meaningful to perform an asymptotic analysis at places where there are hardly any data. It may seem a bit ironical that Chu and Marron make the same assumption "bounded from below" in the same paper (assumption A.4 of Section 3). In an interesting paper, Fan (1990) concludes independently about the Nadaraya-Watson estimator (remark 2, Section 3) "... hence its asymptotic minimax efficiency is arbitrary small."

CONCLUSIONS

Our conclusion is that the convolution weights are clearly superior to evaluation weights for fixed design, since we have the same variance for both methods but a nasty bias for evaluation weights. For random design, the problem seems to us more open: There is a minimax argument, and we would like to repeat a general argument, which is not well quoted by Chu and Marron (Section 3): "The latter authors [Gasser and colleagues] in particular seem to feel that variability is not a major issue, apparently basing their feelings on the premise that it is always easy to gather simply more data." What we said when discussing the structural bias of the evaluation weights was the following (Gasser and Engel, 1990): "These bias problems are particularly accentuated in the scientific process of many empirical sciences: studies are usually replicated by sticking to the design of the previously published study. In this way, qualitatively misleading phenomena as obtained by the Nadaraya-Watson estimator will be attributed even more confidence."

OUTLOOK

One way out of this problem has been opened by Fan (1990), who showed that for random design local polynomials have the same bias as convolution weights and the same variance as evaluation weights (the equivalence of local polynomials to convolution type kernel estimators for fixed design had been shown by Müller, 1987). A further possibility for improving the variance properties of convolution weights has been described by Chu and Marron in Section 6.

Comment

Birgit Grund and Wolfgang Härdle

1. OBJECTIVES OF SMOOTHING

Smoothing has become a standard data analytic tool. A good indicator of this is the increased offer of smoothing procedures in a variety of standard statistical software packages. It is therefore high time to provide background information that enables statisticians and users to critically evaluate the—in the meantime—rich basket of smoothing tools. The paper by Chu and Marron meets this demand for information and compares two different kernel regression estimators on an easy, understandable level. The authors combine successfully careful mathematical discussion with heuristic arguments in a well-done exposition. Cleverly chosen striking examples provide an easy access to not immediately apparent problems in smoothing for data analysis. We congratulate the authors to this valuable contribution.

Among the many objectives of smoothing, there are certainly the two perhaps most discussed. These are P1: to find structure; and P2: to construct estimators from a probability distribution.

We agree that the interplay of these two objectives is vital for an honest parameter-free data

Birgit Grund is Assistant Professor, School of Statistics, University of Minnesota, 352 Classroom-Office Building, 1994 Bufford Avenue, St. Paul, Minnesota 55108, and CORE Research Fellow, Université Catholique de Louvain, Belgium. Wolfgang Härdle is Professor of Statistics, CORE, Université Catholique de Louvain, 34 Voie du Roman Pays, B01348 Louvain-la-Neuve, Belgium. He is currently visiting CentER, Tilburg University, The Netherlands.

analysis. Each responsible statistician should be aware of the limitations of the used methods. Even if we pretend to follow mainly P1, that is, to look for structure in the data, breakpoints, etc., without caring too much for theoretical optimality, we impose implicitly certain assumptions on the underlying probability structure. In the case of nonparametric curve regression those assumptions could concern the design distribution (uniform or with modes), the observations (independent/dependent), the error structure conditional on X (homoscedastic/heteroscedastic) and some features of the regression curve. Thus, the degree of trusting our own results is always an indicator for trusting the validity of our model, whether we recognize or neglect its existence.

On the other hand, "elaborated" methods, which provide estimators with good theoretical properties, more obviously require a bunch of assumptions. We are aware of them, but usually can't guarantee their validity. Therefore, any outcome of a smoothing algorithm should be regarded skeptically and checked whether it is plausible, regardless of whether we mean to follow P1 or P2.

The problem of the unknown underlying probability structure is also present, if we decide to trust either the evaluation or the convolution estimator. Certainly, the latter has its deficiencies for a nonuniform design. Figure 5 in Chu and Marron makes it very clear. One should, of course, not conclude from this and also the mean square error discussion there that the evaluation estimator is the "universal wonderful super-smoother" in all situations. We shortly demonstrate in Section 2 that there is no uniform outperformance of one estimator over the other; in a simple example we display where and to which extend we can expect superiority of the evaluation or the convolution estimator.

In our opinion, there are other important objectives, too, for example P3: *computational efficiency*.

Particularly the problem of computational efficiency is oftenly underestimated by theoreticians, although it influences the choice and applicability of the smoothing method significantly in real life. Certainly, P3 becomes a vital issue when iterative algorithms have to be used, in optimizing smoothing parameters or solving for implicitly defined functions like in the generalized additive modeling; see Hastie and Tibshirani (1990).

In Section 3, we demonstrate how kernel-type estimators can be modified to ensure fast computation.

2. EVALUATION OR CONVOLUTION?

Let us confine to the decision problem: evaluation or convolution estimator?

We assume the random design model with homoscedastic variances, given by

$$Y_j = m(X_j) + \varepsilon_j,$$

for $i = 1, \ldots, n$, where the (X_j, Y_j)'s are identically distributed variables; the design variable X_1 has the probability density f; the ε_j's have mean 0 and variance σ^2. Following the notation of the paper by Chu and Marron, we denote the evaluation and the convolution estimator by $\hat{m}_E$ and $\hat{m}_C$, respectively.

Obviously, there is a trade-off between bias and variance. For the heuristical understanding of estimates, it is certainly advisable to regard both effects separately. Nevertheless, in real life, we have to decide for the one or the other estimator taking into account both bias and variance simultaneously, and the final choice of the "better" estimate will depend as well on the underlying problem as on the optimality criterion.

In this section, we compare $\hat{m}_E$ and $\hat{m}_C$ by the relative efficiency

$$RE = \frac{IMSE_A(\hat{m}_C, h_{IMSE_A})}{IMSE_A(\hat{m}_E, h_{IMSE_A})}, \tag{2.1}$$

where $IMSE_A(\hat{m}, h)$ denotes the leading term of the integrated mean square error of $\hat{m}$, for $\hat{m}$ representing either $\hat{m}_E$ or $\hat{m}_C$. Using the notation in Section 5 of the paper by Chu and Marron, we have

$$IMSE_A(\hat{m}, h) = n^{-1}h^{-1}V + h^4B^2, \tag{2.2}$$

where $V = \int v\,dx$ and $B^2 = \int b^2\,dx$. The optimal bandwidth h_{IMSE_A} is defined to minimize the right-hand side of (2.2).

Simple calculus provides us

$$\frac{IMSE_A(\hat{m}_C, h_{IMSE_A})}{IMSE_A(\hat{m}_E, h_{IMSE_A})} = \frac{\left[\int (m'')^2\,dx\right]^{1/5}(3/2)^{4/5}}{\left[\int (m'' + 2m'f'/f)^2\,dx\right]^{1/5}}. \tag{2.3}$$

The following simple examples are designed to demonstrate the interplay of the design distribution and the shape of the true regression curve in determining the relative efficiencies of $\hat{m}_E$ and $\hat{m}_C$.

EXAMPLE 1. We consider the class of regression curves

$$m_\gamma(x) = \left(\frac{1+x}{2}\right)^{\gamma+1}, \tag{2.4}$$

for $x \in [-1, 1]$ and for $\gamma \geq 0$, and the family of random designs with densities

$$(2.5) \qquad f_\omega(x) = (1 - \omega)(1/2) + \omega\varphi_{[-1,1]}(x),$$

for $\omega \in [0, 1]$, where $\varphi_{[-1,1]}$ denotes the density of the standard normal distribution, truncated to $[-1, 1]$. For $\omega = 0$, the design variable X is uniformly distributed on $[-1, 1]$; the most concentrated design in this example is the truncated standard normal distribution ($\omega = 1$).

Some representatives of the regarded regression curves are shown in Figure 1. The parameter $\gamma = 0$ corresponds to a linear function, and for growing γ the curves deviate more and more from the straight line.

Figure 2 below displays RE (see (2.1)), for all combinations of designs f_ω and true regression curves m_γ. With respect to the γ-scale here, the curves in Figure 1 are equidistant and represent the whole range.

We have chosen the above families for convenient control of bias and variance.

Under uniform design ($\omega = 0$) both $\hat{m}_E$ and $\hat{m}_C$ have the same bias, but $\hat{m}_C$ has a bigger variance. More precisely, $v_C = 3v_E/2$ for all $x \in [-1, 1]$, for all bandwidths and any regression curve. This setting causes

$$RE = (3/2)^{4/5} \quad \text{for all } \gamma \geq 0$$

and is reflected by the straight line on the right front side of the box.

The left front side of the box in Figure 2 corresponds to estimating a straight line under increasingly nonuniform design, the ideal background for the convolution estimator. We see that the trade-off between bias and variance begins to favor $\hat{m}_C$ at about $\omega \approx 1/3$.

The region where the convolution estimator is superior to the evaluation estimator ($RE \leq 1$) is rather small for this example. Even under the most nonuniform design ($\omega = 1$) the convolution estimator is better just for $\gamma < 0.5$ (see Figure 1).

EXAMPLE 2. We regard the same class of regression curves, but the class of random designs is now given by the densities

$$f_\omega(x) = \varphi_{[-1,1]}\left(\frac{x}{\omega}\right),$$

for $\omega > 0$, that is, the truncated $N(0, \omega^2)$ normal distributions. Figure 3 below shows some representatives. We see that $\omega = 2.3$ describes almost the uniform design.

The relative efficiency RE is displayed in Figure 4. Note, that the densities in Figure 3 are not

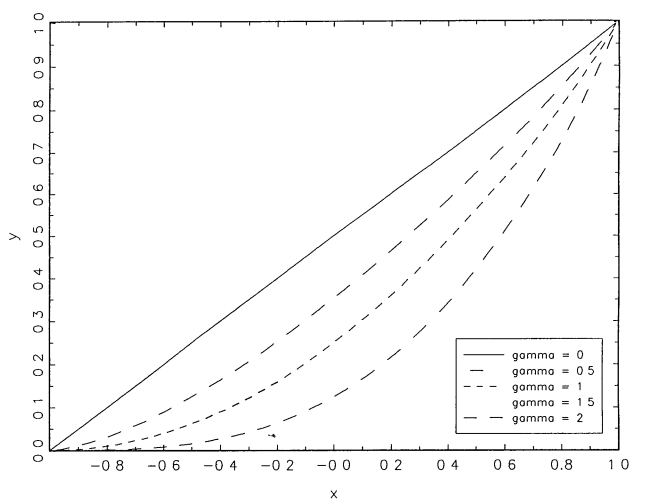

FIG. 1. *Regression curves* $m_\gamma(x) = ((1 + x)/2)^{1+\gamma}$ *for different values of* γ. *With respect to the* γ*-scale of the Figures* 2 *and* 4, *the curves are equidistant and cover the whole range.*

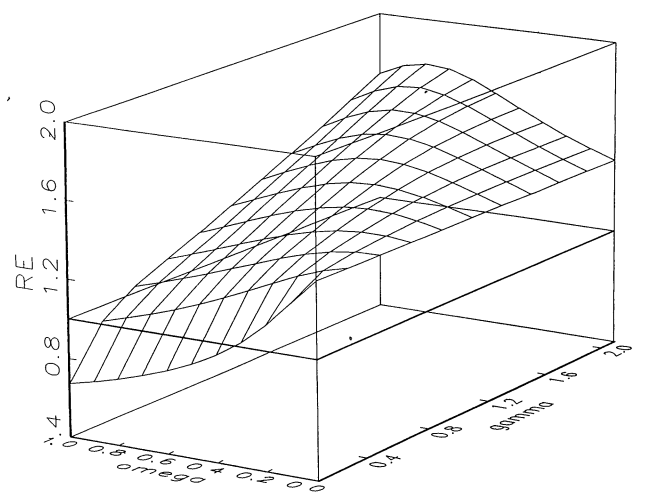

FIG. 2. *Relative efficiency RE of* $\hat{m}_C$ *and* $\hat{m}_E$, *in dependence from design density* f_ω *and regression curve* m_γ; *see Example* 1.

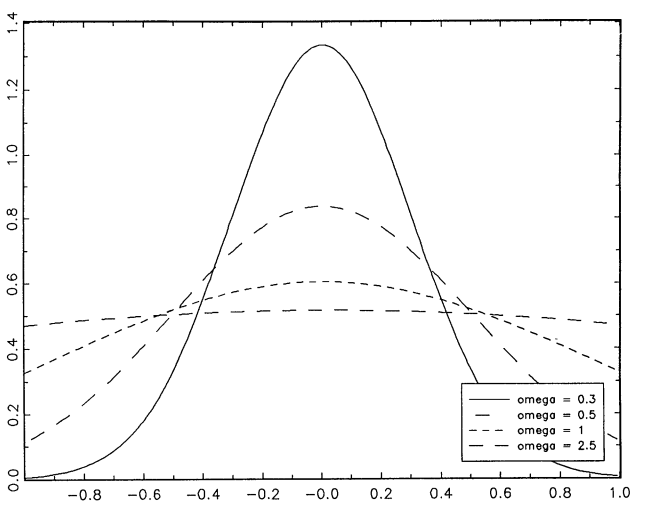

FIG. 3. *Truncated normal densities* $f_\omega(x) = \varphi_{[-1,1]}(x/\omega)$ *for different values of* ω.

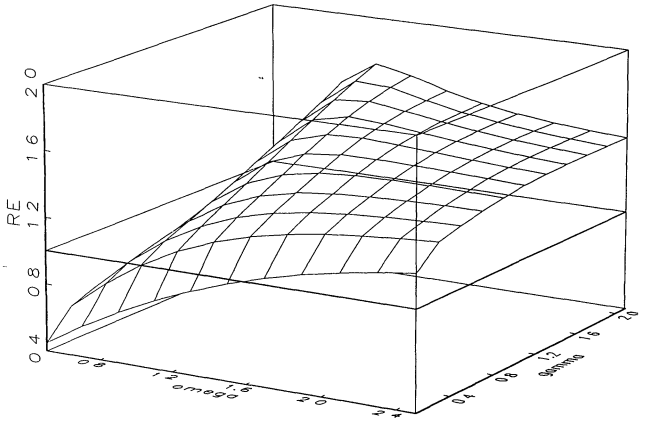

FIG. 4. *Relative efficiency RE of $\hat{m}_C$ and $\hat{m}_E$, in dependence from design density f_ω and regression curve m_γ; see Example 2.*

equidistant with respect to the ω-scale of Figure 4.

Figure 4 provides a similar impression about the relative behavior of the convolution and the evaluation estimator as we saw in Example 1. Again, the convolution estimator is preferable for regression curves with low slope (γ small) and rather concentrated design (ω small). The region where the convolution estimator is superior seems to be greater here. But note that values $\omega \leq 0.4$ correspond to a rather peaky design density.

3. COMPUTATIONALLY FAST ESTIMATORS

We have motivated the need for fast smoothing techniques. One possibility to speed up computation is to use weighted averaging of rounded points (WARPing). This technique is based on the following three steps: discretize the data, generate kernel weights and convolute the binned data with the kernel.

Regression data are discretized by counting the X observations that fall into bins $[(j - 1/2)\delta, (j + 1/2)\delta)$, where δ denotes the (small) bandwidth and j varies over all integers. Additionally, one records the sum of the response Y's in these bins and maintains a pointer structure to nonempty bins. We describe this technique here only for the evaluation estimator, it could of course be applied also to the $\hat{m}_C$ estimator.

WARPing is designed for kernels with compact support. Let us assume that K is supported on $[-1, 1]$. Define the index function

$$(3.1) \qquad \iota(x) = j \Leftrightarrow x \in \left[\left(j - \tfrac{1}{2}\right)\delta, \left(j + \tfrac{1}{2}\right)\delta\right),$$

which returns the index of the small bin that x belongs to. Then the WARPing approximation to the evaluation estimator $\hat{m}_E$ is

$$\hat{m}_{M,K}(x) = \frac{\sum_{i=1}^{n} K\left((\iota(x) - \iota(X_i))/M\right) Y_i}{\sum_{i=1}^{n} K\left((\iota(x) - \iota(X_i))/M\right)};$$

here the integer $M \approx h\delta^{-1}$ stands for the bandwidth of the discretized kernel. An easy recalculation leads to

$$(3.2) \qquad \hat{m}_{M,K}(x) = \frac{\sum_{l=1-M}^{M-1} K(l/M) Y_{\bullet \iota(x)+l}}{\sum_{l=1-M}^{M-1} K(l/M) n_{\iota(x)+l}},$$

where n_j and $Y_{\bullet j}$ denote the number of X's that fall into bin j and the sum of the corresponding Y's, respectively.

Formula (3.2) shows that essentially the problem of varying h depends now only on the number of bins, which is usually orders of magnitude smaller than the sample size n.

Thus, the above mentioned iterations and successive calls of kernel smoothing subroutines is performed much faster. Suppose we want to estimate m at N points. The evaluation kernel estimator requires $O(nN)$ operations for a kernel with noncompact support like the Gaussian kernel. For a kernel with compact support, this numerical effort is reduced slightly to $O(nhN)$. For the WARPing

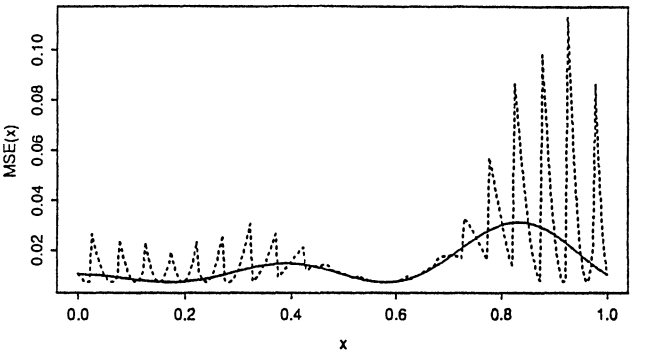

FIG. 5. *Leading term of the MSE of $\hat{m}_E$ (solid line) and of the WARPing step function $\hat{m}_{M,K}$. Underlying model: $m(x) = x \sin(2\pi x) + 3x$, uniform design, $\sigma^2 = 0.25$. Parameters: $h = 0.25$, $M = 5$, $n = 100$, quartic kernel.*

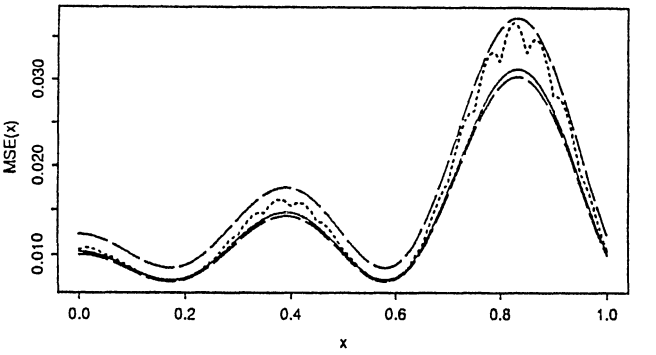

FIG. 6. *Leading term of the MSE of $\hat{m}_E$ (solid line) and of the corresponding WARPing polygon function (dotted line), with conservative bounds for the latter (dashed line). The same model and parameters as in Figure 5.*

approximation, we need n operations to discretize the data into N_B nonempty bins. Thus, the numerical effort for this method is of order $O(n + N_B M)$.

Of course, the WARPing method introduces a discretization bias. The bias may be reduced by joining the obtained discrete step function (see (3.2)), via a polygon. Breuer (1990) has computed for $m(x) = x \sin(2\pi x) + 3x$ and uniform design the MSE as a function of x for both the $\hat{m}_E$ estimator and the WARPed estimator $\hat{m}_{M,K}$.

In Figure 5, the discretization bias is seen to be quite drastic, although we gained in speed of computation. The linear interpolant has a much better bias behavior, as is seen in Figure 6. For this estimator conservative bounds for the numerical discretization error and its effect on $MSE(x)$ can be given and are displayed in Figure 6 as long dashed lines.

ACKNOWLEDGMENT

The work of the first author was in part financially supported by CentER, Tilburg.

Comment

Jeffrey D. Hart

Chu and Marron have provided us with a clear and thorough account of the relative merits of evaluation and convolution type kernel regression estimators. One is left with the impression that neither type of estimator is to be preferred universally over the other. We learn, for example, that the weights of the convolution estimator sometimes have the unsettling behavior exhibited in Figures 6b and 7 of Chu and Marron. The authors make it clear that there are a number of factors, including type of design (fixed or random), design density and nature of underlying regression function, that need to be considered before choosing an estimator type. Having reading their article, I now have a slight preference for $\hat{m}_E$ over $\hat{m}_C$ in the random design case, at least in the absence of any information about the design density or regression curve. When the design points are nonrandom and evenly spaced, I prefer $\hat{m}_C$, since its convolution form appeals to me, and since boundary kernels are easy to construct with $\hat{m}_C$ (see Gasser and Müller, 1979). Below I will mention a modification of $\hat{m}_C$ that I feel is a viable competitor of $\hat{m}_E$ even in the random design case.

The authors' point about the down weighting phenomenon of the convolution estimator is certainly well taken. However, I would like to question an aspect of their comparison of the variances of $\hat{m}_E$ and $\hat{m}_C$. As the authors note in Section 4, the biases of the two estimators are not comparable, the bias of $\hat{m}_E$ being smaller in some cases and that of $\hat{m}_C$ smaller in other cases. It follows that "good" bandwidths for the estimators will generally be different. Why then is it sensible to compare $\mathrm{Var}(\hat{m}_E)$ and $\mathrm{Var}(\hat{m}_C)$ at the same value of h?

A little-used but informative way of comparing the errors of $\hat{m}_E$ and $\hat{m}_C$ is to consider the limiting distribution of

$$(1) \qquad \frac{|\hat{m}_E(x) - m(x)|}{|\hat{m}_C(x) - m(x)|}.$$

Unlike an MSE comparison, this approach takes into account the joint behavior of the two estimators. Suppose that Chu and Marron's assumptions (A.1)-(A.5) hold and that the design density is $U(0, 1)$. Suppose further that the bandwidths of $\hat{m}_E$ and $\hat{m}_C$ minimize their respective MSEs. Then it can be shown that, for each x, the ratio (1) converges in distribution to

$$(2) \qquad \left(\frac{2}{3}\right)^{2/5} \frac{|Z_1 + 1/2|}{|Z_2 + 1/2|} = R$$

as $n \to \infty$, where (Z_1, Z_2) have a bivariate normal distribution with $Z_1 \sim N(0, 1)$, $Z_2 \sim N(0, 1)$ and

$$\mathrm{Corr}(Z_1, Z_2) = \left(\frac{2}{3}\right)^{3/5} \int K(z) K\left(\left(\frac{2}{3}\right)^{1/5} z\right) dz \Big/ \int K^2 = \rho_K.$$

Jeffrey D. Hart is Associate Professor of Statistics, Texas A&M University, College Station, Texas 77843.