SPATIALLY-ADAPTIVE PENALTIES FOR SPLINE FITTING

David Ruppert and Raymond J. Carroll *

January 6, 1999
Revised, July 17, 1999

Abstract

We study spline fitting with a roughness penalty that adapts to spatial heterogeneity in the regression function. Our estimates are pth degree piecewise polynomials with $p-1$ continuous derivatives. A large and fixed number of knots is used and smoothing is achieved by putting a quadratic penalty on the jumps of the pth derivative at the knots. To be spatially adaptive, the logarithm of the penalty is itself a linear spline but with relatively few knots and with values at the knots chosen to minimize GCV. This locally-adaptive spline estimator is compared with other spline estimators in the literature such as cubic smoothing splines and knot-selection techniques for least-squares regression. Our estimator can be interpreted as an empirical Bayes estimate for a prior allowing spatial heterogeneity. In cases of spatially heterogeneous regression functions,

*David Ruppert is Professor, School of Operations Research & Industrial Engineering, Cornell University, Ithaca, New York 14853-3801 (E-mail: davidr@orie.cornell.edu). Ruppert's research was supported an NSF grant. R.J. Carroll is University Distinguished Professor of Statistics, Nutrition and Toxicology, Texas A&M University, College Station, TX 77843-3143 (E-mail: carroll@stat.tamu.edu). Carroll's research was supported by a grant from the National Cancer Institute (CA-57030) and by the Texas A & M Center for Environmental and Rural Health through a grant from the National Institute of Environmental Health Sciences (P30-E509106). Carroll's work was partially completed during visits to Sonderforschungsbereich 373 at the Humboldt Universität zu Berlin. Matt Wand kindly showed us his manuscript on a comparison of regression spline smoothing method. We thank George Casella, Rob Kass, and Marty Wells for references to the empirical Bayes literature. This paper has benefitted substantially from the constructive comments of two referees and an associate editor.

empirical Bayes confidence intervals using this prior achieve better pointwise coverage probabilities than confidence intervals based on a global-penalty parameter. We first develop our method for univariate models and then extend it to additive models.

Key words and phrases. Additive models, Bayesian inference, confidence intervals, hierarchical Bayesian model, regression splines.

1 Introduction

In this paper we study a variant of smoothing splines that we call penalized splines or, following Eilers and Marx (1996), p-splines. What is new is that we allow the penalty to vary spatially to adapt to possible spatial heterogeneity in the regression function. This spatial adaptivity can result in improved precision and also better confidence bounds on the regression function.

Suppose that we have data (x_i, y_i) where x_i is univariate,

$$y_i = m(x_i) + \epsilon_i, \tag{1}$$

m is a smooth function equal to the conditional mean of y_i given x_i, and the ϵ_i's are independent, mean zero errors with a constant variance, σ^2. The extension to additive models is straightforward and is mentioned in Section 8. To estimate m we use a regression spline model

$$m(x; \boldsymbol{\beta}) := \beta_0 + \beta_1 x + \cdots + \beta_p x^p + \sum_{k=1}^{K} \beta_{p+k} (x - \kappa_k)_+^p, \tag{2}$$

where $p \geq 1$ is an integer, $\boldsymbol{\beta} := (\beta_0, \ldots, \beta_p, \beta_{p+1}, \ldots, \beta_{p+K})^\mathsf{T}$ is a vector of regression coefficients, $(u)_+^p := u^p I(u \geq 0)$, and $\kappa_1 < \cdots < \kappa_K$ are fixed knots.

When fitting model (2) to noisy data, care is needed to prevent overfitting which causes a rough fit tending interpolate the data. The traditional methods of obtaining a smooth spline estimate are knot selection, e.g., Friedman and Silverman (1989), Friedman (1991), and Stone, Hansen, Kooperberg, and Truong (1997), and smoothing splines (Wahba, 1990; Eubank, 1988). With the first set of methods, the knots are selected from a set of candidate knots by a technique similar to stepwise regression and then, given the selected knots, the coefficients are estimated by ordinary least squares. Smoothing splines have a knot at each unique value of x and control overfitting by using least-squares estimation with a roughness penalty. The penalty is on the integral of the square of a specified derivative, usually the second. The penalized least-squares estimator has the form of a ridge regression estimate. Luo and Wahba (1997) proposed a hybrid between knot selection and smoothing splines—they follow knot selection by penalized least-squares estimation. Recently, there have appeared Bayesian methods that yield weighted averages of (essentially) least-squares estimates. The averages are over the sets of possible knots, with a set's weight given by the posterior probability that the set is the "true set" (Smith and Kohn, 1996; Denison, Mallick, and Smith, 1998).

In this paper we use a penalty approach similar to smoothing splines but with less knots. We allow K in (2) to be large but typically far less than n. Unlike knot-selection techniques, we retain all candidate knots. As with smoothing splines, a roughness penalty is placed on $\{\beta_{p+k}\}_{k=1}^{K}$ which is the set of jumps in the pth derivative of $m(x;\boldsymbol{\beta})$. We could view this as a penalty on the $(p+1)$th derivative of $m(x;\boldsymbol{\beta})$ where that derivative is a generalized function. Eilers and Marx (1996) developed this method of "p-splines," though they have traced the original idea to O'Sullivan (1986, 1988). Eilers and Marx use equally-spaced knots and they use the B-spline basis, whereas we use sample quantiles of x as knots and the truncated power-function basis. Also, they consider a somewhat more general class of penalties than we need here.

Because smoothness is controlled by a roughness penalty, once a certain minimum number of knots is reached, further increases in the number of knots cause little noticeable change in the fit given by a p-spline. In applications we have seen to actual data, using between 5 and 40 knots works well. In certain difficult problems used in simulation studies, we have seen the need for more than 40 knots; see Section 5. However, in the example of that section, using more than 80 knots does not improve over 80 knots, and 80 knots is still far less than the number of knots used by a smoothing spline which is the sample size of 400. We recommend letting κ_k be the $k/(K+1)$th sample quantile of the x_i's, which we call "equally-spaced sample quantiles."

We treat the number of knots as a user-specified tuning parameter. Although the choice of the number of knots is often not crucial, it is important that a certain minimum number of knots be used. Therefore, some users may want a completely automatic algorithm, and we propose such a procedure in Section 3.

We define $\widehat{\boldsymbol{\beta}}(\alpha)$ to be the minimizer of

$$\sum_{i=1}^{n}\Big\{y_i - m(x;\boldsymbol{\beta})\Big\}^2 + \sum_{k=1}^{K}\alpha(\kappa_k)\beta_{p+k}^2, \tag{3}$$

where $\alpha(\cdot)$ is a penalty function. Eilers and Marx (1996) use a constant α, that is, their α is the same for all knots, though its value will depend on the data. A constant penalty weight is also used in the smoothing spline literature. We will call a spline fit with a constant value of α a global penalty spline. Local penalty splines are those with α varying across the knots.

A single penalty weight is not suitable for functions that rapidly oscillate in some regions and are rather smooth in other regions. The inferiority, in terms of MSE, of splines having a single smoothing parameter is shown in a simulation study by Wand (1997). In that study,

p-splines are not competitive with knot-selection methods for regression spline fitting for regression functions with significant spatial inhomogeneity.

Another problem with having only a single smoothing parameter concerns inference. Smoothing splines and p-splines are both Bayes estimates for certain priors. A single smoothing parameter corresponds to a spatially homogeneous prior. For example, for a p-spline, the prior is that the $\{\beta_{p+k}\}_{k=1}^K$ are iid $N(0,\tau^2)$ where τ^2 equals σ^2/α; see Section 4. The polynomial coefficients, $\beta_0,\ldots,\beta_p$, are given an improper prior, uniform on $p+1$ dimensional space. Such priors on $\{\beta_{p+k}\}_{k=1}^K$ are not appropriate for spatially heterogeneous m. Consider confidence intervals based on the posterior variance of $m(\cdot)$ as in Wahba (1983) and Nychka (1988). As Nychka shows, the resulting confidence bands have good *average* (over x) coverage probabilities but do not have accurate pointwise coverage probabilities in areas of high oscillations of m or other "features."

Our methodology is described in Section 2. A fully automatic estimator with all tuning parameters selected by the data is presented in Section 3. Bayesian inference is discussed in Section 4 and Monte Carlo simulations are in Section 5. Section 6 contains are example using data from an experiment where atmospheric mercury is monitored by LIDAR. In Section 7 we extend our methodology to additive models, and finally Section 8 contains discussion and conclusions.

2 A Local Penalty Method

Here is a simple approach to spatially varying α. Choose another set of the knots, $\{\kappa_k^*\}_{k=1}^M$, where M is smaller than K and such that $\{\kappa_1^*=\kappa_1<\cdots<\kappa_M^*=\kappa_K\}$. The penalty at one of these "subknots" (or "α-knots"), say κ_k^*, is controlled by a parameter α_k^*. The penalties at the original knots, $\{\kappa_k\}_{k=1}^K$, are determined by linear interpolation, say, of the penalties at the $\{\kappa_k^*\}_{k=1}^M$. The interpolation is on the log-penalty scale to ensure positivity of the penalties. Thus, we have a penalty $\alpha(\kappa_k)$ at each κ_k but these penalties depend only upon $\boldsymbol{\alpha^*}:=(\alpha_1^*,\ldots,\alpha_M^*)^{\mathsf{T}}$. Therefore, $(\alpha(\kappa_1),\ldots,\alpha(\kappa_K))$ is a function of $\boldsymbol{\alpha^*}$. This function is not derived explicitly but rather is computed by using a linear interpolation algorithm; we used MATLAB's build-in linear interpolator. One could, of course, we other interpolation methods, e.g., cubic interpolation. If linear interpolation is used, then $\log(\alpha(\cdot))$ is a linear spline with knots at $\{\kappa_k^*\}_{k=1}^M$.

Let $\boldsymbol{Y}:=(y_1,\ldots,y_n)^T$ and $\boldsymbol{X}$ be the "design matrix" for the regression spline so that

the ith row of $\boldsymbol{X}$ is

$$\boldsymbol{X}_i := (\,1, \quad x_i, \quad \cdots \quad x_i^p, \quad (x_i - \kappa_1)_+^p, \quad \cdots \quad (x_i - \kappa_K)_+^p\,). \tag{4}$$

Let $\boldsymbol{D}(\boldsymbol{\alpha}^*)$ be a diagonal matrix whose first $(1+p)$ diagonal elements are 0 and whose remaining diagonal elements are $\alpha(\kappa_1), \ldots, \alpha(\kappa_K)$, which depend only on $\boldsymbol{\alpha}^*$. Then standard calculations show that $\widehat{\boldsymbol{\beta}}(\boldsymbol{\alpha}^*)$ is given by

$$\widehat{\boldsymbol{\beta}}(\boldsymbol{\alpha}^*) = \Big(\boldsymbol{X}^T\boldsymbol{X} + \boldsymbol{D}(\boldsymbol{\alpha}^*)\Big)^{-1}\boldsymbol{X}^T\boldsymbol{Y}. \tag{5}$$

This is a ridge regression estimator that shrinks the regression spline towards the least-squares fit to a pth degree polynomial model (Hastie and Tibshirani, 1990, Section 9.3.6).

The smoothing parameter $\boldsymbol{\alpha}^* = (\alpha_1^*, \ldots, \alpha_M^*)$ can be determined by minimizing

$$\mathrm{GCV}(\boldsymbol{\alpha}^*) = \frac{\|\boldsymbol{Y} - \boldsymbol{X}\ \widehat{\boldsymbol{\beta}}(\boldsymbol{\alpha}^*)\|^2}{(1 - \mathrm{df}(\boldsymbol{\alpha}^*)/n)^2}.$$

Here

$$\mathrm{df}(\boldsymbol{\alpha}^*) = \mathrm{tr}\Big\{\Big(\boldsymbol{X}^\mathsf{T}\boldsymbol{X} + \boldsymbol{D}(\boldsymbol{\alpha}^*)\Big)^{-1}\Big(\boldsymbol{X}^\mathsf{T}\boldsymbol{X}\Big)\Big\} \tag{6}$$

is the degrees of freedom of the smoother which is defined to be the trace of the smoother matrix (Hastie and Tibshirani, 1990, Section 3.5). The right-hand side of (6) is suitable for computing since it is the trace of a matrix whose dimension is only $(1+p+K)^2$.

A search over an M-dimensional grid is not recommended because of computation cost. Rather, we recommend that one start with $\alpha_1^*, \ldots, \alpha_M^*$ each equal to the best global value of α chosen by minimizing GCV. Then each α_k^* is varied, with the others fixed, over a one-dimensional grid centered at the current value of α_k^*. On each such step, α_k^* is replaced by the α-value minimizing GCV on this grid. This minimimizing of GCV over each α_k^* is repeated a total of N_{iter} times. Although minimizing GCV over the α_k^*'s one at a time in this manner does not guarantee finding the global minimum of GCV over $\alpha_1^*, \ldots, \alpha_M^*$, our simulations show that this procedure is effective in selecting a satisfactory amount of local smoothing. The minimum GCV global α is a reasonably good starting value for the smoothing parameters and each step of our algorithm improves about this start in the sense of lowering GCV. Since each α_k^* control the penalty only over a small range of x, the optimal value of one α_k^* should depend only slightly upon the other α_k^*. We believe this is the reason that our one-at-a-time search strategy works effectively.

3 A Completely Automatic Algorithm

The local penalty method has three tuning parameters, the number of knots K, the number of subknots M, and the number of iterations N_{iter}. The exact values of the tuning parameters are not crucial provided they are within certain acceptable ranges—the crucial parameter is $\boldsymbol{\alpha}^*$ which is selected by GCV. However, users may want a completely automatic algorithm which requires no user-specified parameters and attempts to ensure that the tuning parameters are within acceptable ranges. An automatic algorithm would need to balance the need for the tuning parameters to be large enough to obtain a good fit with the need that the tuning parameters not be so large that the computation time is excessive. (Overfitting is not a concern because it is controlled by $\boldsymbol{\alpha}^*$.)

In this section, we propose such a procedure based on the following principle: as the complexity of m increases each of K, M, and N_{iter} should increase. The algorithm uses a sequence of values of (K, M, N_{iter}) where each parameter is non-decreasing in the sequence. The algorithm stops when there is no appreciable decrease in GCV between two successive values of (K, M, N_{iter}). Monte Carlo experimentation discussed in Section 5.2 shows that the values of N_{iter} and M have relatively little effect on the fit, at least within the ranges studied. However, it seems reasonable to increase N_{iter} and M slightly with K. On the other hand, for a given K computation time is roughly proportional to $M \times N_{iter}$, so we avoid $N_{iter} > 2$ and $M > 6$.

Specifically, the sequence of values of (K, M, N_{iter}) that we use are (10,2,1), (20,3,2), (40,4,2), (80,6,2), (120,6,2). We compare GCV, minimized over $\boldsymbol{\alpha}^*$, using (10,2,1) and (20,3,2). If the value of GCV for (20,3,2) is more than a constant C times the GCV value of (10,2,1) then we conclude that further increases in the tuning parameters will not appreciably decrease GCV. In the simulations we used $C = .98$ and that choice worked well. Therefore, we stop and use (20,3,2) as the final value of the three tuning parameters. Otherwise, we fit using (40,4,2) and compare its GCV value to that of (20,3,2). If the value of GCV for (40,4,2) is more than C times the GCV value of (20.3.2) then we stop and use (40,4,2) as the final value of the three tuning parameters. Otherwise, we continue in this manner, comparing (40,4,2) to (80,6,2), etc. If very complex m were contemplated, then one could, of course, continue using increasing larger values of the tuning parameters.

Note that the final tuning parameter vector is selected from (20,3,2), (40,4,2), (80,6,2), and (120,6,2). The vector (10,2,1) is used only to check if one can stop at (20,3,2).

4 Bayesian Inference

The p-spline method has an interpretation as a Bayesian estimator in a linear model. See Box and Tiao (1973) and Lindley and Smith (1972) for a discussion of Bayesian linear models. Suppose that $\epsilon_1, \ldots, \epsilon_n$ are iid $N(0, \sigma^2)$ and that the prior on $\boldsymbol{\beta}$ is $\mathbf{N}\{0, \boldsymbol{\Sigma}(\boldsymbol{\alpha}^*)\}$, where $\boldsymbol{\Sigma}(\boldsymbol{\alpha}^*)$ is a covariance matrix depending on $\boldsymbol{\alpha}^*$. Here $\mathbf{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the multivariate normal distribution with mean and covariance matrix $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$. For now, assume that σ^2 and $\boldsymbol{\alpha}^*$ are known. Then the posterior log density of $\boldsymbol{\beta}$ given $\boldsymbol{Y}$ is, up to an additive function of $\boldsymbol{Y}$ and $(\sigma^2, \boldsymbol{\alpha}^*)$, given by

$$-\frac{1}{2}\left\{\frac{1}{\sigma^2}\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \boldsymbol{\beta}^{\mathsf{T}}\boldsymbol{\Sigma}(\boldsymbol{\alpha}^*)^{-1}\boldsymbol{\beta}\right\}. \tag{7}$$

The maximum a posteriori (MAP) estimator of $\boldsymbol{\beta}$, i.e., the mode of the posterior density, maximizes (7). Now let $\beta_0, \ldots, \beta_p$ have improper unif$(-\infty, \infty)$ priors and let $\{\beta_{p+k}\}_{k=1}^K$ be independent with β_{p+k} having a $N(0, \sigma^2/\alpha_k)$ distribution. Then

$$\boldsymbol{\Sigma}^{-1}(\boldsymbol{\alpha}^*) = \sigma^2 \mathrm{diag}(0, \ldots, 0, \alpha_1, \ldots, \alpha_K) \tag{8}$$

and the MAP estimator minimizes (3). (More precisely, we let $\beta_0, \ldots, \beta_p$ have a $N(0, \sigma_1^2)$ prior and then (8) holds in the limit as $\sigma_1 \to \infty$.)

Of course, the α_k will not be known in practice. Empirical Bayes methods replace unknown "hyperparameters" in a prior by estimates and then treat these hyperparameters as fixed. For example, if $\{\alpha_k^*\}_{k=1}^M$ are estimated by GCV and then considered fixed, one is using empirical Bayes inference. Standard calculations show that when $\boldsymbol{\alpha}^*$ and σ^2 are known, then the posterior distribution of $\boldsymbol{\beta}$ is

$$\mathbf{N}\Big[\widehat{\boldsymbol{\beta}}(\boldsymbol{\alpha}^*),\, \sigma^2\{\boldsymbol{X}^{\mathsf{T}}\boldsymbol{X} + \boldsymbol{\Sigma}(\boldsymbol{\alpha}^*)\}^{-1}\Big]. \tag{9}$$

Also, the posterior distribution of $\boldsymbol{m} := \{m(x_1), \ldots, m(x_n)\}^{\mathsf{T}}$ is

$$\mathbf{N}\Big[\boldsymbol{X}\widehat{\boldsymbol{\beta}}(\boldsymbol{\alpha}^*),\, \sigma^2\boldsymbol{X}\{\boldsymbol{X}^{\mathsf{T}}\boldsymbol{X} + \boldsymbol{\Sigma}(\boldsymbol{\alpha}^*)\}^{-1}\boldsymbol{X}^{\mathsf{T}}\Big]. \tag{10}$$

An approximate Bayes posterior replaces $\boldsymbol{\alpha}^*$ and σ^2 in (9) and (10) by estimates. Assuming that $\boldsymbol{\alpha}^*$ has been estimated by GCV, one need only estimate σ^2 by $\|\boldsymbol{Y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}}(\widehat{\boldsymbol{\alpha}}^*)\|^2/\{n - \mathrm{df}(\widehat{\boldsymbol{\alpha}}^*)\}$ where $\mathrm{df}(\boldsymbol{\alpha}^*)$ is defined by (6). This gives the approximate posterior distribution for $\boldsymbol{m}$

$$\mathbf{N}\Big[\boldsymbol{X}\widehat{\boldsymbol{\beta}}(\widehat{\boldsymbol{\alpha}}^*),\, \widehat{\sigma}^2\boldsymbol{X}\{\boldsymbol{X}^{\mathsf{T}}\boldsymbol{X} + \boldsymbol{\Sigma}(\widehat{\boldsymbol{\alpha}}^*)\}^{-1}\boldsymbol{X}^{\mathsf{T}}\Big]. \tag{11}$$

The approximate Bayes $100(1-\gamma)\%$ confidence interval for $m(x_i)$ is

$$\widehat{m}(x_i) \pm \Phi^{-1}(1-\gamma/2)\,\mathrm{std}(\widehat{m}(x_i))$$

where $\widehat{m}(x_i) = \boldsymbol{X}_i\widehat{\beta}(\widehat{\alpha})$ is the ith element of the posterior mean in (11), $\mathrm{std}\{\widehat{m}(x_i)\}$ is the square root of the ith diagonal of the posterior covariance matrix in (11), and Φ is the standard normal CDF.

Hastie and Tibshirani (1990) make an interesting and cogent argument against using confidence bands about the regression line, but instead plotting a sample of curves from the posterior distribution. By following their recommendation, one gets a much better sense of what the true regression curve might look like. Regardless of whether one samples from the posterior or looks at confidence intervals (or does both!), a posterior that reflects any spatial heterogeneity that may exist will give a more accurate picture of the true function.

Since they estimate hyperparameters but then pretend that the hyperparameters were known, these approximate Bayesian methods do not account for extra variability in the posterior distribution caused by the estimation of hyperparameters in the prior; for discussion see, for example, Morris (1983), Laird and Louis (1987), Kass and Steffey (1989), or Carlin and Louis (1996). Everything else held constant, the under-estimation of posterior variance should become worse as M increase, since each α_m^* will be determined by less data and will therefore be more variable. As Nychka (1988) has shown empirically, this under-estimation does not appear to be a problem for a global penalty which has only one hyperparameter. However, the local penalty has M hyperparameters. We have found for local-penalty splines that the pointwise approximate posterior variance of $\widehat{m}$ is too small in the sense that it noticeably underestimates the frequentist's MSE.

A simple correction to this problem is to multiply the pointwise posterior variances of the local-penalty $\widehat{m}$ from (11) by a constant so that the average pointwise posterior variance of $\widehat{m}$ is the same for the global and local penalty estimators. The reasoning behind this correction is as follows. As stated above, the global penalty approximate posterior variance from (11) is nearly equal to the frequentist's MSE on average. The local penalty estimate has an MSE that varies spatially but should be close, on average, to the MSE of the global penalty estimate and therefore also close, on average, to the estimated posterior variance of the global penalty estimator. We found that this adjustment is effective in guaranteeing coverage probabilities at least as large as nominal, though in extreme cases of spatial heterogeneity the adjustment can be conservative; see Section 5.4. The reason for the later is that in cases of severe spatial heterogeneity, the local penalty MSE will be less, on average, than that of

the global penalty estimate. Then, there will be an over-correction and the local penalty MSE will be over-estimated by this adjusted posterior variance. The result is that confidence intervals constructed with this adjustment should be conservative. The empirical evidence in Section 5 supports this conjecture. It that section, we refer to these adjusted intervals as local-penalty, conservative.

Another correction would be to use a fully Bayesian hierarchical model, where the hyperparameters are given a prior. Deely and Lindley (1981) first considered such Bayesian empirical Bayes methods. An exact Bayesian analysis for p-splines would seem to require Gibbs sampling or other computationally intensive techniques. Given the number of parameters involved and the model complexity, an accurate MCMC analysis could takes days or weeks of computer time. In contrast, our algorithm with the adjustment above can be computed in a matter of seconds.

There are intermediate positions between the quick, ad hoc conservative adjustment just proposed and an exact, fully Bayesian analysis. One that we now describe is an approximate fully Bayesian method that uses a small bootstrap experiment and a delta-method correction adopted from Kass and Steffey's (1989) "first order approximation." (Kass and Steffey considered conditionally independent hierarchical model, which are also called empirical Bayes models, but their ideas apply directly to more general hierarchical Bayes models.)

Here is how the Kass and Steffey approximation applied to p-splines. Let $m_i = m(x_i) = \boldsymbol{X}_i\boldsymbol{\beta}$. The posterior variance of m_i is calculated from the joint posterior distribution of $(\boldsymbol{\beta}, \boldsymbol{\alpha}^*)$ and by a standard identity is

$$\text{var}(m_i) = E\{\text{var}(m_i|\boldsymbol{\alpha}^*)\} + \text{var}\{E(m_i|\boldsymbol{\alpha}^*)\}.$$

$E\{\text{var}(m_i|\boldsymbol{\alpha}^*)\}$ is well-approximated by the posterior variance of m_i when $\boldsymbol{\alpha}^*$ is treated as known and fixed at its posterior mode (Kass and Steffey; 1989). Thus, $\text{var}\{E(m_i|\boldsymbol{\alpha}^*)\}$ is the extra variability in posterior distribution of m_i that the approximate posterior variance given by (11) does not account for. We estimate $\text{var}\{E(m_i|\boldsymbol{\alpha}^*)\}$ by the following steps and add this estimate to the posterior variance given by (11). The three steps are:

1. Use a parametric bootstrap to estimate $\text{var}(\log(\widehat{\boldsymbol{\alpha}}^*))$. (Here the log function is applied coordinate-wise to the vector $\boldsymbol{\alpha}^*$.)

2. Numerically differentiate $\boldsymbol{X}\widehat{\boldsymbol{\beta}}(\boldsymbol{\alpha}^*)$ with respect to $\log(\boldsymbol{\alpha}^*)$ at $\boldsymbol{\alpha}^* = \widehat{\boldsymbol{\alpha}}^*$. We use one-sided numerical derivatives with a step-length of 0.1.

3. Put the results from 1. and 2. into the delta-method formula:

$$\mathrm{var}\{E(m_i|\boldsymbol{\alpha}^*)\} \approx \left(\frac{\partial \boldsymbol{X}\widehat{\boldsymbol{\beta}}(\widehat{\boldsymbol{\alpha}}^*)}{\partial \boldsymbol{\alpha}^*}\right)^{\mathsf{T}} \mathrm{var}(\widehat{\boldsymbol{\alpha}}^*) \left(\frac{\partial \boldsymbol{X}\widehat{\boldsymbol{\beta}}(\widehat{\boldsymbol{\alpha}}^*)}{\partial \boldsymbol{\alpha}^*}\right) \tag{12}$$

When (12) is added to the approximate posterior variance from (11), we call the corresponding confidence intervals "local-penalty, corrected." Since the correction, (12), is a relatively small portion of the corrected posterior variance, it need not be estimated by the bootstrap with as great a precision as when a variance is estimated entirely by a bootstrap. It our simulations, we used only 25 bootstrap samples in step 1.

In the simulations of the next section, the local-penalty, conservative intervals are close to the more computationally intensive local-penalty, corrected intervals. Since the latter have a theoretical justification, this closeness is some justification for the former.

5 Simulations

5.1 Mean squared error comparison

We performed a small Monte Carlo experiment using the "spatial variability" scenario in Wand (1997) so that our results could be compared with his. The x's were equally spaced on [0,1], n was 400, and the ϵ_i's were independent $N(0, (0.2)^2)$. The regression function, whose spatial variability was controlled by a parameter j, was

$$m(x; j) = \sqrt{x(1-x)}\sin\left\{\frac{2\pi(1+2^{(9-4j)/5})}{x+2^{(9-4j)/5}}\right\}. \tag{13}$$

We used both $j = 3$ which gave low spatial variability and $j = 6$ which gives severe spatial variability; see panels (a) and (b) of Figure 1. We used both 40 and 80 knots. When we used 40 knots, then $\{\kappa_{k(j)} : j = 1, 10, 20, 30, 40\}$ were the subknots used for the local penalty. For 80 knots, $\{\kappa_{k(j)} : j = 1, 20, 40, 60, 80\}$ were the subknots. In all cases, N_{iter} was 1 and quadratic splines were used. For each of the four combinations of j and K, we simulated 250 data sets and applied the global and local penalty function estimators to each. Boxplots of

$$\log_{10}(\mathrm{RMSE}) = \log_{10}\left(\sqrt{n^{-1}\sum_{i=1}^{n}(\hat{m}(x_i) - m(x_i))^2}\right).$$

are shown in Figure 1.

From the results in Figure 1 we may draw the following conclusions:

- Locally varying penalties are as effective as a global penalty when there is little spatial variability. There appears to be little or no cost in statistical efficiency when a local penalty is used but not needed.

- For severe spatial variability, the local penalty approach is far superior to a global penalty.

- There is little difference between using 40 and 80 knots, except for one important situation. If one uses a local penalty and $j = 6$, then 80 knots is significantly better than 40. The reason is that 80 knots allows the spline to track the rapid oscillations on the left, but only if a local penalty is used.

Also, comparing the results in Figure 1 to the results in Wand (1997) for $j = 6$, the local-penalty approach is somewhat better than the Bayesian method of Smith and Kohn (1996) and the stepwise selection method of Stone, Hansen, Kooperberg, and Truong (1997). However, Wand's simulations used code provide by Smith that had 35 knots "hard-wired" into it (Wand, personal communication). With more knots, the Smith and Kohn method could very well be competitive with the local-penalty method.

We have also looked at moderate spatial variability ($j = 4$ or 5). There the local-penalty estimator is better than the global-penalty estimator, and again the local-penalty estimator is as good as the Bayesian and stepwise methods studied by Wand.

To compare the local and global-penalty splines with other smoothers besides those in Wand's (1997) study, we used one of the sampling situations, Case 6, in Luo and Wahba (1997). The regression function there is

$$m(x) = \sin\{2(4x - 2)\} + 2\exp\{-16^2(x - .5)^2\}.$$

We used the same values of σ^2 and n as Luo and Wahba ($n = 256$ and $\sigma = 0.3$) and used equally spaced x's on [0,1] as they did. Our results for local and global-penalty splines and those of Luo and Wahba for their hybrid adaptive spline (HAS), smoothing splines (SS), SureShrink of Johnstone and Donoho, and MARS of Friedman (1991) are given in Table 1

Denison, Mallick, and Smith (1998) tested their Bayesian splines on the same example as in Luo and Wahba (1997), but they used $n = 200$ instead of 256 and reported MSE values instead of medians of squared errors. They found MSE values of 0.0096 and 0.0087 for linear and quadratic splines, while for the same sampling situation we found MSE values of 0.0075 and 0.0083 for the local and global penalty quadratic splines.

Table 1: Median of squared errors (Interquartile range of squared errors) for six smoothers. The results for HAS, SS, SureShrink, and MARS are from Luo and Wahba (1997).

HAS	SS	SureShrink	MARS	Local PS	Global PS
.007	.006	.018	.007	.0053	.0061
(.006)	(.003)	(.004)	(.004)	(.0035)	(.0029)

We also tried cubic p-splines with both global and local penalties, but we found cubic p-splines somewhat inferior to quadratic p-splines.

5.2 Effects of the tuning parameters

We conducted a Monte Carlo experiment to learn further how the tuning parameters affect the accuracy of the local p-spline. The regression function (13) was used with j varying as a factor with levels 3, 4, 5, and 6. The sample size, values of x, and σ were the same as in Section 5.1. There were three other factors: K with levels 20, 40, 80, and 120; M with levels 3, 4, 6, and 8; and N_{iter} with levels 1, 2, and 3. A full factorial design was used with two replications for a total of 384 runs.

The response was log(MSE). First, a quadratic response surface with two-way interactions was fit to all four factors. Then, to look at the data from a slightly different perspective, quadratic response surfaces in the three tuning parameters were fit with j fixed at each of its four levels. This second perspective was more illuminating. We found that for $j = 3$, 4, or 5, the tuning parameters had no appreciable effects on log(MSE). For $j = 6$, only the number of knots, K, had an effect on log(MSE). That effect is nonlinear—log(MSE) decreases rapidly as K increase up to about 80 but then log(MSE) levels off.

In summary, for the scenario we simulated, of three tuning parameters only K has a detectable effect on log(MSE). It is important the K be at least a certain minimum value depending on the regression function, but after K is sufficiently large further increases in K do not affect accuracy.

5.3 The automatic algorithm

We tested the algorithm in Section 3 that chooses all tuning parameters automatically. As just mentioned, it is important that the number of knots, K, be sufficiently large that all significant features of the regression function can be modeled. Thus, the main function of

the automatic algorithm is to ensure that K is sufficiently large. As reported in Section 5.2 the number of subknots and the number of iterations were not noticed to affect accuracy, but in our proposed algorithm we allowed them to increases slightly with K.

For each of $j = 3$ and 6 we used the algorithm on 250 data sets, with $n = 400$ and the standard deviation of the ϵ's equal to 0.2 as before. Recall that the algorithm can choose as the final value of (K, M, N_{iter}) one of the vectors (20,3,2), (40,4,2), (80,6,2), and (120,6,2). With $j = 3$, the first vector was chosen 249 times and the second vector once. The tuning parameter vector (20,3,2) gives MSE values quite similar to larger tuning parameter values, so stopping at (20,3,2) is clearly appropriate. With $j = 6$ the fourth tuning parameter vector was chosen 247 times, while the third vector was chosen the remaining 3 times. As we saw in the last section, 80 knots is preferable here to a lesser number of knots. However, using 120 knots offers no improvement over 80 (but is no worse either). Therefore, selection of either of the two largest possible values of the tuning parameters, which happened in all 250 trials, is the appropriate choice in this situation.

We conclude that the automatic algorithm can supply reasonable values of the tuning parameters when the user has little idea how to choose them. The automatic algorithm is, of course, slower than using a fixed, user-specificied tuning parameter vector since the automatic algorithm can require up to five fits. This slowness is not a serious problem when fitting a few data sets, but does slow down Monte Carlo simulations. Therefore, for the remainder of the study we use fixed values of (K, M, N_{iter}).

5.4 Bayesian inference

To compare posterior distribution with and without a local penalty, we used a spatially heterogeneous regression function

$$m(x) = \exp\{-400(x - .6)^2\} + \frac{5}{3}\exp\{-500(x - .75)^2\} + 2\exp\{-500(x - .9)^2\}. \qquad (14)$$

The x_i were equally spaced on $[0, 1]$, the sample size was $n = 300$, and the ϵ_i were normally distributed with $\sigma = .5$. We used quadratic splines with $K = 40$ knots, the number of subknots was $M = 4$, and the number of iterations to minimize GCV using the local penalty was $N_{iter} = 1$.

Figure 2 shows a typical data set and the global and local penalty estimates. The global penalty estimate has a small penalty chosen by GCV to accommodate the oscillations on the right, but the unfortunate side effect is undersmoothing on the left. The local penalty

removes this problem. Figure 3 shows the pointwise MSE and squared bias of the global-penalty estimator calculated from 500 Monte Carlo samples. Also shown is the pointwise posterior variance given by (11) averaged over the 500 repetitions. The posterior variance should be estimating the MSE. We see that the posterior variance is constant, except for boundary effects, and cannot detect the spatial heterogeneity in the MSE. Figure 4 is a similar figure for the local-penalty estimator. Two posterior variances are shown, the conservative adjustment and the Kass/Steffey type correction. One can see that the MSE is somewhat different than in Figure 3 since the estimator adopts to spatial heterogeneity. Also, the posterior variance tracks the MSE better than for the global-penalty estimator and the corrected version of the posterior variance tends to be a little closer to the MSE than the adjusted version.

In Figure 5 we present the Monte Carlo estimates of the pointwise coverage probabilities of nominal 95% Bayesian confidence intervals based on the global and local-penalty estimators. These coverage probabilities have been smoothed by p-splines to remove some of the Monte Carlo variability. All three confidence interval procedures achieve pointwise coverage probabilities close to 95%. Because the local penalty methods are somewhat conservative, the global penalty method is, on average, the closest to 95%, but the local penalty methods avoid low coverage probabilities around features in m.

6 An Example

The LIDAR (LIght Detection And Ranging) uses the reflection of laser-emitted light to detect chemical compounds in the atmosphere; see Sigrist (1994).

A typical LIDAR data set, shown in Figure 7, was taken from Holst et. al (1995). The horizontal variable, range, is the distance traveled before the light is reflect back to its source. The vertical variable, logratio, is the logarithm of the ratio of received signals at frequencies on and off the resonance frequency of the chemical species of interest, which is mercury in this example.

An interesting feature of this example is that there is scientific interest in the first derivative, m', as well as m itself, because $-m'(x)$ is proportional to concentration at range x; see Ruppert et al. (1997) for further discussion. For the estimation of m a global penalty works satisfactorily. Figure 7 shows the global penalty fit. The local penalty fit was not included in that figure, since it would be difficult to distinguish from the global penalty fit.

However, for the estimation of m', a local penalty appears to improve upon a global

penalty. Figure 8 shows the derivatives (times -1) of fitted splines and their confidence intervals using global and local penalties. Notice that the confidence intervals using the local penalty are generally narrower than for the global penalty, except at the peak where the extra width should be reflecting real uncertainty. The local penalty estimate has a sharper peak and less noise in the flat areas.

A referee has made the valid point that choosing $\boldsymbol{\alpha}^*$ to estimate m and then using this $\boldsymbol{\alpha}^*$ to estimate m' is a common, but questionable, practice. It is our intention to investigate methods that choose $\boldsymbol{\alpha}^*$ to estimate m', but we haven't done this yet. GCV, though it targets m, does seem effective in choosing the right amount of smoothing for estimating m' in this example.

7 Additive Models

7.1 An algorithm for additive models

Until now, we have confined our attention to univariate splines, but our ideas can be easily extended to additive models. Suppose we have L predictor variables and that $\boldsymbol{x}_i = (x_{i,1}, \ldots, x_{i,L})^{\mathsf{T}}$ is the vector of predictor variables for the ith case. The additive model is

$$y_i = \beta_0 + \sum_{l=1}^{L} m_l(x_{i,l}) + \epsilon_i.$$

Let the lth predictor variable have K_l knots, $\kappa_{1,l} \ldots, \kappa_{K_l,l}$. Then the additive spline model is

$$m(\boldsymbol{x}, \boldsymbol{\beta}) = \beta_0 + \sum_{l=1}^{L} \Big\{ \beta_{1,l} x_l + \cdots + \beta_{p,l} x_l^p + \sum_{k=1}^{K_l} \beta_{p+k,l} (x_l - \kappa_{k,l})_+^p \Big\}$$

The parameter vector is $\boldsymbol{\beta} = (\beta_0, \beta_{1,1}, \ldots, \beta_{p+K_l,1}, \ldots, \beta_{p+K_L,L})^{\mathsf{T}}$. Let $\alpha_l(\cdot)$ be the penalty function for the lth predictor. Then the penalized criterion to minimize is

$$\sum_{i=1}^{n} \Big\{ y_i - m(\boldsymbol{x}_i; \boldsymbol{\beta}) \Big\}^2 + \sum_{l=1}^{L} \alpha_l(\kappa_{k,l}) \beta_{p+k,l}^2.$$

As discussed in Marx and Eilers (1998), one need not use backfitting to fit a additive spline model. Rather, all L components can be estimated simultaneously.

Consider three levels of complexity of the penalty:

1. $\alpha_l(\cdot) \equiv \alpha$ (a common global penalty)

2. $\alpha_l(\cdot) \equiv \alpha_l$ (separate global penalties)

3. $\alpha_L(\cdot)$ is a linear spline (separate local penalties)

The following algorithm allows one to fit separate local penalties using only one-dimensional grid searches for minimizing GCV. First one minimizes GCV using a common global penalty. For this penalty to be reasonable, one must standardize the predictors so that they have common standard deviations, say. Then using the common global penalty as a starting value, one minimizes GCV over separate global penalties. The L penalty parameters are varied one-at-a-time during minimization, with the rationale that the optimal value of α_l depends only slightly on the $\alpha_{l'}, l' \neq l$. Finally, using separate global penalties as starting values, one minimizes GCV over separate local penalties. The lth local penalty has M_l parameters so there are a total of $M_1 + \cdots + M_L$ penalty parameters. These are varied in succession to minimize GCV.

7.2 Simulations of an additive model

To evaluate the practicality of this algorithm we used a variation of the simulation example is Section 5.4 where we added two spatially homogeneous component functions to the spatially hetergeneous function (14). Thus, there were three predictor variables, which for each case were independently distributed as uniform(0,1) random variables. The components of m were $m_1(x_1) = \sin(4\pi x_1)$, $m_2(x_2) = x_2^3$, and $m_3(x_3)$ was the same as $m(x)$ in (14). As in Section 5.4, $n = 300$ and the ϵ's were iid $N(0, 0.25)$. We used quadratic splines and 10, 10, and 40 knots for m_1, m_2, and m_3, respectively. The local penalty estimate had 4 subknots for all four functions.

First consider computation time. For a single data set and using our MATLAB program on a SUN Ultra 1, the common global penalty estimate took 2.1 seconds to compute, the separate global penalty estimate took an additional 1.5 second to compute, and then separate local penalties estimate took an addition 10.4 seconds to compute. Thus, local penalties are more computationally intensive than global penalties, but still feasible for small L. Now consider larger values of L. Everything else held constant, the number of parameters of an additive model grows linearly in L and, since matrix inversion time is cubic in dimension, the time for a single fit should grow cubically in L. Since the number of fits needed for the sequential grid searching described above will grow linearly in L, the total computation time for local penalties should be roughly proportional to L^4. To test this rough calculation empirically, we found the computation time for fitting additive models with 300 data point, 10 knots per variable, and 4 subknots per variable. L took 5 values from 1 to 15. Figure 9

is a log-log plot of computation time versus L. A linear fit on the log-scale is also shown; its slope is 2.45, not 4 as the quartic model predicts. The actual data show log-times that are nonlinear in $\log(L)$ with an increasing slope. Thus, a quartic model of time as a function of L may work for large values of L, but a quadratic or cubic model would be better for L in the "usual" range of 1 to 15. A likely reason that the quartic model doesn't fit well for smaller L is that the quartic model ignores parts of the computation that are linear, quadratic, and cubic in L. The computation time for 8 variables is about 1.5 minutes, but for 15 variables it is about 10.5 minutes. It seems clear that local additive fitting is feasible up to at least 8–10 variables and maybe 15 variables, but is only "interactive" up to 3 variables.

An important point to keep in mind is that computation times are largely independent of the sample size n. The reason for this is that once $\boldsymbol{X}^\mathsf{T}\boldsymbol{X}$ and $\boldsymbol{X}^\mathsf{T}\boldsymbol{Y}$ have been computed, all computation times are independent of n and the computation of $\boldsymbol{X}^\mathsf{T}\boldsymbol{X}$ and $\boldsymbol{X}^\mathsf{T}\boldsymbol{Y}$ is quite fast unless n is enormous.

Now consider statistical efficiency. The MSEs computed over 500 Monte Carlo samples for the separate local penalties estimator were 0.010, 0.0046, and 0.0165 for m_1, m_2, and m_3, respectively. Thus, m_2 is relatively easy to estimate and m_3 is slightly more difficult to estimate than m_1. The ratio of the MSE for common global penalties to separate local penalties were 1.26, 2.36, and 1.23 for m_1, m_2, and m_3, respectively. The ratio of the MSE for separate global penalties to separate local penalties were 0.85, 0.88, 1.20 for m_1, m_2, and m_3, respectively. Thus, for all three component functions, the common local penalty estimator with a single smoothing parameter is less efficient than the fully-adaptive estimator with separate local penalties. For the spatially homogeneous functions, m_1 and m_2, there is some loss of efficiency when using local penalties rather than separate global penalties, but the spatially heterogeneous m_3 is best estimated by a local penalty. These findings are somewhat different than what we found for univariate regression where no efficiency loss was noticed when a local penalty was used where a global penalty would have been adequate. There may be practical situations where one knows that a certain component function is spatially heterogeneous but the other component functions are not. Then greater efficiency should be achievable by using global penalties for the spatially homogeneous component functions and local penalties for the spatially heterogeneous ones.

The results in this section provide evidence that sequential one-dimensional grid searches to find the smoothing parameter vector are effective. The reason for this is that the optimal value of one tuning parameter depends only weakly upon the other tuning parameters. The result is that searches over a rather large number of tuning parameter (up to 60 when L is

15 and there are four subknots per variable) do appear feasible.

A study of Bayesian inference for additive models is beyond the scope of the present paper.

8 Summary and Conclusions

Spatial adaptivity is important for improved precision of point estimators and improved accuracy of confidence intervals.

The local-penalty spline is effective in increasing efficiency as measured by MSE when the regression function is spatially heterogeneous in complexity. The local-penalty method of Bayesian inference has good coverage probability throughout the range of the predictor variable, though it is somewhat conservative with coverage probabilities typically a bit higher than nominal. This conservativeness may be due to the ad hoc "adjustment" we make for estimation of multiple smoothing parameters. The adjustment is to multiply the pointwise posterior variance of the local-penalty $\widehat{m}$ by a constant so that its average posterior variance is same as the global-penalty spline. We also considered a more theoretically justified "correction" based on the work of Kass and Steffey (1989). This correction is slighly less conservative than the ad hoc adjustment and would be recommended over the adjustment except that the correction increases computation cost considerably because one step involves a small bootstrap.

For the test cases we have studied that have a moderate spatial heteregeneity, local-penalty splines with knots at equally-spaced quantiles of x perform as well as, and perhaps a bit better than, estimators using sequential knot selection.

In practice, reasonable values of the tuning parameters (K, M, N_{iter}) can often be specified by the user. However, an automatic algorithm that selects these tuning parameters by GCV has proved effective.

When a global penalty is appropriate, there seems to be little or no loss of efficiency in using local penalties, at least in the univariate case. For additive models, there can be some loss of efficiency when using a local penalty where a global penalty is appropriate.

REFERENCES

Box, G. E. P., and Tiao, G. C. (1973), *Bayesian Inference in Statistical Analysis*, Reading, MA: Addison-Wesley.

Carlin, B.P., and Louis, T.A. (1996), *Bayes and Empirical Bayes Methods for Data Analysis*, London: Chapman and Hall.

Deely, J.J., and Lindley, D.V., (1981), "Bayes Empirical Bayes," *Journal of the American Statistical Association*, 76, 833–841.

Denison, D.G.T., Mallick, B.K., and Smith, A.F.M. (1998), "Automatic Bayesian curve fitting," *Journal of the Royal Statistical Society, Series B*, 60, 333-350.

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

Eubank, R. L. (1988), *Spline Smoothing and Nonparametric Regression*, New York and Basil: Marcel Dekker.

Friedman, J.H. (1991), "Multivariate adaptive regression splines (with discussion)," *The Annals of Statistics*, 19, 1–141.

Friedman, J.H., and Silverman, B.W. (1989), "Flexible parsimonious smoothing and additive modeling (with discussion)," *Technometric*, 31, 3–39.

Green, P. J. (1987), "Penalized likelihood for general semi-parametric regression models," *International Statistical Review*, 55, 245–259.

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

Holst, U., Hössjer, O., Björklund, C., Ragnarson, P., and Edner, H. (1996), "Locally weighted least squares kernel regression and statistical evaluation of LIDAR measurements," *Environmetrics*, 7, 401–416.

Kass, R.E., and Steffey, Duane (1989), "Approximate Bayesian inference in conditionally independent hierarchical models (parametric empirical Bayes models)," *Journal of the American Statistical Association*, 84, 717–726.

Laird, N.M., and Louis, T.A., (1987), "Empirical Bayes confidence intervals based on bootstrap samples (with discussion)," *Journal of the American Statistical Association*, 82, 739–757.

Lindley, D. V., and Smith, A. F. M. (1972), "Bayes estimates for the linear model (with discussion)," *Journal Royal Statistical Society, Series B*, 34, 1–41.

Luo, Z., and Wahba, G. (1997). "Hybrid adaptive splines," *Journal of the American Statistical Association*, 92, 107–116.

Marx B.D., and Eilers P.H.C. (1998), "Direct generalized additive modeling with penalized likelihood," *Computational Statistics and Data Analysis* 28, 193–209.

Morris, C.N. (1983). "Parametric empirical Bayes inference: theory and applications (with discussion)," *Journal of the American Statistical Association*, 78, 47–65.

Nychka, D. (1988), "Bayesian confidence intervals for smoothing splines," *Journal American Statistical Association*, 83, 1134–1143.

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

O'Sullivan, F. (1988), "Fast computation of fully automated log-density and log-hazard estimators," *SIAM Journal of Scientific and Statistical Computation*, 9, 363–379.

Ruppert, D., Wand, M.P., Holst, U., and Hössjer, O. (1997). Local polynomial variance function estimation. *Technometrics*, 39, 262–273.

Sigrist, M., (1994). *Air monitoring by spectroscopic techniques (Chemical Analysis Series, Vol. 127),* Wiley.

Smith, M., and Kohn, R. (1996), "Nonparametric regression using Bayesian variable selection," *Journal of Econometrics*, 75, 317–344.

Stone, C.J., Hansen, M., Kooperberg, C., and Truong, Y. K. (1997). "Polynomial splines and their tensor products in extended linear modeling (with discussion)," *The Annals of Statistics*, 25, 1371–1470.

Wahba, G. (1983), "Bayesian 'confidence intervals' for the cross-validated smoothing spline," *Journal Royal Statistical Society, Series B*, 45, 133–150.

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

Wand, M.P. (1997), "A Comparison of Regression Splines Smoothing Procedures," Manuscript.

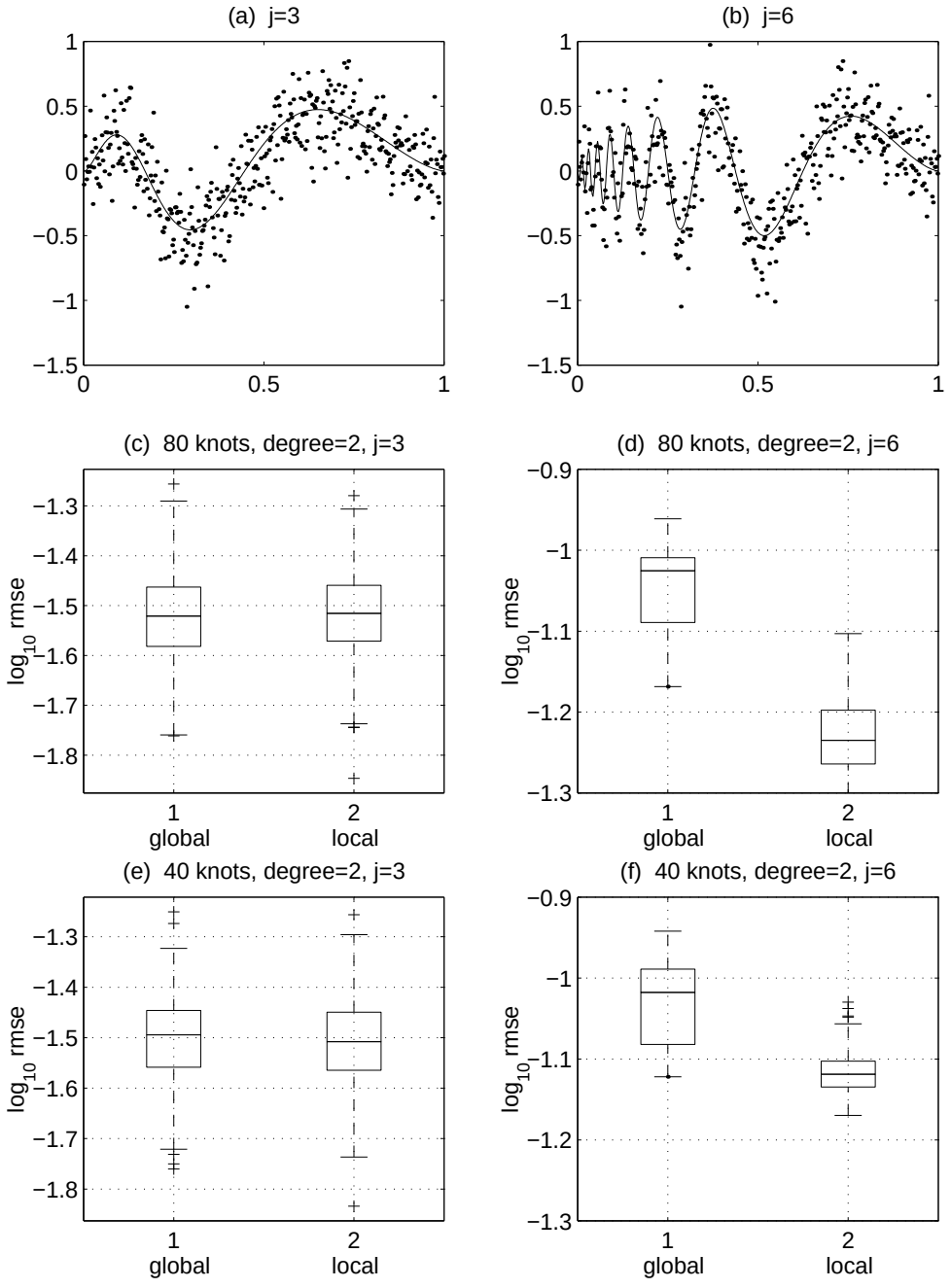

Figure 1: *Comparison of global and local-penalty parameters under low ($j = 3$) and severe ($j = 6$) spatial variablility in the oscillations of the regression function. (a) The regression function (solid) and one sample (dots) when $j = 3$. (b) Same as (a) but $j = 6$. (c) Boxplots of $\log_{10}$(RMSE) for 250 simulated samples using global and local-penalty parameters. 80 knots, quadratic splines, and $j = 3$. (d) Same as (c) but $j = 6$. (e) Same as (c) but 40 knots. (f) Same as (e) by $j = 6$.*

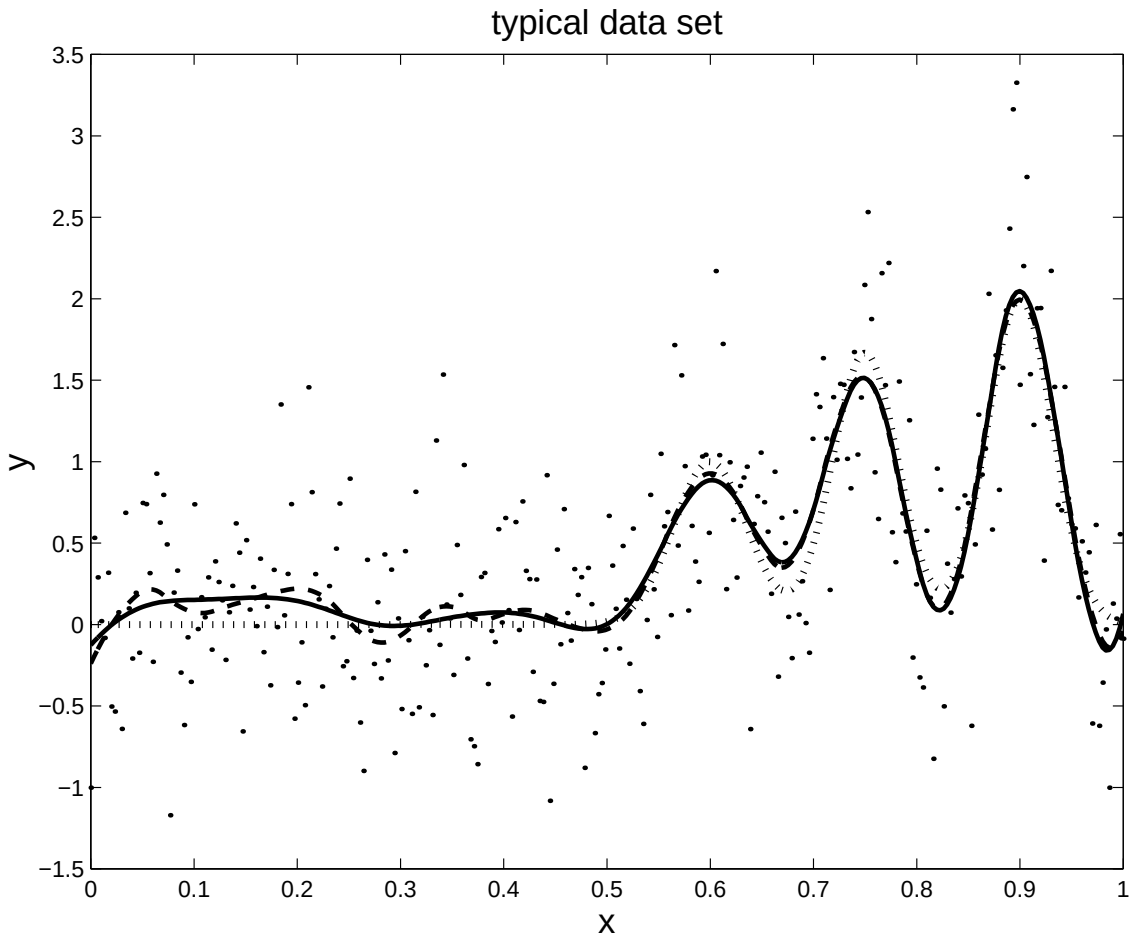

Figure 2: *Typical data in the Bayesian inference study. Local and global penalty splines with 95% confidence intervals based on the local spline.*

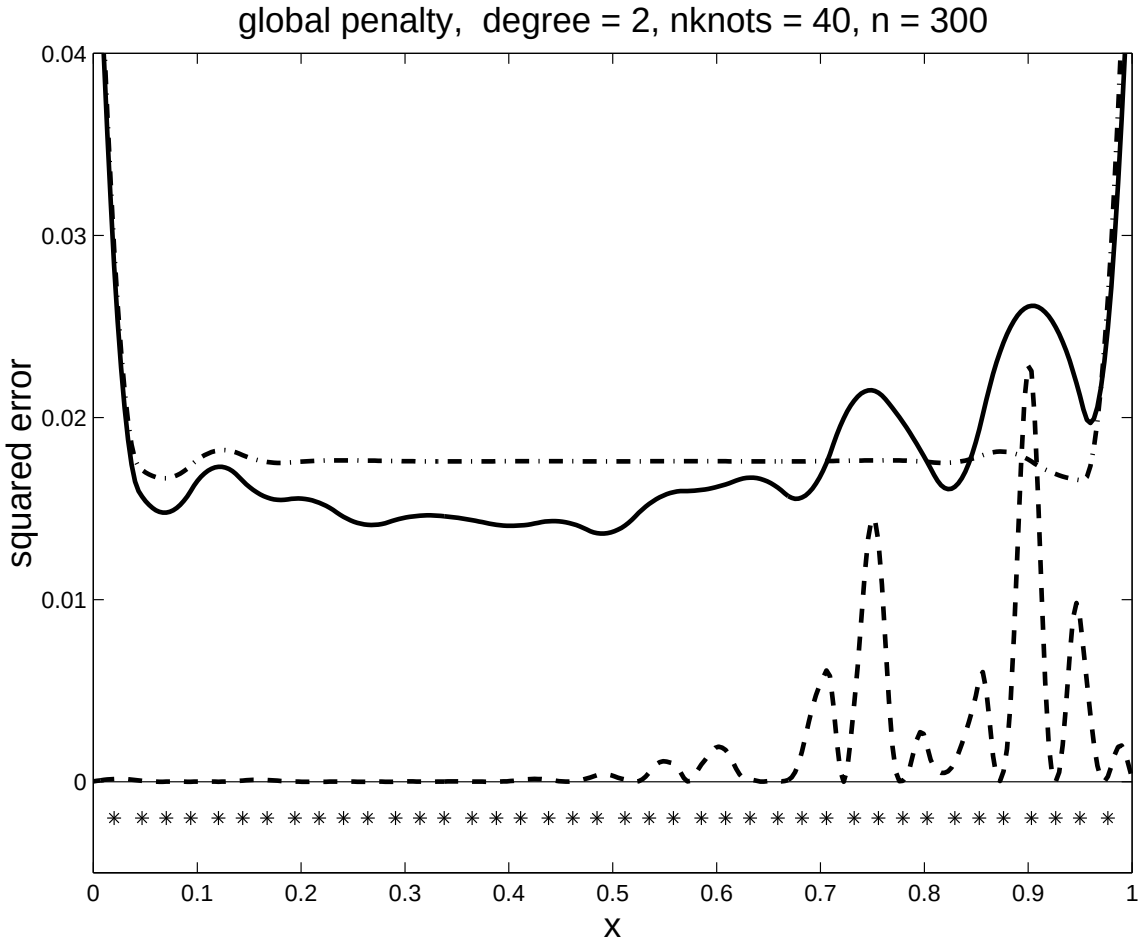

Figure 3: *Bayesian inference study. Behavior of the global-penalty estimator. Plots of pointwise MSE, squared bias, and average (over Monte Carlo trials) posterior variance. The MSE and the posterior variance have been smoothed to reduce Monte Carlo variance. The posterior variance assumes that* $\boldsymbol{\alpha}^*$ *is known, so the variability in* $\widehat{\boldsymbol{\alpha}}^*$ *is not taken into account.*

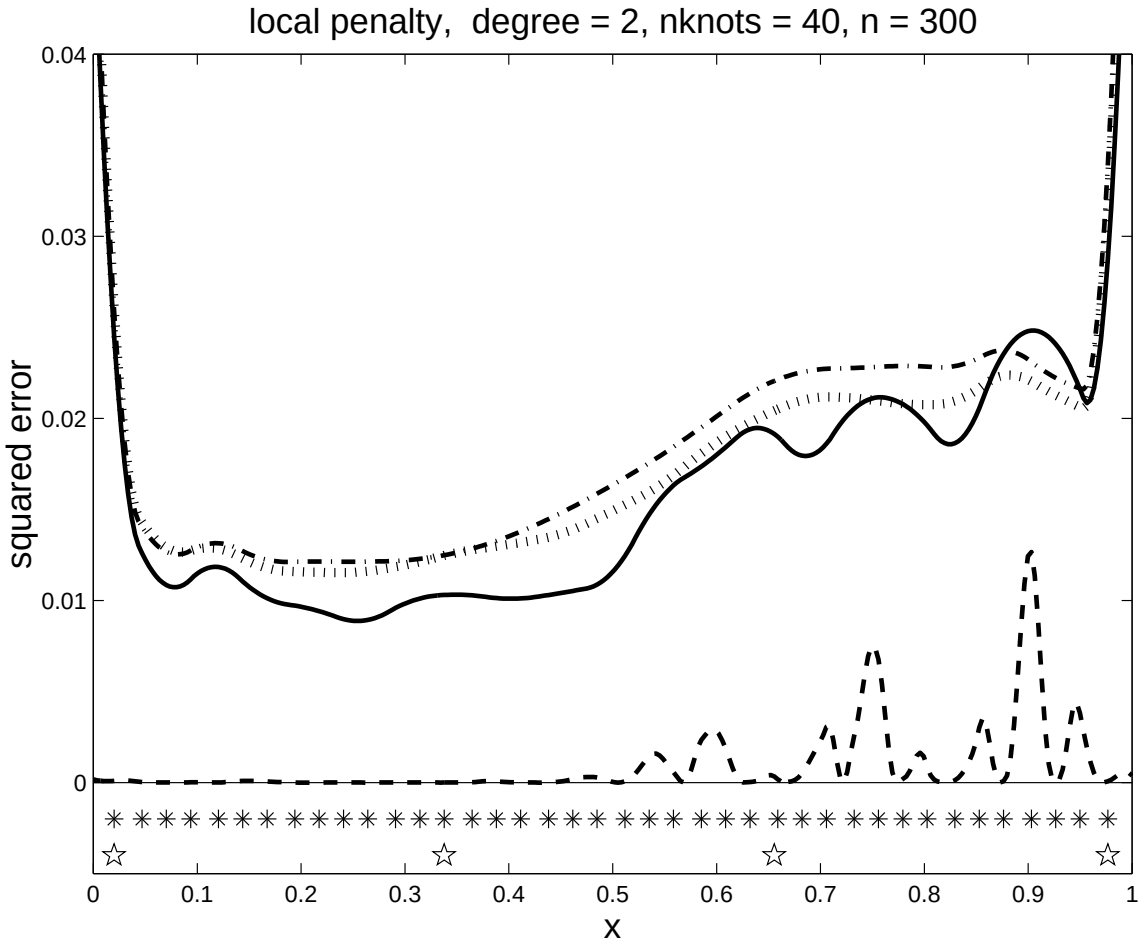

Figure 4: *Bayesian inference study. Behavior of the local-penalty estimator. Plots of pointwise MSE, squared bias, and average (over Monte Carlo trials) posterior variance. The MSE and the posterior variance have been smoothed to reduce Monte Carlo variance. The posterior variance has been corrected for variability in* $\widehat{\boldsymbol{\alpha}}^*$ *in two ways—the conservative (cons.) adjustment and the Kass-Steffey correction (corr.) are described in the text.*

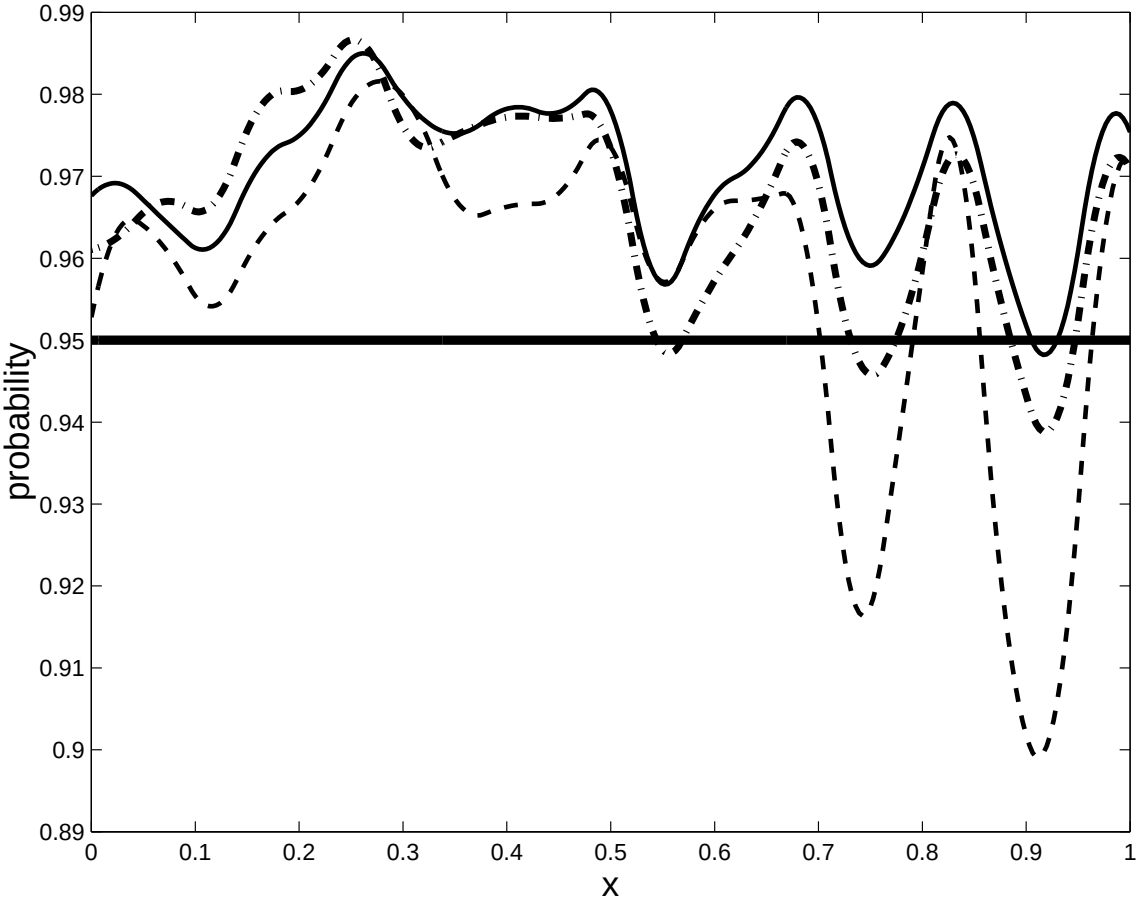

Figure 5: *Bayesian inference study using function (14). Pointwise coverage probabilities of 95% Bayesian confidence intervals for $m(x_i)$ using global and local penalties. The probabilities have been smoothed to remove Monte Carlo variability. The local penalty intervals use the conservative adjustment to the posterior variance (conservative) and the Kass-Steffey correction (corrected).*

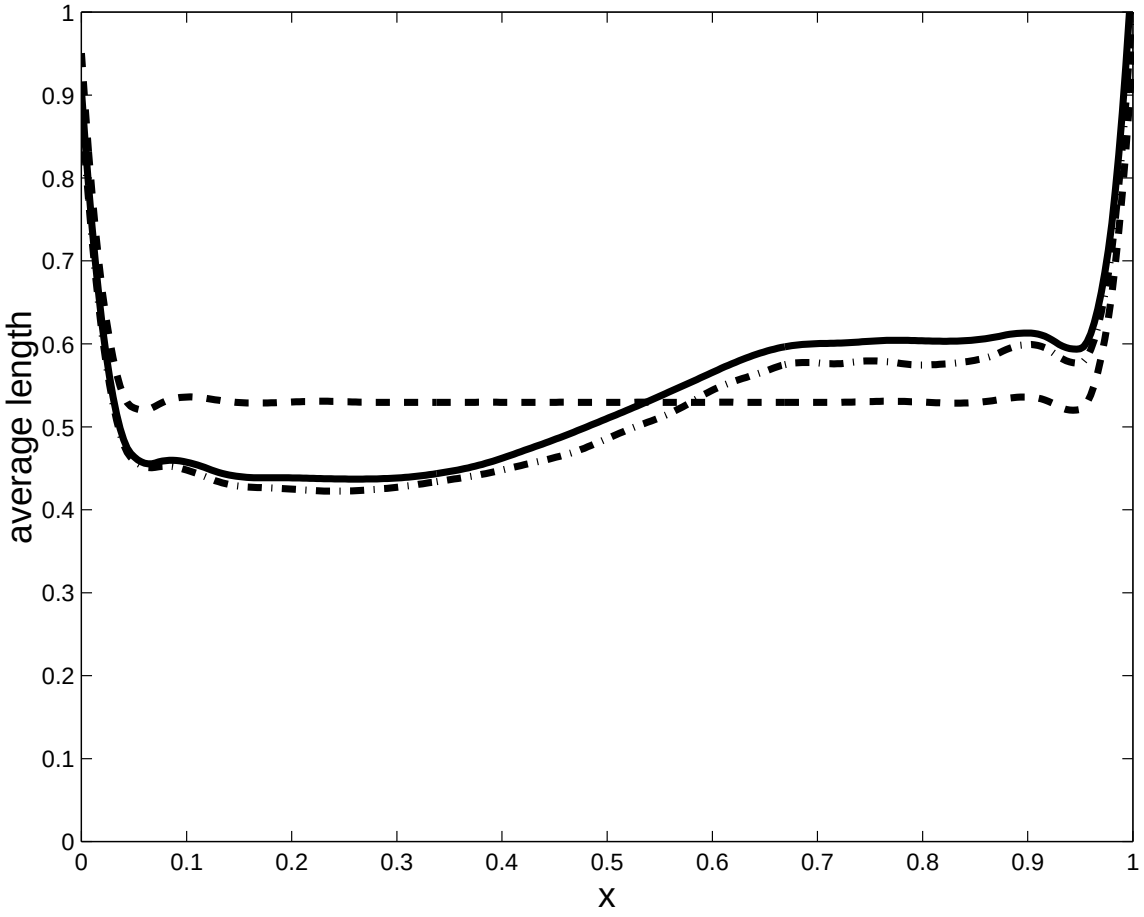

Figure 6: *Bayesian inference study using function (14). Expected lengths of 95% Bayesian confidence intervals for $m(x_i)$ using global and local penalties. The average (over Monte Carlo trials) lengths have been smoothed to remove Monte Carlo variability. The local penalty intervals use the conservative adjustment to the posterior variance (conservative) and the Kass-Steffey correction (corrrcted).*

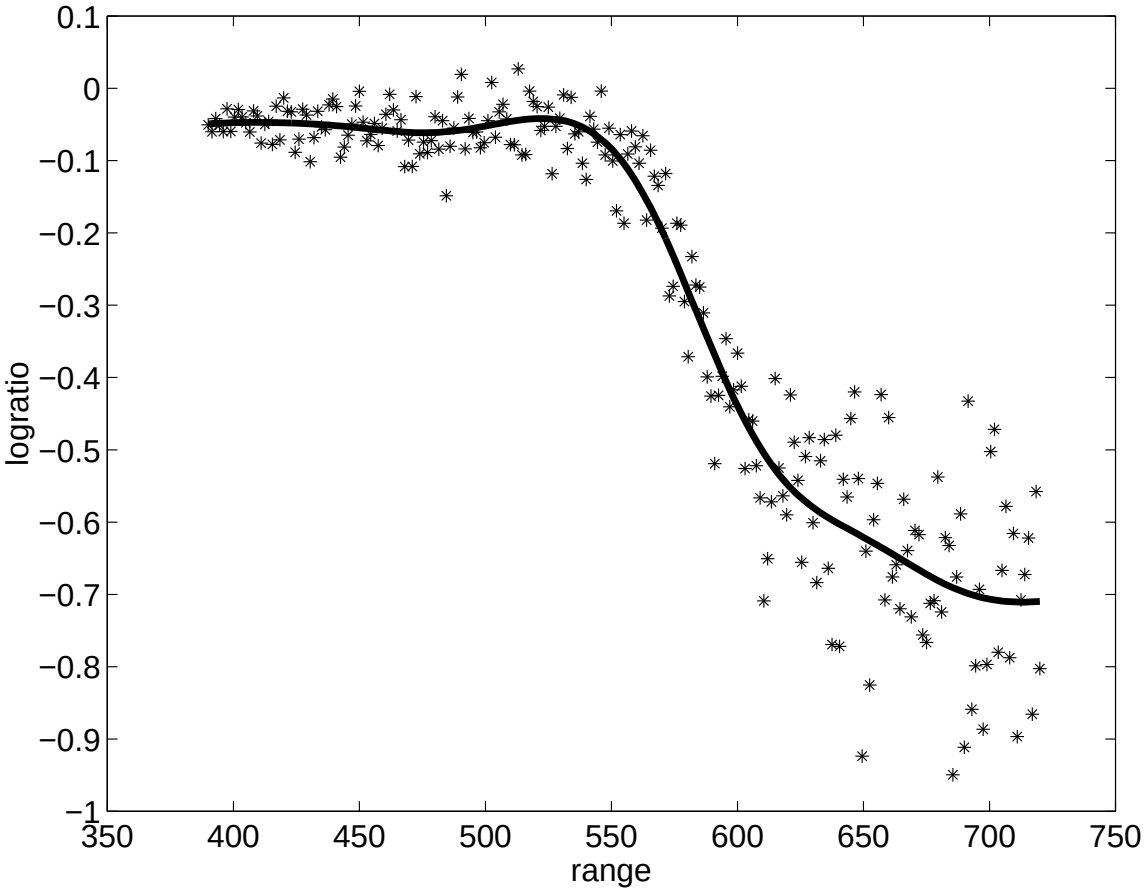

Figure 7: *LIDAR data. A global-penalty, quadratic spline fit has been added.*

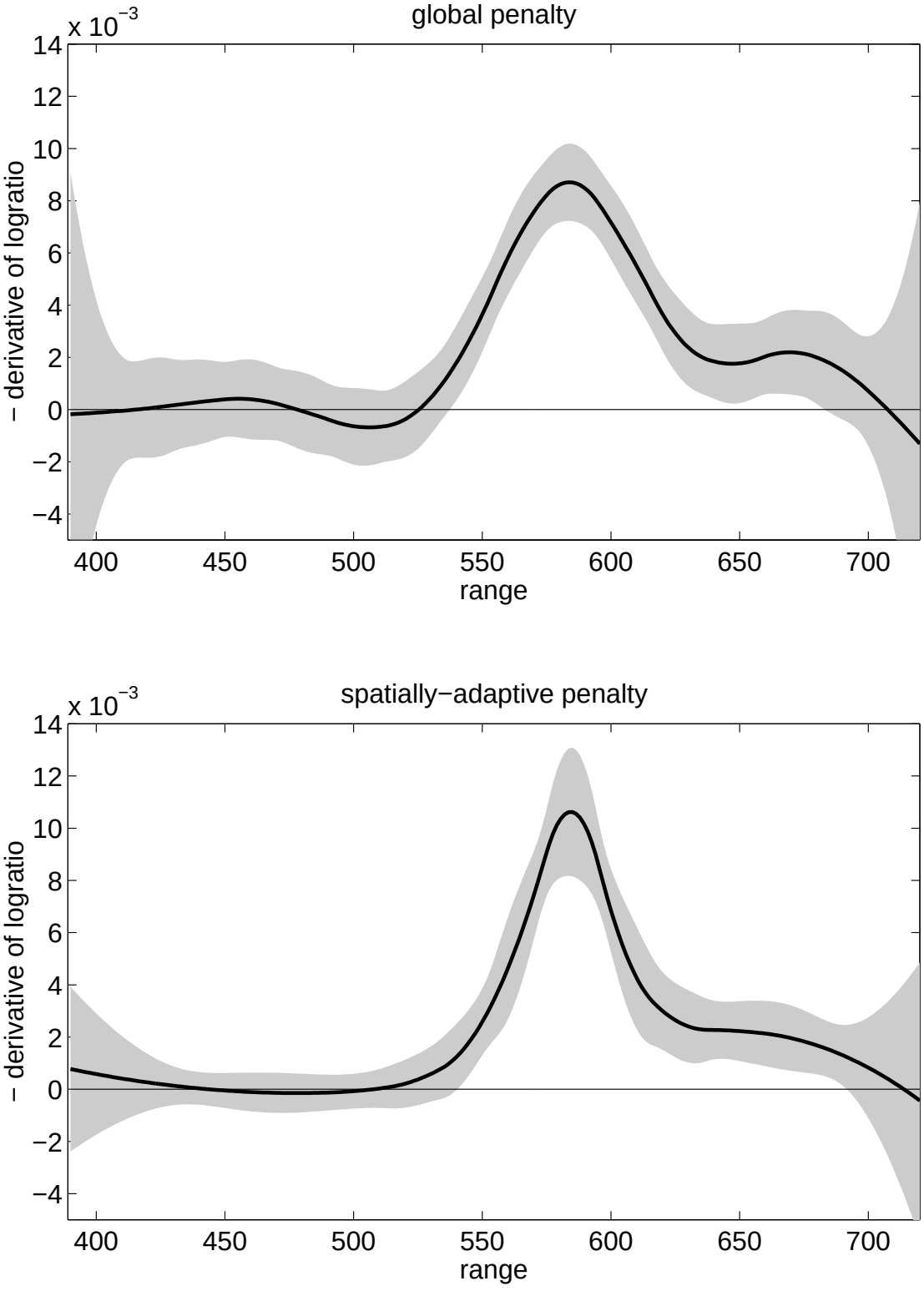

Figure 8: *Estimates of $-m'(\text{range})$ with global and spatially adaptive penalties. The shaded regions show pointwise 95% confidence intervals.*

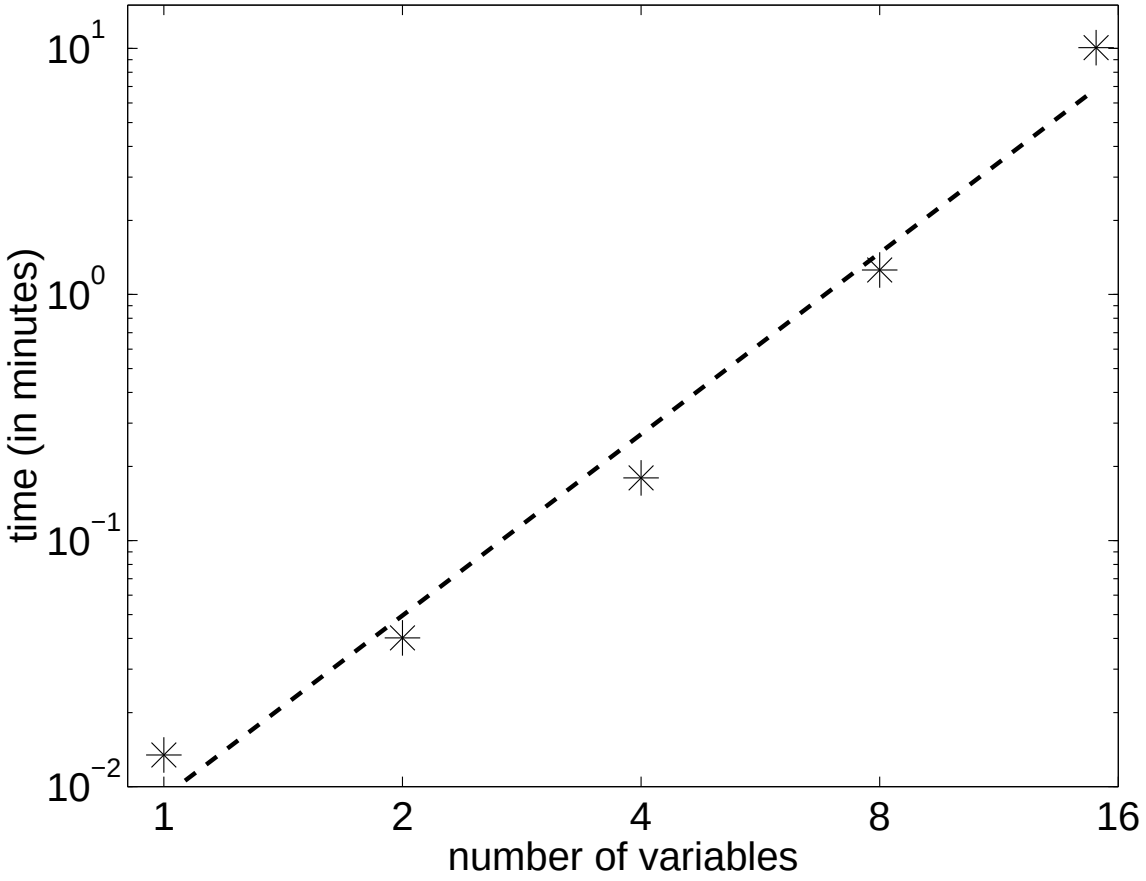

Figure 9: *Log-log plot of the computation time for fitting an additive model with local penalties as a function of the number of variables, L. A linear fit is also shown. The slope of the linear fit is 2.45.*