[6] Aykroyd, R.G. and Mardia, K.V. (2003). A wavelet approach to shape analysis for spinal curves. *Journal of Applied Statistics*, **30**, 605–623. MR1986349

[7] Berry, S.M., Carroll, R.J. and Ruppert, D. (2002). Bayesian smoothing and regression splines for measurement error problems. *Journal of the American Statistical Association*, **97**, 160–169. MR1947277

[8] Bishop, C.M. (2006). *Pattern Recognition and Machine Learning.* New York: Springer. MR2247587

[9] Breheny, P. (2011). `ncvreg 2.3`. Regularization paths for SCAD- and MCP-penalized regression models. R package. `http://cran.r-project.org`

[10] Brumback, B.A. and Rice, J.A. (1998). Smoothing spline models for the analysis of nested and crossed samples of curves (with discussion). *Journal of the American Statistical Association*, **93**, 961–994. MR1649194

[11] Buja, A., Hastie, T. and Tibshirani, R. (1989). Linear smoothers and additive models. *The Annals of Statistics*, **17**, 453–510. MR0994249

[12] Carvalho, C.M., Polson, N.G. and Scott, J.G. (2010). The horseshoe estimator for sparse signals. *Biometrika*, **97**, 465–480. MR2650751

[13] Craven, P. and Wahba, G. (1979). Smoothing noisy data with spline functions: estimating the correct degree of smoothing by the method of generalized cross-validation. *Numerische Mathematik*, **31**, 377–403. MR0516581

[14] Currie, I.D. and Durbán, M. (2002). Flexible smoothing with P-splines: a unified approach. *Statistical Modelling*, **4**, 333–349. MR1951589

[15] Daubechies, I. (1988). Orthonormal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics*, **41**, 909–996. MR0951745

[16] Donoho, D.L. (1995). De-noising by soft-thresholding. *IEEE Transactions on Information Theory*, **41**, 613–627. MR1331258

[17] Donoho, D.L. and Johnstone, I.M. (1994). Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, **81**, 425–456. MR1311089

[18] Durbán, M., Harezlak, J., Wand, M.P. and Carroll, R.J. (2005). Simple fitting of subject-specific curves for longitudinal data. *Statistics in Medicine*, **24**, 1153–1167. MR2134571

[19] Efron, B. (2004). The estimation of prediction error: covariance penalties and cross-validation (with discussion). *Journal of the American Statistical Association*, **99**, 619–642. MR2090899

[20] Efron, B., Hastie, T., Johnstone, I. and Tibshirani, R. (2004). Least angle regression. *The Annals of Statistics*, **32**, 407–451. MR2060166

[21] Eilers, P.H.C. and Marx, B.D. (1996). Flexible smoothing with B-splines and penalties (with discussion). *Statistical Science*, **11**, 89–121. MR1435485

[22] Faes, C., Ormerod, J.T. and Wand, M.P. (2011). Variational Bayesian inference for parametric and nonparametric regression with missing data. *Journal of the American Statistical Association*, **106**, 959–971.

[23] Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, **96**, 1348–1360. MR1946581

[24] Fan, J. and Song, R. (2010). Sure independence screening in generalized linear models with NP-dimensionality. *The Annals of Statistics*, **38**, 3567–3604. MR2766861

[25] Fitzmaurice, G., Davidian, M., Verbeke, G. and Molenberghs, G. (Eds.) (2008). *Longitudinal Data Analysis: A Handbook of Modern Statistical Methods.* Boca Raton, Florida: Chapman & Hall/CRC.

[26] Frank, I.E. and Friedman, J.H. (1993). A statistical view of some chemometrics regression tools. *Technometrics*, **35**, 109–135.

[27] Friedman, J., Hastie, T. and Tibshirani, R. (2009). `glmnet 1.1`: lasso and elastic-net regularized generalized linear models. R package. http://cran.r-project.org

[28] Friedman, J., Hastie, T. and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, Volume 33, Issue 1, 1–22.

[29] Gelman, A. (2006). Prior distributions for variance parameters in hierarchical models. *Bayesian Analysis*, **1**, 515–533. MR2221284

[30] Gradshteyn, I.S. and Ryzhik, I.M. (1994). *Tables of Integrals, Series, and Products*, 5th Edition. San Diego, California: Academic Press. MR1243179

[31] Green, P.J. and Silverman, B.W. (1994). *Nonparametric Regression and Generalized Linear Models.* London: Chapman and Hall. MR1270012

[32] Griffin, J.E. and Brown, P.J. (2011). Bayesian hyper lassos with non-convex penalization. *Australian and New Zealand Journal of Statistics*, to appear.

[33] Hart, J.P., McCurdy, M.R., Ezhil, M., Wei W., Khan, M., Luo, D., Munden, R.F., Johnson, V.E. and Guerrero, T.M. (2008). Radiation pneumonitis: correlation and toxicity with pulmonary metabolic radiation

response. *International Journal of Radiation Oncology, Biology, Physics*, **4**, 967–971.

[34] Hastie, T. (1996). Pseudosplines. *Journal of the Royal Statistical Society, Series B*, **58**, 379–396. MR1377839

[35] Hastie, T. and Efron, B. (2007). `lars 0.9`. Least angle regression, lasso and forward stagewise regression. R package. http://cran.r-project.org

[36] Hastie, T.J. and Tibshirani, R.J. (1990). *Generalized Additive Models*. London: Chapman and Hall. MR1082147

[37] Hastie, T., Tibshirani, R. and Friedman, J. (2009). *The Elements of Statistical Learning*, Second Edition. New York: Springer. MR2722294

[38] Hurvich, C. M., Simonoff, J. S. and Tsai, C. (1998). Smoothing parameter selection in nonparametric regression using an improved Akaike information criterion. *Journal of the Royal Statistical Society, Series B*, **60**, 271–293. MR1616041

[39] Johnstone, I.M. and Silverman, B.W. (2005). Empirical Bayes selection of wavelet thresholds. *The Annals of Statistics*, **33**, 1700–1752. MR2166560

[40] Kerkyacharian, G. and Picard, D. (1992). Density estimation in Besov spaces. *Statistics and Probability Letters*, **13**, 15–24. MR1147634

[41] Ligges, U., Thomas, A., Spiegelhalter, D., Best, N. Lunn, D., Rice, K. and Sturtz, S. (2009). `BRugs 0.5`: OpenBUGS and its R/S-PLUS interface BRugs. http://www.stats.ox.ac.uk/pub/RWin/src/contrib/

[42] Marley, J.K. and Wand, M.P. (2010). Non-standard semiparametric regression via `BRugs`. *Journal of Statistical Software*, Volume 37, Issue 5, 1–30.

[43] Marron, J.S., Adak, S., Johnstone, I.M., Neumann, M.H. and Patil, P. (1998). Exact risk analysis of wavelet regression. *Journal of Computational and Graphical Statistics*, **7**, 278–309.

[44] Minka, T., Winn, J., Guiver, G. and Knowles, D. (2009). Infer.Net 2.4. Microsoft Research Cambridge, Cambridge, UK. http://research/microsoft.com/infernet

[45] Morris, J.S. and Carroll, R.J. (2006). Wavelet-based functional mixed models. *Journal of the Royal Statistical Society, Series B*, **68**, 179–199. MR2188981

[46] Morris, J.S., Vannucci, M., Brown, P.J. and Carroll, R.J. (2003). Wavelet-based nonparametric modeling of hierarchical functions in colon carcinogenesis. *Journal of the American Statistical Association*, **98**, 573–597. MR2011673

[47] Nason, G.P. (2008). *Wavelet Methods in Statistics with R*. New York: Springer. MR2445580

[48] Nason, G.P. (2010). `wavethresh 4.5`. Wavelets statistics and transforms. R package. http://cran.r-project.org

[49] Ormerod, J.T. and Wand, M.P. (2010). Explaining variational approximations. *The American Statistician*, **64**, 140–153. MR2757005

[50] Osborne, M.R., Presnell, B. and Turlach, B.A. (2000). On the LASSO and its dual. *Journal of Computational and Graphical Statistics*, **9**, 319–337. MR1822089

[51] O'Sullivan, F. (1986). A statistical perspective on ill-posed inverse problems (with discussion). *Statistical Science*, **1**, 505–527. MR0874480

[52] Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems*. San Mateo, California: Morgan Kaufmann. MR0965765

[53] Pericchi, L.R. and Smith, A.F.M. (1992). Exact and approximate posterior moments for a normal location parameter. *Journal of the Royal Statistical Society, Series B*, **54**, 793–804. MR1185223

[54] R Development Core Team (2011). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, http://www.R-project.org

[55] Robert, C.P. and Casella, G. (2004). *Monte Carlo Statistical Methods*, 2nd ed. New York: Springer. MR2080278

[56] Ruppert, D., Wand, M.P. and Carroll, R.J. (2003). *Semiparametric Regression*. New York: Cambridge University Press. MR1998720

[57] Ruppert, D., Wand, M.P. and Carroll, R.J. (2009). Semiparametric regression during 2003-2007. *Electronic Journal of Statistics*, **3**, 1193–1256. MR2566186

[58] Spiegelhalter, D.J., Thomas, A., Best, N.G., Gilks, W.R. and Lunn, D. (2003). BUGS: Bayesian inference using Gibbs sampling. Medical Research Council Biostatistics Unit, Cambridge, UK. http://www.mrc-bsu.cam.ac.uk/bugs.

[59] Staudenmayer, J., Lake, E.E. and Wand, M.P. (2009). Robustness for general design mixed models using the t-distribution. *Statistical Modelling*, **9**, 235–255. MR2756419

[60] Stein, C. (1981). Estimation of the mean of a multivariate normal distribution. *The Annals of Statistics*, **9**, 1135–1151. MR0630098

[61] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, **58**, 267–288. MR1379242

[62] Vidakovic, B. (1999). *Statistical Modeling by Wavelets*. New York: Wiley. MR1681904

[63] Wahba, G. (1990). *Spline Models for Observational Data*, Philadelphia, Pennsylvania: Society for Industrial and Applied Mathematics. MR1045442

[64] Wainwright, M.J. and Jordan, M.I. (2008). Graphical models, exponential families, and variational inference. *Foundation and Trends in Machine Learning*, **1**, 1–305.

[65] Wand, M.P. (2009). Semiparametric regression and graphical models. *Australian and New Zealand Journal of Statistics*, **51**, 9–41. MR2504101

[66] Wand, M.P. and Ormerod, J.T. (2008). On semiparametric regression with O'Sullivan penalized splines. *Australian and New Zealand Journal of Statistics*, **50**, 179–198. MR2431193

[67] Wand, M.P., Ormerod, J.T., Padoan, S.A. and Frühwirth, R. (2011). Mean field variational Bayes for elaborate distributions. *Bayesian Analysis*, to appear.

[68] Wang, Y. (1998). Mixed effects smoothing spline analysis of variance. *Journal of the Royal Statistical Society, Series B*, **60**, 159–174. MR1625640

[69] Wang, S.S.J. and Wand, M.P. (2011). Using Infer.NET for statistical analyses. *The American Statistician*, **65**, 115–126.

[70] Welham, S.J., Cullis, B.R., Kenward, M.G. and Thompson, R. (2007). A comparison of mixed model splines for curve fitting. *Australian and New Zealand Journal of Statistics*, **49**, 1–23. MR2345406

[71] Wood, S.N. (2003). Thin-plate regression splines. *Journal of the Royal Statistical Society, Series B*, **65**, 95–114. MR1959095

[72] Wood, S.N. (2006). *Generalized Additive Models: An Introduction with R.* Boca Raton, Florida: Chapman & Hall/CRC. MR2206355

[73] Wood, S.N. (2011). `mgcv 1.7`. GAMs with GCV/AIC/REML smoothness estimation and GAMMs by PQL. R package. `http://cran.r-project.org`

[74] Zhang, C.-H. (2010). Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, **38**, 894–942. MR2604701

[75] Zhang, D., Lin, X., Raz, J. and Sowers, M. (1998). Semi-parametric stochastic mixed models for longitudinal data. *Journal of the American Statistical Association*, **93**, 710–719 MR1631369

[76] Zhao, W. and Wu, R. (2008). Wavelet-based nonparametric functional mapping of longitudinal curves. *Journal of the American Statistical Association*, **103**, 714–725. MR2524004

[77] Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society, Series B*, **67**, 301–320. MR2137327

[78] Zou, H., Hastie, T. and Tibshirani, R. (2007). On the "degrees of freedom" of the lasso. *The Annals of Statistics*, **5**, 2173–2192. MR2363967