International Journal of Analysis and Applications

New Robust Estimators for the Nonparametric Regression Model: Application and Simulation Study

Faten Mohamed Ali Soliman[1,2], Amany Moussa Mohamed[1], Mohamed R. Abonazel[1,*]

[1]*Department of Applied Statistics and Econometrics, Faculty of Graduate Studies for Statistical Research, Cairo University, Giza, Egypt*

[2]*Obour Higher Institute for Management, Computers and Information Systems, Obour, Egypt*

**Corresponding author:* mabonazel@cu.edu.eg

ABSTRACT. This paper introduces new two robust kernel-based estimators (S Kernel and MM Kernel) for the nonparametric regression mode in the presence of outliers. Through comprehensive simulations, we evaluate their performance using Mean Squared Error (MSE), Mean Absolute Error (MAE), and Relative Efficiency (RE) under varying sample sizes and outlier contamination levels. Results demonstrate that robust estimators consistently outperform traditional kernel estimator, delivering the lowest estimation errors and highest efficiency, particularly in high-contamination scenarios. In contrast, the traditional kernel estimator proves highly sensitive to outliers. Also, our results highlight the superiority of the robust M Kernel estimator. This paper advances the field of robust nonparametric regression, offering practical solutions for datasets prone to outliers.

1. Introduction

Nonparametric regression models have garnered significant attention in the theoretical and applied statistics literature over the past few decades. The primary reason for their growing popularity and importance is that they do not rely on any specific form of the regression function characterizing the relationship between the dependent and explanatory variables. This makes them robust to model misspecification, as they do not impose any restrictive assumptions about the functional form of the regression model. One of the most popular methods for nonparametric Kernel regression was proposed by [1] and [2] and is known as the "Nadaraya-Watson" estimator (N-W), the Priestly-Chao estimator (PCE) suggested by [3] and Reweighted Nadaraya-Watson estimator (RNWE) introduced by [4]. In

Received Oct. 3, 2024

2020 Mathematics Subject Classification. 62G05, 62G08, 62G35.

Key words and phrases. nonparametric kernel; local polynomial; M kernel estimation; S kernel estimation; MM kernel regression; Monte Carlo simulation.

https://doi.org/10.28924/2291-8639-23-2025-163

ISSN: 2291-8639

the nonparametric estimation of the regression function, the local polynomial least squares (LS) regression is an effective and popular method, and its asymptotic theory [5].

The Speckman estimator is frequently utilized for parameter estimation in the partially linear model (PLM). It employs a kernel smoothing technique to estimate the nonparametric component of the model [6]. Abonazel et al. [7] modified the Speckman estimator by using the spline smoothing approach, and they showed that the partial residuals technique (PRT) based on the spline smoothing approach is more efficient than traditional PRT based on the kernel smoothing approach. Robust methods are defined by their capability to accurately fit most of the data, even in the presence of outliers. When only a small number of outliers exist, these methods produce results comparable to those of classical (non-robust) methods [8]. Elgohary et al. [9] proposed new modified estimators by combining the least trimmed squares (LTS) and ridge estimation methods.

Outliers can impact the estimation of a nonparametric function both at the edges and throughout the data range. The structure and development of robust estimation in those two straightforward models—locally weighted LS regression and local most minimal absolute deviation (LAD) regression—inspired several first proposed robust nonparametric regression estimators. By giving outliers a negative weight, locally weighted LS techniques seek to lessen the impact of outliers. They consist of the spline smoother and the Kernel [3] locally weighted polynomial LS fitting [10] and [11]. Local LAD regression differs from other methods. Fan and Hall [12] offered a framework and provided information on its asymptotic efficiency, while Tukey [13] proposed different versions of local median smoothing.

Ghement et al. [14] developed new estimators in the family to have weak consistency and asymptotic normality. They provided a metric known as max bias which allowed them to thoroughly and adequately quantify each M-robustness. Mahmoud et al. [15] suggested estimating the nonparametric regression function's derivative function when the data are noisy and have curves, our strong nonparametric derivative functions were created by designing three weights and integrating them into kernel smoothing. Wang et al. [16] suggested a reliable technique for doing so. Zhu and Jordan [17] presented a general framework for designing generative adversarial networks (GANs) to tackle high-dimensional robust statistics problems. This framework focuses on estimating unknown parameters of the true distribution from samples that have been adversarially corrupted. Salibian-Barrera [18] used many traditional nonparametric regression estimators (and their robust versions) can be quite challenging when dealing with a moderate or large number of explanatory variables. Therefore, recently proposed robust nonparametric regression methods, which scale effectively with an increasing number of covariates, are also examined.

By considering the estimated function of an explanatory variable, the derivative of a nonparametric function is calculated. The nonparametric regression function must be constructed correctly, and the data must not be noisy for this technique to be successful.

Otherwise, it might lead to inaccurate derivative estimates, which would be troublesome given the noise in the data.

The main objective of this paper is to introduce new robust nonparametric estimators. The remainder of the essay is structured as follows. Robust (M) nonparametric kernel estimator is presented in Section 2. Section 3 presents our proposed estimators. Section 4 compares the effectiveness of the estimators through a Monto Carlo simulation study. The results of the real-life application are presented in Section 5. Finally, Section 6 concludes this study.

2. Robust Nonparametric Kernel Estimator

Consider the case where(v_i, g_i), $1 \leq i \leq n$, are i.i.d. observations of a two-dimensional random vector (g,v) with $v_i = m_0(g_i) + \varepsilon_i$, where v_i refers to dependent variable, g_i refers to independent variable, m(.) is the unidentified mean function, and ε_ia stochastic error with a mean of zero and an unknown one can estimate $m(g_i)$ at around g_j as following using Taylor series[16].

$$m\,(g_i) = m(g_j) + \alpha_1\,(g_j - g_i\,) + \alpha_2\,(g_j - g_i\,)^2 + \cdots + \alpha_p\,(g_j - g_i\,)^p. \qquad (1)$$

They expressed the estimated nonparametric function as a linear function of the observed responses at the value g_i, $\widehat{m}(g_i)$. In other words, $\widehat{m}(g_i) = \sum_{j=1}^{n} \ddot{I}_j(g_i)v_j$, where $\ddot{I}_j(g_i)$ s are weights that are independent of g_j. In local polynomial regression with weights, a unique problem of weighted least squares is solved at each target point g_i for i=1,...,n. By employing weighted least squares and Kernel weights to fit the *p*th-degree polynomial model K $[h^{-1}\,(g_j - g_i\,)]$, the smoothing is achieved at a point g_i. The next weighted least squares regression issue, then, locally fits this polynomial.

$$\min_\alpha \sum_{j=1}^{n} [v_j - m(g_j)]^2\, {w^k}_j(g_i)\,. \qquad (2)$$

The Kernel weights used to estimate the unknown function at value g_i are denoted by ${w^k}_j(g_i)$, where k_h is a Kernel function, h is a bandwidth, and ${w^k}_j(g_i)$ is a Kernel function. When using (1), outliers have a considerable effect on the local polynomial estimator.

Our trusted weights are described as$R_j(g_i) = w_j^k(g_i) \times w_j^k$, j= (1, 2, …, n) where $w_j^k$is an appropriately selected smooth convex loss function, such as the kernel (Huber or bisquare) functions. The subsequent robust weighted least squares regression problem is locally tackled by the robust polynomial.

$$\min_\alpha \sum_{j=1}^{n} [v_j - \sum_{c=0}^{p} \beta_k\,(g_j - g_i\,)^c]^2\;R_j(g_i)$$
$$= \min_\alpha \sum_{j=1}^{n} [v_j - m(xg_j)\,]^2 R_j(g_i) \qquad (3)$$

$R_j(g_I)$ is expected to reduce the weight impact of outliers on derivative estimation, where *i* represents all observations and *j* denotes the outliers.

Can be written (1) in matrix as min $(v\text{-}g\alpha)^t R_x(v - g\alpha)$, where v= $(v_1, \ldots, v_n)^t$, $\alpha = (\alpha_0, \ldots, \alpha_n)$, g= $\begin{bmatrix} 1 & (g-g_1) & \cdots & (g-x_1)^p \\ \vdots & \vdots & \ddots & \vdots \\ 1 & (g-g_n) & \cdots & (g-g_n)^p \end{bmatrix}$, and R_g is the n×n diag matrix of weight, R_g = diag [R_1 (g_i),…, R_n (g_i)]. Assuming the matrix is invertible the (g^tR_gg) robust weighted least squares estimate is generated with no loss of generality

$$\widehat{\alpha}_R = (g^tR_gg)^{-1}\, g^tR_gv \tag{4}$$

Next, the authors discuss how to obtain the weights w_j^k. They propose three robust weight functions using the "M Estimator," "S Estimator," and "MM Estimator" approaches. These three weight functions are derived as functions of the residuals, leverage values, or both, with the goal of assigning weights that reduce the effect of unusual observations on the analysis. The three robust weight formulations are then explained in detail in the following sections.

The first type of weight is obtained from the derivative of a carefully selected smooth convex loss function $\emptyset(.)$ that is commonly used for M estimation, such as the Huber or Tukey's square functions. These loss functions are used to derive weights that reduce the influence of unusual observations on the analysis. Therefore, outlying observations have a low weight value. This weight was established as a result of the rescaled residuals. It assumes the form of,

$$w_j^m = \frac{\emptyset'\{[v_j-\widehat{m}(g_j)]/\hat{s}}{\{[v_j-\widehat{m}(g_j)]/\hat{s}} = \frac{\emptyset'(r_j^m)}{(r_j^m)} \text{ (5), j=1, 2…, n} \tag{5}$$

where v_i refers to dependent variable $\widehat{m}(g_i)$ represents the predicted result derived from local polynomial regression with weighting, $\hat{s}$ is the interquartile range of the residuals and r_j^m= $[v_j - \widehat{m}(g_j)]/\hat{s}$ is the rescaled residual linked to the valueg$_j$.When g_j is an atypical value, r_j^m is anticipated to be substantial, thus the weighting w_j^m will be concentrated.

Algorithm (1): M Kernel Estimator

Step 1. Introduced two distinct weight forms to develop a robust derivative estimator: one based on residual leverage and the other derived from Kernel smoothing to regulate smoothness, or a combination of both.

Step 2. To safeguard against noisy data influencing our estimate using local polynomial of the nonparametric function and its derivative, we employ various weights to manage outliers, high leverage and other forms of influential effects.

Step 3. Use the (Gaussian) kernel to determine the kernel weights for each value of g_i, $w_j^{k(t)}(g_i)$ and estimate the nonparametric function $\widehat{m}^t(g_i)$ and its derivative $\widehat{m}_t'(g_i)$ using $R_j(g) = w_j^{k(t)}(g_i) \times w_j^{k(t-1)}$ where $w_j^{k(t)}(g_i)$ are kernel weights at g_i $w_j^{k(t-1)}(g_i)$ are the M robust weight (i , j =1,2,…,n) at iteration t-1.

Step 4. To achieve convergence, increment t by 1 and repeat steps 2 and 3. One method to verify convergence involves calculating the correlation coefficient between estimates at iterations *t* and t−1. Stop when this coefficient is less than 0.99 otherwise, continue with steps 1 and 2 until this condition is met.

Disadvantages should mitigate the impact of outliers on estimating the nonparametric function and its derivative. Only, M-estimation encounters challenges in effectively estimating the derivative function when g values are isolated from the main cluster and the residual is minimal.

3. Proposed Estimators

After presenting the previous method and clarifying the weak points, two estimators were proposed to avoid the weak points, namely the S estimator and MM estimator, by relying on the first estimator using the rescaled residuals divided by a function of the leverage values and the second estimator modification of the classical kernel regression approach. Instead of basing the weights given to each observation on their proximity to the target point, the weights are determined by the distance of the observation from the regression surface.

S Kernel Estimator

The basic M-weight aims to mitigate the impact of outliers on estimating both the nonparametric function and its derivative. However, M-estimation struggles to accurately estimate the derivative function when residuals are small, and g values are significantly distant from the majority cluster. One potential solution involves using the residual standard deviation. This weight is defined as a function of the residual standard deviation, assigning a smaller weight to outlier observations. It is expressed as follows:

$$\mathrm{w}_\mathrm{j}^\mathrm{s} = \frac{\emptyset'\{[\mathrm{v_j}-\widehat{\mathrm{m}}(\mathrm{g_j})]/\sqrt{\frac{1}{nk}\sum_{i=1}^{n} w_i e_i^2}\}}{\{[\mathrm{v_j}-\widehat{\mathrm{m}}(\mathrm{g_j})]/\sqrt{\frac{1}{nk}\sum_{i=1}^{n} w_i e_i^2}\}} = \frac{\emptyset'(\mathrm{r_j^s})}{(\mathrm{r_j^s})}, \text{ i=1, …, n, j=1, …, k} \tag{6}$$

where v_j is the dependent variable $\widehat{\mathrm{m}}'(\mathrm{g_i})$ represents the estimated mean value obtained from local polynomial regression with weighting in $v_\mathrm{i} = m(\mathrm{g_j}) + \Sigma$, $\frac{1}{n}\sum_{i=1}^{n} w_i e_i$is the residual standard deviation, $\emptyset'(.)$is the function's derivative of $\emptyset(.)$ and $(\mathrm{r_j^s})= [\mathrm{v_j} - \widehat{\mathrm{m}}(\mathrm{g_j})]/\sqrt{\frac{1}{nk}\sum_{i=1}^{n} w_i e_i^2}$. When $\mathrm{g_j}$ is an unusual value, $\mathrm{r_j^s}$is expected to be large so the weight $\mathrm{w_j^s}$will be compact, two wight weight obtain kernel gaussian regression and obtain weight S- Robust estimation we can calculate the nonparametric function by defining $R_j(\mathrm{g_i}) = \mathrm{w_j^k}(\mathrm{g_i}) \times \mathrm{w_j^k}$, where $\mathrm{w_j^k}(\mathrm{g_i})$ are kernel weights at $\mathrm{g_i}\mathrm{w_j^k}(\mathrm{g_i})$ are the robust weights,

$$\widehat{\mathrm{m}}\mathrm{R_g}(\mathrm{g_i})= (\mathrm{g}^t\mathrm{R_g}\ \mathrm{g})\text{-1}\ \mathrm{g}^t\mathrm{R_g}\mathrm{v} \tag{7}$$

Algorithm (2): S Kernel Estimator

Step 1. To safeguard our estimation using a local polynomial for the nonparametric function and its derivative against noisy data, outlier values, high leverage, strongly impactful effects and other forms of disturbance, various weights are employed.

Step 2. Using $\mathrm{w_j^m}$ when iteration =1 and $\mathrm{w_j^s}$when iteration > 1

Step 3. Apply using the (Gaussian) kernel for computing the kernel weights for each value of $\mathrm{g_i}\mathrm{w_j^{k(t)}}(\mathrm{g_i})$, to estimate the nonparametric function $\widehat{\mathrm{m}}^\mathrm{t}(\mathrm{g_i})$ and its derivative, $\widehat{\mathrm{m}}_\mathrm{t}'(\mathrm{g_i})$, using $\mathrm{R_j}(\mathrm{g}) = \mathrm{w_j^{k(t)}}(\mathrm{g_i}) \times \mathrm{w_j^{s(t-1)}}$ where $\mathrm{w_j^{k(t)}}(\mathrm{g_i})$are kernel weights at $\mathrm{g_i}\ \mathrm{w_j^{s(t-1)}}(\mathrm{g_i})$ are the robust weights (i, j =1,2,…,n)at iteration t-1.

Step 4. Continue incrementing t by 1 and repeating steps 2 and 3 until convergence is achieved. One method to confirm convergence is by calculating the correlation coefficient between estimates at iterations t and t−1. Stop when this coefficient falls below 0.99 and continue with steps 1 and 2 until this condition is met.

MM Kernel Estimator

In kernel regression, the regression function is estimated by taking a weighted average of the observed data points within a neighborhood of a target point, where the weight given to each observation depends on its proximity to the target point. Outliers in the data can have a disproportionate influence on the estimated regression function if they are located within the neighborhood of a target point, as they will be given a large weight in the computation of the weighted average.

The MM estimator of robustness can be seen as a modification of the classical kernel regression method, where the weights given to each observation in the computation of the weighted average are based on the distance of the observation from the regression surface, rather than its proximity to the target point. This modification helps to enhance the robustness of the estimated regression function, as it reduces the impact of outliers on the estimated regression function.

Algorithm (3): MM Kernel Estimator

Step 1. An initial robust regression estimator is used to estimate the residual scale. This initial estimator may be inefficient, but it should have a high breakdown point.

Step 2. A regression M−estimator is computed using a bounded loss function and standardized residuals. The final estimator will retain the high breakdown of the initial one, but its efficiency is improved by the use of an appropriate loss function.

Step 3. Continue incrementing t by 1 and repeating steps 2 and 3 until convergence is achieved. One method to confirm convergence is by calculating the correlation coefficient between estimates at iterations t and t−1. Stop when this coefficient falls below 0.99 and continue with steps 1 and 2 until this condition is met.

Next, a comparison is made between the three estimators (M kernel estimator, S kernel estimator, and MM kernel estimator) mentioned above through simulation and real data, and the results obtained are clarified.

4. Monto Carlo Simulation Study

In this section, the Monte Carlo study has been designed to compare the performances of different estimators for robust nonparametric. The comparison between robust estimators (M, S, MM) and nonparametric estimator (Kernel) for different sample sizes and percentages of outliers. R software is used to perform our Monte Carlo simulation study based on "*nprobust*" package in R [19], [20]. For further information on how to make Monte Carlo simulation studies using R, see [21], [22].

The simulated model is carried out based on $v_{\mathrm{i}} = m(\mathrm{g_i}) + \varepsilon_i$ in the simulation study.

Table 1. Simulation factor

Factor	Notation	Values
Sample size	n	50,70,100,150,200, 300, and 500
Percentages of outliers	outliers%	5,10,20, and 30

According to Table 1, we performed the simulation experiments with varying values of n and percentages of outliers as follows:

- The values of (g) variable were generated from uniform (0, 1), and calculated m$(g) = g$+$2e^{-16g^2}$, see Figure 1.

- The error term of the model is generated from normal (0, 0.03)

- The estimated Mean Squared Error (MSE) and Mean Absolute Error (MAE) for each estimator are calculated as follows:

$$\text{MSE} = \frac{1}{1000}\sum_{l=1}^{1000}\left(\frac{1}{n}\sum_{i=1}^{n}(\widehat{m}(g_i) - m(g_i))^2\right)_l, \tag{8}$$

$$\text{MAE} = \frac{1}{1000}\sum_{l=1}^{1000}\left(\frac{1}{n}\sum_{i=1}^{n}|(\widehat{m}(g_i) - m(g_i)|\right)_l. \tag{9}$$

where $(\widehat{\mathrm{m}}(\mathrm{g_i}) - \mathrm{m}(\mathrm{g_i}))$ is the difference between estimated and true values. Also, the Relative Efficiency (RE) for each robust estimator is calculated as

$$\text{RE}=\frac{\text{MSE}(\widehat{\mathrm{m}}(\mathrm{g}))^K}{\text{MSE}(\widehat{\mathrm{m}}(\mathrm{g}))^\pi}; \pi = 1,2,3. \tag{10}$$

where π stands for the three robust (M, MM, S) estimators, and $\text{MSE}(\widehat{\mathrm{m}}(\mathrm{g}))^K$ is the MSE for the Kernel estimator.

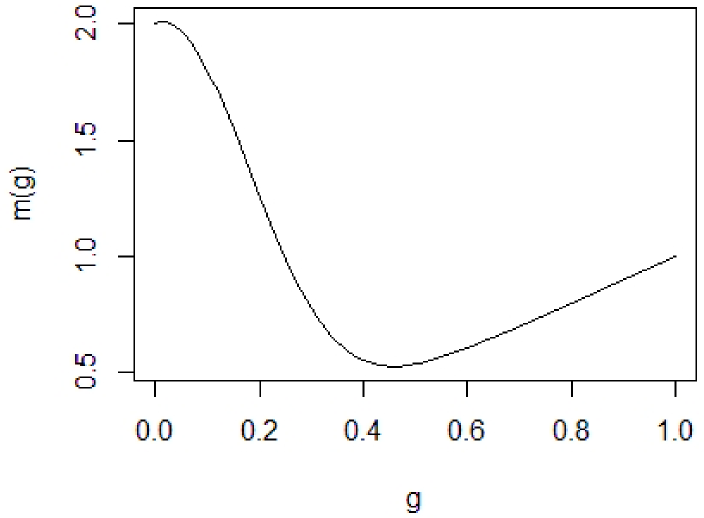

Figure 1 density function of m(g) = g+$2e^{-16g^2}$.

The results of simulation are recorded in Tables 2 to 4. These tables present the of MSE, MAE and RE for each estimator (Kernel, M Kernel, S Kernel, and MM Kernel), respectively, under different sample sizes (n = 50 to 500) and outlier percentages (5% to 30%).

From the simulation results, we can note that:

1. Performance Based on MSE (Table 2)

- The M Kernel consistently outperforms the other estimators across all sample sizes and outlier percentages, demonstrating the lowest MSE values. For example, at n = 50 and 5% outliers, the MSE for M Kernel (0.0131) is significantly lower than that of the standard Kernel (0.0267). This trend holds even higher outlier percentages (e.g., 30%), demonstrating the robustness of the M Kernel.
- The S Kernel and MM Kernel show competitive performance, often yielding lower MSE values compared to the standard Kernel. However, their performance is slightly inferior to the M Kernel in most scenarios.
- M Kernel exhibits the best performances among the estimators for all sample sizes and percentages of outliers, in terms of the smaller MSEs. At n = 50, and for 5% of outliers, for instance, MSE when using M Kernel (0.0131) is much smaller than that for the normal Kernel (0.0267). This observation applies to even larger outlier percentages (e.g., 30%), indicating the stability of the M Kernel.
- The S Kernel and MM Kernel are competitive with the standard Kernel and typically result in smaller MSE values. But in most cases, their performance is worse than the M Kernel.
- As expected, increasing the sample size (n) reduces the MSE for all estimators, indicating improved estimation accuracy with larger n. For example, at 5% outliers, the MSE for the M Kernel decreases from 0.0131 (n = 50) to 0.001 (n = 500).

Table 2. Mean Squared Error (MSE) values for different estimators.

Outlier %	**n**	**Kernel**	**M Kernel**	**S Kernel**	**MM Kernel**
5	**50**	0.0267	0.0131	0.0151	0.0145
	75	0.0181	0.0083	0.0094	0.0093
	100	0.0146	0.0067	0.0076	0.0077
	150	0.0096	0.0038	0.0041	0.0042
	200	0.0066	0.0031	0.0033	0.0034
	300	0.0046	0.0018	0.002	0.002
	500	0.0024	0.001	0.0012	0.0011
10	**50**	0.0487	0.0326	0.0361	0.037
	75	0.03	0.0194	0.0192	0.0233
	100	0.0228	0.0143	0.0144	0.0172
	150	0.0162	0.0096	0.0098	0.0118
	200	0.0105	0.0065	0.0065	0.0076
	300	0.0072	0.004	0.0041	0.0047
	500	0.0037	0.0021	0.0021	0.0023
20	**50**	0.0916	0.0729	0.0728	0.0761
	75	0.0595	0.0475	0.0463	0.0497
	100	0.0412	0.0345	0.0356	0.0388
	150	0.0292	0.0216	0.0216	0.024
	200	0.0207	0.0153	0.0145	0.0168
	300	0.0134	0.0093	0.009	0.0106

	500	0.0069	0.0049	0.0047	0.0059
30	50	0.1461	0.1288	0.1293	0.1243
	75	0.0936	0.0803	0.078	0.0755
	100	0.0693	0.0641	0.0635	0.0632
	150	0.0455	0.0366	0.0352	0.035
	200	0.0337	0.0267	0.0258	0.0253
	300	0.0219	0.0174	0.0161	0.0157
	500	0.0108	0.0083	0.0082	0.0081

Table 3. Mean Absolute Error (MAE) values for different estimators.

Outlier %	N	Kernel	M Kernel	S Kernel	MM Kernel
5	50	0.1265	0.0862	0.0888	0.0893
	75	0.1029	0.0701	0.0706	0.0744
	100	0.0936	0.0619	0.0649	0.0653
	150	0.0757	0.0477	0.0484	0.05
	200	0.0634	0.0422	0.0428	0.0441
	300	0.0538	0.0334	0.0347	0.0356
	500	0.039	0.0252	0.0265	0.0265
10	50	0.1685	0.1325	0.1352	0.1399
	75	0.1331	0.1049	0.1025	0.1146
	100	0.117	0.0899	0.0901	0.0987
	150	0.0985	0.0736	0.0734	0.0814
	200	0.078	0.0609	0.0609	0.0656
	300	0.0664	0.0485	0.0477	0.0529
	500	0.0483	0.0359	0.0359	0.0381
20	50	0.2316	0.2031	0.2009	0.2055
	75	0.1845	0.1636	0.1616	0.1688
	100	0.1575	0.1393	0.1401	0.1461
	150	0.1311	0.1101	0.1091	0.1152
	200	0.1087	0.0939	0.0911	0.0993
	300	0.0904	0.0749	0.0718	0.079
	500	0.0645	0.0541	0.0518	0.0596
30	50	0.2899	0.2685	0.2672	0.2614
	75	0.232	0.2157	0.2087	0.2067
	100	0.2035	0.1888	0.1876	0.1871
	150	0.1628	0.1421	0.1389	0.1388
	200	0.1374	0.1234	0.1194	0.1178
	300	0.1142	0.1	0.095	0.0943
	500	0.08	0.0698	0.0683	0.0682

2. Performance Based on MAE (Table 3)

- The same as that observed for the MSE, the M kernel obtains the smallest absolute error value in almost all test cases, which reinforces the robustness of the method. For instance, at n = 100 and 10% outliers, the value of MAE for M Kernel (0.0899) is much lower than that of standard Kernel (0.117).

- The S Kernel and MM Kernel are also similar, having MAE similar to the M Kernel especially at the larger n. This indicates that also minimizing absolute
- The standard Kernel exhibits the highest MAE values, especially under high outlier contamination (e.g., 30%), highlighting its sensitivity to outliers.

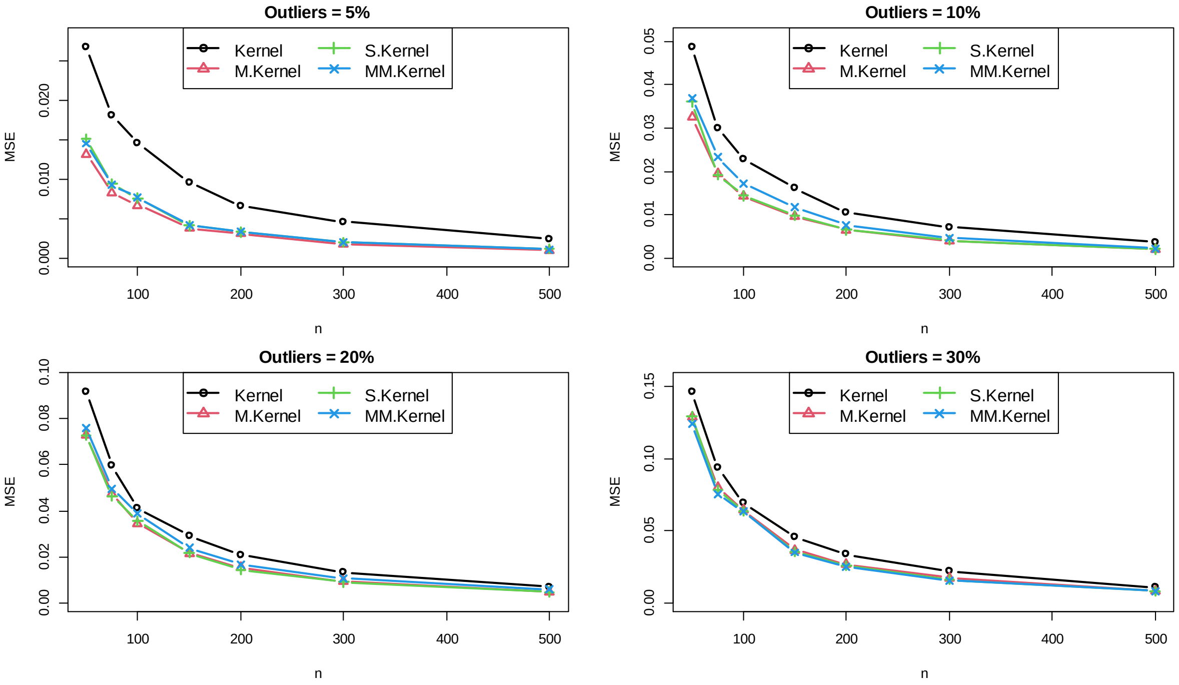

Figure 2. MSE for robust and non-robust estimators

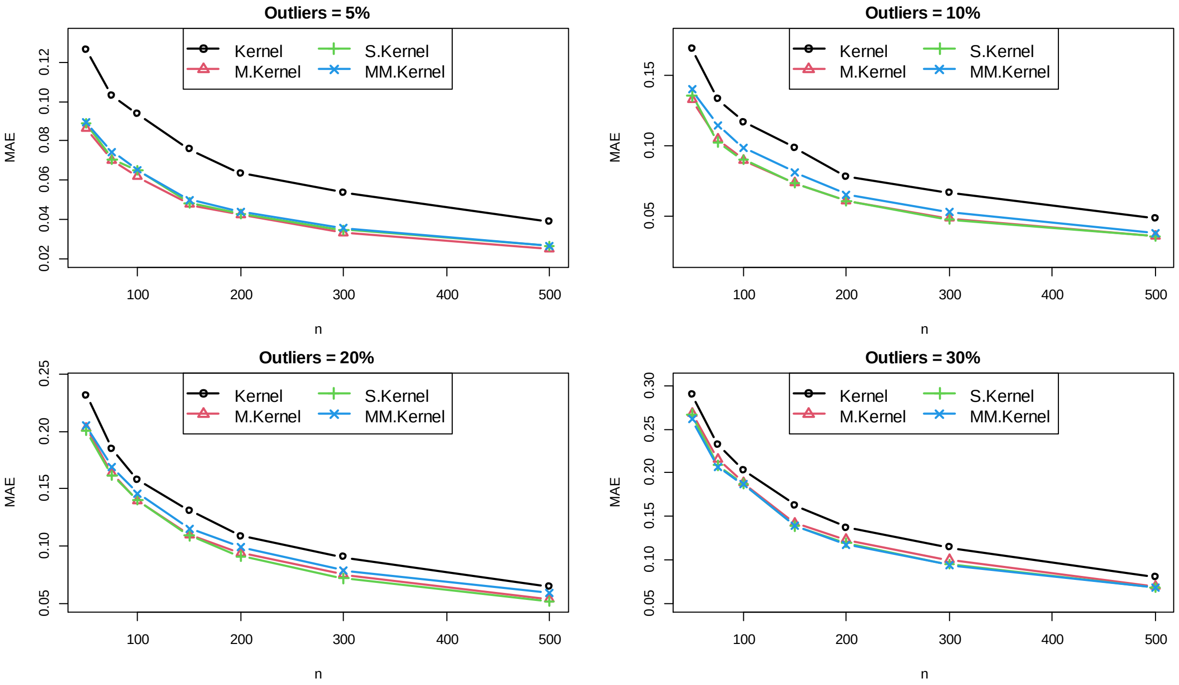

Figure 3. MAE for robust and non-robust estimators

The graphical representations (Figures 2 and 3) visually reinforce the trends observed in tables 2 and 3, showing a clear decline in MSE and MAE as sample size increases, with the M Kernel consistently positioned as the top performer.

3. Relative Efficiency (RE) (Table 4)

- The RE values once again corroborate that M Kernel is better (comment: precedence across the scenarios). For example, for n = 50 and 5 % outliers we see that the RE of the M Kernel (2:0382) is more than double the efficiency of the raw Kernel. This trend is observed also for larger percentages of outliers, but at the cost of a reduced improvement in efficiency (e.g., ARE = 1.1343 at n = 50 and 30% outliers).
- The S Kernel and MM Kernel also show RE values greater than 1 in most cases, indicating their efficiency relative to the standard Kernel. However, their RE values are generally lower than those of the M Kernel, particularly for smaller sample sizes and higher outlier percentages.

Table 4. Relative Efficiency (RE) values for different estimators.

Outlier %	N	M Kernel	S Kernel	MM Kernel
5	**50**	2.0382	1.7682	1.8414
	75	2.1807	1.9255	1.9462
	100	2.1791	1.9211	1.8961
	150	2.5263	2.3415	2.2857
	200	2.1290	2.0000	1.9412
	300	2.5556	2.3000	2.3000
	500	2.4000	2.0000	2.1818
10	**50**	1.4939	1.3490	1.3162
	75	1.5464	1.5625	1.2876
	100	1.5944	1.5833	1.3256
	150	1.6875	1.6531	1.3729
	200	1.6154	1.6154	1.3816
	300	1.8000	1.7561	1.5319
	500	1.7619	1.7619	1.6087
20	**50**	1.2565	1.2582	1.2037
	75	1.2526	1.2851	1.1972
	100	1.1942	1.1573	1.0619
	150	1.3519	1.3519	1.2167
	200	1.3529	1.4276	1.2321
	300	1.4409	1.4889	1.2642
	500	1.4082	1.4681	1.1695
30	**50**	1.1343	1.1299	1.1754
	75	1.1656	1.2000	1.2397
	100	1.0811	1.0913	1.0965
	150	1.2432	1.2926	1.3000
	200	1.2622	1.3062	1.3320
	300	1.2586	1.3602	1.3949
	500	1.3012	1.3171	1.3333

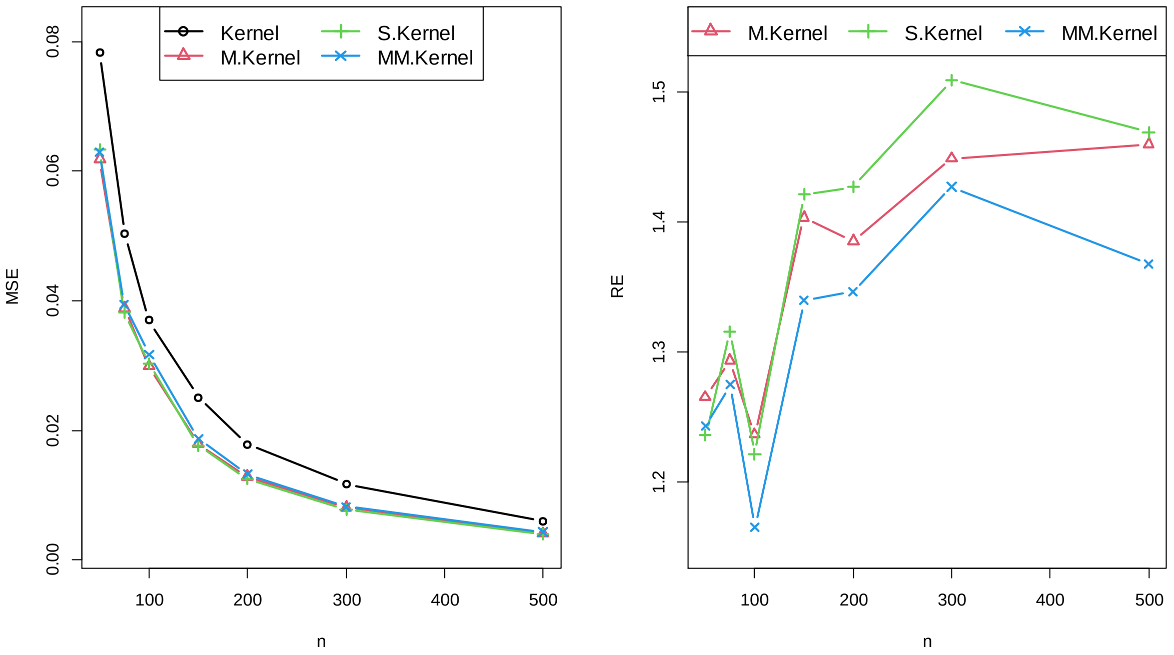

Figure 4 MSE and RE for different estimators of m(g) categorized by sample size levels.

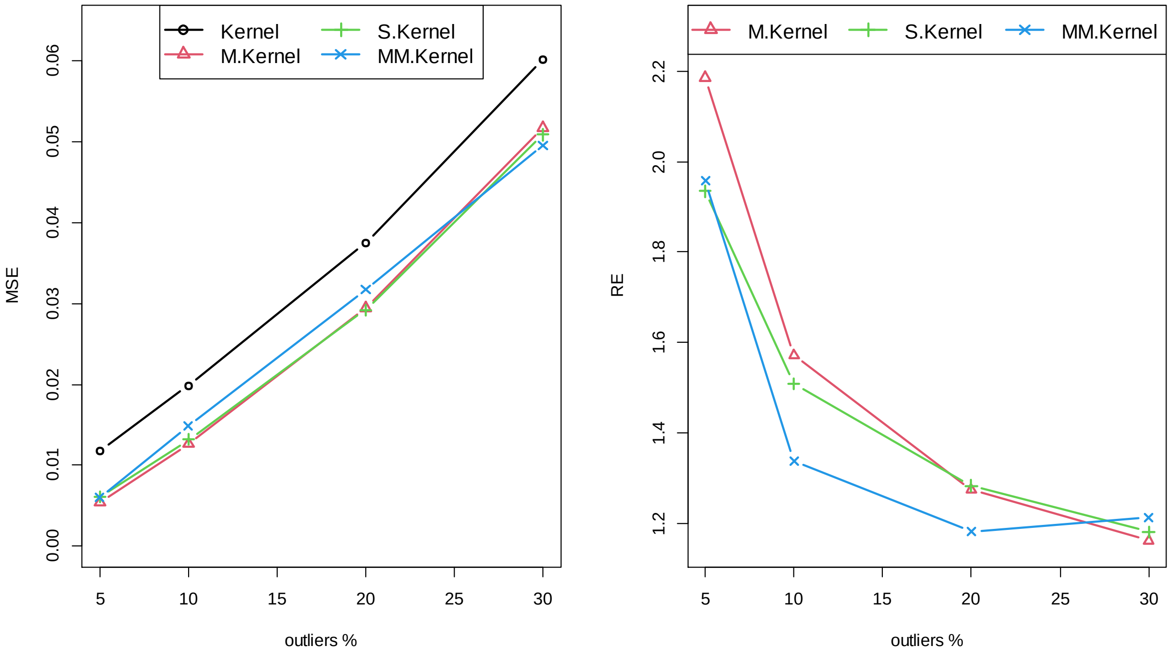

Figure 5 MSE and RE for different estimators of m(g) categorized by the proportion of outliers.

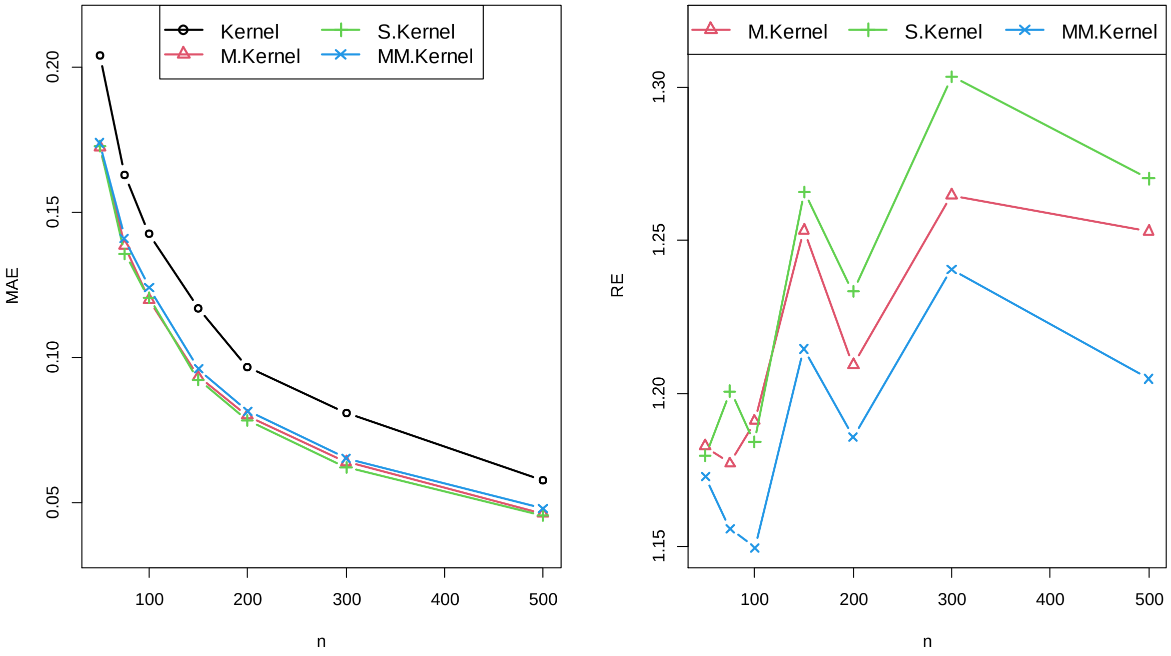

Figure 6 MAE and RE for different estimators of m(g) categorized by sample size levels.

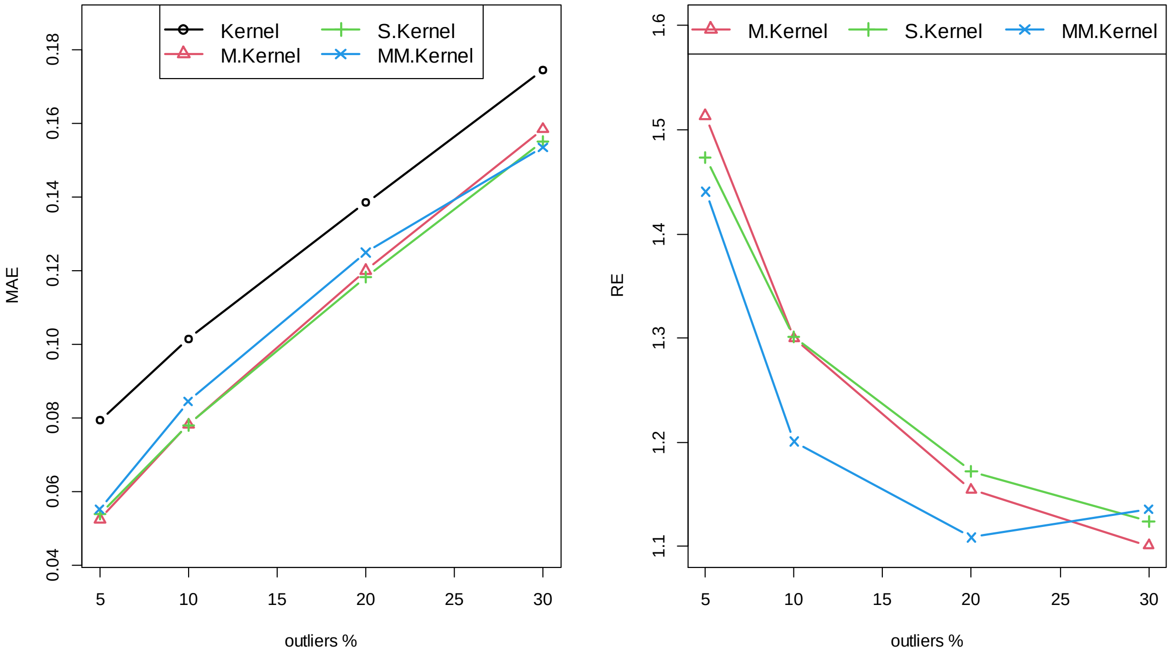

Figure 7 MAE and RE for different estimators of m(g) categorized by the proportion of outliers.

The figures (Figures 4–7) provide a visual representation of the simulation results, comparing the performance of the Kernel, M Kernel, S Kernel, and MM Kernel estimators under varying sample sizes and outlier proportions. Below is a discussion of the key insights derived from these figures:

1. Performance by Sample Size (Figures 4 and 6)

- For all estimators the MSE (Figure 4) and MAE (Figure 6) decrease as the sample size (n) grows. This is consistent with statistical theory that one expects estimation accuracy to improve with larger sample sizes.
- The M Kernel is the winner across most sample sizes, and reaches the smallest values of MSE and MAE, so we confirm that it is the most robust and the most efficient. However, in the case of large samples, the S Kernel is the best.
- The standard Kernel estimator performs the worst, particularly for smaller sample sizes (e.g., n = 50), where its MSE and MAE are significantly higher than those of the robust estimators.

2. Performance by Outlier Proportion (Figures 5 and 7)

- The standard Kernel estimator performs poorly in the presence of attitude anomaly, as it can be seen from significant MSE (Figure 5) and MAE (Figure 7) increase with an outlier ratio increases from 5% to 30%. This supports its sensitivity toward the contaminated in data.
- The M Kernel is highly robust with very little degradation of error metrics even in the presence of 30% outliers. This reflects its resilience to outlying data points.
- The S Kernel and MM Kernel also show insensitivity to outliers but are more sensitive than the M Kernel, especially when the contamination level is high.
- The RE is high for even the M Kernel for all outlier proportions, however the efficiency advantage reduces as we go towards extreme contamination (30%). This implies that even the M Kernel is weak to very high levels of outliers.
- The S Kernel and MM Kernel exhibit some efficiency gains over the standard Kernel but are not competitive with the M Kernel.

The visualizations reinforce the simulation findings from Tables 2–4, providing an intuitive understanding of how sample size and outlier proportions influence estimator performance.

5. Real-Life Application

To confirm the results obtained through comparison between the proposed estimators using simulation, it was applied to the data wine industry. Wine certification and quality assessment are key elements within this context. Certification prevents the illegal adulteration of wines (to safeguard human health) and assures quality for the wine market. Quality evaluation is often part of the certification process and can be used to improve winemaking (by identifying the most influential factors) and to stratify wines such as premium brands (useful for setting prices) see [23]. The data set available in https://archive.ics.uci.edu/ml/datasets/wine+quality.

Table 5. descriptive data of Quality and Alcohol variables

Variable	Min	Q1	Median	Mean	Q3	Max
Quality	1.00	5.00	6.00	5.63	6.00	8.00
Alcohol	8.40	9.30	10.20	10.44	11.10	14.19

Analyzing 1143 observations and examining the data revealed a noticeable disparity between the highest and lowest values, indicating the presence of outliers. To validate the results derived from the data description, a boxplot was formed as shown in Figure 8.

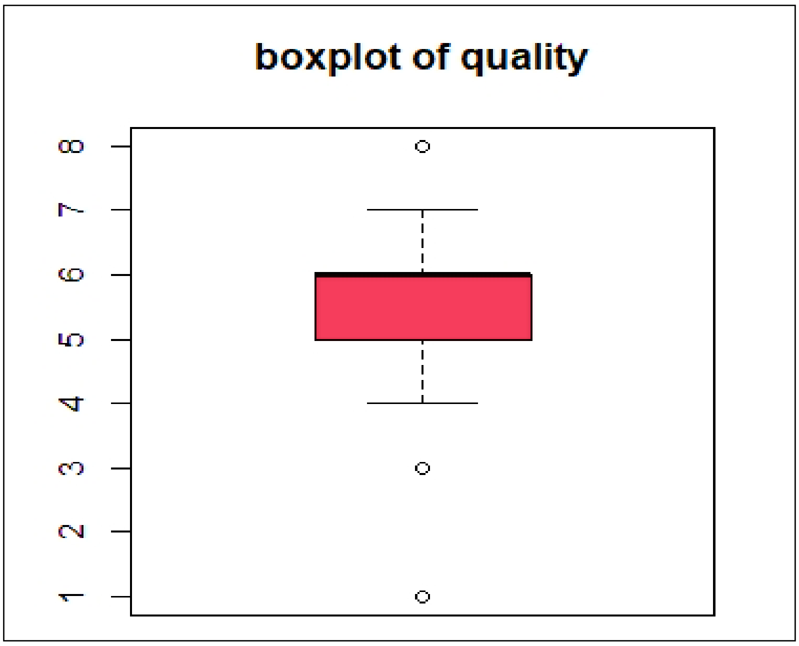

Figure 8 Boxplot for quality (independent variable)

Additionally, the low coefficient of determination prompted us to utilize nonparametric as shown in Table 6 following the residual summary and Figure 9.

Table 6. descriptive statistics of the residuals

Statistic	Min.	Q1	Median	Q3	Max.
Value	-6.09	-0.38	0.13	0.53	2.61

Outliers, high-leverage points, and highly influential points are frequently present in Wine data, whether within the data range or at the boundaries. These points can significantly impact derivative estimation and obscure the true underlying pattern, highlighting the need for a robust estimator. Current methods used to analyze Wine data often lack robustness in addressing these challenges.

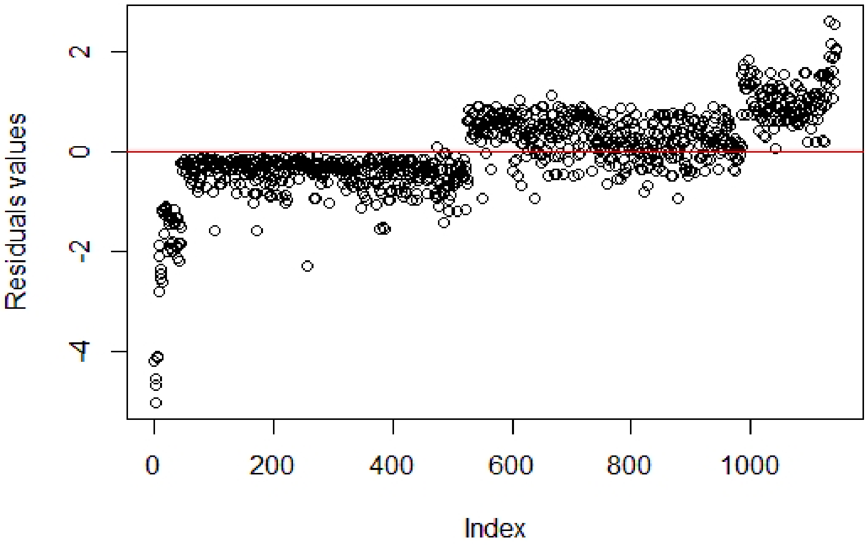

Figure 9 scatter plot for residual

We are focused on employing a robust estimator for the derivative and utilizing it to identify potential change points by estimating the derivative function, using performance measures to compare the different proposed capabilities, we obtained the following results MSE and MAE for (Kernel, M Kernel, S Kernel and MM Kernel) as shown in Table7.

Table 7. MSE and MAE for different estimators of the dataset

Measure	Kernel	M Kernel	S Kernel	MM Kernel
MSE	0.0090	0.0083	0.0082	0.0080
MAE	0.1000	0.099	0.089	0.087

In Table 7, goodness of fit measures of the Kernel, M Kernel, S Kernel, and MM Kernel estimators are presented. The best estimator for this data is MM Kernel because it has the minimum MSE and MAE values.

6. Conclusion

We find that the S Kernel and MM Kernel estimators proposed in this paper are better than the standard Kernel estimator in nonparametric regression with outlier-contaminated data and have the smallest MSE and MAE as well as the highest relative efficiency for all different sample sizes and the different proportion of outliers. The findings, which are backed up by a simulation study and a real data application, demonstrate their robustness and stability, even in highly contaminated situations. In contrast, the standard Kernel estimator is highly sensitive to outliers. These findings advocate for the adoption of robust estimators in practical applications where data integrity is uncertain, offering a reliable solution for accurate and efficient regression estimation. In future work, we can develop our robust estimators on nonparametric count panel data models as an extension of [24] or develop our robust estimators on semiparametric generalized linear models as an extension of [25].

Conflicts of Interest: The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] É.A. Nadaraya, On Non-Parametric Estimates of Density Functions and Regression Curves. Theory Probab. Appl. 1965, 10 (1), 186–190. https://doi.org/10.1137/1110024.

[2] G. Watson, Smooth Regression Analysis, Sankhyā Ser. A 26 (1964), 359-372.

[3] T. Gasser, H.G. Müller, Estimating Regression Functions and Their Derivatives by the Kernel Method, Scand. J. Stat. (1984), 171-185. https://www.jstor.org/stable/4615954.

[4] P. Hall, R.C.L. Wolff, Q. Yao, Methods for Estimating a Conditional Distribution Function, J. Amer. Stat. Assoc. 94 (1999), 154–163. https://doi.org/10.1080/01621459.1999.10473832.

[5] J. Fan, I. Gijbels, Local Polynomial Modelling and Its Applications, CRC Press, Boca Raton, 1996.

[6] S. Meshaal El-sayed, M. Reda Abonazel, M. Metwally Seliem, B-spline Speckman Estimator of Partially Linear Model, Int. J. Syst. Sci. Appl. Math. 4 (2019), 53-59. https://doi.org/10.11648/j.ijssam.20190404.12.

[7] M.R. Abonazel, N. Helmy, A. Azazy, The Performance of Speckman Estimation for Partially Linear Model Using Kernel and Spline Smoothing Approaches, Int. J. Math. Arch. 10 (2019), 10-18.

[8] M. Abonazel, A. Rabie, The Impact of Using Robust Estimations in Regression Models: An Application on the Egyptian Economy, J. Adv. Res. Appl. Math. Stat. 4 (2019), 8-16.

[9] M. M. Elgohary, M. R. Abonazel, N. M. Helmy, A. R. Azazy, New Robust-Ridge Estimators for Partially Linear Model, Int. J. Appl. Math. Res. 8 (2019), 46-52. https://doi.org/10.14419/ijamr.v8i2.29932.

[10] W.S. Cleveland, Robust Locally Weighted Regression and Smoothing Scatterplots, J. Am. Stat. Assoc. 74 (1979), 829-836. https://doi.org/10.1080/01621459.1979.10481038.

[11] B.W. Silverman, Some Aspects of the Spline Smoothing Approach to Non-Parametric Regression Curve Fitting, J. R. Stat. Soc. Ser. B: Stat. Methodol. 47 (1985), 1-21. https://doi.org/10.1111/j.2517-6161.1985.tb01327.x.

[12] J. Fan, P. Hall, On Curve Estimation by Minimizing Mean Absolute Deviation and Its Implications, Ann. Stat. 22 (1994), 867-885. https://doi.org/10.1214/aos/1176325499.

[13] J.W. Tukey, Exploratory Data Analysis, Addison-Wesley, Reading, 1977.

[14] I.R. Ghement, M. Ruiz, R. Zamar, Robust Estimation of Error Scale in Nonparametric Regression Models, J. Stat. Plan. Inference 138 (2008), 3200-3216. https://doi.org/10.1016/j.jspi.2008.01.005.

[15] H.F.F. Mahmoud, B. Kim, I. Kim, Robust Nonparametric Derivative Estimator, Commun. Stat. - Simul. Comput. 51 (2020), 3809-3829. https://doi.org/10.1080/03610918.2020.1722836.

[16] G. Wang, F. Manhardt, F. Tombari, X. Ji, GDR-Net: Geometry-Guided Direct Regression Network for Monocular 6D Object Pose Estimation, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Nashville, TN, USA, 2021: pp. 16606–16616. https://doi.org/10.1109/CVPR46437.2021.01634.

[17] B. Zhu, J. Jiao, M.I. Jordan, Robust Estimation for Non-Parametric Families via Generative Adversarial Networks, in: 2022 IEEE International Symposium on Information Theory (ISIT), IEEE, Espoo, Finland, 2022: pp. 1100–1105. https://doi.org/10.1109/ISIT50566.2022.9834844.

[18] M. Salibian-Barrera, Robust Nonparametric Regression: Review and Practical Considerations, Econ. Stat. 4 (2023), 4. https://doi.org/10.1016/j.ecosta.2023.04.004.

[19] M.R. Abonazel, A Practical Guide for Creating Monte Carlo Simulation Studies Using R, Int. J. Math. Comput. Sci. 4 (2018), 18-33.

[20] A. Kamel, M.R. Abonazel, A Simple Introduction to Regression Modeling Using R, Comput. J. Math. Stat. Sci. 2 (2023), 52-79. https://doi.org/10.21608/cjmss.2023.189834.1002.

[21] M.D. Cattaneo, M. Jansson, X. Ma, Lpdensity: Local Polynomial Density Estimation and Inference, arXiv:1906.06529 (2019). http://arxiv.org/abs/1906.06529.

[22] S. Calonico, M.D. Cattaneo, M.H. Farrell, Nprobust: Nonparametric Kernel-Based Estimation and Robust Bias-Corrected Inference, arXiv:1906.00198 (2019). http://arxiv.org/abs/1906.00198.

[23] P. Cortez, A. Cerdeira, F. Almeida, T. Matos, J. Reis, Modeling Wine Preferences by Data Mining from Physicochemical Properties, Decis. Support Syst. 47 (2009), 547-553. https://doi.org/10.1016/j.dss.2009.05.016.

[24] M.R. Abonazel, A.H. Youssef, E.G. Ahmed, On Robust M, S and Mm Estimations for the Poisson Fixed Effects Panel Model with Outliers: Simulation and Applications, J. Stat. Comput. Simul. 95 (2025), 1190-1207. https://doi.org/10.1080/00949655.2024.2449102.

[25] M.M. Seliem, S.M. El-Sayed, M.R. Abonazel, Developing a Semi-Parametric Zero-Inflated Beta Regression Model Using P-Splines: Simulation and Application, Stat. Optim. Inf. Comput. 13 (2024), 1103-1119. https://doi.org/10.19139/soic-2310-5070-2220.