Statistica Sinica **20** (2010), 1143-1163

DATA DRIVEN ADAPTIVE SPLINE SMOOTHING

Ziyue Liu and Wensheng Guo

University of Pennsylvania

Abstract: In classical smoothing splines, the smoothness is controlled by a single smoothing parameter that penalizes the roughness uniformly across the whole domain. Adaptive smoothing splines extend this framework to allow the smoothing parameter to change in the domain, adapting to the change of roughness. In this article we propose a data driven method to nonparametrically model the penalty function. We propose to approximate the penalty function by a step function whose segmentation is data driven, and to estimate it by maximizing the generalized likelihood. A complexity penalty is added to the generalized likelihood in selecting the best step function from a collection of candidates. A state space representation for the adaptive smoothing splines is derived to ease the computational demand. To allow for fast search among the candidate models, we impose a binary tree structure on the penalty function and propose an efficient search algorithm. We show the consistency of the final estimate. We demonstrate the effectiveness of the method through simulations and a data example.

Key words and phrases: Binary tree, complexity penalty, generalized maximum likelihood, model selection, state space method.

1. Introduction

The central spirit of nonparametric smoothing is to let the data determine the amount of smoothing. In classical smoothing splines, the amount of smoothing is controlled by a single smoothing parameter, and considerable research has focused on how to choose the smoothing parameter using data driven criteria. When the homogeneity of the smoothness cannot be reasonably assumed across the whole domain, a natural extension is to allow the smoothing parameter to vary over the domain as a penalty function of the independent variable, adapting to the change of roughness (Robinson (1985), Wahba (1995), Pintore, Speckman, and Holmes (2006)). Similar to the classical smoothing splines, a key open problem in the adaptive smoothing is how to select the smoothing parameter, now a penalty function of the independent variable, by a data driven method. The goal of this paper is to develop a nonparametric method to model the penalty function that can adapt to the data structure automatically through an efficient algorithm.

Adaptive smoothing has long been an interesting topic in the statistical community. The basic solution is to allow the smoothing parameter, the bandwidth, or the placement of knots, to vary across the domain, adapting to the change of roughness (Müller and Stadtmüller (1987), Friedman and Silverman (1989), Brockmann, Gasser, and Herrmann (1993), Donoho and Johnstone (1994, 1995), Fan and Gijbels (1995), Luo and Wahba (1997), diMatteo, Genovese, and Kass (2001), Zhou and Shen (2001), Wood, Jiang, and Tanner (2002), Miyata and Shen (2003)). In penalized regression splines, Ruppert and Carroll (2000) modeled the penalty function by a linear interpolation on the logarithmic scale, Baladandayuthapani, Mallick, and Carroll (2005) modeled the penalty function from a full Bayesian approach and used Markov chain Monte Carlo for computation, and Krivobokova, Crainiceanu, and Kauermann (2008) developed a fast and simple algorithm for the Bayesian P-spline based on the Laplace approximation of the marginal likelihood. In smoothing splines, the adaptiveness can be achieved by modeling the smoothing parameter as a penalty function of the independent variable. This approach formulates the adaptive smoothing as a minimization problem with a new penalty function. As a result, the estimate has the same form as the smoothing spline and many existing methods developed for classical smoothing splines can be easily adapted. Robinson (1985) first mentioned this idea in the discussion of Silverman (1985). Wahba (1995) derived the reproducing kernels for a generic penalty function and suggested modeling it by B-splines. Pintore, Speckman, and Holmes (2006) studied the solution of the penalized least squares estimate in which the smoothing parameter is a varying function across the domain under the reproducing kernel Hilbert space (RKHS) approach.

The fundamental idea of nonparametric smoothing is to let the data choose the amount of smoothing, which consequently decides the model complexity needed for the data (Gu (1998)). Most of the research in this area focus on the development of data driven criteria such as cross validation (CV), generalized cross validation (GCV) (Craven and Wahba (1979)), and generalized maximum likelihood (GML) (Wecker and Ansley (1983), Wahba (1985)). The extension to adaptive smoothing splines poses new challenges in letting the data choose the optimal smoothing, as the smoothing parameter is now a varying function in the domain. The structure of the penalty function itself also controls the complexity of the final model, and needs to be determined from the data. The whole penalty function can then be estimated using some data driven criteria such as CV, GCV, or GML. The key challenge is how to impose a flexible yet parsimonious structure for the penalty function.

We propose to model the penalty function by a step function where the segmentation is data driven, for the following reasons. First, in smoothing splines

the penalty is on the mth derivative of the regression function, and the mth derivative is only assumed to be square integrable, not necessarily continuous. Therefore the penalty function should be allowed to have discontinuities. This can be seen in our numeric examples and data application. Second, even when the penalty function is continuous, a step function with data driven segmentation is a good approximation. The value of the step function in each segment is an average of the penalty function in that segment. Note that the step function approximation of the penalty function is different from the step function approximation of the regression function itself. From the Bayesian point of view, the penalty is only a smoothness prior (Wahba (1978)), and approximating the penalty function in a short segment by its mean usually can lead to a good estimate of the regression function that can be viewed as the posterior mean. This is also confirmed in our simulations.

An immediate consequence of the step function approximation is that the number of segments serves as a measure of complexity, which leads to a natural criterion for selecting the best segmentation. For a given segmentation, we can maximize the extended version of the generalized likelihood (Wecker and Ansley (1983), Wahba (1985)), which is the marginal likelihood from the equivalent Bayesian model. Borrowing a similar idea from the Akaike information criterion (AIC) (Akaike (1974)), we propose to penalize the complexity by the number of degrees of freedom used for the penalty function, which equals the number of segments. Thus an AIC-like model selection criterion is formulated. We term it "AIC-like" criterion as the generalized likelihood is not a true likelihood and its justification is only through the mathematical equivalence of the joint density function of the Bayesian model and the penalized least squares criterion (Wahba (1978)). This AIC-like criterion is straightforward to calculate and works well in our simulations even though it is not a true AIC. We also derive the state space representation of the adaptive smoothing splines and propose an $O(n)$ algorithm for model fitting and model selection to alleviate the computational burden.

Equipped with the step function approximation and the model selection criterion, we can theoretically fit the adaptive model with a flexible segmentation. However, without imposing any constraint on the possible segmentations, the total number of possible models is daunting. We propose to impose a binary tree structure on the possible segmentations with the depth prespecified. We then develop a search algorithm similar to the Best Basis algorithm (BBA) (Coifman and Wickerhauser (1992)) to enable fast and automatic search for the best segmentation.

The rest of the article is organized as follows. Section 2 introduces the background of smoothing splines and adaptive smoothing splines. Section 3 presents the proposed method and the asymptotic rate. Section 4 shows the visual quality

of the fitted results for some typical examples. Section 5 presents the results from an extensive simulation comparing the performance of our proposed algorithm with wavelet shrinkage, smoothing splines with prespecified smoothness pattern, and Bayesian adaptive P-spline. In Section 6 we apply our method to an epileptic electroencephalograms (EEG) data example.

2. Background

Consider the smoothing problem

$$y(t_i) = f(t_i) + \varepsilon_i, \quad i = 1, \ldots, n,$$

where f is the regression function, ε_i are independent and identically distributed with $\mathrm{E}(\varepsilon_i) = 0$ and $\mathrm{var}(\varepsilon_i) = \sigma^2$. In classical smoothing splines, f is estimated by

$$\min_{f \in \mathcal{W}_m} \left[\frac{1}{n} \sum_{i=1}^{n} \{y_i - f(t_i)\}^2 + \lambda \int_0^1 \{f^{(m)}(t)\}^2 \, dt \right], \tag{2.1}$$

where the Sobolev space $\mathcal{W}_m$ comprises functions that are absolutely continuous up to the $(m-1)$th derivative and have square integrable mth derivatives on $[0, 1]$.

When the homogeneity of smoothness cannot be reasonably assumed across the whole domain, a natural extension is to allow the smoothing parameter to vary over the domain, adapting to the smoothness pattern. Wahba (1995) suggested replacing (2.1) by

$$\min_{f \in \mathcal{W}_m} \left[\frac{1}{n} \sum_{i=1}^{n} \{y_i - f(t_i)\}^2 + \int_0^1 \lambda(t) \{f^{(m)}(t)\}^2 \, dt \right], \tag{2.2}$$

to achieve adaptive smoothing. She derived the corresponding reproducing kernel as

$$K_\lambda(s, t) = \int_0^1 \lambda(u)^{-1} \frac{(s-u)_+^{m-1}}{(m-1)!} \frac{(t-u)_+^{m-1}}{(m-1)!} \, du, \tag{2.3}$$

where $(t)_+ = \max(0, t)$.

Pintore, Speckman, and Holmes (2006) studied the solution and properties of adaptive smoothing splines under the RKHS approach for a given λ. They demonstrated the adaptive smoothing splines by imposing an equal-size piecewise structure for the penalty function, where the number of segments is prespecified and therefore is not data driven. Abramovich and Steinberg (1996) investigated the equivalent Bayesian model

$$F(t) = \sum_{j=1}^{m} d_j \phi_j(t) + \sigma \int_0^1 \lambda(s)^{-1/2} \frac{(t-s)_+^{m-1}}{(m-1)!} \, dW(s),$$

where $d = (d_1, \ldots, d_m) \sim \mathrm{N}(0, \kappa I)$, $\kappa \to \infty$, $\phi_j(t) = t^{j-1}/(j-1)!$, $W(s)$ is the standard Weiner process, and $y(t_i) = F(t_i) + \varepsilon_i$. Let $y = (y_1, \ldots, y_n)'$ and denote the solution to (2.2) as f_λ; they showed $\lim_{\kappa\to\infty} \mathrm{E}\{F(t)\,|y\} = f_\lambda(t)$.

3. The Proposed Method

In this section we approximate the penalty function as a step function with data driven segmentation. We derive the corresponding state space representation for efficient estimation. We propose a model selection criterion based on penalizing the generalized maximum likelihood. We develop an automatic algorithm to search for the optimal segmentation. We show the consistency of the final estimate.

3.1 Step function approximation

We propose to approximate the penalty function

$$\lambda(t) \approx \sum_{k=1}^{K} \lambda_k I_{t\in A_k}, \quad \lambda_k > 0, \quad k = 1, \ldots, K, \tag{3.1}$$

where $I_{t\in A_k}$ is the indicator function, $A_k = [\tau_{k-1}, \tau_k)$ and $0 = \tau_0 < \tau_1 < \cdots < \tau_K = 1$. The collection of all A_k forms a segmentation of the interval $[0, 1]$ that is uniquely defined by the number of segments K, and the lengths of each.

Minimization of the penalized least squares in (2.2) projects the function estimation problem from the infinite-dimensional Sobolev space onto a finite-dimensional subspace. Within the functional subspace, the regression function takes the form

$$f(t) = \sum_{j=1}^{m} d_j \phi_j(t) + \sum_{i=1}^{n} c_i \xi_i(t), \tag{3.2}$$

where $\xi_i(\cdot) = K_\lambda(t_i, \cdot)$, and the basis $\phi_j(\cdot)$ and $\xi_i(\cdot)$ span the functional subspace. For a given stepwise penalty function defined in (3.1), the reproducing kernel has a closed form (Pintore, Speckman, and Holmes (2006)). For $v \in (t_i, 1]$,

$$K_\lambda(t_i, v) = \sum_{k=1}^{K} \sum_{j=1}^{m} \lambda_k^{-1} (-1)^j \left\{ \frac{(t_i - \tau_k)_+^{m-1+j} (v - \tau_k)^{m-j}}{(m-1+j)!\,(m-j)!} - \frac{(t_i - \tau_{k-1})_+^{m-1+j} (v - \tau_{k-1})^{m-j}}{(m-1+j)!\,(m-j)!} \right\}$$

and, for $v < t_i$ with $v \in (\tau_l, \tau_{l+1})$,

$$K_\lambda(t_i, v) = \sum_{k=1}^{K}\sum_{j=1}^{m} \lambda_k^{-1}(-1)^j \left\{ \frac{(t_i-\tau_k)^{m-1+j}(v-\tau_k)_+^{m-j}}{(m-1+j)!\,(m-j)!} - \frac{(t_i-\tau_{k-1})^{m-1+j}(v-\tau_{k-1})_+^{m-j}}{(m-1+j)!\,(m-j)!} \right\} + \lambda_l^{-1}(-1)^m \frac{(t_i-v)^{2m-1}}{(2m-1)!},$$

Consequently the mth and higher derivatives have jumps at τ_k (Pintore, Speckman, and Holmes (2006)):

$$\left| f_\lambda^{(m-1+l)}\left(\tau_j^+\right) - f_\lambda^{(m-1+l)}\left(\tau_j^-\right) \right| = \left| \sum_{k=1}^{n} c_k \frac{(t_k-\tau_j)_+^{m-l}}{(m-l)!} \right| \left| \frac{\lambda\left(\tau_j^-\right) - \lambda\left(\tau_j^+\right)}{\lambda\left(\tau_j^-\right)\lambda\left(\tau_j^+\right)} \right|,$$

for $(1 \le l \le m)$.

Then one has $\xi_i(t) \in C^{2m-2}$ for $t \neq \tau_k$ and $\xi_i(t) \in C^{m-1}$ for $t = \tau_k$, where C^p denotes functions continuous up to the pth derivative. Consequently $f \in C^{2m-2}$ for $t \in (\tau_{k-1}, \tau_k)$, which has the same property as the classical smoothing splines; but $f \in C^{m-1}$ for $t = \tau_k$, which is less smooth than the classical smoothing splines. Hence the step function model not only allows different penalties for different segments, but also allows abrupt changes between two consecutive segments.

From the RKHS point of view, different segmentations define different basis $\xi_i(\cdot)$, which consequently span different functional subspaces. Adapting to the smoothness pattern is essentially finding the optimal functional subspace for the data.

3.2. Estimation through state space method

The RKHS estimation approach is computationally intensive. In this section, we propose an equivalent state space model extending Wecker and Ansley (1983). First consider the m-dimensional stochastic process. Write the $(m-\gamma)$th derivative of F, $F^{(m-\gamma)}$, as $x^{(\gamma)}$. Define $x(t) = \left[x^{(m)}(t), \ldots, x^{(1)}(t)\right]'$ with elements

$$x^{(\gamma)}(t) = \sum_{i=0}^{\gamma-1} d_{m-i}\frac{(t)^i}{i!} + \sigma\sum_{k=1}^{K}\lambda_k^{-1/2}\int_{\tau_{k-1}}^{\tau_k}\frac{(t-u)_+^{\gamma-1}}{(\gamma-1)!}\,dW(u),\ \gamma = m, \ldots, 1,$$

It is straightforward to show that $\lim_{\kappa\to\infty} \mathrm{E}\{x^{(m)}(t)\,|y\} = \lim_{\kappa\to\infty} \mathrm{E}\{F(t)\,|y\} = f_\lambda(t)$ following Wahba (1978).

Define the $m \times m$ matrix $H(t_i, t_j)$ by

$$H_{ij} = \begin{pmatrix} 1 & (t_i - t_j) & \dots & \frac{(t_i-t_j)^{m-1}}{(m-1)!} \\ 0 & 1 & \dots & \frac{(t_i-t_j)^{m-2}}{(m-2)!} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 1 \end{pmatrix}$$

and the $m \times 1$ random vector η_{ij} with element

$$\eta_{ij}^{(\gamma)} = \sigma\{\lambda_{l-1}^{-1/2} \int_{t_j}^{\tau_l} \frac{(t_i - h)^{\gamma-1}}{(\gamma-1)!}\, dW(h) + \sum_{q=l}^{u} \lambda_q^{-1/2} \int_{\tau_q}^{\tau_{q+1}} \frac{(t_i - h)^{\gamma-1}}{(\gamma-1)!}\, dW(h)$$
$$+\lambda_u^{-1/2} \int_{\tau_u}^{t_i} \frac{(t_i - h)^{\gamma-1}}{(\gamma-1)!}\, dW(h)\},\ \gamma = m, \dots, 1,$$

where $t_j \in [\tau_{l-1}, \tau_l)$, $t_i \in [\tau_u, \tau_{u+1})$ and $t_i > t_j$.

For any three time points $t_i \geq t_j \geq t_s$, it is straightforward to verify that $H_{is} = H_{ij}H_{js}$ and $\eta_{is} = H_{ij}\eta_{js} + \eta_{ij}$. Thus we have the state space representation

$$y_j = Zx(t_j) + e_j, x(t_j) = H_{j,j-1}x(t_{j-1}) + \eta_{j,j-1},$$
$$Z = [1 \quad 0 \cdots 0], \qquad \eta_{j,j-1} \sim \mathrm{N}(0, \Omega_{j,j-1}),$$

where $e_j \sim \mathrm{N}\left(0, \sigma^2\right)$ and $\Omega_{j,j-1}$ has pqth entry

$$\Omega_{j,j-1}(pq) = \lambda^{-1}(t_j)\, \sigma^2 \{\frac{(t_j - t_{j-1})^{2m+1-p-q}}{(2m+1-p-q)(m-p)!(m-q)!}\}.$$

The simple form of $\Omega_{j,j-1}$ is a direct result of the piecewise constant structure of λ.

Forward filtering and backward smoothing, which give the solution f_λ as the posterior mean, can be implemented in $O(n)$ steps. The algorithm is given in the on-line supplement at `http://www.stat.sinica.edu.tw/statistica/`. The readers are referred to Durbin and Koopman (2001) for details of the algorithm.

3.3. Parameter estimation and the model selection criterion

We first extend the generalized likelihood (Wahba (1985)) to the adaptive smoothing splines. Define the $n \times n$ matrix $\Sigma_\lambda = K_\lambda(t_i, t_j)_{i,j=1,\dots,n}$, and the $n \times m$ matrix $T = \{\phi_j(t_i)\}_{i=1,\dots,n, j=1,\dots,m}$. Let $T = (Q_1 : Q_2)\left(R^T : \mathbf{0}^T\right)^T$ be the QR decomposition of T, where Q_1 is $n \times m$, Q_2 is $n \times (n-m)$, $Q = (Q_1, Q_2)$ is orthogonal, and R is upper triangular. Let $z = Q_2'y$, which is independent of

$d = (d_1, \ldots, d_m)$ and $z \sim \mathrm{N}\left(0, \sigma^2\left(Q_2'\Sigma_\lambda Q_2 + I\right)\right)$. The generalized loglikelihood for $\theta = (\lambda_1, \ldots, \lambda_K, \sigma^2)$ based on z is

$$l\left(\theta|\, y\right) = -\frac{n-m}{2}\log\left(2\pi\right) - \frac{1}{2}\log\left[\det\{\sigma^2\left(Q_2'\Sigma_\lambda Q_2 + I\right)\}\right] \\ -\frac{1}{2}y'Q_2\{\sigma^2\left(Q_2'\Sigma_\lambda Q_2 + I\right)\}^{-1}Q_2'y.$$

In the state space method, we impose a diffuse prior on the initial state vector $x(0) \sim \mathrm{N}(0, \kappa I)$ with $\kappa \to \infty$. The loglikelihood calculated in the filtering step with this diffuse prior is the same as $l(\theta|y)$ defined above. The readers are referred to Koopman (1997) for the details on diffuse initialization. Maximizing $l(\theta|y)$ gives the GML estimate $\hat{\theta}$. Kohn, Ansley and Tharm (1991) showed by extensive simulation that GML outperforms GCV in many typical settings.

For model selection, we propose to penalize the complexity of the segmentation by the degrees of freedom in the penalty function, which is the number of segments, K. Thus an AIC like criterion is formulated, and we term it GAIC as generalized AIC,

$$\mathrm{GAIC} = -l\left(\hat{\theta}|y\right) + K.$$

When comparing two candidate segmentations, the one with smaller GAIC criterion is preferred. Computationally this criterion is efficient because it can be done simultaneously with parameter estimation by GML.

Similar to other penalized likelihood criteria, GAIC is a trade-off between the goodness of fit, $l(\hat{\theta}|y)$, and the complexity penalty. The penalty form is motivated by the classical AIC (Akaike (1974)), but the generalized likelihood is not a true likelihood, it is derived from the mathematically equivalent Bayesian model (Wahba (1985)). While this model selection criterion works well in our simulations, the theoretical properties require further investigation. A referee suggested that a BIC-like criterion can also be formulated based on the generalized likelihood. We study this in our simulation and it performs similarly to our proposed GAIC criterion. The key finding here is that generalized likelihood behaves similarly to a true likelihood even though it is not one.

3.4. Search algorithm

The number of possible segmentations without imposing any structure is 2^{n-2}, which is daunting. We therefore impose a binary tree structure on the step function. We first grow the binary tree to the maximal depth J, which needs to be prespecified. We then sequentially trim the leaves if trimming the subtree leads to a smaller overall GAIC.

For $j = 0, \ldots, J$, we write a partition of $[0, 1]$ at the jth level as

$$[0, 1] = B_{j,1} \oplus B_{j,2} \oplus \cdots \oplus B_{j,R_j},$$

where $\oplus$ denotes that two segments are kept separate and R_j is the number of segments in the partition. At the deepest level, $R_J = 2^J$ and for $r = 1, \ldots, R_j$, $B_{Jr} = [(r-1)/2^J, r/2^J)$. The number of possible models that can be generated by trimming the tree, denoted as M_J, can be calculated sequentially as $M_{J+1} = M_J^2 + 1$ (Coifman and Wickerhauser (1992)). Thus for $J = 0, 1, 2, 3, 4, 5$, $M_J = 1, 2, 5, 26, 677, 458330$, and $M_{J+1} \geq 2^{2^J}$.

We then propose an algorithm similar to BBA (Coifman and Wickerhauser (1992)) to search the binary tree in 2^J steps. For $j = J-1, J-2, \ldots, 0$, and $r = 1, \ldots, R_j$, we use the GAIC to determine whether to trim the subtree in the lower level. Define the two settings as

$$\begin{aligned} S_1: &\quad B_{j,1} \oplus \cdots \oplus B_{j,r-1} \oplus \{B_{j+1,2r-1} \cup B_{j+1,2r}\} \oplus B_{j+1,2r+1} \oplus \cdots B_{j+1,R_j} \\ S_2: &\quad B_{j,1} \oplus \cdots \oplus B_{j,r-1} \oplus B_{j+1,2r-1} \oplus B_{j+1,2r} \oplus B_{j+1,2r+1} \oplus \cdots B_{j+1,R^j}, \end{aligned}$$

where $\cup$ denotes collapsing two adjacent intervals. We start with $R_j = R_{j+1}$, and if $GAIC(S_1) \leq GAIC(S_2)$, trim the subtree, define $B_{j,r} = \{B_{j+1,2r-1} \cup B_{j+1,2r}\}$, and $R_j = R_j - 1$; if $GAIC(S_1) > GAIC(S_2)$, keep the subtree, define $B_{j,r} = B_{j+1,2r-1}$ and $B_{j,r+1} = B_{j+1,2r}$. When we finish the jth level, R_j is the number of segments kept at the jth level and eventually R_0 is the final number of segments chosen in the model, denoted as K previously. While J can grow with the sample size n, K is much smaller than number of initial segments (2^J), and is assumed to be fixed, determined only by the underlying true signal.

In practice the maximal depth J needs to be chosen according to the sample size and computational constraint. Starting with a larger J we allow the true underlying penalty function to be approximated in smaller steps, but this is done at the heavy computational expense. Another consideration is that enough observations are needed in each segment to ensure a reliable estimate of the smoothing parameter for that segment. In classical smoothing splines, $25 - 30$ observations are usually needed to ensure a reliable estimate of smoothing parameters (Wahba (1990, p.65)). We find this to be true in adaptive smoothing splines through our simulations. Therefore $J \leq (\log n - \log 25)/\log 2$. Our recommendation is that for a complex signal, one can choose to grow the tree to the maximal depth where, at the finest level, each segment has only 25 observations. When the roughness of the signal does not change rapidly, smaller J can be adopted to ease the computational demand.

3.5. Bayesian confidence intervals

Many results in smoothing splines can be easily extended for adaptive smoothing splines. The Bayesian confidence intervals are defined pointwise (Wahba (1983)) as

$$[C_L, C_U]_j = \lim_{\kappa \to \infty} \left(\mathrm{E}\{F(t_j)\,|y\} \pm Z_{\alpha/2} \times [\mathrm{Var}\{F(t_j)\,|y\}]^{1/2} \right),$$

where Z_α is the $(1-\alpha)$100th standard normal percentile. Although confidence intervals are calculated pointwise, they have a nice curvewise coverage rate from a frequentist viewpoint. Define the average coverage probability(ACP$(1-\alpha)$) at the nominal $1-\alpha$ level as

$$ACP\,(1-\alpha) = \frac{1}{n}\sum_{j=1}^{n} \mathrm{I}\{f\,(t_j) \in [C_L, C_U]_j\},$$

Nychka (1988) showed that ACP is close to its nominal level both asymptotically and by simulation.

3.6. Consistency

In this section we prove the consistency of the adaptive smoothing spline. While the maximal depth of the binary tree J usually depends on the sample size n, we assume the final number of segments K does not depend on n. This means that the optimal segmentation only depends on the structure of the underlying signal. The results are summarized by the following lemma and theorem.

Lemma. *Let Σ have elements $\Sigma(i,j) = K(t_i, t_j)$ where $K(\cdot,\cdot)$ is the reproducing kernel for classical smoothing splines. Let $\delta_{1n}^* \geq \cdots \geq \delta_{nn}^*$ be the ordered eigenvalues of Σ, similarly $\delta_{1n} \geq \cdots \geq \delta_{nn}$ be the ordered eigenvalues of Σ_λ. Let $\lambda_{min} = \min\{\lambda(t)\}$ and $\lambda_{max} = \max\{\lambda(t)\}$. Then for every i,*

$$\lambda_{max}^{-1}\delta_{in}^* \leq \delta_{in} \leq \lambda_{min}^{-1}\delta_{in}^*. \tag{3.3}$$

Theorem. *The integrated risk, IR, is of the order*

$$\mathrm{IR}_n\,(\lambda) = \int_0^1 \mathrm{E}\{f\,(t) - f_\lambda\,(t)\}^2 p\,(t)\;dt \leq O\,(\lambda_{max}) + O\left(\lambda_{min}^{-1/2m} n^{-1}\right),$$

where the design density $p\,(t)$ satisfies $\int_0^{t_j} p\,(t)\;dt = (2j-1)/(2n)$. As $n \to \infty$, $\lambda_{min} = O(n^{-2m/(2m+1)})$ and $\lambda_{max} = O(n^{-2m/(2m+1)})$, the asymptotic rate of IR is $O(n^{-2m/(2m+1)})$.

The proofs are given in the on-line supplement at `http://www.stat.sinica.edu.tw/statistica/`.

As in classical smoothing splines, IR can be decomposed into bias and variance parts. The proof shows that the bias part is bounded above by λ_{max}. If we choose λ_{max} as the global smoothing parameter, the bias will achieve its upper bound, but regions other than the one corresponding to λ_{max} will be oversmoothed. On the other hand, the variance part is bounded above by $\lambda_{min}^{-1/2m}$. If we choose λ_{min} as the global smoothing parameter, the variance will achieve its

upper bound, but regions other than the one corresponding to λ_{min} will be undersmoothed. By letting λ_{max} and λ_{min} have the same rate, we get the optimal rate $O(n^{-2m/(2m+1)})$, which is the same as for classical smoothing splines.

Adaptive smoothing allows different balances of bias and variance for different roughness patterns. Therefore we expect that the finite sample performance of adaptive smoothing splines, for signals with strong roughness heterogeneity, will outperform classical smoothing splines. This is confirmed by our simulations in Section 5.

4. Numerical Examples

In this section we examine the visual quality of the proposed method and the resultant penalty functions. We work with four functions: Blocks, Bumps, HeaviSine, and Doppler, used in Donoho and Johnstone (1994) and Donoho and Johnstone (1995). These four functions are examples where the classical smoothing spline does not work well because of smoothness inhomogeneity.

We use the same settings as in Donoho and Johnstone (1994) and Donoho and Johnstone (1995): $n = 2,048$, with independent Gaussian noise $\varepsilon_i \sim \mathrm{N}(0, 1)$. Signals are rescaled so that the signal-to-noise ratio (SNR) is 7. For wavelet reconstruction we use the hybrid version of SureShrink, with levels $j = 5, \ldots, 10$. We work with $m = 2$, thus cubic smoothing splines from this point on, and we take $J = 4$ as the depth of the tree.

Figure 1 to Figure 4 present those four functions with the true signal, the simulated data and the smoothed results from wavelet SureShrink and the proposed method. In general the visual qualities of the proposed method and Wavelet SureShrink are comparable. Wavelet SureShrink is better at uniforml denoising while small wiggles are left over the whole domain. For Blocks, over some region the proposed method does not smooth out large noises, while it is better at smoothing out almost all wiggles over long flat regions. For Bumps, the proposed method is better than SureShrink almost everywhere except it is less denoised for a short interval around 0.2. For HeaviSine and Doppler the proposed method and SureShrink are comparable.

Figure 5 shows the estimated penalty functions. The penalty functions do adapt to the smoothness pattern of the signals, for example for the HeaviSine signal, two small penalties are chosen for intervals around 0.3 and 0.7, where there are two abrupt changes. For Blocks, Bumps and HeaviSine, the differences of $\hat{\lambda}_k$s between two consecutive segments are so large that continuity cannot be reasonably assumed. The estimated penalty function for Doppler implies a continuous λ, while f_λ under the stepwise penalty function still looks good.

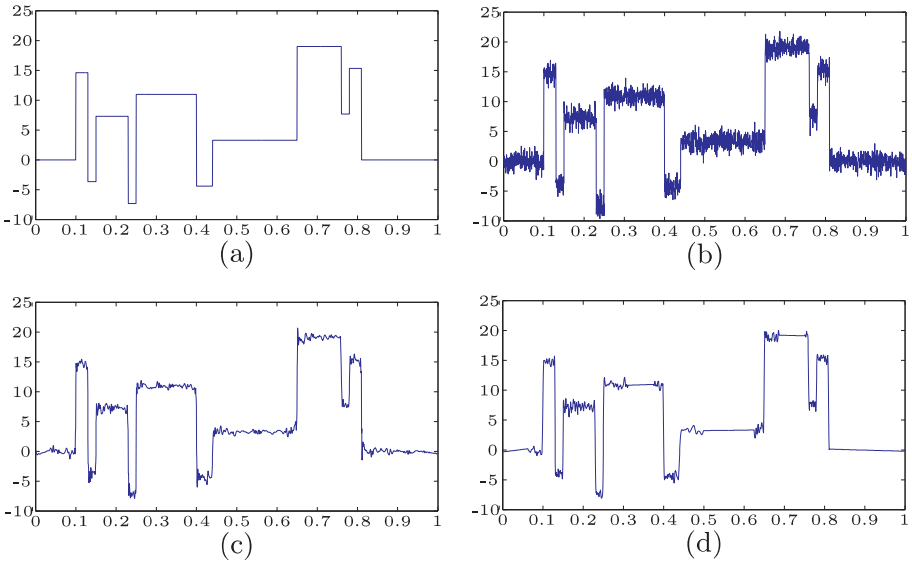

Figure 4.1. Numerical example: *Blocks.* (a) The true function; (b) with i.i.d. Gaussian noise at SNR=7; (c) reconstruction from Wavelet hybrid SureShrink; (d) final estimate from the proposed method.

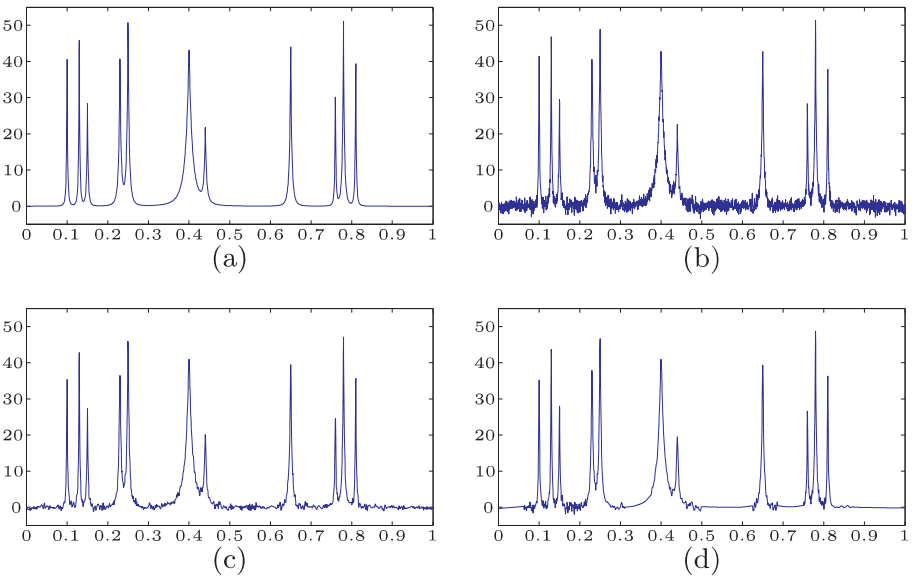

Figure 4.2. Numerical example: *Bumps.* (a) The true function; (b) with i.i.d. Gaussian noise at SNR=7; (c) reconstruction from Wavelet hybrid SureShrink; (d) final estimate from the proposed method.

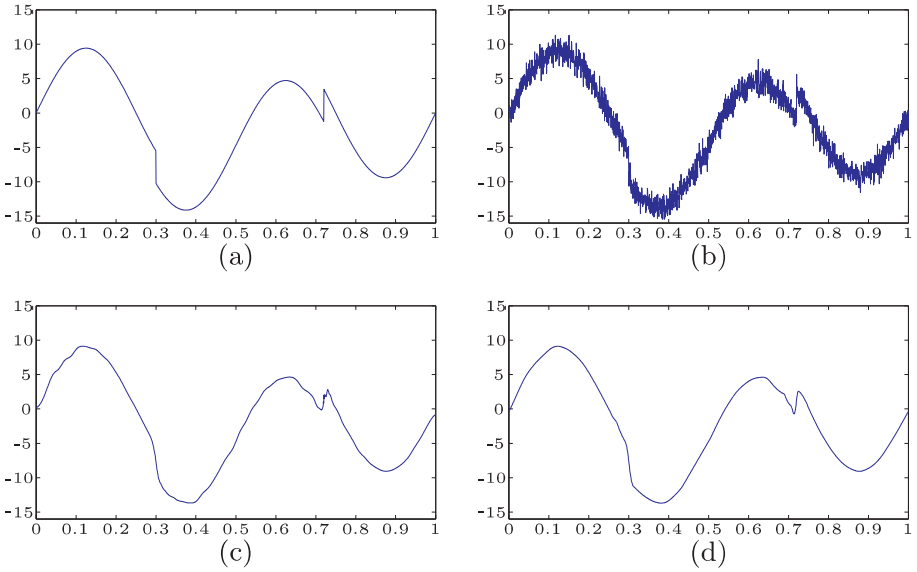

Figure 4.3. Numerical example: *HeaviSine*. (a) The true function; (b) with i.i.d. Gaussian noise at SNR=7; (c) reconstruction from Wavelet hybrid SureShrink; (d) final estimate from the proposed method.

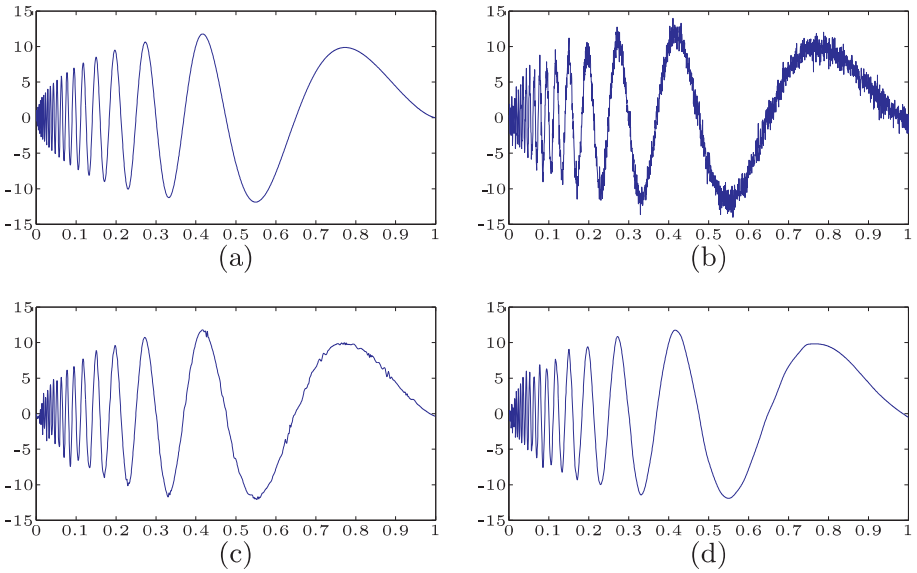

Figure 4.4. Numerical example: *Doppler*. (a) The true function; (b) with i.i.d. Gaussian noise at SNR=7; (c) reconstruction from Wavelet hybrid SureShrink; (d) final estimate from the proposed method.

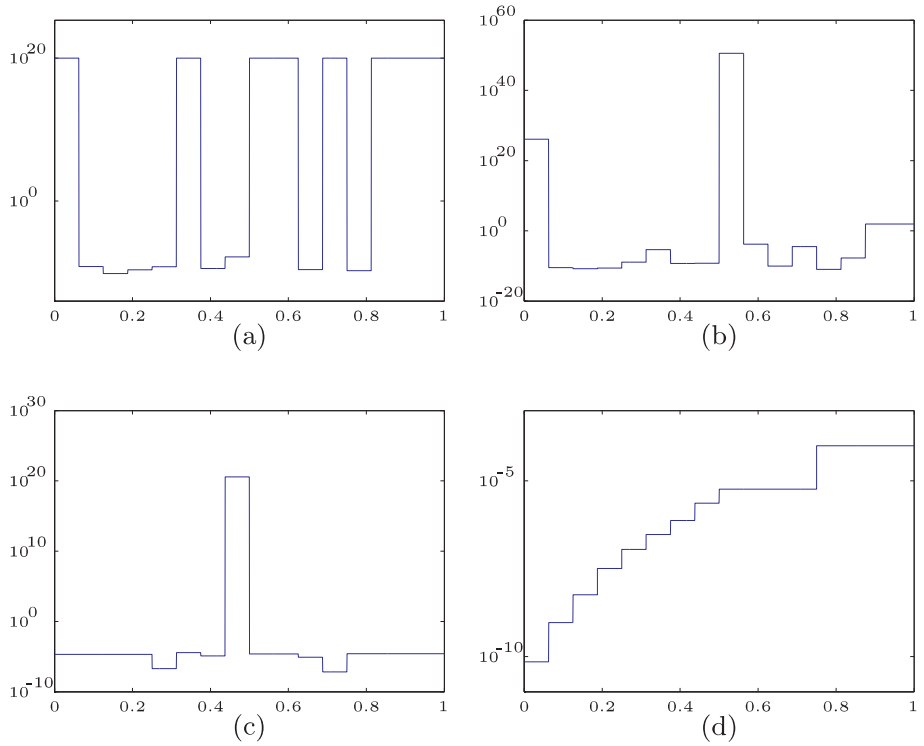

Figure 4.5. Estimated penalty functions: (a) Blocks; (b) Bumps; (c) HeaviSine; (d) Doppler. The penalty for Blocks is truncated above at 10^{20}.

5. Simulation

We conducted a simulation to investigate the performance of the proposed method by comparing the true mean square errors(TMSE),

$$\text{TMSE} = \frac{1}{n}\sum\{f(t_i) - f_\lambda(t_i)\}^2.$$

We also examined the performance of Bayesian confidence intervals by ACP at the nominal level of 95%. Besides the four functions used in the previous section, we added two more sine functions: Sin-141 and Sin-1414. For Sin-141, we divided $[0,1]$ into three intervals of equal-length $B_1 \oplus B_2 \oplus B_3$ and the signal was generated as $\sin(6\pi t)\text{I}\{t \in (B_1, B_3)\} + \sin(24\pi t)\text{I}(t \in B_2)$ where $I(.)$ is the indicator function. For Sin-1414, we divided $[0,1]$ into four intervals of equal-length $B_1 \oplus B_2 \oplus B_3 \oplus B_4$ and the signal was generated as $\sin(6\pi t)\text{I}\{t \in (B_1, B_3)\} + \sin(24\pi t)\text{I}\{t \in (B_2, B_4)\}$.

In adaptive smoothing, Blocks and Bumps represent frequent and irregular abrupt changes in smoothness. HeaviSine represents slow change but for a few

abrupt changes. Doppler represents gradual changes. In Sin-1414, the binary tree structure covered the true. We expect the proposed method to select the true step function almost every time. In Sin-141, the binary tree structure did not cover the true. We ran the proposed method, a smoothing spline with one global smoothing parameter (SS1), a smoothing spline with four smoothing parameters on equal-sized segments (SS4), a smoothing spline with eight smoothing parameters on equal-sized segments (SS8), the wavelet hybrid SureShrink, and the Bayesian adaptive P-spline (Krivobokova, Crainiceanu, and Kauermann (2008)) by R package AdaptFit. For computational consideration we let $n = 1,024$. We tried $J = 4$ and $J = 5$ as the depth of the binary tree for a few replicates in our simulations, and found that they performed similarly. To speed up the simulation, we used $J = 4$ for the full scale simulation. Independent and identically distributed Gaussian noises were added and the signals were scaled to have SNR levels of 7 and 3, strong and weak signals, respectively. Each setting was repeated 100 times. The median TMSE and the range between the first and the third quartiles are summarized in Table 1. The median ACP for the proposed method varied from 0.939 to 0.962, which are close to the nominal level 0.95.

First we compared the performance of the proposed method with Bayesian adaptive P-spline. We set the options of AdaptFit as: 80 knots for the regression function; 20 knots for the penalty function; 100 maximal iterations for the mean function (the default is 20); 1,000 maximal iterations for the variance of random effects estimation (the default is 50). As pointed out by T. Krivobokova, the author of AdaptFit, Bayesian adaptive P-spline is developed to smooth continuous functions with continuous smoothing parameters. Therefore for functions with abrupt changes like Blocks, Bumps and HeaviSine, AdaptFit is not supposed to perferm well, which was confirmed by our simulations. AdaptFit also has a convergence problem. For all the settings except for HeaviSine signal, there were always some cases that would not converge despite increased maximal iteration numbers. Following a suggestion from T. Krivobokova, we also tried 160 knots for the regression function in Doppler signal. Subsequently the percentage of converged runs increased and TMSE decreased: at SNR 7, TMSE decreased from 1.936 to 0.659, at SNR 3 TMSE decreased from 0.383 to 0.164. This indicates that since P-spline uses only a subset of knots, it cannot adapt to abrupt changes as quickly as the proposed method that uses all the data points as knots. In addition adaptive P-spline models the penalty function as continuous, but in situations like Blocks and Heavisine, the roughness changes abruptly and the step function approximation used by the proposed method is more robust.

We compared the proposed method with Wavelet, SS1, SS4 and SS8. For Blocks, Bumps, HeaviSine and Doppler, Table 1 shows that in general the proposed method outperformed other methods; the better performance is more obvious at higher SNR levels. There were two exceptions: for Blocks at SNR

Table 5.1. TMSE result: Median (Range).

	SNR	Proposed	SS1	SS4	SS8	SureShrink	P-spline
Blocks	7	0.5447	1.0720	1.0041	0.8008	0.4809	2.5656
		(0.0420)	(0.0568)	(0.0523)	(0.0540)	(0.0436)	(0.0264)
	3	0.2732	0.4018	0.3903	0.3403	0.3669	0.5408
		(0.0248)	(0.0237)	(0.0247)	(0.0293)	(0.0324)	(0.0200)
Bumps	7	0.3900	1.0288	0.8122	0.5244	0.5845	†
		(0.0382)	(0.0847)	(0.0663)	(0.0427)	(0.0530)	
	3	0.3018	0.6326	0.5451	0.3980	0.4887	‡
		(0.0348)	(0.0464)	(0.0501)	(0.0335)	(0.0545)	
HeaviSine	7	0.0691	0.1212	0.0999	0.1293	0.1309	0.1272
		(0.0140)	(0.0133)	(0.0113)	(0.0866)	(0.0193)	(0.0140)
	3	0.0464	0.0485	0.0460	0.0753	0.0645	0.0518
		(0.0148)	(0.0117)	(0.0123)	(0.0265)	(0.0160)	(0.0090)
Doppler	7	0.1125	0.4702	0.1695	0.1310	0.2740	1.9360
		(0.0190)	(0.0441)	(0.0240)	(0.0269)	(0.0315)	(0.0114)
	3	0.0881	0.2645	0.1259	0.1048	0.1834	0.3833
		(0.0184)	(0.0286)	(0.0210)	(0.0251)	(0.0260)	(0.0098)
Sin-1414	7	0.0893	0.1127	0.0893	0.0898	0.0995	0.1450
		(0.0178)	(0.0168)	(0.0178)	(0.0181)	(0.0184)	(0.0158)
	3	0.0624	0.0807	0.0624	0.0631	0.0707	0.0728
		(0.0157)	(0.0139)	(0.0157)	(0.0159)	(0.0156)	(0.0137)
Sin-141	7	0.0608	0.0849	0.0661	0.0664	0.0814	0.0518
		(0.0150)	(0.0156)	(0.0156)	(0.0159)	(0.0162)	(0.0129)
	3	0.0438	0.0596	0.0466	0.0583	0.0572	0.0401
		(0.0127)	(0.0132)	(0.0124)	(0.0215)	(0.0157)	(0.0142)

†: None of 1,000 runs converged.
‡: 11 of 1,000 runs converged, the median value was 3.1745.

7, wavelet SureShrink performed slightly better than the proposed method; for HeaviSine at SNR 3, SS4 performed slightly better than the proposed method. For Sin-1414, the proposed method outperformed other methods. This is due to the fact that the binary tree covered the true structure and indeed chose the true for every repeat. For Sin-141 the proposed method outperformed the others. In this case the true structure was not covered by the binary tree, but we can see that the approximation performed reasonably well.

6. Application to an EEG data

Epilepsy is one of the most common neurological disorders. About one quarter of epileptic seizure cannot be controlled by medication or surgery. The idea of predicting seizure so that preventive treatment can be given before the clinical onset has fascinated neurologists since the 1970s (see Mormann et al. (2007) and references therein). As part of the effort to construct a statistical predic-

tion framework, our first step was to characterize EEG. Early research on EEG from epilepsy patients focused on high voltage low frequencies ($\leq$ 25Hz), which exhibit a continuously build-up of energy before seizure. Recent biomedical research and statistical analysis of the time-varying spectral (Qin, Guo and Litt (2009)) show that the low voltage frequency band $26-50$Hz also has an important role in epileptic seizure. The rapid discharges in frequency band $26-50$Hz might indicate seizure's spatial-temporal organization.

Accordingly, studying the high frequencies may help neurologists determine the spatial-temporal initiation of seizure. Figure 6(a) shows a 15-minute long intracranial EEG series. The sampling rate is $200Hz$ and the seizure onset is at the $8th$ minute (Litt et al. (2001)). For every half second we calculated the time-varying log-spectral band power of $26-50$Hz,

$$y(t) = \sum_{\nu=26Hz}^{50Hz} \log\{p_\nu(t)\},$$

where $p_\nu(t)$ is the raw periodograms at frequency ν and time t. The band powers are usually very noisy, as shown in Figure 6 (b), and smoothing is needed before further analysis. Figure 6 (e) shows the fit of the classical smoothing spline. The global smoothing parameter shown in Figure 6 (f) is a result of compromising between the flat regions and the abrupt-changing regions in order to capture the big jump. This compromise led to under-smoothing the flat regions and over-smoothing the abrupt changing regions. As a consequence the information around seizure was lost and small false waves were left everywhere. This impedes the effort in identifying seizure initiation. For competing adaptive smoothing methods, wavelet SureShrink did not smooth the data much as shown in Figure 6 (c); Bayesian adaptive P-spline by AdaptFit did not converge with maximal iteration number of 100,000.

Figure 6(d) shows the fit from the proposed method. The profile before seizure is almost a straight line, and more details are preserved before the onset of seizure. The post-seizure region shows a smooth recovery. This is a big contrast to the result from the classical smoothing spline where artifacts were generated in the before and after seizure periods due to undersmoothing. For this given EEG series, the band power of $26-50$Hz shows little change until 33 seconds before the seizure. Then it begins to fluctuate and keeps fluctuating for the next 33 seconds; this may be a meaningful predictor of seizure and may help identify the spatiotemporal initiation of seizure. There is a sharp increase at the beginning of the seizure and a sharp decrease at the end of the seizure. After seizure, the band power drops below the pre-seizure level, indicating a suppression of neuron activities. Then it begins to regain slowly, but not back to the pre-seizure level

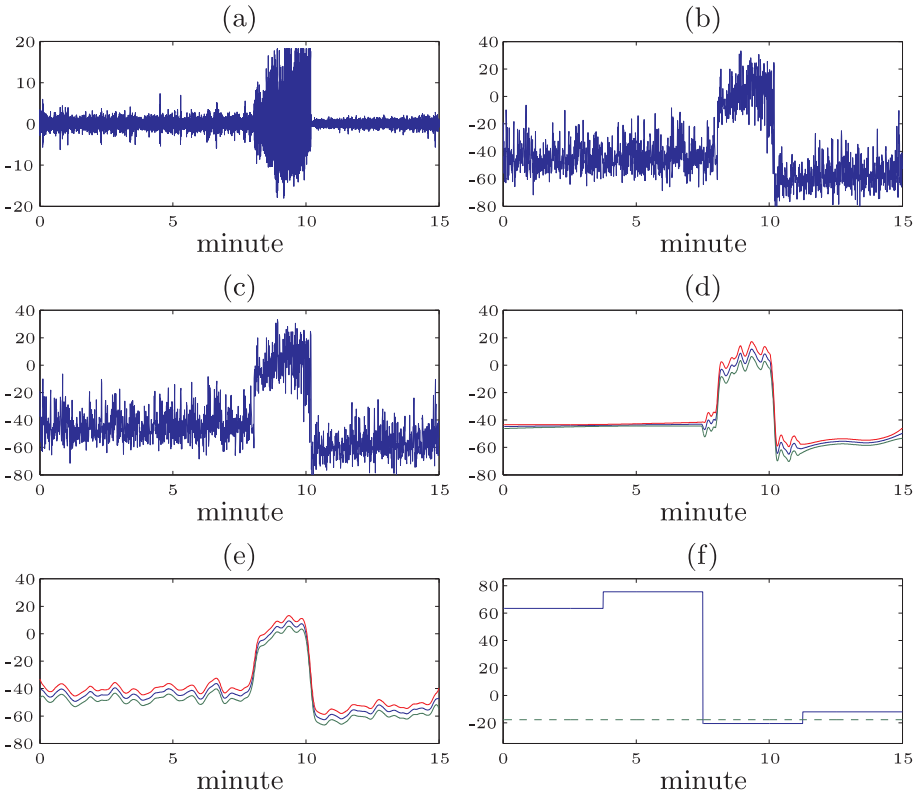

Figure 6.6. *EEG data example.* (a)Raw series. (b) log spectral band power. (c) Reconstruction from Wavelet hybrid SureShrink. (d) Final estimate from the proposed method with 95% Bayesian Confidence intervals. (e) Final estimte from smoothing spline with 95% Bayesian confidence intervals. (f) Estimated penalty functions on the logarithm scale: broken line for the smoothing spline, solid line for the proposed method.

after 5 minutes. This suggests that it takes more than 5 minutes for neurons to recover.

7. Discussion

We have proposed a data driven method to model the penalty function in adaptive smoothing splines. This method outperformed wavelet shrinkage and Bayesian adaptive P-spline in our simulations when the roughness of the underly signal changed rapidly. This is due to the robustness of the step-function approximation to the true underlying penalty function. Another key finding is that, while the generalized likelihood is not a true likelihood, the generalized likelihood based model selection criteria, such as the proposed GAIC, performed well in our simulation. However, our method is computationally expensive compared to wavelet

shrinkage and Bayesian P-splines, mainly due to the large number of candidate models and the proposed tree searching algorithm. Despite the $O(n)$ algorithm of the proposed equivalent state space model, our method still takes significant longer computation time than does the Bayesian P-spline. For example, in our simulation setting of $n = 1,024$ using a regular desktop PC, Wavelet SureShrink only took seconds, the Bayesian P-spline method took a few minutes, while the proposed method required 40-50 minutes. This heavy computational demand is a serious limitation and motivates us to explore faster search algorithms for the best segmentation.

Acknowledgements

We would like to thank the Joint Editors, the Associate Editor, and two referees for their helpful comments that significantly improved the presentation of the paper. The research was supported by NCI R01 grant CA84438 and NIDDK training grant T32 DK60455. Please send correspondence to the second author.

References

Abramovich, F. and Steinberg, D. M. (1996). Improved inference in nonparametric regression using L_k-smoothing splines. *J. Statist. Plann. Infer.* **49**, 327-341.

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Trans. Auto. Cont.* **19**, 716-723.

Baladandayuthapani, V., Mallick, B. K. and Carroll, R. J. (2005). Spatially adaptive Bayesian penalized regression splines (P-splines). *J. Comput. Graph. Statist.* **14**, 378-394.

Brockmann, M., Gasser, T., and Herrmann, E. (1993). Locally adaptive bandwidth choice for kernel regression estimators. *J. Amer. Statist. Assoc.* **88**, 1302-1309.

Craven, P. and Wahba, G. (1979). Smoothing noise data with spline functions. *Numer. Math.* **31**, 377-403.

Coifman, R. R. and Wickerhauser, M. V. (1992). Entropy-based algorithm for best basis selection. *IEEE Trans. Inform. Theory*, **38**, 713-718.

diMatteo, I., Genovese, C. R. and Kass, R. E. (2001). Baysian curve-fitting with free-knot splines. *Biometrika* **88**, 2055-1071.

Donoho, D. L. and Johnstone, I. M. (1994). Ideal spatial adaptation by wavelet shrinkage. *Biometrika* **81**, 425-455.

Donoho, D. L. and Johnstone, I. M. (1995). Adapting to unknown smoothness via wavelet shrinkage. *J. Amer. Statist. Assoc.* **90**, 1200-1224.

Durbin, J. and Koopman, S. J. (2001). *Time Series Analysis by State Space Methods.* Oxford, New York.

Fan, J. and Gijbels, I. (1995). Data-driven bandwidth selection in local polynomial fitting: variable bandwidth and spatial adaptation. *J. Roy. Statist. Soc. Ser. B* **57**, 371-394.

Friedman, J. H. and Silverman, B. W. (1989). Flexible parsimonious smoothing and additive modeling. *Technometrics* **31**, 3-21.

Gu, C. (1998). Model indexing and smoothing parameter selection in nonparametric function estimation. *Statist. Sinica.* **8**, 607-646.

Kohn, R., Ansley, C. F. and Tharm, D. (1991). The performance of cross-validation and maximum likelihood estimators of spline smoothing parameters. *J. Amer. Statist. Assoc.* **86**, 1042-1050.

Koopman, S. J. (1997). Exact initial Kalman filtering and smoothing for non-stationary time series models. *J. Amer. Statist. Assoc.* **92**, 1630-1638.

Krivobokova, T., Crainiceanu, C. M. and Kauermann, G. (2008). Fast adaptive penalized splines. *J. Comput. Graph. Statist.* **17**, 1-20.

Litt, B., Esteller, R., Echauz, J., D'Alessandro, M., Shor, R., Henry, T., Pennell, P., Epstein, C., Bakay, R., Dichter, M. and Vachtsevanose, G. (2001). Epileptic seizures may begin hours in advance of clinical onset: a report of five patients. *Neuron* **30**, 51-64.

Luo, Z. and Wahba, G. (1997). Hybrid adaptive splines. *J. Amer. Statist. Assoc.* **92**, 107-116.

Miyata, S. and Shen, X. (2003). Adaptive free-knot splines.*J. Comput. Graph. Statist.* **12**, 197-213.

Mormann, F., Andrzejak, R. G., Elger, C. E. and Lehnertz, K. (2007). Seizure prediction: the long and winding road. *Brain* **130**, 314-333.

Müller, H. G. and Stadtmüller, U. (1987). Variable bandwidth kernel estimation of regression curves. *Ann. Statist.* **15**, 182-201.

Nychka, D. (1988). Bayesian confidence intervals for smoothing splines. *J. Amer. Statist. Assoc.* **83**, 1134-1143.

Pintore, A., Speckman, P. and Holmes, C. C. (2006). Spatially adaptive smoothing splines. *Biometrika* **93**, 113-125.

Qin, L., Guo, W. and Litt, B. (2009). A time-frequency functional model for locally stationary time series data. *J. Comput. Graph. Statist.* To appear.

Robinson, A. (1985). *In discussion* Some aspects of the spline smoothing approach to nonparametric regression curve fitting, by Silverman, B. W. *J. Roy. Statist. Soc. Ser. B* **47**, 50.

Ruppert, D. and Carroll, R. J. (2000). Spatially-adaptive penalties for spline fitting. *Aust. N. Z. J. Statist.* **42**, 205-223.

Silverman, B. W. (1985). Some aspects of the spline smoothing approach to nonparametric regression curve fitting (with discussion). *J. R. Stat. Soc. Ser. B* **47**, 1-52.

Wahba, G. (1978). Improper priors, spline smoothing and the problem of guarding against model errors in regression. *J. Roy. Statist. Soc. Ser. B* **40**, 364-372.

Wahba, G. (1983). Bayesian "confidence intervals" for the cross validated smoothing spline. *J. Roy. Statist. Soc. Ser. B* **45**, 133-150.

Wahba, G. (1985). A comparison of GCV and GML for choosing the smoothing parameter in the generalized spline smoothing problem. *Ann. Statist.* **13**, 1378-1402.

Wahba, G. (1990). *Spline Models for Observational Data.* CBMS-NSF Regional Conference Series in Applied Mathematics. SIAM, Philadelphia.

Wahba, G. (1995). *In discussion* Wavelet shrinkage: Asymptopia? by Donoho, D. L. and Johnstone, I.M. *J. R. Stat. Soc. Ser. B* **57**, 360-361.

Wecker, W. E. and Ansley, C. F. (1983). The signal extraction approach to nonlinear regression and spline smoothing. *J. Amer. Statist. Assoc.* **78**, 81-89.

Wood, S. A., Jiang, W. and Tanner, M. (2002). Bayesian mixture of splines for spatially adaptive nonparametric regression. *Biometrika* **89**, 513-528.

Zhou, S. and Shen, X. (2001). Spatially adaptive regression splines and accurate knot selection schemes. *J. Amer. Statist. Assoc.* **96**, 247-259.

Department of Biostatistics and Epidemiology, University of Pennsylvania School of Medicine, Philadelphia, PA 19104-6021, USA.

E-mail: zliu5@mail.med.upenn.edu

Department of Biostatistics and Epidemiology, University of Pennsylvania School of Medicine, Philadelphia, PA 19104-6021, USA.

E-mail: wguo@mail.med.upenn.edu

(Received April 2008; accepted March 2009)