Electronic Journal of Statistics
Vol. 8 (2014) 2848–2878
ISSN: 1935-7524
DOI: 10.1214/14-EJS966

Global adaptive smoothing regression[†,*]

Francesco Giordano

Department of Economics and Statistics, University of Salerno
Via Giovanni Paolo II, 84084 Fisciano, Italy
e-mail: giordano@unisa.it

and

Maria Lucia Parrella

Department of Economics and Statistics, University of Salerno
Via Giovanni Paolo II, 84084 Fisciano, Italy
e-mail: mparrella@unisa.it

Abstract: We propose an adaptive smoothing method for nonparametric regression. The central idea of the proposed method is to "calibrate" the estimated function through an *adaptive bandwidth function*, which is a kind of intermediate solution between the global bandwidth (constant on the support) and the local bandwidth (variable with x). This also allows to correct the bias of the local polynomial estimator, with some benefits for the inference based on such estimators. Our method, which uses the Neural Network technique in a preliminary (pilot) stage, is based on a rolling, plug-in, bandwidth selection procedure. It automatically reaches a trade-off between the efficiency of global smoothing and the adaptability of local smoothing. The consistency and the optimal convergence rate of the resulting bandwidth estimators are shown theoretically. A simulation study shows the performance of our method for finite sample size.

AMS 2000 subject classifications: 62G08, 65D10, 82C32.

Received November 2013.

1. Aims and motivations

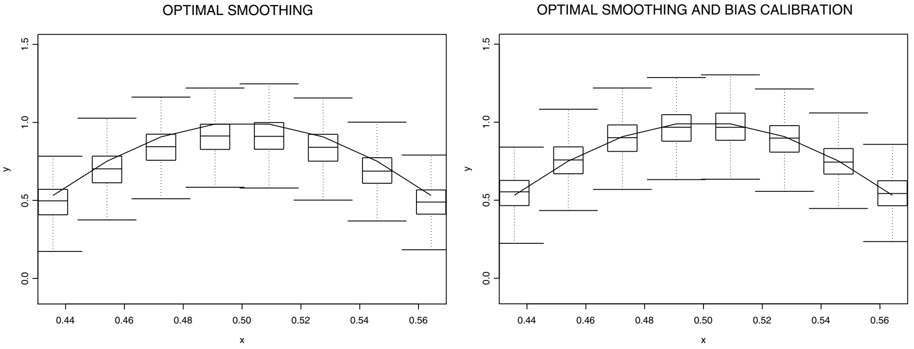
Consider the real bivariate process $\{Y, X\}$. A general regression setup is

$$m_\phi(x) = E\left\{\phi(Y)|X = x\right\}, \tag{1}$$

which includes several special cases through appropriate definition of the function ϕ. Given a realization of the process $\{Y_i, X_i; i = 1, \ldots, n\}$, the unknown function $m_\phi(\cdot)$ and its derivatives $m_\phi^{(v)}(\cdot)$ can be estimated nonparametrically

[*] Financial support by the Italian Ministry of Education, University and Research (MIUR), PRIN Research Project 2010–2011 – prot. 2010J3LZEN, "Forecasting economic and financial time series: understanding the complexity and modelling structural change", is gratefully acknowledge.

[†] The authors gratefully acknowledge support from the University of Salerno grant program "Finanziamento di attrezzature scientifiche e di supporto per i Dipartimenti e i Centri Interdipartimentali" – prot. ASSA098434, 2009.



(a) (b)

Fig 1. *Boxplots of the local polynomial estimations for the conditional mean function of model 6, used in the simulation study, with samples of* $n = 200$ *observations. The solid lines represent the true regression function. Plot in (a) is made using the estimated optimal bandwidth, but without bias calibration. Plot in (b) includes the procedure with bias calibration.*

using the local polynomial estimator (LPE). Denote with $\hat{m}_\phi(x;h)$ such estimator, where h denotes the *smoothing parameter (bandwidth)*. The good theoretical properties of the LPE and its conceptual simplicity determine the success of such estimators. But a serious drawback of the LPE is its strong dependence on the bandwidth parameter, which has a remarkable affect on the bias and the variance of the estimator. By studentizing the estimator, we can highlight its bias term

$$\begin{aligned}\frac{\hat{m}_\phi(x;h) - m_\phi(x)}{\sqrt{Var[\hat{m}_\phi(x;h)]}} &= \frac{\hat{m}_\phi(x;h) - E[\hat{m}_\phi(x;h)]}{\sqrt{Var[\hat{m}_\phi(x;h)]}} + \frac{E[\hat{m}_\phi(x;h)] - m_\phi(x)}{\sqrt{Var[\hat{m}_\phi(x;h)]}} \\ &= Z_n(x) + \frac{Bias[\hat{m}_\phi(x;h)]}{\sqrt{Var[\hat{m}_\phi(x;h)]}}.\end{aligned}$$

Now, if the true asymptotic optimal local bandwidth $h_L^{opt}(x)$ is used, we can show that

$$Z_n(x) \xrightarrow{d} N(0,1) \qquad \frac{Bias[\hat{m}_\phi(x;h_L^{opt})]}{\sqrt{Var[\hat{m}_\phi(x;h_L^{opt})]}} \longrightarrow \pm\sqrt{\frac{2v+1}{2(p-v+1)}}, \tag{2}$$

where p is the order of the local polynomial estimator. So, *even when the bandwidth is the best one*, there is a non-vanishing bias into the normal limit of the estimator, given that p and v are fixed (see also the example in the left panel of fig. 1). This has implications for the inference. For example, if we do not consider the bias term explicitly, the confidence intervals are not centred around the true value of the function. As a consequence, the coverage, the size and the power of the confidence intervals and tests are all affected. So, for optimal smoothing regression, it is necessary both optimal bandwidth selection and bias correction. Anyway, the bias and the bandwidth must be estimated, and the quality of such estimations has strong implications on the final estimate of the regression function.

The aim of this paper is to propose an adaptive smoothing method based on both data-driven bandwidth estimation and bias correction. Our method works through a rolling plug-in bandwidth selection procedure which also provides all the estimations required for bias correction, as will be explained in section 7.3.

Figure 1 shows an example. It reports the boxplots of the local polynomial estimations for the conditional mean function of model 6, used in the simulation study of section 7.1 (for samples of size $n = 200$). The solid lines represent the true regression function. Plot (a) is obtained using our estimated optimal bandwidth, but without bias correction. Plot (b) includes bias correction.

The central idea of our proposed method is to "calibrate" the local polynomial estimated function through an *adaptive bandwidth function*, which is a kind of intermediate solution between the global bandwidth (constant on the support) and the local bandwidth (variable with x). The motivation of our proposal is based on the consideration made in Wang & Gasser (1996) that the rate of convergence of a global bandwidth estimator is generally much faster than the rate of convergence of any local bandwidth estimator. So the advantage (adaptability) of the local bandwidth function may not tranfer in the estimated local bandwidth function, at least for small samples. With our rolling procedure, we estimate global bandwidths which are adaptive on the support, still maintaining the advantage of global smoothing (efficiency), as will be shown in the simulation study. Moreover, we also obtain better results compared with the local bandwidth estimators, such as the one proposed in Prewitt & Lohr (2006).

The most important theoretical result of this paper concerns the rates of convergence of the proposed bandwidth and bias estimators, which also influence the final rate of the local polynomial estimator $\hat{m}_\phi(x; h)$. We show in Theorem 1 that our global bandwidth estimator reaches the optimal rate of convergence, assuming conditions that are less stringent than those used by other global bandwidth estimators (as in Fan & Huang (1999)). Furthermore, the result for our local bandwidth estimator is even more interesting. In fact, we show in Theorem 2 that the proposed rolling procedure is a useful "trick" which lets to estimate the true local bandwidth asymptotically, with a relative rate of convergence which confirms the optimal relative rate of convergence for any local bandwidth estimator shown in Wang & Gasser (1996).

The method proposed in this paper for bandwidth selection differs from the usual plug-in procedures for local bandwidth selection that have been proposed in the context of regression estimation. In particular, it differs from the approach used in the paper of Prewitt & Lohr (2006), which directly estimates the local bandwidth function $h_L^{opt}(x)$. This and similar approaches usually produce highly variable bandwidth functions. On the other hand, it differs from the approaches which partition the X space and apply a global methodology to each subset, such as Fan & Gijbels (1995). Both of these methods require a *smoothing* of the estimated bandwidth function, to avoid roughness or change in the bandwidths between partitions. As a consequence, they introduce additional smoothing selection problems in the final stage of the procedure (say *post-smoothing*). Our approach avoids this problem by introducing a rolling scheme with an automatic rule for the selection of the rolling interval, here denoted with a, which determines the right "degree of locality". Our method borrows the idea of Hall & Schucany (1989) of considering an interval around the point of estimation x, although they use a cross-validation procedure for density estimation, while we propose a plug-in method for regression estimation, with better rates of convergence.

The smoothing method proposed in this paper uses the neural networks in a preliminary (pilot) stage, in order to estimate the optimal bandwidth. A parameter d, denoting the number of nodes of the neural network function, is therefore introduced and must be set externally to our smoothing procedure. The presence of such a *pilot parameter* is not new in the context of kernel regression. In fact, it has the same role as the well known *pilot bandwidths* used in the classic plug-in bandwidth selection procedures. Anyway, even if the role is the same, there are important differences in their behaviour. First of all, setting the parameter d is much simpler than setting the pilot bandwidths, as it will be explained in section 3. Moreover, we will show in the simulation study that variations in (misspecification of) the parameter d have little effect on the final results of the smoothing regression, contrary to what happens with the pilot bandwidths.

This paper is organised as follows. Section 2 introduces the adaptive smoothing method. Sections 3 and 5 describe in details the estimation procedure based on the neural networks technique. In section 4 we propose a methodology to select the optimal value of the parameter a. In sections 6 and 8 we derive the rate of convergence for our bandwidth estimators, in particular section 6 concerns the general bandwidth estimator while section 8 concerns the local bandwidth estimator, obtained asymptotically when $a = a_n \to 0$ for $n \to \infty$. Then we show the performance of the proposed method: section 7.1 shows the results of a simulation study; section 7.2 gives an example of application of our procedure to a real dataset, while section 7.3 explains how to correct the bias of the final local polynomial estimator. All the technical proofs are contained in the appendix.

2. Our proposal: From global to local smoothing

The local polynomial estimator is given by a weighted least squares regression

$$\hat{m}_\phi^{(v)}(x; h) = \sum_{i=1}^{n} \phi(Y_i) W_{K,v,p}(x - X_i; h), \tag{3}$$

where $v = 0, 1, \ldots$ denotes the degree of the derivative ($v = 0$ for the function itself) and the weights $W_{K,v,p}(\cdot; h)$ are derived from a kernel function, $K(\cdot)$, and from a local approximation of the function $m_\phi(x)$ through a polynomial of order p. The kernel function is a symmetric and bounded density function defined on $[-1, 1]$. The bandwidth h, which regulates the smoothness of the estimated function, must be such that $h \to 0$, $nh^{2v+1} \to \infty$ when $n \to \infty$.

Table 1

Coefficients $B_{p,K}^{(v)}$ and $V_{p,K}^{(v)}$ for the Epanechnikov kernel $K(u) = 0.75(1-u^2)\mathbb{I}(u \leq 1)$

	$B_{p,K}^{(v)}$			$V_{p,K}^{(v)}$		
v	$p=1$	$p=2$	$p=3$	$p=1$	$p=2$	$p=3$
0	1/10	—	1/504	3/5	—	5/4
1	—	1/14	—	—	15/7	—
2	—	—	1/36	—	—	35/4

The degree of the polynomial can be $p = 0, 1, 2, \ldots$, respectively for Nadaraya-Watson estimator, local linear estimator, local quadratic estimator, and so on. For theoretical reasons, the parameter p is generally fixed to $p = v + 1$ or in such a way that $p - v$ is odd. To slim the equations, we omit the symbols p, v and K from the notation of the estimator $\hat{m}_\phi(x; h)$. See Fan & Gijbels (1996) for a detailed description of the LPE and its properties.

The asymptotic mean squared error of the estimator $\hat{m}_\phi(x; h)$ can be decomposed, as is usual, into the sum of the squared asymptotic bias and asymptotic variance

$$\begin{aligned} AMSE\{\hat{m}_\phi(x;h)\} &= ABias^2[\hat{m}_\phi(x;h)] + AVar[\hat{m}_\phi(x;h)] \\ &= \mathbb{B}^2(x)h^{2(p+1-v)} + \mathbb{V}(x)\frac{1}{nh^{2v+1}}, \quad \forall x \in \mathbb{R}, \end{aligned} \tag{4}$$

where, denoting with $f_X(\cdot)$ the density of the random variable X and with σ_ϕ^2 the conditional variance $Var\{\phi(Y)|X = x\}$ (for simplicity the model is supposed to be homoscedastic), we have

$$\mathbb{B}(x) = B_{p,K}^{(v)} m_\phi^{(p+1)}(x) \qquad \mathbb{V}(x) = \frac{V_{p,K}^{(v)}\sigma_\phi^2}{f_X(x)}. \tag{5}$$

The coefficients $B_{p,K}^{(v)}$ and $V_{p,K}^{(v)}$ are known, and depend on the parameters K, v and p. Table 1 reports the values for the most used kernel function and for different values of p and v (see Fan & Gijbels (1996) for the formulas). Note that only the cases where $p - v$ is odd are reported. Taking account of the *bias-variance trade-off*, the asymptotic plug-in optimal local bandwidth is derived by minimizing the $AMSE$ at the point x, giving

$$h_L^{opt}(x) = \left\{ \frac{(2v+1)\mathbb{V}(x)}{2n(p-v+1)\mathbb{B}^2(x)} \right\}^{1/(2p+3)} \qquad \forall x \in \mathbb{R}. \tag{6}$$

Note that the (6) is a local bandwidth function, which performs a smoothing of the regression function which varies with the support of the estimation. We use the notation h_G^{opt} to denote a constant bandwidth that minimises some global measure of the estimation error, as the integrated asymptotic mean square error ($AMISE$). We expect local bandwidths to perform better than global bandwidths. In any case, the estimator of $h_L^{opt}(x)$ is expected to be less efficient than the estimator of h_G^{opt} (note that $h_L^{opt}(x)$ is a function while h_G^{opt} is a value). So, the question is how to evaluate whether there is an effective gain from using the *estimated* local bandwidth instead of the *estimated* global bandwidth.

In this paper we propose an hybrid method that aims to combine the advantages of local (adaptability) and global (efficiency) smoothing. We call the method Global Adaptive Smoothing (GAS). Our idea is based on the use of a *rolling window* procedure. Given a point $x \in \chi$, where χ is a compact set of $\mathbb{R}$ (see the appendix for more details), the optimal bandwidth for that point is derived by estimating a global bandwidth on the interval centred on x, $I_x = [x - a/2, x + a/2]$, for a given $a > 0$. So, the interval I_x must be of positive length and it must contain at least one observed point. By moving the interval I_x, we derive an estimation of the optimal local bandwidth for other points x on the support of the estimation. Define the $AMISE$ on I_x as

$$\int_{I_x} AMSE\{\hat{m}_\phi(u; h)\} f_X(u) du. \tag{7}$$

The optimal bandwidth on I_x can be derived by minimizing the (7). Denote such bandwidth with h_{I_x}. It is equal to

$$h_{I_x} = \left\{ \frac{(2v+1)\mathbb{V}_{I_x}}{2n(p-v+1)\mathbb{B}_{I_x}} \right\}^{1/(2p+3)} \tag{8}$$

where

$$\mathbb{B}_{I_x} = (B_{p,K}^{(v)})^2 \int_{x-a/2}^{x+a/2} [m_\phi^{(p+1)}(u)]^2 f_X(u) du, \qquad \mathbb{V}_{I_x} = V_{p,K}^{(v)} \sigma_\phi^2 a. \tag{9}$$

We need to estimate the functionals in (9) and to plug them into the (8) in order to have the estimated bandwidth $\hat{h}_{I_x}$. Then the final regression estimator of $m_\phi(x)$ is $\hat{m}_\phi(x; \hat{h}_{I_x})$. In section 3, we propose to estimate the functionals σ_ϕ^2 and $m_\phi^{(p+1)}$ using the neural networks technique, generalizing a procedure in Giordano & Parrella (2008). Actually, other nonparametric estimators could be used for the estimation of these functionals, such as splines or local polynomials (as traditionally done in the other plug-in methods). Each one of these alternatives implies, of course, the necessity of setting some pilot parameters (such as the number of knots or the pilot bandwidths). However, the classic plug-in methods for bandwidth selection are known to be crucially dependent on the correct identification of such pilot parameters. Instead, here we choose the neural network because it is a global approximator and it allows our smoothing estimator to be stable against the misspecification of the pilot parameter (which in our case is given by the number of nodes d). This is shown in the simulation study.

Note that we do not need to estimate the density $f_X(\cdot)$, since it is implicitly estimated by integration (as shown in section 3).

A tuning parameter, a, which determines the width of the window around x, is introduced to tune between the global and local bandwidths, since

$$\begin{array}{ccc} \infty \longleftarrow & a & \longrightarrow 0 \\ & \Downarrow & \\ h_G^{opt} \longleftarrow & h_{I_x} & \longrightarrow h_L^{opt}(x). \end{array} \tag{10}$$

For a fixed point x and a finite sample size n, the bandwidth h_{I_x} reaches the optimal bandwidths $h_L^{opt}(x)$ and h_G^{opt} in limit, while for $0 < a < \infty$ it assumes some intermediate value which is suboptimal from a theoretical point of view, but which can be the best choice when estimating h_{I_x}. Note that the rolling window procedure automatically performs a "smoothing" of the estimated h_{I_x}, given that $a > 0$. Therefore, it is not necessary to post-smooth the estimated bandwidth function in a second stage, as with other plug-in local bandwidth selectors (see, for example, the papers of Fan & Gijbels (1995); Prewitt & Lohr (2006); Gluhovsky & Gluhovsky (2007)). In section 4 we suggest a possible strategy for the automatic selection of the parameter a.

3. The neural network GAS algorithm

In this section we present a procedure for estimating the unknown functionals in (9). For simplicity, we consider the problem of estimating the conditional mean function $m(\cdot)$. So, we start from the following model

$$Y_i = m(X_i) + \varepsilon_i, \qquad i = 1, 2, \ldots, n \tag{11}$$

where the ε_i are $i.i.d$ with mean zero and variance σ^2_ε. We consider one point of estimation $x \in \chi$, around which we define the interval I_x. The bandwidth function is estimated by implementing the rolling procedure described in section 2. We consider a nonparametric estimator $q(x; \hat{\boldsymbol{\eta}})$, where $\hat{\boldsymbol{\eta}}$ is given by

$$\hat{\boldsymbol{\eta}} = \arg\min_{\boldsymbol{\eta}} \sum_{i=1}^{n} \left[Y_i - q_d(X_i; \boldsymbol{\eta})\right]^2, \tag{12}$$

and $q_d(u; \boldsymbol{\eta})$ is a *Feedforward Neural Network* (FNN), with one input layer and one hidden layer. It is defined as

$$q_d(u; \boldsymbol{\eta}) = \sum_{k=1}^{d} c_k \Gamma\left(a_k u + b_k\right) + c_0, \tag{13}$$

where $\boldsymbol{\eta} = (c_0, c_1, \ldots, c_d, a_1, \ldots, a_d, b_1 \ldots b_d)$ is the vector of the parameters of the FNN to be estimated, d is the number of nodes in the hidden layer and $\Gamma(\cdot)$ is the *activation function*. We consider the class of feedforwad Neural Networks with a sigmoidal activation function, i.e. a measurable function $\Gamma(\cdot)$ on $\mathbb{R}$ such that $\Gamma(x) \to 1$ when $x \to \infty$ and $\Gamma(x) \to 0$ when $x \to -\infty$.

The parameter d acts as a tuning parameter for the neural networks function, and must be identified by means of some optimality criteria. So, it has a role similar to the pilot bandwidths of the classic plug-in bandwidth selection procedures. Anyway, the effects of this parameter on the bandwidth estimator are rather different, as will be evidenced in the simulation study. Moreover, the selection of such tuning parameter is simpler than the selection of a pilot bandwidth, given that the first is a positive integer number (generally some units), whereas the second is a positive real number. Another important difference with pilot bandwidths is that the parameter d for the estimation of the derivative function is the same used for the estimation of the function itself (while the pilot bandwidths used for the estimation of the derivative function must be of different order than the bandwidths used for the estimation of the function).

We define the residuals as $\hat{\varepsilon}_i = Y_i - q(X_i; \hat{\boldsymbol{\eta}})$ and $\tilde{\varepsilon}_i = \hat{\varepsilon}_i - \bar{\hat{\varepsilon}}$, where $\bar{\hat{\varepsilon}} = \frac{1}{n}\sum_{i=1}^n \hat{\varepsilon}_i$. Thus we propose the following estimator for σ_ε^2

$$\hat{\sigma}_\varepsilon^2 = \frac{1}{n}\sum_{i=1}^n \tilde{\varepsilon}_i^2. \tag{14}$$

Next, we need to estimate the derivative function $m^{(p+1)}(x)$. We use the previous NN estimate, taking the derivative of order $p+1$ of the estimated NN function

$$\hat{m}^{(p+1)}(X_i) = q^{(p+1)}(X_i; \hat{\boldsymbol{\eta}}), \qquad i = 1, \dots, n. \tag{15}$$

Note that h_{I_x} can be written as

$$h_{I_x} = \left\{ \frac{(2v+1)\mathbb{V}_{\omega_{I_x}}}{2n(p-v+1)\mathbb{B}_{\omega_{I_x}}} \right\}^{1/(2p+3)} \tag{16}$$

where, given that $d\omega_{I_x}(u) = du/\mu^X(I_x)$,

$$\mathbb{B}_{\omega_{I_x}} = (B_{p,K}^{(v)})^2 \int_{I_x} [m^{(p+1)}(u)]^2 f_X(u) d\omega_{I_x}(u), \qquad \mathbb{V}_{\omega_{I_x}} = \frac{V_{p,K}^{(v)} \sigma_\varepsilon^2 \ a}{\mu^X(I_x)}. \tag{17}$$

In the (17), $\mu^X(I_x)$ is the measure of I_x with respect to the distribution function of X. We prefer to express the functionals in (17) with respect to the conditional distribution function of X because this provides some advantages from a computational point of view.

We can then propose the following two estimators for the functionals in (17) with respect to a set I_x of dimension a:

$$\widehat{\mathbb{B}}_{\omega_{I_x}} = \frac{(B_{p,K}^{(v)})^2 \sum_{i=1}^n \left[\hat{m}^{(p+1)}(X_i)\right]^2 \mathbb{I}(X_i \in I_x)}{\sum_{i=1}^n \mathbb{I}(X_i \in I_x)}, \quad \widehat{\mathbb{V}}_{\omega_{I_x}} = \frac{V_{p,K}^{(v)} \hat{\sigma}_\varepsilon^2 a}{n^{-1}\sum_{i=1}^n \mathbb{I}(X_i \in I_x)}. \tag{18}$$

The final bandwidth estimator, for a given value of the parameter a, is obtained by plugging the (18) into the (16)

$$\hat{h}_{I_x} = \left\{ \frac{(2v+1)\widehat{\mathbb{V}}_{\omega_{I_x}}}{2n(p-v+1)\widehat{\mathbb{B}}_{\omega_{I_x}}} \right\}^{1/(2p+3)}. \tag{19}$$

4. Setting the parameter *a*

Here we suggest a strategy for the selection of the parameter a. First, suppose that the functionals in the (4) are known. Given that

$$h_L^{opt}(x) = \arg\min_h AMSE\{\hat{m}_\phi(x; h)\} \qquad x \in \chi, \tag{20}$$

we expect asymptotically an increase in the value of the local $AMSE$ when using a bandwidth different from $h_L^{opt}(x)$. Using the (4), (5) and (6), the relative increment is given by

$$\begin{aligned}&\frac{AMSE\{\hat{m}_\phi(x;h_{I_x})\}-AMSE\{\hat{m}_\phi(x;h_L^{opt}(x))\}}{AMSE\{\hat{m}_\phi(x;h_L^{opt}(x))\}} \qquad (21)\\ =\;& \frac{2v+1}{2p+3}\left(\frac{h_{I_x}}{h_L^{opt}(x)}\right)^{2(p+1-v)}+\frac{2(p+1-v)}{2p+3}\left(\frac{h_{I_x}}{h_L^{opt}(x)}\right)^{-(2v+1)}-1.\end{aligned}$$

So, for fixed v and p, it is directly connected with the ratio $h_{I_x}/h_L^{opt}(x)$. The minimum is reached, as expected, at $h_{I_x}/h_L^{opt}(x)=1$, which leads to the following condition

$$\frac{\mathbb{B}_{I_x}}{\mathbb{V}_{I_x}}=\frac{\mathbb{B}^2(x)}{\mathbb{V}(x)}, \qquad \forall x\in\chi. \tag{22}$$

There are two opposite cases for condition (22). The first one is when the function $\mathbb{B}^2(x)/\mathbb{V}(x)$ is constant over χ, which means that $h_L^{opt}(x)\equiv h_G^{opt}$, on χ. In such a case, we have $h_{I_x}=h_G^{opt},\forall x\in\chi$ and $\forall a>0$. The second is when the ratio $\mathbb{B}^2(x)/\mathbb{V}(x)$ is not constant. In this case, given the (10) and the (20), the only solution for the minimization of the (21) is $a=0$, *i.e.* the local bandwidth. Anyway, we may argue that in many cases the ratio $\mathbb{B}^2(x)/\mathbb{V}(x)$ is not constant but rather “smooth”, such that it can be seen as substantially constant over subintervals. In this case, there can exist a solution for the (22) for some $a>0$. Moreover, note that the (22) is actually a pointwise condition, suggesting a pointwise optimal value, a_x^*. But, if we are interested in the estimation of the function $m_\phi(x)$ on the whole support χ, we may prefer to identify a constant value of a which can be considered globally optimal, though locally suboptimal. In such a way, we maintain the procedure computationally simple. This is the reason why we introduce the following global condition

$$\int_\chi\frac{\mathbb{B}_{I_x}}{\mathbb{V}_{I_x}}dx=\int_\chi\frac{\mathbb{B}^2(x)}{\mathbb{V}(x)}dx, \tag{23}$$

which is not equivalent to (22) but which can be used to estimate a global parameter a. This is only one possible criterion to choose the parameter a. Our idea is to balance, in mean over the support χ, $\mathbb{B}^2(x)/\mathbb{V}(x)$ (related to the local smoothing) and $\mathbb{B}_{I_x}/\mathbb{V}_{I_x}$ (related to the global adaptive smoothing). In this way, we find a global parameter a which is able to capture the trade-off between local and global smoothing.

Under model (11), using the (17) the condition (23) can be written as follows

$$aR=\int_\chi R_a(x)dx, \tag{24}$$

where we compare the functionals $R_a(x)=\int_{x-a/2}^{x+a/2}[m^{(p+1)}(v)]^2f_X(v)dv/\mu^X(I_x)$ and $R=\int_\chi[m^{(p+1)}(v)]^2f_X(v)dv/\mu^X(\chi)$. From Proposition 1, we have

$$\hat{R} \;=\; \frac{\sum_{i=1}^n\left[\hat{m}^{(p+1)}(X_i)\right]^2\mathbb{I}(X_i\in\chi)}{\sum_{i=1}^n\mathbb{I}(X_i\in\chi)}$$

$$\hat{R}_a(x) \quad = \quad \frac{\sum_{i=1}^{n} \left[\hat{m}^{(p+1)}(X_i)\right]^2 \mathbb{I}(X_i \in I_x)}{\sum_{i=1}^{n} \mathbb{I}(X_i \in I_x)}$$

as consistent estimators for R and $R_a(x)$ in equation (24), for some given values of a and x. Let n_x be the number of estimation points in χ. So $\{x_1, x_2, \ldots, x_{n_x}\} \subset \chi$. Then, we can write $\frac{1}{n_x}\sum_{j=1}^{n_x} \hat{R}_a(x_j)$ in place of $\int_\chi R_a(x)dx$. The estimated $\hat{a}_n$ is obtained by solving the equation

$$a\hat{R} = \frac{1}{n_x}\sum_{j=1}^{n_x} \hat{R}_a(x_j). \tag{25}$$

It can be easily shown that the (24) admits always a solution $a > 0$ and that the estimator $\hat{a}$ in the (25) is consistent in probability.

Overall, both the tuning parameters d and a are essential but they have different roles in our smoothing procedure. The parameter a represents the trade-off between local and global smoothing while the parameter d works in the so called stage of *pre-smoothing*, so it has a role similar to the well known *pilot bandwidths* typically used in the classic plug-in procedures.

5. Computational considerations on the NN-GAS algorithm

The estimation of the parameter a does not increase appreciably the computational burden of the bandwidth selecting procedure. To explain why it is so, we present schematically the steps of the GAS algorithm.

- *Step 1*: using the (12), we estimate the neural networks function $q(X_i; \hat{\boldsymbol{\eta}})$, for $i = 1, \ldots, n$. To this end, a BIC procedure can be used to derive the optimal value of d.
- *Step 2*: by the (14) and (15), using $q(X_i; \hat{\boldsymbol{\eta}})$, we derive the NN estimator of the variance, $\hat{\sigma}_\varepsilon^2$, and the NN estimator of the derivative, $\hat{m}^{(p+1)}(X_i), i = 1, \ldots, n$.
- *Step 3*: given the points $x_j \in \chi, j = 1, \ldots, n_x$, we estimate the global parameter, $\hat{a}_n$, by solving equation (25).
- *Step 4*: plug-in the estimated $\widehat{\mathbb{B}}_{\omega_{I_x}}$ and $\widehat{\mathbb{V}}_{\omega_{I_x}}$ into the (16), to derive $\hat{h}_{I_x}$, for the different x_j, $j = 1, \ldots, n_x$, using a estimated in step 3.
- *Output*: the final local polynomial estimator $\hat{m}_\phi(x_j; \hat{h}_{I_x}), j = 1, \ldots, n_x$.

Note that only step 1 is computationally intensive, since it implies a nonlinear minimization in order to estimate the neural networks function. The other steps are very fast, given that they are based on simple calculations of such estimated values, or simple averages of the estimated values falling in each interval I_x.

Note that the NN estimations in steps 1 and 2 are global estimations, *i.e.* they do not depend on the points of estimation x. This means that they must not be repeated for different values of x. On the other side, in steps 3–4 the NN global estimations are transformed, through the (18), into "local" estimations. Finally, only step 4 must be repeated for each desired point of estimation $x \in \chi$. Anyway, this takes a little more computing time, as will be shown now.

The algorithm for the estimation of a in step 3 is based on successive splittings of the interval in order to derive the solution of the (25).

To give an idea of the computing time, consider, for example, a sample of size $n = 500$ and $n_x = 50$ points of estimation $x_j \in \chi$. Let us consider three cases: in the first, we point to estimate the pure global bandwidth (in such case, step 3 is skipped and step 4 must be adapted for the global bandwidth estimator); in the second case, we point to estimate the variable bandwidth h_{I_x} for a fixed parameter a (in this case, we only skip step 3); in the last case, we perform the full algorithm. We run the algorithm 100 times on a dual core with Intel Pentium 2.00 GHz. In the first case (pure global bandwidth), the average computing time is 8.03 seconds. In the second case, the time is 8.16 seconds. So, the step 4 for the 50 points takes only 0.14 seconds. Finally, in the last case of local bandwidth, the average time to run the whole algorithm is 10.50 seconds. These computing times are satisfactory for a *local* bandwidth selection procedure. For example, the average time for one iteration of the Prewitt-Lohr method is 12.67 seconds. Moreover, as pointed out in subsection 7.1, the BIC criterion is consistent for selecting the parameter d, but it is not very sensible in case of oversmoothing (i.e., the bandwidth estimates are substantially stable for $d \geq 3$). So, if we fix $d = 4$ we avoid the BIC selection step and we have only 4.61 seconds for our procedure.

6. Theoretical results for the rate of convergence

In this section we investigate the rate of convergence for our GAS procedure, and compare it with the rates of convergence of the most popular global bandwidth selectors proposed in the literature.

When analysing the rate of convergence of a global bandwidth selector, two aspects need to be considered: the first involves the closeness of the $AMISE$ approximation to the $MISE$ (being the AMISE the leading term of the MISE expansion); the second concerns the quality of the functional estimation. In particular, concerning the first aspect, using the same arguments as in section 2 of Ruppert *et al.* (1995), we can write

$$MISE\{\hat{m}_\phi(x;h)|X_1,..,X_n\} = AMISE\{\hat{m}_\phi(x;h)\} + o_p\left\{h^{2(p+1-v)} + \frac{1}{nh^{2v+1}}\right\}, \tag{26}$$

so the bandwith $h_G^{opt} = \arg\min_h AMISE\{\hat{m}_\phi(x;h)\}$ represents the first order approximation of the true optimal bandwidth minimizing the $MISE$, denoted with h^{MISE}.

Suppose for simplicity that $v = 0$ and $p = 1$, so that $p - v$ is odd. The authors Fan & Huang (1999) note that it is

$$\frac{h_G^{opt} - h^{MISE}}{h^{MISE}} = O_p(n^{-2/5}), \tag{27}$$

showing an upper bound rate of convergence for *any* plug-in bandwidth estimator based on the estimation of h_G^{opt}. They show also that it is useless to consider further terms in the asymptotic expansion of the $MISE$ to improve the rate of convergence. The only way to change this rate is to modify the kernel estimator in order to decrease the order of its bias and/or variance, based, for example, on the proposal in He & Huang (2009). Although our procedure can be generalised to those setups, for simplicity we do not consider this point further.

The second aspect, concerning the quality of the functional estimation, is of more interest to us, because it highlights the real differences among the bandwidth estimators. Ruppert *et al.* (1995) show that the best rate of convergence achieved by their bandwidth selectors is $O_p(n^{-2/7})$ when using a local cubic polynomial fit to estimate the derivative function $m_\phi^{(2)}$. Fan & Huang (1999) obtain a rate of the order $O_p(n^{-2/5})$ for the pre-asymptotic substitution method proposed by Fan & Gijbels (1995), which reaches the upper bound in the (27), but they consider a local polynomial fit of order $p = 5$ to estimate the derivative function $m_\phi^{(2)}$, which has several implications. First of all, an increase in the variability of the estimator; second, the need for more data to avoid invertibility problems; finally, the need of more stringent assumptions on the model, in terms of the existence of higher order derivatives. Moreover, although the results in Ruppert *et al.* (1995) and Fan & Huang (1999) are a big improvement on the cross-validation procedures – which are $O_p(n^{-1/10})$, as shown in Härdle *et al.* (1988) – they are handicapped by the need to assume an "optimal" order for the pilot bandwidths used in the polinomial fit of the derivative $m_\phi^{(2)}$, which overlooks that these pilot bandwidths also need to be estimated. When the pilot bandwidths are estimated, the final rate of convergence of the bandwidth selector becomes worse.

Our global smoothing method achieves the best rate of convergence without requiring any recursive bandwidth estimations. Let

$$h_G^{opt} = \left\{ \frac{\mathbb{V}_G}{4n\mathbb{B}_G} \right\}^{1/5} \tag{28}$$

where

$$\mathbb{B}_G = (B_{1,K}^{(0)})^2 \int_\chi [m^{(2)}(u)]^2 f_X(u) du, \qquad \mathbb{V}_G = V_{1,K}^{(0)} \sigma_\epsilon^2 |\chi|, \tag{29}$$

and $|\chi|$ is the length of the closed and bounded interval χ. Moreover, suppose that $m^{(2)}(\cdot) \not\equiv 0$ for each I_x. If we consider I_x we can write $MISE_{I_x}$ as in (26). Note that $AMISE_{I_x}$ is defined in (7). In this case, the bandwidth which minimizes $AMISE_{I_x}$ is h_{I_x} (see (8)). Let $\hat{h}_G^{opt}$ be the estimator of the (28). We can state the following two results.

Theorem 1. *Under assumptions (a1)–(a6) in the appendix, the bandwidth estimator, $\hat{h}_{I_x}$, has the following uniform rate of convergence to the true bandwidth $h^{MISE_{I_x}}$ w.r.t. a*

$$\sup_{a>0} \left\{ \frac{\hat{h}_{I_x} - h^{MISE_{I_x}}}{h^{MISE_{I_x}}} \right\} = O_p\left(n^{-2/5} \log n \right).$$

Corollary 1. *Under assumptions (a1)–(a6) in the appendix, the global bandwidth estimator, $\hat{h}_G^{opt}$, has the following rate of convergence to the true bandwidth h^{MISE}*

$$\frac{\hat{h}_G^{opt} - h^{MISE}}{h^{MISE}} = O_p\left(n^{-2/5}\right).$$

Remark 1. In Theorem 1 and Corollary 1, we consider a sequence for d given in Assumption (a6). However, we can use a selection criterion as BIC about d. The penalty term in the BIC criterion is $n_d/n(\log n)$, where n_d indicates the number of parameters to estimate in a Feedforward Neural Networks with d hidden layer neurons. Using assumptions (a1)–(a6) in the appendix, we have

$$\frac{1}{n}\sum_{i=1}^{n}\left[Y_i - q_{\hat{d}}(X_i;\hat{\boldsymbol{\eta}})\right]^2 = O_p\left(\left(\frac{\log n}{n}\right)^{1/2}\right),$$

where

$$(\hat{\boldsymbol{\eta}}, \hat{d}) = \arg\min_{\boldsymbol{\eta}, d} \frac{1}{n}\sum_{i=1}^{n}\left[Y_i - q_d(X_i;\boldsymbol{\eta})\right]^2 + \frac{n_d(\log n)}{n}.$$

In order to prove this result, it is sufficient to use together Theorem 4 in Barron (1994), Lemma 1 and the same arguments as in the proof of Lemma 2.

7. The NN-GAS procedure at work

7.1. Results from a simulation study

In this section we report the results of a Monte Carlo experiment aimed at assessing the numerical performance of the NN-GAS procedure. Since our procedure can be used to perform both global and local smoothing, we compare it with the most widely used procedures for global and local bandwidth selection. In relation to global smoothing, we compare our procedure with Ruppert *et al.* (1995) plug-in bandwidth selection algorithm (using their package implemented in the R environment) and with the Cross-Validation method. Here, we denote such bandwidth selection algorithms as RW and CV, respectively. Note that the CV method uses a prediction error measure, contrary to plug-in methods which use MISE based measures. In such a way, we give an idea of the performance basing on different criteria. In relation to local smoothing, our benchmark is the method proposed by Prewitt & Lohr (2006), who suggest a local bandwidth selector based on the eigenvalue approach. We denote this bandwidth selection algorithm by PL. We reply here the same simulation study of Prewitt & Lohr (2006). In such way, we compare indirectly our NN-GAS procedure also with their competitors Fan & Gijbels (1995); Ruppert *et al.* (1995); Choi *et al.* (2000).

We consider six different models, and we want to estimate the conditional mean function $m(x)$. Table 2 reports the details. The first four models were considered by Fan & Gijbels (1995) and Prewitt & Lohr (2006). Models 5 and 6 were used by Ruppert *et al.* (1995) and Prewitt & Lohr (2006). We consider different values of n and σ_ε and we generate 500 replications for each setting

TABLE 2

Models used in the simulation study

Model	$m(x)$	$f_X(x)$
1	$\sin(2x) + 2\exp(-16x^2)$	$0.4N(-1, 0.6^2) + 0.6N(1, 1)$
2	”	$0.5N(-1.1, 0.8^2) + 0.5N(1.1, 0.8^2)$
3	”	$0.2N(-1.1, 0.8^2) + 0.8N(1.1, 0.8^2)$
4	”	$N(0, 1)$
5	$1 - 48x + 218x^2 - 315x^3 + 145x^4$	$U(0, 1)$
6	$\sin(5\pi x)$	$U(0, 1)$

TABLE 3

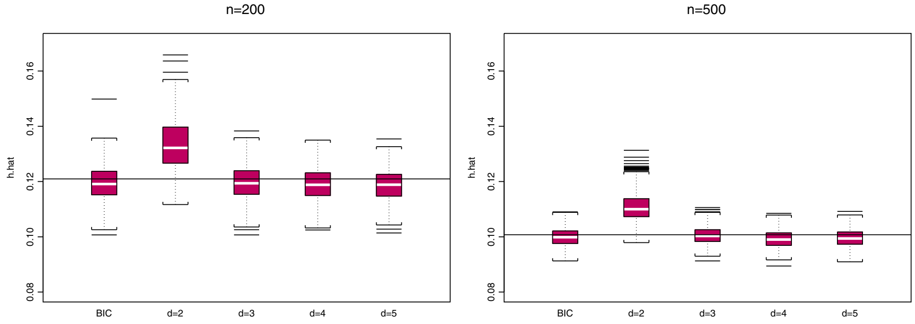
Median, mean and standard deviation of the Integrated Square Error (respectively MedISE, MISE and SDISE), observed when estimating the function $m(x)$ for the 500 replications of models 1–6, using the NN-GAS method for global smoothing. On the right, we report the mean (m) and the standard deviation (s) of the estimated tuning parameters: the number of nodes d and the global bandwidth h_G^{opt}

Model			*NN-GAS estimations*			$\hat{d}$		$\hat{h}_G^{opt}$	
#	σ_ε	n	*MedISE*	*MISE*	*SDISE*	$m_{\hat{d}}$	$s_{\hat{d}}$	$m_{\hat{h}_G}$	$s_{\hat{h}_G}$
1 (a)	0.3	200	0.008	0.026	0.361	3.1	0.3	0.118	0.006
(b)	0.3	500	0.003	0.004	0.001	3.3	0.5	0.100	0.003
(c)	0.7	200	0.030	0.033	0.015	2.3	0.5	0.167	0.019
(d)	0.7	500	0.014	0.015	0.006	3.0	0.3	0.137	0.009
2 (a)	0.3	200	0.007	0.010	0.020	2.8	0.5	0.113	0.005
(b)	0.3	500	0.003	0.003	0.001	3.2	0.5	0.094	0.003
(c)	0.7	200	0.029	0.032	0.019	2.1	0.3	0.155	0.015
(d)	0.7	500	0.012	0.013	0.006	2.3	0.5	0.130	0.008
3 (a)	0.3	200	0.009	0.009	0.006	3.1	0.3	0.129	0.008
(b)	0.3	500	0.004	0.004	0.001	3.2	0.5	0.108	0.004
(c)	0.7	200	0.030	0.035	0.050	2.6	0.5	0.183	0.033
(d)	0.7	500	0.014	0.015	0.006	3.0	0.2	0.148	0.012
4 (a)	0.3	200	0.012	0.020	0.085	4.1	0.6	0.142	0.013
(b)	0.3	500	0.005	0.005	0.002	4.5	0.6	0.117	0.005
(c)	0.7	200	0.039	0.043	0.020	2.3	0.6	0.221	0.034
(d)	0.7	500	0.018	0.019	0.007	3.6	0.8	0.168	0.018
5 (a)	—	200	0.037	0.041	0.020	2.2	0.4	0.093	0.008
(b)	—	500	0.018	0.019	0.008	2.6	0.5	0.073	0.006
6 (a)	—	200	0.017	0.018	0.007	3.8	0.5	0.056	0.003
(b)	—	500	0.008	0.008	0.003	4.2	0.4	0.047	0.001

configuration. For models 5 and 6, we fix the variance of the error σ_ε^2 as in Ruppert *et al.* (1995). The number of points in which the regression estimation is performed is fixed to $n_x = 50$ (chosen uniformly on the support χ).

We use the Epanechnikov kernel function for the local polynomial estimations and the logistic activation function for the neural network estimations. The number of nodes in the hidden layer, d, is selected following a BIC optimization procedure. Finally, all the estimations are derived without implementing bias correction, in order to take the comparison with the competitors fair.

Table 3 assesses the performance of our method using global smoothing, which means that we consider the pure global bandwidth estimator, that is constant on the support of estimation. We want to stress here that the procedure computes once for all the estimated global bandwidth $\hat{h}_G^{opt}$, and then it is used to smooth the function $m(x)$ for each point of estimation x. For the RW method,



(a) (b)

Fig 2. *Boxplots of the NN-GAS estimations of the global bandwidth* h_G^{opt}*, obtained for different values of d (=number of nodes in the hidden layer of the NN function). The solid lines represent the true global bandwidth. Plot in (a) considers model 1 and* $n = 200$*. Plot in (b) refers to model 1 and* $n = 500$*. In both cases, the first boxplot on the left summarizes the results obtained when estimating d through the BIC.*

we use the package implemented by Ruppert *et al.* (1995) in the R environment, with all the pilot estimations set automatically. We report the median, mean and standard deviations of the integrated squared error (denoted with $MedISE$, $MISE$, and $SDISE$, respectively), calculated over the 500 replications of models 1–6 (note that here the integrated squared error is taken as the sum of the LP estimates obtained over the $n_x = 50$ estimation points). On the right of the table, we also report some summary statistics giving evidence of the influence of the parameter d (= number of nodes in the hidden layer of the NN function) on the bandwidth estimations. In particular, we report the mean value and the standard deviation of $\hat{d}$, denoted by $m_{\hat{d}}$ and $s_{\hat{d}}$, respectively. The results show that the variance of $\hat{d}$ is low, while its mean value is not greater than 5. For completeness, we also report the mean and the standard deviation of the estimated global bandwidth ($m_{\hat{h}}$ and $s_{\hat{h}}$, respectively).

In order to give more evidence of the sensitivity of the estimations from the pilot parameter d, figure 2 shows the boxplots of the NN-GAS estimations of the global bandwidth h_G^{opt}, obtained for model 1 and for fixed values of d (ranging from 2 to 5). The solid lines represent the true global bandwidth. Plot in (a) considers $n = 200$, while plot in (b) refers to $n = 500$. In both cases, the first boxplot on the left summarizes the results obtained when estimating d through the BIC algorithm. Remember from table 3 that the mean value of $\hat{d}$, for such model, is approximately 3, as desired. Note also from the boxplots of figure 2 that undersmoothing the NN function (= setting $d < 3$) may cause some problems in the bandwidth estimations, while oversmoothing (= setting $d > 3$) is not so crucial. The situation is similar for the other models.

Table 4 compares the NN-GAS method with other smoothing methods: 1) the oracle smoothing estimator based on the true asymptotic global bandwidth h_G^{opt}; 2) the smoothing estimator based on the RW estimated global bandwidth; 3) the

TABLE 4

Relative increments in the median, mean and standard deviation of ISEs, with respect to the results of the NN-GAS method reported in table 3, observed when estimating the function $m(x)$ through global smoothing with different bandwidths: the true optimal global bandwidth h_G^{opt}, the RW's estimated global bandwidth and the CV's estimated global bandwidth. Note that these values are calculated using the formula in (30), and they must be multiplied by 100 to give the percentage variations. The asterisked cases were omitted because the RW method produces few results, due to invertibility problems

	True global bandwidth			*RW global bandwidth*			*CV global bandwidth*		
#	$\overset{MedISE}{\Delta}$	$\overset{MISE}{\Delta}$	$\overset{SDISE}{\Delta}$	$\overset{MedISE}{\Delta}$	$\overset{MISE}{\Delta}$	$\overset{SDISE}{\Delta}$	$\overset{MedISE}{\Delta}$	$\overset{MISE}{\Delta}$	$\overset{SDISE}{\Delta}$
1a	-0.03	-0.02	-0.01	+1.16	+4.01	+1.79	+0.03	+0.20	+1.55
1b	-0.01	-0.01	0.00	+0.85	+1.49	+16.70	+0.12	+0.16	+0.94
1c	-0.06	-0.05	0.00	***	***	***	0.06	+0.69	+23.99
1d	-0.03	-0.03	-0.01	+0.79	+0.83	+1.86	+0.08	+0.12	+0.52
2a	0.00	+0.01	+0.05	+1.21	+5.98	+14.82	+0.12	+0.07	-0.48
2b	-0.01	-0.01	0.00	+0.96	+1.32	+15.42	+0.09	+0.14	+0.53
2c	-0.04	-0.21	-0.90	***	***	***	+0.07	+0.07	-0.40
2d	-0.04	-0.03	-0.01	+0.80	+0.80	+0.78	+0.08	+0.12	+0.48
3a	-0.02	-0.05	-0.28	***	***	***	0.11	+0.52	+7.52
3b	-0.03	-0.01	0.00	+0.96	+1.43	+15.83	+0.11	+0.20	+0.55
3c	-0.09	-0.08	-0.05	***	***	***	0.09	+0.20	+0.91
3d	-0.06	-0.05	-0.03	+0.99	+1.08	+2.39	+0.10	+0.17	+0.45
4a	0.00	+1.25	+5.37	+0.74	+4.52	+5.35	-0.01	+0.40	+0.66
4b	-0.02	-0.01	+0.01	+0.89	+1.33	+10.59	+0.06	+0.15	+3.14
4c	+0.01	-0.01	-0.47	***	***	***	0.04	+0.05	+0.02
4d	-0.02	-0.02	0.00	+0.57	+0.67	+1.64	+0.08	+0.09	+0.29
5a	+0.03	0.00	-0.02	+1.03	+0.97	+0.53	+0.17	+0.20	+0.91
5b	-0.07	-0.07	-0.13	+0.77	+0.73	+0.28	+0.07	+0.12	+0.31
6a	-0.07	-0.05	-0.09	+0.97	+1.36	+14.85	+0.09	+0.10	+0.18
6b	+0.01	-0.01	+0.08	+0.81	+0.75	+0.32	+0.08	+0.10	+0.31

smoothing estimator based on the CV estimated global bandwidth. All the results are reported as relative increments with respect to the results reported in table 3, in particular we use the formula

$$\overset{MedISE}{\Delta} = \frac{MedISE(\text{competitor}) - MedISE(\text{NN-GAS})}{MedISE(\text{NN-GAS})}, \tag{30}$$

and similarly for the other two indicators $MISE$ and $SDISE$. So, a positive value of the (30) shows a better performance of the NN-GAS method compared with the competitor, and the value of $\overset{MedISE}{\Delta}$ multiplied by 100 gives the percentage increment in the $MedISE$. For example, looking at the first line of the table, we note that the oracle smoothing estimator has a $MedISE$ which is the 3% lower than the MedISE of the NN-GAS method presented in table 3. This represents the loss due to the estimation of the asymptotic optimal global bandwidth. The positive values which appear in the column of the oracle smoothing estimator are more interesting to comment. They show that, in some cases, the NN-GAS procedure performs even better than the oracle smoothing procedure: this is due to the use of the asymptotic expression of the optimal bandwidth, which for small samples can be distant from the true optimal bandwidth (in fact, note that all positive and negative values converge to zero for increasing n). It

TABLE 5

Relative increments in the mean and standard deviation for the Prediction Error, with respect to the results of the NN-GAS method, observed when estimating the function $m(x)$ through global smoothing with the CV's estimated global bandwidth. The values must be multiplied by 100 to give the percentage variations

Model			Mean	SD
	σ_ε	n		
3	0.7	200	+0.0098	+0.0618
	0.7	500	+0.0198	+0.0456
5	—	200	+0.1654	+0.0410
	—	500	+0.0624	+0.0086
6	—	200	+0.1377	+0.0847
	—	500	+0.0818	+0.0330

is interesting to note that, in general, the relative variations in the column of the oracle smoothing estimator are not very high, contrary to what happens in the other two columns relative to RW and CV methods. The NN-GAS method clearly outperforms the RW and the CV methods, especially in terms of variability. The asterisks hide some values that, though definitely favoring the NN-GAS method, are not reliable because they are based on only a few results for the RW. The reason for this is that the pilot estimations in the RW method are based on the estimation of higher order derivatives through LPE, using $p = 3$, and this implies some problems of invertibility for small samples (as said in the introduction, this is one of the drawbacks of the classic plug-in methods which is solved with our method, since the NN-GAS procedure uses the same pilot estimation of the function $m(x)$ in order to derive the estimate of the derivative $m^{(p+1)}(x)$).

Finally, we evaluate the performance of our estimator using a different measure, the prediction error. We compare our method with the CV method, reporting the relative increment in the same style as in Table 4 (of course, we have corrected such an indicator by the variance of the error, given that the prediction error converges to such value asymptotically). Therefore, positive values in Table 5 show an advantage for our method compared with the CV. Given the computational burden of the new measure, we considered only some of the models (choosing among the most interesting ones). The performance of our bandwidth estimator, based on the new measure, can be considered satisfactory.

Table 6 shows the results for local smoothing, which means that we suppose that the bandwidth is variable on the support of the regression function. Thus, for each model we fix the value of a as the value that produces the true minimum $MISE$. Table 6 is organised as follows. The results for the neural network method are reported in absolute values, as in table 3. On the right of the table, we present some statistics on the estimated values of the tuning parameters d and a, in particular mean and standard deviation. For the parameter d, one can note that the results are nearly similar to those in table 3. This confirms that the NN-GAS local bandwidth is estimated on the bases of the same NN estimations used for the global smoothing procedure. Finally, the last column in table 6 tracks the estimated values for the tuning parameter a, which are divided by

Table 6

Median, mean and standard deviation of ISEs, observed when estimating the function $m(x)$ for the 500 replications of models 1–6, using local smoothing. On the right, we report the mean (m) and the standard deviation (s) of the estimated tuning parameters: the number of nodes d and the length of the interval a (the last is divided by the length of the interval χ, to give a relative measure of a)

Model				*NN-GAS method*			$\hat{d}$		$\hat{a}_r = \hat{a}/range(\chi)$	
#		σ_ε	n	*MedISE*	*MISE*	*SDISE*	$m_{\hat{d}}$	$s_{\hat{d}}$	$m_{\hat{a}_r}$	$s_{\hat{a}_r}$
1	(a)	0.3	200	0.006	0.007	0.004	3.1	0.3	0.4	0.03
	(b)	0.3	500	0.003	0.003	0.001	3.3	0.5	0.4	0.01
	(c)	0.7	200	0.025	0.027	0.013	2.3	0.5	0.4	0.06
	(d)	0.7	500	0.011	0.012	0.006	3.0	0.3	0.4	0.03
2	(a)	0.3	200	0.006	0.006	0.003	2.7	0.5	0.5	0.02
	(b)	0.3	500	0.003	0.003	0.001	3.2	0.5	0.5	0.01
	(c)	0.7	200	0.022	0.025	0.012	2.0	0.2	0.5	0.03
	(d)	0.7	500	0.010	0.011	0.004	2.3	0.5	0.5	0.02
3	(a)	0.3	200	0.006	0.006	0.003	3.1	0.3	0.5	0.03
	(b)	0.3	500	0.003	0.003	0.001	3.2	0.4	0.5	0.02
	(c)	0.7	200	0.028	0.031	0.017	2.6	0.5	0.5	0.03
	(d)	0.7	500	0.011	0.012	0.006	3.0	0.2	0.5	0.02
4	(a)	0.3	200	0.007	0.007	0.003	4.2	0.6	0.2	0.02
	(b)	0.3	500	0.003	0.004	0.001	4.5	0.7	0.2	0.01
	(c)	0.7	200	0.030	0.033	0.015	2.3	0.6	0.2	0.02
	(d)	0.7	500	0.014	0.016	0.007	3.5	0.8	0.2	0.01
5	(a)	—	200	0.038	0.041	0.020	2.2	0.4	0.9	0.06
	(b)	—	500	0.018	0.019	0.009	2.6	0.5	0.9	0.05
6	(a)	—	200	0.017	0.018	0.007	3.8	0.5	1.0	0.06
	(b)	—	500	0.008	0.008	0.003	4.2	0.4	1.0	0.02

the length of the interval χ, in order to have a relative measure. Such estimated values for a are obtained solving the equation (25). In this case, we use the first approach described in section 4, for two reasons. First, from a theoretical point of view, we have a faster rate of convergence, shown in Remark 1. Second, from a computational point of view, we should not worry about any empty intervals, I_x. One can note that, from model 1 to model 4, we have some benefits to consider the our approach, since the estimated relative measure $\hat{a}_r$ is rather small with respect to one. Instead, for models 5 and 6, the estimated $\hat{a}$ suggest a substantial global smoothing.

In table 7, we compare the previous results with some alternatives. First, we want to evaluate what happens when we estimate the parameter a, as reported in the last column of table 6. When we estimate the parameter a, we compare the corresponding ISE's results with those reported in table 6. To show the relative increments, we report on the left of table 7 the Δ*MedISE*, Δ*MISE* and Δ*SDISE* statistics. Note that, in general, the relative increment is definitely lower than the increment obtained when comparing the NN-GAS with the PL method (right side of table 7). Indeed, in some cases there is even a decrease (negative values), showing that there is still room for improvement to the NN-GAS results. Looking at the right of table 7, our procedure works better than PL procedure. Note that we implement the PL procedure considering the best possible configuration of settings (in particular, the results are sensitive to the choice of the bandwidth grids in the two minimization steps, so we try different

Table 7

Relative increments in the median, mean and standard deviation of ISEs, with respect to the results of the NN-GAS method reported in table 6, observed when estimating the function $m(x)$ through local smoothing with different bandwidths: the NN-GAS local bandwidth with a estimated using the (25); the true optimal local bandwidth $h_L^{opt}(x)$ and the PL's estimated local bandwidth. Note that the values reported here are calculated using the formula in (30), and they must be multiplied by 100 to give the percentage variations

	NN-GAS with $\hat{a}$			*True local bandwidth*			*PL local bandwidth*		
	MedISE	*MISE*	*SDISE*	*MedISE*	*MISE*	*SDISE*	*MedISE*	*MISE*	*SDISE*
#	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ
1a	+0.01	0.00	-0.18	+0.22	+0.13	-0.46	+0.39	+0.30	-0.23
1b	+0.04	+0.03	+0.02	+0.21	+0.22	+0.03	+0.47	+0.47	+0.29
1c	+0.06	+0.07	+0.18	-0.03	-0.04	-0.11	+0.03	+0.01	-0.08
1d	-0.08	-0.02	+0.13	+0.03	+0.05	-0.25	+0.19	+0.19	-0.03
2a	-0.03	-0.03	-0.04	-0.14	-0.14	-0.18	+0.15	+0.15	+0.08
2b	+0.02	+0.02	-0.01	-0.10	-0.09	-0.06	+0.26	+0.26	+0.12
2c	0.00	-0.01	+0.01	-0.13	-0.14	-0.18	+0.15	+0.10	+0.01
2d	-0.01	0.00	+0.04	-0.12	-0.08	+0.05	+0.17	+0.19	+0.18
3a	+0.10	+0.24	+3.32	-0.09	0.00	+1.94	+0.51	+0.46	+0.34
3b	+0.05	+0.06	+0.03	-0.08	-0.09	-0.10	+0.54	+0.55	+0.24
3c	0.00	0.00	-0.01	-0.30	-0.28	-0.23	+0.04	+0.01	-0.15
3d	0.00	0.00	0.00	-0.16	-0.16	-0.19	+0.27	+0.24	+0.05
4a	0.00	+0.02	+0.12	+0.11	+0.06	-0.15	+1.19	+1.09	+0.54
4b	-0.15	-0.12	+0.17	-0.13	-0.11	-0.15	+1.53	+1.46	+0.90
4c	+0.29	+0.30	+0.35	+0.01	-0.05	-0.30	+0.20	+0.14	-0.09
4d	+0.06	+0.09	+0.45	-0.06	-0.08	-0.34	+0.43	+0.36	-0.10
5a	-0.01	-0.01	-0.01	-0.20	-0.17	-0.1	+0.21	+0.21	+0.03
5b	-0.03	-0.03	+0.02	-0.20	-0.20	-0.17	+0.38	+0.34	+0.05
6a	+0.01	+0.02	+0.05	-0.09	-0.07	-0.09	+0.26	+0.26	+0.16
6b	0.00	0.00	0.00	-0.09	-0.09	-0.04	+0.33	+0.31	+0.14

values and use the best solution). Sometimes, the PL shows lower variability for short datasets. This is not surprising, since the variability of the PL is limited by the use of the grid of bandwidths, while the NN-GAS can generate unbounded values of bandwidths. As the size of the sample increases, the estimations become more stable, and the NN-GAS also improves over the PL in terms of varability.

Finally, the central columns in table 7 report the $\Delta MedISE$, $\Delta MISE$ and $\Delta SDISE$ statistics for the oracle local smoothing estimator, which is based on the use of the true asymptotic local bandwidth, $h_L^{opt}(x)$. The results seem quite satisfactory.

In order to compare with the results in Prewitt & Lohr (2006), we also report in table 8 the results when estimating the first derivative of the conditional mean function, denoted here by $m'(x)$. The results are comparable with those in tables 6 and 7, although, as expected, they reflect the greater difficulty to estimate the derivative function (which is a latent function).

7.2. *Application to a real dataset*

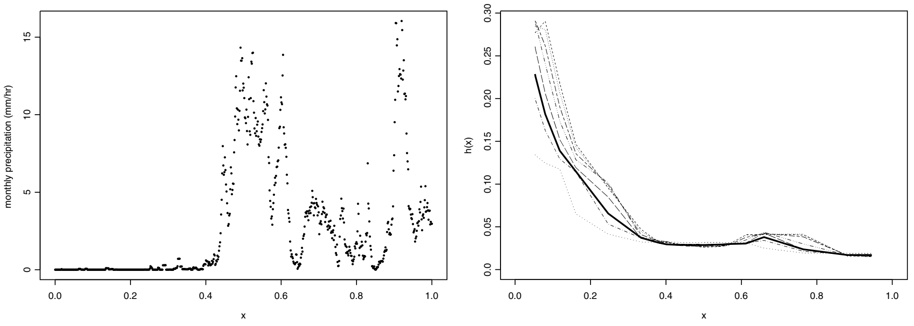
In this section we analyse a real dataset downloaded from the data archive of the NASA, at the following address:

`https://mynasadata.larc.nasa.gov/`

TABLE 8

Median, mean and standard deviation of ISEs, observed when estimating the derivative function $m'(x)$ for the 500 replications of models 1–6, when using local smoothing. In the central column, the results for the neural network method are reported in absolute value. On the right, the results for the PL method are reported as relative increments with respect to the NN-GAS method

Model			*NN-GAS method*					*PL method*		
	σ_ε	n	$m_{\hat{d}}$	$s_{\hat{d}}$	*MedISE*	*MISE*	*SDISE*	$\overset{MedISE}{\Delta}$	$\overset{MISE}{\Delta}$	$\overset{SDISE}{\Delta}$
1	0.3	200	3.1	0.4	0.364	0.386	0.166	+0.687	+0.668	+0.411
	0.3	500	3.3	0.5	0.003	0.003	0.001	+0.452	+0.452	+0.214
	0.7	200	2.3	0.5	1.184	1.355	0.791	+0.241	+0.147	-0.200
	0.7	500	2.9	0.3	0.637	0.684	0.343	+0.307	+0.253	-0.076
2	0.3	200	2.8	0.5	0.395	0.417	0.166	+0.470	+0.437	+0.306
	0.3	500	3.2	0.4	0.002	0.003	0.001	+0.266	+0.262	+0.126
	0.7	200	2.1	0.3	1.232	1.466	0.996	+0.192	+0.069	-0.375
	0.7	500	2.3	0.5	0.010	0.011	0.005	+0.196	+0.181	+0.091
3	0.3	200	3.1	0.3	0.441	0.494	0.259	+0.720	+0.681	+0.401
	0.3	500	3.3	0.5	0.003	0.003	0.001	+0.585	+0.530	+0.251
	0.7	200	2.6	0.5	1.650	2.636	3.794	-0.001	-0.333	-0.809
	0.7	500	3.0	0.2	0.011	0.012	0.005	+0.262	+0.257	+0.084
4	0.3	200	4.2	0.6	0.405	0.421	0.151	+2.824	+2.787	+2.752
	0.3	500	4.5	0.6	0.003	0.004	0.001	+1.534	+1.443	+0.814
	0.7	200	2.3	0.6	1.314	1.466	0.792	+0.700	+0.532	-0.190
	0.7	500	3.6	0.8	0.014	0.015	0.007	+0.456	+0.386	-0.050



(a) (b)

FIG 3. *Application of the NN-GAS procedure to estimate the average monthly precipitation, measured at latitude 0 and longitudes from 154.9W to 19.9E, using the dataset downloaded from the data archive of the NASA. Plot in (a) gives the observed values. Plot in (b) shows the estimated bandwidth functions for different values of the parameter a (in bold the estimated optimal one).*

The x value are the longitudes from 154.9W to 19.9E, for a total of 700 observations, while the y values give the monthly precipitation, measured in mm/hr, observed at the zero latitude in January of the year 1999. For our analysis we scaled the x values (longitudes) in the range (0–1).

Figure 3 reports the results of the analysis. Plot in (a) gives the observed values. Plot in (b) shows the estimated bandwidth functions for different values of the parameter a. We consider $n_x = 20$. It is evident how the bandwidth

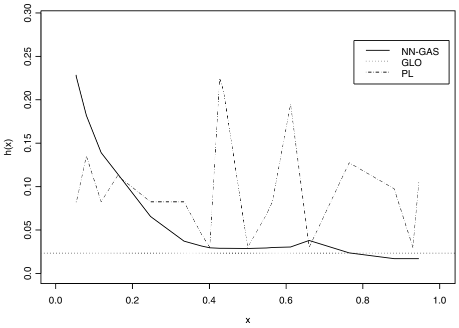


Fig 4. *Application of the NN-GAS procedure to estimate the average monthly precipitation, measured at latitude 0 and longitudes from 154.9W to 19.9E, using the dataset downloaded from the data archive of the NASA. The solid line represents the local bandwidths by our method (NN-GAS), dotted line is the constant global bandwidth using the Neural Networks and the dashed line is the local bandwidth by Prewitt-Lohr method.*

locally adapts to the peculiar structure of the dataset as long as the parameter a changes. The bold line denotes the optimal one, that is the bandwidth function derived by estimating the optimal value of the parameter a using the criterion suggested in equation 25. The estimate of a is 0.26 against an interval of length 1. Looking at Figure 3 we can argue that there is an advantage to use a local bandwidth selection procedure because the data structure seems to be very different behaviour on the x values.

Figure 4 reports the local bandwidth estimates by our method (NN-GAS) (solid line), the local bandwidth estimates by PL method (dashed line) and the global bandwidth derived using the Neural Network approach.

If we compare the mean squared residuals using the obseved values, we have that the global approach and the local one (NN-GAS) give nearly the same value. Therefore, contrary to what one could expect, by observing the scatter-plot of the data, the optimal bandwidth function is almost equivalent to the global bandwidth function, so a substantial global smoothing should be preferred to a local smoothing for the analysed dataset. All that because the estimated global bandwidth is very low as we can note in Figure 4. Instead, the same measure for PL method gives a greater value. Since we do not know the true unknown function, it is more interesting to analyze the two local bandwidth curves from NN-Gas and PL methods. In this way, it is important to outline that the estimated local bandwidth by NN-GAS is a very smooth curve with respect to the PL estimates (Figure 4). Besides, the behaviour of NN-GAS bandwidth curve respects the data points. When the data are nearly constant then we have a high value for the estimated bandwidth, otherwise the estimated bandwidth is lower. Finally, this aspect is important because a higher bandwidth implies that nh gives a greater value so we can reduce the variability of nonparametric estimator which is important for inference as in the case of confidence bands and so on. Finally, by these considerations we can suggest to use a local bandwidth approach. Moreover, our method gives an automatic trade-off between the adaptability (local bandwidth) and efficiency (global bandwidth).

7.3. Bias correction of the local polynomial estimator

Having estimated $\mathbb{V}_{\omega_{I_x}}$ and h_{I_x}, we have all the information to correct the bias of $\hat{m}_\phi(x; h_{I_x})$. In fact, given the (2), we can write

$$\begin{aligned} Bias[\hat{m}_\phi(x; h^{opt})] &\approx sign\{m_\phi^{(p+1)}(x)\}\sqrt{Var[\hat{m}_\phi(x; h^{opt})]\frac{2v+1}{2(p-v+1)}} \\ &\approx \underset{I_x}{sign}\{m_\phi^{(p+1)}(x)\}\sqrt{\frac{(2v+1)\mathbb{V}_{I_x}}{2n(p-v+1)(h_{I_x})^{2v+1}}}, \end{aligned} \tag{31}$$

where the operator $sign_{I_x}$ determines the mean of the sign operator on I_x (it can be estimated, for example, through the mean of the signs of the estimated function (15) in the interval I_x). Finally, the bias corrected local polynomial estimator is

$$\tilde{m}_\phi(x; \hat{h}_{I_x}) = \hat{m}_\phi(x; \hat{h}_{I_x}) - \widehat{Bias}[\hat{m}_\phi],$$

where $\widehat{Bias}[\hat{m}_\phi]$ is obtained from (31) substituting the unknown functionals with their estimators. This correction has been implemented in plot (b) of figure 1.

8. Further extensions: Rate of convergence for the GAS local bandwidth estimator

Note that our method can be used to estimate the optimal local bandwidth when considering a sequence of values a_n such that $a_n \to 0$ for $n \to \infty$. For the sake of completeness, in this section we derive the rate of convergence of our local bandwidth estimator. We show that it almost reaches the upper bound for a local bandwidth estimator established in Wang & Gasser (1996).

First of all, we consider the results in Fan *et al.* (1996). They show that (Theorem 2), for $x_0 \in \chi$, $v = 0$ and $p = 1$, it is

$$\frac{h^{MSE}(x_0) - h_L^{opt}(x_0)}{h_L^{opt}(x_0)} = O_p(n^{-2/5}\log n), \tag{32}$$

where the $h^{MSE}(x_0)$ is the bandwidth that minimises the true MSE at the point x_0 and $h_L^{opt}(x_0)$ is defined in (6). Hence, the ideal bandwidth $h^{MSE}(x_0)$ behaves in first order like the asymptotical optimal bandwidth $h_L^{opt}(x_0)$, and the (32) gives the upper bound for the rate of convergence of a plug-in bandwidth estimator based on the approximation of $h_L^{opt}(x_0)$ and uniformly with respect h. For the non uniform case, the bound becomes

$$\frac{h^{MSE}(x_0) - h_L^{opt}(x_0)}{h_L^{opt}(x_0)} = O_p(n^{-2/5}). \tag{33}$$

Let $\hat{h}_{I_{x_0}}$, defined in (19), be the estimator of $h_{I_{x_0}}$ in the point x_0, with $p = 1$ and $v = 0$. Moreover, suppose that $m^{(2)}(\cdot) \not\equiv 0$ in I_{x_0}. Now, given the bound in (33), the following result holds.

Theorem 2. *Under assumptions (a1)–(a4) in the appendix, with $p = 1$, if $a_n \to 0$ and $na_n^5 \to \infty$, when $n \to \infty$, and if the second derivative of $f_X(x)$ and the fourth derivative of $m(x)$ are bounded $\forall x \in I_{x_0}^n$ and $m^{(2)}(x_0) \neq 0$, then the bandwidth estimator $\hat{h}_{I_{x_0}^n}$ has the following rate of convergence to the true ideal local bandwidth $h^{MSE}(x_0)$*

$$\frac{\hat{h}_{I_{x_0}^n} - h^{MSE}(x_0)}{h^{MSE}(x_0)} = O_p\left(\left(\frac{\log na_n^5}{na_n^5}\right)^{1/2}\right) + O(a_n^2) + O_p(n^{-2/5}),$$

where $I_{x_0}^n = [x_0 - a_n/2; x_0 + a_n/2]$.

Remark 2. Theorem 2 states that $\frac{\hat{h}_{I_{x_0}^n} - h_L^{opt}(x_0)}{h_L^{opt}(x_0)} \xrightarrow{p} 0$ when $a_n \to 0$, $na_n^5 \to \infty$ as $n \to \infty$. The rate of convergence depends on a_n. But when we consider such a sequence we estimate, asymptotically, the local bandwidth in a point x_0. If we consider the sequence $a_n = O(n^{-1/9})$, we have the best rate of convergence for our local bandwidth estimator which is $O_p((\log n)^{1/2} n^{-2/9})$, by Theorem 2. This result almost reaches the upper bound stated in Wang & Gasser (1996), that is $O_p(n^{-2/9})$.

Remark 3. The proposed procedure can be generalised to the cases with multivariate predictors. First, using Barron (1993) and Chen & Shen (1998) we have already the theory to deal with the multivariate neural network estimators. Second, using the properties of the product kernel, we can refer to the well known results in the literature (see, for example, Ruppert & Wand (1994)) in order to derive the multivariate functionals for the optimal bandwidth.

Appendix: The consistency of the NN-GAS procedure

Let $\chi \subset \mathbb{R}$ be a compact set such that $\chi \subseteq \mathbb{S}_X$, where $\mathbb{S}_X$ is the support of the random variable X.

Assumptions

(a1) $E(\varepsilon_i) = 0$, $E(\varepsilon_i^4) < \infty$.
(a2) The density function of X, $f_X(\cdot)$, is positive and bounded $\forall x \in \chi$.
(a3) $X_i \sim i.i.d.$, $\varepsilon_i \sim i.i.d.$ and the X_i's are independent of the ε_i's, $\forall i$.
(a4) The sigmoidal activation function, $\Gamma(\cdot)$, has a continuous $p+1$ derivative.
(a5) The derivative $m^{(p+1)}(x)$ is continuous $\forall x \in \chi$.
(a6) $d \equiv d_n = O((\frac{n}{\log n})^{1/2})$.

Now, the compact set, χ, can be the support of the random variable X if it is compact. Instead, if the support of X is not compact we can set χ such that its measure is positive, γ, for example $\gamma = 0.95$.

Now, we state the consistency result for the neural estimators. Consider $\{Y_i^*, X_i^*\}$ the process where X_i^* are the random variables $X_i \in \chi$, while Y_i^* and ε_i^* are the corresponding random variables Y_i and ε_i, using model (11),

$i = 1, 2, \dots, n^*$, with $n^* = n\mu^X(\chi)$. Thus $n^* = O(n)$. We have the neural network estimator, $q(x; \hat{\boldsymbol{\eta}}^*)$, as in (12)

$$\hat{\boldsymbol{\eta}}_i^* = \arg\min_{\boldsymbol{\eta}} \sum_{i=1}^{n^*} \left[Y_i^* - q(X_i^*; \boldsymbol{\eta})\right]^2, \qquad i = 1, 2, \dots, n^*.$$

Lemma 1. *Under the assumptions (a1)–(a6), the neural network estimator of $m(\cdot)$ is consistent in the sense that:*

$$\int_{\chi} \left(q\left(x; \hat{\boldsymbol{\eta}}^*\right) - m(x)\right)^2 d\mu^X(x) = O_p\left(\left(\frac{\log n}{n}\right)^{1/2}\right).$$

Proof. We have to consider two cases. First, the support of X is compact. So we can fix $\chi = \mathbb{S}_X$. Second, if the support of X is not compact then we build the random variable X^* which has a compact support. Note that, in this case, $n^* = O(n)$. Moreover, using assumption (a3) we have that the function $m^{(p+1)}(\cdot)$ is square integrable on χ. Thus we can apply the results in Barron (1993). Moreover, using the Case 1.1 in Chen & Shen (1998) the result follows.

The next two Lemmas show some preliminary results which are used in Propositions 1 and 2.

Let $\hat{n}^* = n\mu_n^X(\chi)$, where $\mu_n^X(\chi) = \frac{1}{n}\sum_{i=1}^n \mathbb{I}(X_i \in \chi)$ with $\mathbb{I}(\cdot)$ the indicator function.

Lemma 2. *Under the assumptions (a1)–(a6), the estimator $\hat{\sigma}_\varepsilon^2 = \frac{1}{\hat{n}^*}\sum_{i=1}^{\hat{n}^*} \tilde{\varepsilon}_i^{2*}$ is consistent in the sense that*

$$\hat{\sigma}_\varepsilon^2 - \sigma_\varepsilon^2 = O_p\left(\left(\frac{\log n}{n}\right)^{1/2}\right).$$

Proof. By (14), we can write $\hat{\sigma}_\varepsilon^2 = \frac{1}{\hat{n}^*}\sum_{i=1}^{\hat{n}^*} \hat{\varepsilon}_i^{*2} - (\bar{\hat{\varepsilon}}^*)^2$. We must show that

$$\frac{1}{\hat{n}^*}\sum_{i=1}^{\hat{n}^*} \hat{\varepsilon}_i^{*2} - \sigma_\varepsilon^2 = O_p\left(\left(\frac{\log n}{n}\right)^{1/2}\right)$$

and

$$\left(\bar{\hat{\varepsilon}}^*\right)^2 = O_p\left(\left(\frac{\log n}{n}\right)^{1/2}\right).$$

We show only the first convergence result. The second is straightforward. Fix $\hat{n}^* = n^*$. Note that $n^* = O(n)$. Using model (11), we have that

$$\begin{aligned} \left[\frac{1}{n^*}\sum_{i=1}^{n^*} \hat{\varepsilon}_i^{*2}\right] &= \left[\frac{1}{n^*}\sum_{i=1}^{n^*} \left[m(X_i^*) - q\left(X_i^*; \hat{\boldsymbol{\eta}}^*\right) + \varepsilon_i^*\right]^2\right] = \\ &= \frac{1}{n^*}\sum_{i=1}^{n^*} \left[m(X_i^*) - q\left(X_i^*; \hat{\boldsymbol{\eta}}^*\right)\right]^2 + \frac{1}{n^*}\sum_{i=1}^{n^*} \varepsilon_i^{*2} + \end{aligned}$$

$$\begin{aligned} &+ \quad \frac{2}{n}\sum_{i=1}^{n^*}\varepsilon_i^*\left[m(X_i^*) - q\left(X_i^*;\hat{\boldsymbol{\eta}}^*\right)\right] \\ &= \quad I_1 + I_2 + I_3. \end{aligned}$$

For I_1, using the Markov's inequality, we have, for $\delta > 0$

$$P_{X^*|\hat{\boldsymbol{\eta}}^*}\left(\frac{1}{n^*}\sum_{i=1}^{n^*}[q(X_i^*;\hat{\boldsymbol{\eta}}^*) - m(X_i^*)]^2 \geq \delta\right) \leq \frac{1}{\delta}\int_\chi (q(x;\hat{\boldsymbol{\eta}}^*) - m(x))^2 d\mu^X(x).$$

But $\int_\chi (q(x;\hat{\boldsymbol{\eta}}^*) - m(x))^2 d\mu^X(x) = O_p((\frac{\log n}{n})^{1/2})$ by Lemma 1. So we can conclude that $I_1 = O_p((\frac{\log n}{n})^{1/2})$.

For I_2, by assumption (a1), it follows that $\frac{1}{n^*}\sum_{i=1}^{n^*}\varepsilon_i^{*2} - \sigma_\varepsilon^2 = O_p(n^{-1/2})$. Therefore, $I_2 - \sigma_\varepsilon^2 = O_p(n^{-1/2})$.

For I_3, we have that

$$\begin{aligned} &P_{X^*|\hat{\boldsymbol{\eta}}^*}\left(\frac{1}{n^*}\sum_{i=1}^{n^*}\varepsilon_i^*\left[q\left(X_i^*;\hat{\boldsymbol{\eta}}^*\right) - m(X_i^*)\right] \geq \delta\right) \leq \\ \leq \quad &\frac{1}{\delta^2}\frac{\sigma_\varepsilon^2}{n^*}\int_\chi \left(q\left(x;\hat{\boldsymbol{\eta}}^*\right) - m(x)\right)^2 d\mu^X(x) = O_p\left(\frac{\log^{1/2} n}{n}\right). \end{aligned}$$

All that implies

$$\frac{1}{n^*}\sum_{i=1}^{n^*}\hat{\varepsilon}_i^{*2} - \sigma_\varepsilon^2 = O_p\left(\left(\frac{\log n}{n}\right)^{1/2}\right).$$

Now we have to consider $\hat{n}^*$. By definition $\hat{n}^* = n^* + n(\mu_n^X(\chi) - \mu^X(\chi))$. But $\mu_n^X(\chi) - \mu^X(\chi) = O_p(n^{-1/2})$. Since $n^* = O(n)$, it implies that $\hat{n}^* = n(1 + O_p(n^{-1/2}))$.

Finally, the result follows. □

Lemma 3. *Under the assumptions (a1)–(a6), the following result holds*

$$\frac{1}{\hat{n}^*}\sum_{i=1}^{\hat{n}^*}\left(q^{(p+1)}\left(X_i^*;\hat{\boldsymbol{\eta}}^*\right) - m^{(p+1)}(X_i^*)\right)^2 = O_p\left(\left(\frac{\log n}{n}\right)^{1/2}\right).$$

Proof. Let $\mathcal{M}$ be the space of functions of model (11). By assumptions (a2) and (a3) we have that $\int_\chi (m^{(j)}(x))^2 d\mu^X(x) < \infty$, $\forall j \in \{0, 1, \ldots, p+1\}$ and $\forall m \in \mathcal{M}$.

Put $\hat{n}^* = n^*$. Note that $n^* = O(n)$. It is sufficient to show that

$$\int_\chi \left(q^{(j)}\left(x;\hat{\boldsymbol{\eta}}^*\right) - m^{(j)}(x)\right)^2 d\mu^X(x) = O_p\left(\left(\frac{\log n}{n}\right)^{1/2}\right)$$

$\forall j \in \{0, 1, \ldots, p+1\}$. When $j = 0$ we have Lemma 1. For $j > 0$ we can write

$$\int_{\chi} \left(q^{(j)}\left(x; \hat{\boldsymbol{\eta}}^*\right) - m^{(j)}(x) \right)^2 d\mu^X(x) \leq \left\| D^{(j)} \right\|^2 \int_{\chi} \left(q\left(x; \hat{\boldsymbol{\eta}}^*\right) - m(x) \right)^2 d\mu^X(x)$$

where $\|D^{(j)}\|^2 = \sup_{m \in \mathcal{M}} \int_{\chi} (m^{(j)}(x))^2 d\mu^X(x)$. But $\|D^{(j)}\|^2 = C < \infty$ because $D^{(j)}$ is a linear and bounded operator on $\mathcal{M}$.

Let $I_x = [x-a, x+a]$, with $a > 0$ and $\forall x \in \chi$. By assumption (a2) it follows that $\mu^X(I_x) > 0$. Moreover, the number of observed values in I_x from (11) tends to infinity when $n \to \infty$ with probability one.
We can write the estimator $\widehat{\mathbb{B}}_{\omega_{I_x}}$ from (18), that is

$$\widehat{\mathbb{B}}_{\omega_{I_x}} = \frac{B_{p,K}^2 \sum_{i=1}^{\hat{n}^*} \left[q^{(p+1)}\left(X_i^*; \hat{\boldsymbol{\eta}}^*\right) \right]^2 \mathbb{I}(X_i^* \in I_x)}{\sum_{i=1}^{\hat{n}^*} \mathbb{I}(X_i^* \in I_x)}. \tag{34}$$

Proposition 1. *Under the assumptions (a1)–(a6), $\widehat{\mathbb{B}}_{\omega_{I_x}}$, defined in (34), $I_x \subseteq \chi$, is consistent in the sense that:*

$$\widehat{\mathbb{B}}_{\omega_{I_x}} - \mathbb{B}_{\omega_{I_x}} = O_p\left(\left(\frac{\log n}{n} \right)^{1/2} \right)$$

where $\mathbb{B}_{\omega_{I_x}}$ is defined in (17).

Proof. The estimator in (34) can be written as

$$\widehat{\mathbb{B}}_{\omega_{I_x}} = \frac{B_{p,K}^2 \frac{1}{\hat{n}^*} \sum_{i=1}^{\hat{n}^*} \left[q^{(p+1)}\left(X_i^*; \hat{\boldsymbol{\eta}}^*\right) \right]^2 \mathbb{I}(X_i^* \in I_x)}{\frac{1}{\hat{n}^*} \sum_{i=1}^{\hat{n}^*} \mathbb{I}(X_i^* \in I_x)}.$$

The quantity $B_{p,K}^2$ is known. Put $\hat{n}^* = n^*$. Note that $n^* = O(n)$. Then

$$\frac{1}{n^*} \sum_{i=1}^{n^*} \mathbb{I}(X_i^* \in I_x) = \mu^X(I_x) \left(1 + O_p(n^{-1/2}) \right).$$

Given the neural network estimator based on $\{Y_i^*, X_i^*\}$, $i = 1, 2, \ldots, n^*$ and using Lemma 3 we have that

$$\frac{1}{n^*} \sum_{i=1}^{n^*} \left(q^{(p+1)}\left(X_i^*; \hat{\boldsymbol{\eta}}^*\right) - m^{(p+1)}(X_i^*) \right)^2 \mathbb{I}\left(X_i^* \in I_x\right) = O_p\left(\left(\frac{\log n}{n} \right)^{1/2} \right).$$

Therefore, it implies that

$$\frac{1}{n^*} \sum_{i=1}^{n^*} \left(q^{(p+1)}\left(X_i^*; \hat{\boldsymbol{\eta}}^*\right) \right)^2 \mathbb{I}(X_i^* \in I_x)$$
$$= \int_{I_x} \left(m^{(p+1)}(x) \right)^2 f_X(x) dx \left(1 + O_p\left(\left(\frac{\log n}{n} \right)^{1/2} \right) \right).$$

Now, we have

$$\widehat{\mathbb{B}}_{\omega_{I_x}} = \mathbb{B}_{\omega_{I_x}} \frac{1 + O_p\left(\left(\frac{\log n}{n}\right)^{1/2}\right)}{1 + O_p(n^{-1/2})}.$$

Using the Taylor's expansion it follows that

$$\widehat{\mathbb{B}}_{\omega_{I_x}} - \mathbb{B}_{\omega_{I_x}} = O_p\left(\left(\frac{\log n}{n}\right)^{1/2}\right).$$

Finally, using the same arguments as in the end of the proof of Lemma 2, the proof is complete. □

Now we have to consider the estimator $\widehat{\mathbb{V}}_{\omega_{I_x}}$ in (18).

Proposition 2. *Using the same assumptions as in Proposition 1, then $\widehat{\mathbb{V}}_{\omega_{I_x}}$, defined in (18), with $I_x \subseteq \chi$ is consistent in the sense that:*

$$\widehat{\mathbb{V}}_{\omega_{I_x}} - \mathbb{V}_{\omega_{I_x}} = O_p\left(\left(\frac{\log n}{n}\right)^{1/2}\right).$$

Proof. It is immediate applying Lemma 2 and following the same arguments as in Proposition 1. □

Now, we consider the optimal bandwidth and its plug-in estimator for the unknown function $m(\cdot)$ in model (11) or its derivatives, using the Local Polynomial Estimator.

Let $I_x^n = [x - a_n/2; x + a_n/2]$ where $a_n \to 0$ when $n \to \infty$. Let $\hat{h}_{I_x}$ be the estimator of h_{I_x}, defined in the (19).

Proposition 3. *Using the assumptions (a1)–(a4), then*

i) if $a > 0$, the assumptions (a5) and (a6) hold, and assuming that $m^{(p+1)}(\cdot) \not\equiv 0$ in I_x, then $\hat{h}_{I_x}$ is consistent in the sense that

$$\frac{\hat{h}_{I_x} - h_{I_x}}{h_{I_x}} = O_p\left(\left(\frac{\log n}{n}\right)^{1/2}\right);$$

ii) if $na_n^5 \to \infty$ when $n \to \infty$, assuming that $m^{(p+3)}(x)$ is bounded on χ, and assuming that $m^{(p+1)}(\cdot) \not\equiv 0$ in I_x^n for some $n \geq 1$ and $m^{(p+1)}(x) \neq 0$, then $\hat{h}_{I_x^n}$ is consistent in the sense that:

$$\frac{\hat{h}_{I_x^n} - h_{I_x^n}}{h_{I_x^n}} = O_p\left(\left(\frac{\log na_n^5}{na_n^5}\right)^{1/2}\right).$$

Proof. First, consider a value $a > 0$, for the part *i)*. Then we have, by assumptions, that $\widehat{\mathbb{B}}_{\omega_{I_x}} > 0$ in probability, when $n \to \infty$. Therefore, by Proposition

1 and Proposition 2 and using the same approach as in Ruppert *et al.* (1995) (page 1265), we have that $\frac{\hat{h}_{I_x}-h_{I_x}}{h_{I_x}} = O_p((\frac{\log n}{n})^{1/2})$.

When we have a sequence $\{a_n\}$, for the part *ii)*, it is necessary to assure that the number of observations in I_x^n goes to infinity in some sense in order to apply Proposition 1 and 2.

By assumption (a2), and using the mean value theorem, we have that $n\mu^X(I_x^n) = O(na_n)$. It can be shown that $\mu_n^X(I_x^n) - \mu^X(I_x^n) = O_p((n/a_n)^{-1/2})$. Therefore, we have that $n\mu_n^X(I_x^n) = n\mu^X(I_x^n)[1 + O_p((n/a_n)^{-1/2})]$ so that $n\mu_n^X(I_x^n) = O_p(na_n)$.

Since $m^{(p+3)}(x)$ is bounded on χ, using Hornik *et al.* (1994) we have that

$$\left[\int_{x-a_n/2}^{x+a_n/2} \left(m_n^{(p+1)}(x) - m^{(p+1)}(x)\right)^2 f_X(x)dx\right]^{1/2} = O\left(\frac{a_n^2}{\sqrt{d_n}}\right)$$

where $m_n^{(p+1)}(x)$ is the derivative of order $p+1$ of the neural network function with the true weights and d_n is the number of neurons in the hidden layer.

Following the proof of Case 1.1 in Chen & Shen (1998) and given that the number of observations in $[x - a_n/2, x + a_n/2]$ is $O(a_n n)$, we have to find d_n such that the following relation

$$\frac{d_n^2}{a_n^4} \log d_n = O(a_n n) \tag{35}$$

is satisfied. It can be shown that $d_n = (\frac{a_n^5 n}{\log(a_n^5 n)})^{1/2}$ satisfies the (35) since $a_n^5 n \to \infty$ when $n \to \infty$. Using the same arguments as in the proof of Lemma 2, we have that

$$\frac{\hat{h}_{I_x^n} - h_{I_x^n}}{h_{I_x^n}} = O_p\left(\left(\frac{\log na_n^5}{na_n^5}\right)^{1/2}\right).$$

□

Proof of Theorem 1. Since we have the compact set χ, then there exists a finite a^* such that $I_x^a = \chi$, $\forall a \geq a^*$ and $\forall x \in \chi$. (Note that $I_x^a \equiv I_x$. We write I_x^a instead of I_x only to give more evidence for the parameter a). So we can build the set $I^* = [b_1, b_2]$ with $b_1 > 0$ and $b_2 = a^* > b_1$.

By part *i)* of Proposition 3, with $p = 1$ and $v = 0$, we can write

$$\begin{aligned}\sup_{a\in I^*}\left\{\frac{\hat{h}_{I_x^a} - h^{MISE_{I_x^a}}}{h^{MISE_{I_x^a}}}\right\} \quad &\leq \quad \sup_{a\in I^*}\left\{\frac{\hat{h}_{I_x^a}}{h_{I_x^a}} - 1\right\} \sup_{a\in I^*}\left\{\frac{h_{I_x^a}}{h^{MISE_{I_x^a}}}\right\} \qquad (I \cdot II) \quad + \\ &+ \sup_{a\in I^*}\left\{\frac{h_{I_x^a}}{h^{MISE_{I_x^a}}} - 1\right\} \qquad (III).\end{aligned}$$

By Theorem 2 of Fan *et al.* (1996) it follows that $III = O_p(n^{-2/5}\log n)$ because we have a continuous mapping between $a \in I^*$ and the bandwidth $h_{I_x^a} \in [n^{-b3}, n^{-b_4}]$, for suitable b_3 and b_4. So, $II = 1 + O_p(n^{-2/5}\log n)$.

Let $X_n(a) = \int_{I_x^a}(q(u;\boldsymbol{\eta}) - m(u))^2 f_X(u)du$ and $\hat{X}_n(a) = \int_{I_x^a}(q(u;\hat{\boldsymbol{\eta}}^*) - m(u))^2 f_X(u)du$ as in Lemma 1, where $q(\cdot;\cdot)$ is the neural network function with the true parameter vector and estimator, respectively.

To show the rate of convergence for I, it is sufficient to derive the rate of convergence for $\sup_{a\in I^*}\hat{X}_n(a)$ and the result follows using Lemmas 2, 3 and Propositions 1, 2 and 3, part (i).

There exists a finite sequence in I^*, say $\{a_i\}$, $1 \leq i \leq M$, with M finite integer, such that

$$\sup_{a\in I^*}\hat{X}_n(a) \leq \max_{1\leq i\leq M}\hat{X}_n(a_i) + \max_{1\leq i\leq M}\sup_{|a_i-a|\leq M^{-1}}\left|\hat{X}_n(a_i) - \hat{X}_n(a)\right| = I' + II'.$$

By Lemma 1 we have that $I' \leq MO_p(d_n^{-1}) = O_p(d_n^{-1})$ where d_n is defined in assumption (a6). Using Hornik *et al.* (1994) it follows that $|X_n(a_1) - X_n(a_2)| \leq \frac{C|a_1-a_2|}{d_n}$, with $0 < C < \infty$. Therefore, using again Lemma 1, we have that $II' \leq MO_p(d_n^{-1})$. Since M is a finite constant then we can conclude that $\sup_{a\in I^*}\hat{X}_n(a) = O_p(d_n^{-1})$. The proof is complete. □

Proof of Corollary 1. By (10), we can find an arbitrary large value of a such that $h_{I_x} \equiv h_G^{opt}$. Therefore, by part *i)* of Proposition 3, with $p = 1$ and $v = 0$, and using the (27), we have

$$\begin{aligned}\frac{\hat{h}_{I_x} - h^{MISE}}{h^{MISE}} &= \frac{\hat{h}_{I_x} - h_{I_x}}{h_{I_x}}\frac{h_{I_x}}{h^{MISE}} + \frac{h_{I_x} - h^{MISE}}{h^{MISE}} \\ &= O_p\left(\left(\frac{\log n}{n}\right)^{1/2}\right) + O_p(n^{-2/5}).\end{aligned}$$

So the rate of convergence is dominated by the $O_p(n^{-2/5})$ term. □

Proof of Theorem 2. We have that

$$\frac{\hat{h}_{I_{x_0}^n} - h^{MSE}(x_0)}{h^{MSE}(x_0)} = \frac{\hat{h}_{I_{x_0}^n} - h_L^{opt}(x_0)}{h_L^{opt}(x_0)}\frac{h_L^{opt}(x_0)}{h^{MSE}(x_0)} + \frac{h_L^{opt}(x_0) - h^{MSE}(x_0)}{h^{MSE}(x_0)}. \tag{36}$$

Now we can write the first term at the right hand side of (36) as:

$$\frac{\hat{h}_{I_{x_0}^n} - h_L^{opt}(x_0)}{h_L^{opt}(x_0)} = \frac{\hat{h}_{I_{x_0}^n} - h_{I_{x_0}^n}}{h_{I_{x_0}^n}}\frac{h_{I_{x_0}^n}}{h_L^{opt}(x_0)} + \frac{h_{I_{x_0}^n} - h_L^{opt}(x_0)}{h_L^{opt}(x_0)}. \tag{37}$$

By part *ii)* of Proposition 3, with $p = 1$ and $v = 0$, it follows that $\frac{\hat{h}_{I_{x_0}^n} - h_{I_{x_0}^n}}{h_{I_{x_0}^n}} = O_p((\frac{\log na_n^5}{na_n^5})^{1/2})$. Using the Taylor's expansion for $\mathbb{B}_{I_{x_0}} \equiv \mathbb{B}(a;x_0)$ with respect to a, we have

$$\mathbb{B}(a;x_0) = \mathbb{B}(0;x_0) + \mathbb{B}'(0;x_0)a + \mathbb{B}''(0;x_0)\frac{a^2}{2} + \mathbb{B}'''(\tilde{a};x_0)\frac{a^3}{6}$$

where $\tilde{a}$ is a value between 0 and a. Using the conditions of the Theorem, it follows that $\mathbb{B}(a;x_0) = (m''(x_0))^2 f_X(x_0)a + c_1 a^3$, with $c_1 < \infty$. After some algebra, we can write $h_{I^n_{x_0}}$ as:

$$h_{I^n_{x_0}} = h_L^{opt}(x_0) O\left(\left(\frac{1}{1+c_1 a_n^2}\right)^{1/5}\right)$$

When we consider a_n instead of a, then also c_1 depends on n. But, in this case $c_1 < \infty$, $\forall n$ and for $n \to \infty$. So, for simplicity we remain the constant c_1.

Therefore, the last term in (37) is

$$\frac{h_{I^n_{x_0}}}{h_L^{opt}(x_0)} - 1 = O\left(\left(\frac{1}{1+c_1 a_n^2}\right)^{1/5}\right) - 1 = O(a_n^2).$$

So, we have that

$$\frac{\hat{h}_{I^n_{x_0}} - h_L^{opt}(x_0)}{h_L^{opt}(x_0)} = O_p\left(\left(\frac{\log n a_n^5}{n a_n^5}\right)^{1/2}\right) + O(a_n^2).$$

Using the (33), it follows that

$$\frac{\hat{h}_{I^n_{x_0}} - h^{MSE}(x_0)}{h^{MSE}(x_0)} = O_p\left(\left(\frac{\log n a_n^5}{n a_n^5}\right)^{1/2}\right) + O(a_n^2) + O_p(n^{-2/5}).$$

□

Acknowledgements

The authors are very grateful to the referees for their valuable comments, which resulted in an improvement over the original manuscript.

References

Barron, A. R. (1993). "Universal approximation bounds for superpositions of a sigmoidal function", *IEEE Transactions on Information Theory*, 39, 930–945. MR1237720

Barron, A. R. (1994). "Approximation and estimation bounds for artificial neural networks", *Machine Learning*, 14, 115–133.

Cao, R. (2001). "Relative efficiency of local bandwidths in kernel density estimation", *Statistics*, 35, 113–137. MR1820680

Chen, X. and Shen, X. (1998). "Sieve extremum estimates for weakly dependent data", *Econometrica*, 66, 289–314. MR1612238

Choi, E., Hall, P. and Rousson, V. (2000). "Data sharpening methods for bias reduction in nonparametric regression", *Annals of Statistics*, 28, 1339–1355. MR1805786

Fan, J. and Gijbels, I. (1995). "Data-driven bandwidth selection in local polynomial fitting: Variable bandwidth and spatial adaptation", *Journal of the Royal Statistical Society, series B*, 57, 371–394. MR1323345

— (1996). *Local Polynomial Modelling and Its Applications*, Chapman and Hall, London. MR1383587

Fan, J., Gijbels, I., Hu, T.-C. and Huang, L.-S. (1996). “A study of variable bandwidth selection for local polynomial regression”, *Statistica Sinica*, 6, 113–127. MR1379052

Fan, J. and Huang, L.-S. (1999). “Rates of convergence for the pre-asymptotic substitution bandwidth selector”, *Statistics and Probability Letters*, 43, 309-316. MR1708098

Gao, J. and Gijbels, I. (2008). “Bandwidth selection in nonparametric kernel testing”, *Journal of the American Statistical Association*, 103, 1584–1594. MR2504206

Giordano, F. and Parrella, M. L. (2008). “Neural networks for bandwidth selection in local linear regression of time series”, *Computational Statistics & Data Analysis*, 52, 2435–2450. MR2411949

Gluhovsky, I. and Gluhovsky, A. (2007). “Smooth location-dependent bandwidth selection for local polynomial regression”, *Journal of the American Statistical Association*, 102, 718–725. MR2370862

Hall, P. and Schucany, W. R. (1989). “A local cross-validation algorithm”, *Statistics and Probability Letters*, 8, 109–117. MR1017876

Härdle, W., Hall, P. and Marron, J. S. (1988). “How far are automatically chosen regression smoothing parameters from their optimum?”, *Journal of the American Statistical Association*, 83, 86–99. MR0941001

He, H. and Huang, L.-S. (2009). “Double-smoothing for bias reduction in local linear regression”, *Journal of Statistical Planning and Inference*, 139, 1056–1072. MR2479849

Hornik, K., Stinchcombe, M., White, H. and Auer, P. (1994). “Degree of approximation results for feedforward networks approximating unknown mappings and their derivatives”, *Journal of Neural Computation*, 6, 1262–1275.

Prewitt, K. and Lohr, S. (2006). “Bandwidth selection in local polynomial regression using eigenvalues”, *Journal of the Royal Statistical Society, series B* , 68, 135–154. MR2212579

Ruppert, D., Sheather, M. P. and Wand, P. (1995). “An effective bandwidth selector for local least squares regression”, *Journal of the American Statistical Association*, 90, 1257–1270. MR1379468

Ruppert, D. and Wand, P. (1994). “Multivariate locally weighted least squares regression”, *Annals of Statistics*, 22, 1346–1370 MR1311979

Stone, C. J. (1980). “Optimal Rates of Convergence for Nonparametric Estimators”, *The Annals of Statistics*, 8, 1348–1360. MR0594650

Wang, K. and Gasser, T. (1996). “Optimal Rate for Estimating Local Bandwidth in Kernel Estimators of Regression Functions”, *Scandinavian Journal of Statistics*, 23, 303–312. MR1426122