Non-parametric Regression with Basis Selection from Multiple Libraries

Jeffrey C. Sklar, Junqing Wu, Wendy Meiring, and Yuedong Wang *

April 29, 2012

Abstract

New non-parametric regression procedures called BSML (Basis Selection from Multiple Libraries) are proposed in this paper for estimating a complex function by a linear combination of basis functions adaptively selected from multiple libraries. Different classes of basis functions are chosen to model various features of the function, e.g. truncated constants can model change points in the function, while polynomial spline representers may be used to model smooth components. The generalized cross-validation and covariance inflation criteria are used to balance goodness-of-fit and model complexity where the model complexity is estimated adaptively by either the generalized degrees of freedom or covariance penalty. The cross-validation method is also considered for model selection. Spatially adaptive regression and model selection in multivariate non-parametric regression will be used to illustrate the flexibility and efficiency of the BSML procedures. Extensive simulations show that the BSML procedures are more adaptive than some well-known existing non-parametric regression methods. Analyses of real data sets are used to illustrate the BSML procedures. This article has supplementary materials online.

---

*Jeffrey Sklar (email: jsklar@calpoly.edu) is Associate Professor, Statistics Department, California Polytechnic State University, San Luis Obispo, CA 93407. Junqing Wu (email: wjqpu@yahoo.com) is Marketplace Manager, Revenue at Microsoft Advertising, 11155 NE 8th St, Bravern 1/11145, Bellevue, WA 98004. Wendy Meiring (email: meiring@pstat.ucsb.edu) is Associate Professor, and Yuedong Wang (email: yuedong@pstat.ucsb.edu) is Professor, Department of Statistics and Applied Probability, University of California, Santa Barbara, California 93106. Yuedong Wang's research was supported by a grant from the National Science Foundation (DMS-0706886). Address for correspondence: Yuedong Wang, Department of Statistics and Applied Probability, University of California, Santa Barbara, California 93106-3110, USA.

Key words and phrases: Basis selection; Covariance inflation criterion; Covariance penalty; Generalized cross-validation; Generalized degrees of freedom; Non-parametric regression; Over-complete bases; Spatial adaptivity; Smoothing spline ANOVA.

1 Introduction

Consider the non-parametric regression model

$$y_i = f(x_i) + \epsilon_i, \quad \epsilon_i \overset{iid}{\sim} N(0, \sigma^2), \quad i = 1, \ldots, n, \tag{1}$$

where x_i's are design points on an arbitrary domain $\mathcal{X}$. Our goal is to estimate the unknown function f. For an introduction to this vast field, see Hastie and Tibshirani (1990), Wahba (1990), Green and Silverman (1994), Gu (2002), Fan and Gijbels (1996), Ruppert, Wand and Carroll (2003) and Wang (2011), among others.

In this article we assume that f can be approximated well by a linear combination of basis functions. The term "basis" is used loosely in this paper to represent a collection of functions which may be used to capture certain features (signals) in the function f, where this collection of functions may or may not constitute a basis of a functional space. Two key ingredients in non-parametric modeling are the choice of basis functions and a model selection procedure that selects basis functions and/or controls the balance between goodness-of-fit and model complexity. There are many choices for the family of basis functions including Fourier, spline, radial and wavelet bases to name only a few (see Chapter 5 of Hastie, Tibshirani and Friedman (2009)). Within each family, there are many choices for the type of basis, e.g. B-splines, truncated polynomials and reproducing kernel representers for the spline family. Within each type, there are many choices for the order of basis, such as linear, cubic or quintic for polynomial splines. Ideally, a basis should be chosen to achieve an excellent approximation with a small number of basis functions. However, basis function families/types/orders differ in their global and local adaptivity. Different families/types/orders may be best suited for different parts (or components) of the target function. A universally best basis does not exist.

In this article we propose a new non-parametric regression method called Basis Selection from Multiple Libraries (BSML), with two variants known as BSML-C (BSML with combined libraries) and BSML-S (BSML with separate libraries). Non-parametric regression is usually constructed using only one class of basis functions (e.g. cubic spline). Progress has been made to obtain sparse representations using multiple classes of basis functions in signal processing and function estimation (see for example Tropp (2004) and Gribonval and Nielsen (2003)).

To our knowledge, fusion among different classes of basis functions has not been studied extensively in the context of non-parametric regression. Substantial open research questions remain in this context, especially when fusing substantially different classes of basis functions. Our goal is to develop more adaptive non-parametric methods by data-driven selection of basis functions from different classes of basis functions, with the classes potentially differing in their smoothness and other properties.

Unlike the case of a single basis family where the representation usually is unique, finding the "best" approximation with multiple libraries and noisy data is a challenging problem. The naïve adoptions of greedy search (Luo and Wahba 1997) and basis pursuit (Chen, Donoho and Saunders 2001) may not work well. We develop the BSML adaptive basis selection methods based on adaptively estimated model complexities. These BSML-C and BSML-S methods are general in the sense that they can be applied to any generic libraries of candidate bases. We illustrate the flexibility and efficiency of the BSML procedures using two interesting applications: spatially adaptive regression, and model selection in multivariate non-parametric regression. Extensive simulations show that the BSML method is more adaptive when compared with Hybrid Adaptive Splines (HAS), Multivariate Adaptive Regression Splines (MARS), COmponent Selection and Selection Operator (COSSO) and L_1 norm based procedures.

The outline of the paper is as follows. Section 2 presents the general BSML procedure. Sections 3 and 4 present applications of the BSML methodology to spatial adaptive regression and multivariate regression. Section 5 provides a summary and further discussion.

2 Basis Selection from Multiple Libraries

The basic idea behind the BSML methodology is to explore the advantage of multiple libraries of basis functions using advanced model selection methods. The resulting BSML procedures advance both key ingredients in non-parametric modeling by incorporating more comprehensive basis functions as well as more adaptive model selection procedures.

2.1 Approximating the Function Using Multiple Libraries

A limitation of most existing adaptive non-parametric regression procedures is their use of a single class of basis functions. A truly flexible procedure should be able to choose from a variety of basis functions of different constructions to model complex features of a regression function.

Let $\mathcal{L}_0 = \{\phi_1, \ldots, \phi_m\}$, $\mathcal{L}_l = \{\psi_{l,1}, \ldots, \psi_{l,n_l}\}$, $l = 1, \ldots, L$, be $L+1$ libraries of basis functions. Each library consists of basis functions with similar properties. Otherwise, there is no limitation on the number of libraries and the elements of each library. There may be different families of basis functions for each $\mathcal{L}_l$: e.g. Fourier, wavelets or smoothing spline. There may be different types of basis functions for each $\mathcal{L}_l$: e.g. B-spline, truncated polynomial or reproducing kernel representer. There may be different orders of basis functions for each $\mathcal{L}_l$: e.g. linear, cubic or quintic. To allow flexibility of the procedure, we do not restrict the family, type, order, etc. of the basis functions in $\mathcal{L}_l$. In fact, as discussed earlier, libraries do not need to be bases. They could be any sets of functions that may be used to approximate the target function.

The idea of multiple libraries is close to the representation using overcomplete bases proposed in the wavelet and machine learning literature (Lewicki and Sejnowski 2000, Chen et al. 2001). Overcomplete representations have attracted a great deal of attention in Engineering. Many methods have been proposed to represent signals using overcomplete bases for achieving simultaneously the following goals: speed, sparsity, separation, resolution and stability (Coifman and Wickerhauser 1992, Mallat and Zhang 1993, Chen et al. 2001). Multiple libraries may contain a large set of diverse basis functions such that relatively few are required to represent any particular signal. Various methods have been proposed within the field of signal and image processing for learning sparse overcomplete representations, with methods based on the L_1 norm being especially popular (Mallat and Zhang 1993, Chen et al. 2001, Lewicki and Sejnowski 2000).

Unlike the case of a single basis where the representation is unique, finding the "best" approximation in multiple libraries with noisy data is a challenging problem. Simple extensions of HAS (Luo and Wahba 1997), matching pursuit (Mallat and Zhang 1993) and L_1 norm procedures (Chen et al. 2001, Efron, Hastie, Johnstone and Tibshirani 2004) to the case of multiple libraries do not work well (see Section 2 of the Supplement for an illustration).

2.2 Model Selection

Basis functions in $\mathcal{L}_0$ are automatically entered into the model (note that $\mathcal{L}_0$ could be an empty set). Up to a total pre-specified number M (including those in $\mathcal{L}_0$), the procedure then selects basis functions from $\mathcal{O} = \cup_{l=1}^{L} \mathcal{L}_l$ one at a time according to a criterion. For the BSML-C procedure, the criterion is to maximize the reduction in the residual sum of squares (RSS). For the BSML-S procedure, the criterion is more sophisticated, and is presented in Section 2.4. Within each BSML procedure, denote the sequentially selected

basis functions as ϕ_k for $k = m+1, \cdots, M$. Let $\mathcal{B}_k = \{\phi_1, \cdots, \phi_k\}$ for $k = m, \cdots, M$ where $\mathcal{B}_m = \mathcal{L}_0$. For simplicity, we write "model $\mathcal{B}_k$" for "a linear combination of the basis functions in $\mathcal{B}_k$". We need to develop model selection criteria to select the "best" model among $\{\mathcal{B}_k,\ k = m, \ldots, M\}$. In the following we adapt the Generalized Cross Validation (GCV), Covariance Inflation Criterion (CIC), and K-fold Cross-Validation (CV) criteria for model selection in the BSML-C procedure.

Let $\boldsymbol{y} = (y_1, \cdots, y_n)^T$ and $\boldsymbol{f} = (f(x_1), \ldots, f(x_n))^T$. For $k = m, \cdots, M$, let $\hat{f}_k$ be the estimate of the function f based on $\mathcal{B}_k$ and $\hat{\boldsymbol{f}}_k = (\hat{f}_k(x_1), \cdots, \hat{f}_k(x_n))^T$. Denote $\mathcal{M}_k$ as the modeling procedure leading to $\hat{\boldsymbol{f}}_k$. Note that the modeling procedure $\mathcal{M}_k$ includes both basis functions selection and estimation. Define the mean squared error (MSE) based on $\mathcal{M}_k$ as

$$\mathrm{MSE}(k) = \frac{1}{n}\mathrm{E}(||\hat{\boldsymbol{f}}_k - \boldsymbol{f}||^2). \tag{2}$$

It is easy to check that

$$\begin{aligned} \frac{1}{n}\mathrm{E}\{||\hat{\boldsymbol{f}}_k - \boldsymbol{y}||^2\} &= \frac{1}{n}E\{||\boldsymbol{y} - \boldsymbol{f}||^2 + 2(\boldsymbol{y} - \boldsymbol{f})^T(\boldsymbol{f} - \hat{\boldsymbol{f}}_k) + ||\boldsymbol{f} - \hat{\boldsymbol{f}}_k||^2\} \\ &= \sigma^2 - \frac{2}{n}C(\mathcal{M}_k) + \mathrm{MSE}(k), \end{aligned}$$

where

$$C(\mathcal{M}_k) = \mathrm{E}(\boldsymbol{y} - \boldsymbol{f})^T(\hat{\boldsymbol{f}}_k - \boldsymbol{f}) = \sum_{i=1}^{n} \mathrm{cov}\{\hat{f}_k(x_i), y_i\} \tag{3}$$

is the covariance penalty (Tibshirani and Knight 1999, Efron 2004). Then we have the covariance inflation criterion (CIC) (Tibshirani and Knight 1999, Shen and Huang 2006)

$$\mathrm{CIC}(k) = \frac{1}{n}\mathrm{RSS}(\mathcal{B}_k) + \frac{2}{n}C(\mathcal{M}_k) \tag{4}$$

as an unbiased estimate of $(\mathrm{MSE}(k) + \sigma^2)$, where $\mathrm{RSS}(\mathcal{B}_k) = ||\hat{\boldsymbol{f}}_k - \boldsymbol{y}||^2$ is RSS for model $\mathcal{B}_k$.

Ye (1998) introduced the generalized degrees of freedom (GDF) as a measure of model complexity for the whole procedure:

$$D(\mathcal{M}_k) = \sum_{i=1}^{n} \frac{\partial \mathrm{E}_{\mathbf{f}}(\hat{f}_{ik})}{\partial f_i} = \frac{1}{\sigma^2}C(\mathcal{M}_k), \tag{5}$$

where $\hat{f}_{ik} = \hat{f}_k(x_i)$ and $f_i = f(x_i)$. GDF extends the standard degrees of freedom to general modeling procedures, and can be viewed as the sum of the sensitivities of the fitted values

to a small change in the response. A highly flexible modeling procedure will have a large GDF and covariance penalty because the fitted values will be close to the observed values.

Based on (5), the CIC (4) can be written as

$$\mathrm{CIC}(k) = \frac{1}{n}\mathrm{RSS}(\mathcal{B}_k) + \frac{2\sigma^2}{n}D(\mathcal{M}_k). \tag{6}$$

Criterion (6) requires an estimate of σ^2 when it is unknown. It is preferable to use an estimator of σ^2 that does not require an estimate of f. A difference-based estimator such as the Rice estimator or a regression estimator may be used (Tong and Wang 2005). We use the Monte Carlo algorithms suggested in Ye (1998), Tibshirani and Knight (1999), Efron (2004) and Shen and Huang (2006) to obtain estimates of $D(\mathcal{M})$ and $C(\mathcal{M})$, say $\hat{D}(\mathcal{M})$ and $\hat{C}(\mathcal{M})$. Section 4.2 of the Supplement describes these Monte Carlo algorithms and Section 3 of the Supplement discusses the estimation of σ^2.

The generalized cross-validation (GCV) criterion is defined as (Luo and Wahba 1997)

$$\mathrm{GCV}(k) = \frac{\mathrm{RSS}(\mathcal{B}_k)}{\{n - c(\mathcal{M}_k)\}^2}, \tag{7}$$

where $c(\mathcal{M}_k)$ is a measure of model complexity for the modeling procedure $\mathcal{M}_k$. To correct for the bias incurred by adaptive model selection, Luo and Wahba (1997) used $c(\mathcal{M}_k) = m + (k - m) \times \mathrm{IDF}$ in the HAS procedure where the complexity of each selected basis function was inflated by a factor of IDF (Inflated Degrees of Freedom). Luo and Wahba (1997) suggested that IDF = 1.2 is a good choice for their HAS procedure. The same GCV criterion was used in MARS (Friedman 1991) with IDF = 3, TURBO (Friedman and Silverman 1989) with IDF = 2 and SARS (Zhou and Shen 2001) with IDF = 3. The choices of the IDF in these procedures are ad hoc and this inflation factor is fixed throughout the selection process. Empirical evidence (see Section 1 of the Supplement) reveals that appropriate inflation factors for adaptively selecting basis functions in search procedures such as HAS are not constant, but depend on many factors including the true function, the type of basis functions, the signal-to-noise ratio, and the basis functions that have been selected. It is not surprising that Friedman (1991) and Luo and Wahba (1997) recommended different IDFs because they used quite different bases: truncated polynomials in Friedman (1991) and cubic spline representers in Luo and Wahba (1997).

Since the true function is never known and there is no clear rule for deciding the IDF, we propose to estimate the GDF or covariance penalty at each step of the selection procedure and incorporate it into the GCV criterion. Specifically, we set $c(\mathcal{M}_k) = D(\mathcal{M}_k)$ in the GCV criterion (7) for all analyses in this paper. Alternatively, we could set $c(\mathcal{M}_k) = C(\mathcal{M}_k)/\sigma^2$ where σ^2 is estimated by a difference-based or a regression estimator.

K-fold cross-validation (CV) may also be used to select the best model among $\{\mathcal{B}_k,\ k = m, \cdots, M\}$. Divide the original sample into K subsamples at random. For a fixed $1 \le j \le K$, apply the forward selection procedure to all the data except the jth subsample. Denote the resulting estimates of the function as $\hat{f}_k^{(j)}$ for $k = m, \ldots, M$. Let $\boldsymbol{y}^{(j)}$ be the vector of response variables in the jth subsample and $\hat{\boldsymbol{f}}_k^{(j)}$ be the vector of the function $\hat{f}_k^{(j)}$ evaluated at the design points in the jth subsample. Then the K-fold CV estimate of the MSE is

$$\mathrm{CV}(k) = \frac{1}{n}\sum_{j=1}^{K} ||\hat{\boldsymbol{f}}_k^{(j)} - \boldsymbol{y}^{(j)}||^2, \qquad \text{for } k = m, \ldots, M. \tag{8}$$

2.3 The BSML-C Procedure

Combining steps together, we have:

The BSML-C procedure

1. *Initialization:* set $\mathcal{B}_m = \mathcal{L}_0$ and let M be an upper limit on the total number of basis functions to be selected (including all bases in $\mathcal{L}_0$).

2. For $k = m+1, \ldots, M$, do:

   (a) *Forward Selection:* find ϕ_k from the remaining basis functions not yet selected in $\mathcal{O}$ that maximizes the reduction in the residual sum of squares

   $$\phi_k = \underset{\psi \in \mathcal{O} \cap \mathcal{B}_{k-1}^c}{\operatorname{argmax}} \left\{\mathrm{RSS}(\mathcal{B}_{k-1}) - \mathrm{RSS}(\mathcal{B}_{k-1} \cup \{\psi\})\right\}.$$

   (b) *Update:* $\mathcal{B}_k = \mathcal{B}_{k-1} \cup \{\phi_k\}$.

   (c) *Estimate GDF and covariance penalty as needed:* calculate $\hat{D}(\mathcal{M}_k)$ and $\hat{C}(\mathcal{M}_k)$.

3. *Elimination:* choose k^*, $m \le k^* \le M$, as the minimizer of one of the following criteria: the CIC in (4), the GCV criterion in (7) and the CV criterion in (8).

4. *Final model:* fit a standard or ridge regression model of $\boldsymbol{y}$ on the final selected basis functions $\mathcal{B}_{k^*}$.

BSML-C does not distinguish basis functions from different libraries in the selection process. Often, different libraries have different degrees of adaptivity and complexity. It may be beneficial to consider libraries separately in the selection process, and for the model selection criterion to incorporate a different measure of complexity for each library. We now describe an alternative BSML procedure, BSML-S, that treats different libraries separately.

2.4 The BSML-S Procedure

Let $\mathcal{B}_{k-1}$ be the $k-1$ basis functions selected at step $k-1$. At step k, for each library $\mathcal{L}_l$, $1 \leq l \leq L$, we first find a basis function, $\psi_{l,j_l}^{(k)}$, from the remaining basis functions not yet selected in $\mathcal{L}_l$ to maximize the reduction in the RSS

$$\psi_{l,j_l}^{(k)} = \underset{\psi \in \mathcal{L}_l \cap \mathcal{B}_{k-1}^c}{\operatorname{argmax}} \left\{ \mathrm{RSS}(\mathcal{B}_{k-1}) - \mathrm{RSS}(\mathcal{B}_{k-1} \cup \{\psi\}) \right\}. \tag{9}$$

We then need to select a basis function, ϕ_k, to be included in the model at step k, from the collection of L candidate basis functions, $\Psi_k = \left\{ \psi_{1,j_1}^{(k)}, \ldots, \psi_{L,j_L}^{(k)} \right\}$.

To allow basis functions in different libraries to compete on an equal footing, in addition to the basis selection cost within each library $\hat{D}(\mathcal{M}_{k,l})$ where $\mathcal{M}_{k,l}$ is the modeling procedure that includes both the selection of $\mathcal{B}_{k-1} \cup \{\psi_{l,j_l}^{(k)}\}$ and the estimation based on $\mathcal{B}_{k-1} \cup \{\psi_{l,j_l}^{(k)}\}$, we need to account for differences in complexity among libraries. Again, we estimate the library complexity using GDF. Let $\mathcal{M}_k^{(l)}$ be the modeling procedure that, starting with $\mathcal{L}_0$, selects $k-m$ additional basis functions one at a time from library $\mathcal{L}_l$ and estimates the function f based on k selected basis functions. Let $\hat{D}(\mathcal{M}_k^{(l)})$ be the estimated GDF of $\mathcal{M}_k^{(l)}$. We estimate the library cost at step k for library l by

$$\hat{A}_k(\mathcal{L}_l) = \frac{\hat{D}(\mathcal{M}_k^{(l)}) - m}{k-m}, \tag{10}$$

which is an estimated IDF for the k^{th} basis function from library $\mathcal{L}_l$, if all $k-m$ basis functions had been adaptively selected from $\mathcal{L}_l$. We select all $k-m$ basis functions from library $\mathcal{L}_l$ so that $\hat{A}_k(\mathcal{L}_l)$ represents an average library cost.

We choose the kth basis function $\phi_k \in \Psi_k$ to minimize the doubly penalized criterion

$$\mathrm{DPC}(\phi_k) = \mathrm{RSS}(\mathcal{B}_{k-1} \cup \{\phi_k\}) + c_1 \sigma^2 \hat{D}(\mathcal{M}_{k,l}) + c_2 \sigma^2 \hat{A}_k(\mathcal{L}_l), \tag{11}$$

where c_1 and c_2 are constants. The first two components in (11) constitute the final prediction error criterion (Akaike 1970) which includes the commonly used AIC and BIC criteria. Similar to the idea of adding a penalty term to account for selection bias due to basis selection in a single library, we add the last term in (11) to account for selection bias for selection between libraries. The need for an extra penalty term has been recognized theoretically by Yang (1999), Lebarbier (2005) and Picard, Robin, Lavielle, Vaisse and Daudin (2005). The DPC criterion (11) is similar to the ABC criterion in Yang (1999) where ways to compute model space (library) complexity were discussed. However, none of the existing theoretical criteria can be applied directly in our setting, since they depend on unknown constants. We

have found in our simulations that $c_1 = \ln(n)$ and $c_2 = 2$ work well, so we use these values in our simulations and examples. A relatively larger range of combinations of c_1 and c_2 also provide good estimates (see Section 5.2 of the Supplement). Again, a difference-based or regression estimator of σ^2 may be used in (11).

The K-fold CV method from Section 2.2 may also be used to select ϕ_k in Ψ_k. The only difference is that the candidate models are now $\{\mathcal{B}_{k-1} \cup \{\psi_{l,j_l}^{(k)}\},\ l = 1, \ldots, L\}$.

To use the GCV criterion to decide the final number of basis functions k^*, we need to estimate the GDF, $\hat{D}(\mathcal{M}_k)$, at each step k. However, calculation of $\hat{D}(\mathcal{M}_k)$ is computationally intensive for the BSML-S procedure. We define the cost for selecting $\psi_{l,j_l}^{(k)}$ conditional on $\mathcal{B}_{k-1}$ as

$$\hat{D}\left(\psi_{l,j_l}^{(k)} | \mathcal{B}_{k-1}\right) = \max\left\{\ \hat{D}(\mathcal{M}_{k,l}) - (k-1),\ 1\ \right\}.$$

Small scale simulations indicate that $m + \sum_{h=m+1}^{k} \hat{D}(\psi_{l_h,j_{l_h}}^{(h)} | \mathcal{B}_{h-1})$ provides a reasonable approximation for $\hat{D}(\mathcal{M}_k)$. Therefore we minimize the following modified generalized cross-validation criterion to select k^*

$$\mathrm{GCV}(k) = \frac{\mathrm{RSS}(\mathcal{B}_k)}{\left\{n - \left(m + \sum_{h=m+1}^{k} \hat{D}(\psi_{l_h,j_{l_h}}^{(h)} | \mathcal{B}_{h-1})\right)\right\}^2}. \tag{12}$$

By compiling these stages, we have:

The BSML-S procedure

1. *Initialization:* set $\mathcal{B}_m = \mathcal{L}_0$ and let M be an upper limit on the number of basis functions to be selected (including those in $\mathcal{L}_0$).

2. *Forward selection*: for $k = m+1, \ldots, M$, do

   (a) *Select within each library:* for $l = 1, \ldots, L$ do

      i. Select $\psi_{l,j_l}^{(k)} \in \mathcal{L}_l$ according to (9).

      ii. *Estimate GDF and conditional cost as needed*: compute $\hat{D}(\mathcal{M}_{k,l})$ and $\hat{D}\left(\psi_{l,j_l}^{(k)} | \mathcal{B}_{k-1}\right)$.

   (b) *Select between libraries:* select $\phi_k \in \Psi_k$ to minimize (11) or the K-fold CV criterion.

   (c) *Update:* $\mathcal{B}_k = \mathcal{B}_{k-1} \cup \{\phi_k\}$.

3. *Elimination*: choose k^*, $m \le k^* \le M$, as the minimizer of one of the following criteria: the CIC in (4), the GCV criterion in (12) and the CV criterion in (8).

4. *Final model:* fit a standard or ridge regression model of $\boldsymbol{y}$ on the final selected basis functions $\mathcal{B}_{k^*}$.

2.5 Bootstrap Confidence Intervals

Consider model (1) and assume that f can be represented as $f(x) = \sum_{l=0}^{L} g_l(x)$ where $g_l \in \text{span}\{\mathcal{L}_l\}$ and $\text{span}\{\mathcal{L}_l\}$ represents the linear space spanned by basis functions in $\mathcal{L}_l$. Let L_0 be any well-defined functional. We construct confidence intervals for L_0 applied to the following form of linear combinations of the $L+1$ components of f,

$$f_{\boldsymbol{\gamma}} = \sum_{l=0}^{L} \gamma_l g_l, \tag{13}$$

where $\gamma_l = 1$ when g_l is to be included and $\gamma_l = 0$ otherwise. A confidence interval for $f_{\boldsymbol{\gamma}}$ evaluated at a particular point, say x, corresponds to the special case when L_0 is the evaluational functional $L_0 g_l = g_l(x)$ for each $l \in \{0, 1, \ldots, L\}$.

Let $\hat{f}$ and $\hat{\sigma}^2$ be pilot estimates of f and σ^2 respectively. Let

$$y^*_{i,b} = \hat{f}(x_i) + \epsilon^*_{i,b}, \quad i = 1, \ldots, n;\ b = 1, \ldots, B$$

be B bootstrap samples where $\epsilon^*_{i,b} \overset{iid}{\sim} \text{N}(0, \hat{\sigma}^2)$. Apply the BSML-C or BSML-S procedure to the bth bootstrap sample $\{y^*_{i,b},\ i = 1, \ldots, n\}$ and denote $\hat{f}^*_{\boldsymbol{\gamma},b}$ as the estimate of $f_{\boldsymbol{\gamma}}$ in (13). The $100(1-\alpha)\%$ percentile bootstrap confidence interval of $L_0 f_{\boldsymbol{\gamma}}$ is

$$(L_0 \hat{f}_{\boldsymbol{\gamma},L}, L_0 \hat{f}_{\boldsymbol{\gamma},U}),$$

where $L_0 \hat{f}_{\boldsymbol{\gamma},L}$ and $L_0 \hat{f}_{\boldsymbol{\gamma},U}$ are the lower and upper $\alpha/2$ quantiles of the B bootstrap estimates of $L_0 f_{\boldsymbol{\gamma}}$, i.e., the specified quantiles of $\{L_0 \hat{f}^*_{\boldsymbol{\gamma},b},\ b = 1, \ldots, B\}$.

In the smoothing spline literature, it is well-known that, due to nonuniform bias, the bootstrap and Bayesian confidence intervals have an across-the-function coverage property which is weaker than the pointwise coverage property (Wang and Wahba 1995, Wang 2011). Efforts have been made to construct confidence intervals with more uniform pointwise coverage (Cummins, Filloon and Nychka 2001). One approach is to reduce bias by a slight undersmoothing (Hall 1992). We have found that, in general, the cross-validation method tends to overfit. Therefore, we apply the BSML-C or BSML-S procedure with cross-validation selection of the final model to derive a undersmoothed pilot fit $\hat{f}$. Since the true function is known

in the generation of bootstrap samples (namely $\hat{f}$), instead of using the CIC, GCV or CV criterion in Step 3, we use the average squared error $\text{ASE}(k,b) = \sum_{i=1}^{n}(\hat{f}^*_{b,k}(x_i) - \hat{f}(x_i))^2/n$ to select the final model for each bootstrap sample where $\hat{f}^*_{b,k}$ is the estimate of f in the kth iteration of the BSML-C or BSML-S procedure based on the bth bootstrap sample. Extensive simulations (some of them are shown in Section 5.3 of the Supplement) indicate that the percentile bootstrap confidence intervals based on the BSML procedures have more uniform pointwise coverage than the smoothing spline Bayesian confidence intervals. For all percentile bootstrap confidence intervals in presented simulations and data analyses, we used 10-fold CV to generate the pilot fit and ASE to select the optimal k within each bootstrap.

2.6 Computation and the BSML package

We have developed a user-friendly R package called `bsml` which implements the HAS, BSML-C and BSML-S procedures. The `bsml` package is available at `http://cran.r-project.org` (Wu, Sklar, Wang and Meiring 2011). R code for an example in Section 3.3 is given in Section 4.3 of the Supplement, together with a brief description of the forward selection process and Monte Carlo based estimation of $D(\mathcal{M})$ and $C(\mathcal{M})$. Also see Section 2.2, Sklar (2003), and Wu (2011).

3 Spatially Adaptive Regression

3.1 Existing Methods

In this section we consider the estimation of spatially inhomogeneous curves defined on $x \in [0,1]$. A polynomial smoothing spline model of order-m assumes that f in model (1) belongs to the reproducing kernel Hilbert space (RKHS) (Wahba 1990)

$$W_m[0,1] = \left\{ f : f, f', \ldots, f^{(m-1)} \text{ absolutely continuous}, \int_0^1 \left(f^{(m)}(x)\right)^2 dx < \infty \right\}. \quad (14)$$

For a fixed smoothing parameter λ, the corresponding spline estimate of f, $\hat{f}_\lambda$, is the minimizer of the penalized least squares (PLS) criterion

$$\sum_{i=1}^{n}(y_i - f(x_i))^2 + \lambda \int_0^1 \left(f^{(m)}(x)\right)^2 dx. \quad (15)$$

The smoothing parameter λ controls the smoothness of the spline fit. Data-based methods such as GCV are used frequently to choose λ. The space $W_m[0,1]$ can be decomposed into $W_m[0,1] = \mathcal{H}_0 \oplus \mathcal{H}_1$ where $\mathcal{H}_0$ contains polynomials that are not penalized.

Specifically, let $k_r(x)$ for $r = 0, 1, 2, \ldots$ be scaled rth Bernoulli polynomials defined recursively by $k_0(x) = 1$, $k_r'(x) = r k_{r-1}(x)$ and $\int_0^1 k_r(x)dx = 0$. Then $\mathcal{H}_0 = \text{span}\,(\mathcal{R}_{0,m})$ with $\mathcal{R}_{0,m} = \{k_0(x), k_1(x), \ldots, k_{m-1}(x)\}$. Also $\mathcal{H}_1$ is a RKHS with reproducing kernel $R_m(s,t) = k_m(s)k_m(t) + (-1)^{m-1}k_{2m}(s-t)$. Let $\mathcal{R}_{1,m} = \{R_m(x_1, x), \ldots, R_m(x_n, x)\}$. Then the spline estimate $\hat{f}_\lambda \in \text{span}\,\{\mathcal{R}_{0,m} \cup \mathcal{R}_{1,m}\}$ (Wahba 1990).

The PLS criterion (15) relies on a single global smoothing parameter, λ, to control the trade-off between the goodness-of-fit and the smoothness of the estimated function over the entire domain $[0, 1]$. Thus an implicit assumption is that f is smooth with relatively homogeneous curvatures over the entire domain. If the true function is spatially inhomogeneous, then spline estimates tend to over-smooth in regions where f is rough and under-smooth in regions where f is smooth. To make the spline estimate spatially adaptive, Luo and Wahba (1997) proposed the HAS procedure, which is a special case of the BSML-C procedure with $L = 1$, $\mathcal{L}_0 = \mathcal{R}_{0,2}$, and $\mathcal{L}_1 = \mathcal{R}_{1,2}$. HAS uses GCV criterion (7) with $c(\mathcal{M}_j) = 2+(j-2)\times\text{IDF}$ with fixed IDF = 1.2 at the elimination step. In several examples, Luo and Wahba (1997) show HAS to be spatially adaptive with comparable or better performance than wavelets.

Spatially adaptive methods have been actively researched. Other spline-based spatially adaptive methods include MARS (Friedman 1991), adaptive knots selection schemes (Zhou and Shen 2001, Miyata and Shen 2003), variable smoothness penalties (Abramovich and Steinberg 1996, Ruppert and Carroll 2000, Liu and Guo 2010), and Bayesian methods (Denison, Mallick and Smith 1998, DiMatteo, Genovese and Kass 2001, Dias and Gamerman 2002). Beyond the spline literature, spatially adaptive methods have also been developed for local polynomials (Fan and Gijbels 1995) and wavelets (Wang 1995). All methods are limited by using a single family of basis functions to approximate the regression function. Even though a single family, often of infinite dimension, may eventually be able to capture a spatially heterogeneous signal in the data, the methods based on a single basis are limited in their adaptivity with finite samples.

3.2 Simulations

We simulate from model (1), for each of six test functions, f, listed in Table 2, and displayed in Figure 1. Functions LW6 and LW7 are examples 6 and 7 in Luo and Wahba (1997). We also use those authors' Heavisine function scaling, and their definition of signal to noise ratio (SNR). That is, the SNR is $SD(f)/\sigma$ where $SD(f)$ is the standard deviation of the function $f(x)$ across values of x. We ran simulations for eight values of σ, corresponding to a regular grid of SNR values, SNR $\in \{1, 2, 3, 4, 5, 6, 7, 8\}$, for each function f in Table 2.

For each sample size n, we use a regular grid of design points $\{x_i = i/n : i = 1, \ldots, n\}$. We present detailed results for each f for $n = 256$, but only illustrate sample size changes for the Heavisine function due to space.

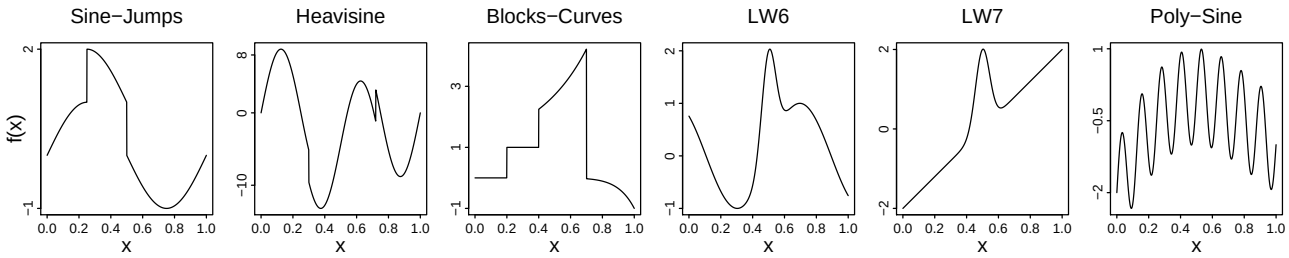

Figure 1: Six functions used in spatially adaptive regression simulations.

Table 1 defines several families of basis functions we use, including truncated polynomial, polynomial spline, and Fourier bases. The family ($\mathcal{P}_2$) of basis functions derived from periodic spline reproducing kernels (RK) of order-2 are defined based on the space (Wahba 1990)

$$W_2(per) = \left\{ g : g^{(j)} \text{ abs. cont.}, \; g^{(j)}(0) = g^{(j)}(1), \; j = 0, 1, \; \int_0^1 \left(g^{(2)}(t)\right)^2 dt < \infty \right\},$$

with RK

$$R_{per,2}(s,t) = \sum_{v=1}^{\infty} \frac{2}{(2\pi v)^4} \cos 2\pi v(s-t). \tag{16}$$

Each of the reproducing kernel (cubic or periodic) and truncated polynomial families begins with one basis function corresponding to each of the design points x_i, $i = 1, \ldots, n$.

Note that, when $L = 1$, BSML-C is a simple extension of the HAS procedure which will be referred to as the BSML1 procedure in our simulations (Sklar, Meiring and Wang 2006). We compare the performance of the BSML-S and BSML-C methods to that of HAS, BSML1, and an L_1 norm method. The L_1 norm method used here is a direct extension of basis pursuit (Chen et al. 2001) and LARS (Efron et al. 2004) procedures. For general $\mathcal{A}$ containing $N(\mathcal{A})$ basis functions, this procedure represents $f(x) = \sum_{\psi_j \in \mathcal{A}} \beta_j \psi_j(x)$ and estimates coefficients by minimizing

$$\frac{1}{n} \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda \sum_{j=1}^{N(\mathcal{A})} |\beta_j|. \tag{17}$$

To obtain L_1-norm lasso estimates with sets $\mathcal{A}$ listed in Table 2, we used the R function `lars` with default options (Hastie and Efron 2011). For each function, the BSML-S, BSML-C and

Table 1: Basis functions notation.

$\mathcal{U}_0$	$\{1\}$, constant functions
$\mathcal{U}_1$	$\{1, x\}$, constant and linear functions
$\mathcal{C}_m$	$\mathcal{R}_{(1,m)}$, polynomial spline representers as defined in Section 3.1
$\mathcal{P}_2$	$\{R_{per,2}(x_i, x)\}$, periodic smoothing spline representers of order-2 given in (16)
$\mathcal{T}_m$	$\left\{(x - x_1)_+^m, (x - x_2)_+^m, \ldots, (x - x_n)_+^m\right\}$, truncated polynomials
$\mathcal{F}$	$\{\sin(16\pi kx), \cos(16\pi kx) : k = 1, 2, 3, \ldots, 25\}$, Fourier basis functions

Table 2: Six functions $f(x)$ used in simulations, and three candidate basis function collections labelled $\{\mathcal{G}_i : i = 0, 1, 2\}$ for reference in Table 3.

	True $f(x)$ in (1) with $1_A(x) = 1$ if $x \in A$, zero otherwise.	$SD(f)$	Basis families		
			$\mathcal{G}_0$	$\mathcal{G}_1$	$\mathcal{G}_2$
Sine-Jumps	$\sin(2\pi x) - 1_{(0.5,1]}(x) + 1_{(0.25,1]}(x)$	1.003	$\mathcal{U}_0$	$\mathcal{P}_2$	$\mathcal{T}_0$
Heavisine	$2.2\left[4\sin(4\pi x) - \text{sign}(x - 0.3) - \text{sign}(0.72 - x)\right]$	6.534	$\mathcal{U}_0$	$\mathcal{P}_2$	$\mathcal{T}_0$
Blocks-Curves	$1_{[.2,.4)}(x) + \exp\left[(x + 0.5)^2\right] 1_{[.4,.7)}(x) - x^{10} 1_{[.7,1]}(x)$	1.458	$\mathcal{U}_1$	$\mathcal{C}_2$	$\mathcal{T}_0$
LW6	$\sin\left[2(4x - 2)\right] + 2\exp\left[-256(x - .5)^2\right]$	0.838	$\mathcal{U}_1$	$\mathcal{C}_2$	$\mathcal{T}_2$
LW7	$(4x - 2) + 2\exp\left[-256(x - .5)^2\right]$	1.264	$\mathcal{U}_1$	$\mathcal{C}_2$	$\mathcal{T}_2$
Poly-Sine	$\sin(16\pi x) - 8(x - .5)^2 + 8(x - .5)^3 1_{(0.5,1]}(x)$	0.840	$\mathcal{U}_1$	$\mathcal{F}$	$\mathcal{T}_2$

Table 3: Basis families for different estimation methods, with notation as in Table 2.

	HAS	BSML1	LARS	BSML-C	BSML-S
$\mathcal{L}_0$	$\mathcal{G}_0$	$\mathcal{G}_0$		$\mathcal{G}_0$	$\mathcal{G}_0$
$\mathcal{L}_1$	$\mathcal{G}_1$	$\mathcal{G}_1$		$\mathcal{G}_1 \cup \mathcal{G}_2$	$\mathcal{G}_1$
$\mathcal{L}_2$					$\mathcal{G}_2$
$\mathcal{A}$			$\bigcup_{i=0}^{2} \mathcal{G}_i$		

L_1 norm (LARS) simulations use basis functions from the three libraries listed in Table 3. HAS and BSML1 use two libraries, $\mathcal{L}_0$ and $\mathcal{L}_1$. GDF estimation is by the algorithm of Ye (1998), using 100 perturbations (see Section 4.2 of the Supplement). The performance of each method was measured by the mean squared error (MSE):

$$\text{MSE}(\hat{f}) = \frac{1}{n}\sum_{i=1}^{n}[f(x_i) - \hat{f}(x_i)]^2.$$

Figures 2 and 3 present results based on 100 simulations for $n = 256$. Each row corresponds to one of the six functions in Table 2. The left panel of each row illustrates the relative performance of the five estimation methods of Table 3 across different SNR values. The vertical axis in the left panel scales the median MSE at each SNR by dividing by σ^2 for clarity, since the MSE decreases as the SNR increases (i.e., as σ decreases). The symbols used are: $\triangle$ for HAS, $+$ for BSML1, $\times$ for LARS, ∇ for BSML-C, and $\square$ for BSML-S. We used $M = 100$ for HAS; and $M = 50$ for BSML1, BSML-C and BSML-S throughout. In the center panel of each row, boxplots are presented to compare MSE values across estimation methods, when estimating f based on $n = 256$ observations, for each of 100 simulated data sets with SNR=5. The right panel presents boxplots for the corresponding total number of basis functions selected by each method.

The BSML-S method has the smallest, or close to the smallest, MSE under all settings in Figures 2 and 3. BSML-S also achieves this often with the fewest number of basis functions. The MSE reductions of BSML-S compared with the HAS method are substantial for all functions except for the LW6 and LW7 functions, which are relatively simple and smooth compared to the other four functions. Both LW6 and LW7 functions can be well approximated by a small number of basis functions from one class of basis, so HAS and BSML1 perform comparatively well. Nevertheless, for the LW6 and LW7 functions, the BSML-S did have slightly smaller median MSE with smaller median number of basis. The BSML-S method is also more stable than all other methods.

The BSML1 method performs slightly better than the HAS method with slightly fewer basis functions. Neither HAS nor BSML1 performs well for the Poly-Sine function, which lies outside the span of the Fourier basis functions. The L_1 norm based method performs well for the Sine-Jumps and Block-Curves functions, but relatively poorly for the other functions. It tends to select a large number of basis functions and therefore does not work well in general with multiple libraries with different types of basis functions. We also ran a naive extension of HAS with two non-null libraries and a fixed IDF of 1.2 (results not shown), but this consistently overfitted, performing worse than HAS with 1 non-null library (results

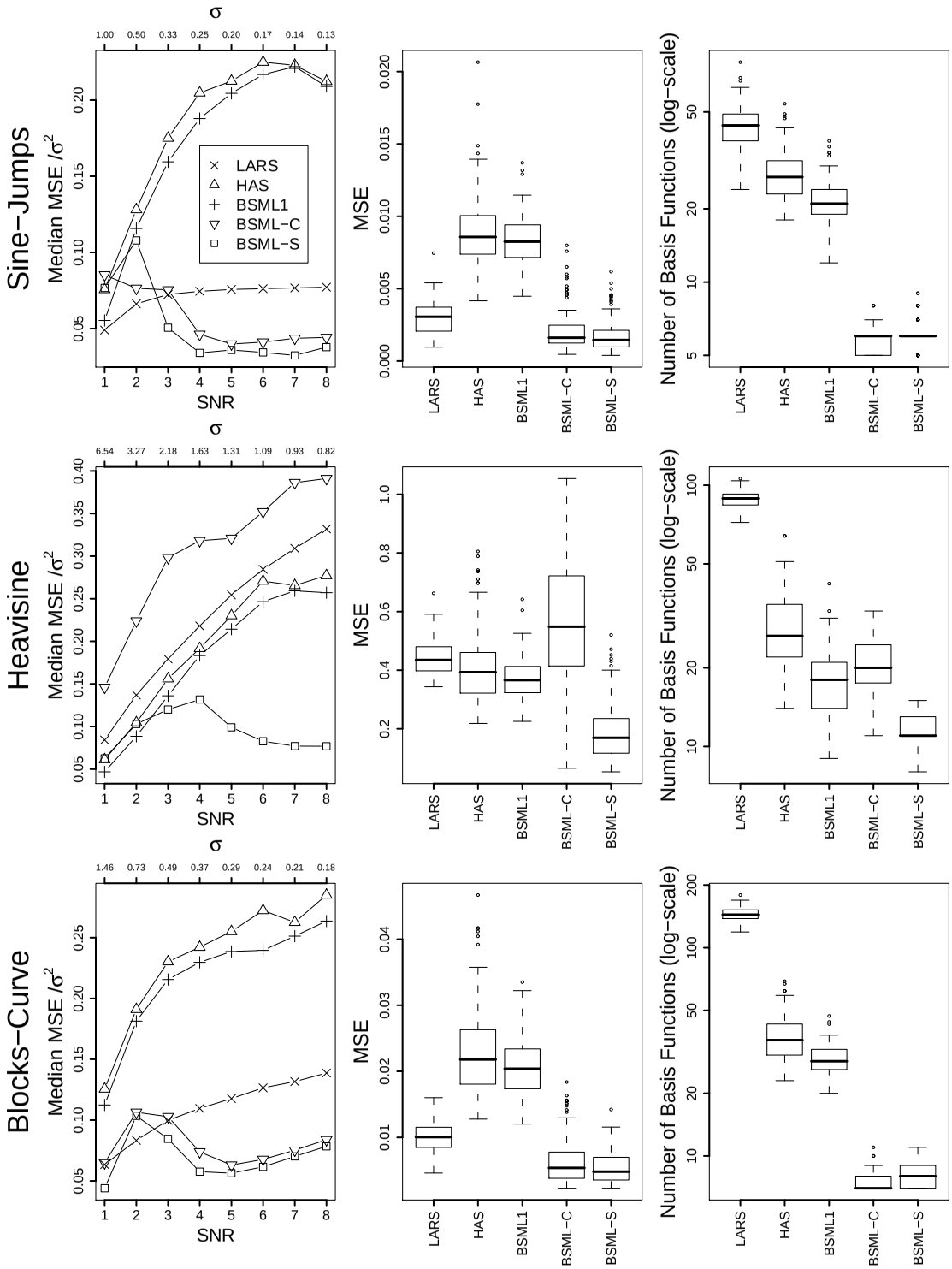

Figure 2: Each row corresponds to one function with the name marked at the left margin. Left panel: median(MSE)/σ^2 from 100 simulated data sets with $n = 256$ for eight σ values (top axis), or equivalently 8 SNR (lower axis). Center panel: boxplots of MSEs for 5 estimation methods with SNR=5. Right panel: corresponding number of basis functions selected with SNR=5. Note that a log spacing is used.

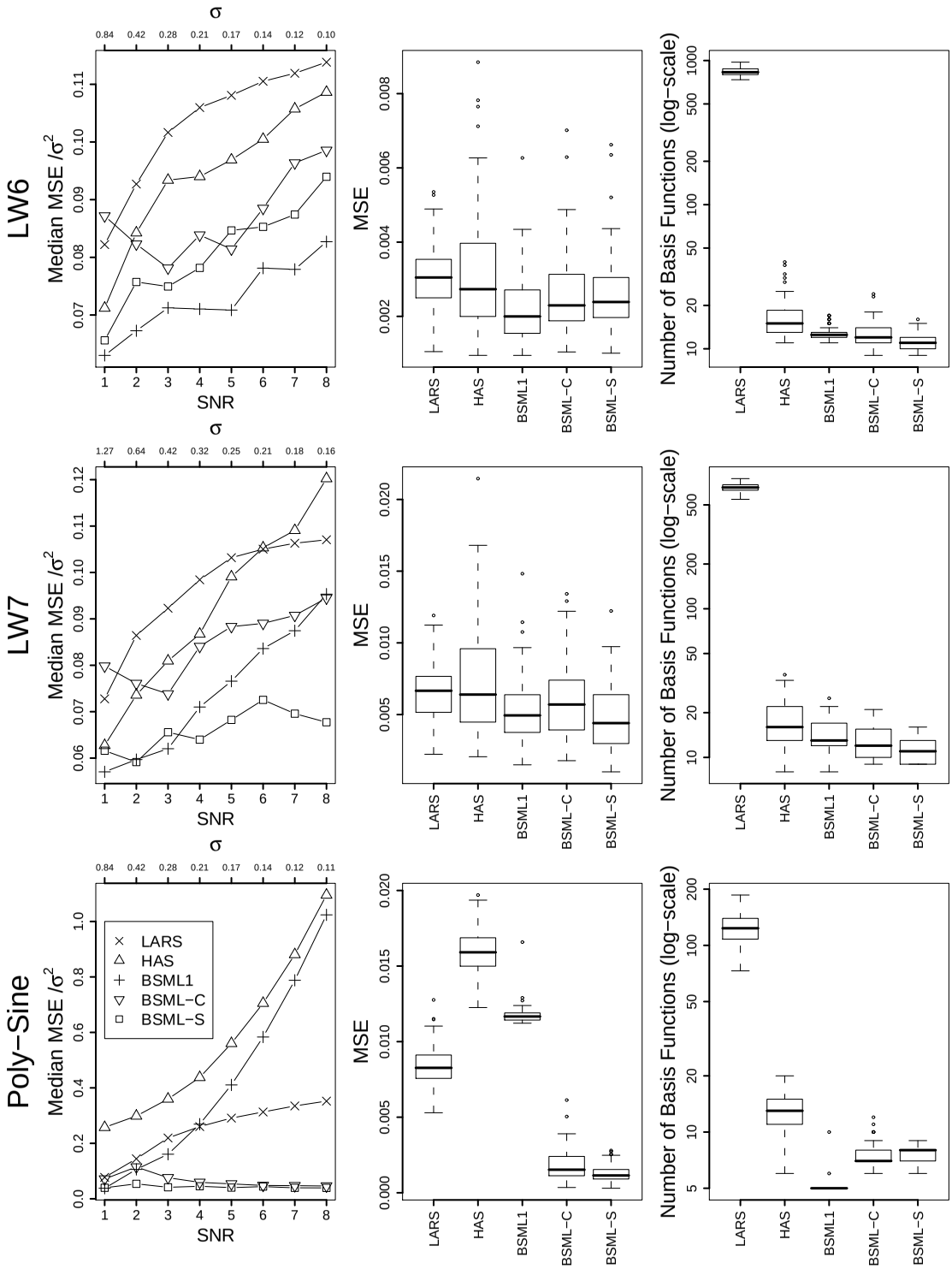

Figure 3: Each row corresponds to one function with the name marked at the left margin. Left panel: median(MSE)/σ^2 from 100 simulated data sets with $n = 256$ for eight σ values (top axis), or equivalently 8 SNR (lower axis). Center panel: boxplots of MSEs for 5 estimation methods with SNR=5. Right panel: corresponding number of basis functions selected with SNR=5. Note that a log spacing is used.

shown). Choosing bases naively from multiple libraries using a constant IDF, as in HAS with two non-null libraries, does not necessarily improve the fit (see also Section 5.1 of the Supplement). When using multiple libraries, there is need of more sophisticated, adaptive estimation of model complexity, such as implemented in BSML using either GDF or the Covariance Penalty.

Except for the Heavisine function, the BSML-C method had similar performance as the BSML-S method. For the Heavisine function it is necessary to treat libraries separately, and BSML-S has superior performance. The BSML-C performance improves relative to LARS as n increases for the heavisine function, with BSML-C performing better than LARS for $n = 1024$ (not shown). We note that the BSML-S method is much more computational intensive than the BSML-C method, but exhibits greater stability that is especially evident in the heavisine simulations.

Figure 4 shows boxplots of MSE values for 100 BSML-S Heavisine fits for each of five sample sizes $n \in \{128, 256, 512, 1024, 2048\}$, and the corresponding number of basis functions chosen by BSML-S. For each of the latter three sample sizes, BSML-S was run after initial selection of 256 basis functions from each library via HAS.

The median MSE decreases with sample size in these simulations. The number of basis functions chosen by BSML-S increases slightly with sample size, but remained relatively stable compared to other methods.

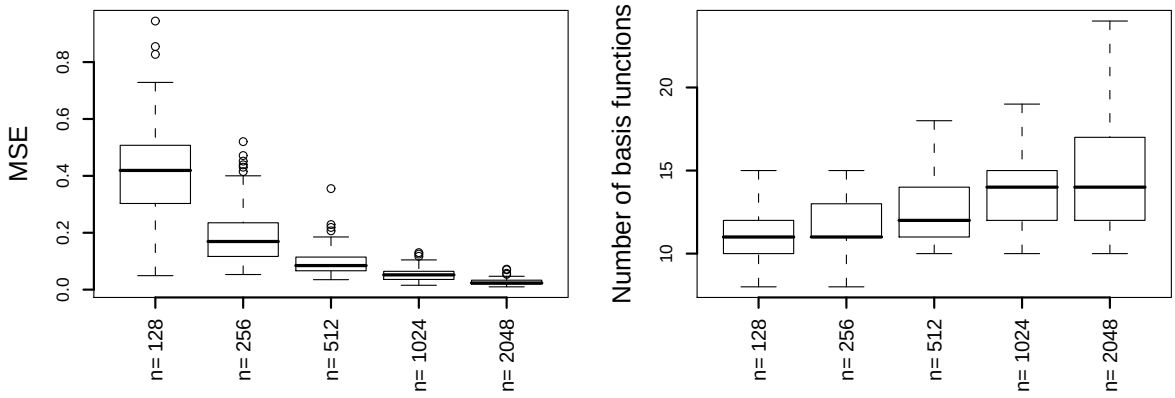

Figure 4: Left panel: Boxplots of MSE of BSML-S fits for 100 simulated data sets from (1) with f the Heavisine function, for sample sizes $n \in \{128, 256, 512, 1024, 2048\}$ and SNR=5. Right panel: Boxplots of the corresponding number of basis functions selected by BSML-S.

The detailed results we present for $n = 256$ are representative of our simulations across

multiple functions. We also have run simulations across multiple sample sizes, as illustrated for the heavisine in Figure 4. Results are similar to those presented here. Our results for spatially adaptive BSML-C and BSML-S support the value of adaptive methods when the SNR is sufficiently large (Wasserman 2006).

3.3 Applications

To illustrate the value of the BSML methods in the practitioner's toolbox, we analyze the US penny thickness data set from Scott (1992), contained in R library `locfit` (Loader 2010). The data are thickness measures of two pennies from each of the years 1945 to 1989 ($n = 90$). As previously noted in the adaptive estimation and change-point literature, the thicknesses changed abruptly at least twice in this time period. Figure 5 shows the observations (dots).

We first transform the explanatory variable into $x = [0, 1]$, and use basis families $\mathcal{G}_0 = \mathcal{U}_1$, $\mathcal{G}_1 = \mathcal{C}_2$, $\mathcal{G}_2 = \mathcal{T}_0$, $\mathcal{G}_3 = \mathcal{T}_1$, and $\mathcal{G}_4 = \mathcal{T}_2$ from Table 1. Each $\mathcal{G}_j,\ j \geq 1$ has 45 elements (one basis function per unique x_i value), corresponding to 45 unique years. The cubic spline with 95% Bayesian confidence intervals, fitted using R library `assist` with GCV (Wang and Ke 2011), oversmooths some of the jumps in the data. HAS with $\mathcal{G}_0$ and $\mathcal{G}_1$ (section 3.1) shows greater adaptivity than the cubic splines, but cannot capture abrupt jumps in the penny thicknesses. Let $\mathcal{O} = \cup_{l=1}^{4}\mathcal{G}_l$. BSML-C used $\mathcal{L}_0 = \mathcal{U}_1$ and adaptively selected from $\mathcal{L}_1 = \mathcal{O}$ using GCV with data-based GDF methods. BSML-S adaptively selected from $L = 4$ candidate basis families $\mathcal{L}_l = \mathcal{G}_l$ for $l = 1, \ldots, 4$, using the DPC (11) and GCV-based elimination. HAS, BSML-C and BSML-S fits use $M = 30$, 300 perturbations in the GDF estimation (Ye 1998), and $B = 300$ for bootstrap percentile confidence interval computations. Our analyses capture the abrupt increase in penny thickness a few years after World War II, and then a subsequent reduction in thickness during the 1970's. Interestingly, adaptive selection from multiple libraries enables us to capture the two major jumps as well as gradually increasing thickness trends around these jumps. These general features compare well with a local linear estimate (Gijbels, Lambert and Qiu 2007) and a spline estimate with constant and linear spaces (Ma and Yang 2011).

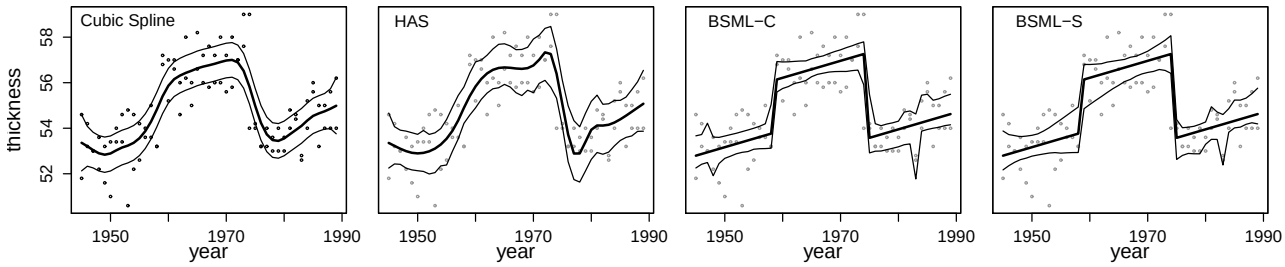

Figure 5: Thickness of 90 US Pennies (dots) with fits. Left to right: Cubic splines using GCV (with 95% Bayesian confidence intervals); HAS, BSML-C and BSML-S (with 95% percentile bootstrap confidence intervals).

4 Model Selection in Multiple Regression

4.1 Smoothing Spline ANOVA

Consider model (1) with multiple explanatory variables $x_1, \cdots, x_d$. The goal is to approximate f using basis functions, resulting in an estimate of f from these noisy data. Many methods have been proposed to approximate the multivariate function. We consider the smoothing spline ANOVA (SS ANOVA) decomposition for comparison with the COSSO method.

Let the domain of each x_k be an arbitrary set $\mathcal{X}_k$ and denote $\boldsymbol{x} = (x_1, \cdots, x_d)$. Let $\mathcal{H}^{(k)}$ be a RKHS on $\mathcal{X}_k$ and

$$\mathcal{H}^{(k)} = \mathcal{H}^{(k)}_{(1)} \oplus \cdots \oplus \mathcal{H}^{(k)}_{(r_k)}, \quad k = 1, 2, \cdots, d.$$

Then the tensor product space $\mathcal{H}^{(1)} \otimes \mathcal{H}^{(2)} \otimes \cdots \otimes \mathcal{H}^{(d)}$ on $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_d$ can be decomposed into

$$\begin{aligned} \mathcal{H}^{(1)} \otimes \mathcal{H}^{(2)} \otimes \cdots \otimes \mathcal{H}^{(d)} &= \left\{\mathcal{H}^{(1)}_{(1)} \oplus \cdots \oplus \mathcal{H}^{(1)}_{(r_1)}\right\} \otimes \cdots \otimes \left\{\mathcal{H}^{(d)}_{(1)} \oplus \cdots \oplus \mathcal{H}^{(d)}_{(r_d)}\right\} \\ &= \sum_{j_1=1}^{r_1} \cdots \sum_{j_d=1}^{r_d} \mathcal{H}^{(1)}_{(j_1)} \otimes \cdots \otimes \mathcal{H}^{(d)}_{(j_d)}. \end{aligned} \tag{18}$$

See Wang (2011) for details about the SS ANOVA decomposition. The number of components in SS ANOVA decomposition (18) increases exponentially as the dimension d increases (curse of dimensionality). To overcome this problem, it is desirable to have model selection methods that determine which components (subspaces) should be included in the model. Lin and Zhang (2006) proposed the COSSO procedure for model selection and estimation based on the SS ANOVA decomposition.

A model containing any subset of components in the SS ANOVA decomposition is referred to as an SS ANOVA model. Given an SS ANOVA model, we can regroup subspaces and write the model space as

$$\mathcal{S} = \mathcal{H}^0 \oplus \mathcal{H}^1 \oplus \cdots \oplus \mathcal{H}^q, \tag{19}$$

where $\mathcal{H}^0$ is a finite dimensional space collecting all functions which are not going to be penalized, and $\mathcal{H}^1, \cdots, \mathcal{H}^q$ are orthogonal RKHS's with RKs R^j for $j = 1, \cdots, q$. The COSSO procedure estimates $f \in \mathcal{S}$ by minimizing the penalized least squares criterion

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - f(\boldsymbol{x}_i))^2 + \lambda \sum_{j=1}^{q} \|P_j f\|, \tag{20}$$

where P_j is the orthogonal projector in $\mathcal{S}$ onto $\mathcal{H}^j$, and parameter λ penalizes the L_1 norm terms, $\|P_j f\|$. For some special models, it has been shown that the COSSO procedure selects the correct model with probability tending to one and leads to consistent estimation of f.

4.2 BSML Model Selection and Simulations

Consider model (1) with model space (19). Let $\mathcal{L}_0 = \mathcal{H}^0$ and $\mathcal{L}_j = \{R^j(\boldsymbol{x}_1, \boldsymbol{x}), \ldots, R^j(\boldsymbol{x}_n, \boldsymbol{x})\}$ for $j = 1, \cdots, q$. Then the BSML procedure can be applied to select basis functions from $\mathcal{O} = \cup_{l=1}^{q} \mathcal{L}_l$ and estimate the function f. A further thresholding procedure may be applied to eliminate libraries with negligible contributions. Specifically, denote the estimated function based on all selected bases as $\hat{f}$. Let $\hat{f}(\boldsymbol{x}) = \hat{f}_0(\boldsymbol{x}) + \sum_{j=1}^{q} \hat{f}_j(\boldsymbol{x})$ where $\hat{f}_j$ for $j = 0, \ldots, q$ are projections of $\hat{f}$ onto $\mathcal{H}_j$. We can eliminate all selected bases in library j if $||\hat{f}_j||/||\hat{f}|| < \tau$ for $j = 1, \ldots, q$ where τ is a small threshold. A similar procedure was used in Lin and Zhang (2006) where the thresholding was achieved through the smoothing parameters.

We use the following simulation to evaluate the performances of the BSML-C and BSML-S procedures in fitting multivariate regression functions and model selection and compare them with the COSSO procedure. Data are generated from model (1) with $\mathbf{x} = (x_1, x_2, x_3, x_4)$ and an additive function $f(\mathbf{x}) = f_1(x_1) + f_2(x_2)$ where $f_1(x_1)$ is the Blocks-Curves function in Table 2 and $f_2(x_2) = -1 + 1.5x_2 + 10\phi(50 \cdot (x_2 - .5))$ where ϕ is the standard normal density function. Note that f does not depend on x_3 and x_4. We consider three sample sizes, $n \in \{100, 300, 500\}$, and three SNR values, SNR $\in \{2, 4, 8\}$. Design points for the four explanatory variables are generated as iid random samples from the uniform distribution on $[0, 1]$. For each combination of sample size and SNR, the simulation is repeated 100 times.

To apply the BSML procedures, we consider the following libraries: $\mathcal{L}_0 = \{1, x_1, x_2, x_3, x_4\}$, $\mathcal{L}_1 = \mathcal{C}_2(x_1)$, $\mathcal{L}_2 = \mathcal{T}_0(x_1)$, $\mathcal{L}_3 = \mathcal{C}_2(x_2)$, $\mathcal{L}_4 = \mathcal{T}_0(x_2)$, $\mathcal{L}_5 = \mathcal{C}_2(x_3)$, $\mathcal{L}_6 = \mathcal{C}_2(x_4)$ where

Table 4: The number of times variables x_3, x_4 are selected out of 100 simulation runs. Results for x_1 and x_2 are not shown since each method always selected these variables. Results with SNR = 8 are not shown since the number of times these variables are selected is zero for all methods.

	SNR = 2			SNR = 4		
	$n = 100$	$n = 300$	$n = 500$	$n = 100$	$n = 300$	$n = 500$
BSML-C	3,4	0,0	0,0	1,0	0,0	0,0
BSML-S	13,12	5,3	1,0	1,1	0,0	0,0
COSSO	9,0	12,2	6,11	0,0	2,0	0,0

$\mathcal{L}_2$ and $\mathcal{L}_4$ are added to the additive SS ANOVA model space of the tensor product of cubic splines to deal with spatial inhomogeneity in f_1 and f_2. For each simulation sample, we apply the BSML-C and BSML-S procedures with these libraries and $M = 40$. We also apply the COSSO procedure using the MATLAB code downloaded from `http://www4.stat.ncsu.edu/~hzhang/software.html`. The estimate of σ^2 from the COSSO procedure is used as the initial estimate of σ^2 in the BSML-C and BSML-S procedures.

Figure 6 shows the median MSE from the BSML-C, BSML-S and COSSO procedures. The BSML procedures have smaller MSEs than the COSSO procedure. We have conducted simulations with smooth functions where the MSEs from the BSML and COSSO procedures are similar (see Section 5.5 of the Supplement). We also applied the MARS procedure (Friedman 1991) using the function "mars" in the R package "mda" (Hastie, Tibshirani, Leisch, Hornik and Ripley 2006). The resulting MSE values (not shown) are substantially larger than those from the BSML and COSSO procedures, perhaps due to the fact that the true function in these simulations is additive. Section 4.4 of the Supplement presents information on computation times for the algorithms compared in this section.

To evaluate the performances in term of variable selection, we apply the thresholding procedure as described above with $\tau = .01$ to the BSML-C, BSML-S and COSSO procedures. For the purpose of variable selection, we combine bases in $\mathcal{L}_1$ and $\mathcal{L}_2$ for variable x_1 and $\mathcal{L}_3$ and $\mathcal{L}_4$ for variable x_2 for thresholding. BSML-C, BSML-S and COSSO all selected x_1 and x_2 100% of the time under all settings. Table 4 shows the number of times each of these methods incorrectly selected variables x_3 and x_4 out of 100 simulations. Overall, all three methods performed quite well. The false selection rate diminishes quickly as SNR increases. As sample size increases, the false selection rate decreases to zero for the BSML procedures. The BSML procedures performed better than COSSO for large sample sizes and small SNRs.

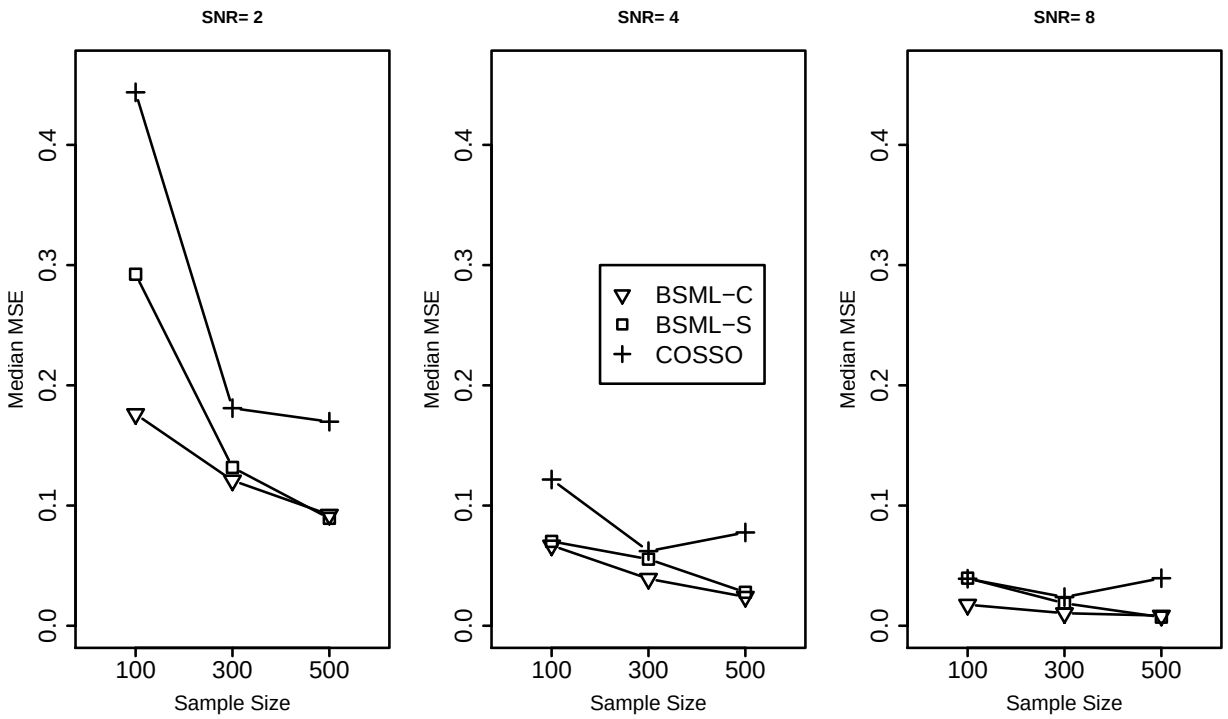

Figure 6: Median MSE vs. n for COSSO (+), BSML-S ($\square$), and BSML-C (∇).

4.3 Application

We analyzed 946 monthly mean ozone thickness measurements (Dobson units) during the 82 years from 1926 to 2007 from Arosa, Switzerland, shown in Figure 7 (a). Data were downloaded from `ftp://iaclin2.ethz.ch/pub_read/maeder/totozone_arosa_monthly` . We first scale the original times (in years and months) into the interval $x \in [0, 1]$. We consider the additive model

$$f(x) \;=\; \mu + f_1(x) + f_2(x), \qquad x \in [0, 1] \tag{21}$$

where f_1 is a periodic function with period 1/82 to model seasonal trend (seasonal main-effect), while $f_2(x)$ models long term trend (year main-effect). We compare three estimation methods: SSANOVA, BSML-C, and BSML-S. Due to space limitations we concentrate on the estimate of the main effect of year $f_2(x)$ in Figure 7(b)-(d), since the estimates of the dominant seasonal cycle variation $f_1(x)$ are very similar for all three methods (not shown). In Figure 7(b)-(d), dots show the Arosa total column ozone annual averages for years with at least nine months of monthly mean data. Annual averages for the years with more than 1/4 of the months missing are not plotted, but all available months are included in the analyses.

Model (21) is a special case of an SS ANOVA model, which we fit via the `ssr` function in the `assist` package (Wang and Ke 2011), using GCV to select smoothing parameters. We assume that $f_1(82x) \in W_2(per) \ominus \{1\}$ (see (16)), where the multiplicative constant 82 makes $f_1(82x)$ a periodic function with period 1, and $f_2(x) \in W_2[0,1] \ominus \{1\}$ (see (14)). Panel (b) shows the fitted year main effect $\hat{f}_2(x)$ together with 95% Bayesian confidence intervals.

Panel (c) shows BSML-C results, and Panel (d) displays those of BSML-S, using libraries $\mathcal{L}_0 = \mathcal{U}_1$, $\mathcal{L}_1 = \mathcal{P}_2$, $\mathcal{L}_2 = \mathcal{C}_2$, and $\mathcal{L}_3 = \mathcal{T}_0$ (see Table 1). For BSML-C and BSML-S, the periodic spline space (16) describes the seasonal cycle, while $\mathcal{L}_2 = \mathcal{C}_2$, and $\mathcal{L}_3 = \mathcal{T}_0$ describe the long-term trend f_2. We used 300 perturbations in the GDF estimation (Ye 1998), $M = 30$, and $B = 300$ for bootstrap percentile confidence interval computations.

Unlike the seasonal cycle estimates (not shown), the estimates of the main effects of years differ across methods (Figure 7(b)-(d)). The SS ANOVA estimate may over-fit the data. The estimate based on BSML-C displays intermediate complexity, including abrupt changes around 1940, a possible dip around 1993 and a gradually decreasing trend in most other years. By comparison, the BSML-S estimate gives a smooth long term trend, except for the bump around 1940. It is interesting in this illustrative example that both the BSML-S and BSML-C main-effects of year especially highlight the period around 1940 as being very different from other time periods, requiring further study. Indeed, there is substantial scientific interest in the atmospheric conditions associated with the high ozone years from 1940 to 1942, potentially linked with a strong El Nino event ((Bronnimann, Luterbacker, Staehelin and Svendby 2004b) and (Bronnimann, Luterbacher, Staehelin, Svendby, Hansen and Svenoe 2004a)). While the scientific interpretation must lie beyond the scope of our paper, this example illustrates the value of BSML-procedures to highlight features of scientific interest in $f(x)$.

5 Discussion

We have introduced a new approach to adaptive non-parametric regression. The combination of multiple libraries with adaptive estimation of selection cost provides the key to flexibility of this new approach. Multiple libraries allows approximation of different components (or regions) by different basis functions. Adaptive estimation of selection cost allows basis functions in different libraries to compete on an equal footing. The method is general since it allows fusion among different families/types/orders of basis functions.

We used spatially adaptive non-parametric regression and multivariate non-parametric

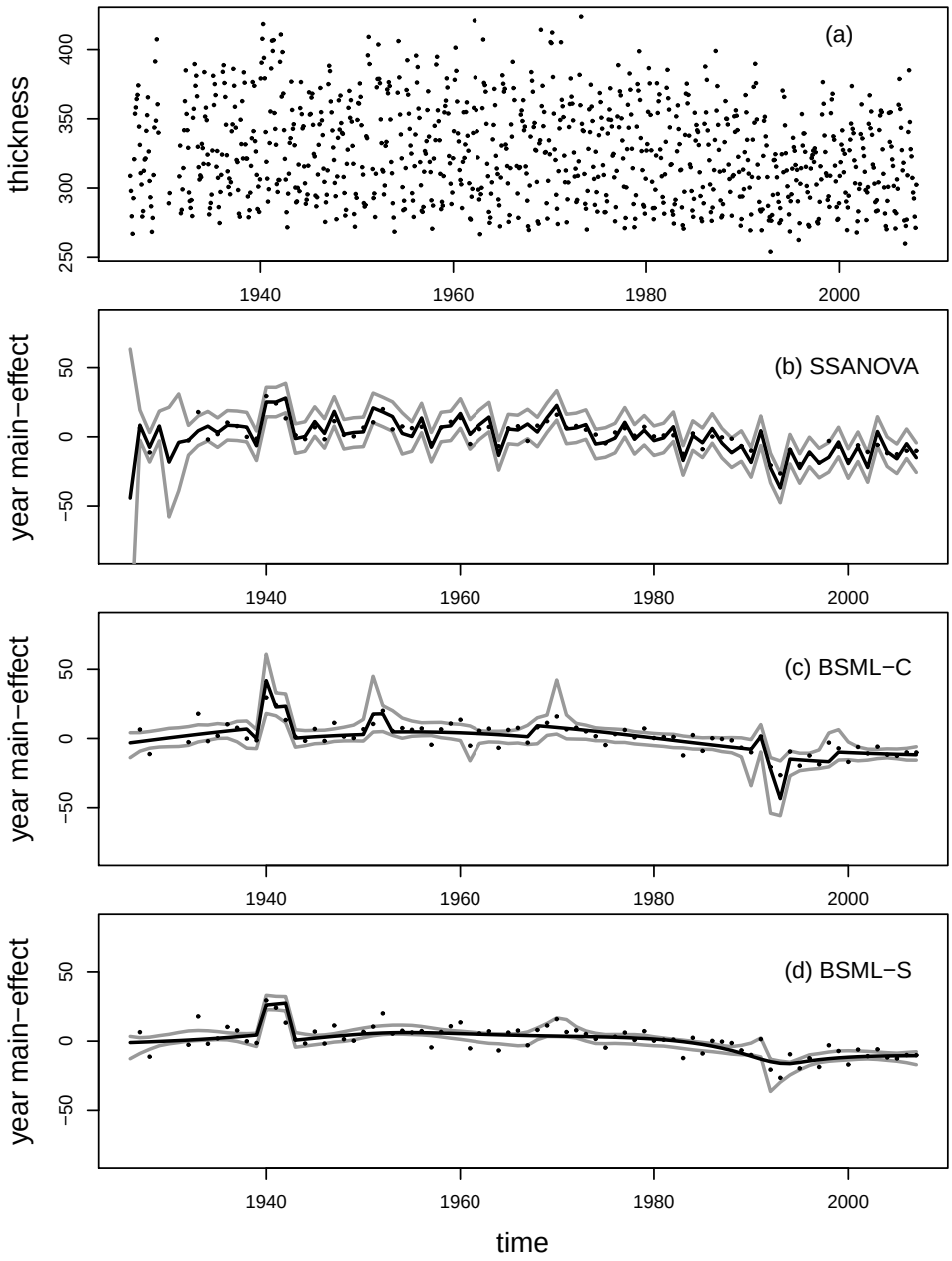

Figure 7: (a) Observations of Arosa total column ozone (monthly mean total ozone records over the period 1927-2007). (b) SSANOVA estimated year effect (estimated $f_2(x)$) with 95% Bayesian confidence intervals. (c), (d) BSML-C, BSML-S estimated year effect with 95% percentile bootstrap confidence intervals. In panels (b)-(d), each dot is the annual average for a year with at least nine months of monthly mean data. Years with fewer than nine months of data were included in the analysis, but their annual averages are not plotted.

model selection to motivate and illustrate the BSML methods. Nevertheless, the BSML procedures are adaptive in the more general sense that each dynamically adjusts its strategy

to take into account the behavior of the function to be estimated (Friedman 1991). As general procedures, they have other potential applications. One of our future research topics is to extend the BSML methodology to different variance/covariance structures and to fit data from exponential families. We also plan to explore further the value of slight over-fitting in the pilot fits to reduce bias in the percentile bootstrap confidence intervals, since these intervals depend on the pilot fit. This will extend the study of coverage properties, beyond the promising results regarding close to nominal coverage already provided in Section 5.3 of the Supplement.

Supplementary Materials

A pdf document containing the following sections:

Adaptive IDF Empirical evidence that IDF depend on many factors including the true function, the type of basis functions, the signal-to-noise ratio, and the basis functions that have been selected.

Challenge Additional motivation for the need of data-based cost criteria.

Variance estimation Difference-based estimators for σ^2.

Computation Details for forward selection, Monte Carlo algorithms for GDF and covariance penalty, the R `BSML` package including the code for the Penny Thickness data, and information on computation times.

More simulations Comparison between different model selection criteria, sensitivity to the choice of c_1 and c_2 in DPC (11), bootstrap confidence interval coverages, sensitivity to library specifications, and additional simulation results.

References

Abramovich, F. and Steinberg, D. (1996). Improved inference in nonparametric regression using l_k-smoothing splines, *Journal of Statistical Planning and Inference* **49**: 327–341.

Akaike, H. (1970). Statistical predictor identification, *Ann. Inst. Statist. Math.* **21**: 203–217.

Bronnimann, S., Luterbacher, J., Staehelin, J., Svendby, T., Hansen, G. and Svenoe, T. (2004a). Extreme climate of the global troposphere and stratosphere in 1940-42 related to El Nino, *Nature* **431**(7011): 971–974.

Bronnimann, S., Luterbacker, J., Staehelin, J. and Svendby, T. (2004b). An extreme anomaly in stratospheric ozone over Europe in 1940-1942, *Geophysical Research Letters* **31**: L08101, doi:10.1029/2004GL019611.

Chen, S. S., Donoho, D. L. and Saunders, M. A. (2001). Atomic decomposition by basis pursuit, *SIAM Review* **43**: 129–159.

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

Cummins, D. J., Filloon, T. G. and Nychka, D. (2001). Confidence intervals for nonparametric curve estimates: toward more uniform pointwise coverage, *Journal of the American Statistical Association* **96**: 233–246.

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

Dias, R. and Gamerman, D. (2002). A Bayesian approach to hybrid splines non-parametric regression, *Journal of Statistical Computation and Simulation* **72**: 285–297.

DiMatteo, I., Genovese, C. and Kass, R. (2001). Bayesian curve fitting with free-knot splines, *Biometrika* **88**: 1055–1071.

Efron, B. (2004). The estimation of prediction error: Covariance penalties and cross-validation (with discussion), *Journal of the American Statistical Association* **99**: 619–632.

Efron, B., Hastie, T., Johnstone, I. and Tibshirani, R. (2004). Least angle regression (with discussion), *Annals of Statistics* **32**: 407–499.

Fan, J. and Gijbels, I. (1995). Data-driven bandwidth selection in local polynomial fitting: Variable bandwidth and spatial adaption, *Journal of the Royal Statistical Society Series B* **57**: 371–394.

Fan, J. and Gijbels, I. (1996). *Local polynomial modelling and its applications*, Vol. 66 of *Monographs on Statistics and Applied Probability*, Chapman & Hall, London.

Friedman, J. H. (1991). Multivariate adaptive regression splines (with discussion), *Annals of Statistics* **19**: 1–67.

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

Gijbels, I., Lambert, A. and Qiu, P. (2007). Jump-preserving regression and smoothing using local linear fitting: a compromise, *Annals of the Institute of Statistical Mathematics* **59**(2): 235–272.

Green, P. J. and Silverman, B. W. (1994). *Nonparametric Regression and Generalized Linear Models: A Roughness Penalty Approach*, London: Chapman and Hall.

Gribonval, R. and Nielsen, M. (2003). Sparse decompositions in unions of bases, *IEEE Transactions on Information Theory* **49**: 3320–3325.

Gu, C. (2002). *Smoothing Spline ANOVA Models*, Springer-Verlag, New York.

Hall, P. (1992). Effect of bias estimation on coverage accuracy of bootstrap confidence intervals for a probability density, *Annals of Statistics* **20**: 675–694.

Hastie, T. and Efron, B. (2011). *lars: Least Angle Regression, Lasso and Forward Stagewise*. R package version 0.9-8.

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

Hastie, T., Tibshirani, R. and Friedman, J. (2009). *The Elements of Statistical Learning, 2nd ed.*, New York: Springer.

Hastie, T., Tibshirani, R., Leisch, F., Hornik, K. and Ripley, B. D. (2006). *mda: Mixture and flexible discriminant analysis*. R package version 0.3-2, S original by T. Hastie & R. Tibshirani, R port by F. Leisch and K. Hornik and B.D. Ripley.

Lebarbier, E. (2005). Detecting multiple change-points in the mean of Gaussian process by model selection, *Signal Processing* **85**: 717–736.

Lewicki, M. S. and Sejnowski, T. J. (2000). Learning overcomplete representations, *Neural Computation* **12**: 337–365.

Lin, Y. and Zhang, H. H. (2006). Component selection and smoothing in multivariate nonparametric regression, *Annals of Statistics* **34**: 2272–2297.

Liu, Z. and Guo, W. (2010). Data driven adaptive spline smoothing, *Statistica Sinica* pp. 1143–1163.

Loader, C. (2010). *locfit: Local Regression, Likelihood and Density Estimation.* R package version 1.5-6.

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

Ma, S. and Yang, L. (2011). A jump-detecting procedure based on spline estimation, *Journal of Nonparametric Statistics* **23**(1): 67–81.

Mallat, S. and Zhang, Z. (1993). Matching pursuit in a time-frequency dictionary, *IEEE Trans. Signal Proc.* **41**: 3397–3415.

Miyata, S. and Shen, X. (2003). Adaptive free-knots splines, *Journal of Computational and Graphical Statistics* **12**: 197–213.

Picard, F., Robin, S., Lavielle, M., Vaisse, C. and Daudin, J. J. (2005). A statistical approach for array CGH data analysis, *BMC Bioinformatics* **6**: Art. No. 27, Feb.

Ruppert, D. and Carroll, R. (2000). Spatially-adaptive penalties for spline fitting, *Australian and New Zealand Journal of Statistics* **42**: 205–223.

Ruppert, D., Wand, M. P. and Carroll, R. J. (2003). *Semiparametric Regression*, Cambridge, New York.

Scott, D. W. (1992). *Multivariate Density Estimation*, Wiley-Interscience, New York.

Shen, X. and Huang, H. (2006). Optimal model assessment, selection, and combination, *Journal of the American Statistical Association* **101**: 554–568.

Sklar, J. C. (2003). Some contributions to spatially adaptive non-parametric regression, Ph.D. Thesis, University of California-Santa Barbara, Dept. of Statistics and Applied Probability.

Sklar, J. C., Meiring, W. and Wang, Y. (2006). Flexible statistical methods for array-based comparitive genomic hybridization analysis, *Progress in Genome Research, C. R. Williams Eds.*, pp. 139–153.

Tibshirani, R. and Knight, K. (1999). The covariance inflation criterion for adaptive model selection, *J. R. Stat. Soc. Ser. B Stat. Methodol.* **61**: 529–546.

Tong, T. and Wang, Y. (2005). Estimating residual variance in nonparametric regression using least squares, *Biometrika* **92**: 821–830.

Tropp, J. A. (2004). Greed is good: algorithmic results for sparse approximations, *IEEE Transactions on Information Theory* **50**: 2231–2242.

Wahba, G. (1990). *Spline Models for Observational Data*, SIAM, Philadelphia. CBMS-NSF Regional Conference Series in Applied Mathematics, Vol. 59.

Wang, Y. (1995). Jump and sharp cusp detection by wavelets, *Biometrika* **82**: 385–397.

Wang, Y. (2011). *Smoothing Splines: Methods and Applications*, Chapman and Hall, New York.

Wang, Y. and Ke, C. (2011). *assist: A Suite of S-Plus Functions Implementing Smoothing Splines*. R package version 3.1.2.

Wang, Y. and Wahba, G. (1995). Bootstrap confidence intervals for smoothing splines and their comparison to Bayesian confidence intervals, *J. Statist. Comput. Simul.* **51**: 263–279.

Wasserman, L. (2006). *All of Nonparametric Statistics*, Springer, New York.

Wu, J. (2011). Basis selection from multiple libraries, Ph.D. Thesis, University of California-Santa Barbara, Dept. of Statistics and Applied Probability.

Wu, J., Sklar, J. C., Wang, Y. and Meiring, W. (2011). *bsml: Basis Selection from Multiple Libraries*. R package version 1.4-1.

Yang, Y. (1999). Model selection for nonparametric regression, *Statistica Sinica* **9**: 475–499.

Ye, J. M. (1998). On measuring and correcting the effects of data mining and model selection, *Journal of the American Statistical Association* **93**: 120–131.

Zhou, S. and Shen, X. (2001). Spatially adaptive regression splines and accurate knot selection schemes, *Journal of the American Statistical Association* **96**: 247–259.