Leitenstorfer, Tutz:

Smoothing with Curvature Constraints based on Boosting Techniques

Sonderforschungsbereich 386, Paper 467 (2006)

Online unter: http://epub.ub.uni-muenchen.de/

Smoothing with Curvature Constraints based on Boosting Techniques

Florian Leitenstorfer, Gerhard Tutz
Ludwig-Maximilians-Universität München
Akademiestraße 1, 80799 München
{leiten,tutz}@stat.uni-muenchen.de

16th January 2006

Abstract

In many applications it is known that the underlying smooth function is constrained to have a specific form. In the present paper, we propose an estimation method based on the regression spline approach, which allows to include concavity or convexity constraints in an appealing way. Instead of using linear or quadratic programming routines, we handle the required inequality constraints on basis coefficients by boosting techniques. Therefore, recently developed componentwise boosting methods for regression purposes are applied, which allow to control the restrictions in each iteration. The proposed approach is compared to several competitors in a simulation study. We also consider a real world data set.

Keywords: Shape constrained smoothing, Concavity, Regression splines, Boosting.

1 Introduction

Nonparametric regression methods provide widely used and powerful tools for analysts which are interested in not imposing a strictly parametric model on the data, but want the data to "tell" the underlying structure. However, often it is useful to incorporate prior knowledge of the shape of the underlying regression function, such as monotonicity.

In the present paper, we focus on another type of constraints, which are important especially in economics. For example, human capital theory predicts that the logarithm of wage is a concave function of experience, and economic theory assumes that the observed relationship between input and output will be concave and non-decreasing, when producers are maximizing profit. There are

different approaches for smoothers that can handle such curvature constraints. Delecroix, Simioni, and Thomas-Agnan [3, 4] propose to project an arbitrary consistent smoother onto a suitable cone in some function space for convex or concave estimates. Dierckx [6] suggests to restrict B-spline coefficients in a regression spline setting in order to enforce concavity, whereas He and Ng [8] outline a related procedure based on quantile regression methodology. A smoothing spline approach that allows for several types of shape constraints including convexity or monotonicity is given in [13]. An overview over various curvature constrained regression methods is given in [5].

The approach presented here is in the spirit of regression splines, i.e. we expand f into basis functions, $f = \sum \alpha_j B_j$. The restrictions which have to be imposed on the coefficients α_j for certain shape constraints depend on the properties of the basis functions B_j. In the case of curvature constraints, we suggest to use a truncated power series basis. The novelty compared to previous approaches based on regression splines (e.g. [6]) is that estimation of the coefficients is not based on common routines for solving linear or quadratic programming problems. Instead, the α_j are estimated by boosting. Bühlmann and Yu [2] propose a boosting algorithm constructed from the L_2-loss, which is suitable for high dimensional predictors in an additive model context. The extension of L_2Boost to the fitting of high dimensional linear models [1] can be adapted to the present context. Since basis functions are selected componentwise in a stepwise fashion, this procedure can be seen as a knot selection technique. The constraints on α_j are incorporated in the selection step.

The paper is organized as follows: in Section 2, the boosting algorithm for smoothing with concavity or convexity restrictions is given. Section 3 summarizes the results of a simulation study, and in Section 4 we apply the proposed method to a real world data set.

2 Curvature Constraints by Boosting

Consider a conventional nonparametric regression problem, i.e. for dependent variable y_i and covariate x_i, $i = 1, \ldots, n$, the model

$$y_i = f(x_i) + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, \sigma^2), \tag{1}$$

is assumed, where $f(.)$ is an unknown smooth function on the interval $[a, b] = [x_{min}, x_{max}]$. In the following, it is postulated that $f(.)$ is a concave function. A sufficient condition for concavity of $f(.)$ is

$$\frac{\partial^2}{\partial x^2} f(x) \leq 0 \quad \text{for} \quad x \in (a, b).$$

It follows immediately that a monotonic decreasing first derivative is sufficient for concavity of the function (for convexity, replace $\leq$ by $\geq$, which leads to an

increasing first derivative). In order to incorporate the curvature constraint into the estimation of $f(.)$, we suggest to expand $f(.)$ into a truncated power series basis of degree $q = 2$,

$$f(x) = \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \sum_{j=1}^{m} \alpha_{2+j}(x - \tau_j)_+^2, \tag{2}$$

where $\{\tau_j\}$ is a given sequence of knots. This expansion is continuously differentiable on (a, b) and twice differentiable on $(a, b)\backslash\{\tau_j\}$, with a second derivative given by

$$\frac{\partial^2}{\partial x^2} f(x) = 2\alpha_2 + \sum_{j=1}^{m} 2\alpha_{2+j} I(x > \tau_j), \tag{3}$$

where $I(.)$ denotes the indicator function. Since the first derivative of (2) is continuous, it suffices to ensure that (3) is non-positive on $(a, b)\backslash\{\tau_j\}$ to guarantee concavity. This property is quite easy to control, since (3) has the shape of a step function with jumps at the knots $\{\tau_j\}$. Thus, a sufficient condition on the vector of basis coefficients, $\boldsymbol{\alpha} = (\alpha_0, \ldots, \alpha_{2+m})'$ to fulfil the concavity condition is given by

$$\sum_{j=0}^{k} \alpha_{2+j} \leq 0 \quad \text{for} \quad k = 0, \ldots, m, \tag{4}$$

i.e. that starting from α_2, the consecutive sums of basis coefficients have to be non-positive.

In order to obtain estimates that fulfill restriction (4) we propose boosting techniques. Boosting has originally been developed in the machine learning community to improve classification procedures (e.g. [11]). With Friedman's [7] gradient boosting machine it has been extended to regression modeling (see [2], [1]). The basis concept in boosting is to obtain a fitted function iteratively by fitting in each iteration a "weak" learner to the current residual. When estimating smooth functions a weak learner is a fitting procedure that restricts the fitted model to low degrees of freedom. Componentwise boosting in the sense of [2] means that in one iteration, only the contribution of one variable is refitted. Boosting for curvature constrained fits uses a similar procedure, however componentwise does not refer to variables but to basis functions. Thus in each iteration, besides the intercept and the linear coefficient α_1, which is not under restriction, only the contribution of one basis function is updated. This update makes it easy to control the property (4). In order to allow for high flexibility of the fitting procedure we use a large number of basis functions. The procedure then automatically selects an appropriate subset of basis functions.

The weak learner we use is ridge regression ([9]) with basis functions as predictors. In matrix notation the data are given by $\mathbf{y} = (y_1, \ldots, y_n)'$, $\mathbf{x} = (x_1, \ldots, x_n)'$. The expansion into basis function yields the data set $(\mathbf{y}, \mathbf{B})$, where

$\mathbf{B} = (\mathbf{1}, B_1(\mathbf{x}), \ldots, B_{2+m}(\mathbf{x}))$, with $B_1(\mathbf{x}) = \mathbf{x}$, $B_2(\mathbf{x}) = \mathbf{x}^2$ and $B_{2+j}(\mathbf{x}) = (\mathbf{x} - \tau_j)_+^2$ for $j = 1, \ldots, m$. For convenience let $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_n)'$ with components $\mu_i = E(y_i|x_i)$ denote the vector of means.

CurveBoost

Step 1 (Initialization)

Standardize $\mathbf{y}$ to zero mean, i.e. set $\hat{\alpha}_0 = \bar{y}$, $\hat{\boldsymbol{\alpha}}^{(0)} = (\bar{y}, 0, \ldots, 0)'$ and $\hat{\boldsymbol{\mu}}^{(0)} = (\bar{y}, \ldots, \bar{y})'$.

Step 2 (Iteration)

For $l = 1, 2, \ldots$, compute the current residuals $\mathbf{u}^{(l)} = \mathbf{y} - \hat{\boldsymbol{\mu}}^{(l-1)}$.

1. *Fitting step*
   For $r = 0, \ldots, m$, let $\mathbf{B}_{(r)} = (\mathbf{1}, B_1(\mathbf{x}), B_{2+r}(\mathbf{x}))$. Compute the ridge regression estimator
   $$\hat{\boldsymbol{\alpha}}_{(r)} = (\mathbf{B}_{(r)}'\mathbf{B}_{(r)} + \lambda\boldsymbol{\Lambda})^{-1}\mathbf{B}_{(r)}'\mathbf{u}^{(l)},$$
   where $\hat{\boldsymbol{\alpha}}_{(r)} = (\hat{\alpha}_{0(r)}, \hat{\alpha}_{1(r)}, \hat{\alpha}_{2(r)})'$ and $\boldsymbol{\Lambda} = diag(0, 1, 1)$.

2. *Selection step*
   For $r = 0, \ldots, m$, compute the potential update of the basis coefficient $\hat{\alpha}_{2+r,new} = \hat{\alpha}_{2+r}^{(l-1)} + \hat{\alpha}_{2(r)}$ and check the concavity constraints from (4),
   $$\hat{\alpha}_{2+r,new} + \sum_{j \in \{0,\ldots,k\} \setminus \{r\}} \hat{\alpha}_{2+j}^{(l-1)} \leq 0, \quad k = r, \ldots, m.$$
   If the constraint is not satisfied for all r, stop. Otherwise, from the subset of $\{0, \ldots, m\}$ where the constraint is fulfilled, one chooses the component r_l that minimizes $||\mathbf{u}^{(l)} - \mathbf{B}_{(r)}\hat{\boldsymbol{\alpha}}_{(r)}||^2$.

3. *Update*
   Set
   $$\begin{aligned} \hat{\alpha}_0^{(l)} &= \hat{\alpha}_0^{(l-1)} + \hat{\alpha}_{0(r_l)}, \quad \hat{\alpha}_1^{(l)} = \hat{\alpha}_1^{(l-1)} + \hat{\alpha}_{1(r_l)}, \\ \hat{\alpha}_{2+j}^{(l)} &= \begin{cases} \hat{\alpha}_{2+j}^{(l-1)} + \hat{\alpha}_{2(r_l)}, & j = r_l, \\ \hat{\alpha}_{2+j}^{(l-1)}, & \text{otherwise}, \end{cases} \end{aligned}$$
   and
   $$\hat{\boldsymbol{\mu}}^{(l)} = \hat{\boldsymbol{\mu}}^{(l-1)} + \mathbf{B}_{(r_l)}\hat{\boldsymbol{\alpha}}_{r_l}.$$

In order to prevent overfitting, it is necessary to include a stopping criterion. An appropriate criterion is the AIC criterion which balances goodness-of-fit with the degrees of freedom. In order to use it in a smoothing problem, the hat matrix of the smoother has to be given. For the present procedure, it can be obtained in a similar way as for componentwise L2Boost in linear models, proposed by [1]. With $\mathbf{S}_l = \mathbf{B}_{(r_l)}(\mathbf{B}'_{(r_l)}\mathbf{B}_{(r_l)} + \lambda\mathbf{\Lambda})^{-1}\mathbf{B}_{(r_l)}$, $l = 1, 2, \ldots$ and $\mathbf{S}_0 = \frac{1}{n}\mathbf{1}_n\mathbf{1}'_n$, $\mathbf{1}_n = (1, \ldots, 1)'$, one has in the lth iteration

$$\hat{\boldsymbol{\mu}}^{(l)} = \hat{\boldsymbol{\mu}}^{(l-1)} + \mathbf{S}_l\mathbf{u}^{(l)} = \hat{\boldsymbol{\mu}}^{(l-1)} - \mathbf{S}_l(\hat{\boldsymbol{\mu}}^{(l-1)} - \mathbf{y}),$$

and therefore

$$\hat{\boldsymbol{\mu}}^{(l)} = \mathbf{H}_l\mathbf{y},$$

where

$$\mathbf{H}_l = \mathbf{I} - (\mathbf{I} - \mathbf{S}_0)(\mathbf{I} - \mathbf{S}_1)\cdots(\mathbf{I} - \mathbf{S}_l) = \sum_{j=0}^{l}\mathbf{S}_j\prod_{i=0}^{j-1}(\mathbf{I} - \mathbf{S}_i).$$

Since $\mathbf{H}_l$ corresponds to the hat matrix after the lth iteration, $tr(\mathbf{H}_l)$ may be considered as degrees of freedom of the estimate. The suggested stopping rule for boosting iterations is based on the corrected AIC criterion proposed by [10], given by

$$\mathrm{AIC}_c(l) = \log(\hat{\sigma}^2) + \frac{1 + tr(\mathbf{H}_l)/n}{1 - (tr(\mathbf{H}_l) + 2)/n},$$

where $\hat{\sigma}^2 = \frac{1}{n}(\mathbf{y} - \hat{\boldsymbol{\mu}}^{(l)})'(\mathbf{y} - \hat{\boldsymbol{\mu}}^{(l)})$. Thus, the optimal number of boosting iterations, which in our framework plays the role of a smoothing parameter, is determined by $l_{opt} = \arg\min_l \mathrm{AIC}_c(l)$.

An alternative stopping criterion may be BIC (see [12]), given by

$$BIC(l) = \log(\hat{\sigma}^2) + \log(n)\frac{tr(\mathbf{H}_l)}{n}.$$

Since the complexity of the fit is supposed to be penalized stronger by BIC, we expect an earlier stopping of the algorithm, compared to AIC_c.

3 Simulation study

In order to assess the performance of the CurveBoost algorithm, we conduct a simulation study in the style of [3]. For a nonparametric regression problem as given in (1), three types of concave functions are considered:

- $f_1(x) = 3\exp(-x^2/5)$,
- $f_2(x) = -x^2/2 + 3$,

- $f_3(x) = \begin{cases} x + 3.5, & \text{if} \quad x < -0.5 \\ 3, & \text{if} \quad -0.5 \leq x \leq 0.5 \\ -x + 3.5, & \text{if} \quad x > 0.5, \end{cases}$

all on the domain [-1.5,1.5]. The design points x_i are drawn from a $U[-1.5, 1.5]$-distribution, and we investigate sample sizes of $n = 60, 100, 200$. The errors are i.i.d. drawn from a $\mathcal{N}(0, \sigma^2)$-distribution with several levels of noise given by $\sigma = 0.25, 0.5, 0.75, 1$.

For the CurveBoost fit, a truncated power series bases of degree $q = 2$ is used, with $m = 40$ knots placed at the $j/(m+1)$th sample quantiles ($j = 1, \ldots, m$) of the x_i. For convenience, the predictor variable is always rescaled to $[0, 1]$. A ridge parameter of $\lambda = 50$ is chosen. To save computing time, boosting is stopped after a maximum number of $L = 1500$ iterations throughout the simulations.

For each setting, the proposed method is compared to an unconstrained smoothing spline of degree three (SS), where the smoothing parameter is chosen by GCV. Furthermore, we apply two earlier approaches to smoothing with curvature constraints based on regressions splines. The first is the so-called COBS procedure by [8], which belongs to the quantile regression framework. It uses the L_1-loss function and a quadratic B-spline basis. The curvature constrained estimate is given as a solution of a linear programming problem. The method is implemented in the R library `cobs`. For the present simulations, we use the pure regression spline solution with no additional penalization. We start with 40 knots placed at the sample quantiles and perform stepwise knot deletion based on the AIC criterion.

Another method for curvature constrained estimates is proposed by Dierckx [6]. It is based on cubic B-splines and can be expressed as a quadratic programming problem (for details, see [6, p.120 et sqq.]). In the current implementation, we use an initial number of 20 interior knots placed at the sample quantiles and again do stepwise knot deletion based on AIC. The quadratic programming problem is solved by the function `solveQP()`, implemented in the R library `quadprog` written by B. A. Turlach. The method is referred to as QProg.

Figure 1 shows typical data sets for the three types of functions for $n = 60$ and $\sigma = 0.5$, along with the fits of the considered smoothing methods. It is seen that unconstrained smoothing might yield wiggly curves, while the restricted approaches provide proper fits in such cases. For a systematic investigation of the performance of the competitors, we use the average squared error given by

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

as a measure of comparison. In Table 1, the results for $f_1(.)$ are given, which is a section of a radial function, i.e. the degree of curvature varies with x. Therefore, $S = 200$ data sets were drawn and the mean of the ASE is reported. It is seen that AIC_c-stopped CurveBoost improves the estimates compared to the constrained

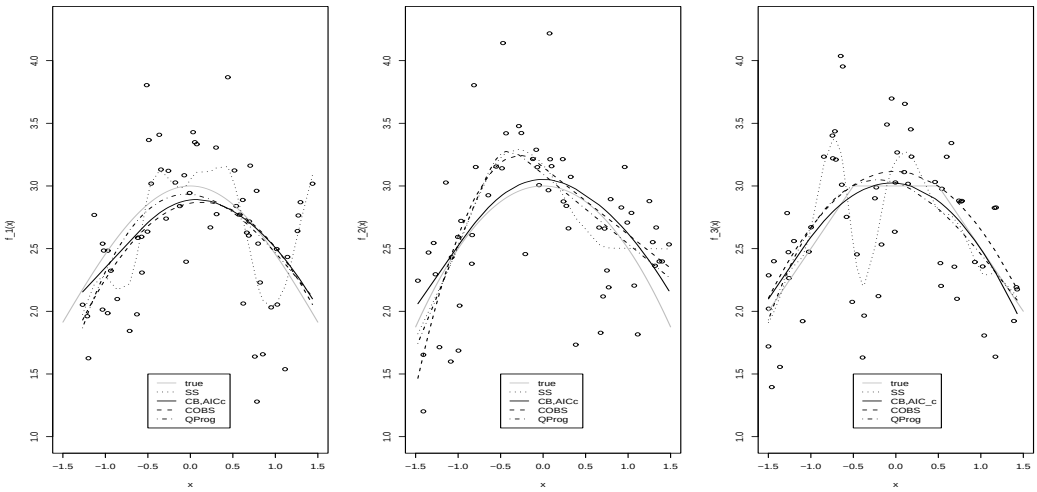

FIGURE 1: *Typical data sets for $n = 60$, $\sigma = 0.5$, and several estimates for $f_1(.)$ (left), $f_2(.)$ (mid), and $f_3(.)$ (right).*

and unconstrained competitors throughout all considered settings. Interestingly BIC-stopped CurveBoost does considerably worse when the noise is high. This might be explained by our observation that in some cases, boosting is stopped too early due to the stronger penalized complexity of the fit. QProg outperforms COBS especially in the high noise case.

Table 2 shows the results for $f_2(.)$, which is a polynomial of degree two, implying a constant degree of curvature. It is seen that the competitors behave quite similar as in the case of $f_1(.)$.

Finally, in Figure 2 boxplots of the simulation results for the piecewise linear function $f_3(.)$ are given. Note that this concave function does not fulfill the smoothness assumptions. In each panel, boxplots for $n = 60$ and 100 are drawn for a certain noise level. Also in this case, AIC_c-stopped CurveBoost yields the best performance of all considered smoothing methods, whereas COBS and QProg–if at all–outperform the unconstrained splines only in the higher noise cases.

4 Application

In order to illustrate the proposed approach, we consider a real world data set previously used by [14]. The data are taken from a 1971 Canadian Census Public Use Tape, where the age and income of $n = 205$ Canadian workers were recorded.

		SS	CB (AIC_c)	CB (BIC)	COBS	QProg
$\sigma = 0.25$	$n = 60$	0.0073	**0.0035**	**0.0035**	0.0061	0.0055
	$n = 100$	0.0033	**0.0022**	**0.0023**	0.0037	0.0035
	$n = 200$	0.0019	**0.0013**	**0.0013**	0.0022	0.0018
$\sigma = 0.5$	$n = 60$	0.0283	**0.0131**	**0.0145**	0.0222	0.0186
	$n = 100$	0.0125	**0.0080**	**0.0083**	0.0129	0.0117
	$n = 200$	0.0072	**0.0042**	**0.0044**	0.0072	0.0061
$\sigma = 0.75$	$n = 60$	0.0632	**0.0337**	0.0453	0.0476	**0.0392**
	$n = 100$	0.0278	**0.0184**	**0.0225**	0.0278	0.0237
	$n = 200$	0.0155	**0.0091**	**0.0099**	0.0149	0.0123
$\sigma = 1$	$n = 60$	0.1107	**0.0635**	0.0822	0.0831	**0.0677**
	$n = 100$	0.0494	**0.0348**	0.0529	0.0476	**0.0407**
	$n = 200$	0.0272	**0.0162**	**0.0205**	0.0253	0.0207

TABLE 1: *Function 1, mean averaged squared error over 200 simulated datasets for several fitting methods. The two best performers are given in bold faces.*

		SS	CB (AIC_c)	CB (BIC)	COBS	QProg
$\sigma = 0.25$	$n = 60$	0.0075	**0.0035**	**0.0035**	0.0057	0.0055
	$n = 100$	0.0036	**0.0023**	**0.0023**	0.0033	0.0034
	$n = 200$	0.0020	**0.0013**	**0.0014**	0.0017	0.0017
$\sigma = 0.5$	$n = 60$	0.0286	**0.0131**	**0.0144**	0.0219	0.0189
	$n = 100$	0.0129	**0.0080**	**0.0083**	0.0124	0.0116
	$n = 200$	0.0074	**0.0043**	**0.0044**	0.0067	0.0060
$\sigma = 0.75$	$n = 60$	0.0638	**0.0333**	0.0451	0.0476	**0.0402**
	$n = 100$	0.0283	**0.0184**	**0.0225**	0.0271	0.0238
	$n = 200$	0.0160	**0.0092**	**0.0099**	0.0145	0.0124
$\sigma = 1$	$n = 60$	0.1112	**0.0631**	0.0833	0.0830	**0.0686**
	$n = 100$	0.0501	**0.0349**	0.0530	0.0463	**0.0406**
	$n = 200$	0.0278	**0.0163**	**0.0209**	0.0250	0.0210

TABLE 2: *Function 2, mean averaged squared error over 200 simulated datasets for several fitting methods. The two best performers are given in bold faces.*

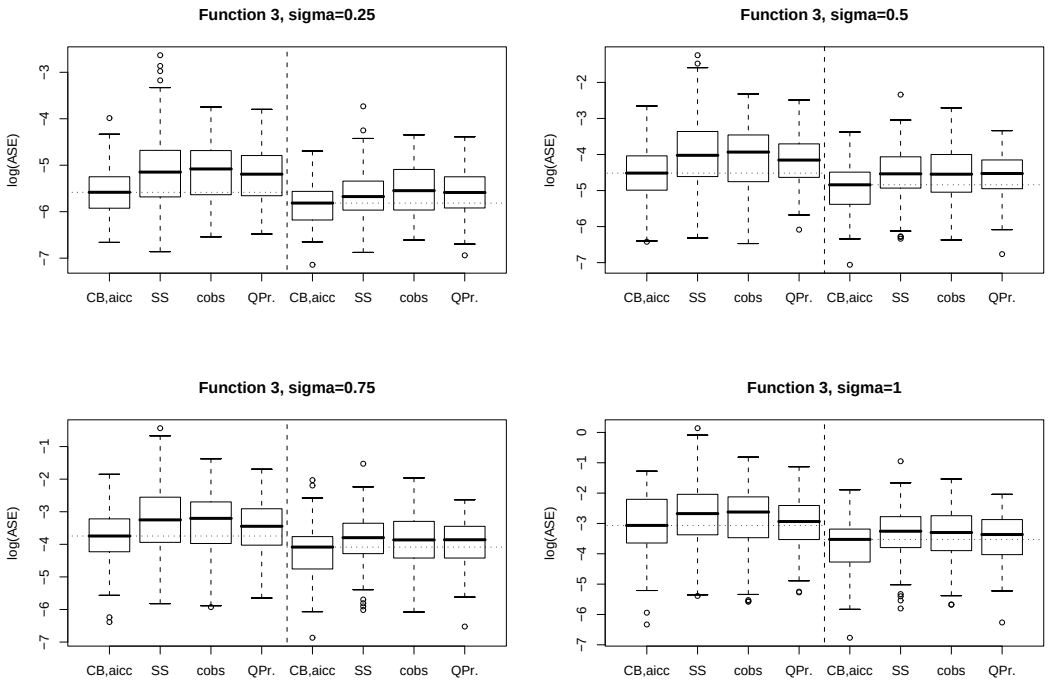

FIGURE 2: *Boxplots of* log *ASE for different fitting methods for* $f_3(.)$ *with different noise levels. In each panel, sample sizes of* $n = 60$ *(left) and* $n = 100$ *(right) are given.*

As mentioned earlier, economic theory assumes a concave relationship between working experience and the logarithm of the income (see e.g. [5]).

In Figure 3, the unconstrained smoothing spline fit is given, along with an AIC_c-stopped CurveBoost and the restricted fits by COBS and QProg. The same settings of knots and parameter selection as in the simulations are used (boosting stops after $l_{opt}^{AIC_c} = 15584$ iterations). It is seen that the concavity assumption is violated by the unconstrained fit at an age about 40 to 50. QProg is influenced by some observations with low income values at the left boundary. We observed similar behavior also in the simulations. Since COBS uses a L_1-loss function, it yields a rather robust fit. CurveBoost seems to provide a sensible compromise between robustness and accuracy.

5 Conclusion

A novel approach to curvature constrained fitting based on regression splines has been proposed. In contrast to former approaches which are based on linear or quadratic programming methodology, estimation is done by a boosting algorithm which controls the curvature constraint in a quite easy way in each step. Sim-

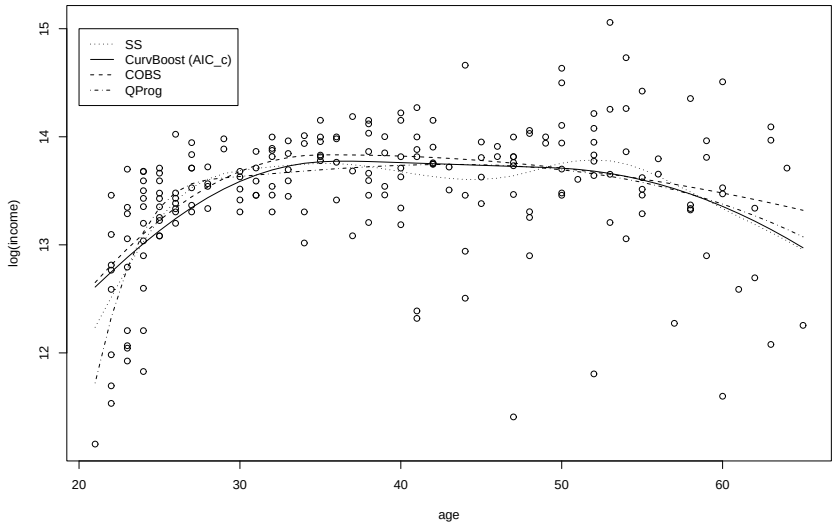

FIGURE 3: *Age and income data, along with an unconstrained spline fit (dotted), and concave fits by CurveBoost (AIC_c-stopped, solid), COBS (dashed) and Dierckx (dash-dotted).*

ulations suggest that the proposed procedure is very competitive and is able to outperform more traditional approaches in a variety of settings.

The approach may be extended to an additive setting with $p > 1$ covariates by using p sets of basis functions and by including all of the basis functions in the fitting and selection step of the algorithm. Furthermore, a similar algorithm can be derived for curvature *and* monotonicity restricted smoothers by modifying the basis functions and constraints slightly.

Acknowledgements

This research was supported by the Deutsche Forschungsgemeinschaft (SFB 386, "Statistical Analysis of Discrete Structures").

References

[1] Bühlmann, P. (2006). *Boosting for high–dimensional linear models.* The Annals of Statistics, to appear.

[2] Bühlmann, P., Yu, B. (2003). *Boosting with the L_2-loss: regression and classification.* Journal of the American Statistical Association **98**, 324-339.

[3] Delecroix, M., Simioni, M., Thomas-Agnan, C. (1995). *A Shape Constrained Smoother: Simulation Study.* Computational Statistics **10**, 155–175.

[4] Delecroix, M., Simioni, M., Thomas-Agnan, C. (1996). *Functional Estimation under Shape Constraints.* Journal of Nonparametric Statistics **6**, 69–89.

[5] Delecroix, M., Thomas-Agnan, C. (2000). *Spline and Kernel Regression under Shape Restrictions.* In: Schimek, M. G. (ed), Smoothing and Regression. Wiley, New York. 69–89.

[6] Dierckx, P. (1993). *Curve and Surface Fitting with Splines.* Oxford Science Publications, Oxford.

[7] Friedman, J. H. (2001). *Greedy function approximation: a gradient boosting machine.* The Annals of Statistics **29**, 1189–1232.

[8] He, X., Ng, P. (1999). *COBS: Qualitatively Constrained Smoothing via Linear Programming.* Computational Statistics **14**, 315–337.

[9] Hoerl, A. E., Kennard, R. W. (1970). *Ridge Regression: Bias Estimation for Nonorthogonal Problems.* Technometrics **12**, 55–67.

[10] Hurvich, C. M., Simonoff, J. S., Tsai, C. (1998). *Smoothing Parameter Selection in Nonparametric Regression Using an Improved Akaike Information Criterion.* Journal of the Royal Statistical Society **B 60**, 271–293.

[11] Schapire, R. E. (1990). *The Strength of Weak Learnability.* Machine Learning **5**, 197–227.

[12] Schwarz, G. (1978). *Estimating the Dimension of a Model.* Annals of Statistics **6**, 461–464.

[13] Turlach, B. A. (2005). *Shape constrained smoothing using smoothing splines.* Computational Statistics **20**, 81–103.

[14] Ullah, A. (1985). *Specification Analysis of Econometric Models.* Journal of Quantitative Economics **2**, 197–209.