

## *Research Article*

# **Robust Wild Bootstrap for Stabilizing the Variance of Parameter Estimates in Heteroscedastic Regression Models in the Presence of Outliers**

**Sohel Rana,<sup>1,2</sup> Habshah Midi,<sup>1,2</sup> and A. H. M. R. Imon<sup>3</sup>**

<sup>1</sup> *Department of Mathematics, Faculty of Science, Universiti Putra Malaysia, Serdang, 43400 Selangor, Malaysia*

<sup>2</sup> *Laboratory of Computational Statistics and Operations Research, Institute for Mathematical Research, Universiti Putra Malaysia, Serdang, Selangor 43400, Malaysia*

<sup>3</sup> *Department of Mathematical Sciences, Ball State University, Muncie, IN 47306, USA*

Correspondence should be addressed to Soheli Rana, [srana\\_stat@yahoo.com](mailto:srana_stat@yahoo.com)

Received 31 July 2011; Revised 31 October 2011; Accepted 2 November 2011

Academic Editor: Ben T. Nohara

Copyright © 2012 Soheli Rana et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Nowadays bootstrap techniques are used for data analysis in many other fields like engineering, physics, meteorology, medicine, biology, and chemistry. In this paper, the robustness of Wu (1986) and Liu (1988)'s Wild Bootstrap techniques is examined. The empirical evidences indicate that these techniques yield efficient estimates in the presence of heteroscedasticity problem. However, in the presence of outliers, these estimates are no longer efficient. To remedy this problem, we propose a Robust Wild Bootstrap for stabilizing the variance of the regression estimates where heteroscedasticity and outliers occur at the same time. The proposed method is based on the weighted residuals which incorporate the MM estimator, robust location and scale, and the bootstrap sampling scheme of Wu (1986) and Liu (1988). The results of this study show that the proposed method outperforms the existing ones in every respect.

## **1. Introduction**

Bootstrap technique was first proposed by Efron [1]. It is a computer intensive method that can replace theoretical formulation with extensive use of computer. The attractive feature of the bootstrap technique is that it does not rely on the normality or any other distributional assumptions and is able to estimate standard error of any complicated estimator without any theoretical calculations. These interesting properties of the bootstrap method have to be traded off with computational cost and time. There are considerable papers that deal with bootstrap methods in the literatures (see [2–5]). The classical bootstrap methods are known

to be a good general procedure for estimating a sampling distribution under the independent and identically distributed (i.i.d.) models. Let us consider a standard linear regression model:

$$Y = X\beta + \varepsilon, \quad (1.1)$$

where  $Y = (y_1, y_2, \dots, y_n)^T$ ,  $X = (x_1, x_2, \dots, x_n)^T$ , and  $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n)^T$ . In this equation  $\beta$  is a  $k \times 1$  vector of unknown parameters,  $Y$  is an  $n \times 1$  vector,  $X$  is an  $n \times k$  data matrix of full rank  $k \leq n$ , and  $\varepsilon$  is an  $n \times 1$  vector of unobservable random errors with  $E(\varepsilon) = 0$  and  $V(\varepsilon) = \sigma^2 I$ . In practice the i.i.d. set-up is often violated, as, for example, the homoscedastic assumption of  $\text{Var}(\varepsilon_i) = \sigma^2 I$  is often violated. Wu [6] proposed a weighted bootstrap technique which gives better performance under both the homoscedastic and heteroscedastic models. However, a better alternative approximation is developed by Liu [7] following the suggestions of Liu [7] and Beran [8]. This type of weighted bootstraps is called the *wild bootstrap* in the literature. Several attempts have been made to use the Wu and Liu wild bootstrap techniques to remedy the problem of heteroscedasticity (see [6, 7, 9, 10]).

Salibian-Barrera and Zamar [11] pointed out that the problem of classical bootstrap is that the proportion of outliers in the bootstrap sample might be greater than that of the original data. Hence, the entire inferential procedure of bootstrap would be erroneous in the presence of outliers. As an alternative, robust bootstrap technique has been drawn a greater attention to the statisticians (see [11–15]). However, not much work is devoted to bootstrap technique when both outliers and heteroscedasticity are present in a data. Those wild bootstrap techniques can only rectify the problem of heteroscedasticity and not resistant to outliers. Moreover, these procedures are based on the OLS estimate which is very sensitive to outliers. We introduce the classical wild bootstrap in Section 2. In Section 3, we discuss the newly proposed robust wild bootstrap methods. A numerical example and a simulation study are presented in Sections 4 and 5, respectively. The conclusion of the study is given in Section 6.

## 2. Wild Bootstrap Techniques

In regression analysis, the most popular and widely used bootstrap technique is the fixed- $x$  resampling or bootstrapping the residuals [2]. This bootstrapping procedure is based on the ordinary least squares (OLS) residuals summarized as follows.

*Step 1.* Fit a model  $y_i = f(x_i, \beta_{\text{ols}})$  by the OLS method to the original sample of observations to get  $\hat{\beta}_{\text{ols}}$  and hence the fitted model is  $\hat{y}_i = f(x_i, \hat{\beta}_{\text{ols}})$ .

*Step 2.* Compute the OLS residuals  $\hat{\varepsilon}_i = y_i - \hat{y}_i$  and each residual  $\hat{\varepsilon}_i$  has equal probability,  $1/n$ .

*Step 3.* Draw a random sample  $\varepsilon_1^*, \varepsilon_2^*, \dots, \varepsilon_n^*$  from  $\hat{\varepsilon}_i$  with simple random sampling with replacement and attached to  $\hat{y}_i$  for obtaining fixed- $x$  bootstrap values  $y_i^{*b}$  where  $y_i^{*b} = f(x_i, \hat{\beta}_{\text{ols}}) + \varepsilon_i^{*b}$ .

*Step 4.* Fit the OLS to the bootstrapped values  $y_i^{*b}$  on the fixed- $x$  to obtain  $\hat{\beta}_{\text{ols}}^{*b}$ .

*Step 5.* Repeat Steps 3 and 4 for  $B$  times to get  $\hat{\beta}_{\text{ols}}^{*b1}, \dots, \hat{\beta}_{\text{ols}}^{*bB}$  where  $B$  is the bootstrap replications.

We call this bootstrap scheme  $\text{Boot}_{\text{ols}}$  since it is based on the OLS method.

When heteroscedasticity is present in the data, the variances of the data are different and neither of these bootstrap schemes can yield efficient estimates of the parameters. Wu [6] showed that they are inconsistent and asymptotically biased under the heteroscedasticity. Wu [6] proposed a *wild bootstrap* (weighted bootstrap) that can be used to obtain the standard error which is asymptotically correct under heteroscedasticity of unknown form. Wu slightly modified Step 3 of the OLS bootstrap and kept the other steps unchanged. For each  $i$ , draw a value  $t_i^*$ , with replacement, from a distribution with zero mean and unit variance and attached to  $\hat{y}_i$  for obtaining fixed- $x$  bootstrap values  $y_i^{*b}$ , where  $y_i^{*b} = f(x_i, \hat{\beta}_{\text{ols}}) + t_i^* \hat{\varepsilon}_i / \sqrt{1 - h_{ii}}$  and  $h_{ii} = x_i^T (X^T X)^{-1} x_i$  is the  $i$ th leverage. Note that the variance of  $t_i^* \hat{\varepsilon}_i$  is not constant when the original errors are not homoscedastic. Therefore, this bootstrap scheme takes into consideration the nonconstancy of the error variances. As an alternative [6],  $t_i^*$  can be chosen, with replacement, from  $a_1, a_2, \dots, a_n$ , where

$$a_i = \frac{\hat{\varepsilon}_i - \bar{\hat{\varepsilon}}}{\sqrt{n^{-1} \sum_{i=1}^n (\hat{\varepsilon}_i - \bar{\hat{\varepsilon}})^2}} \quad (2.1)$$

with  $\bar{\hat{\varepsilon}} = n^{-1} \sum_{i=1}^n \hat{\varepsilon}_i$ . For a regression model with intercept term,  $\bar{\hat{\varepsilon}}$  approximately equals zero. This is nonparametric implementation of Wu's bootstrap since the resampling is done from the empirical distribution function of the (normalized) residuals. We call this method Wu's bootstrap and denote it by  $\text{Boot}_{\text{wu}}$ .

Following the idea of Wu [6], another wild bootstrap technique was proposed by Liu [7] in which  $t_i^*$  is randomly selected from a population that has third central moment equal to one with zero mean and unit variance. Such kind of selection is used to correct the skewness term in the Edgeworth expansion of the sampling distribution of  $I^T \hat{\beta}$ , where  $I$  is an  $n$ -vector of ones. Liu's bootstrap can be conducted by drawing random numbers  $t_i^*$  in the following two ways.

- (1)  $t_i^* = Z_i - E(Z_i)$ ,  $i = 1, 2, \dots, n$ , and  $Z_1, Z_2, \dots, Z_n$  are independently and identically distributed having density  $g_z(x) = [\alpha^\beta / (\beta - 1)!] x^{\beta-1} e^{-\alpha x} I_{(x>0)}$ , where  $\alpha = 2$  and  $\beta = 4$ .
- (2)  $t_i^* = H_i D_i - E(H_i)E(D_i)$ ,  $i = 1, 2, \dots, n$ , where  $H_1, H_2, \dots, H_n$  are independently and identically distributed normal distribution with mean  $(1/2)(\sqrt{17/6}) + \sqrt{1/6}$  and variance  $1/2$ .  $D_1, D_2, \dots, D_n$  are also independently and identically distributed normal distribution with mean  $(1/2)(\sqrt{17/6}) - \sqrt{1/6}$  and variance  $1/2$ .  $H_i$ 's and  $D_i$ 's are independent.

It is worth mentioning that selecting random numbers  $t_i^*$  by procedure 1 or procedure 2 of Liu [7] will produce third central moment equal to one. Following Cribari-Neto and Zarkos [16], we consider the second procedure of drawing the random sample  $t_i^*$ . We call this bootstrap scheme as  $\text{Boot}_{\text{liu}}$ .

### 3. Proposed Robust Wild Bootstrap Techniques

We have discussed the classical wild bootstrap procedures which are based on the OLS residuals. It is now evident that the OLS suffers a huge setback in the presence of outliers since

it has 0% breakdown [17]. Since the wild bootstrap samples are based on the OLS residuals, it is not resistant to outliers. Hence, in this article we propose to use the high-breakdown and high-efficiency robust MM estimator [18] to obtain the robust residuals. It is expected that for good data point, the residuals of the MM estimator are approximately the same as the OLS residuals. On the other hand, the residuals of the MM estimator would be larger for outlier observation. We assign weights to the MM residuals. The standardized residuals  $|\hat{\varepsilon}_i^{\text{MM}}|/\sigma_{\text{MM}}$  are computed, where  $\sigma_{\text{MM}}$  is the square root of the mean squares error of the residuals of the MM estimates (see [19]). Following the idea of Furno [20], weights equal to one and  $c/(|\hat{\varepsilon}_i^{\text{MM}}|/\sigma_{\text{MM}})$  are assigned to  $|\hat{\varepsilon}_i^{\text{MM}}|/\sigma_{\text{MM}} \leq c$  and  $|\hat{\varepsilon}_i^{\text{MM}}|/\sigma_{\text{MM}} > c$ , respectively, where  $c$  is an arbitrary constant which is chosen between 2 and 3. We multiply the new weights with the residuals of the MM estimates and the resultants are denoted by  $\hat{\varepsilon}_i^{\text{WMM}}$ . It is now expected that not only the residuals corresponding to the good data points but also the residuals corresponding to the bad data point of the MM residuals tend to be similar to the OLS residuals with no outliers. Based on the new weighted residuals  $\hat{\varepsilon}_i^{\text{WMM}}$ , we propose to robustify  $\text{Boot}_{\text{ols}}$ ,  $\text{Boot}_{\text{wu}}$ , and  $\text{Boot}_{\text{liu}}$ . We call the resulting robust bootstraps  $\text{RBoot}_{\text{ols}}$ ,  $\text{RBoot}_{\text{wu}}$ , and  $\text{RBoot}_{\text{liu}}$ .

We propose to replace the OLS residuals by  $\hat{\varepsilon}_i^{\text{WMM}}$  in Step 3 of the  $\text{Boot}_{\text{ols}}$ . That is, the bootstrap sample  $\varepsilon_1^*, \varepsilon_2^*, \dots, \varepsilon_n^*$  is drawn from  $\hat{\varepsilon}_i^{\text{WMM}}$  with simple random sampling and the other steps remain unchanged. We call this bootstrap scheme  $\text{Rboot}_{\text{ols}}$ . Now we will discuss the formulation of robust wild bootstrap based on Wu's procedures. The algorithm is summarized as follows.

*Step 1.* Fit a model  $y_i = x_i\beta + \varepsilon_i$  by the MM estimator to the original sample of observations to get the robust parameters  $\hat{\beta}_{\text{MM}}$  and hence the fitted model is  $\hat{y}_i = x_i\hat{\beta}_{\text{MM}}$ .

*Step 2.* Compute the residuals of the MM estimate  $\hat{\varepsilon}_i^{\text{MM}} = y_i - \hat{y}_i$ . Then assign weight to each residual,  $\hat{\varepsilon}_i^{\text{MM}}$ , such that the weight equals 1 if  $|\hat{\varepsilon}_i^{\text{MM}}|/\sigma_{\text{MM}} \leq c$  and equals  $c/(|\hat{\varepsilon}_i^{\text{MM}}|/\sigma_{\text{MM}})$  if  $|\hat{\varepsilon}_i^{\text{MM}}|/\sigma_{\text{MM}} > c$ .

*Step 3.* The final weighted residuals of the MM estimates denoted by  $\hat{\varepsilon}_i^{\text{WMM}}$  are formulated by multiplying the weights obtained in Step 2 with the residuals of the MM estimates. That is,  $\hat{\varepsilon}_i^{\text{WMM}} = 1 \times \hat{\varepsilon}_i^{\text{MM}}$  if the observation corresponds to good data point (no outliers) and  $\hat{\varepsilon}_i^{\text{WMM}} = c/(|\hat{\varepsilon}_i^{\text{MM}}|/\sigma_{\text{MM}}) \times \hat{\varepsilon}_i^{\text{MM}}$  if the observation corresponds to outliers.

*Step 4.* Construct a bootstrap sample  $(y_i^*, X)$ , where

$$y_i^* = x_i\hat{\beta}_{\text{MM}} + \frac{t_i^*\hat{\varepsilon}_i^{\text{WMM}}}{(1 - h_{ii})}, \quad (3.1)$$

and  $t^*$  is a random sample following Wu [6] procedure.

*Step 5.* The OLS procedure is then applied to the bootstrap sample  $(y_i^*, X)$ , and the resultant estimate is denoted by  ${}^R\hat{\beta}^* = (X^T X)^{-1} X^T y^*$ . Here, the robust estimates are very reliable since the bootstrap sample is constructed based on the robust weighted residuals,  $\hat{\varepsilon}_i^{\text{WMM}}$ .

*Step 6.* Repeat Steps 4 and 5 for  $B$  times, where  $B$  is the bootstrap replications.

As discussed earlier, in the classical scheme of Wu's bootstrap, the quantity  $t_i^*$  is drawn from a population that has mean zero and variance equal to one or,  $t_i^*$  can be drawn from normalized residuals  $a_1, a_2, \dots, a_n$ , that is,

$$a_i = \frac{\hat{\varepsilon}_i - \bar{\hat{\varepsilon}}}{\sqrt{n^{-1} \sum_{i=1}^n (\hat{\varepsilon}_i - \bar{\hat{\varepsilon}})^2}}. \quad (3.2)$$

However, following Maronna et al. [21], we suggest computing the robust normalized residuals based on *median* and *normalized median absolute deviations* (NMADs) instead of *mean* and *standard deviation* which are not robust. Thus,

$${}^R a_i = \frac{\hat{\varepsilon}_i^{\text{WMM}} - \text{median}(\hat{\varepsilon}_i^{\text{WMM}})}{\text{NMAD}_{\text{norm}}(\hat{\varepsilon}_i^{\text{WMM}})}, \quad (3.3)$$

where  $\text{NMAD}_{\text{norm}} = \text{median}\{|\hat{\varepsilon}_i^{\text{WMM}} - \text{median}(\hat{\varepsilon}_i^{\text{WMM}})|\} / 0.6745$ . We call this proposed robust nonparametric bootstrap as  $\text{RBoot}_{\text{wu}}$ .

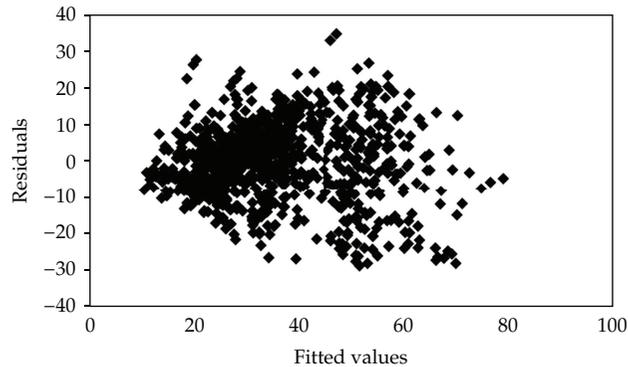
In this paper we also want to robustify the wild bootstrap based on the Liu [7] algorithm. It is important to note that the only difference between the Wu and Liu implementation of wild bootstrap is the choice of the random sample  $t_i^*$ . In the proposed robust bootstrap based on the Liu wild bootstrap, we choose the random sample  $t_i^*$  exactly the same manner as the classical Liu bootstrap. We call this bootstrap scheme as  $\text{RBoot}_{\text{liu}}$ .

#### 4. Numerical Example

In this section, a numerical example is presented to assess the performance of the robust wild bootstrap methods. In order to compare the robustness of the classical and robust wild bootstrap in the presence of outliers, the Concrete Compressive Strength data is taken from Yeh [22]. Concrete is the most important material in civil engineering. The concrete compressive strength is a function of the eight output such as cement ( $\text{Kg}/\text{m}^3$ ), blast furnace slag ( $\text{Kg}/\text{m}^3$ ), fly ash ( $\text{Kg}/\text{m}^3$ ), water ( $\text{Kg}/\text{m}^3$ ), superplasticizer ( $\text{Kg}/\text{m}^3$ ), coarse aggregate ( $\text{Kg}/\text{m}^3$ ), fine aggregate ( $\text{Kg}/\text{m}^3$ ), and age of testing (days). The residuals versus fitted values are plotted in Figure 1 that show a funnel shape suggesting a heterogeneous error variances for the data (see [19]).

We checked whether this data set contain any outliers or not by using Least trimmed of Squares (LTSs) residuals. It is found that 61 observations (about 6% of the sample of size 1030) appear to be outliers. The robust and non-robust (Classical) wild bootstrap methods were then applied to the data by considering two types of situations, namely, the data with outliers and data without outliers (omitted the outlying data points). The results are based on 500 bootstraps and are given in Table 1.

The standard errors of the parameter estimates from robust and nonrobust wild bootstrap methods are exhibited in Table 1. The average standard errors of the parameter estimates are also shown. When there are no outliers, the standard errors of the classical wild bootstrap are reasonably closed to the standard errors of the robust wild bootstrap. It is interesting to note that the classical wild bootstrap methods provide larger standard errors compared to the wild bootstrap methods when outliers are present in the data.



**Figure 1:** Residuals versus Fitted values plot of Concrete Compressive Strength data.

**Table 1:** Wild bootstrap standard errors of the parameters for the Concrete Compressive Strength data.

Standard error (Se)	Classical wild bootstrap			Robust wild bootstrap		
	Boot <sub>ols</sub>	Boot <sub>wu</sub>	Boot <sub>liu</sub>	RBoot <sub>ols</sub>	RBoot <sub>wu</sub>	RBoot <sub>liu</sub>
Data with outliers						
Intercept	29.0037	27.0670	19.4559	27.1955	19.2343	16.5458
Cement	0.00899	0.00874	0.00612	0.00857	0.00641	0.00503
Blast Furnace Slag	0.01090	0.01022	0.00740	0.01047	0.00757	0.00605
Fly Ash	0.01283	0.01296	0.00864	0.01180	0.00971	0.00694
Water	0.04584	0.04129	0.02914	0.04301	0.02986	0.02566
Superplasticizer	0.09824	0.09564	0.06640	0.10253	0.07104	0.05706
Coarse Aggregate	0.01019	0.00956	0.00687	0.00928	0.00666	0.00559
Fine Aggregate	0.01113	0.01056	0.00772	0.01062	0.00772	0.00653
Age	0.00796	0.00550	0.00556	0.00301	0.00419	0.00175
<i>Average Se</i>	3.24553	3.02905	2.17708	3.04387	2.15305	1.85116
Data without outliers						
Intercept	26.3944	20.6483	17.5582	25.5838	20.7367	16.9330
Cement	0.00856	0.00687	0.00554	0.00810	0.00666	0.00531
Blast Furnace Slag	0.01000	0.00805	0.00679	0.00984	0.00785	0.00648
Fly Ash	0.01188	0.01014	0.00757	0.01143	0.01022	0.00736
Water	0.04068	0.03051	0.02646	0.03837	0.03080	0.02580
Superplasticizer	0.09384	0.07151	0.05940	0.09498	0.07146	0.05974
Coarse Aggregate	0.00909	0.00752	0.00602	0.00903	0.00739	0.00575
Fine Aggregate	0.01047	0.00823	0.00699	0.01025	0.00826	0.00679
Age	0.00992	0.00842	0.00690	0.01016	0.00837	0.00619
<i>Average Se</i>	2.95431	2.31106	1.96487	2.86400	2.32086	1.89516

We cannot make a final conclusion yet, just by observing the results of the real data, but a reasonable interpretation up to this stage is that the classical wild bootstrap is affected by outliers.

## 5. Simulation Study

In this section, the performances of the proposed robust wild bootstrap estimators are evaluated based on a simulation study. At first we generate some artificial data to see

the performance of proposed bootstrap techniques. The final investigation of the performance of the proposed estimators is verified by the simulation approach on bootstrap samples.

### 5.1. Artificial Data

We follow the data generation technique of Cribari-Neto and Zarkos [16] and MacKinnon and White [23]. The design of this experiment involves a linear model with two covariates:

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \sigma_i \varepsilon_i. \quad (5.1)$$

We consider the sample sizes  $n = 20, 60, 100$ . For  $n = 20$  the covariate values  $x_{1i}$  were obtained from  $U(0,1)$  and the covariate values  $x_{2i}$  were obtained from  $N(0,1)$ . These observations were replicated three and five times for creating the sample of size  $n = 60$  and  $n = 100$ , respectively. The data generation was performed using  $\beta_0 = \beta_1 = \beta_2 = 1$ . For all  $i$  under the homoscedasticity,  $\sigma_i = 1$ . However, the main interest here is to find the heteroscedastic model. In this respect, we create a heteroscedastic generating mechanism following Cribari-Neto [24]'s work, where

$$\sigma_i^2 = \exp(3.2x_{1i}). \quad (5.2)$$

The degree of heteroscedasticity was measured by

$$\hat{\rho} = \frac{\max(\sigma_i^2)}{\min(\sigma_i^2)}, \quad i = 1, 2, \dots, n, \quad (5.3)$$

The degree of heteroscedasticity remains constant for different sample sizes since the covariate values are replicated for generating different sample sizes. In our study the degree of heterogeneity was approximately  $\hat{\rho} = 4$ . We focus on the situation where regression design would include outliers. To generate a certain percentages of outliers in Model (5.1), some i.i.d. normal errors  $\varepsilon_i$ 's were replaced by  $N(5, 10)$ . Hence the contaminated heteroscedastic model becomes

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \sigma_i \varepsilon_{i(\text{cont})}, \quad (5.4)$$

where  $\varepsilon_{i(\text{cont.})} = \alpha N(0, 1) + (1 - \alpha)N(5, 10)$  and  $\alpha$  is chosen according to level of percentage of outliers. In this study we choose the 5%, 10%, 15%, and 20% outliers in the model; that is,  $\alpha$  is 0.95, 0.80, 0.85, and 0.80, respectively. Now for each sample size, the OLS, the classical, and the proposed robust wild bootstrap were then applied to the data. The replications of the bootstrap were 500 in each model for the different sample sizes. It is noteworthy that the bootstrap is extremely computer intensive, and S-plus programming language was used for computing the bootstrap estimates.

The wild bootstrap standard errors of the estimates for different sample sizes and different percentage of contaminations are computed. The bootstrap standard errors of  $\text{Boot}_{\text{ols}}$ ,

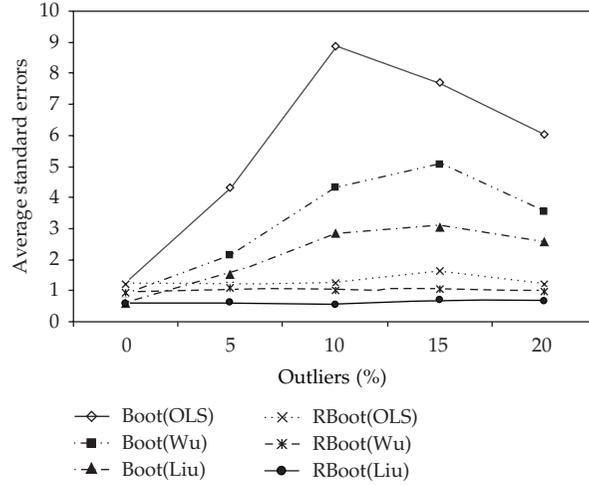


Figure 2: The average effect of outliers on standard errors of parameters for sample size  $n = 20$ .

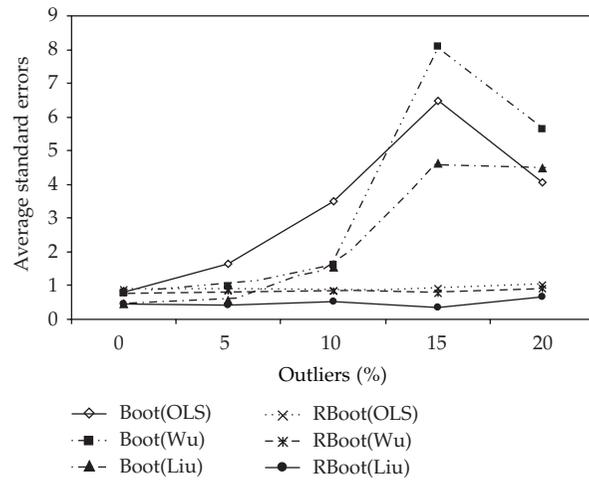


Figure 3: The average effect of outliers on standard errors of parameters for sample size  $n = 60$ .

$\text{Boot}_{\text{Wu}}$ , and  $\text{Boot}_{\text{Liu}}$  are obtained by taking the square root to the main diagonal of the covariance matrix:

$$(B-1)^{-1} \sum_{b=1}^B (\hat{\beta}^{*b} - \bar{\beta}^*) (\hat{\beta}^{*b} - \bar{\beta}^*)^T, \quad (5.5)$$

where  $\bar{\beta}^* = (1/B) \sum_{b=1}^B \hat{\beta}^{*b}$ . On the other hand, the bootstrap standard errors of  $\text{RBoot}_{\text{ols}}$ ,  $\text{RBoot}_{\text{Wu}}$ , and  $\text{RBoot}_{\text{Liu}}$  are obtained by taking the square root to the main diagonal of the covariance matrix as given in (5.5); the only essential difference is, however, we replace the usual bootstrap estimates by the robust bootstrap estimates.

The influences of outliers on the standard errors of the estimates are visible in Figures 2, 3, and 4. In these plots, the average standard errors of the parameters estimates are plotted

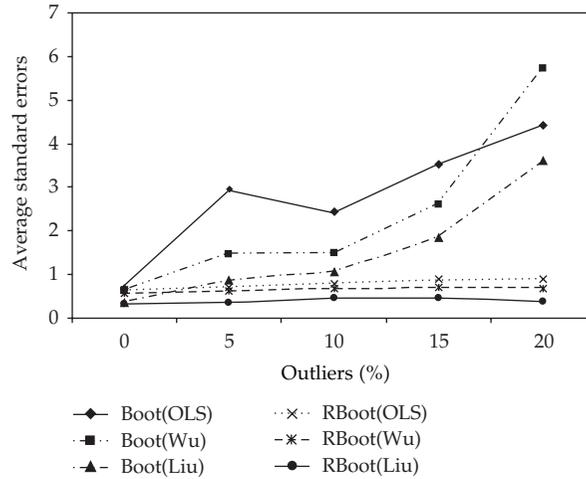


Figure 4: The average effect of outliers on standard errors of parameters for sample size  $n = 100$ .

at different levels of outliers for different bootstrap methods. The results presented in Figures 2–4 show that the performances of the wild bootstrap estimates are fairly close to the classical estimates at the 0% level of contamination. It emerges that the average standard errors of the  $RBoot_{wu}$  and  $RBoot_{liu}$  are closer to the average standard errors of the classical  $Boot_{wu}$  and  $Boot_{liu}$ , respectively, in “clean” data, regardless of the percentage of outliers. However, at the 5%, 10%, 15%, and 20% levels of contaminations, the classical standard errors of the bootstrap estimates become unduly large. On the contrary, it is interesting to see that not much influence is visible for the robust wild bootstrap techniques of  $RBoot_{wu}$  and  $RBoot_{liu}$ , at the different percentage levels of outliers. It is also observed that the performance of  $RBoot_{liu}$  is the best overall followed by  $RBoot_{wu}$ .

### 5.2. Simulation Approach on Bootstrap Sample

In the previous section, we used artificial data sets for different sample sizes. Now we would like to investigate the performances of different bootstrap estimators where data sets are generated by Monte Carlo simulations. Let us consider a heteroscedastic model which is given by

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \sigma_i \varepsilon_i. \tag{5.6}$$

The covariate values of  $x_{1i}$  and  $x_{2i}$  are generated from  $U(0,1)$  for sample sizes 20, 60, and 100. We have also considered  $\beta_0 = \beta_1 = \beta_2 = 1$  as the true parameters in this model and the heteroscedasticity generating function was  $\sigma_i^2 = \exp(0.4x_{1i} + 0.4x_{2i})$ . In this study the level of heteroscedasticity is set as  $\hat{\rho} = \max(\sigma_i^2) / \min(\sigma_i^2) = 4$ .

In each simulation run and for the different sample size,  $\varepsilon_i$ 's were generated from  $N(0, 1)$  for the data with no outliers. However, for generating the 5% and 10% outliers, the 95% and 90% of  $\varepsilon_i$ 's were generated from  $N(0, 1)$  and the 5% and 10% were generated from  $N(0, 20)$ . It is worth mentioning that although such simulations are extremely computer intensive, the simulation for each sample size entails a total of 250000 replications with 500 replications and

**Table 2:** Biasness measures of the non-robust and robust wild bootstrap.

% outliers	Coeff.	Boot <sub>ols</sub>	Boot <sub>wu</sub>	Boot <sub>liu</sub>	RBoot <sub>wu</sub>	RBoot <sub>liu</sub>
Sample Size $n = 20$						
0%	$\beta_0$	0.0115	-0.0246	-0.0070	-0.0701	-0.0799
	$\beta_1$	-0.5604	-0.5005	-0.4813	-0.3338	-0.2796
	$\beta_2$	0.3753	0.3526	0.2968	0.3576	0.3348
	Mean	0.3157	0.2926	0.2617	0.2538	0.2314
5%	$\beta_0$	-0.9419	-1.1981	-1.3702	0.0357	0.0821
	$\beta_1$	2.9588	2.8570	3.1427	-0.3130	-0.3450
	$\beta_2$	0.8742	1.3083	1.4716	0.4164	0.4077
	Mean	1.5916	1.7878	1.9948	0.2550	0.2783
10%	$\beta_0$	-5.6740	-5.3968	-5.5439	-0.2166	-0.215
	$\beta_1$	6.7826	6.9083	6.9500	0.2450	0.2714
	$\beta_2$	7.4401	6.8714	7.0577	0.4190	0.4073
	Mean	6.6322	6.3921	6.5172	0.2935	0.2979
Sample Size $n = 60$						
0%	$\beta_0$	0.0197	0.0612	0.0458	-0.0123	-0.0056
	$\beta_1$	0.0159	-0.0189	0.0218	0.0400	0.0345
	$\beta_2$	0.0080	-0.0209	-0.0247	0.0462	0.0174
	Mean	0.0145	0.0337	0.0308	0.0328	0.0192
5%	$\beta_0$	-0.1451	-0.1466	-0.0648	-0.0190	-0.0045
	$\beta_1$	0.4342	0.4011	0.2954	0.0921	0.0921
	$\beta_2$	0.1110	0.1452	0.0526	0.0358	0.0073
	Mean	0.2301	0.2310	0.1376	0.0490	0.0346
10%	$\beta_0$	-0.7390	-0.7896	-0.7478	0.0164	-0.0033
	$\beta_1$	0.9916	0.9875	0.96302	0.0793	0.0794
	$\beta_2$	0.8974	1.0292	1.0004	0.0181	0.0005
	Mean	0.8760	0.9354	0.9037	0.0379	0.0277
Sample Size $n = 100$						
0%	$\beta_0$	0.0226	0.0216	0.0240	-0.0505	-0.0554
	$\beta_1$	-0.1086	-0.1037	-0.1099	-0.0514	-0.0457
	$\beta_2$	-0.0448	-0.0422	-0.0433	0.0371	0.0387
	Mean	0.0587	0.0558	0.0591	0.0463	0.0466
5%	$\beta_0$	0.1218	0.1396	0.1095	0.0205	0.0048
	$\beta_1$	-0.1854	-0.3399	-0.2944	-0.1880	-0.1699
	$\beta_2$	-0.1944	-0.1522	-0.1174	-0.0163	-0.0038
	Mean	0.1672	0.2106	0.1738	0.0749	0.0595
10%	$\beta_0$	0.7546	0.6492	0.8318	-0.2921	-0.2748
	$\beta_1$	-1.1835	-1.0402	-1.1809	0.3250	0.3294
	$\beta_2$	-0.8359	-0.7686	-1.0436	0.2800	0.2558
	Mean	0.9247	0.8193	1.0187	0.2990	0.2867

500 bootstrap samples each. This simulation procedure was performed following the design of Cribari-Neto and Zarkos [16] and Furno [20].

The simulation results for the different bootstrap methods are presented in Tables 2–4. Table 2 shows the biasness measures of the non-robust and robust wild bootstrap techniques. It is observed that for the different sample sizes, the biasness of the Boot<sub>ols</sub>, the Boot<sub>liu</sub>, and

**Table 3:** Standard errors of the non-robust and robust wild bootstrap.

% outliers	Coeff.	Boot <sub>ols</sub>	Boot <sub>wu</sub>	Boot <sub>liu</sub>	RBoot <sub>wu</sub>	RBoot <sub>liu</sub>
Sample Size $n = 20$						
0%	$\beta_0$	1.1491	1.0414	0.9540	1.0568	0.9774
	$\beta_1$	1.2106	1.3220	1.0294	1.2739	1.1894
	$\beta_2$	1.6257	1.4866	1.3845	1.4827	1.3737
	Mean	1.3285	1.2833	1.1226	1.2711	1.1802
5%	$\beta_0$	4.4748	4.4700	3.2555	0.8503	0.9795
	$\beta_1$	6.7426	8.0992	6.5537	1.1291	1.2546
	$\beta_2$	5.4343	4.9812	3.2680	1.4656	1.6396
	Mean	5.5506	5.8501	4.3591	1.1483	1.2912
10%	$\beta_0$	11.2898	12.5354	10.4426	1.1493	0.9913
	$\beta_1$	12.9386	15.0904	12.5433	1.3296	1.15814
	$\beta_2$	15.5132	16.8739	14.1209	1.4812	1.1552
	Mean	13.2472	14.8332	12.3689	1.3200	1.1015
Sample Size $n = 60$						
0%	$\beta_0$	0.7500	0.7296	0.6483	0.7455	0.6800
	$\beta_1$	0.6632	0.6711	0.5644	0.7213	0.6072
	$\beta_2$	0.9779	0.9865	0.8358	0.9979	0.8787
	Mean	0.7970	0.7957	0.6828	0.8216	0.7220
5%	$\beta_0$	2.9250	2.3879	2.0964	0.7677	0.7043
	$\beta_1$	4.6423	5.3209	4.6852	0.7659	0.7000
	$\beta_2$	3.4306	2.1242	1.7946	0.9833	0.9184
	Mean	3.6660	3.2777	2.8587	0.8390	0.7742
10%	$\beta_0$	6.3512	7.9335	6.5333	0.8164	0.7583
	$\beta_1$	6.7028	8.5494	6.9565	0.8880	0.8150
	$\beta_2$	8.7297	10.9102	9.1101	1.1144	1.0058
	Mean	7.2612	9.1310	7.5333	0.9396	0.8597
Sample Size $n = 100$						
0%	$\beta_0$	0.4658	0.4900	0.3778	0.4326	0.3500
	$\beta_1$	0.6340	0.6531	0.5712	0.6441	0.5901
	$\beta_2$	0.6355	0.6717	0.5055	0.5681	0.4619
	Mean	0.5784	0.6049	0.4848	0.5483	0.4673
5%	$\beta_0$	2.6526	1.9689	1.8733	0.4616	0.3887
	$\beta_1$	4.6667	4.7285	4.7007	0.6805	0.6310
	$\beta_2$	3.1597	1.5921	1.3564	0.5843	0.4947
	Mean	3.4930	2.7632	2.6435	0.5755	0.5048
10%	$\beta_0$	5.7190	7.2003	5.9257	0.4999	0.4379
	$\beta_1$	5.9250	7.4991	6.1414	0.5324	0.4687
	$\beta_2$	7.8683	9.8171	8.0801	0.7999	0.7220
	Mean	6.5041	8.1722	6.7157	0.6107	0.5429

the  $Boot_{wu}$  increases with the increase in the percentage of outliers. On the other hand, the  $RBoot_{wu}$ , and the  $RBoot_{liu}$  are slightly biased with the increase in the percentage of outliers. We can draw the same conclusion from the mean of the biasness of the estimates. The standard errors of the non-robust and robust wild bootstrap are presented in Table 3. It is observed that the standard errors of the classical bootstrap estimates increase with the increase

**Table 4:** Robustness measure of RMSE of the non-robust and robust wild bootstrap.

% outliers	Coeff.	Robustness measure of RMSE				
		Boot <sub>ols</sub>	Boot <sub>wu</sub>	Boot <sub>liu</sub>	RBoot <sub>wu</sub>	RBoot <sub>liu</sub>
Sample Size $n = 20$						
0%	$\beta_0$	—	110.3099	120.4542	102.0019	117.1796
	$\beta_1$	—	94.37567	117.3951	94.5929	109.1911
	$\beta_2$	—	109.20311	117.8290	101.1577	118.0043
	Mean	—	104.6296	118.5594	99.25083	114.7917
5%	$\beta_0$	25.13074	24.83232	32.53479	109.03777	135.0219
	$\beta_1$	18.11883	15.53409	18.35560	103.90257	113.8576
	$\beta_2$	30.31304	32.39676	46.55294	94.03987	109.5074
	Mean	24.52087	24.25439	32.48111	102.3267	119.4623
10%	$\beta_0$	9.0949	8.4203	9.7199	98.2525	106.2831
	$\beta_1$	9.1324	8.0386	9.3035	98.6716	105.1563
	$\beta_2$	9.6977	9.1578	10.5692	108.3857	109.2154
	Mean	9.3083	8.5389	9.8642	101.7699	120.5487
Sample Size $n = 60$						
0%	$\beta_0$	—	102.47133	115.4533	100.52476	110.3378
	$\beta_1$	—	98.81731	117.4509	91.85584	109.0760
	$\beta_2$	—	99.10685	116.9525	97.81249	111.2705
	Mean	—	100.1318	116.6189	96.73103	110.2281
5%	$\beta_0$	25.62126	31.36313	35.77534	92.89488	106.5277
	$\beta_1$	14.22979	12.43393	14.13294	80.01583	93.96979
	$\beta_2$	28.49122	45.93069	54.47049	90.03794	106.4720
	Mean	22.78076	29.90925	34.79292	87.64955	102.3232
10%	$\beta_0$	11.7351	9.4115	11.4105	91.900	98.8490
	$\beta_1$	9.7919	7.7093	9.4474	74.4197	81.2251
	$\beta_2$	11.1437	8.9234	10.6705	87.7518	97.1719
	Mean	10.89023	8.6814	10.50947	84.6905	92.41533
Sample Size $n = 100$						
0%	$\beta_0$	—	95.07261	123.1937	100.26721	131.6080
	$\beta_1$	—	97.26561	110.5779	97.95562	108.6698
	$\beta_2$	—	94.65821	125.5736	100.14950	137.4324
	Mean	—	95.66548	119.7817	99.45744	125.9034
5%	$\beta_0$	17.5633	23.6278	24.8537	89.4943	119.9566
	$\beta_1$	13.7733	13.5688	13.6576	87.9910	98.4320
	$\beta_2$	20.1261	39.8355	46.7975	90.0007	128.7815
	Mean	17.1542	25.6774	28.4363	89.1620	115.7234
10%	$\beta_0$	8.0848	6.4510	7.7940	80.5426	90.1938
	$\beta_1$	10.6465	8.4965	10.2857	103.1180	112.2725
	$\beta_2$	8.0521	6.4702	7.8202	75.1767	83.1701
	Mean	8.9278	7.1393	8.6333	86.2791	95.2121

in the percentage of outliers for different sample sizes. However, the robust bootstrap estimates are slightly affected by these outliers. By investigating the average standard errors of the estimates, it is also observed that the robust wild bootstrap techniques provide less standard error of the estimates in the presence of outliers. Finally, the robustness of different bootstrapping techniques are evaluated based on robustness measures defined in (5.5). Here the percentage robustness measure, that is, the ratio of the RMSEs of the estimators compared

with the RMSEs of the OLS estimator for good data is presented in Table 4. From this table we see that the OLS and the classical bootstrap methods perform poorly. In the presence of outliers, the efficiency of the classical bootstrap estimates is very low. However, the efficiency of the robust bootstrap estimates is fairly closed to 100%.

## 6. Concluding Remarks

This paper examines the performance of classical wild bootstrap techniques which were proposed by Wu [6] and Liu [7] in the presence of heteroscedasticity and outliers. Both the artificial example and simulation study show that the classical bootstrap techniques perform poorly in the presence of outliers in the heteroscedastic model although they perform superbly for “clean” data. We attempt to robustify those classical bootstrap techniques to gain better efficiency in the presence of outliers. The numerical results show that the newly proposed robust wild bootstrap techniques, namely, the  $R\text{Boot}_{\text{wu}}$  and  $R\text{Boot}_{\text{liu}}$  outperform the classical wild bootstrap techniques when both outliers and heteroscedasticity are present in the data.  $R\text{Boot}_{\text{liu}}$  performs slightly better than  $R\text{Boot}_{\text{wu}}$ . Another advantage of using the  $R\text{Boot}_{\text{wu}}$  and the  $R\text{Boot}_{\text{liu}}$  is that no diagnosis for the data is required before the application of these methods.

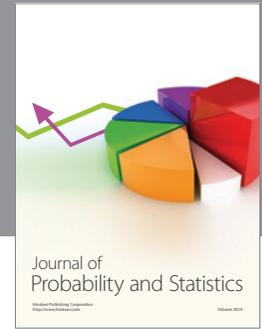
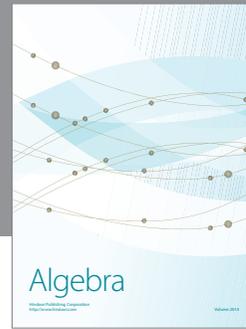
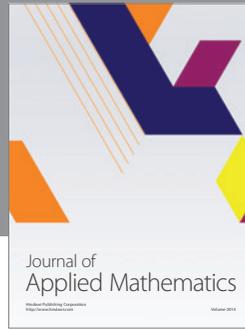
## Acknowledgment

The authors are grateful to the referees for valuable suggestions and comments that help them to improve the paper.

## References

- [1] B. Efron, “Bootstrap methods: another look at the jackknife,” *The Annals of Statistics*, vol. 7, no. 1, pp. 1–26, 1979.
- [2] B. Efron and R. Tibshirani, “Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy,” *Statistical Science*, vol. 1, no. 1, pp. 54–77, 1986.
- [3] B. Efron, “Better bootstrap confidence intervals,” *Journal of the American Statistical Association*, vol. 82, no. 397, pp. 171–200, 1987.
- [4] B. Efron and R. Tibshirani, *An Introduction to the Bootstrap*, CRC Press, 6th edition, 1993.
- [5] H. Midi, “Bootstrap methods in a class of non-linear regression models,” *Pertanika Journal of Science and Technology*, vol. 8, pp. 175–189, 2002.
- [6] C.-F. J. Wu, “Jackknife, bootstrap and other resampling methods in regression analysis,” *The Annals of Statistics*, vol. 14, no. 4, pp. 1261–1350, 1986.
- [7] R. Y. Liu, “Bootstrap procedures under some non-i.i.d. models,” *The Annals of Statistics*, vol. 16, no. 4, pp. 1696–1708, 1988.
- [8] R. Beran, “Prepivoting test statistics: a bootstrap view of asymptotic refinements,” *Journal of the American Statistical Association*, vol. 83, no. 403, pp. 687–697, 1988.
- [9] E. Mammen, “Bootstrap and wild bootstrap for high-dimensional linear models,” *The Annals of Statistics*, vol. 21, no. 1, pp. 255–285, 1993.
- [10] R. Davidson and E. Flachaire, “The wild bootstrap, tamed at last,” *Working Paper IER#1000*, Queen’s University, 2001.
- [11] M. Salibian-Barrera and R. H. Zamar, “Bootstrapping robust estimates of regression,” *The Annals of Statistics*, vol. 30, no. 2, pp. 556–582, 2002.
- [12] G. Willems and S. Van Aelst, “Fast and robust bootstrap for LTS,” *Computational Statistics & Data Analysis*, vol. 48, no. 4, pp. 703–715, 2005.
- [13] A. H. M. R. Imon and M. M. Ali, “Bootstrapping regression residuals,” *Journal of Korean Data and Information Science Society*, vol. 16, pp. 665–682, 2005.

- [14] M. Salibián-Barrera, S. Van Aelst, and G. Willems, "Fast and robust bootstrap," *Statistical Methods & Applications*, vol. 17, no. 1, pp. 41–71, 2008.
- [15] M. R. Norazan, H. Midi, and A. H. M. R. Imon, "Estimating regression coefficients using weighted bootstrap with probability," *WSEAS Transactions on Mathematics*, vol. 8, no. 7, pp. 362–371, 2009.
- [16] F. Cribari-Neto and S. G. Zarkos, "Bootstrap methods for heteroskedastic regression models: evidence on estimation and testing," *Econometric Reviews*, vol. 18, pp. 211–228, 1999.
- [17] P. J. Rousseeuw and A. M. Leroy, *Robust Regression and Outlier Detection*, Wiley Series in Probability and Mathematical Statistics: Applied Probability and Statistics, John Wiley & Sons, New York, NY, USA, 1987.
- [18] V. J. Yohai, "High breakdown-point and high efficiency robust estimates for regression," *The Annals of Statistics*, vol. 15, no. 2, pp. 642–656, 1987.
- [19] A. H. M. R. Imon, "Deletion residuals in the detection of heterogeneity of variances in linear regression," *Journal of Applied Statistics*, vol. 36, no. 3-4, pp. 347–358, 2009.
- [20] M. Furno, "A robust heteroskedasticity consistent covariance matrix estimator," *Statistics*, vol. 30, no. 3, pp. 201–219, 1997.
- [21] R. A. Maronna, R. D. Martin, and V. J. Yohai, *Robust Statistics: Theory and Methods*, Wiley Series in Probability and Statistics, John Wiley & Sons, Chichester, UK, 2006.
- [22] I.-C. Yeh, "Modeling of strength of high-performance concrete using artificial neural networks," *Cement and Concrete Research*, vol. 28, no. 12, pp. 1797–1808, 1998.
- [23] J. G. MacKinnon and H. White, "Some heteroskedasticity-consistent covariance matrix estimators with improved finite sample properties," *Journal of Econometrics*, vol. 29, no. 3, pp. 305–325, 1985.
- [24] F. Cribari-Neto, "Asymptotic inference under heteroskedasticity of unknown form," *Computational Statistics & Data Analysis*, vol. 45, no. 2, pp. 215–233, 2004.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

