Stat Comput
DOI 10.1007/s11222-012-9346-4

Iterative bias reduction: a comparative study

P.-A. Cornillon · N. Hengartner · N. Jegou · E. Matzner-Løber

Received: 28 April 2011 / Accepted: 27 July 2012

Abstract Multivariate nonparametric smoothers, such as kernel based smoothers and thin plate splines smoothers, are adversely impacted by the sparseness of data in high dimension, also known as the curse of dimensionality. Adaptive smoothers, that can exploit the underlying smoothness of the regression function, may partially mitigate this effect. This paper presents a comparative simulation study of a novel adaptive smoother (IBR) with competing multivariate smoothers available as package or function within the R language and environment for statistical computing. Comparison between the methods are made on simulated datasets of moderate size, from 50 to 200 observations, with two, five or 10 potential explanatory variables, and on a real dataset. The results show that the good asymptotic properties of IBR are complemented by a very good behavior on moderate sized datasets, results which are similar to those obtained with Duchon low rank splines.

Keywords Multivariate smoothing · Thin-plate splines · Duchon splines · Kernel regression · Iterative bias reduction

P.-A. Cornillon
IRMAR, Univ. Rennes 2, 35043 Rennes, France
e-mail: pac@uhb.fr

N. Jegou · E. Matzner-Løber (✉)
Univ. Rennes 2, 35043 Rennes, France
e-mail: eml@uhb.fr

N. Jegou
e-mail: nicolas.jegou@uhb.fr

N. Hengartner
Los Alamos National Laboratory, Los Alamos, NW 87545, USA
e-mail: nickh@lanl.gov

1 Introduction

In many applications, one seeks to explain a response variable by a set of potential explanatory variables. Regression, which is a fundamental data analysis tool, solves this problem by estimating the functional relationships between pairs of observations $(X_i, Y_i), i = 1, \ldots, n$. In its simplest form, one models the conditional expectation of the dependent variable Y given the independent variables $X \in \mathbb{R}^d$ by a linear combination of the covariates and estimates the parameters by minimizing a suitable cost function between the observed and the fitted values, usually the sum of squared errors. More generally, one may explicitly specify parametric families for regression functions that describe the conditional expectation of the dependent variable Y given X.

Nonparametric regression provides a more flexible model that does not require the specification of a particular parametric form from the conditional expectation. Instead, it only assumes that the conditional expectation of Y be a smooth function of the covariates X. Typically, nonparametric models are estimated locally, and the predicted values are smoother than the original observations. Hence nonparametric regression estimators are often called smoothers.

Over the past thirty years, numerous smoothers have been proposed: running-mean smoother, running-line smoother, bin smoother, kernel based smoother, splines regression smoother, smoothing splines smoother, locally weighted running-line smoother, just to mention a few. We refer to Buja et al. (1989), Eubank (1988), Fan and Gijbels (1996), and Hastie and Tibshirani (1995) for more in depth treatments of regression smoothers. Most of these smoothers behavior is closely related to a good choice for the smoothing parameter λ and much has been written on how to select an appropriate smoothing parameter (see for example Simonoff 1996). Classical smoothers have to face *the curse*

Published online: 30 August 2012

of dimensionality which could be summarized as follows: as the dimension of the data increases, so does the sparseness of the covariates and as a consequence, nonparametric smoothers must average over larger neighborhoods, which in turn produces more heavily biased smoothers. Optimally selecting the smoothing parameter does not alleviate this problem and as a remedy, the common wisdom is to avoid all together general nonparametric smoothing with moderate sample size in dimensions higher than three. In this cases, it is usual practice in the statistical community to fit structurally constrained regression models such as additive models (Hastie and Tibshirani 1995; Wood 2004), MARS (Friedman 1991), projection pursuit models (Friedman and Stuetzle 1981) or additive L_2-Boosting (Bühlmann and Yu 2003).

The popularity of additive models (or MARS models) stems from its interpretability and from the fact that the estimated regression function converges to the best additive approximation of the true regression function at the optimal univariate mean squared error rate of $n^{-2\nu/(2\nu+1)}$, where ν is the smoothing index (see for example Tsybakov 2009). While additive models do not estimate the true underlying regression function, one hopes that the approximation error will be small enough so that for moderate sample sizes, the prediction mean square error of the additive model is less than the prediction error of a fully nonparametric regression model.

The optimal mean square error rate of convergence depends on both the dimension d of the covariates and the smoothness of the unknown regression function, which is of course unknown. It is well-known that for regression function m from $\mathbb{R}^d$ to $\mathbb{R}$ known to belong to some smoothness functional classes (e.g Holder, Sobolev, Besov), the optimal mean squared error rate of convergence is $n^{-2\nu/(2\nu+d)}$. Thus, if the regression function m is of smoothness index $\nu = 2d$, then the optimal rate is $n^{-4/5}$, a value recognized as the optimal mean squared error of estimates for twice differentiable univariate regression functions. This suggests that nonparametric regression in higher dimensions is practical, provided that the true regression function is known to be sufficiently smooth, and that the smoothing methods exploits this knowledge.

While in practice, one rarely knows *a priori* the smoothness of the regression function, there exists smoothers achieving the optimal asymptotic mean square error without prior specification of the smoothness. Such methods are called adaptive, and we refer the interested reader to Lepski (1991), Gyorfi et al. (2002), Tsybakov (2009) for general discussions on adaptation in nonparametric estimation. Roughly speaking, adaptation can be achieved either by direct estimation (see for example Lepski's method, Lepski 1991, and related papers) or by aggregation of different procedures (see Yang 2000). Even if potential gain can be achieved by these nonparametric adaptive estimators, there is a lack of multivariate adaptive smoothers that work well in practice with moderate sample size n (ranging from a hundred to few thousands observations). Recently, Cornillon et al. (2011b) proposed an adaptive iterative smoothing method that is very promising for such datasets. The method, called *Iterative Bias Reduction* (abbreviated to IBR in this paper), starts out with a biased smoother that has a large smoothing parameter λ (ensuring that the data are over-smoothed) and then proceeds to estimate and correct the bias in an iterative fashion. This approach is attractive in that it uses existing smoothers, yet by iteratively estimating and correcting the bias, it achieves adaptation.

The aim of this paper is (1) to demonstrate, through simulations and applications to a real dataset, the good practical performance of IBR predicted by the asymptotic theory in Cornillon et al. (2011b) for moderate sample sizes and (2) to compare its performances to those obtained by various competitors. All these competitors must be usable for end-user and thus must be included in some R packages. This last consideration leads us to compare IBR to the following methods: additive models (R package **mgcv**), projection pursuit regression (R function **ppr**), MARS (R package **mda**), additive L_2-Boosting (R package **mboost**) and direct multivariate regression modeling such as low rank thin-plate splines or Duchon splines (R package **mgcv**, Wood 2003).

The paper is organized as follows. Section 2 briefly introduces the IBR smoother, discusses how to initiate and stop the iterative procedure, and reviews its theoretical properties. Section 3 presents IBR with thin plate splines and Duchon splines and discusses the choice of the initial values in order to obtain biased (pilot) smoothers. Section 4 assesses the finite sample properties of the IBR smoother by comparing in simulations its performances with other multivariate smoothing methods that have end-user implementation. Section 5 discusses variable selection for nonparametric smoothers, and show through simulations, improvements in the prediction mean squared error. Section 6 applies variable selection for the IBR smoother to the Los Angeles Ozone dataset and concluding remarks end the paper.

2 IBR: iterative bias reduction

2.1 Preliminaries: linear smoother

Suppose that the pairs $(X_i, Y_i) \in \mathbb{R}^d \times \mathbb{R}$ are related through the nonparametric regression model

$$Y_i = m(X_i) + \varepsilon_i, \quad i = 1, \ldots, n, \tag{1}$$

where $m(\cdot)$ is an unknown smooth function and the disturbances ε_i are independent mean zero and variance σ^2 random variables that are independent of all the covariates

$(X_1, \ldots, X_n)$. It is helpful to rewrite equation (1) in vector form by setting $Y = (Y_1, \ldots, Y_n)^t$, $m = (m(X_1), \ldots, m(X_n))^t$ and $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_n)^t$, to get

$$Y = m + \varepsilon. \tag{2}$$

Linear smoothers can be written in vector format as

$$\widehat{m}_1 = S_1 Y, \tag{3}$$

where S_1 is an $n \times n$ smoothing matrix depending on a smoothing parameter and $\widehat{m}_1 = \widehat{Y} = (\widehat{Y}_1, \ldots, \widehat{Y}_n)^t$ denotes the vector of fitted values. The conditional bias of such a linear smoother is

$$B(\widehat{m}_1) = \mathbb{E}[\widehat{m}_1 | X] - m = (S_1 - I)m. \tag{4}$$

2.2 Bias reduction of linear smoothers

The expression (4) for the bias suggests that it can be estimated by smoothing the negative residuals $-R_1 = -(Y - \widehat{m}_1) = -(I - S_1)Y$. That is,

$$\hat{b}_1 := -S_2 R_1 = -S_2(I - S_1)Y \tag{5}$$

estimates the bias using a (possibly) different smoother S_2. Correcting the pilot smoother $\widehat{m}_1$ by subtracting $\hat{b}_1$ yields a *bias corrected* smoother

$$\widehat{m}_2 = S_1 Y + S_2(I - S_1)Y = \big(S_1 + S_2(I - S_1)\big)Y.$$

Since $\hat{m}_2$ is itself a linear smoother, it is possible to correct its bias as well. Repeating the bias reduction step $k-1$ times produces the linear smoother given in the following proposition. We have to keep in mind that in order to reduce the bias, we need a biased initial smoother. Moreover, at each iteration, reducing the bias is done at the cost of increasing the variance. A natural question is how to stop algorithm (c.f. Sect. 2.4).

Proposition 1 (Residual smoothing estimator) *After $k-1$ iterations, the bias corrected estimator can be explicitly written as*

$$\begin{aligned}\hat{m}_k &= S_1 Y + S_2(I - S_1)Y + \cdots \\ &\quad + S_k(I - S_{k-1}) \cdots (I - S_1)Y \\ &= \big[I - (I - S_k)(I - S_{k-1}) \cdots (I - S_1)\big]Y.\end{aligned} \tag{6}$$

An alternative approach is to estimate the bias by plugging in an estimator $\tilde{m} = S_2 Y$ for the regression function m into the expression of the bias (4). This produces the estimator

$$\tilde{b}_1 = (S_1 - I)S_2 Y$$

for the bias.

Proposition 2 (Plug-in estimator) *After $k-1$ iterations, plug-in bias estimator can be explicitly written as*

$$\begin{aligned}\hat{m}_k &= S_1 Y + (I - S_1)S_2 Y + \cdots + (I - S_1)(I - S_2) \cdots S_k Y \\ &= \big[I - (I - S_1)(I - S_2) \ldots (I - S_k)\big]Y.\end{aligned} \tag{7}$$

While in general, these two estimates for the bias lead to distinct bias corrected smoothers (6) and (7), they are identical when the same smoothing matrix is used at every step of the procedure.

Proposition 3 (Iterating the same smoothing matrix) *Taking $S = S_1 = S_2 = \cdots = S_k$, both the plug-in estimator and the residual smoothing estimator agree and the k^{th} iterated bias corrected smoother can be written as*

$$\hat{m}_k = \big[I - (I - S)^k\big]Y. \tag{8}$$

This closed form shows that the qualitative behavior of the sequence of iterative bias corrected smoothers $\widehat{m}_k$ is governed by the spectrum of $I - S$ (see Cornillon et al. 2011b). If the eigenvalues λ_j of $I - S$ are in $[0, 1]$ then as k tends to infinity, the bias converges to 0 and the variance increases to $n\sigma^2$.

In the univariate case, smoothers of the form (8) arise from the L_2-boosting algorithm with a symmetric base smoother S and a convergence factor μ_k equal to one (see Friedman 2001, for a definition of this factor). Thus we can interpret the L_2-boosting algorithm as an iterative bias reduction procedure in this special case. From a historical perspective, the idea of estimating the bias from residuals to correct a pilot estimator of a regression function goes back to the concept of *twicing* introduced by Tukey (1977) to estimate the bias of misspecified multivariate regression models. The idea of iterative debiasing regression smoothers is also present in Breiman (1999) in the context of the *bagging* algorithm. More recently, the interpretation of the L_2-boosting algorithm as an iterative bias correction scheme was alluded to in Ridgeway's discussion of the paper on the statistical interpretation of boosting of Friedman et al. (2000). Bühlmann and Yu (2003) present the statistical properties of the L_2-boosted univariate smoothing splines, while Di Marzio and Taylor (2008) describe the behavior of univariate kernel smoothers after a single bias-correction iteration.

2.3 Prediction with smoothers

The linear smoother defined by (3) predicts the conditional expectation of responses only at the design points. It is useful to extend regression smoothers to enable predictions at arbitrary locations $x \in \mathbb{R}^d$ of the covariates. Such an extension allows us to assess and compare the quality of various

smoothers by how well the smoother predicts new observations.

To this end, write the prediction of the linear smoother S at an arbitrary location x as

$$\hat{m}(x) = S(x)^t Y,$$

where $S(x)$ is a vector column of size n whose entries are the weights for predicting $m(x)$. The vector $S(x)$ is readily computed for many of the smoothers used in practice. For example, for a kernel smoother (with a bandwidth h), one readily obtains that

$$S(x) = \frac{1}{\sum_{l=1}^{n} K(\frac{x-X_l}{h})} \times \left(K\left(\frac{x-X_1}{h}\right), \ldots, K\left(\frac{x-X_n}{h}\right)\right)^t.$$

We want to find a similar equation for the IBR smoother $\hat{m}$. Writing the latter smoother as

$$\begin{aligned}\widehat{m}_k &= \widehat{m}_0 + \widehat{b}_1 + \cdots + \widehat{b}_k \\ &= S\left[I + (I-S) + (I-S)^2 + \cdots + (I-S)^{k-1}\right]Y \\ &= S\widehat{\beta}_k,\end{aligned}$$

it follows that we can predict $m(x)$ by

$$\widehat{m}_k(x) = S(x)^t \widehat{\beta}_k,$$
$$\text{with } \widehat{\beta}_k = \left[I + (I-S) + (I-S)^2 + \cdots + (I-S)^{k-1}\right]Y.$$

The sequence of parameters $\widehat{\beta}_k$ can be computed recursively by

$$\widehat{\beta}_k = Y + (I-S)\widehat{\beta}_{k-1}.$$

2.4 Stopping rules

As we can see from Eq. (8), the qualitative behavior of the iterated estimator is governed by the spectrum of $I - S$. For splines smoothers and kernel smoothers with a positive define kernel, the spectrum lies in the unit interval $[0, 1]$ (Cornillon et al. 2011b). The package **ibr** (Cornillon et al. 2011a) is implemented with these types of smoothers. It follows from Eq. (8) that as the number of iterations k goes to infinity, the sequence of iterated smoothers $\hat{m}_k$ tends to reproduce the raw data Y. Thus iterating the algorithm until convergence is not desirable. However, since each iteration reduces the bias and increases the variance, often a few iterations of the algorithm will produce a better smoother than the pilot smoother. This brings up the important question of how to decide when to stop the iterative bias correction process.

Viewing the latter question as a model selection problem suggests stopping rules based on Akaike Information Criterion, AIC (Akaike 1973), modified AIC criterion (Hurvich et al. 1998), Bayesian Information Criterion, BIC (Schwarz 1978), and Generalized Cross Validation, GCV (Craven and Wahba 1979). These selectors, all implemented in the **ibr** package, can be written in a common form

$$\operatorname*{argmin}_{k \in \mathcal{K}} \left\{ \log \widehat{\sigma_k}^2 + \Phi\left(\operatorname{tr}(S_k)\right)\right\},$$

where $\widehat{\sigma_k}^2 = \frac{1}{n}\|Y - \hat{m}_k\|^2$, ($\|\|$ is the usual Euclidean norm) and

$$\Phi_{\mathrm{AIC}}\left(\operatorname{tr}(S_k)\right) = 2\frac{\operatorname{tr}(S_k)}{n},$$
$$\Phi_{\mathrm{BIC}}\left(\operatorname{tr}(S_k)\right) = \log n \frac{\operatorname{tr}(S_k)}{n}$$
$$\Phi_{\mathrm{AIC_C}}\left(\operatorname{tr}(S_k)\right) = 1 + 2\frac{\operatorname{tr}(S_k)+1}{n - \operatorname{tr}(S_k) - 2},$$
$$\Phi_{\mathrm{GCV}}\left(\operatorname{tr}(S_k)\right) = -2\log\left(1 - \frac{\operatorname{tr}(S_k)}{n}\right).$$

We are interested in choosing the best selector for the number of iterations k among the above listed procedures. It is instructive to observe how each of these criteria behave over the entire range of k, from zero to infinity. When the number of iterations k tends to infinity, $\operatorname{tr}(S_k)$ converges to n. This means that we are almost interpolating the data, which implies that the residual sum of square, and hence $\hat{\sigma}_k$, tends to zero. Thus for splines and kernel smoothers with positive definite kernels, the ratio $\operatorname{tr}(S_k)/n$ increases monotonically to one and the estimated variance decreases to zero with a growing number of iterations k.

Figure 1 shows the different qualitative behavior of the penalization term Φ. Both the AIC and BIC penalties are linear in $\operatorname{tr}(S_k)/n$, and reach 2 and $\log n$, respectively, at $\operatorname{tr}(S_k) = n$. For problem with large σ^2, the AIC and BIC criteria will select the large number of iterations k, producing smoothers that nearly interpolate the data, which defeats the purpose of smoothing. This behavior is consistent with the general experience in nonparametric smoothing, where it is well-known that AIC criterion has a noticeable tendency to select smoothing parameters that are smaller than needed. As this leads to undersmooth the data, Hurvich et al. (1998) introduced a corrected version of the AIC ($\mathrm{AIC_C}$) under the simplifying assumption that the nonparametric smoother $\hat{m}$ is unbiased. This assumption is problematic in our context, as IBR deliberately starts out with a very biased estimate. For these reasons, and because of the asymptotic results given in Theorem 2 in Cornillon et al. (2011b), we advocate using GCV as the default stopping rule and use it in our simulations.

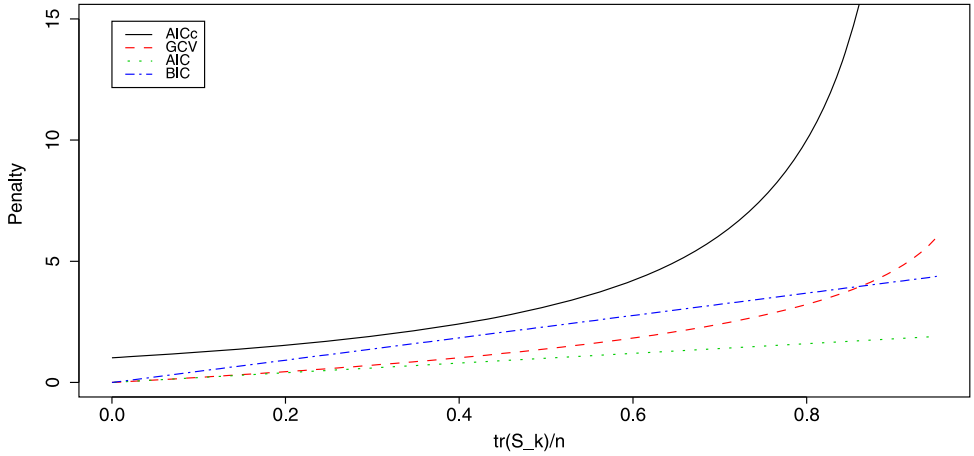


Fig. 1 Φ-penalties for various selectors as a function of $\mathrm{tr}(S_k)/n$

3 IBR with splines or kernels

Before discussing about initial values for IBR, let us present some IBR base smoothers S: thin-plate splines (TPS), Duchon splines and Gaussian kernel.

3.1 Thin plate splines

Suppose the unknown function m from $\mathbb{R}^d \to \mathbb{R}$ belongs to the Sobolev space $\mathcal{H}^{(\nu)}(\Omega) = \mathcal{H}^{(\nu)}$, where ν is an unknown integer such that $\nu > d/2$ and Ω is an open bounded subset of $\mathbb{R}^d$. Recall that thin plate splines (TPS) arise as the solution of the following minimization problem on $\mathcal{H}^{(\nu)}$ (see Gu 2002; Wood 2003)

$$\frac{1}{n}\big\|Y_i - f(X_i)\big\|^2 + \lambda J_\nu^d(f),$$

where

$$J_\nu^d(f) = \sum_{\alpha_1+\cdots+\alpha_d=\nu} \frac{\nu!}{\alpha_1!\cdots\alpha_d!} \times \int\cdots\int \left(\frac{\partial^\nu f}{\partial x_1^{\alpha_1}\cdots\partial x_d^{\alpha_d}}\right)^2 dx_1\cdots dx_d.$$

The first part of the functional to be minimized controls the data fitting while the second part, $J_\nu^d(f)$, controls the smoothness. The trade-off between these two opposite goals is ensured by the choice of the smoothing parameter λ. The null space of $J_\nu^d(f)$ consists of polynomials with maximum degree of $(\nu-1)$. This subspace is of finite dimension $M = \binom{\nu+d-1}{\nu-1}$. Let us denote $\{\phi_1(.),\ldots,\phi_M(.)\}$ a basis of this subspace. If λ is known (and provided $\nu > d/2$ to ensure a continuous solution) the solution of the minimization problem is a TPS which has the following form:

$$g(x) = \sum_{j=1}^{M} \alpha_j \phi_j(x) + \sum_{i=1}^{n} \delta_i \eta_\nu^d\big(\|x - X_i\|\big)$$

where $\|.\|$ denotes the usual Euclidean norm. The vector $\delta \in \mathbb{R}^n$ of coefficients is subject to the constraint $T\delta = 0$ with the matrix T defined as $T_{ij} = \phi_j(X_i)$. Furthermore we have (where $\propto$ denotes proportional to):

$$\eta_\nu^d(r) \propto \begin{cases} r^{2\nu-d}\log(r) & d \text{ even}, \\ r^{2\nu-d} & d \text{ odd}. \end{cases}$$

To determine the vectors of coefficients α and δ, and thus the TPS solution, a closed form solution exists (see, for instance, Gu 2002). The TPS smoother can also be written as a linear smoother $S_\lambda Y$ where the dependency on d and ν is not written explicitly. Usually λ is unknown and has to be estimated from the data. Usual (classical) procedure is to minimize GCV criterion to determine an optimal $\hat{\lambda}$ that ensures the trade-off between smoothness and fitting. Moreover the order ν, which depends on unknown $m(.)$, is unknown and the classical approach is to choose an integer ν_0 without explicit statistical method to rely on. Usually it is chosen as the smallest integer value such as $\nu_0 > d/2$.

3.2 IBR with TPS

The approach proposed here is completely different: we deliberately choose a large λ (which is very easy) to ensure a very biased smoother. We choose ν_0 (as usual) the smallest integer value such as $\nu_0 > d/2$. But if the pilot estimator S_λ is a thin plate estimator of order $\nu_0 \le \nu$, then there exists an optimal number of iterations $k_n^\star$ such that the resulting smoother $\widehat{m}_k$ satisfies (Theorems 1 and 2 in Cornillon et al. 2011b)

$$\mathbb{E}\left[\left(\frac{1}{n}\sum_{j=1}^{n}\big(\hat{m}_{k_n^\star}(X_j) - m(X_j)\big)\right)^2\right] = O\big(n^{-2\nu/(2\nu+d)}\big).$$

While this existence theorem does not provide any practical guidance for finding the optimal number of iterations $k_n^\star$, it can be used in conjunction with Li (1987) to prove optimality of GCV stopping rule (see Cornillon et al. 2011b). Thus, IBR ensures adaptivity: we do not know the true ν but if we choose a ν_0 as usual we are sure to get the optimal rate

of convergence and the optimal number of iterations with GCV. Recall that the classical TPS does not ensure adaptivity. Moreover the choice of starting point for λ and the minimization procedure is greatly simplified in IBR framework compared to the classical TPS. Currently, the optimality of GCV have only been proven for TPS smoothers, although our simulations strongly suggest that a similar result must hold for kernel based smoothers.

3.3 IBR with Duchon splines

It is well-known that beside computational problems, TPS suffer from the fact that the dimension M_0 of the null space of $J^d_{\nu_0}(.)$ increases exponentially with d due to the condition $\nu_0 > d/2$. In his seminal paper Duchon (1977) presents a mathematical framework that extends TPS. Noting that the Fourier transform is isometric the smoothness penalty $J^d_{\nu_0}(f)$ can be replaced by its squared norm in Fourier space, that is,

$$\int \left\| D^{\nu_0} f(t) \right\|^2 dt \quad \text{can be replaced by}$$
$$\int \left\| \mathcal{F}\big(D^{\nu_0} f\big)(\tau) \right\|^2 d\tau.$$

In order to solve the problem of exponential growth of the dimension of the null space of $J^d_{\nu_0}(.)$, and to get new interpolation methods, Duchon introduced a weighting function to define a new smoothness penalty:

$$J^d_{\nu_0,s}(f) = \int |\tau|^{2s} \left\| \mathcal{F}\big(D^{\nu_0} f\big)(\tau) \right\|^2 d\tau.$$

The solution of the new variational problem:

$$\frac{1}{n} \left\| Y_i - f(X_i) \right\|^2 + \lambda J^d_{\nu_0,s}(f),$$

is

$$g(x) = \sum_{j=1}^{M_0} \alpha_j \phi_j(x) + \sum_{i=1}^{n} \delta_i \eta^d_{\nu_0,s}\big(\|x - X_i\|\big),$$

provided that $\nu_0 + s > d/2$ and $s < d/2$. The $\{\phi_j(x)\}$ are still a basis of the subspace spanned by polynomial of degree $\nu_0 - 1$. We also have that:

$$\eta^d_{\nu_0,s}(r) \propto \begin{cases} r^{2\nu_0+2s-d} \log(r) & d \text{ if } 2\nu_0 + 2s - d \text{ is even,} \\ r^{2\nu_0+2s-d} & d \text{ otherwise} \end{cases}$$

still with the same constraint on coefficients: $T\delta = 0$.

For the special case $s = 0$, the Duchon splines reduces to the TPS. But if one wants to have a lower dimension for the null space of $J^d_{\nu_0,s}$, for instance a pseudo-cubic splines with an order $\nu_0 = 2$, one can choose (as suggested by Duchon 1977) $s = \frac{d-1}{2}$.

The same problem of the determination of λ exists for Duchon splines. It can be solved by using $\hat{\lambda}$ which optimizes the GCV criterion. This classical framework is implemented in the R package **mgcv**. In some circumstances (depending on the sample size n, on the dimension d, on the design, on m the unknown regression function and on the error distribution) **mgcv** optimization procedure fails and the user has to use low rank splines (see details in Wood 2003). To our knowledge, no data-driven method in **mgcv** is proposed to help the user in the choice of a sensible rank, but the **mgcv** methods are rather insensitive to this choice.

Obviously, as Duchon splines solution are of the same form as TPS, IBR method with Duchon splines base smoother can be used to circumvent the problem of TPS in high dimension. No choice of rank is needed and the optimization procedure to get an optimal number of iterations is straightforward.

3.4 Initial values of IBR

As discussed previously, the IBR method relies on the choice of a pilot smoother that over-smooths the data. In this section we discuss the choice of the smoothness of the pilot S. Our discussion in the section distinguishes splines (thin-plate or Duchon) based smoother and kernel based smoothers.

Splines smoothers depend on a regularization constant that pre-multiplies the roughness penalty. Qualitatively, "large" values of λ lead to over-smoothing the data whereas "small" values of λ produce under-smooth of the data. What value to take for large and small depend on the design, and it is difficult to define a range of value for λ that over-smooth every dataset without considering the data. Instead of focusing on selecting λ, every smoothing package (with splines smoother) defines and uses an equivalent degree of freedom (edf), taken to be the trace of the smoothing matrix, that is loosely interpreted as the number of independent parameters needed to represent the smoother.

Consider S the smoothing matrix of a splines smoother. The first M_0 eigen-values are equal to one (corresponding of the null space of $J^d_{\nu_0,s}(f)$) and the other eigen-values are all positive and depend on the value of the smoothing parameter λ. Hence

$$\mathrm{tr}(S) = \mathrm{edf} = M_0 + \mathrm{function}(\lambda).$$

As the end-user may not readily know the value of M_0, the requested argument `df` in **ibr** is not the edf itself, but a multiplicative coefficient applied to M_0 to get the edf, i.e., $\mathrm{edf} = M_0 \times \mathtt{df}$. Thus `df` should be chosen greater than 1 to ensure that $\mathrm{edf} > M_0$.

Let us give an example: suppose $d = 5$.

- If the user wants to use TPS ($s = 0$), to ensure continuity, the package requires that at least $\nu_0 = \lfloor d/2 \rfloor + 1 = 3$ and $M_0 = \binom{\nu_0+d-1}{\nu_0-1} = 21$. If `df` is chosen equal to 1.1, the initial TPS of order ν_0 will use a smoothing parameter λ whose trace equals 23.1.

- If the user wants to use Duchon splines, the package will set as default value the pseudo-cubic splines setting: $\nu_0 = 2$ and $s = (d-1)/2 = 2$. This setting leads to $M_0 = 6$. If `df` is chosen equal to 1.1, the initial Duchon splines of order 2 will use a smoothing parameter λ whose trace equals 6.6. Duchon base smoother is obviously more smooth than the TPS base smoother.

As an aside, starting with different `df` values leads to the same solution. Therefore we only report the results with $df = 1.001$.

For TPS, the dimension of the null space of the smoothness penalty M_0 grows exponentially with the number of covariates d for continuous thin plate splines. As a result, these smoothers may not be able to over-smooth the data even in moderate dimensions d. For example, if $d = 8$ then the minimal value for $M_0 = 792$, and it is obvious that in such a case, one needs to have a larger number of observations (at least $n = 792$) to simply be able to compute the smoother. Thus, to have the very smooth pilot required by our method, several thousands of data may be needed and this kind of large datasets is beyond the scope of our paper. This difficulty does not arise with the Duchon splines or kernel smoothers.

3.5 IBR with kernel smoother

Kernel smoothers do not suffer this kind of limitations and can be used with various (moderate) dimension d. In general, multivariate kernel smoothers are governed by a vector of bandwidth, one bandwidth for each explanatory variable. We recall the reader that we do not seek to select an "optimal" bandwidth, just some reasonable one that guarantees that the initial smoother over-smooths the data. As for smoothing splines, our implementation abstracts the particulars of the smoothing parameter (in this case the bandwidth) in favor of the edf. We can get a reasonable pilot smoother by using a single bandwidth on each variable if we standardize the data. Our experience suggests that we obtain better results by selecting a bandwidth that makes each one dimensional smoother (in each variable) having the same small effective degree of freedom, which is the `df` argument. Values of `df` we found to work well in our examples are 1.05 and 1.1.

4 Simulations

In this section, we present some of the results by applying our bias reduction procedure to simulated data sets and compare the results with various competing procedures implemented in R. Our comparisons vary the sample sizes ($n =$ 50, 100 and 200), the pilot smoothers, the noise over signal ratio, the type of errors and consider various functions in $\mathbb{R}^2$, $\mathbb{R}^5$ and $\mathbb{R}^{10}$. Let us expose the settings (all the codes are available on the authors' webpages).

4.1 Settings

4.1.1 Errors distribution

The distribution of errors ε is Gaussian and its variance is chosen such that the noise over signal ratio ($\mathrm{var}(\varepsilon)/\mathrm{var}(m)$) is 5 %, 10 % and 20 %. Each sample of the explanatory variables X_j ($1 \le j \le d$) is drawn uniformly and independently on $[0, 1]$.

4.1.2 Evaluations

Usually, in simulation studies, it is possible to evaluate the error between the true function and the estimator on a grid. Evaluating the error on such a grid in dimension 1 or 2 is easy but it becomes computationally intensive in dimension 5 or 10. For example in dimension 5, a regular grid with 10 points in each direction requires 10^5 points to evaluate the error. Therefore, we propose two measures of the error: the classical Mean Square Error (MSE) and the Mean Square Prediction Error (MSPE). We choose randomly 10 % of the data in the sample (excluding the extreme points in each direction) and denote this test set $\mathcal{T}$. The remaining 90 % of the data (denoted $\mathcal{L}$) is used to estimate m. The MSE is calculated as follows:

$$\mathrm{MSE} = \frac{1}{|\mathcal{L}|} \sum_{j \in \mathcal{L}} \big(\hat{m}(X_j) - m(X_j)\big)^2.$$

We compute the MSPE on the remaining 10 % (the test set denoted by $\mathcal{T}$):

$$\mathrm{MSPE} = \frac{1}{|\mathcal{T}|} \sum_{j \in \mathcal{T}} \big(\hat{m}(X_j) - m(X_j)\big)^2.$$

This measure gives an insight on the behavior of the smoothers between data points as the distance between data points increases with the dimension d.

4.1.3 Competitors

We use for kernel smoother different values of `df` argument, but only one `df` argument for TPS smoother and Duchon smoother (see Sect. 3.4). We compare with smoothers having an R package available: linear models, MARS algorithm of Friedman (1991) as implemented in the R package **mda**, projection pursuit regression using the R function **ppr** where the number of components is chosen by data splitting, additive models instantiated in the R package **mgcv**, low rank Duchon splines and classical TPS as implemented in **mgcv**, additive Boosting Bühlmann and Yu (2003) from **mboost**, and regression trees Breiman et al. (1984) found in the R package **rpart**.

4.1.4 Replicates

We replicate every setting 500 times. However, for iterative smoothing procedures such as IBR, GAM, or maximization procedures such as those based on the choice of an optimal $\hat{\lambda}$ by GCV (low rank TPS, Duchon splines in **mgcv**), it could happen (hopefully in a small number of cases) that the proposed estimator is not well conditioned and huge errors can occur. Such problem could easily be fixed by analyzing the results one by one. However, since we are computing hundreds and hundreds of runs, this is impossible so we decide to exclude for each method its 5 % poorest runs. Therefore all the results are computed with 475 replications.

4.1.5 Results

For each method, we calculate the mean and the standard deviation of the 475 mean square prediction errors. To help with the comparison of the different methods, we divide each value by the smallest one among all the methods which we interpret as a relative efficiency of a method against the best method: this gives the value one to the most efficient method. In almost all the simulations, the method having the smallest error considering the mean has also the smallest variance.

As the level of noise is increasing, the difference between smoothers are decreasing, so we only present the 10 % case. According to our simulations, ranking among MSE or MSPE are relatively the same, so we only present MSPE tables but will have discussions on the difference between MSE and MSPE for a limited number of smoothers.

4.2 Dimension two

In dimension two, we consider one additive function m_1 and three functions with interaction. All the results are given in Table 1 for MSPE.

$m_1(x_1, x_2) = 10(x_1 - 0.5)^2 + 5\exp\big(-(x_2 - 0.3)^2/0.09\big)$
(additive),

$m_2(x_1, x_2) = 10x_1^2 + \exp(2x_2)\{x_1 < 0.5\} + \exp(2x_2)$
(not smooth),

$m_3(x_1, x_2) = 10x_1x_2^2 + 2$
(pure interaction),

$$m_4(x_1, x_2) = 40\frac{\exp(8(x_1 - 0.5)^2 + (x_2 - 0.5)^2)}{\exp(8((x_1 - 0.2)^2 + (x_2 - 0.7)^2))}$$
(complex interaction).

As expected, GAM modeling (**mgcv**) and **gamboost** have the lowest MSPE for the additive function with a slight advantage for **mgcv** package. True nonparametric multivariate modeling such as MARS, IBR, low rank TPS or Duchon splines (**mgcv**) give similar results which are not that far from GAM.

For the other functions, IBR with TPS or Duchon splines are performing as good as low rank TPS or Duchon splines using **mgcv** package, and these 5 competitors are performing much better than structural ones. Notice that modifying the argument `df` can lead to very small improvement for kernel pilot smoother. This suggests that IBR is robust to the choice of `df`.

4.3 Dimension five

In dimension five, we decide to use an additive function with an interaction, a single index function, a three index function and a function with interactions:

$$m_1(x_1, \ldots, x_5) = 10\sin(\pi x_1x_2) + 20(x_3 - 0.5)^2 + 10x_4 + 5x_5,$$

$$m_2(x_1, \ldots, x_5) = 10\exp\left(\frac{(2(x_1 - 0.5) - 2(x_2 - 0.5) + 2(x_3 - 0.5) + x_4 + x_5 - 1)^2}{5}\right),$$

$$m_3(x_1, \ldots, x_5) = 3\frac{(x_1 + 2x_4)}{\sqrt{5}} - 22\cos\left(\frac{\pi}{2}\frac{x_3 + 2x_5}{5}\right) + 10\exp\left(\frac{(2(x_1 - 0.5) - 2(x_2 - 0.5) + 2(x_3 - 0.5) + x_4 + x_5 - 1)^2}{5}\right),$$

$$m_4(x_1, \ldots, x_5) = 6\sin(\pi x_1x_2) + 10\cos\left(\sqrt{(x_3^2 + x_4^2)}\right) + 10x_4x_5.$$

Results of simulations are summarized in Table 2.

As in dimension 2, IBR (kernel and splines), **mgcv** low rank TPS and Duchon Splines globally outperform the other smoothers. As expected, it can be noticed that the relative difference is increasing with n but decreasing with d. But it comes as a relative surprise that even for moderate sample

Table 1 Dimension 2: ratio of the mean and variance of the MSPE over 475 simulations divided for all the competitors by the smallest value (mean or variance)

	R packages		**stats**	**rpart**	**stats**	**mda**	**mgcv**	**mboost**	**ibr**				**mgcv**	
$d=2$	n		par	tree	ppr	mars	gam	gamb	K1.05	K1.1	S	DS	ds	tps
m_1	50	me	19	12	4.6	1.9	1	1	2.3	2	2	1.8	1.8	2
		sd	16	11	5.7	1.8	1	1	2.7	2.4	2.3	2	2	2.3
	100	me	40	16	6.3	1.7	1	1.1	2.6	2.3	2	1.7	1.9	2.2
		sd	27	12	11	1.4	1	1	2.9	2.3	2	1.7	1.7	2
	200	me	73	23	5.7	1.8	1	1	2.4	2.5	1.9	1.7	1.9	2.5
		sd	44	16	17	1.6	1	1	2.4	2.5	1.7	1.5	1.7	2.1
m_2	50	me	3.2	5	2.4	1.9	1.8	1.9	1.4	1.5	1	1.1	1	1
		sd	2.4	3.5	2.2	1.4	1.4	1.4	1.3	1.5	1	1.1	1	1
	100	me	4.5	4.2	3.1	2.5	2.2	2.3	1.5	1.5	1	1.1	1.1	1
		sd	2.6	2.5	2.4	1.5	1.4	1.4	1.3	1.4	1	1.1	1.1	1
	200	me	5.8	4.2	3	2.9	2.6	2.7	1.4	1.4	1	1.1	1.1	1
		sd	3.1	2.4	2.1	1.6	1.3	1.4	1.2	1.3	1	1.1	1	1
m_3	50	me	9.7	21	1.9	10	9	9.2	1.1	1.1	1.2	1.1	1	1.2
		sd	7.8	22	1.6	8.5	7.9	8.8	1.2	1.2	1.3	1.1	1	1.2
	100	me	20	22	3.1	19	18	19	1	1	1.5	1.2	1.1	1.5
		sd	15	20	2.9	15	14	16	1.1	1.1	1.4	1.2	1	1.3
	200	me	43	26	3.7	39	37	38	1	1	1.7	1.3	1.3	1.9
		sd	25	20	3.2	24	22	24	1	1.1	1.3	1.1	1	1.4
m_4	50	me	7.2	9.6	2.6	5.8	5	4.9	1.2	1.2	1.1	1.1	1	1.1
		sd	6.8	11	2.9	5.7	5.1	5.2	1.2	1.4	1.1	1	1	1.2
	100	me	15	12	4.4	9.5	9	8.8	1	1.1	1.2	1.1	1	1.1
		sd	14	16	5.3	9.1	8.6	8.8	1	1.1	1.1	1	1	1.1
	200	me	29	11	6.9	19	18	18	1	1.1	1.4	1.2	1.2	1.4
		sd	25	12	8	16	16	16	1	1.1	1.3	1.1	1.2	1.3

size ($n=50$ that is $|\mathcal{L}|=45$ in learning set) the approximation error of GAM modeling can't be balanced by its low estimation error (compared to fully nonparametric modeling).

However, for the first function, which is nearly additive, GAM performs better than IBR for small sample size ($n=50$ and $n=100$). As n increases, IBR kernel performs better. One can think when n is sufficiently big, IBR kernel yields a better estimation of the slight interaction than the other smoothing procedures.

Roughly speaking the performances of IBR Duchon Splines or low rank **mgcv** Duchon Splines are similar. In Table 2, IBR appears to have a slight advantage but this slight advantage is due to the universal choice of the rank for the low rank Duchon Splines (arbitrarily chosen equal to $n/3$). A manual investigation shows that fine-tuning this choice of rank can lead to an advantage of low rank **mgcv** Duchon Splines over IBR Duchon Splines on MSE at the cost of increasing the MSPE. These two methods remain always in the same range and advantage will differ with the noise level, the rank or the type of function $m(.)$. Obviously, a good choice of rank in low rank smoothing splines can lead to improvement but this is beyond the scope of this paper. Again, the IBR procedure is robust to the choice of initial `df` in dimension five.

All these phenomena can also be observed with the MSE instead of MSPE. Moreover, when the noise level increases the differences between IBR and **mgcv** Duchon splines vanish. The complete results are available on the authors' webpage.

As a conclusion, in dimension two or five, without information on the structure of the regression function, one could advocate the use of IBR or low rank Duchon splines using **mgcv** package. Moreover, one can think that the distance between IBR (or low rank Duchon splines) and GAM gives an idea of the additivity of the function $m(.)$.

4.4 Dimension 10

Let us consider the same functions as in dimension five and just add five superfluous explanatory variables which are pure noise. Results are summarized in Table 3. The

Table 2 Dimension 5: ratio of the mean and variance of the MSPE over 475 simulations divided for all the competitors by the smallest value (mean or variance)

	R packages		**stats**	**rpart**	**stats**	**mda**	**mgcv**	**mboost**	**ibr**				**mgcv**	
$d=5$	n		par	tree	ppr	mars	gam	gamb	K1.05	K1.1	S	DS	ds	tps
m_1	50	me	2.1	6.3	2.8	1.6	1	1.6	1.4	1.3	1.7	1.3	1.8	–
		sd	1.9	4.9	2.7	1.6	1	1.4	1.6	1.4	2.2	1.3	1.8	–
	100	me	2.8	6.2	3.4	1.4	1	1.3	1	1	1.1	1	1.2	2.9
		sd	2.3	4.6	3.1	1.3	1	1.1	1.2	1.2	1.2	1.1	1.3	4
	200	me	5.4	9.9	4.9	2.3	1.9	2.1	1	1	1.3	1.1	1.2	1.3
		sd	4.5	7.3	5.1	2	1.7	1.7	1	1	1.3	1.1	1.2	1.3
m_2	50	me	5.4	6.1	2.2	5.5	5.9	4.6	1	1	1.1	1.1	1.8	–
		sd	5.2	5.5	2.8	5.2	5.3	4.7	1	1	1.1	1	1.7	–
	100	me	8.4	9.2	1.9	8.9	8.2	7.7	1.2	1.1	1	1.2	1.2	4.8
		sd	8.3	7.7	2.7	7.9	7.4	7.5	1.3	1.2	1	1.3	1.4	4.4
	200	me	12	12	1.5	12	11	11	1.4	1.4	1	1.3	1.3	2.1
		sd	11	9	2.2	9.6	9.3	9.6	1.3	1.3	1	1.4	1.5	1.7
m_3	50	me	5.1	6	2.6	5.7	5.5	4.8	1	1	1.1	1	1.7	–
		sd	5.1	5.2	2.8	5	5.3	4.7	1	1	1.1	1	1.6	–
	100	me	8.2	9.6	2.2	9.2	8	7.9	1.2	1.1	1	1.2	1.2	5
		sd	8.3	7.9	2.5	7.9	7.4	7.7	1.3	1.2	1	1.3	1.4	4.6
	200	me	12	13	1.6	12	11	11	1.4	1.4	1	1.3	1.3	2.1
		sd	11	9.5	2.2	9.5	9.3	9.6	1.3	1.3	1	1.4	1.5	1.6
m_4	50	me	2	5.9	2.5	2.1	1.6	2.1	1.2	1.2	1.5	1	1.1	–
		sd	1.7	4.7	2.2	1.9	1.4	1.9	1.3	1.2	1.9	1	1.2	–
	100	me	3	6.9	3.6	2.5	2.2	2.4	1.1	1.1	1.1	1	1.1	3
		sd	2.3	4.9	3.2	2	1.7	1.7	1.1	1.1	1.1	1	1.1	3.8
	200	me	5.6	10	5	4.3	3.9	4	1	1	1.3	1.1	1.1	1.3
		sd	4.2	7.7	5	2.9	2.7	2.8	1	1	1.3	1.1	1.2	1.4

minimum effective degree of freedom for thin plate splines smoother will be $M_0 = 6188$ which is far greater than the number of observations n. Thus thin plate splines smoother cannot be used in dimension 10.

Recall that we have 10 variables with only five active variables and five vacuous variables. This fact is unknown to the users. We construct an initial smoothing matrix using the 10 variables and iterate. So at each step of the algorithm all the variables (even the superfluous ones) are used. The results are not that different than those obtained in the previous section.

The main conclusion of that section is that the nonparametric methods (IBR and **mgcv** Duchon Splines) gives (i) similar results as GAM modeling (**mgcv** package) for nearly additive function (ii) much better results than GAM for non-additive function, even for very small sample in moderate dimension. This fact comes as a surprise as the common wisdom is to advocate structural modeling with a small sample sizes and moderate dimension.

Nonparametric methods appear relatively robust to the possible addition of pure noise variables to the set of explanatory variables. However, potential gains could be obtained if the initial smoothing matrix only contains the variables of interest or at least the variables of interest plus a limited number of unrelated variables.

5 Nonparametric smoothing with variable selection

The IBR method starts with a pilot smoother S for all the explanatory variables $X_1, \ldots, X_d$, and then iterates that smoother. But if some explanatory variables are not related to Y, it seems intuitively clear that excluding them should improve the predictive capability of the smoother. This suggests that variable selection may be beneficial. For computational reasons, we advocate using ascendent variable selection to construct more parsimonious multivariate smoothers.

To proceed with this variable selection procedure, we need to choose a criterion. Classical criterion for variable selection in linear models are AIC or BIC. The GCV criterion, which is well suited for splines smoothing, is also available. For example, let us assume that the selected criterion is BIC. Our forward variable selection procedure starts by building

Table 3 Dimension 10: ratio of the mean and variance of the MSPE over 475 simulations divided for all the competitors by the smallest value (mean or variance)

	R packages		**stats**	**rpart**	**stats**	**mda**	**mgcv**	**mboost**	**ibr**			**mgcv**
$d = 10$	n		par	tree	ppr	mars	gam	gamb	K1.05	K1.1	DS	ds
m_1	50	me	1.7	4.3	2.5	1.3	1	1.3	1.6	1.6	1.6	1.7
		sd	1.5	3.6	2.2	1.3	1	1.2	1.5	1.5	1.5	1.5
	100	me	2.5	5.3	3.3	1.2	1	1.2	2.2	2.2	2.2	2.5
		sd	2.4	4.7	3.6	1.3	1	1.1	2.4	2.3	2.4	2.5
	200	me	2.7	4.9	3.2	1.1	1	1.1	1.7	1.7	1.7	1.9
		sd	2.7	4.1	3.5	1.1	1	1	1.9	1.9	1.8	2.1
m_2	50	me	1.8	1.6	1.8	1.7	2	1.3	1	1	1.1	1.6
		sd	1.6	1.4	2.2	1.6	1.6	1.4	1	1	1.1	1.5
	100	me	2.9	3.2	1.5	3.1	3	2.4	1	1.1	1.1	1.9
		sd	3.2	3.3	2.3	3.2	3	2.8	1	1.2	1.1	1.9
	200	me	5.2	5.4	1.3	5.4	5.1	4.6	1	1.1	1.1	1.1
		sd	4.6	4.1	1.7	4.2	4.1	4.1	1	1.2	1.1	1.2
m_3	50	me	1.7	1.6	1.9	1.8	1.9	1.3	1	1	1.1	1.6
		sd	1.6	1.4	2.1	1.7	1.6	1.3	1	1	1	1.5
	100	me	2.8	3.3	1.6	3.2	2.9	2.6	1	1.1	1.1	1.8
		sd	3.1	3.3	2.3	3.2	2.9	2.9	1	1.2	1.1	1.9
	200	me	5.1	5.4	1.1	5.5	5	4.6	1	1.1	1.1	1.1
		sd	4.6	4	1.3	4.2	4.1	4	1	1.2	1.1	1.2
m_4	50	me	1.2	3.1	1.7	1.1	1.1	1.2	1.1	1.1	1	1.1
		sd	1.1	2.7	1.8	1.2	1.1	1.2	1	1	1	1.1
	100	me	1.4	3.3	2	1.2	1.1	1.1	1.1	1.1	1	1.1
		sd	1.3	2.9	2.2	1.3	1	1	1.1	1.1	1	1.1
	200	me	1.9	3.5	2.2	1.4	1.3	1.4	1.1	1.1	1	1.1
		sd	1.6	2.9	2.2	1.2	1.1	1.1	1.1	1.1	1	1.1

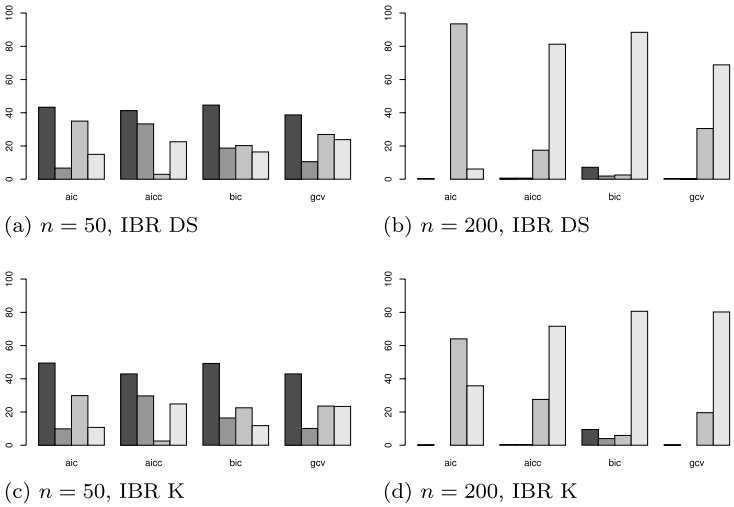
d univariate smoothers, one for each of the explanatory variables, and each smoother with the same equivalent degree of freedom. We apply the IBR algorithm to each of these univariate smoothers and select their respective optimal number of iterations, using GCV. Of these d smoothers, we select the one with the smallest BIC, and fix that variable. Next, we consider all $d - 1$ bivariate smoothers that include the previously selected variable. Again, we apply the IBR algorithm and consider the bivariate model with the smallest BIC. If the latter BIC value is larger than the smallest BIC value from the univariate fit, we stop the forward fitting selection and return the univariate smoother. If not, then we consider all $d - 2$ trivariate smoothers that extend the "best" bivariate smoother. We proceed with the forward selection until no improvement in the BIC is observed. This forward selection procedure has been implemented within the **ibr** package (function **forward**).

5.1 Criteria for variable selection

Here, we report on the performance of our variable selection method using the simulated data in Sect. 4.4. Again, we consider kernel based smoothers and Duchon splines smoother as our model may contain up to 10 variables, which makes TPS not practical. To help understand the results and provide further insights into the qualitative behavior of variable selection for IBR smoothing, we analyze the selected model as follows: We roughly divide the selection results into four categories:

- First category: the variable selection criterion leads to a selected model which misses some of the true variables and include some other which are pure noise ("wrong" category).
- Second category: the variable selection criterion leads to a selected model which misses only some true variables ("not enough" category).
- Third category: the variable selection criterion leads to a selected model which includes all the true variables and some other ("too many" category).
- Fourth category: the variable selection criterion leads to a selected model which is the good one ("exact" category).

As similar results were obtained for the other functions we only present the results for function 2.



(a) $n = 50$, IBR DS (b) $n = 200$, IBR DS

(c) $n = 50$, IBR K (d) $n = 200$, IBR K

Fig. 2 Variable selection features for the function 2. The barplot shows the percentage of occurrences of each category: category one ("wrong") *dark*, two ("not enough") *dark grey*, three ("too many") *grey* and four (exact) *light grey*

As shown in Fig. 2, the percentages of the first category are roughly the same for all the variable selection criteria when n is small. However when n is bigger, that percentage is bigger for BIC. The second category leads to models with poor prediction accuracy. It can be seen that the percentage of this category for BIC is greater than those obtained by the other criteria. That can partly explain the poor prediction accuracy of models selected with BIC because that criterion tends to select more parsimonious models. The percentages of the third category reveals that AIC (especially) and GCV (to a fewer extend) tend to select too many variables (compared to BIC). But since IBR is somewhat robust to the inclusion of a few vacuous variables (see Sect. 4.4), this does not appear to overly degrade the predictive capability of the resulting IBR smoother. The fourth category has a higher percentage for GCV (if using kernel pilot smoother) compared to AIC (and BIC) and explains why GCV is better at selecting good models for prediction. In conclusion, we advocate to use again GCV criteria in our function **forward**.

5.2 Simulation results for variable selection

In Table 4, we compare IBR with variable selection (using GCV) to its competitors available in R: the **leaps** package for classical multivariate regression, **mars**, **gam**, **gamboost** with their built-in selection procedure and ppr.

The conclusion are about the same as in dimension 5: IBR gives the best results except for the first function (nearly additive) with $n = 50$. Again, the argument `df` seems unimportant for the (kernel) pilot smoother: IBR is robust to the choice of `df`. Compared to dimension 5, it can be noticed that the variable selection slightly reduces the differences.

6 Real data example: Los Angeles ozone data

As a real world example, consider the classical data set of ozone concentration in the Los Angeles basin. This is as a standard dataset for comparing the performance of multivariate smoothers (Breiman 1996; Bühlmann and Yu 2003). The sample size of the data is $n = 330$ and the number of explanatory variables $d = 8$ (Pressure, Wind speed, Humidity, Temperature measured at Sandburg, Temperature measured at El Monte, Inversion base height, Pressure gradient, Inversion base temperature and Visibility). We compare our iterative bias procedure with existing methods.

We estimate mean squared prediction error $\mathbb{E}[(Y - \hat{m}(X))^2]$ by randomly splitting the data into 297 training observations and 33 test observations and averaging 50 times over such random partitions.

For the IBR smoother, we use a multivariate Gaussian kernel and select the bandwidth so that the univariate smoother in each of the variables has the same trace, i.e., the same effective degree of freedom, $df = 1.1$. We do so at each iteration.

For Duchon or IBR Duchon, since all the variables are not of the same range, we decide to scale the variables, again this is done at each iteration. The training part is scaled, the smoothers are evaluated and new values (centered by the mean and divided by the standard error evaluated on the training set) are predicted. Figure 3 summarizes the results.

Low rank Duchon Splines and ibrDS perform better than the others methods and lead to a reduction of the mean prediction error of more than 15 % over competing multivariate methods. Recall that L_2 boosting is a component-wise procedure (Friedman 2001) ideally suited to fitting constrained models such as additive models (Bühlmann and Yu 2003). In order to deal with possible interactions, Bühlmann and Yu (2006) include second order and quadratic interaction

Table 4 Ratio of the mean and variance of the errors over 475 simulations divided for all the competitors by the smallest value (mean or variance)

	R packages		**stats**	**stats**	**mda**	**mgcv**	**mboost**	**ibr**			**mgcv**
$d = 10$	n		par	ppr	mars	gam	gamb	K1.1	K1.3	DS	ds
m_1	50	me	1.7	3	1.5	1	1.6	1.6	1.5	1.7	2
		sd	1.8	2.7	1.6	1	1.5	1.9	1.8	1.7	1.9
	100	me	2.5	3.5	1.3	1	1.3	1.1	1	2.3	2.6
		sd	2.4	3.9	1.4	1	1.2	1.5	1.4	2.3	2.7
	200	me	5.2	6.3	2.2	1.9	2.2	1	1.1	4.7	3.8
		sd	4.1	5.4	1.7	1.5	1.5	1	1.1	3.8	3.2
m_2	50	me	1.8	2.4	2.2	2	1.6	1	1.4	1.4	2.1
		sd	1.5	2.3	1.7	1.6	1.5	1	1.5	1.3	1.6
	100	me	4.8	2.9	5.9	5	4.6	1	1.3	3.4	3.6
		sd	4.1	3.2	4.6	3.9	4	1	1.5	3.3	2.7
	200	me	8.2	2.1	9	7.9	7.6	1.1	1	5.7	1.9
		sd	6.9	2.6	6.4	6.2	6.2	1.1	1	5.4	1.8
m_3	50	me	2	2.4	2.3	2	1.7	1	1.3	1.5	2.1
		sd	1.7	2.4	1.9	1.6	1.5	1	1.3	1.4	1.7
	100	me	6.2	3.6	7	5.7	5.6	1	1.3	4	4
		sd	5	3.9	5.4	4.5	4.7	1	1.5	3.7	3.1
	200	me	9.1	1.8	9.4	8.1	8	1.1	1	5.9	1.9
		sd	7.1	2.1	6.7	6.5	6.3	1.1	1	5.5	1.8
m_4	50	me	1.8	2.3	1.5	1.2	1.6	1.1	1	1.1	1.5
		sd	1.6	2.1	1.3	1	1.4	1.2	1	1	1.3
	100	me	2.8	3.6	2.3	1.9	2.1	1	1	2	2.1
		sd	2.2	3.1	1.8	1.3	1.4	1	1	1.5	1.5
	200	me	5.6	6.6	4.2	3.8	4	1	1.1	4.3	3.1
		sd	4	5.4	3.1	2.6	2.6	1	1.1	3.2	2.6

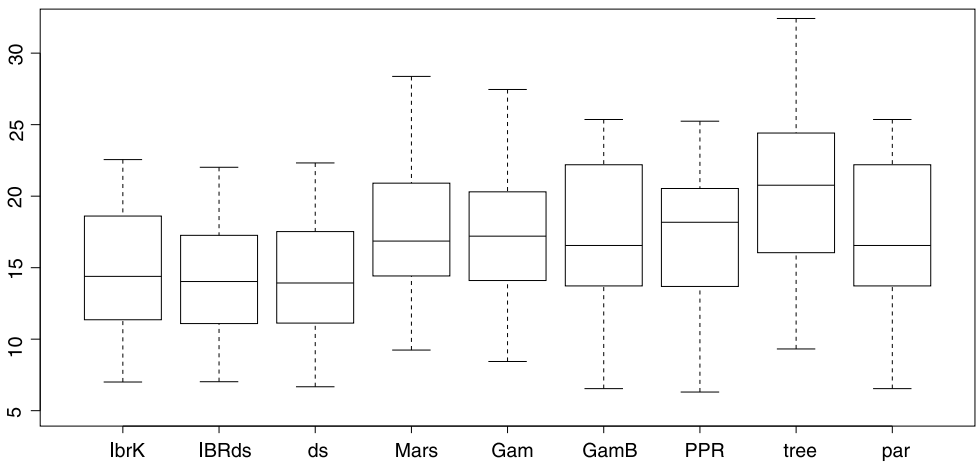


Fig. 3 Boxplot of the MPE for the different competing methods

terms within the L_2 boosting framework. The inclusion of higher order interaction terms increases the number of explanatory variables from 8 to 45. With the interaction terms included, the L_2 boosting proposed by Bühlmann and Yu (2003) yields an out of sample prediction MSE of 15.60. This result is obtained by discarding the two gross outliers in the 50 random partitions. Without removing these outliers, their results remain consistent with the results published in Bühlmann and Yu (2006). Our iterative bias reduction procedure is fully multivariate and finds directly an estimation of $m(X_1, \ldots, X_d)$ (where $d = 8$). As its results are better than additive models or models with low order interactions,

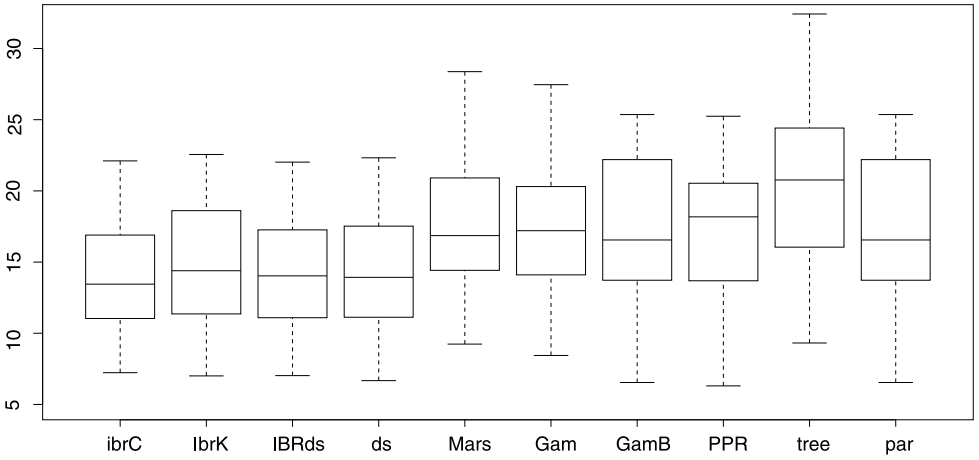


Fig. 4 Boxplot of the MPE for the different competing methods and the variable selection procedure

we can conclude that interaction of high order is significant for this dataset.

The previous results were obtained using all the 8 covariates. Further improvements are possible using variable selection. We apply our forward variable selection method to each of the 50 randomly split data (into 297 observations to fit the model and 33 observations to validate the predictions) to select the predictive variables using the GCV criterion. With high consistency, the procedure selects the 5 variables Wind, Humidity, Temp_Sand, Inv_Base_height, Pressure_Grad. Furthermore, with only these five variables, the mean predicted error drops to 13.8 (Fig. 4), which is a small improvement over the prediction error we had when using all the 8 variables.

7 Conclusion

Cornillon et al. (2011b) propose a new smoothing method IBR that has the desirable property of being simple and yet capable of adaptation, which suggests that it may be used to perform fully nonparametric smoothing in moderate dimensions. This paper compares this new method with classical and non-classical multivariate smoothing methods.

This simulation study shows that even for very moderate learning sample size (such as $n = 45$ or $n = 90$) in moderate dimension (up to $d = 10$) nonparametrics smoothers such as IBR (kernel or splines, package **ibr**) or low rank splines (Duchon or TPS, package **mgcv**) can lead to significant improvement over structurally constrained modeling such as GAM. These two kinds of modeling are very close in performances and can be thought as leading more or less the same results.

One can think that in the light of the results, the classical idea of quantifying the amount of non-additivity in the regression function $m(.)$ by measuring the distance between GAM modeling and a fully nonparametric modeling can be investigated in a practical manner.

Acknowledgements We would like to thank the associate editor and the referees for very valuable remarks and for pointing out to us the work of Duchon (1977).

References

Akaike, H.: Information theory and an extension of the maximum likelihood principle. In: Petrov, B.N., Csaki, B.F. (eds.) Second International Symposium on Information Theory, pp. 267–281. Academiai Kiado, Budapest (1973)

Breiman, L.: Bagging predictors. Mach. Learn. **24**, 123–140 (1996)

Breiman, L.: Using adaptive bagging to Debais regressions. Tech. Rep. 547, Department of Statistics, UC Berkeley (1999)

Breiman, L., Freiman, J., Olshen, R., Stone, C.: Classification and Regression Trees, 4th edn. CRC Press, Boca Raton (1984)

Bühlmann, P., Yu, B.: Boosting with the l_2 loss: regression and classification. J. Am. Stat. Assoc. **98**, 324–339 (2003)

Bühlmann, P., Yu, B.: Sparse boosting. J. Mach. Learn. Res. **7**, 1001–1024 (2006)

Buja, A., Hastie, T., Tibshirani, R.: Linear smoothers and additive models. Ann. Stat. **17**, 453–510 (1989)

Cornillon, P.A., Hengartner, N., Matzner-Løber, E.: Iterative bias reduction multivariate smoothing in R: the IBR package (2011a). arXiv:1105.3605v1

Cornillon, P.A., Hengartner, N., Matzner-Løber, E.: Recursive bias estimation for multivariate regression (2011b). arXiv:1105.3430v2

Craven, P., Wahba, G.: Smoothing noisy data with spline functions: estimating the correct degree of smoothing by the method of generalized cross-validation. Numer. Math. **31**, 377–403 (1979)

Di Marzio, M., Taylor, C.: On boosting kernel regression. J. Stat. Plan. Inference **138**, 2483–2498 (2008)

Duchon, J.: Splines minimizing rotation-invariant semi-norms in Sobolev spaces. In: Shemp, W., Zeller, K. (eds.) Construction Theory of Functions of Several Variables, pp. 85–100. Springer, Berlin (1977)

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

Fan, J., Gijbels, I.: Local Polynomial Modeling and Its Application, Theory and Methodologies. Chapman & Hall, New York (1996)

Friedman, J.: Multivariate adaptive regression splines. Ann. Stat. **19**, 337–407 (1991)

Friedman, J.: Greedy function approximation: A gradient boosting machine. Ann. Stat. **28**, 1189–1232 (2001)

Friedman, J., Stuetzle, W.: Projection pursuit regression. J. Am. Stat. Assoc. **76**, 817–823 (1981)

Friedman, J., Hastie, T., Tibshirani, R.: Additive logistic regression: a statistical view of boosting. Ann. Stat. **28**, 337–407 (2000)
Gu, C.: Smoothing Spline ANOVA Models. Springer, Berlin (2002)
Gyorfi, L., Kohler, M., Krzyzak, A., Walk, H.: A Distribution-Free Theory of Nonparametric Regression. Springer, Berlin (2002)
Hastie, T.J., Tibshirani, R.J.: Generalized Additive Models. Chapman & Hall, New York (1995)
Hurvich, C., Simonoff, G., Tsai, C.L.: Smoothing parameter selection in nonparametric regression using and improved Akaike information criterion. J. R. Stat. Soc. B **60**, 271–294 (1998)
Lepski, O.: Asymptotically minimax adaptive estimation. I: Upper bounds. Opitmally adaptive estimates. Theory Probab. Appl. **37**, 682–697 (1991)
Li, K.C.: Asymptotic optimality for C_p, C_L, cross-validation and generalized cross-validation: discrete index set. Ann. Stat. **15**, 958–975 (1987)
Ridgeway, G.: Additive logistic regression: a statistical view of boosting: discussion. Ann. Stat. **28**, 393–400 (2000)
Schwarz, G.: Estimating the dimension of a model. Ann. Stat. **6**, 461–464 (1978)
Simonoff, J.S.: Smoothing Methods in Statistics. Springer, New York (1996)
Tsybakov, A.: Introduction to Nonparametric Estimation. Springer, Berlin (2009)
Tukey, J.W.: Explanatory Data Analysis. Addison-Wesley, Reading (1977)
Wood, S.N.: Thin plate regression splines. J. R. Stat. Soc. B **65**, 95–114 (2003)
Wood, S.N.: Stable and efficient multiple smoothing parameter estimation for generalized additive models. J. Am. Stat. Assoc. **99**, 673–686 (2004)
Yang, Y.: Combining different procedures for adaptive regression. J. Multivar. Anal. **74**, 135–161 (2000)