Bootstrap-t Confidence Interval on Local Polynomial Regression Prediction

Abil Mansyur, Elmanani Simamora*

Department of Mathematics, Universitas Negeri Medan, Medan, 20221, North Sumatera, Indonesia

Received August 11, 2022; Revised October 7, 2022; Accepted October 25, 2022

Cite This Paper in the Following Citation Styles

(a): *[1] Abil Mansyur, Elmanani Simamora , "Bootstrap-t Confidence Interval on Local Polynomial Regression Prediction," Mathematics and Statistics, Vol. 10, No. 6, pp. 1178 - 1193, 2022. DOI: 10.13189/ms.2022.100604.*

(b): *Abil Mansyur, Elmanani Simamora (2022). Bootstrap-t Confidence Interval on Local Polynomial Regression Prediction. Mathematics and Statistics, 10(6), 1178 - 1193. DOI: 10.13189/ms.2022.100604.*

Abstract In local polynomial regression, prediction confidence interval estimation using standard theory will give coverage probability close to exact coverage probability. However, if the normality assumption is not met, the bootstrap method makes it possible to apply it. The working principle of the bootstrap method uses the resampling method where the sample data becomes a population and there is no need to know the distribution of the sample data is normal or not. Indiscriminate selection of smoothing parameters allows scatterplot results from local polynomial regressions to be rough and can even lead to misleading statistical conclusions. It is necessary to consider the optimal smoothing parameters to get local polynomial regression predictions that are not overfitting or underfitting. We offer two new algorithms based on the nested bootstrap resampling method to determine the bootstrap-t confidence interval in predicting local polynomial regression. Both algorithms consider the search for optimal smoothing parameters. The first algorithm performs paired and residual bootstrap samples, and the second algorithm performs based on residuals with residuals. The first algorithm provides a scatterplot and reasonable coverage probability on relatively large sample data. In contrast, the second algorithm is more powerful for each data size, including for relatively small sample data sizes. The mean of the bootstrap-t confidence interval coverage probability shows that the second algorithm for second-degree local polynomial regression is better than the other three. However, the larger the sample data size gives, the closer the closer the average coverage probability of the two algorithms is to the nominal coverage probability.

Keywords Bootstrap, Confidence Interval, Coverage Probability, Algorithm, Simulation

1. Introduction

Bootstrapping is a resampling method with good performance for estimating interest statistics in nonparametric regression models. One is estimating the confidence interval of local polynomial nonparametric regression prediction. The advantage of the bootstrap method is that it can calculate confidence intervals without using the assumptions of standard theory [1]. The problem is that standard confidence intervals (normal distribution theory) based on an asymptotic approach can be wildly inaccurate in practice [2].

The literature [1-5] presents several methods of estimating second-order non-exact confidence intervals where samples are drawn using the resampling method. In general, there are two methods for bootstrap sampling: paired and residual bootstrap (see [1]). The principle of sampling drawing in both ways is based on parametric and nonparametric. Parametric is based on a selection from a particular distribution, while nonparametric is based on empirical distribution, and parametric properties are more exact than nonparametric properties [1]. However, there are data whose distribution is unknown so we can draw statistical conclusions based on nonparametric data.

Bootstrap-t interval is a method for estimating certain statistical confidence intervals, widely applied in various fields of science related to statistical studies. As in engineering, Zoubir [6] uses bootstrap-t intervals to determine interval estimates in signal processing applications such as radar, sonar and telecommunications. Jung [7] conducted a comparative study of three bootstrap interval estimation methods, including the bootstrap-t interval, in Generalized Structured Component Analysis (GSCA) which is widely used in economics and management. Next, Manly [8] applied the bootstrap-t interval to cases in biologies, such as survival and growth data

Several researchers, see [9-12], have applied the bootstrap method to estimate the statistical confidence interval of interest in local polynomial regression. This study reviews the literature [13-15] to obtain a new algorithm for bootstrap-t intervals. The bootstrap prediction interval provides a coverage probability close to nominal coverage in a small sample without assumptions about the sampling distribution [13]. Polansky [14] revealed theoretical and empirical evidence showing that the bootstrap-t interval reasonably solves the problem of estimating nonparametric statistical intervals. However, the bootstrap-t prediction interval can cause bands to be widely and exhibit unstable behaviour. DiCiccio [2] stated that in relatively small data, the probability behaviour of the bootstrap-t interval coverage tends to be a conservative interval with superior coverage probability and highly variable. However, the simulation results [14] in section 3 pp. 506-513 counter statement [2]. Two methods are offered [14] to stabilise the bootstrap-t interval: the smoothed bootstrap method and adding a constant to the empirical variance. Then, the recent literature [15] discusses the prediction limits of bootstrap on local polynomial regression. They offer a new algorithm using nested residual bootstrap resampling to predict the local polynomial regression prediction band boundary on the ecological response given by the predictor. Unfortunately, the scatterplot provides piecewise points that are not smooth, and the coverage probability is superior coverage for small data.

This article offers a new algorithm for estimating the LPR prediction interval based on the diversity factor of the bootstrap resampling distribution. The bootstrap-t prediction interval is constructed based on two bootstrap sampling processes. This sampling process is known in bootstrap method terminology as nested resampling. The organisation of this article is as follows. Section 1 presents the background of this research. Section 2 summarises the theories and concepts related to low-order local polynomial regression and provides a detailed description of the new algorithm. In this section, Section 3 presents the simulation results for the generation of large and small samples by considering the noise variance. The last section provides conclusions and suggestions for further research.

2. Materials and Methods

This section summarises the theory and concept of local polynomial regression. It then describes the bootstrap-t interval algorithm applied to the prediction of local polynomial regression. There are two algorithms offered based on nested bootstrap resampling. The first algorithm uses the principle of paired and residual bootstrap resampling. The second algorithm is only based on the residual bootstrap method. Both algorithms aim to stabilise the standard error that comes from sampling diversity.

2.1. Local Polynomial Regression

Local Polynomial Regression (LPR) is a nonparametric regression model that aims to obtain a smoother scatterplot of the relationship between the response variable and the dependent variable. The working principle is to smooth the curve using local weighted least squares from a certain point. The smoothing depends on two parameters: the smoothing parameter and the degree of the polynomial *p*. Smoothing parameter *α* determines the number of neighbouring points around the point of interest. At the same time, the degree of the polynomial *p* is the highest power of the predictor. Usually, low polynomial degrees are chosen, namely one and two.

The local polynomial regression model is expressed in the form,

$$y(x_{\mathrm{i}}) = f(x_{\mathrm{i}}) + \varepsilon_{\mathrm{i}},\ i = 1,2,...,n, \tag{1}$$

where the smooth function $f(x_{\mathrm{i}})$ is unknown but estimated. Variable x_{i} is an independent variable or predictor that is independent with an error ε_{i}. Meanwhile, the dependent variable $\mathrm{y}(x_{\mathrm{i}})$ is the response variable from the model (1). The random variable εi is independently identically distributed, having a mean of zero, $\mathrm{E}(\varepsilon_{\mathrm{i}}) = 0$ and constant variance, $\mathrm{Var}(\varepsilon_{\mathrm{i}}) = \sigma^2$.

Suppose that at the point x_0, the function f has a derivative to the degree of the polynomial *p*, then the function $f(x_{\mathrm{i}})$ is an approximation of the Taylor expansion at the point of interest x_0 (see [16]),

$$\begin{aligned} f(x_i) \approx f(x_0) + f^{'}(x_0)(x_{\mathrm{i}} - x_0) + \frac{f^{''}(x_0)}{2!}(x_{\mathrm{i}} - x_0)^2 + \\ \cdots + \frac{f^{p}(x_0)}{2!}(x_{\mathrm{i}} - x_0)^p \equiv \sum_{j=0}^{p} \beta_j (x_{\mathrm{i}} - x_0)^j. \end{aligned} \tag{2}$$

Point x_{i} is a neighbouring point of x_0, and the notation for points around x_0 is denoted by $\mathbb{N}(x_0)$. The parameter *α* is a smoothing parameter whose value is between zero and one. The agreement will determine $k = [n\alpha]$, the number of points on the neighbours $\mathbb{N}(x_0)$. The polynomial match in (2) uses locally weighted least squares by minimising,

$$\min_{\beta_j} \sum_{i=1}^{k} w_i(x_0) \left\{ y(x_{\mathrm{i}}) - \sum_{j=0}^{p} \beta_j (x_{\mathrm{i}} - x_0)^j \right\}^2. \tag{3}$$

The weight value $w_i(x_0)$ is obtained from the function $w_i(x_0) = \mathrm{W}(|x_0 - x_{\mathrm{i}}|/\Delta(x_0))$ where $\Delta(x_0)$ is the maximum of the absolute distance for a point $x_{\mathrm{i}} \in \mathrm{N}(x_0)$ to point x_0. The W function is a weight function that researchers usually choose, which has the following properties [17],

1. $W(u) > 0$, for $-1 < u < 1$;
2. $W(-u) = W(u)$;
3. $W(u)$ is a non-increasing function for $u \geq 0$;
4. $W(u) = 0$, for $u \leq -1$ or $u \geq 1$.

Cleveland [17] chooses the tricube weight function,

$$W(u) = \begin{cases} \left(1-|u|^3\right)^3, & \text{for } |u|<1 \\ 0 & , \text{for } |u|\geq 1 \end{cases}. \tag{4}$$

The choice of weight function in certain statistical studies is not a significant problem. We use the same weight function as [17].

Fan [18] derives the local weighted least squares solution to the problem (3) in the form of a matrix equation,

$$\hat{\boldsymbol{\beta}}_\alpha = (\mathbf{X}_\alpha^T \mathbf{W}_\alpha \mathbf{X}_\alpha)^{-1} \mathbf{X}_\alpha^T \mathbf{W}_\alpha \mathbf{Y} \tag{5}$$

where,

$$\mathbf{X}_\alpha = \begin{pmatrix} 1 & (x_1 - x_0) & \cdots & (x_1 - x_0)^p \\ \vdots & \vdots & & \vdots \\ 1 & (x_\text{k} - x_0) & & (x_\text{k} - x_0)^p \end{pmatrix}$$

$$\mathbf{Y} = \begin{pmatrix} y(x_1) \\ y(x_2) \\ \vdots \\ y(x_\text{k}) \end{pmatrix}$$

$$\hat{\boldsymbol{\beta}}_\alpha = \begin{pmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \\ \vdots \\ \hat{\beta}_p \end{pmatrix},$$

and the weighted diagonal matrix, $\mathbf{W}_\boldsymbol{\alpha} = \text{diag}\{w_i(x_0)\}$ of size $k\times k$. Prediction of local polynomial regression at point x_0 can be written in the form,

$$\hat{y}(x_0) = f(x_0) = \sum_{j=0}^{p} \hat{\beta}_j x_0^j. \tag{6}$$

2.2. Optimal Smoothing Parameter

The smoothing parameter α and the sample size n determine the local window width or the proportion of observations balanced in each local polynomial regression. The selection α determines the smoothness of the scatterplot where it is close to zero, giving a very bumpy scatterplot and overfitting predictions. Meanwhile, for α close to one, it provides a smoother scatterplot but is far from the original data features (data characteristics) and gives an underfitting forecast.

The response variable comes from the trigonometric function f(x) = Sin 2x, where the error is generated from the standard normal distribution with a mean of zero and a variance of 0.4, $\varepsilon_\text{i} \sim N(0, 0.4)$. In contrast, the independent variable design is taken from the lower limit $x_\text{min} = 0$ and the upper limit $x_\text{max} = 1.5\pi$, which has the same distance. Then take the sample size for the paired bivariate variable ($x_\text{i,}$ y_i) by as much as 50 points. Figure 1(a) is a first and second-degree LPR scatterplot for the value of $\alpha = 0.1$. Figure 1 is a scatterplot of LPR predictions of degrees one and two with the choice of smoothing parameter values $\alpha = 0.1$, which is close to zero and $\alpha = 0.99$, which is close to one. The response variable is derived from the trigonometric function $f(x)$ = Sin 2x. The error is generated from a standard normal distribution with a mean of zero and a variance of 0.4, $\varepsilon_\text{i} \sim N(0, 0.4)$. While the independent variable design is taken from the lower limit $x_\text{min} = 0$ and the upper limit $x_\text{max} = 1.5\pi$, which has the same distance, then take the sample size for the paired bivariate variable (x_i, y_i) by as much as 50 points. Figure 1(a) is a first and second-degree LPR scatterplot for the value of $\alpha = 0.1$.

The black curve is a first-degree LPR which is very undulating and gives a slightly accurate prediction of the value of the initial response. The second-degree LPR, shown by the red curve, produces a scatterplot of the piecewise-linear function. Almost all the predicted values are the same as the original response values. These curves provide overfitting predictions, which can be misleading because the variance becomes very large at the new point. The cause of the overfitting prediction on LPR is that there are quite a few points in $\mathbb{N}(x_0)$, so the scatterplot becomes piecewise-linear. Figure 1(b) shows the first and second-degree LPR scatterplots far from the original data features. The predictions produced by these two LPRs are often mentioned as underfitting, which has a high bias. The reason is that many points in $\mathbb{N}(x_0)$ are close to the size of the original sample data.

The overfitting and underfitting predictions from fitting will give unreasonable inferences. Loader [19] expressed the need to control the fit of the curve by using the best or optimal parameter. One way to get the optimal parameter is using cross-validation,

$$\text{CV}(\alpha) = \frac{1}{n} \sum_{i=1}^{n} \left(y(x_\text{i}) - \hat{y}_\alpha^{-i}(x_\text{i})\right)^2, \tag{7}$$

where $\hat{y}_\alpha^{-i}(x_\text{i})$ is the prediction of LPR at point x_i by removing one point $y(x_\text{i})$ on sample data. CV(α) behaviour is very sensitive to small sample data. The small sample size causes the CV(α) score to be monotonously increasing. The small size effect is because the $\mathbf{X}_\alpha^T \mathbf{W}_\alpha \mathbf{X}_\alpha$ matrix in (5) becomes an ill condition or is known as non-invertible matrix terminology.

The procedure for deriving an algorithm to get α optimal is as follows.

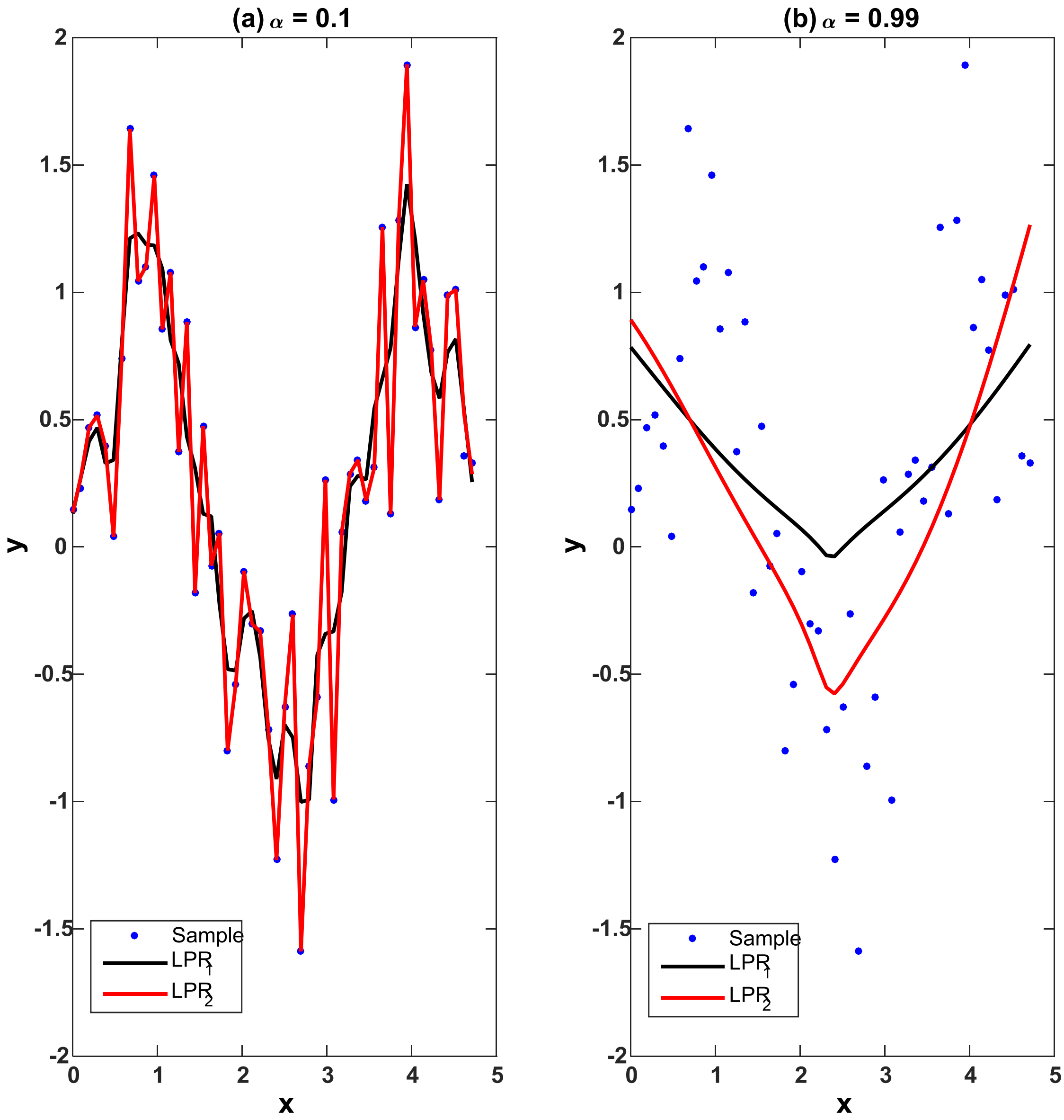

Figure 1. Scatterplot of Predicted Local Polynomial Regression

2.3. Optimal *α* Smoothing Parameter Search Algorithm

1. Determining the lower and upper limits of the *α* smoothing parameter. Give initial values starting from *α* = 0.1 while [19] suggests 0.25. The simulation results should show that if the *α* parameter is less than 0.1, it gives an ill condition.
2. Suppose that the sample data set is $\{(x_1, y(x_2)), (x_1, y(x_2)), \cdots, (x_n, y(x_n))\}$ and the initial value of the smoothing parameter is denoted α_1. For convenience, sample data is denoted in the form,

$$\{(x_1, y_1),\ (x_2, y_2), \cdots,\ (x_n, y_n)\}.$$

Next, there is an iterative process for α_1 as follows.

a. Take the point (x_1, y_1) as the test point.
b. Remove the point (x_1, y_1) from the sample data so that the training data is without (x_1, y_1).
c. Apply α_1 to formula (6) to predict $\hat{y}_1$.
d. Storing the values, $\hat{y}_1$, then repeating steps a to d for taking (x_2, y_2) as a test point and re-entering (x_1, y_1) into the training data and deleting (x_2, y_2). This process continues until the *n*-th sample point.
e. Calculating the CV(α_1) using the formula (7).

3. Take $\alpha_2 = \alpha_1 + \delta_1$, where $\delta_1 \in (0,1)$ and perform the process in step 2 to get CV(α_2). We continue this process and stop until $\alpha_t = \alpha_{t-1} + \delta_{t-1}$ is close to or equal to the upper limit of $\alpha_{\max}$. Suppose the set Ω is a collection of α_t.
4. Determine the best or optimal α with the criteria,

$$\alpha_{\text{opt}} = \min_{\alpha_t \in \Omega} \{\text{CV}(\alpha_t)\} \tag{8}$$

The LPR scatterplot in Figure 2 uses the previously described sample data on $f(x) = \sin 2x$. The α design space starts from $\alpha_{\min} = 0.1$ and $\alpha_{\max} = 0.9$ with an increase in of 0.01. Figure 2(a) is a scatterplot of the first-degree LPR which gives a score of $\text{CV}_{\min} = 0.1938$ for $\alpha_{\text{opt}} = 0.17$. Figure 2(b) is a scatterplot of the second-degree LPR, giving a score of $\text{CV}_{\min} = 0.1905$ for $\alpha_{\text{opt}} = 0.47$. The simulation results show that the $\text{CV}_{\min}$ score of the second-degree LPR is smaller than the first-degree LPR. The α_{opt} value of the first-degree LPR is close to $\alpha_{\min}$, while the second-degree LPR is close to the midpoint.

Figure 3 applies the previously obtained optimal α smoothing parameter. The scatterplot of the black curve (degree one) is less smooth than the red curve (degree two). We should also consider that the smoothness of a curve is also influenced by the degree of the polynomial. The higher the degree of LPR, the smoother the scatterplot will be, and a soft curve will give less bias. However, a high degree of LPR has more factors that need to be considered in a model resulting in a more considerable variance. Researchers usually choose low polynomials: degrees one (linear) and two (squared).

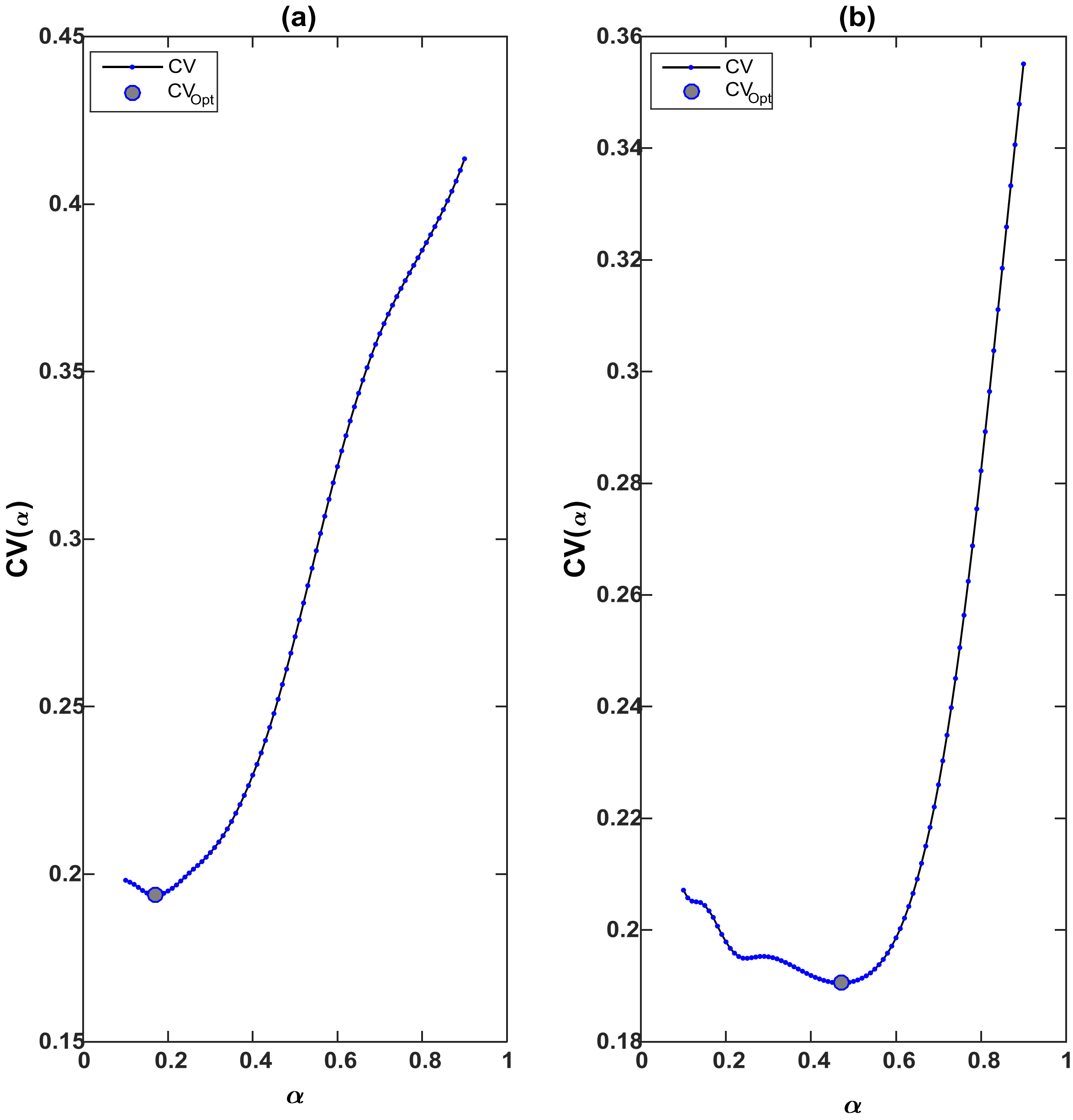

Figure 2. Scatterplot of Cross-Validation

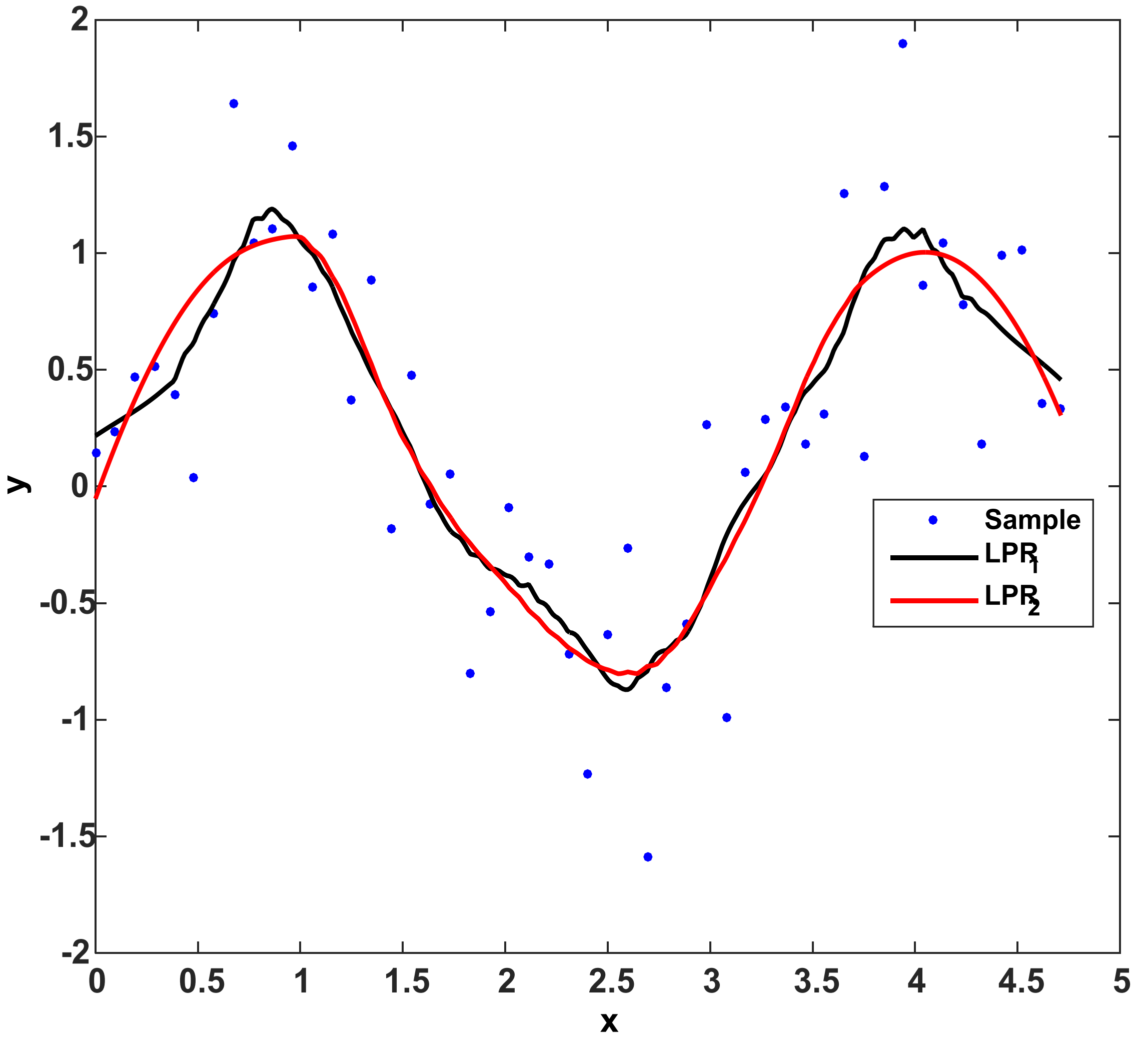

Figure 3. Scatterplot of LPR with Optimal Smoothing Parameter

2.4. Bootstrap-t Prediction Interval

Estimating the predictive confidence interval for LPR requires a large data size, and the prediction residuals are normally distributed. Statistical inference in LPR predictions can be misleading if the data are not large and not normally distributed. The bootstrap method can be used if the requirements for statistical analysis are not met, such as a small sample size and the assumption of a normal distribution is not available. Estimating confidence intervals can use several bootstrap methods. Readers can read more about bootstrap intervals in the literature [1-6].

This study focuses on the Bootstrap-t confidence interval in LPR prediction. Furthermore, this confidence interval is called the bootstrap-t prediction interval for convenience. The following is a development and extension of the procedure from [1] on the bootstrap-t prediction interval at a point x_0. Suppose the predicted expectation is equal to the actual value, $\mathrm{E}(\hat{y}_0) = y_0$, then the Z transformation can be expressed in the form,

$$\mathrm{Z}_0 = \frac{\hat{y}_0 - y_0}{\mathrm{SE}_{\mathrm{boot}}(y_0)}, \tag{9}$$

where $\mathrm{SE}_{\mathrm{boot}}(y_0)$ is the estimated standard error of the defined B bootstrap sample,

$$\mathrm{SE}_{\mathrm{boot}}(y_0) = \left\{ \frac{1}{B} \sum_{b=1}^{B} \left(\hat{y}_0^{*b} - \overline{\hat{y}_0^{*}} \right)^2 \right\}^{1/2}, \tag{10}$$

with $\overline{\hat{y}_0^{*}}$ is the average of the B bootstrap sample defined,

$$\overline{\hat{y}_0^{*}} = \frac{1}{B} \sum_{b=1}^{B} \hat{y}_0^{*b}$$

The y_0 prediction confidence interval is based on the Z distribution,

$$\hat{y}_0 - \mathrm{z}_{\mathrm{upper}} \times \mathrm{SE}_{\mathrm{boot}}(y_0) \le y_0 \le \hat{y}_0 - \mathrm{z}_{\mathrm{lower}} \times \mathrm{SE}_{\mathrm{boot}}(y_0). \quad (11)$$

Note that $\mathrm{z}_{\mathrm{lower}}$ and $\mathrm{z}_{\mathrm{upper}}$ are quantiles of the Z distribution which can be approximated using bootstrap quantiles,

$$\mathrm{Z}_0^{*b} = \frac{\hat{y}_0^{*b} - \hat{y}_0}{\mathrm{se}_0^{*b}}, \quad (12)$$

where se_0^{*b} is an estimate of the standard error of the replication of the $\hat{y}_0^{*b}$ statistic from the second bootstrap sample, which will be explained in detail in the algorithm. Efron [1] suggests 25 times is enough to get a stable standard error. If define $\mathrm{z}_{\mathrm{lower}} = \mathrm{z}^{*(\gamma/2)}$ as the γ/2-th sample quantile of Z^{*1}, Z^{*2}, ⋯, Z^{*B}, where γ is the level of significance, then $P(\mathrm{Z} \le \mathrm{z}_{\mathrm{lower}}) = P(\mathrm{Z} \le \mathrm{z}^{*(\gamma/2)}) \approx \gamma/2$. In the same way it is defined that $\mathrm{z}_{\mathrm{upper}} = \mathrm{z}^{*(1-\gamma/2)}$ so that the form (11) can be expressed,

$$\begin{aligned} &P(\hat{y}_0 - \mathrm{z}^{*(1-\gamma/2)} \times \mathrm{SE}_{\mathrm{boot}}(y_0) \le y_0 \le \hat{y}_0 - \mathrm{z}^{*(\gamma/2)} \times \mathrm{SE}_{\mathrm{boot}}(y_0)) \\ &= P(-\mathrm{z}^{*(1-\gamma/2)} \times \mathrm{SE}_{\mathrm{boot}}(y_0) \le y_0 - \hat{y}_0 \le -\mathrm{z}^{*(\gamma/2)} \times \mathrm{SE}_{\mathrm{boot}}(y_0)) \\ &= P(\mathrm{z}^{*(\gamma/2)} \le \mathrm{Z} \le \mathrm{z}^{*(1-\gamma/2)}) \approx 1-\gamma. \end{aligned} \quad (13)$$

There are two stages of bootstrap sampling from the explanation above. The first step is to take the first bootstrap sample *B* times from the empirical distribution to determine the bootstrap standard error estimate, then from every second bootstrap sample B_1 times to get an estimate of the standard error, se_0^{*b}. Based on this step, we derive two algorithms as follows.

2.5. Algorithm-1: Residual-paired Bootstrap

1. Determining the optimal *α* smoothing parameter using sample data,

$$\{(x_1, y_1),\ (x_2, y_2), \cdots,\ (x_{\mathrm{n}}, y_{\mathrm{n}})\}.$$

2. Applying the results of step 1 to equation (6) to obtain a predictive data set,

$$\{(x_1, \hat{y}_1),\ (x_2, \hat{y}_2), \cdots,\ (x_{\mathrm{n}}, \hat{y}_{\mathrm{n}})\}.$$

3. The first bootstrap sampling for *b* from 1 to *B* with the following steps.

a. Perform paired bootstrap resampling of the original data {(x_1, y_1), (x_2, y_2),⋯, (x_{n}, y_{n})} so that the b-th bootstrap sample is obtained,

$$\left\{\left(x_1^{*b}, y_1^{*b}\right), \left(x_2^{*b}, y_2^{*b}\right), \cdots, \left(x_{\mathrm{n}}^{*b}, y_{\mathrm{n}}^{*b}\right)\right\}.$$

b. Applying the results in steps 1 and 3(a) into formula (6) to obtain the predicted set of the *b*-th bootstrap sample,

$$\left\{(x_1, \hat{y}_1^{*b}), (x_2, \hat{y}_2^{*b}), \cdots, (x_{\mathrm{n}}, \hat{y}_{\mathrm{n}}^{*b})\right\}.$$

c. Determine the residual set { $\mathrm{e}_1^*, \mathrm{e}_2^*, \cdots, \mathrm{e}_{\mathrm{n}}^*$ } from the difference between b-th bootstrap predictions and b-th bootstrap sample, $\mathrm{e}_{\mathrm{i}}^* = \hat{y}_{\mathrm{i}}^{*b} - y_{\mathrm{i}}^{*b}$.

d. The second bootstrap sampling for b_1 from 1 to B_1 with the following steps.

i. Sampling using residual bootstrap to determine b_1-th bootstrap sample

$$\left\{(x_1, \hat{y}_1^{**b_1}), (x_2, \hat{y}_2^{**b_1}), \cdots, (x_{\mathrm{n}}, \hat{y}_{\mathrm{n}}^{**b_1})\right\},$$

where $\hat{y}_{\mathrm{i}}^{**b_1} = \hat{y}_{\mathrm{i}}^{*b_1} + \mathrm{e}_{\mathrm{i}}^{**}$ with $\mathrm{e}_{\mathrm{i}}^{**}$ is retrieved in return from the residual set { $\mathrm{e}_1^*, \mathrm{e}_2^*$, ⋯, $\mathrm{e}_{\mathrm{n}}^*$ }.

ii. Entering the data $(x_1, \hat{y}_1^{**b_1})$, ⋯ ,$(x_{\mathrm{n}}, \hat{y}_{\mathrm{n}}^{**b_1})$ into equation (6) to determine the prediction set of the b_1-th bootstrap sample,

$$\left\{(x_1, \hat{\hat{y}}_1^{**b_1}), (x_2, \hat{\hat{y}}_2^{**b_1}), \cdots, (x_{\mathrm{n}}, \hat{\hat{y}}_{\mathrm{n}}^{**b_1})\right\}.$$

iii. Repeat steps i-ii as much as B_1.

iv. Determine the standard error estimate of the second bootstrap sample for each point using equation (10),

$$\mathrm{se}_{\mathrm{i}}^{*b} = \left\{ \frac{1}{B_1} \sum_{b_1=1}^{B_1} \left(\hat{\hat{y}}_{\mathrm{i}}^{**b_1} - \overline{\hat{\hat{y}}_{\mathrm{i}}^{**}} \right)^2 \right\}^{1/2},$$

where,

$$\overline{\hat{\hat{y}}_{\mathrm{i}}^{**}} = \frac{1}{B_1} \sum_{b_1=1}^{B_1} \hat{\hat{y}}_{\mathrm{i}}^{**b_1},$$

so the set of ordered pairs is

$$\left\{(x_1, \mathrm{se}_1^{*b}), (x_2, \mathrm{se}_2^{*b}), \cdots, (x_{\mathrm{n}}, \mathrm{se}_{\mathrm{n}}^{*b})\right\}.$$

4. Repeat step 3 much *B* times.
5. Determine the estimated standard error of each first bootstrap sample for each point using equation (10) to get set $\mathbf{V}^{*1}$, $\mathbf{V}^{*2}$, ⋯, $\mathbf{V}^{*\mathrm{B}}$ where,

$$\mathbf{V}^{*b} = \left\{(x_1, \mathrm{se}_1^{*b}), (x_2, \mathrm{se}_2^{*b}), \cdots, (x_{\mathrm{n}}, \mathrm{se}_{\mathrm{n}}^{*b})\right\}.$$

6. Determine the set of bootstrap quantiles using equation (12) for each bootstrap sample, $\mathbf{Z}^{*1}$, $\mathbf{Z}^{*2}$, ⋯, $\mathbf{Z}^{*\mathrm{B}}$ where

$$\mathbf{Z}^{*b} = \left\{(x_1, z_1^{*b}), (x_2, z_2^{*b}), \cdots, (x_{\mathrm{n}}, z_{\mathrm{n}}^{*b})\right\}.$$

7. Sort from smallest to largest of the set, $\mathbf{Z}^{*1}$, $\mathbf{Z}^{*2}$, ⋯, $\mathbf{Z}^{*\mathrm{B}}$ so that the (γ/2)-th bootstrap quantile is obtained,

$$\mathbf{Z}^{*(\gamma/2)B} = \{(x_1, z_1^{*(\gamma/2)}), (x_2, z_2^{*(\gamma/2)}), \cdots, (x_{\mathrm{n}}, z_{\mathrm{n}}^{*(\gamma/2)}).$$

and the (1−γ/2)-th bootstrap quantile,

$$\mathbf{Z}^{*(1-\gamma/2)B} = \{(x_1, z_1^{*(1-\gamma/2)}), (x_2, z_2^{*(1-\gamma/2)}), \cdots, (x_{\mathrm{n}}, z_{\mathrm{n}}^{*(1-\gamma/2)}).$$

8. Determine the lower and upper limits of the bootstrap-t prediction interval using the equation (13),

$$\mathbf{CI}_{\text{Lower}} = \{(x_1, \hat{y}_1 - z_1^{*(1-\gamma/2)} \times \text{SE}_{\text{boot}}(y_1)), \cdots, (x_n, \hat{y}_{\text{n}} - z_{\text{n}}^{*(1-\gamma/2)} \times \text{SE}_{\text{boot}}(y_{\text{n}}))\}$$

$$\mathbf{CI}_{\text{Upper}} = \{(x_1, \hat{y}_1 - z_1^{*(\gamma/2)} \times \text{SE}_{\text{boot}}(y_1)), \cdots, (x_{\text{n}}, \hat{y}_{\text{n}} - z_{\text{n}}^{*(\gamma/2)} \times \text{SE}_{\text{boot}}(y_{\text{n}}))\}.$$

The difference between algorithm-2 and algorithm-1 is only in the first bootstrap sampling, where algorithm-2 uses bootstrap residuals. The stages in algorithm-2 are written only different stages from algorithm-1.

2.6. Algorithm-2: Residual-residual Bootstrap

In algorithm-2, steps 1-2 are the same as in algorithm-1.

3. The first bootstrap sampling for *b* from 1 to *B* with the following steps.
   a. Determine the residual set { $\text{e}_1, \text{e}_1, \cdots, \text{e}_{\text{n}}$ } which is the difference between the prediction and the actual value, $\text{e}_{\text{i}} = \hat{y}_{\text{i}} - y_{\text{i}}$.
   b. Generating a bootstrap sample by resampling the bootstrap residual, $y_{\text{i}}^{*b} = y_{\text{i}} + \text{e}_{\text{i}}^{*}$, where e_{i}^{*} is taken with the return of the set of residuals step 3(a). So that the *b*-th bootstrap sample is obtained,

$$\left\{\left(x_1, y_1^{*b}\right), \left(x_2, y_2^{*b}\right), \cdots, \left(x_{\text{n}}, y_{\text{n}}^{*b}\right)\right\}.$$

Steps 3(c) to 8 in algorithm-2 are the same as in algorithm-1.

3. Simulation Results

The sample data design for the simulation of algorithms one and two is generated as follows. The independent variables follow the procedure in section 2.2. The response variable follows the trigonometric function *f*(*x*) = Cos 2*x*. The error is generated from the standard normal distribution with a mean of zero, and the variance of the error is 0.5. The simulation considers the assumption that the error is Gaussian distributed. The following research will discuss leverage and influence points, which may be outliers in nonparametric regression functions' x and y spaces. Outliers are unusual data points that have a dramatic impact on statistical inference.

Figure 4 is the simulation result of algorithm-1 with the generation of sample data with a size of 100. The number of the first bootstrap sample *B* = 1000 times and the second bootstrap sample 100 times. So that there are 100000 times of replication, it is said to be quite an expensive simulation. The confidence level in the interval built is 95% or a significance level of γ = 0.05 is used. Figure 4(a) is a scatterplot of bootstrap-t prediction interval with a first-degree polynomial that has roughness or is more wavy. Figure 4(b) is a scatterplot of the second-degree LPR, which shows it is smoother than the first degree. The interval length at each point indicates that the second-degree LPR is longer than the first-degree. This will have the effect that the probability of coverage of the second-degree LPR prediction interval is closer to the nominal coverage probability than that of the first-degree LPR prediction interval. The coverage probability of the first-degree LPR in Figure 4(a) is 0.88, while the second-degree LPR in Figure 4(b) is 9.45.

Figure 5 is the result of the simulation of algorithm-2 using the same sample data generation process in Figure 4. The scatterplot in Figures 4 and 5 shows a view that is much different from one another. The scatterplot formed by algorithm-2 is smoother than algorithm-1. It is because Figure 5 has fewer waves than Figure 4. Next, the observations in Figure 5 show that the first-degree LPR leads to more wiggling than the second-degree LPR, which means that Figure 5(b) is smoother than Figure 5(a). The probability magnitude of the first-degree LPR prediction interval in algorithm-2 is 0.91, while the second-degree LPR in Figure 5(b) is 0.95. Algorithm-2 shows that the coverage probability for LPR degree one or two is close to the nominal coverage probability.

3.1. Optimal Smoothing Parameter Effect

How will it affect the bootstrap-t prediction interval if algorithm-1 and algorithm-2 do not use the optimal smoothing search parameter? However, it uses arbitrary parameter selection from values between zero and one. Simulation is carried out on the sample data to answer this question by selecting the smoothing parameter of $\alpha = 2/3$. Figure 6 shows that the curvature of the lower and upper bounds of the interval of algorithm-1 does not follow the features of the prediction (curvature of the black line). Figure 6(a) has a coverage probability of 0.95 while Figure 6(b) has a coverage probability of 0.98. Prediction interval becomes superior or becomes conservative interval.

Figure 7 shows the curvature of the lower and upper bounds of the interval of algorithm-2 following the prediction features. However, the curve of the prediction is not in the middle of the coverage probability area. Figure 7(a) shows the curvature of the prediction curve approaching the curvature of the lower limit, while Figure 7(b) comes to the curvature of the upper limit of the interval. The coverage probability of the prediction interval in Figure 7(a) is 0.66, and Figure 7(b) is 0.89. It causes both prediction intervals to be anti-conservative or permissive because they are less than the nominal coverage probability.

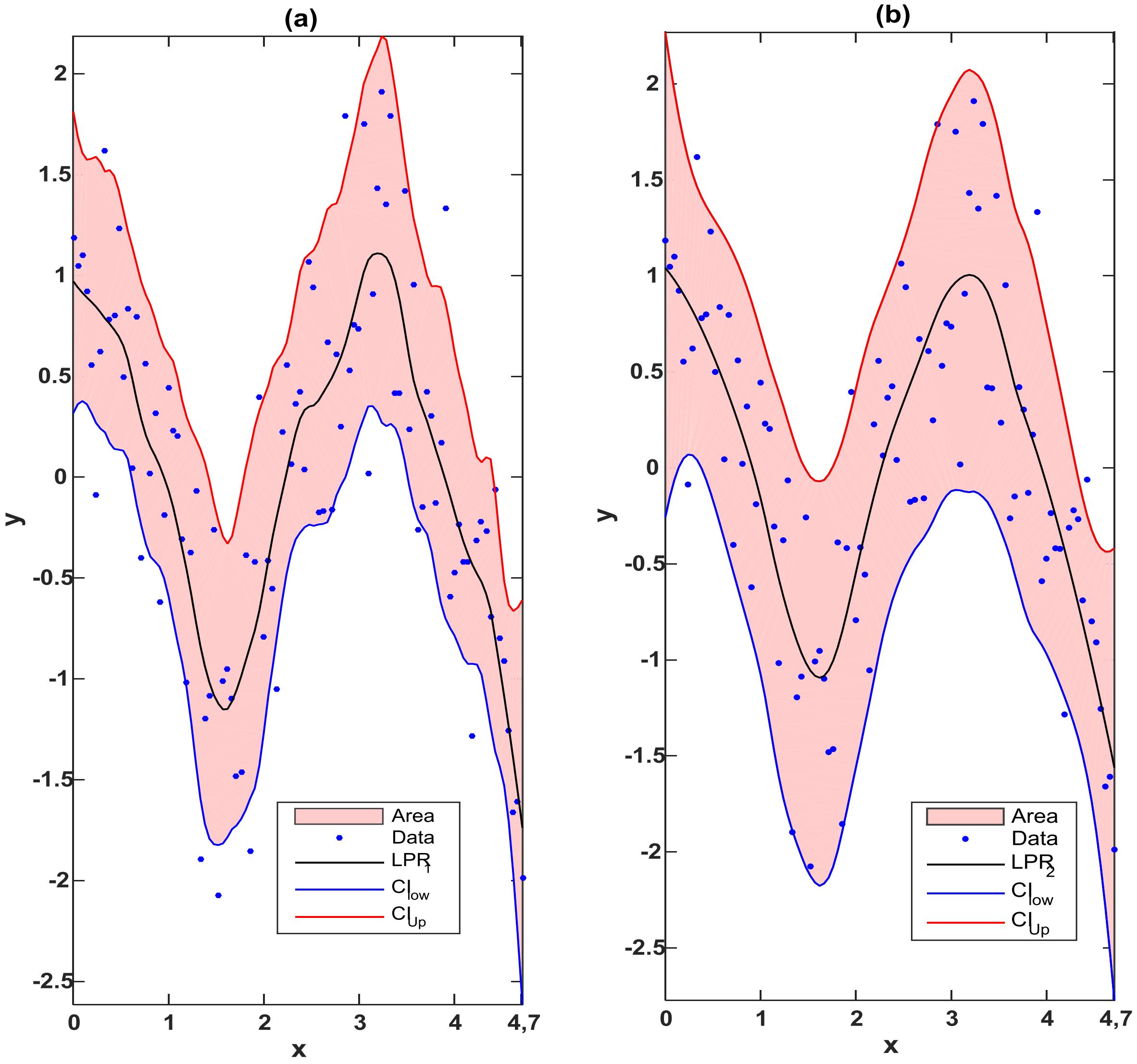

Figure 4. Scatterplot of Bootstrap-t Prediction Interval for Algorithm-1

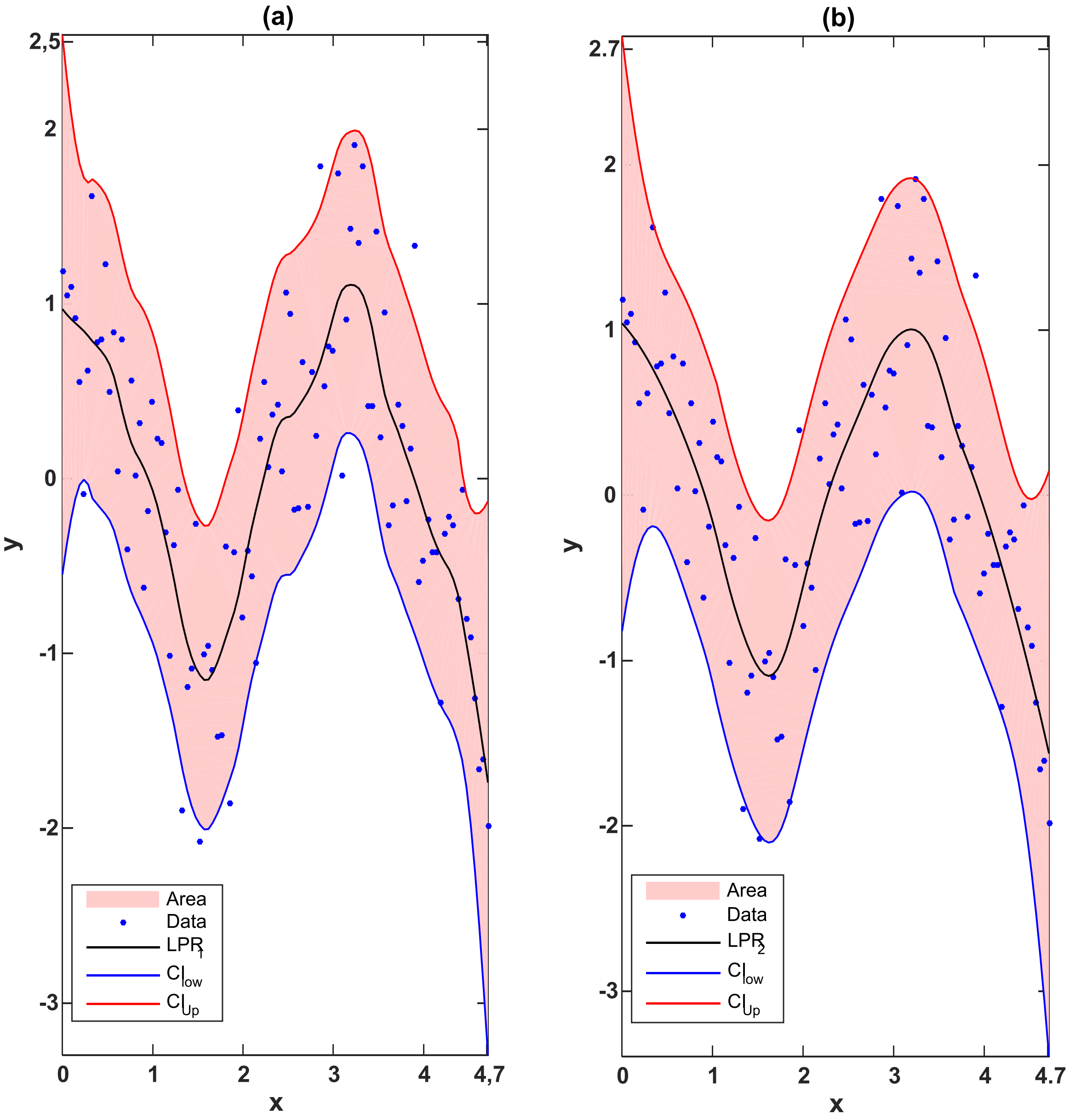

Figure 5. Scatterplot of Bootstrap-t Prediction Interval for Algorithm-2

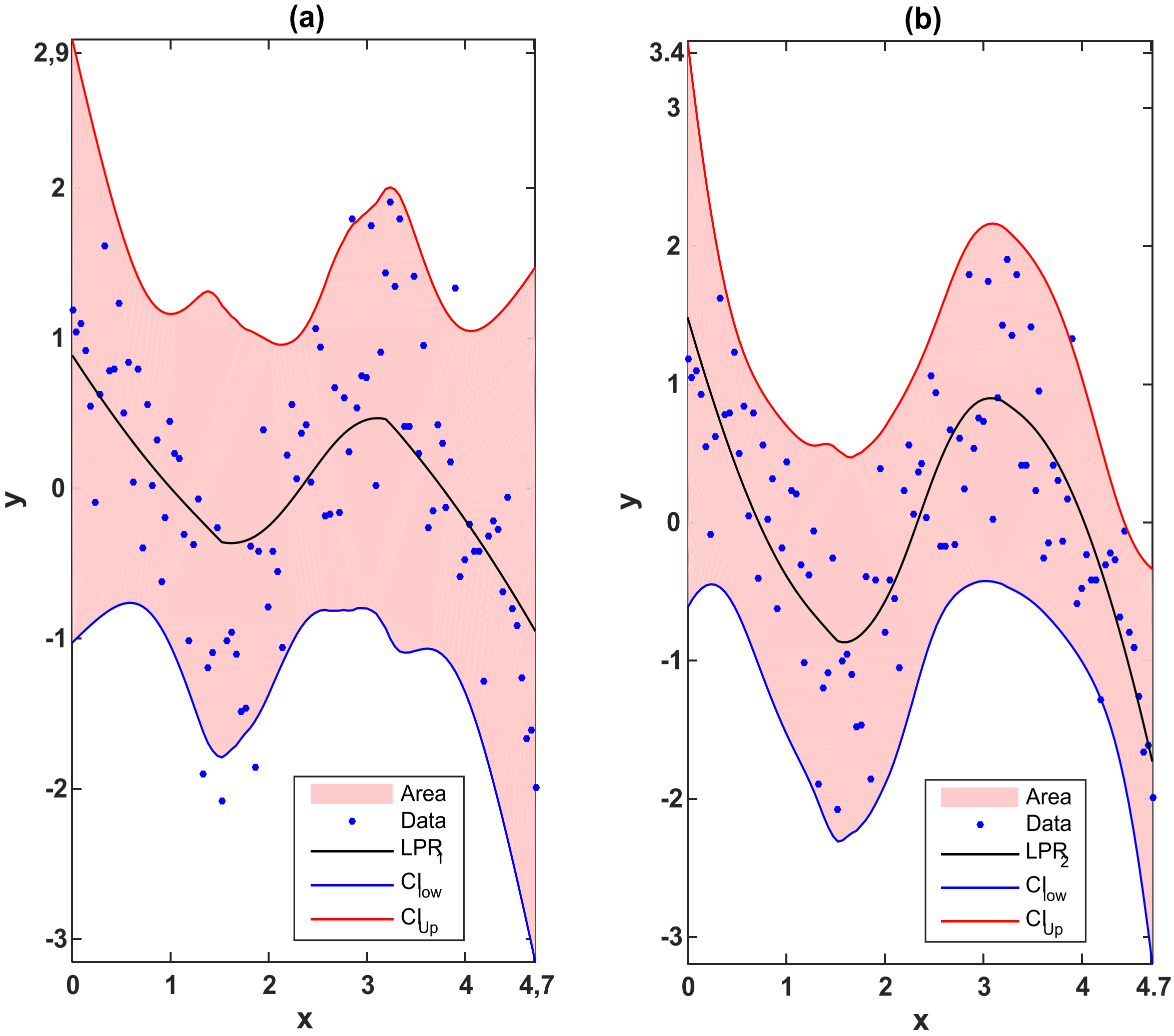

Figure 6. Scatterplot of Algorithm-1 with Smoothing Parameter Selection

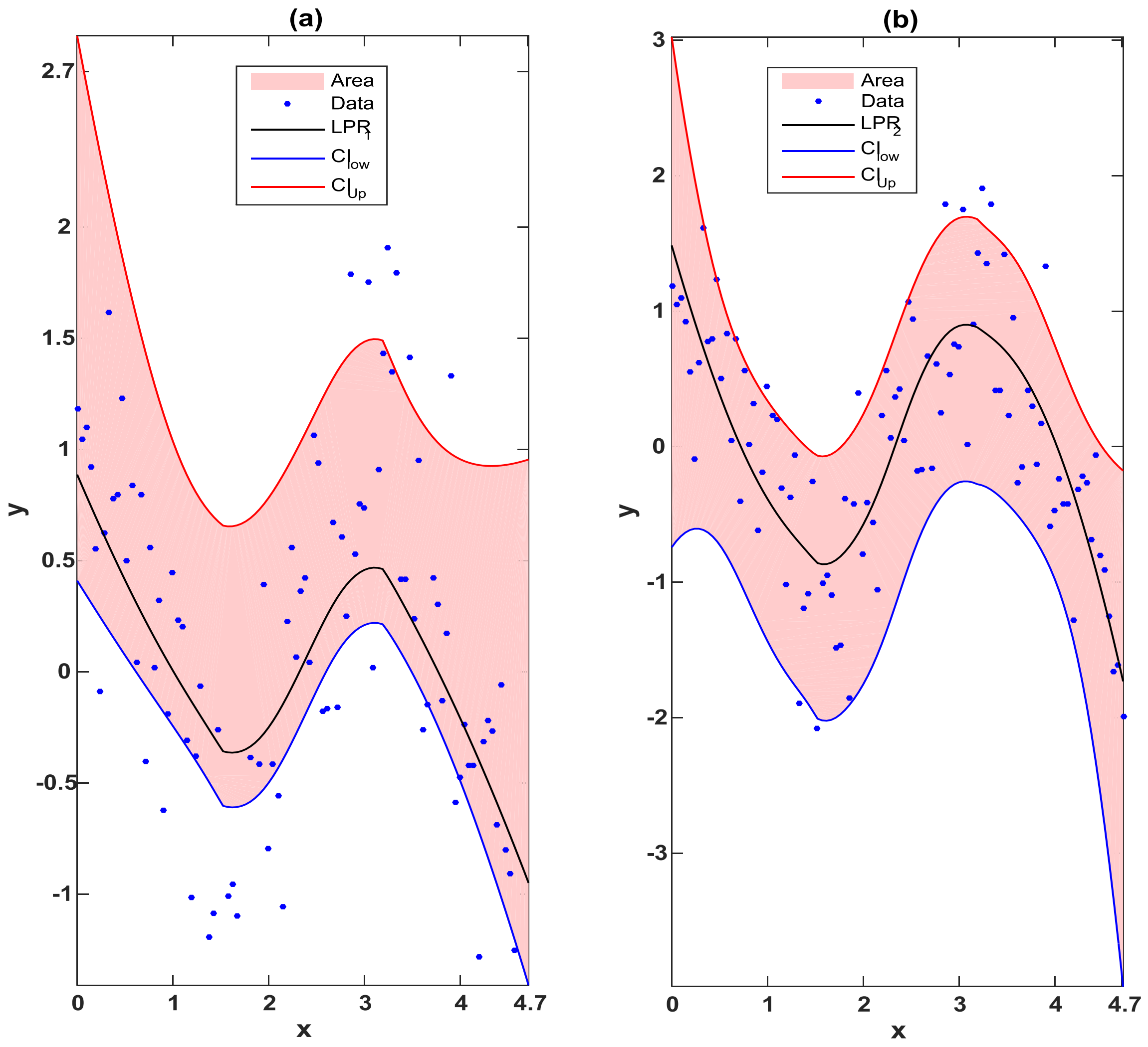

Figure 7. Scatterplot of Algorithm-2 with Smoothing Parameter Selection

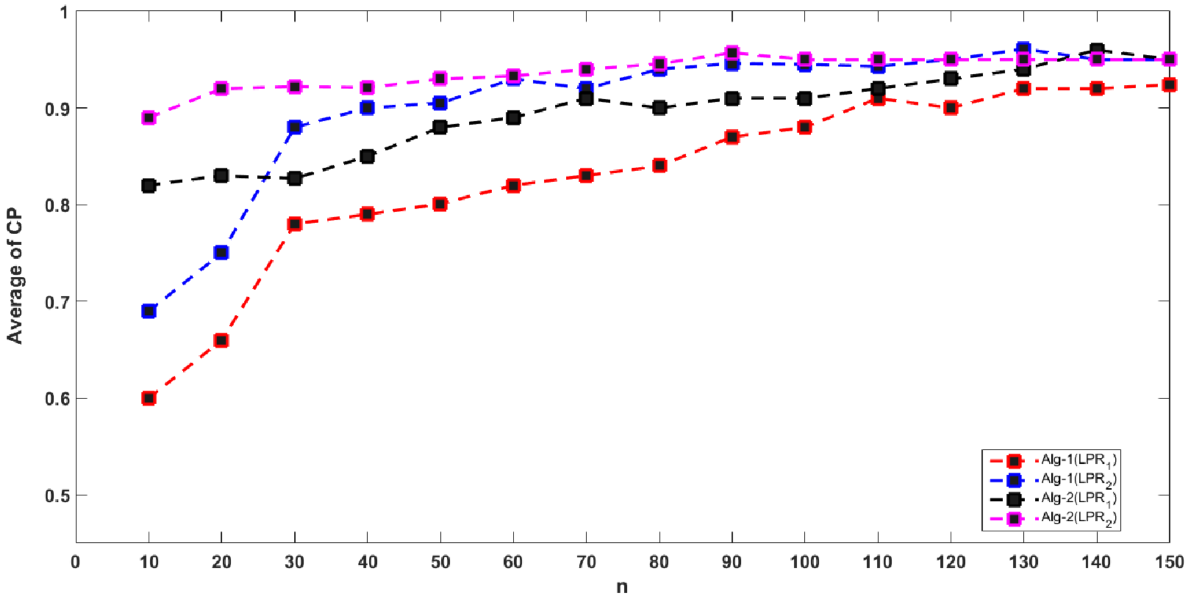

Figure 8. Scatterplot of Algorithm-1 and 2 Increase in Sample Data Size

3.2. Increase in Sample Data Size

The simulation performs ten major replications for each sample data size. The purpose and objective of conducting major replication are to stabilise the randomness of the sample data generation and to see statistical consistency. Because this simulation is expensive, major replication is only carried out in small quantities. The procedure for increasing the sample data size is by adding ten new sample data to the old sample data. We also consider the Monte Carlo simulation in [20] to measure performance on small sample data sizes. Figure 8 shows that algorithm-2 for second-degree LPR (magenta line) performs better than the other three methods. Algorithm-1 with a second-degree LPR (blue line) shows that the coverage probability is still acceptable. The rate of convergence of the coverage probabilities generated by algorithm-2 for the second-degree LPR is faster than the other three. The more significant the increase in sample size indicates that the coverage probability of the four intervals is close to the nominal coverage probability. The reader can refer to the Appendix for the mathematical justification and proof of the two proposed algorithms where the sample data size goes to infinity.

4. Conclusions

The smoothness and fit of the scatterplot of the bootstrap-t prediction interval are influenced by the smoothing parameter and the degree of the polynomial. The smoothing parameter value taken close to zero gives a prediction of being overfitting, while the value of the smoothing parameter close to one gives a forecast of underfitting. Then the choice of any value of the smoothing parameter allows misleading conclusions. Finding the optimal smoothing parameter value is necessary to construct a reasonable confidence interval. The degree of local polynomial regression is also one of the determinants of the smoothness of the scatterplot. The simulation results show that the second-degree LPR gives a scatterplot with fewer waves or wobbles in the prediction curve than the first-degree LPR.

The application of bootstrap resampling at the bootstrap-t prediction interval in local polynomial regression modelling provides two new algorithm proposals, namely algorithm-1 and algorithm-2. Algorithm-1 is based on paired and residual bootstrap resampling, while algorithm-2 is based on residual and residual bootstrap resampling. It shows that both algorithms do resampling twice, whereas the second resampling always uses residual bootstrap. The intent and purpose are always to use bootstrap residuals in the second resampling so that the variance appears is not too large or maintains the predictive features. Two algorithms for first-degree LPR provide a prediction interval that is coarser or more wiggling than second-degree LPR. The relatively large sample data coverage probability indicates that the second-degree LPR prediction interval is close to the nominal coverage probability. The increase in the sample data size indicates that algorithm-2 for second-degree LPR gives superior coverage probability results at each increase in the sample data size. Although the interval estimation is not from the exact interval theory, the bootstrap-t prediction interval from algorithm-2 with a second-degree LPR is more consistent and tends to the nominal coverage probability. However, this is in contrast to the prediction interval formed by algorithm-1 with first-degree LPR, which never reaches the nominal coverage probability. Algorithm-2 gives better results for relatively small sample data sizes.

In general, the two proposed bootstrap-t prediction interval algorithms provide prediction intervals that are still acceptable and work well on relatively small sample data sizes. We conclude that the two algorithms will achieve the probability of nominal coverage if the size of the sample data and the size of many bootstrap samples are to infinity, see the Appendix. The choice is left to the researcher which one to choose or use. Future research allows us to examine the following two things: determining the best and most robust bootstrap interval.

Appendix

Wasserman [21] presents the pivot quantity into two different parts at the bootstrap-t interval. The first resampling uses pivot quantity,

$$Z_n(x) = \frac{\hat{y}_n(x) - y_n(x)}{\mathrm{SE}_{\mathrm{boot}}(y_n(x))}.$$

We assume a new point $y_n(x)$ whose value is unknown but will be predicted to use n data observed. The quantity of pivot based on the second resampling is

$$Z_n^{*b}(x) = \frac{\hat{y}_n^{*b}(x) - \hat{y}_n(x)}{\mathrm{se}^{*b}(y_n(x))}.$$

The second resampling principle uses the residual bootstrap residual method; see algorithm-1. Follow the same way in algorithm-1 to get an estimated Cumulative Density Function (CDF) bootstrap from $Z_n^{*b}(x)$, say $\hat{G}(Z_n^{*b}(x))$, then we define

$$\hat{G}(Z_n^{*b}(x)) = \frac{1}{B}\sum_{b=1}^{B} I(Z_n^{*b}(x) \le z_n^{*b}(x)),$$

where I is an indicator function.

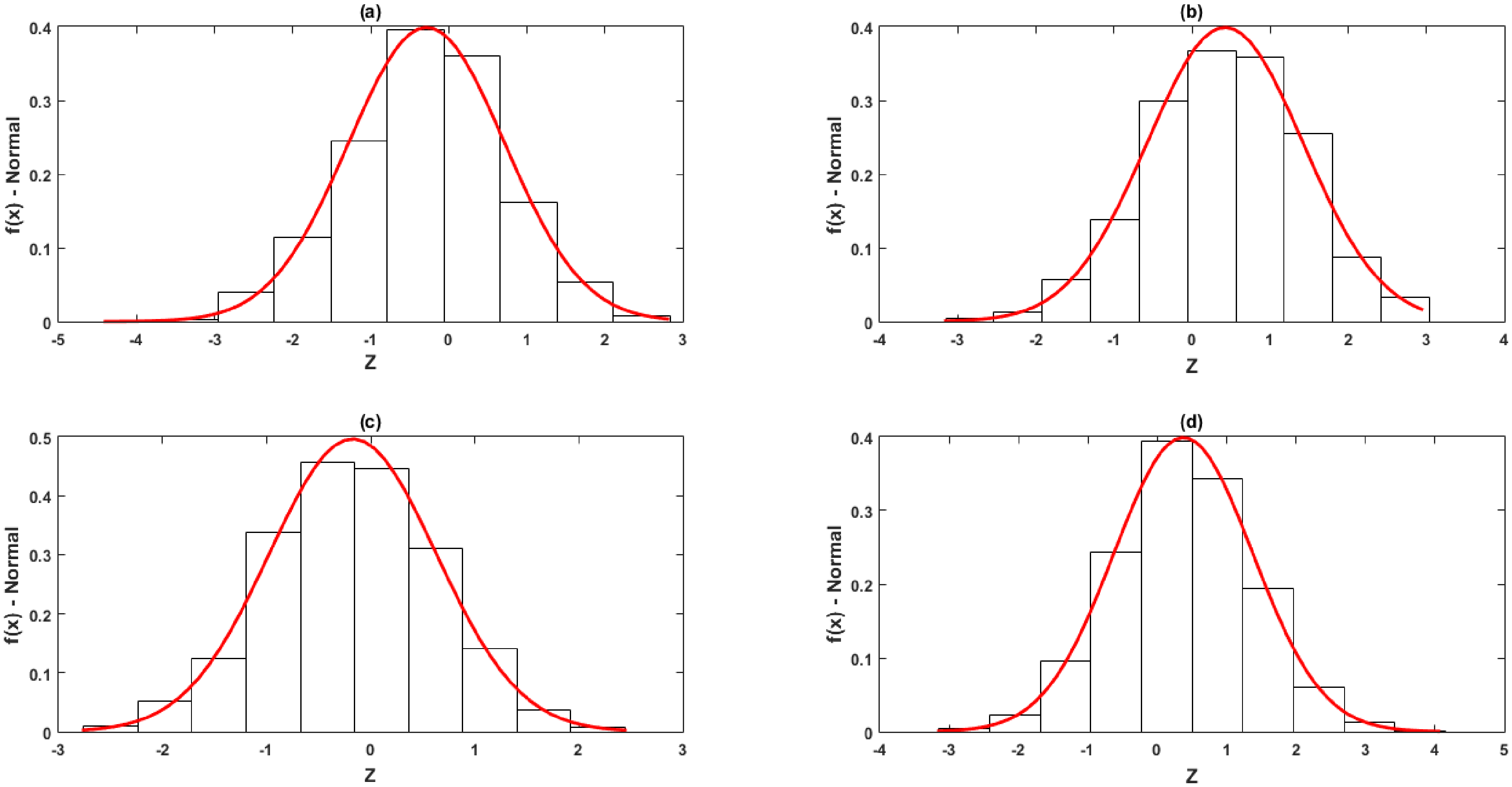

Figure 9. Estimating the $Z_n^{*b}(x)$ Distribution Using Algorithms-1 and 2 with LPR of Degrees One and Two

Efron [1], on page 47, shows that for the *B* bootstrap sample is getting bigger or going to infinity, it will provide an ideal bootstrap estimate. Suppose $\hat{g}(z_n^{*b}(x))$ is a bootstrap version to estimate the Probability Density Function (PDF). We assume that the $\hat{g}(z_n^{*b}(x))$ approaches the distribution of standard normality probability with justification through simulations and propositions (further research as a theorem). Figure 9 is the simulation result of one major replication at point x = 2, which is not observed from the sample data *n* = 100 with the number of bootstrap samples, *B* = 1000. The design of the input point as the independent variable and the output as the dependent variable follows section 3.

Figure 9(a) is a simulation result of algorithm-1 with a first-degree LPR with skewness and kurtosis, − 0.1511 and 3.0420, respectively. Figure 9(b) is a simulation result of algorithm-2 with a first-degree LPR with skewness and kurtosis, − 0.1401 and 2.9791, respectively. Figure 9(c) is a simulation result of algorithm-1 with a second-degree LPR with skewness and kurtosis, − 0.0385 and 2.9080, respectively. Figure 9(d) is a simulation result of algorithm-2 with a second-degree LPR with skewness and kurtosis, − 0.0349 and 3.0145, respectively. The simulation results also support that algorithm-2 with a second-degree LPR is better at approaching the *Z* standard normal distribution.

Proposition

Let $\hat{G}(Z_n^{*b}(x))$ be the ideal bootstrap CDF estimator; then $\hat{G}(Z_n^{*b}(x)) \xrightarrow{d} \Phi$, where Φ is the standard normal CDF.

If the condition satisfies the above proposition, it will give

$$\lim_{n\to\infty,\, B\to\infty} P(Z_n \le z^{*(\gamma/2)}) = P(Z \le z^{\gamma/2}) = \frac{\gamma}{2}.$$

The $\lim\limits_{n\to\infty,\, B\to\infty}$ notation follows Simamora [22], who applies ideal bootstrap estimation and asymptotic theory to show that the bootstrap estimate for the kriging variance is close to zero. Consequently, for *n* and *B* to go to infinity, then

$$\lim_{n\to\infty,\, B\to\infty} \mathrm{SE}^{*B}_{\mathrm{boot}}(y_n(x)) = \lim_{n\to\infty,\, B\to\infty} \hat{\sigma}_n^{*B} \equiv \hat{\sigma}\sum_{i=1}^{n} l_i^2(x),$$

where $\hat{\sigma}$ is the estimated standard deviation of the original sample data, and l_i is the weighted value. Finally, we conclude that Z_n converges in the distribution to *Z*~N(0,1), thus giving $P(z^{\gamma/2} \le Z \le z^{(1-\gamma/2)}) = 1-\gamma$ as the nominal coverage probability.

REFERENCES

[1] Efron, Bradley, and Robert J. Tibshirani. An Introduction to the Bootstrap, CRC Press, 1994.

[2] DiCiccio, Thomas J., and Bradley Efron. Bootstrap Confidence Intervals. Statistical Science, Vol.11, No.3, 189-228, 1996, http://www.jstor.org/stable/2246110

[3] Chernick, M. R. Bootstrap methods: A Guide for Practitioners and Researchers, John Wiley & Sons, 2011

[4] Shao, Jun, and Dongsheng Tu. The Jackknife and Bootstrap, Springer Science & Business Media, 2012.

[5] Ramachandran, K. M., and Tsokos, C. P. Mathematical Statistics with Applications in R, Academic Press, 2020.

[6] Zoubir, A. M., and Iskander, D. R. Bootstrap Techniques for Signal Processing, Cambridge University Press, 2004.

[7] Jung, K., Lee, J., Gupta, V., and Cho, G. Comparison of Bootstrap Confidence Interval Methods for GSCA Using A Monte Carlo Simulation, Frontiers in Psychology, Vol. 10, 2215, 2019. doi: 10.3389/fpsyg.2019.02215

[8] Manly, Bryan FJ. Randomisation, Bootstrap and Monte Carlo Methods in Biology: Texts in Statistical Science, Chapman and hall/CRC, 2018.

[9] Hall, Peter. On Bootstrap Confidence Intervals in Nonparametric Regression, The Annals of Statistics, Vol. 20, No. 2, 695-711, 1992, https://www.jstor.org/stable/224 1979

[10] Mojirsheibani, Majid, and Robert Tibshirani. Some Results on Bootstrap Prediction Intervals. Canadian Journal of Statistics, Vol. 24, No. 4, 549-568, 1996. https://www.jstor.org/stable/3315333

[11] Jacoby, William G. Loess: A Nonparametric, Graphical Tool for Depicting Relationships between Variables. Electoral studies, Vol. 19, No. 4, 577-613, 2000, doi: https://doi.org/10.1016/S0261-3794(99)00028-1

[12] Da Silva, C. M., Ribeiro, F. C. A., Vieira, J. W., and da Silva Filho, C. A.. Bootstrap Interval with Application in Environmental monitoring. International Journal of Environmental Studies, Vol. 77, No. 2, 335-348, 2020, doi: https://doi.org/10.1080/00207233.2019.1630108

[13] Stine, Robert A. Bootstrap Prediction Intervals for Regression. Journal of the American Statistical Association, Vol. 80, No. 392, 1026-1031, 1985.

[14] Polansky, Alan M. Stabilizing Bootstrap‐t Confidence Intervals for Small Samples, Canadian Journal of Statistics 28.3, 501-516, 2000.

[15] Tomal, Jabed H., and Jan JH Ciborowski. Ecological Models for Estimating Breakpoints and Prediction Intervals. Ecology and Evolution, Vol.10, No. 23, 13500-13517, 2020, doi: https://doi.org/10.1002/ece3.6955

[16] De Brabanter, K., De Brabanter, J., Gijbels, I., and De Moor, B. Derivative estimation with local polynomial fitting. Journal of Machine Learning Research, 14(1), 281-301, 2013.

[17] Cleveland, William S. Robust Locally Weighted Regression and Smoothing Scatterplots, Journal of the American Statistical Association, Vol. 74, No. 368, 829-836, 1979, doi: 10.1080/01621459.1979.10481038

[18] Fan, Jianqing, and Irene Gijbels. Local Polynomial Modelling and its Applications, Routledge, 2018.

[19] Loader, Clive. Local regression and likelihood, Springer Science & Business Media, 2006.

[20] Reza Pakyari. Inference on P[Y < X] for Geometric Extreme Exponential Distribution. Mathematics and Statistics, Vol.9, No.4, pp. 527-534, 2021, doi: 10.13189/ms.2021.090412.

[21] Wasserman, Larry. All of Nonparametric Statistics. Springer Science & Business Media, 2006.

[22] Simamora, E., Subanar and Kartiko, S. H., Asymptotic Property of Semiparametric Bootstrapping Kriging Variance in Deterministic Simulation. Applied Mathematical Sciences, Vol. 9. No. 50, pp. 2477-2491, 2015, doi: http://dx.doi.org/10.12988/ams.2015.52104