Introduction

First, we thank the editor, George Casella, for inviting comments on our paper from such a distinguished group of researchers. The diversity of approaches represented here certainly enriches the practice of function estimation. To best address these different perspectives, we have organized our responses around each of their discussions separately. However, there are some common themes that we would like to briefly highlight here because we will return to them repeatedly during our rejoinder.

Priors For function estimation, priors amount to statements about the smoothness of the underlying function ϕ. To the extent that subjective elicitation of prior distributions is possible here, formulations in terms of characteristics of ϕ will be most successful. Priors of convenience developed for generic model selection in regression or other settings may not be appropriate when working with finite-dimensional approximation spaces like splines.

Bayesian confidence intervals In principle, the Bayesian formalism provides an automatic mechanism for making inference about features of ϕ. Unfortunately, we find very few examples of confidence intervals for model-averaged spline estimates in the Bayesian literature on function estimation. Perhaps one difficulty here is that the object we're estimating is not really a cubic spline, and there is always bias present. In our own experience, it has been difficult to properly calibrate the prior distribution to produce both sensible point estimates and confidence intervals.

Bayesian-ism Many of the discussants comment on what is or what is not a Bayesian procedure. Our interest was mainly in evaluating computational methods, and each of the simulation set ups we studied was motivated by our previous experience with greedy schemes. We will not offer opinions as to whether previously published approaches are legitimately Bayesian or not.

Finally, we want to remind readers of an early paper in this area that has gone somewhat unreferenced in the modern literature on Bayesian splines. Halpern (1973) presented a thoughtful treatment of Bayesian knot selection that is very close to the methods mentioned in our paper and in this discussion. Unfortunately, the state of Bayesian computation in 1973 made most applications intractable.

Chipman, McCulloch, and George

We thank Chipman, McCulloch, and George for sharing their experiences formulating highly adaptive Bayesian methods for function estimation. The work by this trio on model selection, stochastic search and Bayesian versions of CART has influenced much of our paper. In terms of their characterization of priors as "a bet on what kind of models we want to consider," we feel that this gamble is best worded in terms of the object being studied, a function. Chipman, George, and McCulloch (2002) take an approach similar to ours when developing tree-based estimators. In their paper we find a nice division of prior specifications into a linear model class (representing separate tree structures; analogous to our linear spaces G) and then a member given the class (through choosing parameters; our selection of $g \in G$). Chipman et al. (2002) specify a prior on G to control the complexity of the trees that one expects, formulated in terms of the number of splits. This can be tuned to reflect one's beliefs that the data can best be described by small or

large trees. As Chipman et al point out, the prior $p(g|G)$ also plays a role in determining how the posterior weights models of different complexity (dimension). Given a knot sequence $\boldsymbol{t}$ and a basis set $B_1, \ldots, B_J$, we select this prior via a distribution on the coefficient vector $\boldsymbol{\beta}$ in the expansion

$$g(x; \boldsymbol{\beta}, \boldsymbol{t}) = \sum_j \beta_j B(x; \boldsymbol{t}) \,. \tag{101}$$

In our paper, this was always taken to be normal with mean zero and covariance λA^{-1}. With a special choice of A not related to smoothing, it is possible to calibrate the influence of $p(g|G)$ on posterior weights for each model class in terms of well-known selection criteria like AIC and BIC. We will have more to say about this in our discussions of Holmes and Kass and Wallstrom.

When developing (greedy) Logspline and Triogram, we attempted to make the "smoothing parameters" understandable and controllable through our software implementation. In so doing, users can tune these methods to agree with their prior beliefs about the smoothness of the function ϕ they are estimating. This also allows users to experiment with strategies like Chipman et al.'s cross-validation. In the early stages of an analysis, however, flexible smoothers are often used in an exploratory way, helping users formulate hypotheses about their data. In this setting, Logspline and Triogram function more like black boxes, and it is important to have reasonable automatic settings for hyperparameters. Fast computation is also important, given the iterative nature of data analysis. In crafting the simulations for this paper, we hoped to identify sensible default values for Bayesian versions of our ELM methods. By expressing the hyperparameters of our models in terms of the complexity of the underlying function, users can still easily incorporate their beliefs about ϕ in a natural way.

Chipman et al. are right about Logspline mixing more rapidly than the Triogram procedures. This difference also appears in the greedy implementation, where many of the Triogram models found during addition were visited again during deletion. At each deletion step, we are only able to remove certain vertices and still maintain a proper triangulation. In fact, at each step the number of candidates for deletion is limited by the structure imposed on the triangulations by the addition process itself; vertices added in the first few steps are typically too "connected" to be eliminated early in the deletion process. We hoped that a Bayesian Triogram procedure would allow us to visit more models. In thinking about how to implement the Triogram sampler, we borrowed from the experience of Chipman et al. with trees. In some sense, the structure imposed by our stepwise triangulation process is similar to the node splitting used for building trees. We also explored the tradeoffs between starting several chains versus a single, long chain. The simulations in our paper represent a technique that seemed to work best overall.

We agree with Chipman et al. on the potential gains of bootstrapping. Our present paper was inspired by an attempt to assess the usefulness of (then) emerging Bayesian computational methods. Over the last 5 years, techniques like stacking, bagging and boosting have emerged from, and in response to, ideas in computer science and machine learning. At this point, they have developed into extremely powerful, general purpose methods for combining estimators. To see what kinds of gains we can expect in our applications, we implemented a bagged version of Logspline based on 25 bootstrap samples. The column for Table 2 in our original paper corresponding to this new estimate is given below. We are not sure why bagging seems to have the hardest time with moderate sample sizes (200 and 1000) for the two smoothest distributions. Overall bagging improves over the greedy algorithm, but not as much as some of the MCMC approaches.

Distribution	normal				slight bimodal				sharp peak				
n	50	200	1000	10000	50	200	1000	10000	50	200	1000	10000	average
ratio over (i)	.88	.96	1.05	.80	.76	.97	.95	.65	.72	.84	.77	.69	.84

Table 101: MISE for bagging logspline on the 12 cases from Table 2

Holmes

We are pleased to have Holmes as a discussant. In the past few years, Holmes and his collaborators have applied the latest sampling techniques to problems in function estimation. From Bayesian versions of MARS that employ reversible jump Markov chain Monte Carlo to wavelet methods via perfect sampling, this group has contributed substantially to the practice of Bayesian curve and surface fitting.

Holmes highlights a rather general problem encountered when specifying priors for Bayesian model selection. For simplicity, consider the normal linear regression model in a J-dimensional spline space G defined by a given knot sequence $\boldsymbol{t}$ of length K, and a basis (101). Let $\boldsymbol{\beta}, \sigma^2$ have a conjugate normal-inverse gamma prior distribution

$$w(\boldsymbol{\beta}, \sigma^2) \;\propto\; \left(\sigma^2\right)^{\frac{-d+k+2}{2}} \exp\left[\frac{-(\boldsymbol{\beta}-\mathbf{b})^t V^{-1}(\boldsymbol{\beta}-\mathbf{b})+a}{2\sigma^2}\right], \tag{102}$$

that depends on several hyperparameters: $a, d \in \mathbb{R}$, the vector $\mathbf{b} \in \mathbb{R}^J$, and a $J \times J$ symmetric, positive definite matrix V. Valid ranges for these parameters include all values that make (102) a proper density. We considered covariance matrices of the form $V = \lambda A^{-1}$ where A is chosen to reflect our beliefs about the smoothness of an unknown function ϕ. For a fixed set of knots, the prior precision parameter λ affects the look of the fitted curve in a fairly predictable way. Given observations $(X_1, Y_1), \ldots, (X_n, Y_n)$, the posterior distribution of $\boldsymbol{\beta}$ is multivariate t with mean

$$\left(B^t B + \frac{1}{\lambda} A\right)^{-1} B^t Y\,, \tag{103}$$

where B is the $n \times J$ design matrix based on the basis (101), $[B]_{ij} = B_j(X_i)$, and $Y = (Y_1, \ldots, Y_n)^t$. Therefore, selecting a large value of λ can eliminate the effect of smoothing, often producing a wiggly fit; while small values tend to shrink $\boldsymbol{\beta}$ toward the null-space of A (for the prior associated with cubic smoothing splines, this is simply the space of linear functions). In Bayesian terms, a large value of λ indicates little information about $\boldsymbol{\beta}$, while a small value of λ indicates a strong belief that ϕ is very smooth. As Holmes points out, however, things become more complicated when comparing spline spaces of different dimensions (having different numbers of knots). In this case λ partially specifies the weight assigned to each space and we are subject to the so-called Lindley-Bartlett paradox; non-informative prior for a fixed model class becomes maximally informative when comparing models of different dimensions. In short, the posterior piles mass on the spline space with the fewest knots.

Holmes presents a well-known covariance-representation that alleviates the paradox and is meant to calm the non-Bayesian. Unfortunately, it does not eliminate the difficulty in specifying λ. Many researchers, including Holmes, have dealt with the problem of prior choice by calibrating Bayesian procedures with frequentist measures. In Holmes and Denison (1999), for example, we find a prior on the precision parameter

λ that is tuned to the degrees of freedom of the fit, DF, or explicitly

$$p(\lambda|\sigma^2) \propto \exp(-c\,DF)\,,$$

where DF is computed as the trace of the smoothing matrix (and the domain of the distribution is truncated to obtain a proper prior). The posterior dependence on c is then compared in terms of an equivalence with classical model selection criteria that is similar to our range of AIC statistics.

Holmes correctly points out that one needs to be careful in blindly matching terms to establish this kind of correspondence, noting the difference between the maximized or profile likelihood

$$p(y|\widehat{\boldsymbol{\beta}},\boldsymbol{t}) = \max_{\boldsymbol{\beta}} p(y|\boldsymbol{\beta},\boldsymbol{t})\,, \tag{104}$$

and the marginal likelihood

$$p(y|\boldsymbol{t}) = \int p(y|\boldsymbol{\beta},\boldsymbol{t})p(\boldsymbol{\beta}|\boldsymbol{t})d\boldsymbol{\beta}\,. \tag{105}$$

Naturally, we agree that the two are very different, and it was a comparison with model selection criteria similar to Holmes' that led us to the geometric prior on model size. In the discussion from Kass and Wallstrom we are reminded that

$$\log p(y|\boldsymbol{t}) \approx \log p(y|\widehat{\boldsymbol{\beta}},\boldsymbol{t}) - \frac{J(\boldsymbol{t})}{2}\log n\,.$$

Therefore, by adjusting the prior on model size, we could recover the general form of BIC and restore some of the protection against overfitting Holmes claims for the marginal likelihood. Inevitably, this kind of benchmarking against known criteria seems useful even for the purest Bayesian, and we will have more to say about it in our response to Kass and Wallstrom. Before we leave this point, we should mention that the latest minimum description length criteria for model selection developed by Barron, Rissanen and Yu (1998) and Hansen and Yu (2001) attempt to make up for the shortcomings of the maximized likelihood by a certain (re)normalization procedure.

In evaluating procedures like Logspline and Triogram, we have looked to theoretical results for guidance. While the results in Stone (1994), Hansen (1994) and Huang (1998, 2001) are asymptotic, they do provide an indication that a spline-based approach to ELMs is sensible. A similar theory for Bayesian versions of these methods is still emerging. For recent results, we refer the interested reader to articles by Zhao (1993, 1998), Barron, Schervish, and Wasserman (1999) and Shen and Wasserman (2001). In the last paper we find rates of convergence associated with so-called sieve-priors, a set-up that most closely mimics our own use of finite-dimensional approximating spaces. Shen and Wasserman point out that while early work in this area produced a rich set of priors that led to consistent estimators, achieving the optimal rate is much more delicate.

Kass and Wallstrom

We first want to thank Kass and Wallstrom for a their contribution to the discussion of Denison, Mallick, and Smith (1998a).

It seems that the work described by Kass and Wallstrom and our own motivation for exploring Bayesian procedures for splines share a common source. In our rejoinder to Stone, Hansen, Kooperberg, and Truong (1997), we presented a connection between the Bayesian method of Smith and Kohn (1996) and our own

greedy schemes that attempted an approximate minimization of a generalized AIC statistic. We began with the straightforward observation that by using Zellner's g-prior (Zellner, 1986) for the coefficients, the resulting posterior is mathematically equivalent (or nearly so) to AIC, BIC or anything "in between." In terms of (102), this amounts to setting $V = c(B^tB)^{-1}$ and $a = d = 0$, and letting c vary from 2 (for AIC) to n (for BIC). Ed George pointed us to this posterior calibration in a paper he crafted in 1997, which eventually appeared as George and Foster (2000). Of course the desire to relate Bayesian and frequentist methods is an old one, and explicitly expressing a posterior in terms of known model selection criteria goes back at least to Smith and Spiegelhalter (1980),[1] although it is visible even in Lindley (1968).

In their 1996 paper, Smith and Kohn were selecting the "best" knot configuration having the highest posterior probability. Given the equivalence between this posterior and our own generalized AIC statistics for ELMs, both procedures were attempting to optimize the same quantity. Having identified a good knot sequence, Smith and Kohn used the posterior mean of $\boldsymbol{\beta}$ to form a curve estimate. Under the g-prior version of (102) this is simply $\frac{c}{c+1}\widehat{\boldsymbol{\beta}}$, where $\widehat{\boldsymbol{\beta}}$ is the OLS estimate (see equation 103). Since the values of c used were at least 100, the posterior mean was essentially $\widehat{\boldsymbol{\beta}}$, and now the only difference between our greedy methods and the Bayesian approach of Smith and Kohn was the search scheme. This was our starting point.

To reduce computation, Smith and Kohn restricted their attention to at most 30 knots, placed at order statistics of the data. With such a small problem, it was possible to even use branch and bound techniques to search all possible combinations of candidate knots from within Splus. Since that point, many other Bayesian spline-based estimators have appeared. Building on examples of Green (1995), Denison, Mallick, and Smith (1998a) treated the knot sequence more like our own greedy schemes, in that each of the data points was a candidate knot. BARS (Bayesian regression splines) proposed by DiMatteo, Genovese, and Kass (2001), is a hybrid that borrows the prior structure of Smith and Kohn, but adapts Green's sampler. In writing our paper, we were naturally aware of the shortcomings of DMS, although our insight came not from formal Bayesian thinking but by a simple comparison of the "objective function" represented by their posterior. Without a prior on the coefficients $\boldsymbol{\beta}$, the posterior did not provide enough of a penalty for large models. We attempted to correct this with the geometric prior on model sizes, yielding a posterior that agrees with BIC. In so doing, our penalty on dimension and that for BARS are the same.

Based on our experience with "free-knot splines," we also considered simulations that paired the geometric prior along with a proper prior on $\boldsymbol{\beta}$, imposing an extra penalty on large models. Recall when working with free-knot splines, we treat the sequence $\boldsymbol{t}$ not as a discrete index for separate models, but rather as a continuous vector of parameters. Kooperberg and Stone (2001) used free-knot splines for density estimation and argued that the most appropriate form of AIC was

$$\log p(y|\widehat{\boldsymbol{\beta}}, \boldsymbol{t}) - \frac{J+K}{2}a$$

where again J is the dimension of the spline space and K is the number of knots in $\boldsymbol{t}$. In short, an extra penalty on dimension was necessary to produce reasonable point estimates. No matter what principle is applied to determine the penalty, it is clear that the tradeoffs between sample size, model dimension and assumed smoothness of ϕ are complex. Since any theoretical guidance is likely to be asymptotic, new procedures should be tested on a wide variety of simulations. However, returning to the BARS paper, we find only three simulations, each of which are based on modestly-sized problems with very similar characteristics. That is, one expects the "optimal" number of knots needed for approximating the three test functions is the similar

[1] These authors speak of a "generalized AIC" criterion, of which BIC is just one example.

and the signal-to-noise ratios were all close (basically 3:1). Curiously, the simulations for BARS only involve 10 runs per test case, making the very strong claims by Kass somewhat premature. It would be good to see BARS run under many more conditions.

Aside from the posterior calibration to BIC, BARS differs from DMS in two other ways. First, in designing the moves for BARS, DiMatteo et al. (2001) borrow a "key idea" of Zhou and Shen (1998), that more knots are needed in regions where a curve changes rapidly. This is nothing more than the definition of a spline space with repeated knots (Schumaker, 1993); that is, as one knot approaches another, the splines lose a derivative. In a cubic spline space, two coalescent knots allow the fit to break in the first derivative, and three coalescent knots produce a break in the function itself. This behavior has guided the design of our own ELM procedures. In regression, if we have reason to believe the underlying curve has sharp features, we might consider relaxing the separation condition in the definition of an allowable space. However, in most cases, for reasonable sample sizes, this condition is not restrictive and the final curves can track strong features when they exist. We should add that in BARS, there is no encouragement in the prior itself for nearby (or coalescent) knots. The "locality property" is purely a function of the proposal.

Aside from the locality property, there is another big difference between BARS and DMS. DiMatteo et al. (2001) use a natural spline basis for BARS, while DMS uses ordinary B-splines. While both are smooth, piecewise polynomials, the natural splines are forced to blend into a linear functions outside the $[t_1, t_K]$. It is well-known that this reduced space improves the variance at the ends of the data dramatically (we know that one of the referees of DMS encouraged the authors of that paper to use a natural spline basis). Kass and Wallstrom state that BARS performs better than a version of DMS that has been "corrected" to agree with BIC. Unfortunately, it is not clear whether the knot proposal distribution or the natural spline basis is resposible for this improvement. DiMatteo et al (2001) has only one panel of plots comparing the variants of DMS and BARS, and this consists of a single estimated curve per method taken from one simulation setup (the test function having a very sharp break in the middle of the domain).

We should add that working with natural splines is a bit trickier than regular B-splines because of the boundary conditions. Breiman (1990) proposes a constrained least squares fit to enforce linearity in the tails during knot deletion. Luo and Wahba (1998) describe a stepwise algorithm for natural splines that would allow BARS to take advantage of 1-degree-of-freedom alterations as we have done for Logspline and Triogram. For natural splines that are linear outside the interval $[0, 1]$, Luo and Wahba use a basis of the form

$$\phi_1(x)\,,\ \phi_2(x)\,, \qquad \text{and} \qquad R(x, t_k)\,,\ k = 1, \ldots, K \tag{106}$$

where $\phi_1(x) = 1$, $\phi_2(x) = k_1(x)$ and

$$R(x, x') = k_2(x)k_2(x') - k_4\left(\,|x - x'|\,\right)\,. \tag{107}$$

The functions k_1, k_2, and k_4 are constant multiples of Bernoulli polynomials and are given by

$$k_1(x) = x - 1/2\,, \qquad k_2(x) = (k_1^2(x) - 1/12)/2$$

and

$$k_4(x) = (k_1^4(x) - k_1^2(x)/2 + 7/240)/24\,.$$

Unlike the truncated power basis, the so-called kernel functions $R(\cdot, \cdot)$ are not "one-sided" but have global support. They do, however, share the property that each candidate knot t_l corresponds to a single function

$R(\cdot, t_l)$. That is, both $(x - t_k)_+^3$ and $R(x, t_k)$ have discontinuities in the $3rd$ derivative only at the point $x = t_k$. As it stands, it is not clear how BARS is implemented and whether or not this kind of shortcut has been used.

Given (103), taking $V = \lambda(X^t X)^{-1}$ in the regression context makes things much easier computationally because the posterior mean for $\boldsymbol{\beta}$ given a single space G is a scale version of the OLS estimate $c/(1+c)\widehat{\boldsymbol{\beta}}$. What's troubling, however, is that the implied shrinkage here is toward zero and depends on the basis (101) used. For moderate values of c, we expect to see odd smoothing behavior which will become even more problematic when we start averaging fits. This does not seem appropriate to us, so for Triogram regression we chose a smoothing prior with a null space consisting of planar functions. (The classical cubic smoothing splines also reduce to linear functions.) Unfortunately, our experiments with Triograms are somewhat inconclusive because any kind of reasonable averaging can improve the piecewise linear fits if the underlying function is smooth (there are bigger effects to overcome before the behavior at the peaks is a concern). Kass and Wallstrom also mention the idea of asigning a prior to λ or setting it in an empirical-Bayes way. The latter approach is followed in George and Foster (2000) for $V = \lambda(X^t X)^{-1}$. For a general prior covariance matrix, a simple iteration is required to select λ in this way. Details can be found in Hansen and Yu (2001).

When considering normal priors on $\boldsymbol{\beta}$ for density estimation, the story is a bit clearer. First, taking a normal prior with covariance proportional to $(X^t X)^{-1}$ does not represent a reduction in computation. Shrinking toward zero in this context produces a uniform density. Despite this seemingly bad property, we investigated the use of this prior, and found that it smoothes away peaks (even when $\lambda = 1/n$, as proposed by Kass and Wallstrom), and it makes methods (v) and (viii) look very much like (vii). We also tried the so-called unit information prior associated with density estimation and achieved similarly discouraging results. Unlike Triograms which generically benefit from averaging, the proper smoothing can have a big effect on the performance in Logspline.

In attempting to recast our results for Logspline, Kass and Wallstrom fail to appreciate the fact that density estimation is a very different problem from (generalized) regression. As mentioned above, the closer you position knots, the greater the (potential) discontinuity in the fitted curve. In density estimation, this extra flexibility can drive the likelihood to infinity if knots are placed too close to each other relative to the scale of the data, an artifact that does not reflect the suitability of a model with nearby knots. When this happens, the search procedures underlying both the stepwise and our MCMC versions of Logspline can get stuck in local solutions. Therefore, in each case we require knots to be at least a few (usually three) data points apart. Keep in mind that for density estimation, regions with sharp features (peaks) contain many more data points, and hence many more candidate knots; there is a natural coupling between structures in the function we're estimating and the spread of the data. In short, the restriction on knot placement does not hinder Logspline's ability to track strong features. In fact, under this restriction, we can ensure that knots proposed fairly close together are much more likely to be retained when they are near a peak, as the likelihood will increase much more with the addition of such knots (see Figure 1 of SHKT97). We have experimented with many different knot location schemes, but for Logspline proposing too many close knots actually reduces the ability of the MCMC algorithm to explore the complete distribution, as the chain gets stuck more easily. On the other hand, with careful knot location, the convergence of MCMC chains in Logspline is very good. We carried out a substantial amount of additional chains, suggesting that the Logspline chains mix well. See also the discussion of Lindstrom, suggesting that a chain of length 500 is

almost as good as a chain of length of 5000. (For single calculations, rather than a large simulation study, we would advise somewhat longer chains too.) Finally, we should add that the situation for (univariate) regression is different because the likelihood is well behaved as knots coalesce providing we use a sensible basis (and do not add so many knots that we lose identifiability of the model). Still, it is not clear if the locality heuristic used in DiMatteo et al (2001) actually performs better or if the potential improvements claimed by Kass and Wallstrom are due to their implementation of the reversible jump algorithm (as compared to that of DMS), their use of the natural spline basis, or their selection test cases (small problems with common noise levels).

We mostly agree with Kass and Wallstrom's interpretation of our results; the label attached to (iii), whether it be Bayesian or approximately Bayesian or quasi-Bayesian, is of little concern for us. As suggested method (vii) looks like smoothing splines; judging by the MISE results, the method proposed by Wahba, Lin, and Leng in their discussion behaves most like this approach. We believe that the "perplexing anomalies" cited by Kass and Wallstrom are primarily caused by a lack of appreciation for Logspline and the stepwise algorithm, which is really quite good. When we compare approaches (iii) and (viii) we note that for small sample sizes (iii) tends to overfit, so that the prior on $\boldsymbol{\beta}$ (which makes it (viii)) helps, and without that prior the overfitting for (iii) hurts relative to the greedy algorithm, which overfits much less, but for large sample sizes there is enough variability in (iii) to compensate for the overfitting. Also, we actually do carry out annealing for method (ii). Details were omitted when we were asked to reduce the length of the paper during revisions.

We agree with Kass and Wallstrom that one of the interesting features of a Bayesian analysis is the assessment of uncertainty. While these Bayesian schemes seem to give sensible point estimates, we do not hold out much hope for confidence intervals. The BARS paper, for example, doesn't present any results with simulated data on coverage, but reserves plots with intervals for real-data problems (assessing their quality subjectively). In Kooperberg and Stone (2002) we examine Bayesian confidence intervals for the income data for versions (v) and (vi) based on an MCMC run of length 100,000. Based on comparisons with bootstrapping procedures we concluded there that these intervals were considerably narrower than frequentist confidence intervals; that is, calibrating the posterior to achieve a reasonable point estimate yielded optimistic intervals. This really emphasizes the points made by Lindstrom in her discussion that to obtain "credible" Bayesian confidence intervals the priors need to be selected much more carefully. We would challenge the Bayesian function estimators to produce a few confidence intervals in their papers. It is very hard to find any that come from simulations where we can judge their usefulness. Wang and Wahba (1995) compare Bayesian intervals for smoothing splines with those for bootstrapping. A similar study for the model-averaged fits would be useful.

Our goal of the paper was to compare Bayesian and stepwise methods, and compare how good they are doing. Our goal was not to advocate one approach or another. From Kass and Wallstrom's discussion of BARS we get the impression that it is a useful procedure. We would like to see a comparison of their procedure with the one by Smith and Kohn (1996), which sounds very similar. However, we feel that the conclusion that "BARS appears to be the best available method for spline based curve fitting in ELMs" is somewhat premature. In particular, we would like to see the performance of BARS

- illustrated on a suite of very different test functions with different sample sizes and different signal-to-noise ratios;
- extended to demanding nonlinear applications, such as density estimation and survival analysis;

- tested in high dimensional applications with thousands of cases;
- compared to greedy (stepwise) algorithms in terms of computing time.

Koenker and Mizera

The L-1 methods made popular by Koenker and his co-authors have filtered in to extremely effective spline methods for estimating median fits (Koenker, Ng and Portnoy, 1994; He, Ng and Portnoy, 1998). Koenker and Mizera base their version of Triograms on a Delaunay triangulation, producing an underlying linear space G from which a single model is selected via a roughness-penalized L-1 error criterion. In Rippa (1990) we find an interesting characterization of the Delaunay triangulation based on roughness that we think is worth mentioning. Suppose we have data points $(\boldsymbol{x}_1, Y_1), \ldots, (\boldsymbol{x}_n, Y_n)$. We want to interpolate the values Y_i at $\boldsymbol{x}_i$ and will do so by creating a triangulation with $\boldsymbol{x}_i$ as the vertices. There are many many ways to do this. For each one, we will measure its roughness via the (Sobolev semi-norm)

$$\sum \int_{\delta} \left[\left(\frac{\partial g}{\partial x} \right)^2 + \left(\frac{\partial g}{\partial y} \right)^2 \right] dxdy$$

where the sum is over all triangles $\delta \in \triangle$. Then, Rippa (1990) shows that the Delaunay triangulation of a set of points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ is a minimal roughness triangulation. With at least one sensible smoothness constraint, the Delaunay triangulation is the appropriate thing to do. However, it is likely that for other smoothness measures, the Delaunay is not optimal and that other structures are more appropriate. Rippa (1992) explores a similar line of reasoning and uses an edge-swap algorithm to find promising triangulations. It is known that we can generate all the triangulations of a given set of points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ by the first move type in Figure 4 of our original paper. In developing the greedy version of Triogram and its Bayesian counterpart, however, we felt that there was much to be gained by taking smaller triangulations that were better adapted to the underlying function being estimated. Hence, it was important to develop a suite of moves that allowed us to step through models of different sizes.

For our prior on $\boldsymbol{\beta}$, we borrowed the penalty from Koenker and Mizera, first communicated to us by Koenker. We were pleased with the form because it built both on their work and on the approach of Nicholls (1998), who considered piecewise constant surfaces over adaptively chosen triangulations. We tried a variety of other penalties, including those suggested in Dyn and Rippa (1990). Each have a null space of planar functions. In simulations with this form using both smooth functions and simple examples exhibiting sharp edges, it seems to perform well. As a further test, we consider a dataset know to exhibit a simple hinge, not aligned with either coordinate axis (Cleveland and Fuentes, 1996). While we have seen that a certain amount of smoothing is possible with the Bayesian estimator when the underlying target function is smooth, in this case, we hope that the sampler will spend time in very simple, "nearby" ridge models. This would allow the non-greedy schemes to still capture ridges effectively. In Figure 101 we present two surfaces, one from simulation set-up (v) and one from (iii). A careful analysis by Cleveland and Fuentes suggested that the best fit was a hinged pair of planes, very similar to the fit in the left-hand panel corresponding to (v). It is clear from this figure that Poisson prior (iii) yields a chain that spends too much time in overly-complex models and the fit is badly degraded. The surface obtained by prior specification (v), on the other hand, is an improvement over the greedy scheme. While it is difficult to tell from the perspective plot, the ridge or central hinge more closely follows the line found by Cleveland and Fuentes (1996).

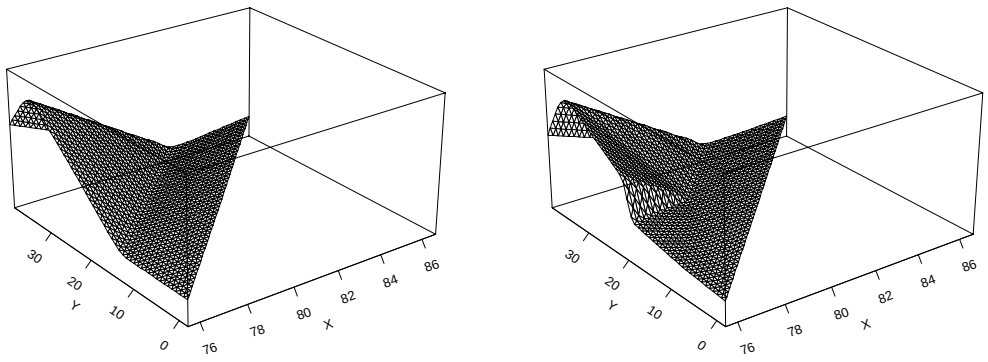


Figure 101: The volt data from Cleveland and Fuentes (1996). On the left we have the fit from prior specification (v), while on the right we display the fit for (iii).

Lindstrom

Lindstrom has a considerable background in working with free-knot splines. She has developed an extremely attractive penalized appraoch to fitting such models (Lindstrom, 1999) and has a long history in working with functional data in medical applications (see Lindstrom, 1995, for example). We appreciate her extensive comments on the computational aspects of our Bayesian schemes. In our implementation of the samplers in this paper, we used the same number of iterations for each of the model selection schemes. The greedy scheme naturally takes far fewer "iterations" or moves. For the income data, on our current machine (which is not the machine on which the original calculations were carried out), the greedy Logspline algorithm takes about 0.5 seconds of CPU time, while a simulated annealing or MCMC run of 5500 iterations takes about 200 seconds. The differences between the various versions of the sampling methods do not have a major influence on the CPU time.

Lindstrom asks us to compare this to stepwise algorithms with random restarts. This comparison appears most relevant to the greedy version (i) and the simulated annealing version (ii). For the income data, version (i) has a BIC value of 161936.4, while the simulated annealing version, using a chain length of 5500 and the best greedy solution as initial knots, has a BIC value of 161918.3. Table 1 contains some alternative approaches using (several) shorter chains, or the stepwise algorithm with several random restarts. (For the random restarts, we positioned 7 knots at randomly selected datapoints, followed by the stepwise addition and deletion algorithm of Stone, Hansen, Kooperberg, and Truong (1997), allowing a maxium of 15 knots.) The bottom line of this seems to be that, at least for this example, random restarts are just as good (but not better) than simulated annealing using the same amount of CPU time. It remains open what works better if we want to get close to the "best" solution. However, for all purposes the differences in BIC are not very large.

We fully agree with Lindstrom's remark that when we want to use Bayesian methods to carry out

method	number of chains	number of seeds or restarts	BIC	SD	seconds CPU time
stepwise (version (i))	1	1	161936.4	–	0.4
simulated annealing (ii)	5500	1	161918.3	–	200
shorter annealing chains	500	1	161922.1	2.5	20
combining real short chains	100	5	161921.7	1.9	20
stepwise with restart	1	50	161921.8	0.8	20

Table 102: CPU and BIC results for various alternative stochastic optimization methods. For the last three methods the BIC value is the average over five repetitions of the complete procedure, and the SD is the corresponding standard deviation.

inference, the choice of priors becomes much more critical than when we primarily want to get a point estimate. See our remarks about the contributions by Kass and Wallstrom and by Chipman, McCulloch and George, and also the example in Kooperberg and Stone (2002). Finally, we agree with Lindstrom that radial basis functions are a sensible approach to multivariate modeling. In this case, we might consider an expansion of the form

$$g(\boldsymbol{x}) = \sum \beta_i R(\boldsymbol{x}, \mathbf{v}_i) \tag{108}$$

for a fixed "kernel function" R and a set of "knots" $\mathbf{v}_1, \ldots, \mathbf{v}_K$. Holmes and Mallick (1998) describes a Bayesian setup that also uses reversible jump Markov chain Monte Carlo, but selects how many knots and where they should be placed. The basis (107) is an example of this kind of expansion for univariate functions. Having said that, we still feel that the Triogram basis has its role in multivariate function estimation. It is better able to capture sharp features in the data, and has the added advantage that the entire procedure is invariant to affine transformations. This property makes it ideal for many spatial applications.

Ruppert

Ruppert is a pioneer in the area of function estimation, contributing kernel, local polynomial and now spline methods, and we are pleased to have such an informative contribution to the discussion. Ruppert presents a penalized formulation that mimics smoothing splines. The so-called P-spline approach originally put forward by Eilers and Marx (1996) was designed as a short-cut to smoothing splines and illustrated for generalized linear models. Ruppert's version of this technique is really nothing more than ridge regression applied to a special truncated-power basis. As such, it inherits a certain degree of familiarity and represents an accessible smoothing method for people with a basic introductory regression course. In comparing Ruppert's method and that of Marx and Eilers, we do find one important difference: the basis. Ruppert makes use of the truncated power basis and (for cubic splines) he penalizes jumps in the 2nd derivative at each knot. That is, penalties of the form

$$\sum_{k=1}^{K} |\beta_{p+k}|^q \tag{109}$$

where $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_{K+p})$ is associated with the truncated power basis $\{1, x, \ldots, x^p, (x-t_1)_+^p, \ldots, (x-t_K)_+^p\}$. Eilers and Marx choose the numerically stable B-spline basis and derive a quadratic penalty by taking

differences of coefficients associated with adjacent basis elements (this short-cut is a rough approximation to the derivative-based penalties we applied in our paper, although we're not convinced of its reasonableness). Ruppert's use of values of q in (109) other than 2 is interesting. As he mentions, this amounts to the lasso of Tibshirani. In Hastie, Tibshirani and Friedman (2001), we find a close connection between forward stagewise fitting like boosting and the the lasso. These authors suggest that boosting is like combining all possible models (108) with a lasso penalty.

Finally, we comment on a couple of the P-spline improvements mentioned by Ruppert. First, reducing the number of knots to ease the computational burden of traditional cubic smoothing splines was examined in detail by O'Sullivan (1988) and is implemented in Splus. To help improve the spatial adaptivity, Ruppert suggests two approaches, the first involves a variable penalty that is again modeled as a spline. A form of this was also suggested in Wahba (1995) as part of a discussion for JRSS(b).

Wahba, Lin, and Leng

Over the years Wahba and her co-workers have pioneered the use of smoothing splines in function estimation. Wahba's influence has been mentioned several times throughout this rejoinder already. As for smoothing spline density estimation, we would have liked to see some of the plots of the estimates corresponding to the Table in this discussion, as the numbers reported suggest that the proposed method best compares to a less extreme version of our approach (vii), which, as was demonstrated in our Table 3 and Figure 2 smoothes away details too much.

When we read Wahba, Lin and Leng's discussion, we realized that the average in Table 2 is not a good summary of the table, and that for a particular method the geometric average would be a fairer comparison. Those are given in Table 2 below. This new summary table, even more than the one in the main paper, shows that we cannot just judge performance by MISE: except for (iv) and (vii) we have no way to choose. Summaries like the number of peaks need to play a role in deciding which version to use. See also the first example in Kooperberg and Stone (1991).

The smoothing spline ANOVA models for multivariate density estimation look quite promising. We are looking forward to seeing actual estimates. In particular we wonder how efficient the algorithm is if many λ parameters have to be estimated simultaneously. Also, we wonder how one would choose the terms in the ANOVA decomposition. Methods like MARS do this automatically (during passes of addition and deletion), but it does not appear to be an easy calculation in the smoothing spline world. Perhaps it would be best to combine the greedy search with the penalized fit as was done in Luo and Wahba (1997) for saturated models.

version	(i)	(ii)	(iii)	(iv)	(v)	(vi)	(vii)	(viii)	Wahba et al.	Bagging
geometric average	1.00	.69	.70	1.07	.76	.75	.95	.74	.79	.83

Table 103: Geometric average for the MISE ratios for the logspline simulation study reported in Table 2.

Additional References

Barron, A., Rissanen, J. and Yu, B. (1998). The Minimum Description Length principle in coding and modeling. *IEEE. Trans. Inform. Theory*, **44**, 2743–2760.

Barron, A., Schervish, M. and Wassserman, L. (1999). The consistence of posterior distributions in nonparametric problems. ANN STAT, **27**, 536–561.

Cleveland, W. S. and Fuentes, M. (1996) Multipanel conditioning: modeling data from designed experiments. Technical Memorandum 96-1, Bell Laboratories, Lucent Technologies.

Eilers, PHC, and Marx, BD. (1996) Flexible smoothing with B-splines and penalties. *STATISTICAL SCIENCE*, **11** 89-102.

George, E. and Foster, D. (2000). Calibration and empirical Bayes variable selection. *Biometrika*, **87** 731–747.

He, X., Ng, P. and Portnoy, S. (1998). Bivariate quantile smoothing splines. *Journal of the Royal Statistical Society, Series B*, **60**, 537–550.

Hansen, M. H. and Yu, B. (2001). Model Selection and the Principle of Minimum Description Length, *JASA*, **96**, No. 454, 746-774.

Holmes, C.C. and Mallick, B.K. (1998). Bayesian radial basis functions of variable dimension. *Neural Computation*, **10**, 1217-1233.

Holmes, C.C. and Denison, D.G.T. (1999). Bayesian wavelet analysis with a model complexity prior. Bayesian Statistics 6 : Oxford University Press.

Huang, J. Z. (1998). Projection estimation in multiple regression with application to functional ANOVA models. *Ann Stat*, **67**, 49–71.

Huang, J. Z. (2001). Concave extended linear modeling: a theoretical synthesis. *Statistica Scinica*, **11**, 173–197.

Koenker, R., Ng, P., and Portnoy, S. (1994). Quantile Smoothing splines. *Biometrika*, **81**, 673–680.

Kooperberg, C. and Stone, C. J. (2001). Logspline density estimation with free knot splines. Technical Report, FHCRC.

Kooperberg, C. and Stone , C. J. (2002). Confidence Intervals for Logspline Density Estimation. Proceedings of the MSRI workshop on Nonlinear Estimation and Classification, submitted.

Lindstrom, M. J. (1995). Self modeling with random scale and shift parameters and a free-knot spline shape function. *Statistics in Medicine,* **14**, 2009-2021.

O'Sullivan, F. (1988). Fast computation of fully automated log-density and log-hazard estimators. *SIAM J of Sci. Stat. Comput.*, **9** 363–379.

Rippa, S. (1990). Minimal roughness property of the Delaunay triangulation. *Computer Aided Geometric Design*, **7**, 489-497.

Rippa, S. (1992) Adaptive approximation by piecewise linear polynomials on triangulations of subsets of scattered data. *SIAM J. Scien. Statist. Comp.* **13** 1123–1141.

Shen, X. and Wasserman, L. (2001). Rates of convergence of posterior distributions. *Ann Stat*, **29**, 687–714.

Smith, A. F. M. and Spielgelhalter, D. J. (1980). Bayes factors and choice criteria for linear models. *Journal of the Royal Statistical Society* **Series B**, **42**, 213–220.

Stone, C. J. (1994). The use of polynomial splines and their tensor products in multivariate function estimation (with discussion). *Ann Stat*, **22**, 118–184.

Wahba, G. (1995) discussion of "Wavelet shrinkage or asymptopia". *JRSSB*, **57**, 337-369.

Wang, Y. and Wahba, G. (1995) Bootstrap Confidence Intervals for Smoothing Splines and their Comparison to Bayesian Confidence Intervals. *J. Statistical Computation and Simulation*, **51** 263-279.

Zellner, A. (1986). On assessing prior distributions and Bayesian regression analysis with g-prior distributions. *Bayesian Inference and Decision Techniques: Essays in Honor of Bruno de Finetti*, Eds. P.K. Goel and A. Zellner, 233–243. Amsterdam: North-Holland.

Zhao, L. (1993). Frequentist and Bayesian aspeacts of some nonparametric estimation. Ph.D. dissertation, Cornell Univeristy.

Zhao, L. (1998). A hierarchical Bayesian aproach in nonparametric function estimation. Technical report, Dept. of Statistics, Univ. of Pennsylvania.