BIAS AND VARIANCE REDUCTION PROCEDURES IN NON-PARAMETRIC REGRESSION

Marike Cockeran
North-West University, Potchefstroom, South Africa
Cornelia J. Swanepoel [1]
North-West University, Potchefstroom, South Africa
e-mail: *cornelia.swanepoel@nwu.ac.za*

Key words: Bagging, Bandwidth, Boosting, Bragging, Cross-validation, Kernel estimators, Non-parametric, Regression.

Abstract: The purpose of this study is to determine the effect of three improvement methods on nonparametric kernel regression estimators. The improvement methods are applied to the Nadaraya-Watson estimator with cross-validation bandwidth selection, the Nadaraya-Watson estimator with plug-in bandwidth selection, the local linear estimator with plug-in bandwidth selection and a bias corrected nonparametric estimator proposed by Yao (2012), based on cross-validation bandwith selection. The performance of the different resulting estimators are evaluated by empirically calculating their mean integrated squared error (MISE), a global discrepancy measure. The first two improvement methods proposed in this study are based on bootstrap bagging and bootstrap bragging procedures, which were originally introduced and studied by Swanepoel (1988, 1990), and hereafter applied, e.g., by Breiman (1996) in machine learning. Bagging and bragging are primarily variance reduction tools. The third improvement method, referred to as boosting, aims to reduce the bias of an estimator and is based on a procedure originally proposed by Tukey (1977). The behaviour of the classical Nadaraya-Watson estimator with plug-in estimator turns out to be a new recommendable nonparametric regression estimator, since it is not only as precise and accurate as any of the other estimators, but it is also computationally much faster than any other nonparametric regression estimator considered in this study.

1. Introduction

In regression analysis, the term *non-parametric* refers to a flexible unknown functional form of the regression curve. A great deal of effort and attention from researchers went into the development of elegant non-parametric regression methods. Especially kernel methods are popular, although they present only a fraction of many approaches towards the construction of flexible models. In this paper, however, we shall restrict our attention to kernel estimation of joint densities and mean regression functions.

[1] Corresponding Author

AMS: 62G08, 62G05, 62H12

Two bootstrap-based variance reduction methods, viz., bagging and bragging, are applied to derive several bandwidth selection procedures. Specifically, three ways of determining bandwidths data-dependently by applying the bagging method, will be defined, and the effects of these bandwidth selectors on the behaviour of various regression estimators, with respect to the mean integrated squared error (MISE) and its components, will be evaluated and demonstrated. The process will then be repeated, using the bragging method. The influence of the proposed bandwidths on the behaviour of the following non-parametric estimators is of specific interest: the Nadaraya-Watson estimator based on improved squared-error cross-validation data-derived bandwidths as well as on new improved plug-in bandwidths, the local linear estimator using the improved rule-of-thumb plug-in bandwidths (Fan and Gijbels, 1996), and a new bias reduction non-parametric (BRNP) kernel regression estimator suggested by Yao (2012), employing the bias-correction idea of Chung and Lindsay (2011), which was developed for density estimation. For this estimator three improved cross-validation bandwidths are applicable since Yao (2012) preferred cross-validation bandwidth over the plug-in bandwidth. The improved kernel regression estimation methods are evaluated comparatively, by applying them to well known regression models that appeared in the literature.

Furthermore, a bias reduction improvement method, viz., the boosting method, is utilized on the classical kernel regression estimators and are evaluated empirically. Boshoff (2009) did a limited simulation study, confirming some influence of boosting, bagging, bragging and combinations thereof, on the Nadaraya-Watson estimator with cross-validation bandwidth selection. However, the effect of boosting, bagging and bragging on the Nadaraya-Watson estimator with plug-in bandwidth selection, the local linear estimator with plug-in bandwidth selection and Yao's estimator with cross-validation bandwidth selection seem to be new contributions to the literature. Computer time for the various methods will be reported on in general. It should be noted that the behaviour of the bias and variance components of the estimators are of interest throughout the study.

The BRNP-estimator turned out to be highly computer-intensive and only a limited study was performed, so that no comparisons were possible with others estimators. Only the behaviour of the BRNP-estimator based on cross-validation bandwidths under various scenarios were reported. However, comparative studies were possible between the bahaviour of the Nadaraya-Watson-estimator and the local linear estimator, both based on plug-in methods, as well as between the Nadaraya-Watson estimator, based on plug-in methods and the Nadaraya-Watson estimator, based on cross-validation methods, for all the various bagging and bragging methods and the boosting method.

The paper is organized as follows. In Sections 2 and 3 basic notation is stated regarding the proposed estimators and the appropriate bandwidth selectors for each estimator. Three procedures to improve the bandwidth selectors and regression estimators discussed in Sections 2 and 3, will be presented in Section 4, together with three algorithms to assist the practitioner. The simulation setup is set out in Section 5, which is summarized in a main algorithm and presented in Section 6. Results and conclusions are discussed in Section 7, and specific comparisons are made between the estimators in their classical and improved forms, as well as between the improvement methods, in Sections 8 to 11 . Section 12 presents a brief discussion on the computer-time involved for each procedure, while in Section 13 graphical illustrations are discussed. Finally, Section 14 captures final remarks and recommendations. A set of references follows as well as graphical illustrations of the various estimators.

2. Basic notation and definitions

In the case of bivariate observations, we wish to explore the association between the covariate X and the response Y. Let $\mathbf{S}_n = \{(X_1,Y_1),(X_2,Y_2),\ldots,(X_n,Y_n)\}$ be a given i.i.d. sample from the population (X,Y). Consider the mean regression model

$$Y_i = m(X_i) + \varepsilon_i, \;\; i = 1,\ldots,n,$$

where we assume that $m(x) = E(Y|X = x)$ is unknown and $E(\varepsilon_i|X_i) = 0$.

2.1. Nadarya-Watson estimator

Nadaraya (1964) and Watson (1964) proposed the following regression estimator

$$\hat{m}_{NW}(x) = \frac{\sum_{i=1}^{n} K_h(x-X_i)Y_i}{\sum_{i=1}^{n} K_h(x-X_i)}, \tag{1}$$

where

$$K_h(u) = h^{-1}K(u/h), \tag{2}$$

with $K(\cdot)$ a bounded (kernel) density function and h a smoothing parameter (bandwidth). Härdle (1990, p.77) defined the pointwise asymptotic bias and variance for the Nadaraya-Watson estimator, which is given by

$$\text{Bias}[m\hat{_{NW}}(x)] = \left(m''(x) + \frac{2m'(x)f'(x)}{f(x)}\right)\frac{h^2}{2}\int u^2K(u)\,du,$$

and

$$\text{Var}[m\hat{_{NW}}(x)] = \frac{\sigma^2(x)}{f(x)nh}\int K^2(u)\,du,$$

where $f(x)$ denotes the marginal density of X and $\sigma^2(x) = \text{Var}(Y|X = x)$.

2.2. The local linear estimator

The local linear regression smoother is defined by

$$\hat{m}_{LL}(x) = \frac{\sum_{i=1}^{n} w_iY_i}{\sum_{i=1}^{n} w_i}, \tag{3}$$

with

$$w_i = K\left(\frac{x-X_i}{h}\right)[s_{n,2} - (x-X_i)s_{n,1}], \tag{4}$$

where

$$s_{n,l} = \sum_{i=1}^{n} K\left(\frac{x-X_i}{h}\right)(x-X_i)^l, \;\; l = 1,2. \tag{5}$$

Furthermore, from Fan and Gijbels (1996, p.17) it follows that the pointwise asymptotic bias and variance of the local linear smoother is given by

$$\text{Bias}[\hat{m_{LL}}(x)] = m''(x)\frac{h^2}{2}\int u^2 K(u)\,du,$$

and

$$\text{Var}[\hat{m_{LL}}(x)] = \frac{\sigma^2(x)}{f(x)nh}\int K^2(u)\,du.$$

2.3. The bias reduction non-parametric regression (BRNP) estimator

The new non-parametric BRNP regression estimator proposed by Yao (2012) deserves attention because it has asymptotic bias of order h^4 in contrast to the local linear estimators' asymptotic bias of order h^2. The estimator of Yao (2012) is based on the following bias-corrected estimator $\hat{f}(x)$ of the design density $f(x)$, developed by Chung and Lindsay (2011) which has asymptotic bias of order h^4:

$$\hat{f}(x) = n^{-2}c(h)\sum_i\sum_j w_j K_{\sqrt{3}h}(X_i - X_j)K_{\sqrt{3}h}(X_i - x)K_{\sqrt{3}h}(X_j - x),$$

where

$$c(h) = 3\sqrt{2\pi}h,$$

and

$$w_j = \left\{\frac{1}{n}\sum_{i=1}^{n} K_{\sqrt{2}h}(X_i - X_j)\right\}^{-1}.$$

Yao (2012) proposed the following bias-corrected non-parametric regression estimator:

$$\hat{m}_{BRNP}(x) = \frac{n^{-2}c(h)\sum_i\sum_j v_j Y_i Y_j K_{\sqrt{3}h}(X_i - X_j)K_{\sqrt{3}h}(X_i - x)K_{\sqrt{3}h}(X_j - x)}{\hat{f}(x)}, \tag{6}$$

where

$$v_j = \left\{\frac{1}{n}\sum_{i=1}^{n} K_{\sqrt{2}h}(X_i - X_j)Y_i\right\}^{-1}. \tag{7}$$

Clearly, $\hat{m}_{BRNP}(x)$ is not a linear smoother, since it is not linear in the response.
Furthermore, Yao (2012) derived the following expressions for the pointwise asymptotic bias and variance of the estimator:

$$\text{Bias}[\hat{m_{BRNP}}(x)] = \frac{h^4\{A - m(x)B\}}{f(x)}, \tag{8}$$

and

$$\mathrm{Var}[m_{B\hat{R}NP}(x)] = \frac{\sigma^2(x)}{nh\sqrt{\pi}f(x)}\left(\sqrt{2}+\frac{1}{4}-\frac{2}{\sqrt{3}}\right), \tag{9}$$

where

$$A = \left[-g^{(4)}(x)+g^{-1}(x)g'(x)g'''(x)+g^{-1}(x)g''(x)^2-g'(x)^2g^{-2}(x)g''(x)\right], \tag{10}$$

and

$$B = \left[-f^{(4)}(x)+f^{-1}(x)f'(x)f'''(x)+f^{-1}(x)f''(x)^2-f'(x)^2f^{-2}(x)f''(x)\right], \tag{11}$$

with $g(x) = m(x)f(x)$. From (8) to (11) it is clear that the asymptotic bias depends on the first four derivatives of both $m(x)$ and $f(x)$.

3. Bandwidth selectors

Both the leave-one-out squared-error cross-validation and the plug-in methods for deriving appropriate bandwidth selectors are now discussed.

3.1. Bandwidth selectors for the Nadaraya-Watson estimator

A *plug-in bandwidth selector* can be derived from explicit expressions for the asymptotic variance and asymptotic squared bias of the Nadaraya-Watson estimator (see e.g. Härdle (1990), by calculating the asymptotic optimal bandwidth for the conditional mean integrated squared-error (MISE), i.e.,

$$h_{opt} = \left[\frac{d_K}{4c_K^2}\right]^{1/5}\left[\frac{\int\sigma^2(x)w_0(x)\,dx}{\int\left\{\frac{1}{2}m''(x)+m'(x)\frac{f'(x)}{f(x)}\right\}^2 w_0(x)f(x)\,dx}\right]^{1/5} n^{-1/5},$$

where $c_K = \int u^2K(u)\,du$ and $d_K = \int K^2(u)\,du$. The unknown quantities $\sigma^2(\cdot)$, $m'(\cdot)$, $m''(\cdot)$, $f(\cdot)$ and $f'(\cdot)$ need to be estimated. Assume that the conditional variance $\sigma^2(x)$ is constant, substitute the pilot estimates $\hat{m}'(\cdot)$, $\hat{m}''(\cdot)$, $\hat{f}'(\cdot)$, $\hat{f}''(\cdot)$ and $\hat{\sigma}^2$, and estimate the denominator by

$$\frac{1}{n}\sum_{i=1}^{n}\left\{\frac{1}{2}\hat{m}''(X_i)+\hat{m}'(X_i)\frac{\hat{f}'(X_i)}{\hat{f}(X_i)}\right\}^2 w_0(X_i).$$

A plug-in bandwidth selector is therefore derived, i.e.,

$$\hat{h}_{plug} = \left[\frac{d_K}{4c_K^2}\right]^{1/5}\left[\frac{\hat{\sigma}^2\int w_0(x)\,dx}{\sum_{i=1}^{n}\left\{\frac{1}{2}\hat{m}''(X_i)+\hat{m}'(X_i)\frac{\hat{f}'(X_i)}{\hat{f}(X_i)}\right\}^2 w_0(X_i)}\right]^{1/5},$$

for a given weight function $w_0(\cdot)$.

The pilot estimate $\hat{m}(x)$ is obtained by fitting a polynomial of degree k globally to the data (X_i, Y_i), $i = 1, \ldots, n$, leading to the parametric fit

$$\hat{m}(x) = \hat{\beta}_0 + \hat{\beta}_1 x + \cdots + \hat{\beta}_k x^k, \tag{12}$$

where k is an appropriate integer. The derivatives $\hat{m}'$ and $\hat{m}''$ are calculated from (12). As usual, $\hat{\sigma}^2$ is defined as the standardized residual sum of squares obtained from the fit. The pilot density estimate $\hat{f}(x)$ is the classical kernel density estimate given by

$$\hat{f}(x) = \frac{1}{n} \sum_{i=1}^{n} K_{\hat{h}}(x - X_i),$$

with $\hat{h} = 1.059 s_n n^{-\frac{1}{5}}$, and s_n being the standard deviation of $X_1, X_2, \ldots, X_n$.

The *leave-one-out squared-error cross-validation method* employs the regression smoother after having deleted the j^{th} observation, to estimate m in the point X_j. The Nadaraya-Watson estimator, when the j^{th} observation has been left out, is given by

$$\hat{m}_{NW,(j)}(X_j) = \frac{\sum_{i=1, i \neq j}^{n} K_h(X_j - X_i) Y_i}{\sum_{i=1, i \neq j}^{n} K_h(X_j - X_i)}, \tag{13}$$

where $K_h(u)$ is defined in (2). The cross-validation function is then calculated as the average of the squared deleted residuals

$$CV(h) = \frac{1}{n} \sum_{j=1}^{n} [Y_j - \hat{m}_{NW,(j)}(X_j)]^2.$$

The cross-validation method chooses the bandwidth that minimizes $CV(h)$. Denote this bandwidth by $\hat{h}_{CV}$. Then the Nadaraya-Watson estimator is determined by using $\hat{h}_{CV}$, i.e.,

$$\hat{m}_{NW}(x) = \frac{\sum_{i=1}^{n} K_{\hat{h}_{CV}}(x - X_i) Y_i}{\sum_{i=1}^{n} K_{\hat{h}_{CV}}(x - X_i)}.$$

3.2. Bandwidth selector for the local linear estimator

From Fan (1992) and Fan and Gijbels (1996, pp.57–58) the plug-in bandwidth selector for the local linear estimator is defined by

$$\hat{h}_{plug} = \left(\frac{d_K \hat{\sigma}^2 \int w_0(x)\,dx}{c_K^2 \sum_{i=1}^{n} \{\hat{m}''(X_i)\}^2 w_0(X_i)} \right)^{1/5},$$

where $\hat{\sigma}^2$ and $\hat{m}''(\cdot)$ are pilot estimates of σ^2 and $m''(\cdot)$, obtained by the same process as above via the fit of a suitable polynomial.

3.3. Bandwidth selector for the BRNP estimator

For the BRNP estimator, the squared-error cross-validation bandwidth selection method is used. The leave-one-out BRNP estimator for a given bandwidth h, is defined for $k = 1,2,\dots,n$ by

$$\hat{m}_{\text{BRNP},(k)}(X_k) = \frac{\sum_{\substack{i=1\\ i\neq k}}^{n}\sum_{\substack{j=1\\ j\neq k}}^{n} v_j Y_i Y_j K_{\sqrt{3}h}(X_i - X_j) K_{\sqrt{3}h}(X_i - X_k) K_{\sqrt{3}h}(X_j - X_k)}{\sum_{\substack{i=1\\ i\neq k}}^{n}\sum_{\substack{j=1\\ j\neq k}}^{n} w_j \sqrt{3}h(X_i - X_j) K_{\sqrt{3}h}(X_i - X_k) K_{\sqrt{3}h}(X_j - X_k)}.$$

The cross-validation function is defined in (13), with NW replaced by $BRNP$ and again the bandwidth that minimizes $CV(h)$ is selected.

4. Improvement methods

Three methods for improving the performance of a given estimation procedure are explored and need to be defined, i.e., boosting, bagging and bragging. Marzio and Taylor (2008) showed that boosting improves the Nadaraya-Watson estimator, as far as bias reduction is concerned. On the other hand, the influence and effect of bootstrap-based improvement methods, viz., bagging and bragging, are aimed on variance reduction.

4.1. Boosting

Boosting is a general method for improving the accuracy of any given 'learning algorithm'. In this study the learning algorithm is the calculation of a non-parametric regression estimator. Boosting methods, known as L_2-boosting (Bühlmann, 2003), were developed for non-parametric regression which involves minimizing the squared error loss function. This boils down to iterative refitting of the residuals. The L_2-boosting algorithm involves the following basic steps: Suppose the data $S_n = \{(X_1,Y_1),(X_2,Y_2),\dots,(X_n,Y_n)\}$ is given. Then fit an initial regression procedure to obtain $\hat{m}_0(\cdot)$, using some bandwidth h, obtained from some bandwidth selection method. Or use the default value $\hat{m}_0(\cdot) \equiv \bar{Y} = \frac{1}{n}\sum_{i=1}^{n} Y_i$. Set $t_0 = 0$. Let $t_i = t_{i-1} + 1$ for $i = 1,\dots,n$. Compute for $i = 1$ the residuals,

$$U_j = Y_j - \hat{m}_{t_{i-1}}(X_j), \quad j = 1,\dots,n.$$

Then fit the residuals $U_1,U_2,\dots,U_n$ to $X_1,X_2,\dots,X_n$ by the base procedure. Call this fit $\tilde{m}_{t_i}(\cdot)$. Update the estimator by

$$\hat{m}_{t_i}(\cdot) = \hat{m}_{t_{i-1}}(\cdot) + \nu\tilde{m}_{t_i}(\cdot),$$

where $0 < \nu \leq 1$ is a real-valued step-length factor. Marzio and Taylor (2008) chose $\nu = 1$.

Repeat the procedure for $i = 2,\dots,n$, where $t_n = T$ the stopping time for the number of iterations. Marzio and Taylor (2008) identified a set of kernels satisfying specific matrix requirements, excluding many popular kernel functions, such as the Epanechnikov, biweight and triweight kernels. However, Gaussian and triangular kernels are permitted. Furthermore, when boosted, the bias decreases exponentially fast towards zero, while the variance increases exponentially slow towards σ^2 for the Nadaraya-Watson estimator. Also, the number of boosting iterations should be emphasized. Boshoff (2009) pointed out that very little improvement was gained from $T = 1$ to $T = 6$. After one boosting iteration, the bias is reduced from $O(h^2)$ to $o(h^2)$. If more than one iteration is performed, Marzio and Taylor (2008) pointed out that, rather than choosing the optimal number of boosting iterations T and the bandwidth h separately, the optimal pair (h,T) should be chosen.

4.2. Bagging and bragging

The basic bagging and bragging methodology was presented by Swanepoel (1988, 1990) from a functional approach, and by Breiman (1996) from an ensemble viewpoint. Swanepoel (1988, 1990) proved satisfactory asymptotic properties of these procedures. Furthermore, Hall and Robinson (2009) showed that bagging can be used to reduce the variability of bandwidth selectors obtained by cross-validation in kernel regression estimation. They suggested that bagging can be applied to the cross-validation bandwidth selection method in at least two ways, namely bagging the cross-validation function $CV(h)$ and bagging $\hat{h}_{CV}$. In this study, bagging is applied to the cross-validation bandwidth selection method and plug-in bandwidth selection methods in three ways.

Swanepoel's (1988, 1990) approach was developed as an effort to construct estimators for a parameter θ, which is written in the form $\theta = \psi(F)$, for some suitable functional ψ, where ψ depends on the distribution function F of the population. From this functional approach it is shown that if $\psi(F)$ can be approximated by a sequence of functionals, namely $\psi_m(F) \approx \psi(F)$, with the approximation becoming increasingly accurate as $m \to \infty$, then $\psi_m(F_n)$ can be taken as an estimator of θ, with $m = m(n)$ suitably chosen. Here F_n refers to the empirical distribution function.

Suppose $T_m(X_1, X_2, \ldots, X_m)$ is some known estimator of θ, such as the sample mean. Then two possible choices of $\psi_m(F)$ are

$$\psi_{m,1}(F) = E[T_m(X_1, X_2, \ldots, X_m)]$$

and

$$\psi_{m,2}(F) = \text{median}[T_m(X_1, X_2, \ldots, X_m)].$$

In this case we have that

$$\psi_{m,1}(F_n) = E_*[T_m(X_1^*, X_2^*, \ldots, X_m^*)]$$

and

$$\psi_{m,2}(F_n) = \text{median}_*[T_m(X_1^*, X_2^*, \ldots, X_m^*)], \tag{14}$$

respectively, where $(X_1^*, X_2^*, \ldots, X_m^*)$ denotes a bootstrap random sample of size m taken from F_n. The ideal bootstrap estimates $\psi_{m,1}(F_n)$ and $\psi_{m,2}(F_n)$ are approximated by Monte Carlo algorithms which will be stated in Section 4.3. The choices $\psi_{m,1}(F_n)$ and $\psi_{m,2}(F_n)$ are nowadays known as *bagging* and *bragging* respectively in the statistical literature.

Bagging and bragging procedures also developed from the theory of ensemble methods as mentioned before. *Bagging* is an acronym for **b**ootstrap **agg**regat**ing** and this term was introduced by Breiman (1996).

The number of bootstrap replications B in practice governs the accuracy of the Monte Carlo approximation and depends on the sample size n. It is expected that bagging will increase bias, but the hope is set on reducing the variance and on a decrease in terms of the MISE. Also, together with the *boosting* algorithm, we may expect a decrease in MISE to some extent.

Bragging stands for **b**ootstrap **r**obust **agg**regat**ing**. The *sample median* is used over the B bootstrap estimates instead of the sample mean in the discussion above, as is reflected in (14).

4.3. Bagging and bragging bandwidths

One suggestion by Hall and Robinson (2009) for bagged cross-validation is to bag h_{CV} when determining the smoothing parameter. From a limited simulation study, Boshoff (2009) involved only squared-error cross-validation and the Nadaraya-Watson estimator and found that at least three different cross-validation bagging methods for determining bandwidths were worthwhile investigating, in order to perform bootstrap aggregation. Throughout, these methods are referred to as $Bag1, Bag2$ and $Bag3$. However, the same procedures can be applied for plug-in methods. If cross-validation and plug-in bandwidth selection are therefore used in the algorithms below, these three methods are referred to as Bag1_{CV}, Bag2_{CV}, Bag3_{CV} and Bag1_{plug}, Bag2_{plug} and Bag3_{plug} respectively.

Similarly, if these 6 bandwidth estimation methods are repeated, but in all algorithms medians are determined instead of averages, i.e.,

$$\hat{h}_{brag_m} = \text{median}_{1\leq b\leq B}\hat{h}^*_{CV}(b), \qquad m = 1,2,3,$$

is replaced for

$$\hat{h}_{bag_m} = \frac{1}{B}\sum_{b=1}^{B}\hat{h}^*_{CV}(b), \qquad m = 1,2,3,$$

for both cross-validation and plug-in bandwidth selection methods, we also obtain methods Brag1_{CV}, Brag2_{CV}, Brag3_{CV}, Brag1_{plug}, Brag2_{plug} and Brag3_{plug}.

In this way, apart from calculating the three classical regression methods, and the three boosted regression methods, 12 regression estimation methods, using improved bandwidths, are also included in the comparative studies below.

For example, the classical Nadaraya-Watson estimator is calculated, boosted and again calculated in 12 other ways, using Bag1_{CV}, Bag2_{CV}, Bag3_{CV}, Bag1_{plug}, Bag2_{plug}, Bag3_{plug} and Brag1_{CV}, Brag2_{CV}, Brag3_{CV}, Brag1_{plug}, Brag2_{plug}, Brag3_{plug}, resulting in 16 algorithms. The same holds for the local linear estimator (which only uses the plug-in bandwidths, resulting in 8 methods) and the *BRNP*-estimator (using only cross-validation methods, resulting also in only 8 methods).

Brief algorithms to determine regression estimators via the first three , i.e., by using Bag1_{CV}, Bag2_{CV}, Bag3_{CV}, for cross-validation selected bandwidths, will follow below. Algorithms for plug-in bagged bandwidths and subsequent regression estimates as well as the 6 bragged equivalents, are determined analogously and will be omitted.

Algorithm 4.1. Bag1$(_{\mathbf{CV}})$ using cross-validation bandwidth selection

1. Let $\mathbf{S}_n = \{(X_1,Y_1),(X_2,Y_2),\ldots,(X_n,Y_n)\}$ be the given sample.

2. Randomly sample m data pairs *without replacement* from $\mathbf{S}_n$ to obtain a bootstrap sample $\mathbf{S}^*_m = \{(X^*_1,Y^*_1),(X^*_2,Y^*_2),\ldots,(X^*_m,Y^*_m)\}$.

3. Determine, over an appropriate grid of h values, a bootstrap version of the leave-one-out cross-validation function by using the bootstrap sample $\mathbf{S}_m^*$ to obtain $CV^*(h)$. Obtain the value $\hat{h}_{CV}^*$ that minimizes $CV^*(h)$.

4. Repeat steps 2 and 3 B times to obtain bootstrap replications of the cross-validation bandwidth $\hat{h}_{CV}^*(1), \hat{h}_{CV}^*(2), \ldots, \hat{h}_{CV}^*(B)$.

5. Calculate the average of the bootstrap replications of the bandwidth
$$\hat{h}_{bag_m} = \frac{1}{B}\sum_{b=1}^{B} \hat{h}_{CV}^*(b).$$

6. Rescale the bandwidth in step (5): $\hat{h}_{bag} = (\hat{h}_{bag_m})\left(\frac{m}{n}\right)^{\frac{1}{5}}$ as was suggested by Hall and Robinson (2009).

7. Fit the estimator $\hat{m}(x)$ using the original sample $\mathbf{S}_n$ and the bandwidth $\hat{h}_{bag}$.

Here $\hat{m}(x)$ can refer to any of the regression estimators defined in (1), (3)–(5) or (6)–(7).

Algorithm 4.2. Bag2($_{\mathbf{CV}}$) using cross-validation bandwidth selection

1. Repeat steps 1 to 5 of Algorithm 4.1.

2. Use each bootstrap sample $\mathbf{S}_m^*$ and the aggregated bandwidth $\hat{h}_{bag_m}$ and fit the estimator $\hat{m}(x)$ to obtain bootstrap versions of the regression estimate
$$\hat{m}^*(x)^{(1)}, \hat{m}^*(x)^{(2)}, \ldots, \hat{m}^*(x)^{(B)}.$$

3. Calculate the average of the bootstrap replications of the regression estimate
$$\hat{m}^*(x) = \frac{1}{B}\sum_{b=1}^{B} \hat{m}^*(x)^{(b)}.$$

Algorithm 4.3. (Bag3$_{\mathbf{CV}}$) using cross-validation bandwidth selection

1. Repeat steps 1 to 3 of Algorithm 4.1.

2. Use the bootstrap sample $\mathbf{S}_m^*$ and corresponding bandwidth $\hat{h}_{CV}^*$ and fit the estimator $\hat{m}(x)$ to obtain a bootstrap version of the regression estimate $\hat{m}^*(x)$.

3. Repeat steps 1 and 2 of this algorithm B times to obtain bootstrap replications of the regression estimate
$$\hat{m}^*(x)^{(1)}, \hat{m}^*(x)^{(2)}, \ldots, \hat{m}^*(x)^{(B)}.$$

4. Calculate the average of the bootstrap replications of the regression estimate
$$\hat{m}^*(x) = \frac{1}{B}\sum_{b=1}^{B} \hat{m}^*(x)^{(b)}.$$

5. Simulation setup

A short main algorithm will be presented below, to summarize the complete simulation process from data generation to evaluation of each procedure, on the hand of the behaviour of the MISE and its components. Necessary information for the simulation study is as follows:

Three specific models, which are used frequently in the literature and which are well-known for being difficult to estimate, will be considered. These models are defined in Table 1 and represented graphically in Figure 1.

Table 1: The underlying regression function $m(x)$.

Model	$m(x)$
1	$x+2\exp(-16x^2)$
2	$\sin(2x)+2\exp(-16x^2)$
3	$0.3\exp\{-4(x+1)^2\}+0.7\exp\{-16(x-1)^2\}$

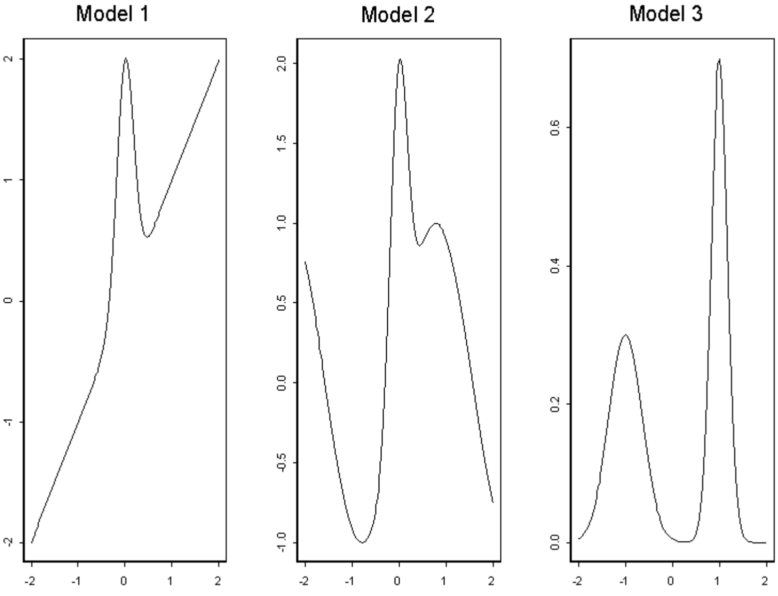


Figure 1: The underlying regression function $m(x)$.

The covariate data were drawn from the $U(-2,2)$ and $N(0,1)$ distributions respectively as was done in Fan and Gijbels (1996). Response data Y are constructed from $Y = m(X) + \sigma\varepsilon$, where the ε's are random errors drawn from the $N(0,1)$ distribution in this study, m is chosen from Table 1 and, for comparison reasons with related papers, σ is chosen as 0.2, 0.6 and 1.0 for Models 1 and 2 and as 0.1, 0.3 and 0.5 for Model 3. Similar to Marzio and Taylor (2008) we used samples of sizes $n = 50$, $n = 100$ and $n = 200$. The $N(0,1)$ and triangular kernels were used throughout. Initial cross-validation bandwidth application needs to be determined on a grid of possible bandwidth values. A grid between 0.01 and 2 is initially used and subdivided in increments of length 0.01. In a preliminary study it was confirmed that this specific choice of the grid is effective.

For the execution of plug-in bandwidth selection algorithms, the weight function is defined as $w_0(x) = I(-1.8 \leq x \leq 1.8)$ if $X \sim U(-2,2)$. This is the same weight function used by Fan and Gijbels (1996, p.112). If $X \sim N(0,1)$, the weight function is defined as $w_0(x) = I(-1.8 \leq x \leq 1.8)\phi(x)$, where $\phi(x)$ is the standard normal density function. For sample sizes $n = 100$ and $n = 200$ a polynomial regression function of degree 12 is used and for $n = 50$ a polynomial regression function of degree 5 is used in (12). Similar to Hall and Robinson (2009) we used 50 bootstrap samples for the application of the bagging and bragging methods throughout.

Bootstrap samples of size $m = [an], 0 < a < 1$, are drawn, where $[x]$ denotes the largest integer value smaller than or equal to x. In the present study bootstrap samples of size $m = n/2$ are used, as was recommended by Bühlmann (2004, p.884). In all leave-one-out cross-validation algorithms, resampling is done *without* replacement as Hall and Robinson (2009) recommend to avoid difficulties caused by ties. In all plug-in bandwidth selection algorithms resampling is done *with* replacement. To save computer time, and for reasons previously mentioned, we allowed only one boosting iteration.

To determine the approximated MISE, as will be seen from the main algorithm specified below, the pointwise MSE has to be calculated for a grid of x-values. A fixed grid is constructed between the values -2 and +2 and 100 grid points are used. Both Marzio and Taylor (2008) and Hall and Robinson (2009) used $MC = 200$ in their simulation studies. In the present study 200 Monte Carlo samples ($MC = 200$) are generated for each of the sample sizes $n = 50$, $n = 100$ and $n = 200$. For a particular combination of model choice, underlying distribution of the covariate variable X and kernel choice, the Nadaraya-Watson with cross-validation bandwidth selection and plug-in bandwidths, the classical local linear estimator with plug-in bandwidth selection and the classical BRNP estimator with cross-validation bandwidth selection were determined. These methods are reflected in step 3 of the main algorithm. Also, for each model and each estimation method mentioned above, the various improvement methods discussed in Section 4 are applied. These improvement methods are listed in Section 4.3. All calculations were done using R 2.15.1 (R Core Team, 2014). The computer code is available on request.

6. The main algorithm

The main algorithm covers the complete process from data generation to evaluating the discrepancy measures as it was explained above.

1. Draw a random sample of size n from the underlying distribution and denote the sample by $\mathbf{S}_n = \{(X_1,Y_1),(X_2,Y_2),\ldots,(X_n,Y_n)\}$.

2. Construct a grid of x-values and denote the minimum grid value by x_{min} and the maximum grid value by x_{max}. In each grid point, x, estimate $m(x)$, for each of the three models defined in

Table 1, by utilising one of the 32 specific estimation methods described above, in a specific parameter scenario.

3. Repeat the above steps MC times and denote the estimates by $\hat{m}_i(x)$, $i = 1,2,\dots,MC$, where $x_{min} = x_1$ and $x_{max} = x_{100}$ The following matrix is obtained

$$\begin{bmatrix} \hat{m}_1(x_{min}) & \hat{m}_2(x_{min}) & \cdots & \hat{m}_{MC}(x_{min}) \\ \hat{m}_1(x_2) & \hat{m}_2(x_2) & \cdots & \hat{m}_{MC}(x_2) \\ \vdots & \vdots & \vdots & \vdots \\ \hat{m}_1(x_{max}) & \hat{m}_2(x_{max}) & \cdots & \hat{m}_{MC}(x_{max}) \end{bmatrix}.$$

4. For the chosen model and estimation method, calculate Monte Carlo approximations to the mean-squared error (MSE), variance and squared bias in each grid point x:

$$\widehat{\text{MSE}}[\hat{m}(x)] = \frac{1}{MC}\sum_{i=1}^{MC}[\hat{m}_i(x) - m(x)]^2,$$

$$\widehat{\text{Var}}[\hat{m}(x)] = \frac{1}{MC}\sum_{i=1}^{MC}\left[\hat{m}_i(x) - \frac{1}{MC}\sum_{i=1}^{MC}[\hat{m}_i(x)]\right]^2$$

and

$$\widehat{\text{Bias}}^2[\hat{m}(x)] = \left[\frac{1}{MC}\sum_{i=1}^{MC}[\hat{m}_i(x)] - m(x)\right]^2.$$

5. Use numerical integration to integrate over the entire range of x-values to obtain approximated global measures:

$$\widehat{\text{MISE}} = \int_{x_{min}}^{x_{max}} \widehat{\text{MSE}}[\hat{m}(x)]dx,$$

$$\text{Approximate Integrated Variance} = \int_{x_{min}}^{x_{max}} \widehat{\text{Var}}[\hat{m}(x)]dx$$

and

$$\text{Approximate Integrated Bias}^2 = \int_{x_{min}}^{x_{max}} \widehat{\text{Bias}}^2[\hat{m}(x)]dx.$$

6. Repeat for all models in Table 1 and for all estimation method procedures.

For numerical integration, the trapezoid rule numerical integration was used, utilizing the R package *caTools* (Tuszynski, 2014). The results of the various simulation executions have been summarized, studied and evaluated.

7. Results and conclusions

The results of the 32 procedures, some ranging over 36 scenarios, have been summarized into 324 tables, which were regrouped into 36 sub-tables. These results are available on request. However, we state the main results below along broader lines, to indicate the main effects of boosting, bagging and bragging on the final estimators, and to point out specific comparative observations between the estimators and the applied bandwidths. In general, the following is true for all scenarios and all models:

a) The MISE decreases as the sample size increases. This is to be expected, since the underlying regression relationship $m(x)$ will be more evident from large samples carrying a lot of information than from smaller samples.

b) Increasing values of the error variance, σ^2, result in larger MISE values, since $m(x)$ will be more difficult to estimate from data that were generated to deviate much from $m(x)$, compared to data generated to follow $m(x)$ closely, i.e., data constructed with small error variance.

c) The $N(0,1)$ kernel performs slightly better with regard to the smallest MISE, due to a longer tail and therefore including more data than the triangular kernel.

d) The underlying distribution of the covariate data influences the MISE values. Smaller MISE values are observed when $X \sim U(-2,2)$, compared to $X \sim N(0,1)$.

e) Throughout the study it is evident that the results obtained by the various procedures, are also model dependent.

For the discussion below we use the terms *classical Nadaraya-Watson estimator, classical local linear estimator and classical BRNP-estimator*, when referring to the *NW*-estimator, the *LL*-estimator and the *BRNP*-estimator, without any improvements applied. The appropriate bandwidth-selection method used will be indicated by a subscript. This notation will enable comfortable remarks and discussions below, regarding the effects of the improvement methods on the various estimators.

7.1. Effects on the Nadaraya-Watson estimator with cross-validation bandwidth (NW_{CV})

a) In general bagging reduced the variance of NW_{CV}.

b) For Models 1 and 2, in all 36 simulation setup scenarios, the boosting procedure led to a smaller integrated squared bias value compared to the integrated squared bias value of the classical NW_{CV}. For Model 3, this was also true for most cases of Bag1, Bag3 and Brag3. But Bag2, Brag1 and Brag2 inflated the variance.

c) In 86% of the simulation scenarios for Model 3, Bag3 performed better than Bag1 and Bag2 in terms of reducing the variance component of the classical NW_{CV}, while in 78% of the scenarios Bag3 also produced the lowest MISE.

d) In most simulation scenarios the bagged NW_{CV} performed better compared to the classical NW_{CV}, in the sense that the MISE was smaller overall. Bag1 performed the best, compared to Bag2 and Bag3.

e) Similarly, bragging reduced the variance component of the classical NW_{CV}-estimator. In general, the bragged NW_{CV} performed better than the classical NW_{CV}, in the sense that the MISE was smaller overall. Brag3 performed better than Brag1 and Brag2, in most simulation scenarios, in terms of reducing the variance component of the classical NW_{CV}, but in most simulation scenarios Brag1 performed the best, compared to Brag2 and Brag3, in terms of providing a lower value for the MISE.

f) The boosted NW_{CV} always had a smaller integrated squared bias value compared to the integrated squared bias value of the classical NW_{CV}, but the lower bias was almost always accompanied by inflated variance. This conclusion was also confirmed by Marzio and Taylor (2008) and Boshoff (2009). Therefore, the boosted NW_{CV} outperformed the classical NW_{CV} in terms of a lower MISE value, only in limited cases. This happened when $X \sim N(0,1)$ and the error variance was small.

7.2. Effects on the Nadaraya-Watson estimator with plug-in bandwidth (NW_{plug})

a) For all three models, in all 36 simulation setup scenarios, the boosting procedure led to a smaller integrated squared bias value compared to the integrated squared bias value of the classical NW_{plug}, usually due to often significant variance inflation. In only limited cases were the boosted MISE lower than the MISE obtained from the classical method. Boosting seemed to be most effective in simulation setup scenarios where the error variance was small.

b) However, in limited cases, bagging reduced the variance of the NW_{plug}. Again, bagging was the most effective in the case of small sample sizes, $X \sim N(0,1)$ and when the triangular kernel was used. Also, in limited cases the bagged NW_{plug} performed better in terms of lower MISE overall, compared to the classical NW_{plug}. Bag3 performed the best, compared to Bag1 and Bag2, in most simulation scenarios, in terms of reduced MISE.

c) Similar deductions can be made for bragging. In limited cases the bragged NW_{plug} outperformed the classical NW_{plug} in terms of the lowest MISE. In most simulation scenarios Brag1 or Brag3 performed the best of the bragging procedures in terms of MISE.

7.3. Effects on the local linear estimator with plug-in bandwidth (LL_{plug})

a) Similar results were obtained as those for NW_{plug}. In cases where the bagged or bragged LL_{plug}-estimator performed better than the classical LL_{plug} in terms of the MISE, it was due to a smaller variance component. Bagging and bragging indeed reduced the variance of LL_{plug} in a substantial number of scenarios.

b) Bag3 and Brag3 were the most effective improvement methods. The Bag3 and Brag3 algorithms produced estimates for $m(x)$ for each bootstrap sample separately and then aggregated these estimates. The aim of the Bag3 and Brag3 methods therefore is to reduce the variability of $\hat{m}(x)$.

c) In many scenarios the bagging and bragging methods were the most effective in terms of smaller MISE, when sample sizes were small.

d) For all three models, in all 36 simulation setup scenarios, the boosting procedure led to a smaller integrated squared bias compared to the integrated square bias of the classical LL_{plug}. Boosting indeed reduced the bias of the classical LL_{plug}. Unfortunately, the boosting improvement method was not effective in terms of the MISE. In most of the simulation setup scenarios, there were a significant increase in the integrated squared variance value. This increase in the variance component resulted in a MISE larger than that of the classical LL_{plug}. Only in a limited number of cases the boosted LL_{plug} had a smaller MISE.

7.4. Effects on the BRNP estimator with cross-validation bandwidth ($BRNP_{CV}$)

a) In several scenarios Bag1 and Brag1 and even Bag3 achieved smaller MISE values than the classical $BRNP_{CV}$. In all such cases the covariate distribution was $U(-2,2)$ and the $N(0,1)$ kernel was used. Bagging and bragging indeed reduced the variance of the classical $BRNP_{CV}$. Bag1 and Brag1 were the most effective improvement methods in terms of the MISE, for Model 2 and Model 3, for almost all choices of sample size and error variance.

b) For all three models, for almost all simulation setup scenarios, the boosted $BRNP_{CV}$ had a smaller integrated squared bias value compared to the integrated squared bias value of the classical $BRNP_{CV}$. The only exception was for Model 3, $X \sim U(-2,2)$, $\sigma = 0.5$ and $n = 100$. Also, in most cases, boosting improved the classical $BRNP_{CV}$ in terms of the MISE values. Hence, we can conclude that boosting was very effective in reducing the bias of the classical $BRNP_{CV}$ estimator. However, an increase in the error variance led to an increase in the values of the discrepancy measure in a limited number of cases.

8. Comparing NW_{CV} to the NW_{plug}

a) In order to compare the results of NW_{CV} to the results of NW_{plug}, a ratio of the obtained MISE values is used:

$$\text{ratio} = \frac{\text{MISE}_{NW_{CV}}}{\text{MISE}_{NW_{plug}}}. \tag{15}$$

If this ratio is larger than one, then NW_{plug} has the smaller MISE value for a specific simulation scenario. If the ratio is less than one, the opposite is true. The results of the comparisons are displayed in Table 2. Recall that for the three models there were 36 simulation setup scenarios, i.e., three sample sizes, three error variances, two kernels and two covariate distributions were considered. Table 2 provides a summary of the percentage of times the classical NW_{CV} performed best in terms of MISE, and the percentage of times the classical NW_{plug} performed best in terms of MISE. We now discuss the results and draw conclusions.

b) From Table 2 it is clear that in the classical case, as well as when boosting were applied, and for almost all the bagging methods, in almost all simulation setup scenarios, the classical NW_{plug} outperformed NW_{CV} in terms of smallest MISE. For Model 1, the classical NW_{plug} had a smaller MISE, compared to that of NW_{CV} for 75% of the simulation setup scenarios. No general pattern could be established since all covariate and kernel distributions, all σ-values and sample sizes were involved. Similar results were obtained for the other models and all estimation procedures.

Table 2: Percentage of times the estimator performed best in terms of MISE for the 36 simulation scenarios.

	Model 1		Model 2		Model 3	
	NW_{CV}	NW_{plug}	NW_{CV}	NW_{plug}	NW_{CV}	NW_{plug}
Classical	25%	75%	31%	69%	19%	81%
Bag1	53%	47%	64%	36%	36%	64%
Bag2	33%	67%	47%	53%	25%	75%
Bag3	33%	67%	53%	47%	50%	50%
Boost	19%	81%	11 %	89%	17%	83%

c) It should be mentioned that for the various scenarios, the MISE profits were not dramatic throughout. MISE comparisons were of the order $\frac{0.02658}{0.02459} = 1.08$ for a large portion of the scenarios. However, for the ratio defined in (15) when comparing the classical estimators, 53% of the MISE ratios were between 1 and 1.2, while 23% were above 1.2 but less than 1.7.

d) Boosting favoured the classical NW_{plug} instead of the classical NW_{CV}, since in 45% of the scenarios the ratio (15) were between 1.2 and 1.7. The bagging procedures showed in 44% of the scenarios ratio values between 1 and 1.2 and only in 9% values above 1.2. Bragging procedures behaved similarly.

9. Comparing LL_{plug} to NW_{plug}

In this section, similar ratios as in (15) were calculated for all simulation scenarious, to compare LL_{plug} with NW_{plug}. The following results were obtained:

Table 3: Percentage of times the estimator performed best in terms of MISE for the 36 simulation scenarios.

	Model 1		Model 2		Model 3	
	LL_{plug}	NW_{plug}	LL_{plug}	NW_{plug}	LL_{plug}	NW_{plug}
Classical	8%	92%	31%	69%	22%	78%
Bag1	6%	94%	31%	69%	19%	81%
Bag2	11%	89%	33%	67%	28%	72%
Bag3	28%	72%	36%	64%	28%	72%
Boost	3%	97%	6%	94%	6%	94%

a) From Table 3 it is evident that NW_{plug} yielded smaller MISE-values than LL_{plug}. In fact, in 34% of the scenarios the ratio between the MISE-values, in the classical case, were above 1.2, and ratio-values of 1.8 occurred. When boosting were applied, ratio-values even exceeded 4.00, and in 58% of the scenarios the ratios-values were above 1.2. Similar results were found for the bagging procedures.

b) Results for the bragging procedures were similar to the results obtained from the bagging procedures, and have been omitted.

10. Comparing $BRNP_{CV}$ to NW_{CV}

This comparison could not be done because the number of Monte Carlo replications were not the same for the two procedures. Only 20 Monte Carlo replications were used to obtain results for the BRNP estimator, because procedures for this estimator is very computer time intensive. See the following paragraph for specific information.

11. Comparing the improvement methods

One of the most important aspects of this paper is to evaluate the improvement methods, i.e., the three bagging methods, the three bragging methods and the boosting procedure. From the discussions in Section 7 it is evident that the behaviour of the improvement methods are model dependent and they show different behaviour for the estimators NW_{CV}, NW_{plug}, LL_{plug} and $BRNP_{CV}$. From Tables 2 and 3 it is also clear that specific estimators of $m(x)$ favour different methods of derived bandwidths and the boosting procedure does not suit all estimation methods of $m(x)$ equally well in terms of small MISE.

To illustrate that no fixed general rule exists to select the best improved bandwidth selection method for the four estimators of $m(x)$ in terms of smallest MISE, and to verify the results stated in Section 7.1, the upper part of Table 4 shows the percentage of times (out of 36 scenarios) that the bag procedures produced the smallest MISE for the three models *among the bag procedures*, for the NW_{CV} estimator. The lower part displays similar percentages for the brag procedures *among the brag procedures*, for the three models, for the NW_{CV} estimator. However, Table 5 shows a percentage comparison of all the improvement methods simultaneously, for Models 1-3, for the NW_{CV} estimator, i.e., it shows the percentage of times (out of a possible 36 scenarios) that each of the bagging, bragging and boosting procedures was responsible for the smallest MISE, for the NW_{CV} estimator. The results displayed in Tables 4 and 5 regarding the behaviour of the improvement methods based on the NW_{CV} estimator, therefore confirm the discussions and conclusions contained in Section 7.1. Similarly, tables for the other estimators show similar model and scenario-specific behaviour and are omitted. However, the main findings acquired from these tables are captured in Sections 7-11 and 14.

Table 4: Percentage of times bagging and bragging procedures performed best among bagging procedures and bragging procedures respectively in terms of MISE, for the 36 simulation scenarios.

	Model 1	Model 2	Model 3
Bag1	80.56%	61.11%	19.44%
Bag2	2.78%	13.89%	2.78%
Bag3	16.67%	25.00%	77.78%
Brag1	88.89%	63.89%	38.89%
Brag2	0.00%	2.78%	0.00%
Brag3	11.11%	33.33%	61.11%

Table 5: Percentage of times bagging, bragging and boosting procedures performed best in terms of MISE, for the 36 simulation scenarios.

	Model 1	Model 2	Model 3
Bag1	33.33%	44.44%	8.33%
Bag2	2.78%	13.89%	0.00%
Bag3	16.67%	25.00%	66.67%
Brag1	38.89%	11.11%	16.67%
Brag2	0.00%	0.00%	0.00%
Brag3	0.00%	0.00%	8.33%
Boost	8.33%	5.56%	0.00%

12. Computer-time

It was shown that the classical NW_{plug} and LL_{plug} performed just as good as the classical NW_{CV} in terms of small MISE-values. Typical computation time (in seconds) for the 1000 Monte Carlo iterations and all the models considered, are:

If $n = 200$, $\sigma = 0.2$, the kernel is Gaussian and the covariate distribution is $U(-2,2)$, then the following times (in seconds) are consumed by calculating classical versions of NW_{CV}, NW_{plug} and LL_{plug} respectively: 1021.74, 4.43 and 3.47. For a triangular kernel the number of seconds become 2725.45, 8.88 and 6.35. If $n = 50$, these numbers become a third of the times.

If the Bag1 procedure is executed, for $n = 200$, $\sigma = 0.2$, the kernel is Gaussian and the covariate distribution is $U(-2,2)$, then the following times are used by NW_{CV}, NW_{plug} and LL_{plug} respectively in seconds: 14279, 153.37 and 37.49. For a triangular kernel the number of seconds become 32935.96, 309,12 and 39.67. Bag2 and Bag3 are a little more time consuming than Bag1. Time for the boosting procedures are almost similar to the times used for classical estimators. However, to determine the classical $BRNP_{CV}$ takes up to 16009.51 seconds and the boosting process another 13592.29 seconds. It is therefore clear that a nonparametric regression estimator utilizing plug-in bandwidth selection methods is computationally more efficient than a nonparametric regression estimator utilizing cross-validation bandwidth selection methods.

In view of the above findings, the NW_{plug} is a nonparametric regression estimator to be recommended for practical purposes, since it is not only a fairly precise and accurate estimator, but it is also computationally much faster than other nonparametric regression estimators considered in this study. It is clear that Yao's estimator is a very computer time intensive estimator compared to the other three estimators. In future studies a plug-in bandwidth selection method for Yao's estimator needs to be derived to improve the computation time of this estimator.

13. Graphical illustrations

Figures 2, 3 and 4 present graphs, derived from simulation studies, to illustrate general aspects of the behaviour of the estimators. The following conclusions are particularly clear from these graphs:

Due to bias reduction, the boosted estimators produce curves that are slightly closer to the underlying regression curve $m(x)$ than the standard estimators. Furthermore, the bagged estimators produce smoother curves, i.e., curves with less variability, than the standard estimators. Only graphs of sample size $n = 200$ are provided to display the behaviour of the four estimators. For smaller sample sizes, graphs show slightly higher variance and bias. As n increases, the estimates lie closer to $m(x)$ (i.e., the integrated bias decreases) and the variability becomes smaller (i.e., the integrated variance decreases).

14. Final remarks and recommendations

This study focused on bias reduction techniques in nonparametric kernel regression, but variance reduction was also a major point of interest throughout. The following final remarks and recommendations regarding the study can be made:

a) The behaviour of the classical NW_{plug} proved to be a recommendable nonparametric regression estimator, since it is not only as precise and accurate as any of the other estimators, but it is also computationally much faster than any other nonparametric regression estimators considered in this study.

b) Boosting reduced the bias component of the classical NW_{CV} in all simulation scenarios considered. Boosting was not effective in terms of providing an estimator for $m(x)$ with a MISE value lower than that of the classical NW_{CV}, due to variance inflation. In general, the boosted NW_{plug} performed better than the classical NW_{plug}. This is also true for LL_{plug}. Small sample sizes and small variances suit the boosting procedure.

c) For the $BRNP_{CV}$, boosting reduced the bias of the estimator in all simulation scenarios considered. Since the $BRNP_{CV}$ is a very computer time intensive estimator, only 20 Monte Carlo iterations were performed. A larger study is on its way, including a new $BRNP_{plug}$-estimator and its properties and behaviour.

d) Bagging methods are very effective in reducing the variance component of the classical NW_{CV}. Specifically Bag3 performed well in terms of reducing the variance component of the classical NW_{CV}. Bag1 performed best in terms of lower MISE than Bag2 and Bag3. Bragging revealed the same patterns as bagging in this case with regard to NW_{CV}. Bagging and bragging methods were also effective in reducing the variance component of $BRNP_{CV}$ in limited cases. In general the bias component of the estimators with cross-validation bandwidth selection increased when bagging and bragging improvement methods were applied to the estimators.

e) Bagged and bragged estimators were not effective when plug-in bandwidth selection methods were used.

f) The MISE decreased as the sample size increased. Increasing values of the error variance, resulted in larger MISE values, since $m(x)$ would be more difficult to estimate from data that were generated to deviate from $m(x)$ much by means of a large error variance, compared to data generated to follow $m(x)$ closely.

g) By using the $N(0,1)$ kernel, smaller MISE values were obtained in most cases, compared to the triangular kernel. This is to be expected, since the $N(0,1)$ kernel has longer tails and therefore more data are included in the estimation process than when using the restricted triangular kernel.

h) Smaller MISE values were obtained when covariate data were from the U(-2,2) distribution than for the scenarios where data were drawn from the $N(0,1)$ distribution.

i) Despite all the effort and new insight into the improvement methods and estimators, no specific rules could be set to predict favourable results. For various models, various scenarios were favoured by the estimators. However, for large samples and not too large variances, non-parametric regression estimators could be presented with reliable properties.

j) Figures 2–5 illustrate general aspects of the simulation studies, such as the bias and variance behaviour of the various methods.

k) Computer package used: All calculations are done using R 2.15.1 (R Core Team, 2014).

Acknowledgement

The authors wish to thank the reviewers for their valuable suggestions which improved the presentation of the paper.

References

BOSHOFF, L. (2009). *Boosting, Bagging, and Bragging Applied to Nonparametric Regression – An Empirical Approach*. Master's thesis, Potchefstroom: NWU.

BREIMAN, L. (1996). Bagging predictors. *Machine Learning*, **24**, 123–140.

BÜHLMANN, P. (2003). Bagging, subagging and bragging for improving some prediction algorithms. *In* AKRITAS, M. G. AND POLITIS, D. N. (Editors) *Recent Advances and Trends in Nonparametric Statistics*. Elsevier, Amsterdam, pp. 9–34.

BÜHLMANN, P. (2004). Bagging, boosting and ensemble methods. *In* GENTLE, J. E., HÄRDLE, W., AND MORI, Y. (Editors) *Handbook of Computational Statistics: Concepts and Methods*. Springer, Berlin, pp. 877–907.

CHUNG, Y. AND LINDSAY, B. G. (2011). A likelihood-tuned density estimator via nonparametric mixture model. *In* HUNTER, D. R., RICHARDS, D. S. P., AND ROSENBERGER, J. L. (Editors) *Nonparametric Statistics and Mixture Models: A Festschrift in Honor of Thomas P. Hettmansperger*. World Scientific Publishing: Hackensack, NJ, pp. 69–89.

FAN, J. (1992). Design-adaptive nonparametric regression. *Journal of the American Statistical Association*, **87** (420), 998–1004.

FAN, J. AND GIJBELS, I. (1996). *Local Polynomial Modelling and Its Applications*, volume 66 of *Monographs on Statistics and Applied Probability*. Chapman and Hall: London.

HALL, P. AND ROBINSON, A. P. (2009). Reducing variability of crossvalidation for smoothing-parameter choice. *Biometrika*, **96** (1), 175–186.

HÄRDLE, W. (1990). *Applied Nonparametric Regression*. Cambridge University Press: Cambridge.

MARZIO, M. D. AND TAYLOR, C. C. (2008). On boosting kernel regression. *Journal of Statistical Planning and Inference*, **138**, 2483–2498.

NADARAYA, E. A. (1964). On estimating regression. *Theory of Probability and its Applications*, **9**, 141–142.

R CORE TEAM (2014). *R: A language and Environment for Statistical Computing*. R Foundation for Statistical Computing: Vienna, Austria.
URL: `http://www.R-project.org/`

SWANEPOEL, J. W. H. (1988). Point estimation based on approximating functionals and the bootstrap. *Technical Report, Dept of Statistics, Potchefstroom University*.

SWANEPOEL, J. W. H. (1990). A review of bootstrap methods. *South African Statistical Journal*, **24**, 1–34.

TUKEY, J. (1977). *Exploratory Data Analysis*. Addison Wesley: Reading, MA.

TUSZYNSKI, J. (2014). *caTools: Tools: moving window statistics, GIF, Base64, ROC AUC, etc.*
URL: `http://CRAN.R-project.org/package=caTools`

WATSON, G. S. (1964). Smooth regression analysis. *Sankhyā Series A*, **26**, 359–372.

YAO, W. (2012). A bias corrected nonparametric regression estimator. *Statistics and Probability Letters*, **82** (2), 274–282.

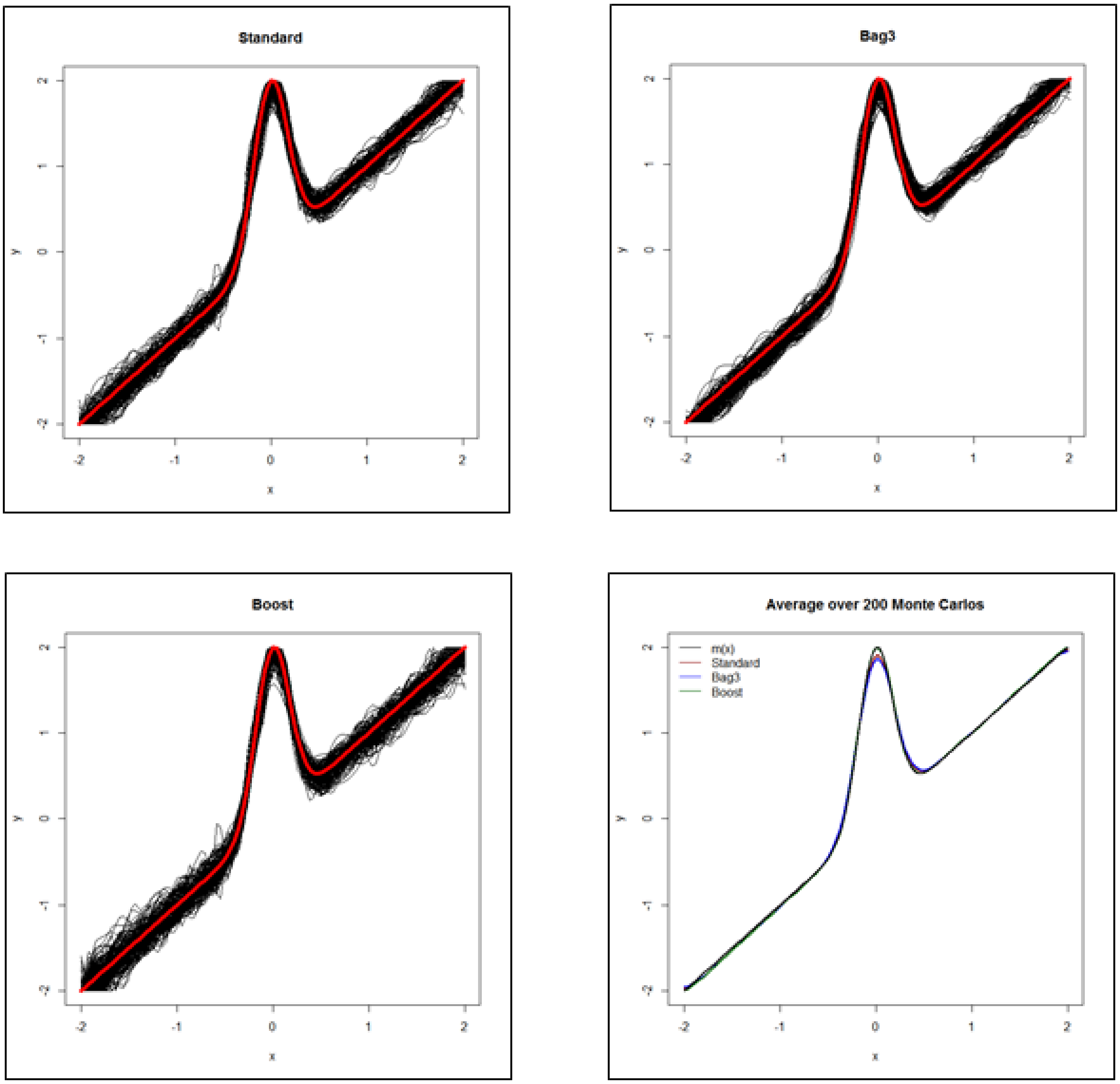


Figure 2: Model 1, $X \sim U(-2,2)$, $N(0,1)$ kernel, $\sigma = 0.2$, $n = 200$.

Top left: The classical (standard) NW_{CV}. *Top right:* The bagged NW_{CV}. *Bottom left:* The boosted NW_{CV}. The black lines represent the estimates for each of the 200 Monte Carlo simulations. The red line indicates $m(x)$. *Bottom right:* The average estimates over the 200 Monte Carlo simulations for Standard, Boost, Bag3, as indicated in the legend.

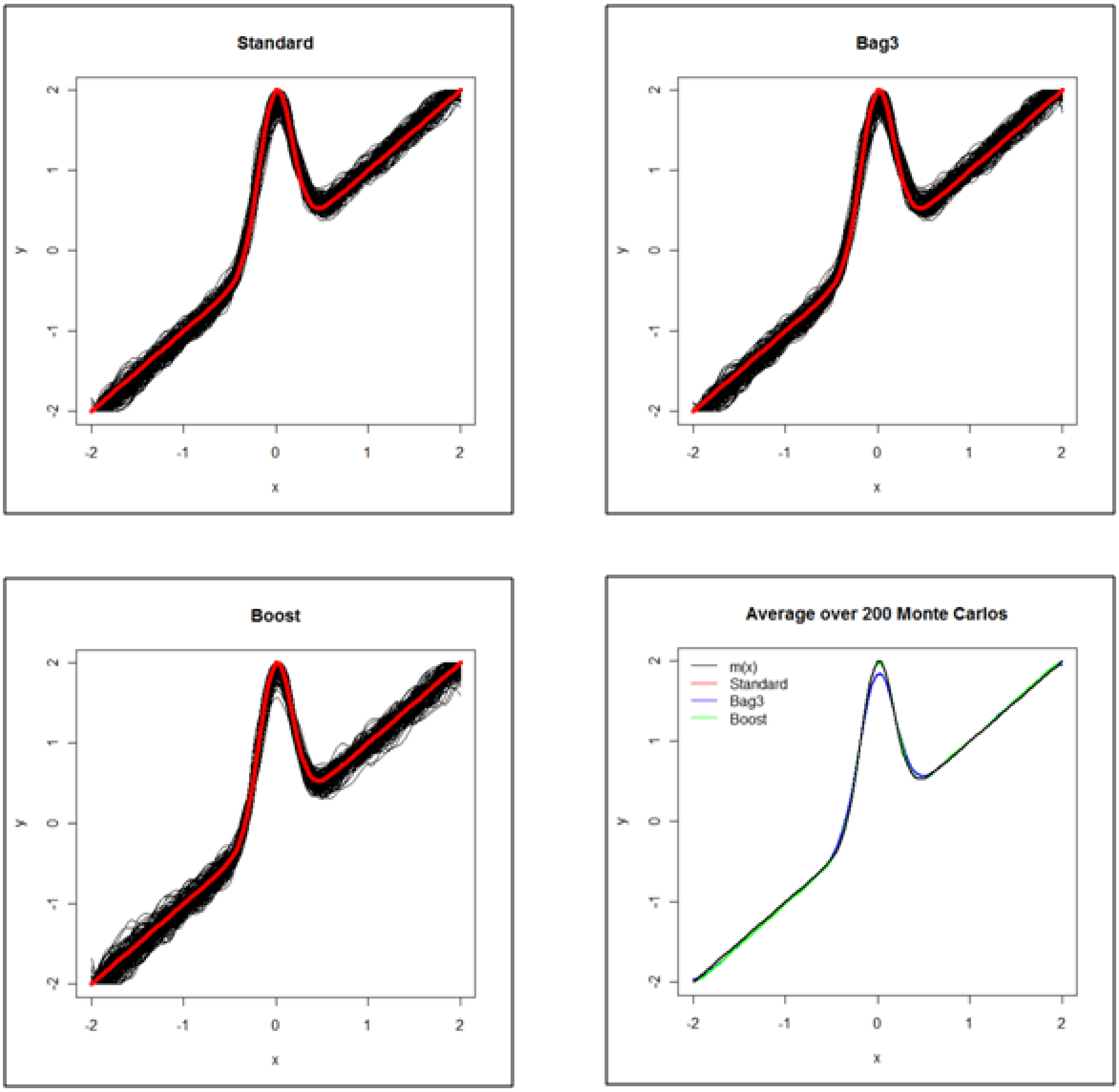
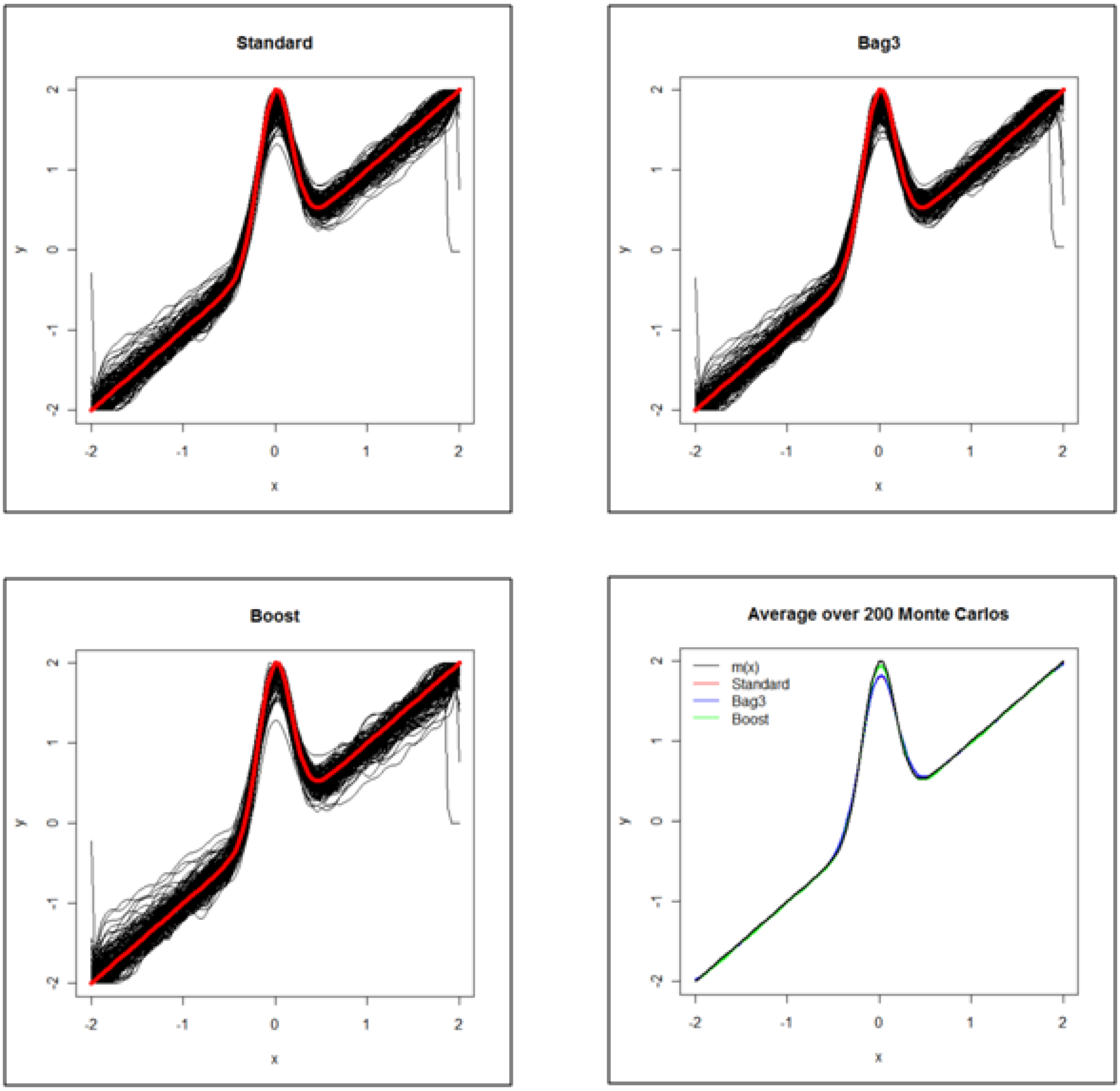


Figure 3: Model 1, $X \sim U(-2,2)$, $N(0,1)$ kernel, $\sigma = 0.2$, $n = 200$.

Top left: The classical (standard) NW_{plug}. *Top right:* The bagged NW_{plug}. *Bottom left:* The boosted NW_{plug}. The black lines represent the estimates for each of the 200 Monte Carlo simulations. The red line indicates $m(x)$. *Bottom right:* The average estimates over the 200 Monte Carlo simulations for Standard, Boost, Bag3, as indicated in the legend.



Standard

Bag3

Boost

Average over 200 Monte Carlos

Figure 4: Model 1, $X \sim U(-2,2)$, $N(0,1)$ kernel, $\sigma = 0.2$, $n = 200$.

Top left: The classical (standard) LL_{plug}. *Top right:* The bagged LL_{plug}. *Bottom left:* The boosted LL_{plug}. The black lines represent the estimates for each of the 200 Monte Carlo simulations. The red line indicates $m(x)$. *Bottom right:* The average estimates over the 200 Monte Carlo simulations for Standard, Boost, Bag3, as indicated in the legend.

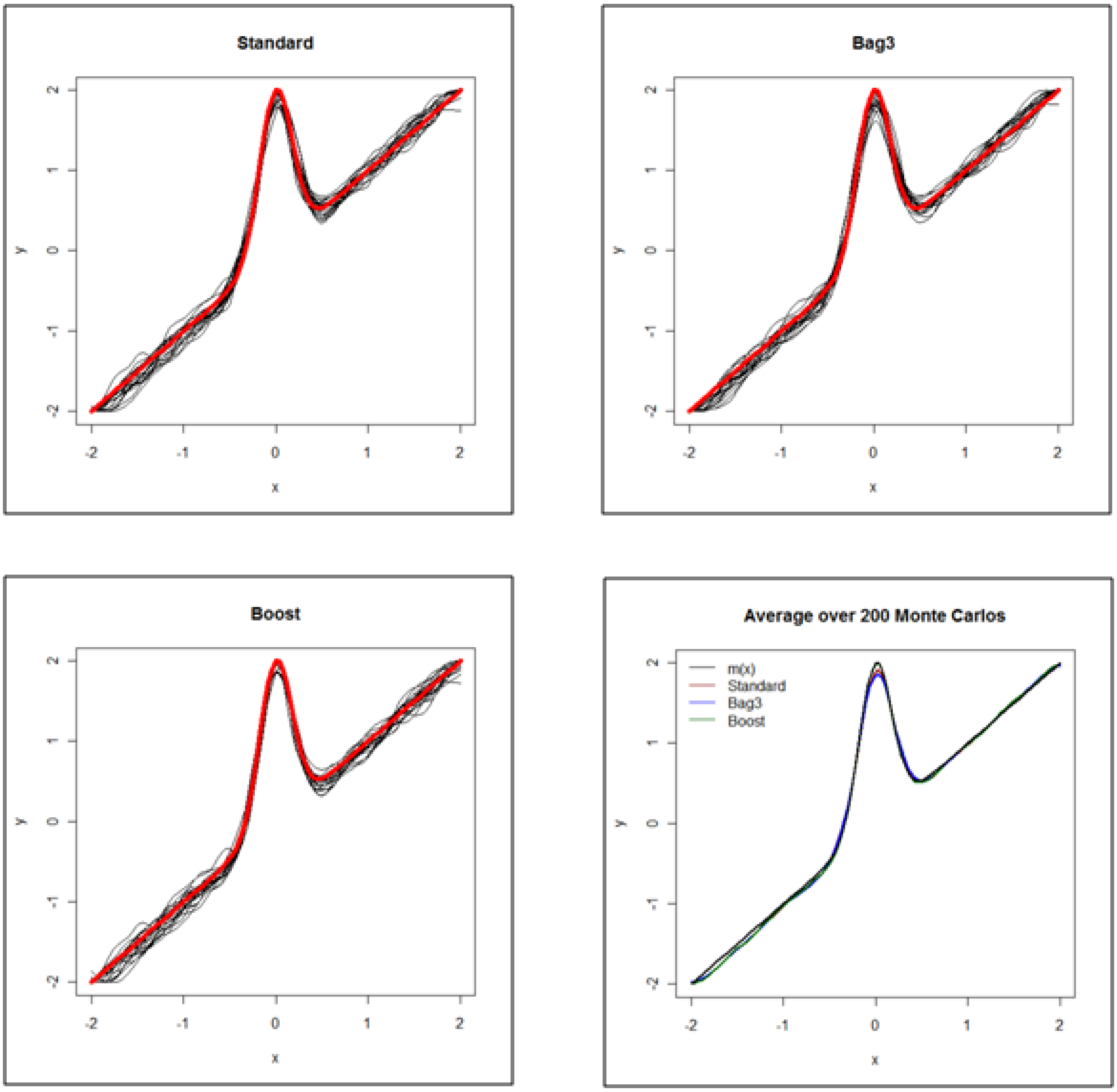


Figure 5: Model 1, $X \sim U(-2,2)$, $N(0,1)$ kernel, $\sigma = 0.2$, $n = 200$.

Top left: The classical (standard) $BRNP_{CV}$. *Top right:* The bagged $BRNP_{CV}$. *Bottom left:* The boosted $BRNP_{CV}$. The black lines represent the estimates for each of the 200 Monte Carlo simulations. The red line indicates $m(x)$. *Bottom right:* The average estimates over the 200 Monte Carlo simulations for Standard, Boost, Bag3, as indicated in the legend.

Manuscript received, 2015-06-09, revised, 2015-11-05, accepted, 2016-01-16.